DPS

# Designing Perimeter Security

Version 1.0

## Student Guide

Cisco Systems has more than 200 offices in the following countries and regions. Addresses, phone numbers, and fax numbers are listed on the Cisco Web site at www.cisco.com/go/offices.

Argentina • Australia • Austria • Belgium • Brazil • Bulgaria • Canada • Chile • China PRC • Colombia • Costa Rica Croatia • Czech Republic • Denmark • Dubai, UAE • Finland • France • Germany • Greece • Hong Kong SAR • Hungary India • Indonesia • Ireland • Israel • Italy • Japan • Korea • Luxembourg • Malaysia • Mexico • The Netherlands New Zealand • Norway • Peru • Philippines • Poland • Portugal • Puerto Rico • Romania • Russia • Saudi Arabia Scotland • Singapore • Slovakia • Slovenia • South Africa • Spain • Sweden • Switzerland • Taiwan • Thailand • Turkey Ukraine • United Kingdom • United States • Venezuela • Vietnam • Zimbabwe

Printed in the USA

# Table of Contents

# Course Introduction

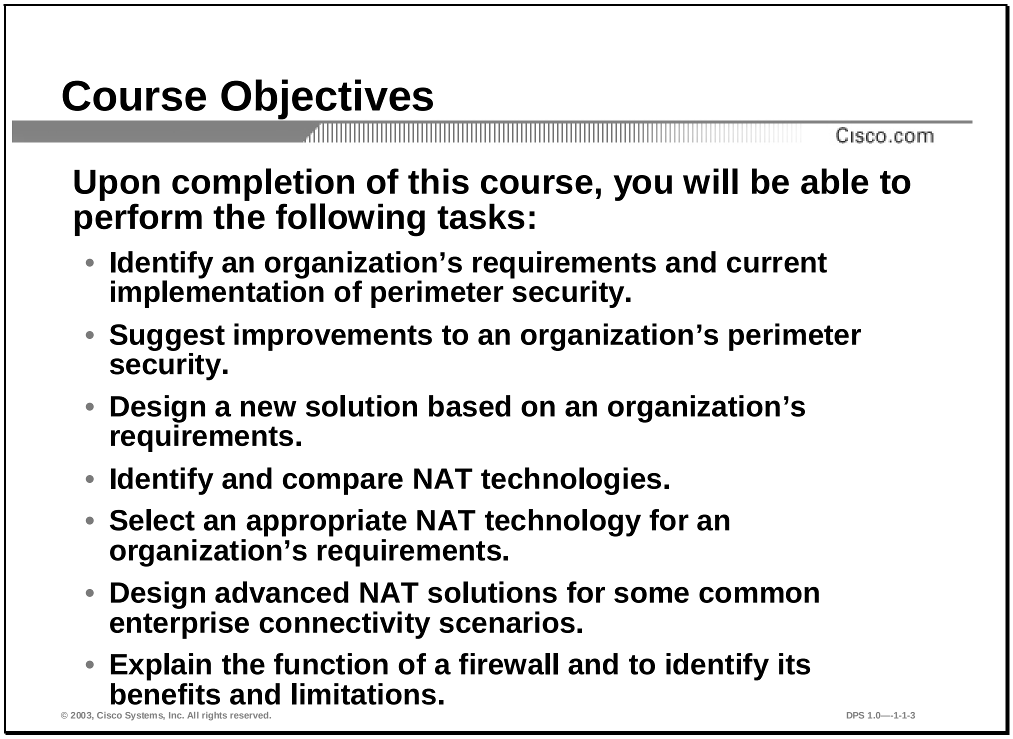
## Overview

This chapter includes the following topics:

- Course objectives
- Course agenda
- Participant responsibilities
- General administration
- Graphic symbols
- Participant introductions
- Cisco security career certifications

# Course Objectives

This section introduces the course and the course objectives.

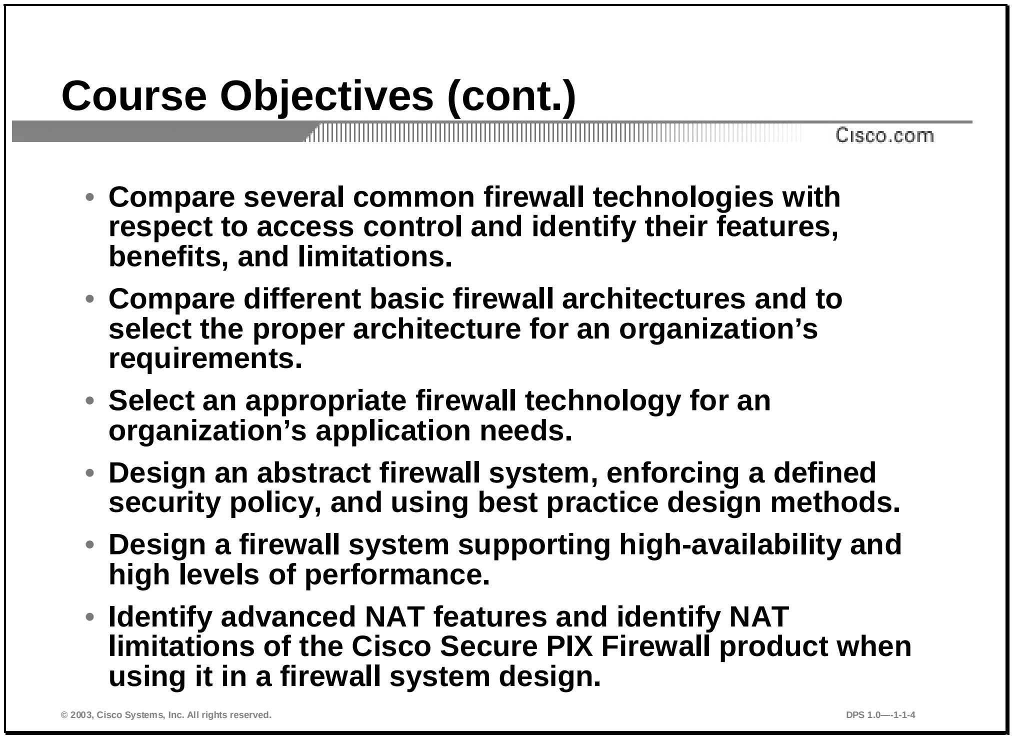


## Course Objectives

**Upon completion of this course, you will be able to perform the following tasks:**

- **Identify an organization’s requirements and current implementation of perimeter security.**
- **Suggest improvements to an organization’s perimeter security.**
- **Design a new solution based on an organization’s requirements.**
- **Identify and compare NAT technologies.**
- **Select an appropriate NAT technology for an organization’s requirements.**
- **Design advanced NAT solutions for some common enterprise connectivity scenarios.**
- **Explain the function of a firewall and to identify its benefits and limitations.**

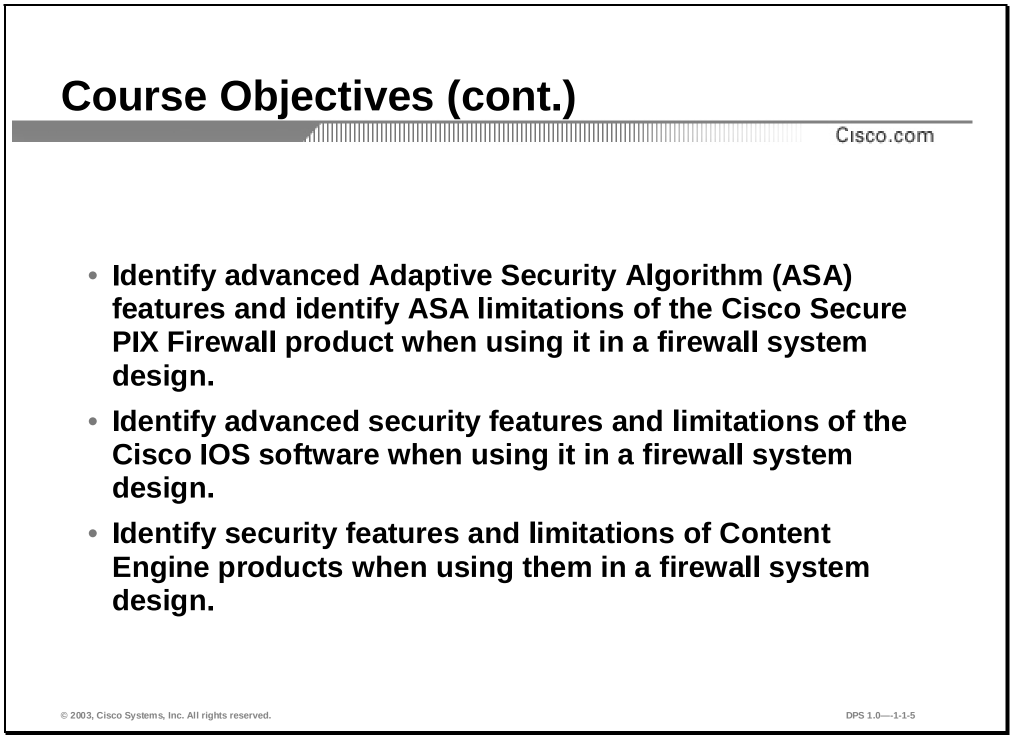
© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—-1-1-3



## Course Objectives (cont.)

- **Compare several common firewall technologies with respect to access control and identify their features, benefits, and limitations.**
- **Compare different basic firewall architectures and to select the proper architecture for an organization’s requirements.**
- **Select an appropriate firewall technology for an organization’s application needs.**
- **Design an abstract firewall system, enforcing a defined security policy, and using best practice design methods.**
- **Design a firewall system supporting high-availability and high levels of performance.**
- **Identify advanced NAT features and identify NAT limitations of the Cisco Secure PIX Firewall product when using it in a firewall system design.**

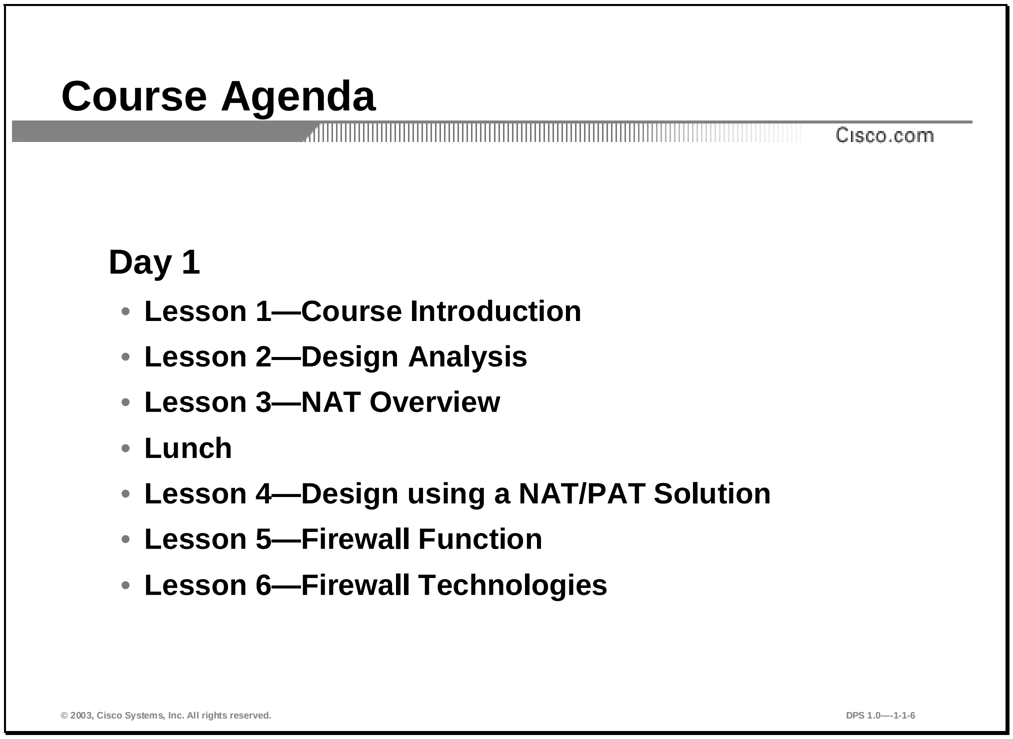
© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—-1-1-4



## Course Objectives (cont.)

- Identify advanced Adaptive Security Algorithm (ASA) features and identify ASA limitations of the Cisco Secure PIX Firewall product when using it in a firewall system design.
- Identify advanced security features and limitations of the Cisco IOS software when using it in a firewall system design.
- Identify security features and limitations of Content Engine products when using them in a firewall system design.

DPS 1.0—-1-1-5

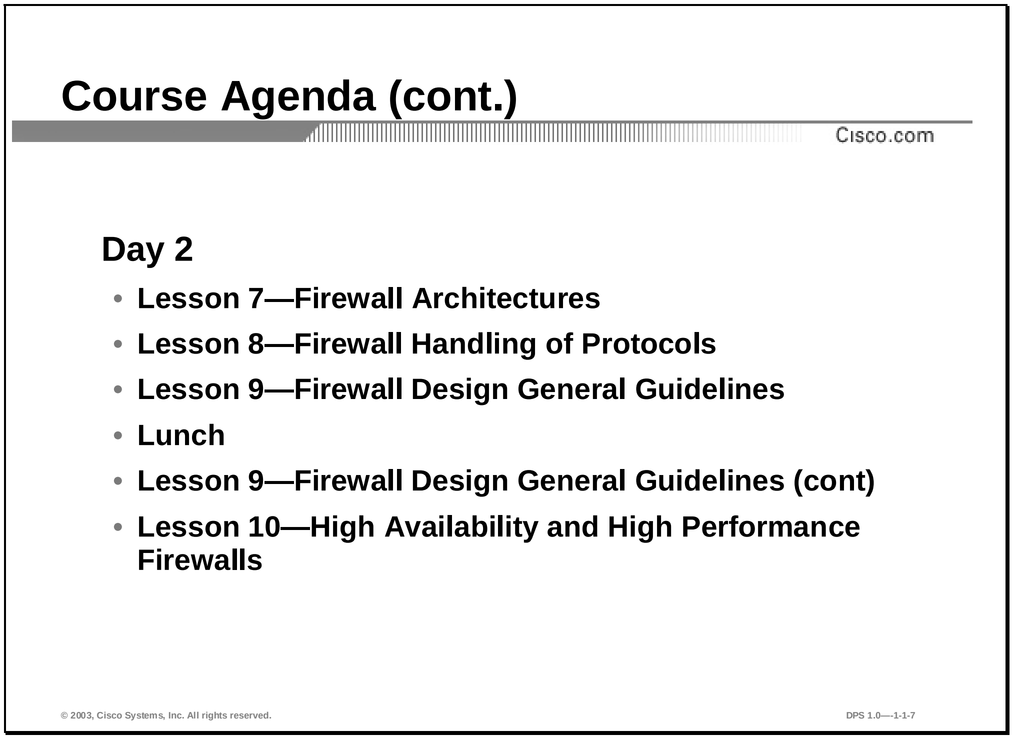


## Course Agenda

### Day 1

- Lesson 1—Course Introduction
- Lesson 2—Design Analysis
- Lesson 3—NAT Overview
- Lunch
- Lesson 4—Design using a NAT/PAT Solution
- Lesson 5—Firewall Function
- Lesson 6—Firewall Technologies

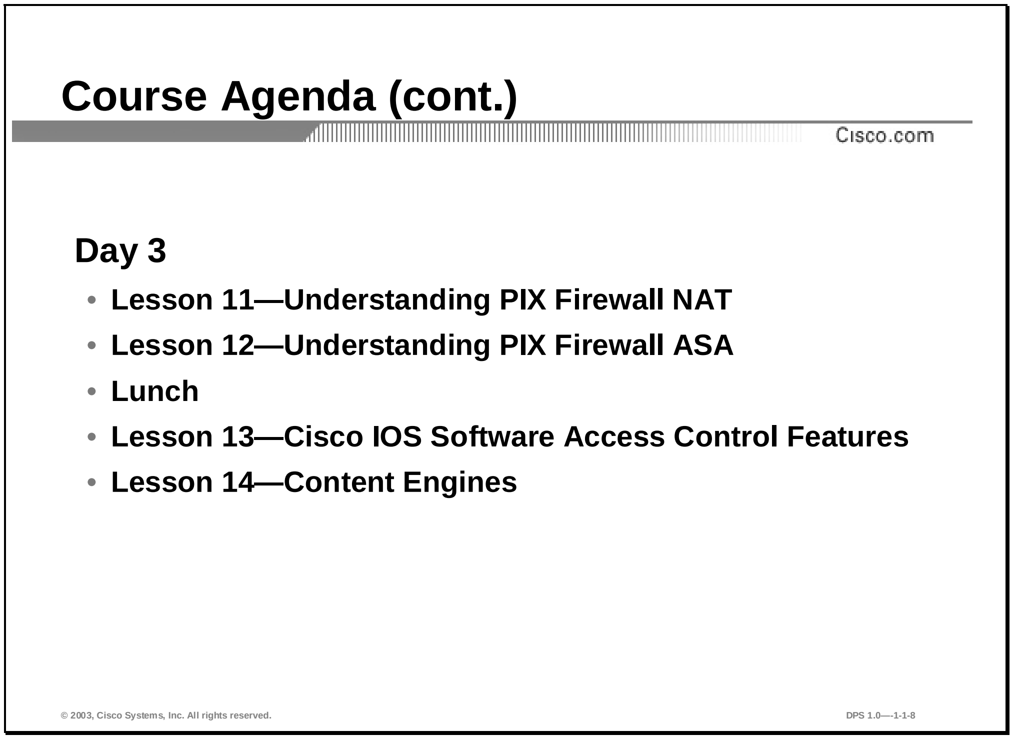
DPS 1.0—-1-1-6



## Course Agenda (cont.)

### Day 2

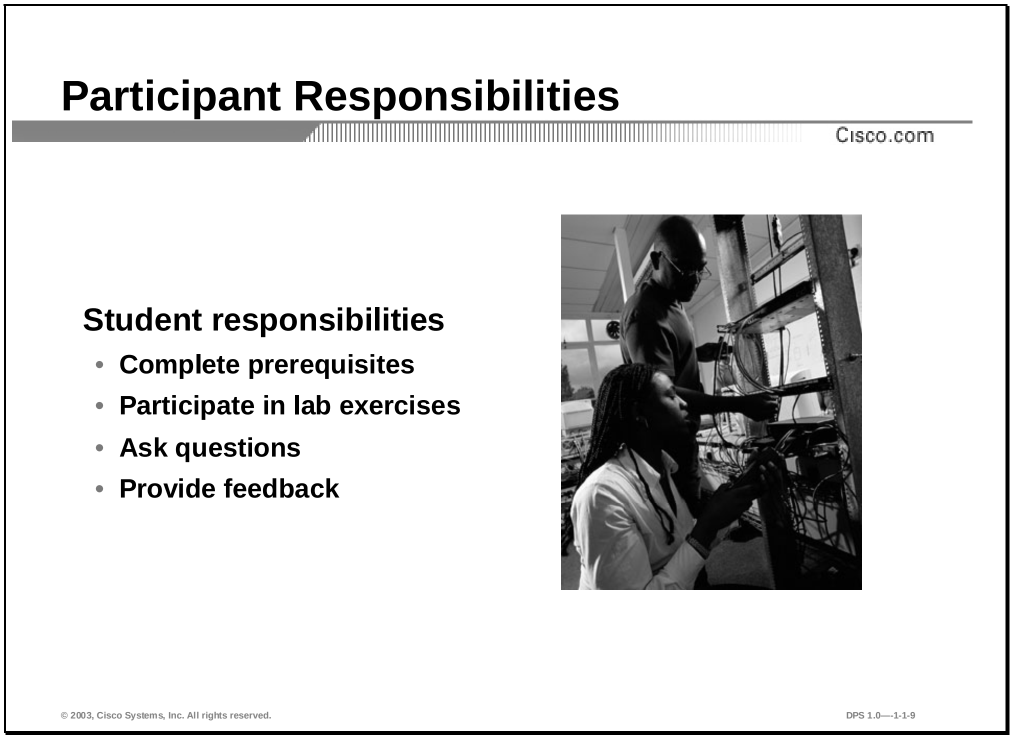
- Lesson 7—Firewall Architectures
- Lesson 8—Firewall Handling of Protocols
- Lesson 9—Firewall Design General Guidelines
- Lunch
- Lesson 9—Firewall Design General Guidelines (cont)
- Lesson 10—High Availability and High Performance Firewalls



## Course Agenda (cont.)

### Day 3

- Lesson 11—Understanding PIX Firewall NAT
- Lesson 12—Understanding PIX Firewall ASA
- Lunch
- Lesson 13—Cisco IOS Software Access Control Features
- Lesson 14—Content Engines

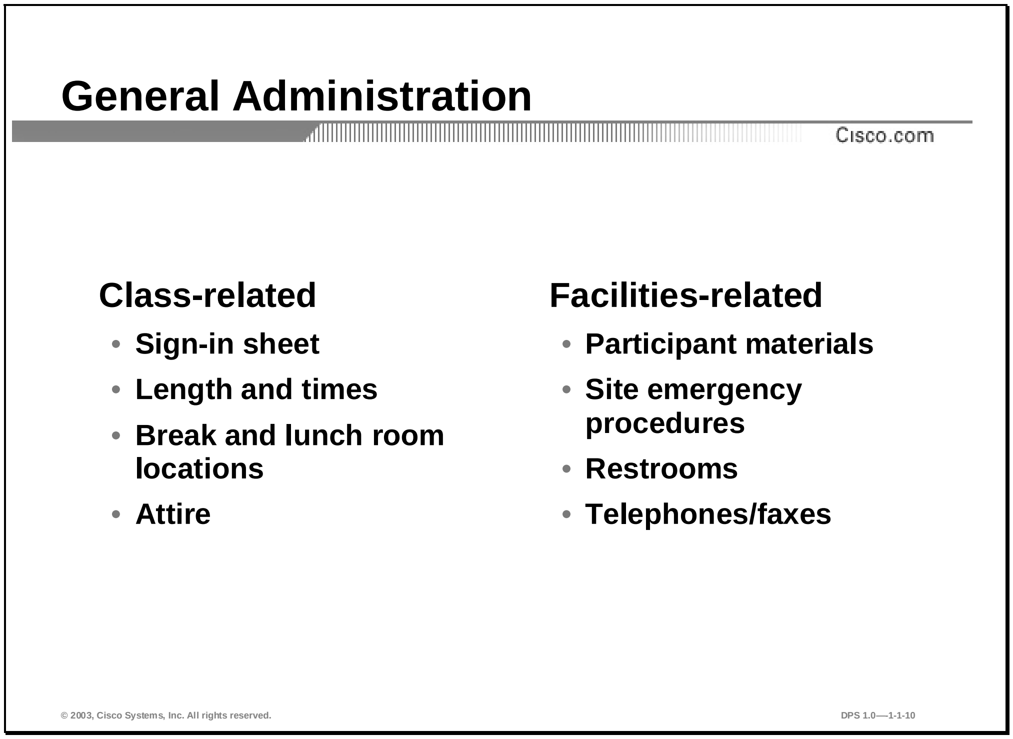


## Participant Responsibilities

Student responsibilities

- Complete prerequisites
- Participate in lab exercises
- Ask questions
- Provide feedback

DPS 1.0—1-1-9



## General Administration

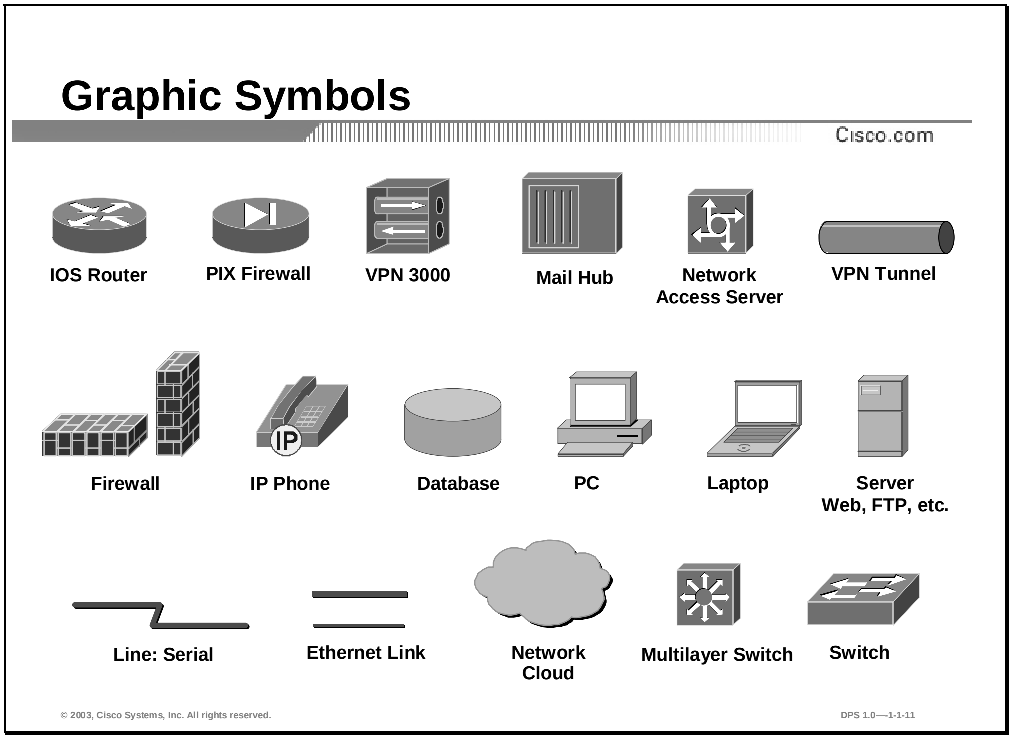
Class-related

- Sign-in sheet
- Length and times
- Break and lunch room locations
- Attire

Facilities-related

- Participant materials
- Site emergency procedures
- Restrooms
- Telephones/faxes

DPS 1.0—1-1-10



## Graphic Symbols

Cisco.com

IOS Router

PIX Firewall

VPN 3000

Mail Hub

Network Access Server

VPN Tunnel

Firewall

IP Phone

Database

PC

Laptop

Server Web, FTP, etc.

Line: Serial

Ethernet Link

Network Cloud

Multilayer Switch

Switch

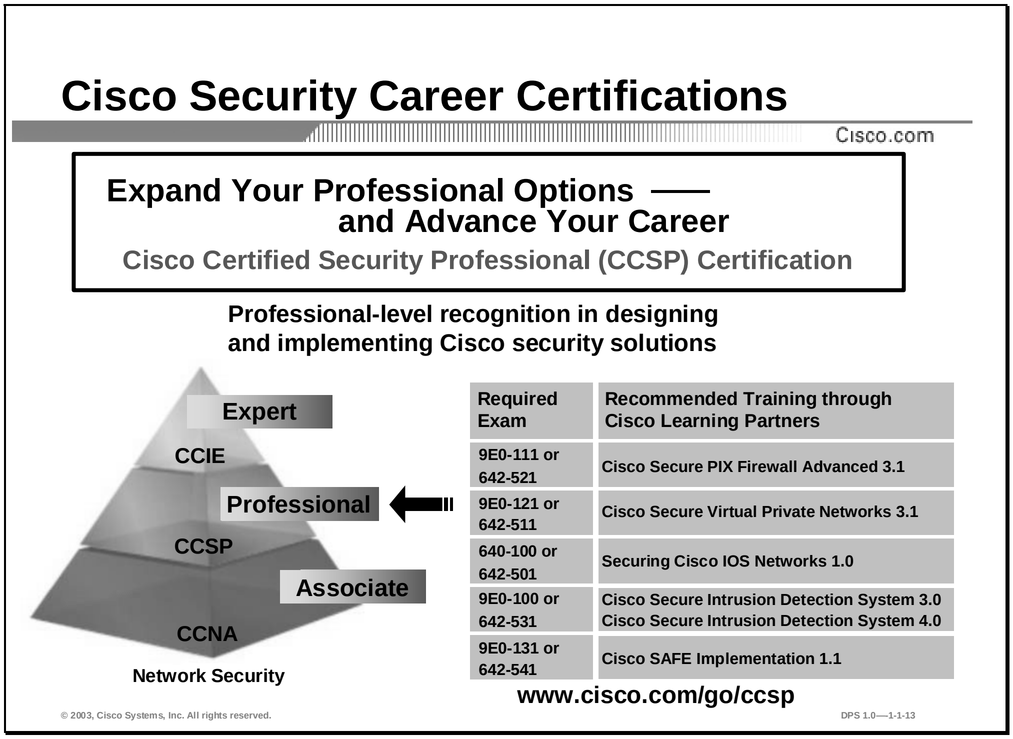
DPS 1.0—1-1-11

## Participant Introductions

Cisco.com

- Your name
- Your company
- Pre-requisites skills
- Brief history
- Objective

DPS 1.0—1-1-12



## Cisco Security Career Certifications

### Expand Your Professional Options —— and Advance Your Career

Cisco Certified Security Professional (CCSP) Certification

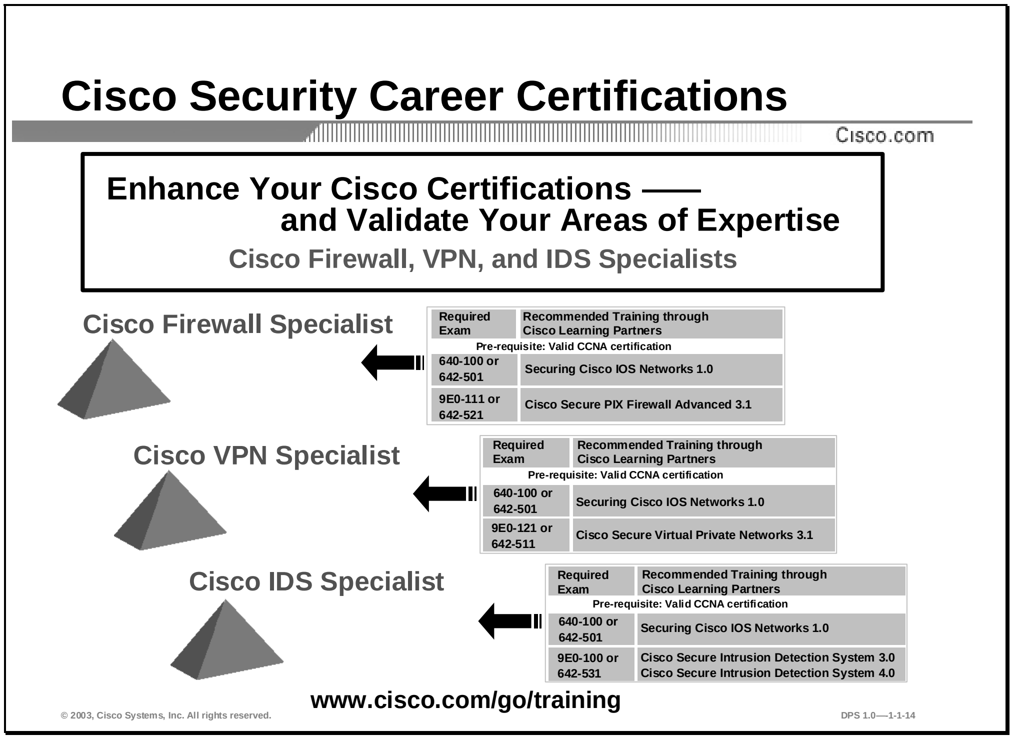
Professional-level recognition in designing and implementing Cisco security solutions

Expert
CCIE
Professional
CCSP
Associate
CCNA
Network Security

| Required Exam | Recommended Training through Cisco Learning Partners |
|---|---|
| 9E0-111 or 642-521 | Cisco Secure PIX Firewall Advanced 3.1 |
| 9E0-121 or 642-511 | Cisco Secure Virtual Private Networks 3.1 |
| 640-100 or 642-501 | Securing Cisco IOS Networks 1.0 |
| 9E0-100 or 642-531 | Cisco Secure Intrusion Detection System 3.0<br>Cisco Secure Intrusion Detection System 4.0 |
| 9E0-131 or 642-541 | Cisco SAFE Implementation 1.1 |

www.cisco.com/go/ccsp

DPS 1.0—1-1-13



## Cisco Security Career Certifications

### Enhance Your Cisco Certifications —— and Validate Your Areas of Expertise

Cisco Firewall, VPN, and IDS Specialists

**Cisco Firewall Specialist**

| Required Exam | Recommended Training through Cisco Learning Partners |
|---|---|
| Pre-requisite: Valid CCNA certification | |
| 640-100 or 642-501 | Securing Cisco IOS Networks 1.0 |
| 9E0-111 or 642-521 | Cisco Secure PIX Firewall Advanced 3.1 |

**Cisco VPN Specialist**

| Required Exam | Recommended Training through Cisco Learning Partners |
|---|---|
| Pre-requisite: Valid CCNA certification | |
| 640-100 or 642-501 | Securing Cisco IOS Networks 1.0 |
| 9E0-121 or 642-511 | Cisco Secure Virtual Private Networks 3.1 |

**Cisco IDS Specialist**

| Required Exam | Recommended Training through Cisco Learning Partners |
|---|---|
| Pre-requisite: Valid CCNA certification | |
| 640-100 or 642-501 | Securing Cisco IOS Networks 1.0 |
| 9E0-100 or 642-531 | Cisco Secure Intrusion Detection System 3.0<br>Cisco Secure Intrusion Detection System 4.0 |

www.cisco.com/go/training

DPS 1.0—1-1-14

# Design Analysis

## Overview

### Importance

This lesson serves as a baseline for building any perimeter solutions, which require specific connectivity and/or security functionality. The requirements and network properties identified with processes identified in this lesson are a primary requirement for any following design and implementation process.

### Lesson Objective

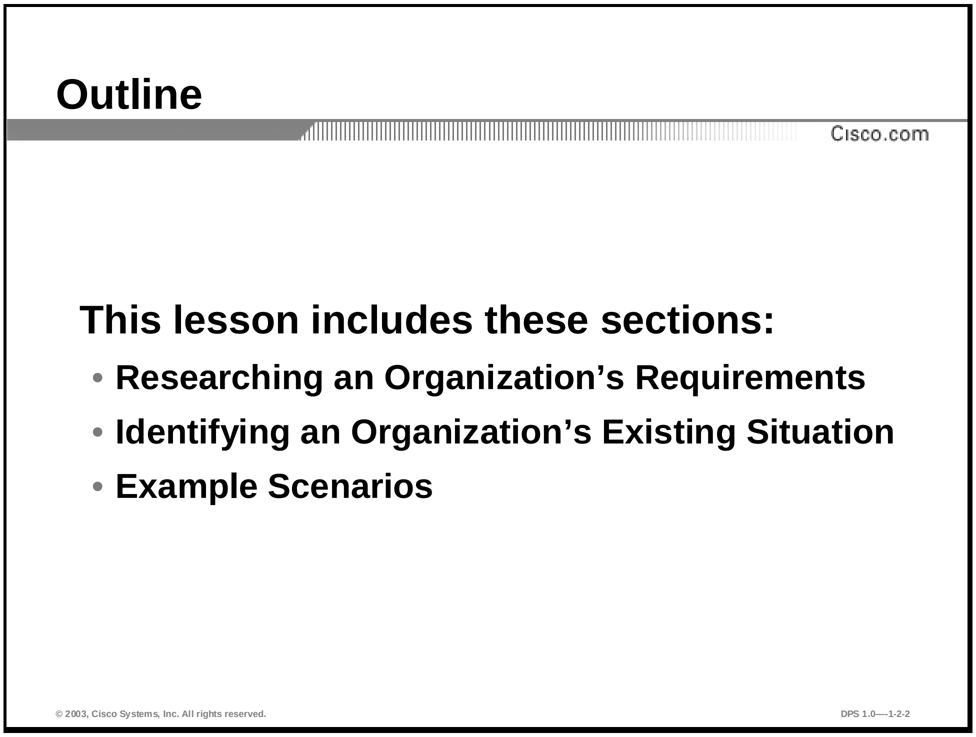
The lesson will enable the learner to identify an organization's requirements and current implementation of perimeter security in order to suggest improvements and to design a new solution based on an organization's requirements, taking into account existing limitations.

## Learner Skills and Knowledge

To fully benefit from this lesson, you must have these prerequisite skills and knowledge:

- A solid knowledge of basic Internet multihoming concepts
- A solid knowledge of enterprise Internet connectivity options
- A basic knowledge of security policy development and risk assessment methods

## Outline



### Outline

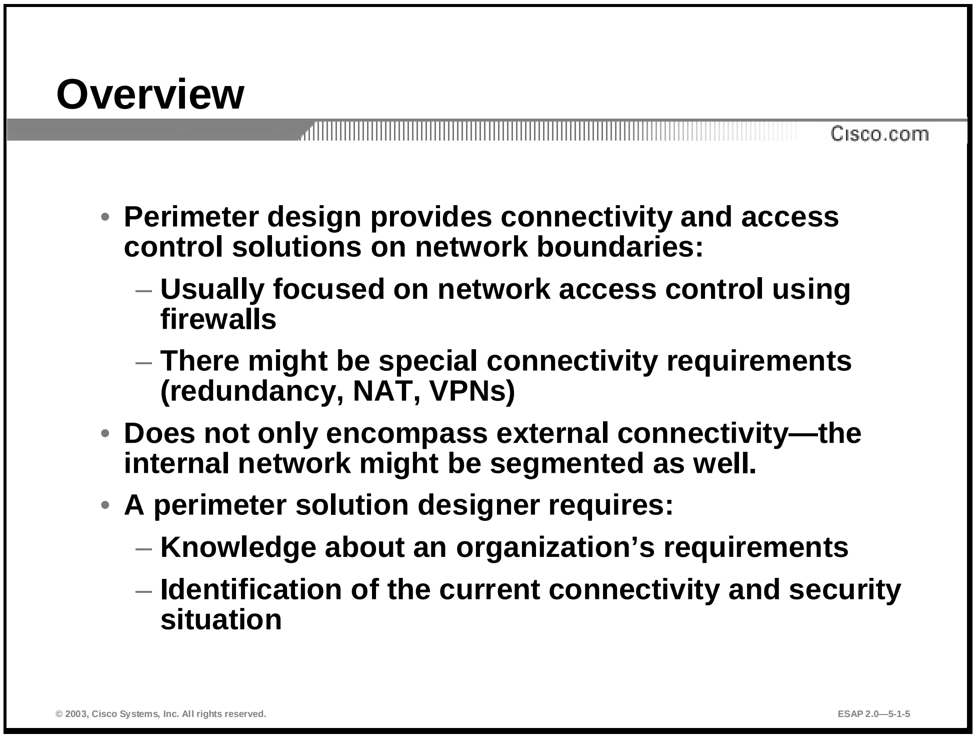
Cisco.com

**This lesson includes these sections:**

- **Researching an Organization’s Requirements**
- **Identifying an Organization’s Existing Situation**
- **Example Scenarios**

DPS 1.0—1-2-2

# Overview



## Overview

Cisco.com

- **Perimeter design provides connectivity and access control solutions on network boundaries:**
  - **Usually focused on network access control using firewalls**
  - **There might be special connectivity requirements (redundancy, NAT, VPNs)**
- **Does not only encompass external connectivity—the internal network might be segmented as well.**
- **A perimeter solution designer requires:**
  - **Knowledge about an organization’s requirements**
  - **Identification of the current connectivity and security situation**

ESAP 2.0—5-1-5

## Introduction

Perimeter security design focuses on providing connectivity and access control enforcement solutions on network boundaries. Perimeter security does not exclusively focus on external connectivity, but also addresses communication between any two perimeters, depending on the definition of a perimeter.

## Definition

A perimeter is a clearly defined part of a network. Access across the perimeter must be controlled.

## Example

For example, a network might also be segmented on the “inside”. A large enterprise network might be divided into security zones (or perimeters), each zone containing a particular part of the network, where access control needs to be enforced. Examples of security zones (perimeters) include server farms, individual branch offices, IT labs and classrooms, different departments (engineering, finance, HR): all require policies for access control.

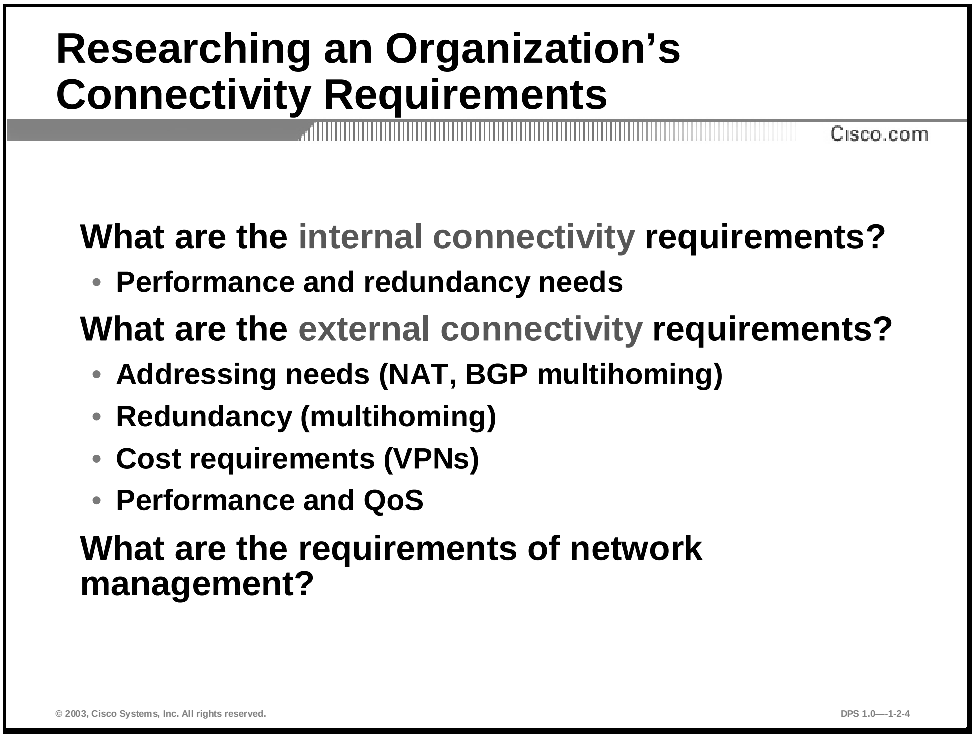
That said, perimeter security and connectivity solutions are focused on some well-known approaches:

- Access control between perimeters is usually enforced with network firewalls, which connect perimeters together

- Connectivity requirements usually fall into one of the well-known categories, such as the need for redundant connectivity (high availability), resolution of addressing problems (Network Address Translation [NAT] solutions), or secure communication over untrusted networks (virtual private networks [VPNs])

To provide perimeter design solutions, a network security architect needs to be aware of an organization’s connectivity and security requirements, as well as the current state of connectivity and security in the network.

# Researching an Organization’s Requirements



## Researching an Organization’s Connectivity Requirements

Cisco.com

What are the internal connectivity requirements?

- Performance and redundancy needs

What are the external connectivity requirements?

- Addressing needs (NAT, BGP multihoming)
- Redundancy (multihoming)
- Cost requirements (VPNs)
- Performance and QoS

What are the requirements of network management?

## Objective

This section will enable the learner to identify the organizations requirements of perimeter security.

## Introduction

To design a perimeter solution, the requirements of an organization need to be clearly stated and analyzed to provide an optimal design. Factors such as internal vs. external connectivity requirements, performance, and the desired level of security need to be addressed and agreed on to strike an optimal balance of functionality in the final solution.

## Analyzing Connectivity Requirements

Because they often are implemented in different ways, the perimeter designer should focus separately on internal and external connectivity when researching an organization’s connectivity requirements. Internal connectivity focuses more on high performance and automatic network operation (for example, through interior routing protocols), while external connectivity is more static and controlled (for example, using Border Gateway Protocol [BGP] for routing and control of addressing through NAT).

When analyzing internal connectivity requirements, performance and redundancy generally play a central role. Many internal computing resources need to be highly available, with sufficient performance available all the time to enable seamless connectivity.

External connectivity usually focuses on the interfacing of a high-speed, highly redundant network with a network outside an organization’s control. An external partner or a service provider, therefore, often limits connectivity options. The requirements can be separated into:

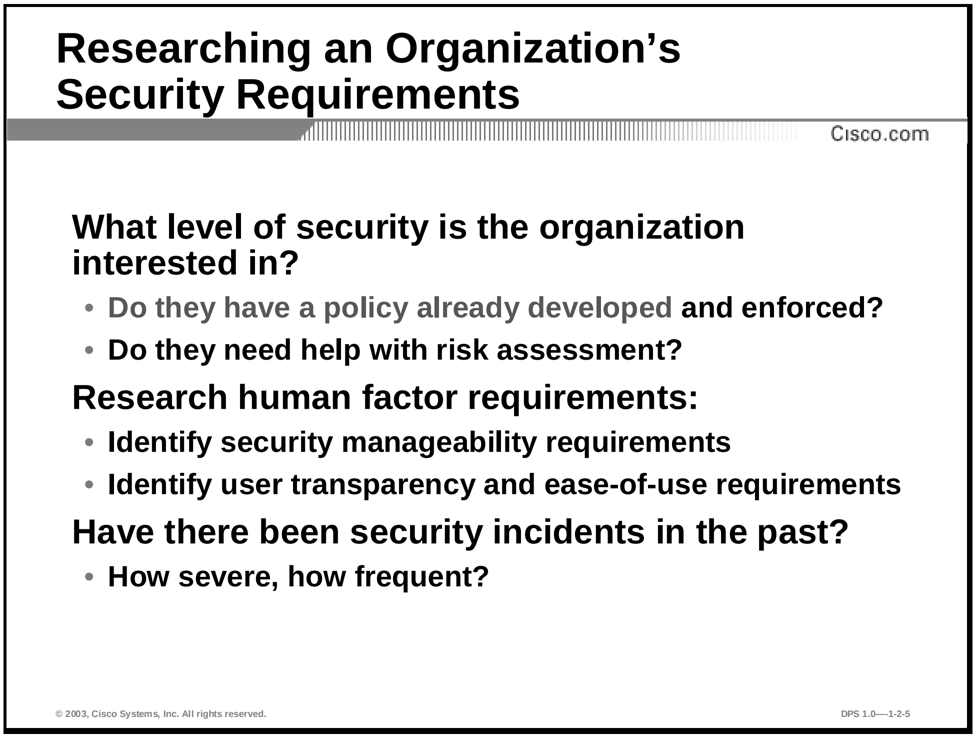
- Addressing needs when connecting to external networks, such as the Internet or a business partner. Addressing needs might be different in high-availability scenarios, such as BGP multihoming with provider-independent address space.

- Analyzing redundancy requirements for external connections, such as the Internet connection. Conducting business over the Internet often requires fast-converging multihoming solutions to maximize the availability of external connections.

- Cost requirements of external connectivity might influence the selection of the transport technology, such as classic TDM networks, or VPN links as an alternative.

- Performance needs might also dictate the choice of transport technology, and require the use of quality of service (QoS) mechanisms on external links.

- Addressing the network management needs in the context of perimeter design. Monitoring external connections might require the access control policy to include network management traffic between perimeters.



## Researching an Organization’s Security Requirements

Cisco.com

**What level of security is the organization interested in?**

- **Do they have a policy already developed and enforced?**
- **Do they need help with risk assessment?**

**Research human factor requirements:**

- **Identify security manageability requirements**
- **Identify user transparency and ease-of-use requirements**

**Have there been security incidents in the past?**

- **How severe, how frequent?**

DPS 1.0—1-2-5

## Analyzing Security Requirements

When researching an organization’s security requirements, the designer should first and foremost analyze the organization’s security policy and understand how it applies to the organization’s network. The designer should also be aware of the extent to which the policy has been implemented and verify that the current security measures actually implement the policy requirements.

If the organization does not have a policy already developed and enforced, it is possible that they require help with risk assessment, which will result in the development of an informal or formal security policy.

**Note** The establishment of a formal policy before implementing security measures is the most reliable method of ensuring consistent implementation.

In terms of security, the designer must consider human factors when identifying the needs for security manageability and end user experience. To ensure end users do not become frustrated and try to bypass security deliberately, the transparency of security mechanisms should be considered as a high priority requirement. The policy should address these issues, as well as provide guidelines for end user security awareness training.

If a history of security incidents exists, the designer should analyze it to identify previously overlooked weaknesses in the organization’s policy or security implementation. The severity of incidents and their frequency should provide valuable input to the designer.

## Practice

Q1) Which are frequent connectivity needs in perimeter security design?

A) firewall implementation

B) NAT implementation

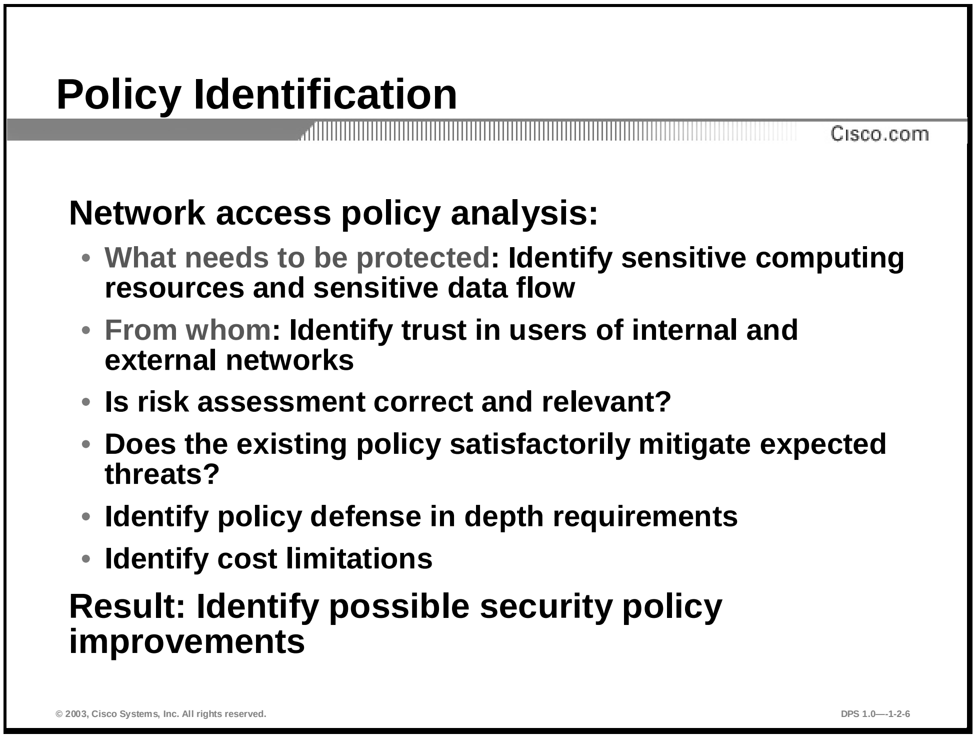
C) BGP multihoming implementation

D) IDS implementation

E) local address allocation

F) IGP implementation

# Identifying an Organization's Existing Situation



## Policy Identification

**Network access policy analysis:**

- What needs to be protected: **Identify sensitive computing resources and sensitive data flow**
- From whom: **Identify trust in users of internal and external networks**
- **Is risk assessment correct and relevant?**
- **Does the existing policy satisfactorily mitigate expected threats?**
- **Identify policy defense in depth requirements**
- **Identify cost limitations**

**Result: Identify possible security policy improvements**

DPS 1.0—-1-2-6

## Objective

This section will enable the learner to identify the organizations current situation and possible limitations for perimeter security design.

## Introduction

When a designer has reviewed and analyzed the organization's requirements, the current state of the network and organizational practices needs to be identified to verify their current compliance with the requirements, and identify possible improvements and the potential need to redesign a part of the system, or to rebuild a part of the system from scratch to satisfy the requirements.

## Policy Identification

If a security policy exists, the designer should analyze it to identify the security requirements, which will influence the design of the perimeter solution. Initially, two basic areas of the policy should be examined:

- The policy should identify the assets that require protection. This will help the designer provide the correct level of protection for sensitive computing resources, and identify the flow of sensitive data in the network.

- The policy should identify possible attackers. This will give the designer insight into the level of trust assigned to internal and external users, ideally identified by more specific categories such as business partners, customers of an organization, outsourcing IT partners.

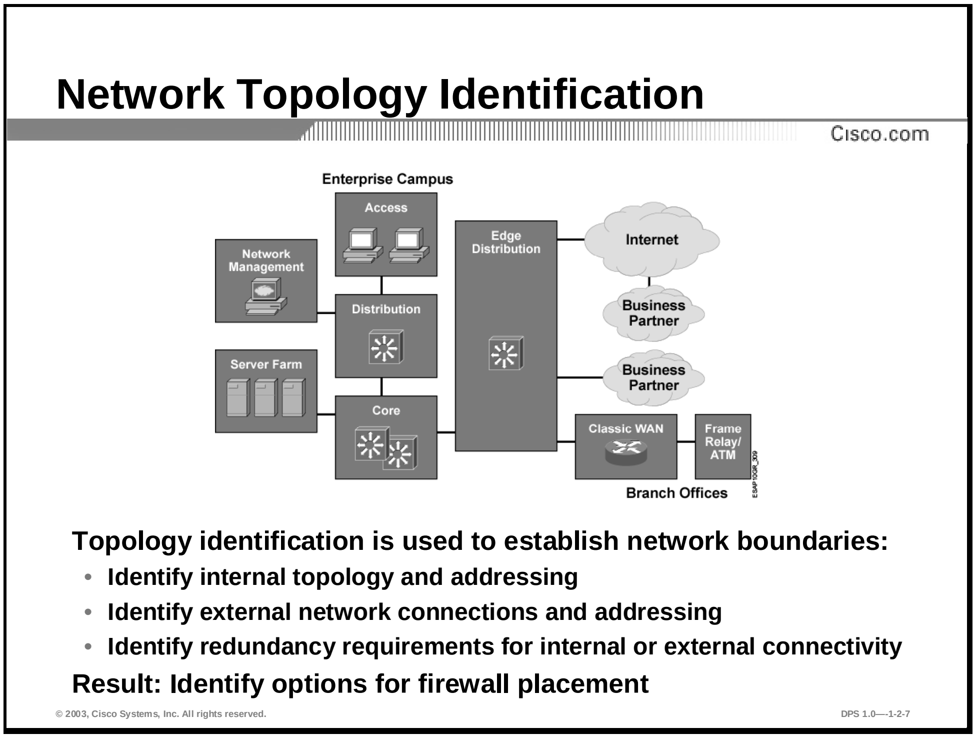
The designer should also be able to evaluate if the policy was developed using correct risk assessment procedures—that is, did the policy development include all relevant risks for the organization and not overlook important threats? The designer should also re-evaluate the policy mitigation procedures to determine if they satisfactorily mitigate expected threats. This ensures that the policy, which the designer will work with, is up to date and complete.

Organizations who need a high level of security assurance will require defense-in-depth mechanisms to be deployed to avoid single-points-of-failure. The designer also needs to work with the organization to determine how much investment in security measures is acceptable for the resources that require protection.

The result of policy analysis will be:

- The evaluation of policy correctness and completeness

- Identification of possible policy improvements, which need to be made before the security implementation stage



## Network Topology Identification

Cisco.com

**Topology identification is used to establish network boundaries:**

- **Identify internal topology and addressing**
- **Identify external network connections and addressing**
- **Identify redundancy requirements for internal or external connectivity**

**Result: Identify options for firewall placement**

DPS 1.0—1-2-7

# Topology Identification

The next step in identification of an organization’s current situation is the identification of network topology. This will provide a detailed insight into the definition of network boundaries.

| **Note** | Identification of network connections within the topology might identify connections that an organization is not aware of. From the security perspective, this is crucial to prevent any data leaks over “backdoor” connections between network perimeters. |
|---|---|

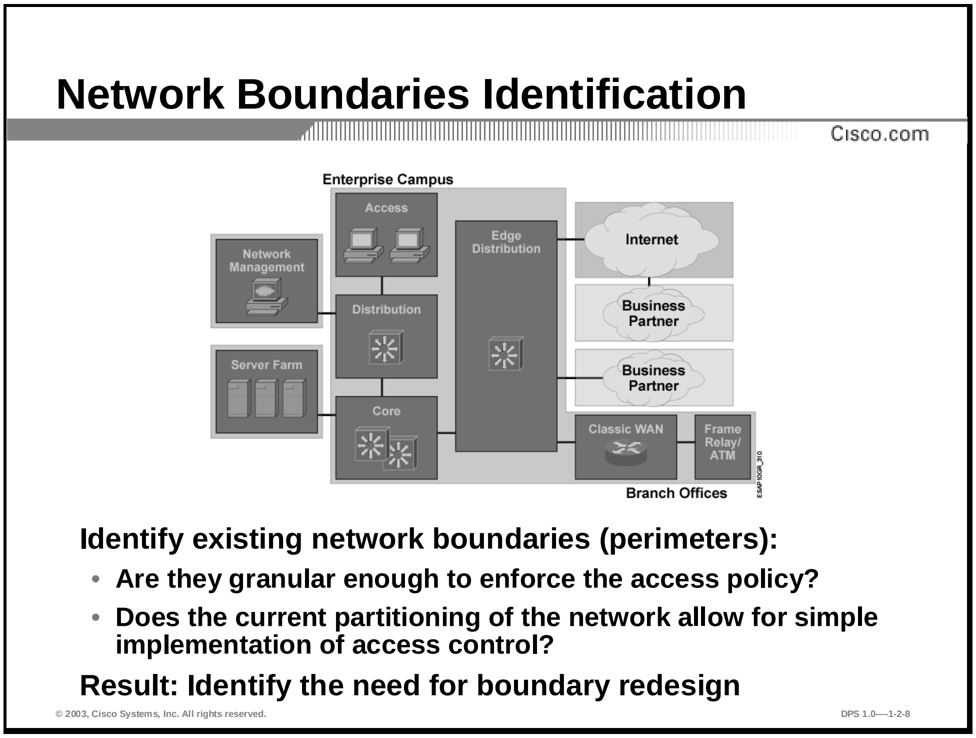
Topology identification can be broken down into multiple parts, including identification of:

- **Internal topology and addressing:** Allows the designer to define the internal network boundary, and robustly define perimeter boundaries, if the internal network is to be segmented. The designer should also identify the addressing of the internal network—required in the future for the firewall rule design.

- **Connections to external networks, and the addressing of external networks:** Enables the designer to enforce access control efficiently at the correct choke points. It also helps the designer identify the need for routing implementation and network address translation.

- **Redundancy requirements for inside and outside connectivity:** Helps the designer to design redundant external connectivity (for example, multihoming) and to build in redundancy into the security elements.

To summarize, topology identification provides the designer with various options for placement of firewalls and the implementation of boundary connectivity between perimeters.



Network Boundaries Identification

Cisco.com

Identify existing network boundaries (perimeters):

- Are they granular enough to enforce the access policy?
- Does the current partitioning of the network allow for simple implementation of access control?

Result: Identify the need for boundary redesign

DPS 1.0—1-2-8

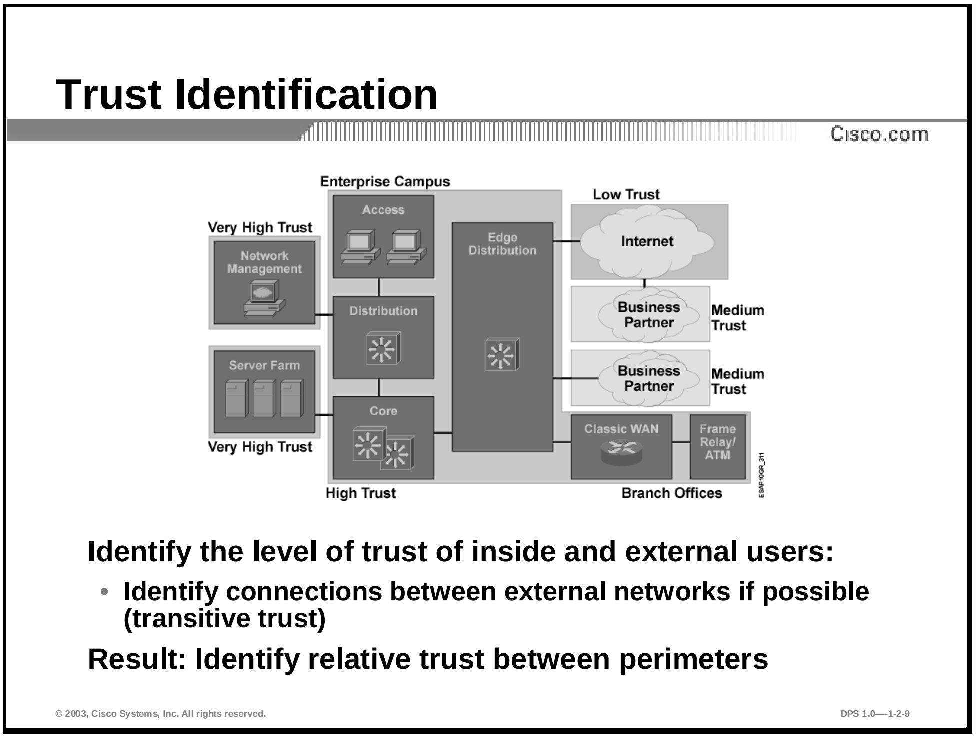
## Network Boundaries' Identification

With an identified topology, the designer needs to identify boundaries, which define perimeters in a network. As access control will be performed between the perimeters, the boundaries need to be set so they provide the necessary granularity of access control. Moreover, perimeters, which are not clearly defined result in very complex policy enforcement, which might result in compromises of the policy.

The result of boundary identification is the possible identification of a need to redesign boundaries to implement access control in policy-compliant, effective, and simple fashion.

## Example

An enterprise, which requires twenty access servers for dial-in connectivity, has connected those access servers in various access LAN networks in their central site. To perform access control for dial-up users, firewalls would need to be deployed at each access server, while a more centralized placement could allow a single firewall to implement the policy. This is a more robust and scalable approach.



## Trust Identification

**Identify the level of trust of inside and external users:**

- **Identify connections between external networks if possible (transitive trust)**

**Result: Identify relative trust between perimeters**

DPS 1.0—1-2-9

# Trust Identification

When network perimeters are identified, the trust level of those perimeters needs to be determined. The trust level is determined by the following factors:

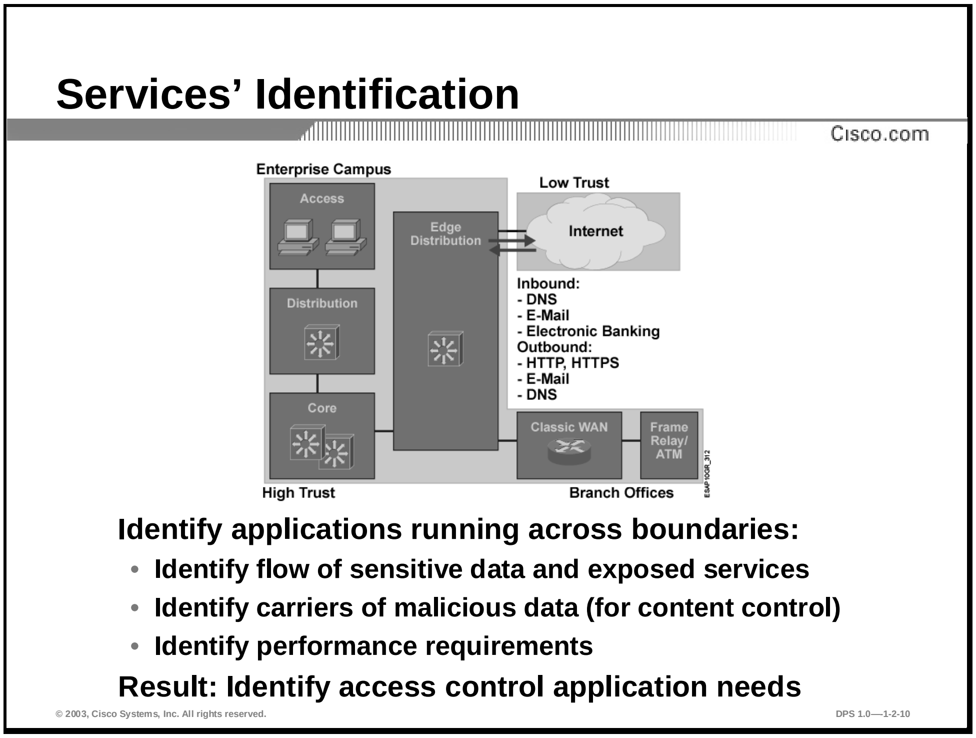
- How trusted are users inside the perimeter in question? Is it likely that those users could compromise a computing resource on a more trusted perimeter?

- How trusted is the infrastructure of the perimeter? Is it physically secure enough not to allow confidentiality or integrity violation of transit data? Is it possible that an attacker might compromise a resource in that perimeter, and use that resource to attack other perimeters?

This identification results in assigning relative levels of trust to perimeters, which are allowed to communicate. To identify the relative trust relationship of two perimeters (that is, identifying one perimeter as being more trusted than another) the levels of trust can be labeled with simple terms, such as “untrusted”, “trusted”, or “conditionally trusted”.

# Example

The PIX Firewall allows a user to specify the level of trust of PIX Firewall interfaces that connect the PIX Firewall to neighboring perimeters. The security level property of the interface is a number from 0 to 100, 100 and 0 specifying the highest and lowest level of trust respectively.



## Services' Identification

**Identify applications running across boundaries:**

- **Identify flow of sensitive data and exposed services**
- **Identify carriers of malicious data (for content control)**
- **Identify performance requirements**

**Result: Identify access control application needs**

DPS 1.0—1-2-10

## Services' Identification

The next step is to identify services—protocols and applications running across network boundaries. This identification can be broken down into multiple parts, including identification of the:

- Location and flow of sensitive data, and the requirement for exposing trusted perimeter services to untrusted perimeters. This enables the designer to identify the direction of application sessions, the complexity of required applications, and the possible firewall architectures for exposing trusted services to untrusted perimeters.

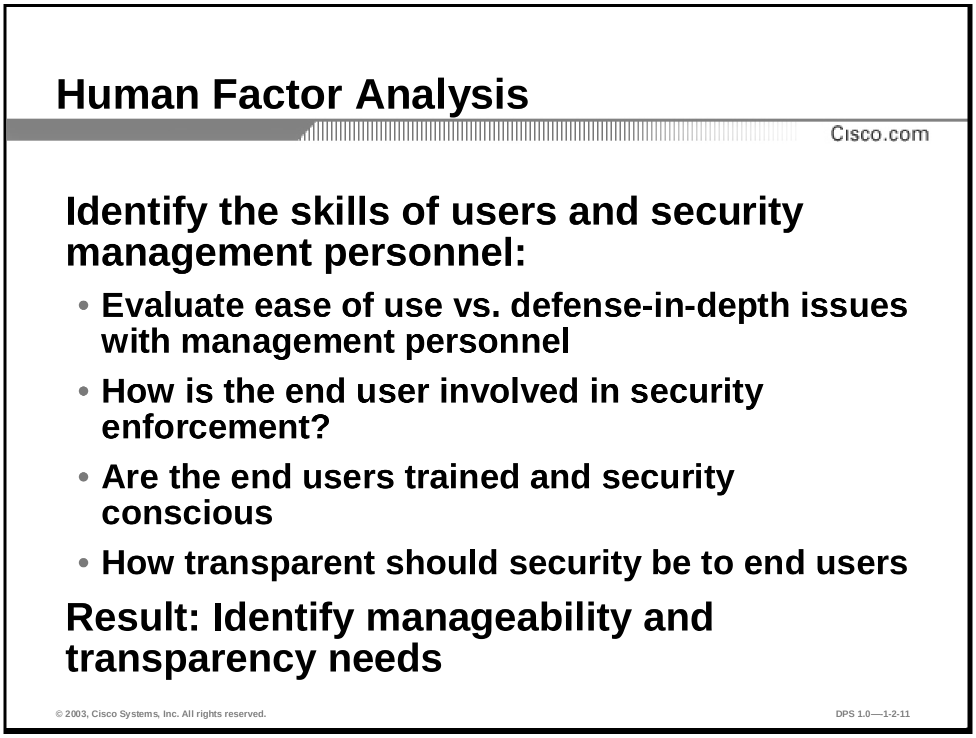
- Possibilities of malicious data entering trusted perimeters over application protocols. This identifies content filtering requirements to comply with an organization's policy.

- Performance requirements for a particular application. This provides the designer with information on which access control technologies to use.

This process results in the identification of access control application needs. That is, it provides the designer with information on how:

- Granular the access technologies must be to provide the required filtering.

Applications will be relayed across network boundaries.



## Human Factor Analysis

Identify the skills of users and security management personnel:

- Evaluate ease of use vs. defense-in-depth issues with management personnel
- How is the end user involved in security enforcement?
- Are the end users trained and security conscious
- How transparent should security be to end users

Result: Identify manageability and transparency needs

DPS 1.0—1-2-11

# Human Factor Analysis

Human factors in perimeter design are identified from two perspectives:

- The skills of security management personnel and their ability to manage security without compromising it.
- The skills of end users and their involvement in the enforcement of security policies. If the end users are not trained properly, their actions can inadvertently compromise security. This has to be taken into consideration for risk assessment. This risk has to be mitigated using either end user training, or technology which helps prevent user mistakes.

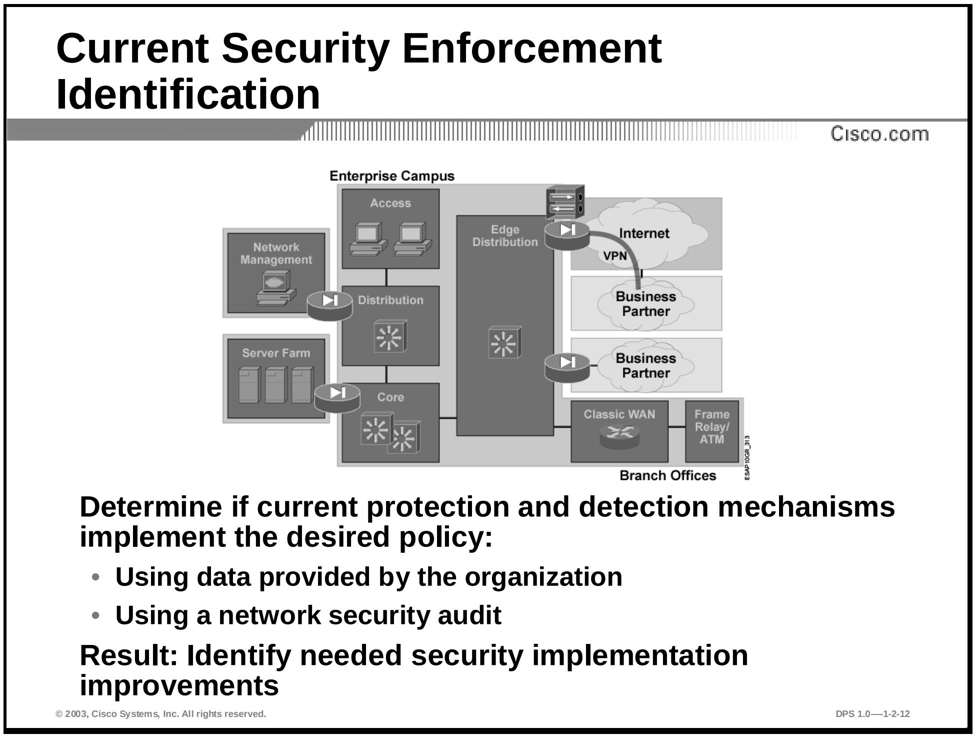
# Example

If end-users cannot to be trusted to encrypt all sensitive email messages to recipients outside the enterprise, an encrypting email gateway can be set up to perform this automatically. Alternatively, a VPN technology can provide a secure path independently of user actions.

# Example

An organization might have a clearly defined policy, but no personnel to either implement the policy properly, or to manage it operationally. The network security designer must take this into consideration, as he/she will be required to implement the network access control policy. The network security designer has to provide the organization with either an extremely easy-to-manage solution (perhaps with the help of an outside organization, which provides outsourced security management), or identify training needs for the organization’s personnel.

The analysis of human factors enables the designer to identify manageability needs, and provide the proper transparency of security mechanisms to end users.



## Current Security Enforcement Identification

Determine if current protection and detection mechanisms implement the desired policy:

- Using data provided by the organization
- Using a network security audit

Result: Identify needed security implementation improvements

DPS 1.0—1-2-12

## Evaluation of Current Security Policy Enforcement

The identification process might involve a process to determine how the current protection mechanisms implement the desired policy. This can be performed in two, often-complementary ways:

- **Performing a network audit using internal or external (tiger team) testing:** Sometimes called the “black box” approach, the auditor simply observes the network’s response to the penetration attempts. Interesting results can be obtained from this approach, but it depends heavily on the source of audit and network configuration (such as access control restrictions).

- **Acquiring all the information from the network owner:** The auditor can request network device and firewall configurations to gain understanding of policy enforcement. Traditionally, this is a more successful approach, if the organization’s documentation practices are reliable. Often, a combination of both methods is required for good results.

Also, the organization’s detection and response capabilities need to be analyzed, to provide policy and implementation guidelines for establishing or upgrading the monitoring capability (such as reporting tools and intrusion detection systems).

This process identifies whether or not the enforcement complies with the policy. If the enforcement does not comply with the policy this process also identifies the required security improvements.

# Practice

Q1) Why is it necessary to determine the level of trust in each perimeter?

A) to provide access control

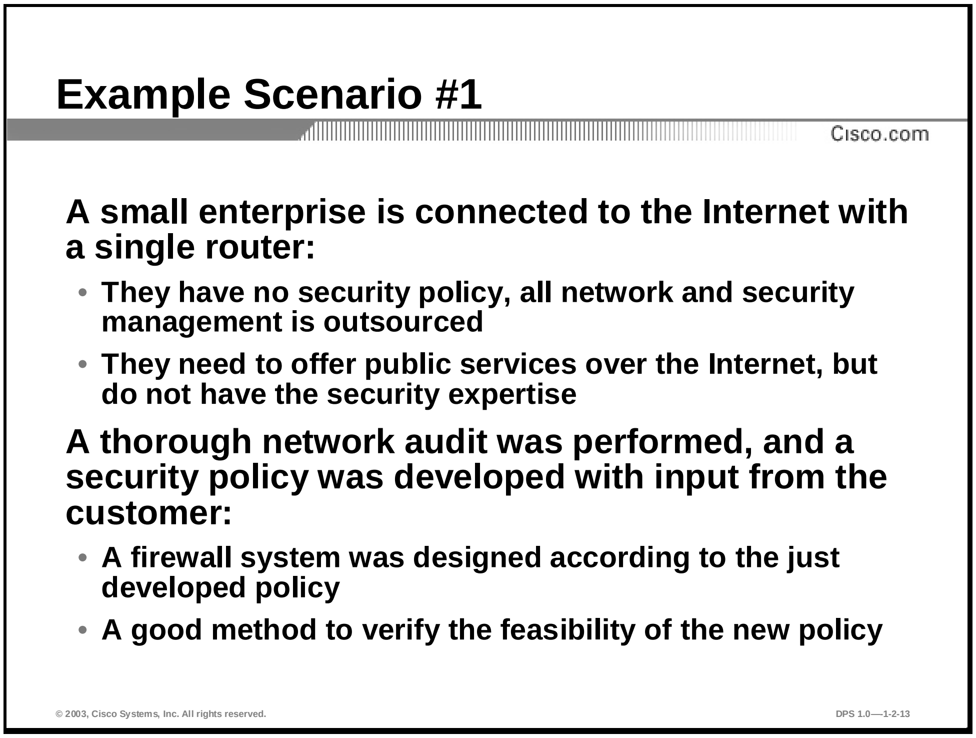
B) to determine relative trust between perimeters

C) to determine optimal connectivity topology

D) to address the human factor properly

E) to identify backdoors

# Example Scenarios



## Example Scenario #1

A small enterprise is connected to the Internet with a single router:

- They have no security policy, all network and security management is outsourced
- They need to offer public services over the Internet, but do not have the security expertise

A thorough network audit was performed, and a security policy was developed with input from the customer:

- A firewall system was designed according to the just developed policy
- A good method to verify the feasibility of the new policy

DPS 1.0—1-2-13

## Objective

This section will enable the learner to recognize common perimeter security situations and requirements in enterprise networks.

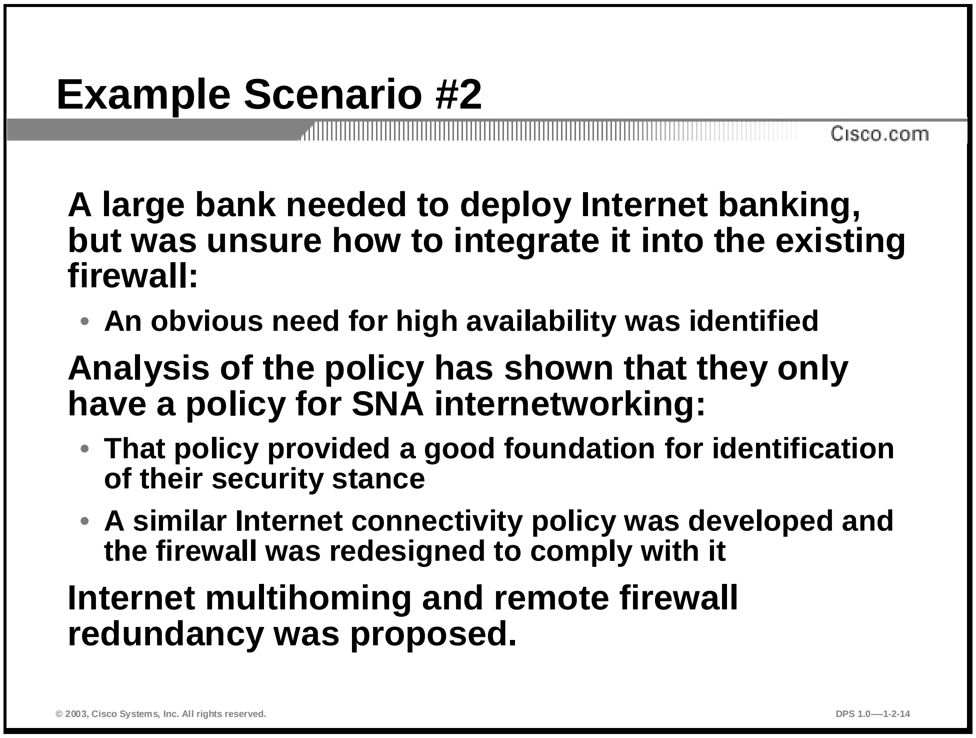
## Introduction

The case studies presented in this section provide some examples of how perimeter security and connectivity needs are addressed by various organizations.

## Example Scenario

The first example scenario focuses on a small enterprise network, which is connected to the Internet with a small router. No security policy exists, as there was no security process in place due to the lack of expertise inside the organization, and the organizations is outsourcing all security services to external partners. So far, all connectivity was strictly outbound to external networks, and a need was expressed to offer public services to the Internet.

The organization decided to tackle security more seriously and to start the process of security, instead of relying on outside partners to make all the decisions for them. First, a snapshot of the current state and vulnerability of the network was determined through an external audit. Then clear goals were set regarding the security requirements, and a policy was developed. To verify if the policy is sound and feasible to implement, a firewall was set up according the policy statement. An additional audit verified the compliance of firewall design and implementation with the desired policy.



## Example Scenario #2

Cisco.com

A large bank needed to deploy Internet banking, but was unsure how to integrate it into the existing firewall:

- An obvious need for high availability was identified

Analysis of the policy has shown that they only have a policy for SNA internetworking:

- That policy provided a good foundation for identification of their security stance
- A similar Internet connectivity policy was developed and the firewall was redesigned to comply with it

Internet multihoming and remote firewall redundancy was proposed.

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—1-2-14

# Example Scenario

This example scenario focuses on a large bank, which needs to deploy public services, such as electronic banking, over the Internet. The bank has an existing firewall, but is not comfortable with integrating a new, complex service in the firewall with a very high level of security, as is required. Also, high availability is a primary need for all connectivity and security functionality in the upcoming solution.

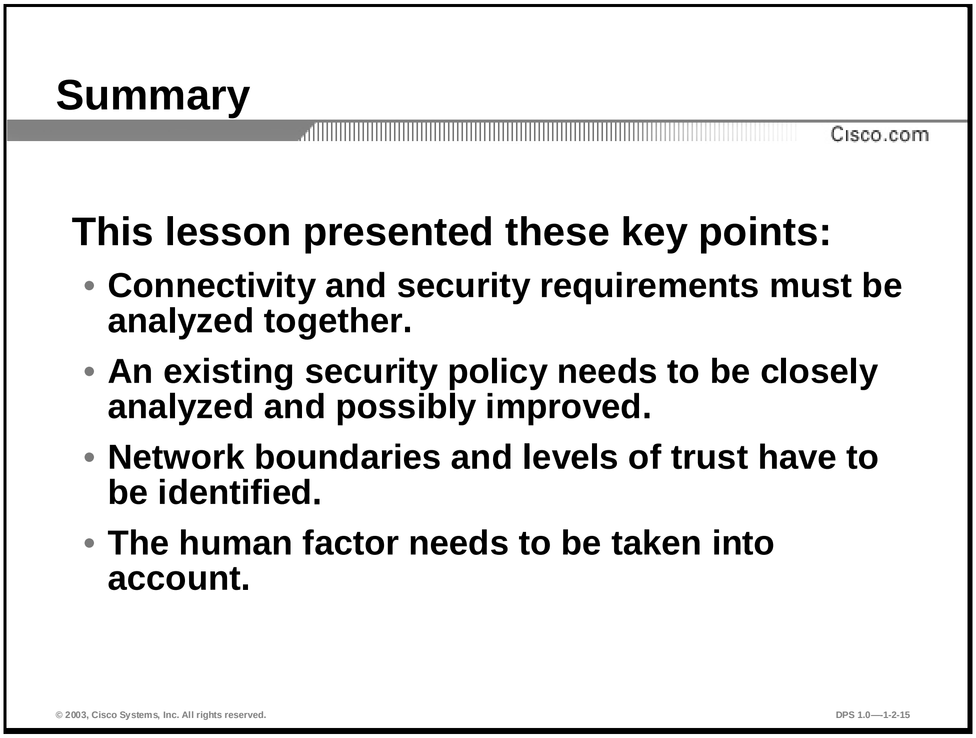
The bank has claimed to have a security policy, but close examination has shown that the security policy they have applies only to legacy SNA internetworking. However, that policy taken as a baseline, a new Internet connectivity policy was developed, taking into account specific threats of Internet connectivity. The firewall was redesigned to comply with it, and the electronic banking solution was integrated into it.

Such an example can show that even outdated, or not directly applicable policies can serve as valuable input to the designer, when a new piece of policy needs to be agreed on. Existing policies can provide hints about the global security goals and strategy of an organization, which reflect in every subsequent application-specific policy.

To address the high-availability needs for this particular application, Internet multihoming and full firewall/server redundancy was proposed as a part of the solution.

# Summary

This section summarizes the key points discussed in this lesson.



## Summary

This lesson presented these key points:

- Connectivity and security requirements must be analyzed together.
- An existing security policy needs to be closely analyzed and possibly improved.
- Network boundaries and levels of trust have to be identified.
- The human factor needs to be taken into account.

© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—1-2-15

## Next Steps

After completing this lesson, go to:

- Network Address Translation (NAT) Solutions module, NAT Overview lesson

## References

For additional information, refer to these resources:

- Cisco Systems Information Security Policies, http://wwwin.cisco.com/infosec/policies/
- Security Posture Assessment, http://wwwin.cisco.com/cmc/cc/serv/mkt/sup/advsv/pavsup/sposass/

# Quiz: Design Analysis

Complete the quiz to assess what you have learned in this lesson.

## Objectives

This quiz tests your knowledge on how to:

- Identify an organization's requirements and current implementation of perimeter security in order to suggest improvements
- Design a new solution based on those requirements, taking into account existing limitations

## Instructions

Answer these questions:

1. What are some possible special requirements for external connectivity?
2. Why does an existing security policy need to be analyzed and not simply obeyed?
3. How are levels of trust in outside networks determined?
4. What are some human factor considerations in perimeter security design?

## Scoring

You have successfully completed the quiz for this lesson when you earn a score of 80 percent or better.

# NAT Overview

## Overview

### Introduction

This lesson gives you a compact overview of Network Address Translation (NAT) technology, with a special focus on Cisco IOS and PIX Firewall configurations. Besides typical addressing scenarios, this lesson will also explain technical details of great importance for real-world implementations, such as protocol compatibility and NAT security considerations.

### Importance

NAT is one of the most important functions of an enterprise's perimeter configuration, and therefore central to security and connectivity considerations. Configuring NAT requires in-depth knowledge about the mechanisms and also about the side effects that might occur.

### Lesson Objective

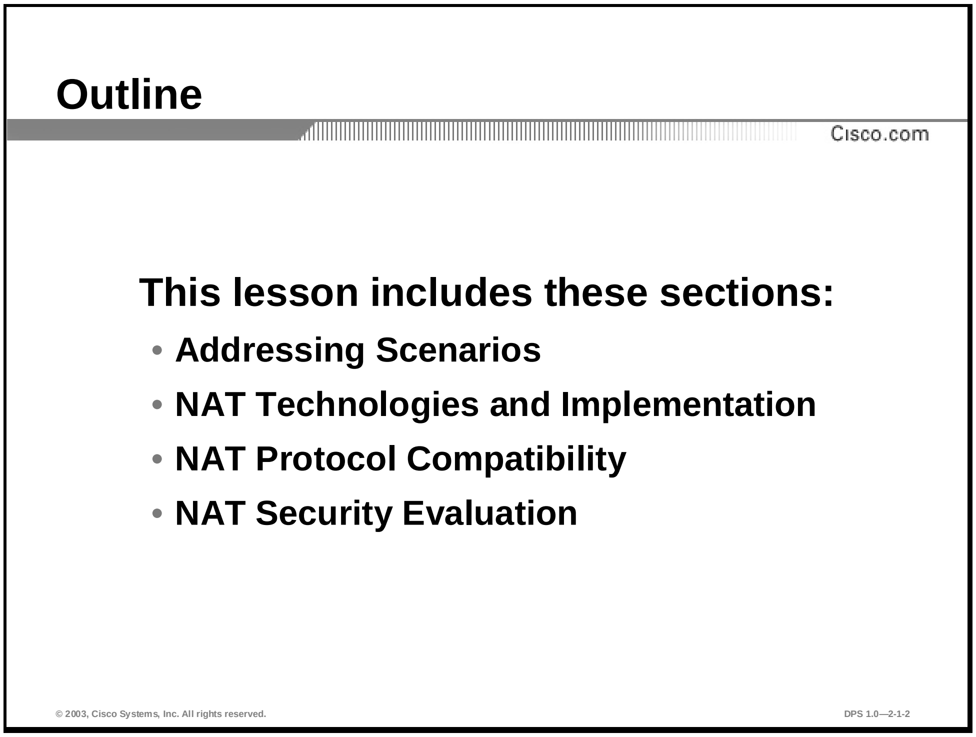
The lesson will enable the learner to identify and compare NAT technologies, and select an appropriate NAT technology for an organization's requirements.

## Learner Skills and Knowledge

To fully benefit from this lesson, you must have these prerequisite skills and knowledge:

- A solid understanding about IP addressing and routing
- Experience with basic perimeter security issues

## Outline



Outline

Cisco.com

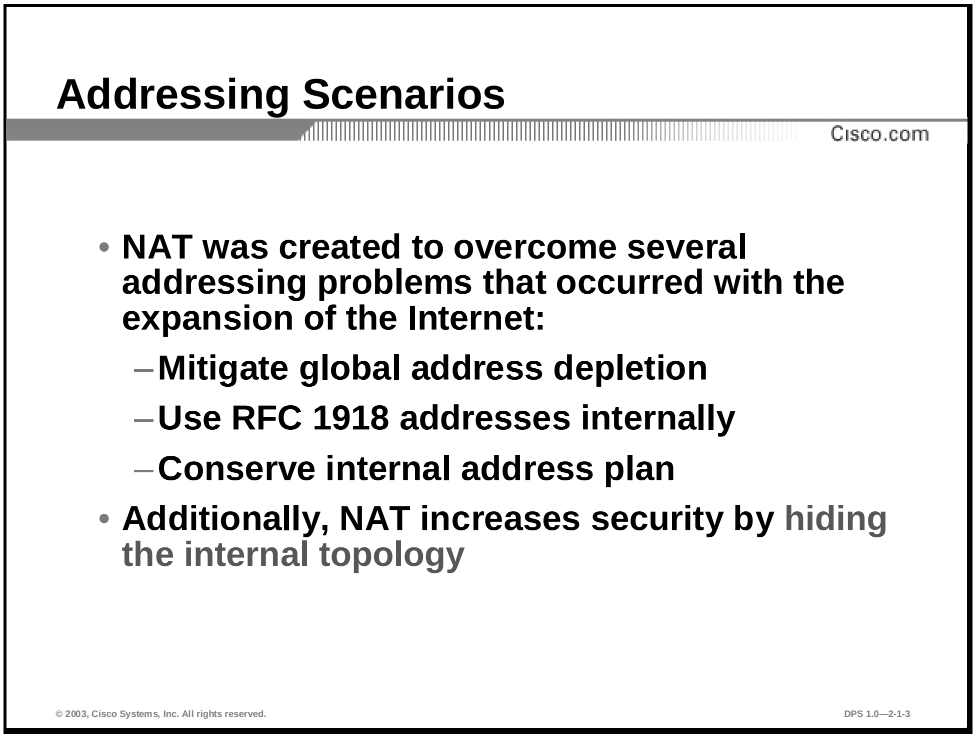
**This lesson includes these sections:**

- **Addressing Scenarios**
- **NAT Technologies and Implementation**
- **NAT Protocol Compatibility**
- **NAT Security Evaluation**

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—2-1-2

# Addressing Scenarios



## Addressing Scenarios

Cisco.com

- NAT was created to overcome several addressing problems that occurred with the expansion of the Internet:
  - Mitigate global address depletion
  - Use RFC 1918 addresses internally
  - Conserve internal address plan
- Additionally, NAT increases security by hiding the internal topology

DPS 1.0—2-1-3

## Objective

The section will enable the learner to identify common IP addressing situations that require NAT to be used.

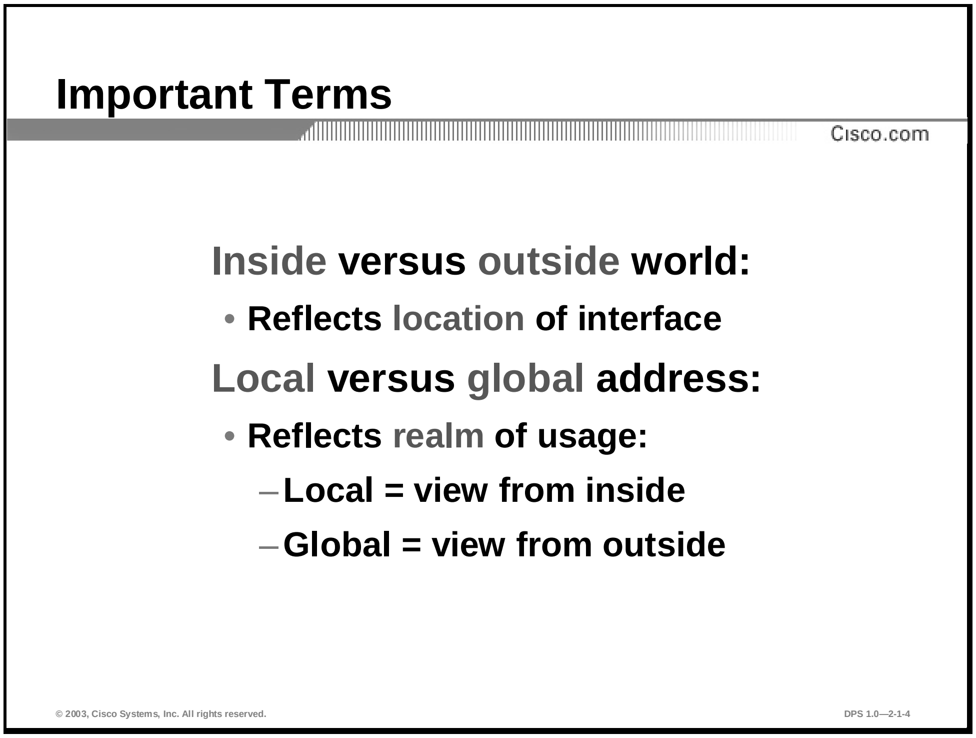
## Introduction

Invented in May 1994 by Paul Francis and Kjeld Borch Egevang, NAT became a popular technique to save official network addresses and to hide a network's topology from the Internet. P. Francis and K. Egevang have written several RFCs about NAT, most importantly RFC-1631: "The IP Network Address Translator (NAT)".

## When to use NAT?

In the modern world, NAT is critical to mitigate the global Internet address depletion. Very often, private networks are assigned network numbers from the address blocks defined in RFC 1918. Because these addresses are intended for local use only, NAT is required to connect to the Internet. NAT is sometimes used to preserve an enterprise's inside addresses, for example, when changing the Internet Service Provider (ISP).

**Note** The Cisco implementation of NAT can also be used for applications not related with address translation. For instance, another NAT usage is simple TCP load sharing (although Cisco recommends more sophisticated solutions, such as Cisco LocalDirector). This lesson only covers the main purpose of NAT, namely network address translation.



## Important Terms

**Inside versus outside world:**

- **Reflects location of interface**

**Local versus global address:**

- **Reflects realm of usage:**
  - **Local = view from inside**
  - **Global = view from outside**

DPS 1.0—2-1-4

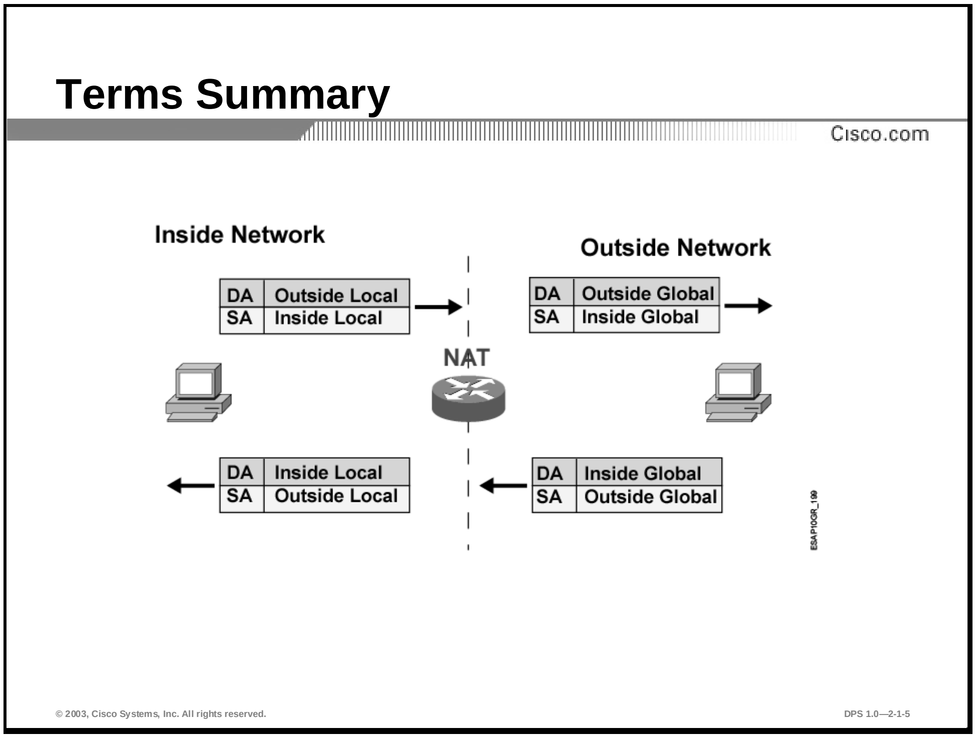
## Conventions

Four terms are central to NAT and unfortunately many people, and also some documents, confuse them. In order to understand all the mechanisms around NAT it is very important to know the exact meaning of these terms:

- Interfaces, and associated IP addresses, can be located “inside” or “outside” a network boundary. The inside area is typically an enterprise’s network, while the outside area is identical with the Internet or any other network not considered private.

- Addresses have either “local” or “global” meaning—no matter whether the corresponding interfaces are located inside or outside. Local addresses are used by inside hosts and routers, while global addresses can only be used by outside devices.

IP packets with global source or destination addresses do not occur in the inside network, and conversely, no IP packet exists in the outside network with local source or destination addresses.



## Terms Summary

Cisco.com

| | |
|---|---|
| DA | Outside Local |
| SA | Inside Local |

| | |
|---|---|
| DA | Outside Global |
| SA | Inside Global |

| | |
|---|---|
| DA | Inside Local |
| SA | Outside Local |

| | |
|---|---|
| DA | Inside Global |
| SA | Outside Global |

DPS 1.0—2-1-5

## Definition

All addresses are either inside or outside and either local or global.

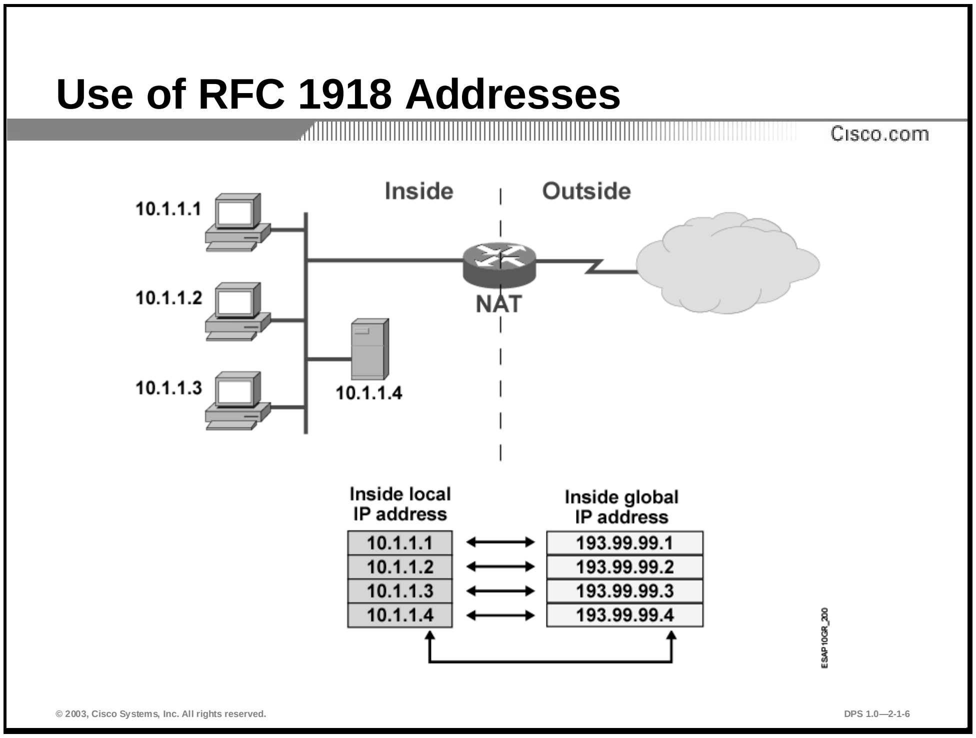
The NAT router is responsible for translating global addresses to local ones and vice versa. Following is a generic example—a host in the inside network, with a configured IP address of 10.1.1.1, wants to send packets to a host in the outside network, with a configured IP address of 209.23.5.2.

- The outside host has a global (from outside) view of the inside host; it will send packets to the **inside global** destination address. If the NAT device translates the inside host's IP address, then its **inside global** address will be different from its **inside local** (viewed from the inside) address. For example, the inside host 10.0.0.1 (**inside local** address) may be translated to the outside as 199.89.60.1 (**inside global** address).

- The inside host has a **local** (from inside) view of the **outside** host; it will connect to an **outside local** destination address. If the NAT device translates the outside host's IP address, then its **outside local** address will be different from its **outside global** (viewed from the outside) address. For example, the outside host 209.23.5.2 (**outside global** address) may be translated to the inside as 192.168.10.2 (**outside local** address).

- If the outside host sends packets back to the inside host, the inside global address 199.89.60.1 is used as destination address and the outside global address 209.23.5.2 is used as source address. The NAT router again translates both addresses to local numbers— that is, the inside local address 10.0.0.1 is written into the IP header as the destination address and the outside local address 192.168.10.2 replaces the source address.

**Note** Typically, when connecting to the Internet, only the inside local address is translated into an inside global address, while the outside local address is identical to the outside global.



## Use of RFC 1918 Addresses

| Inside local IP address | | Inside global IP address |
|---|---|---|
| 10.1.1.1 | ⟷ | 193.99.99.1 |
| 10.1.1.2 | ⟷ | 193.99.99.2 |
| 10.1.1.3 | ⟷ | 193.99.99.3 |
| 10.1.1.4 | ⟷ | 193.99.99.4 |

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—2-1-6

## RFC 1918 Address Blocks

RFC 1918 defines three private IP address blocks for local use. These addresses must not be used within the Internet. Internet boundary routers should filter any routing updates containing such addresses. Therefore, RFC 1918 addresses are safer for local use than official (“global”) addresses. In addition, invalid routes may occur in the Internet routing tables because global addresses are not filtered.

The RFC 1918 address blocks are:

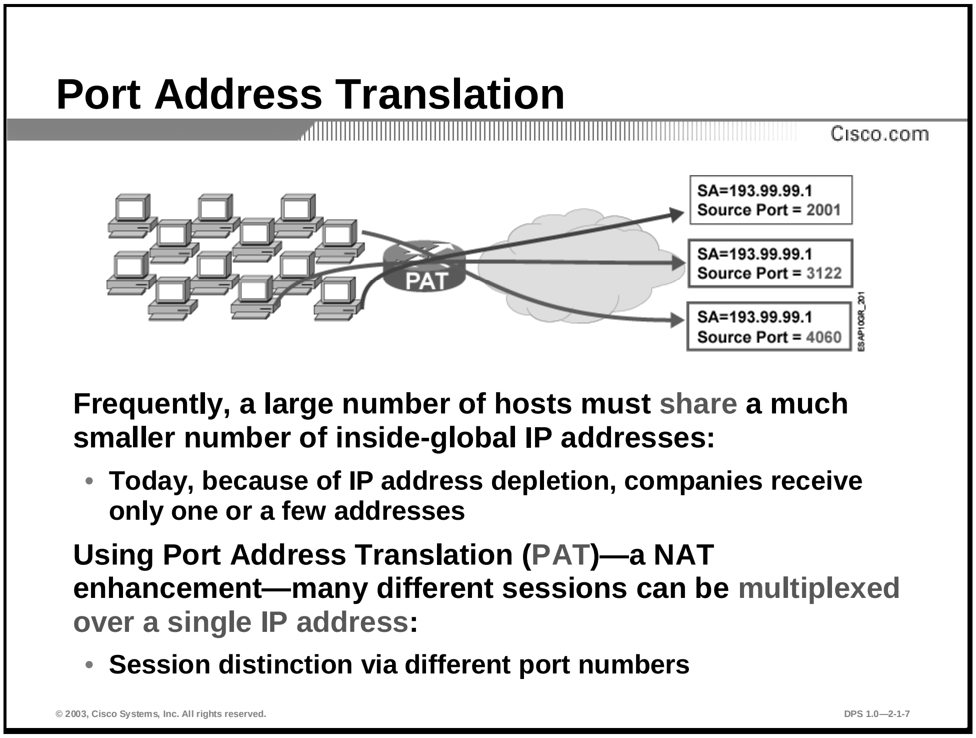
- 10.0.0.0 – 10.255.255.255 (prefix 10/8)
- 172.16.0.0 – 172.31.255.255 (prefix 172.16/12)
- 192.168.0.0 – 192.168.255.255 (prefix 192.168/16)

## Static and Dynamic NAT

Standard NAT maps each inside local address to one inside global address for each connection. The pool of inside global addresses must be sufficiently large to handle the maximum number of outgoing connections with different source addresses. This mapping can be defined either statically or dynamically:

- **Static NAT:** Associates dedicated inside local addresses with dedicated inside global addresses.

- **Dynamic NAT:** Selects addresses from an inside global address pool that has been configured in advance. The host-ID is conserved during the translation.



## Port Address Translation

Cisco.com

PAT
SA=193.99.99.1
Source Port = 2001
SA=193.99.99.1
Source Port = 3122
SA=193.99.99.1
Source Port = 4060
ESAP10GR_201

**Frequently, a large number of hosts must share a much smaller number of inside-global IP addresses:**

- **Today, because of IP address depletion, companies receive only one or a few addresses**

**Using Port Address Translation (PAT)—a NAT enhancement—many different sessions can be multiplexed over a single IP address:**

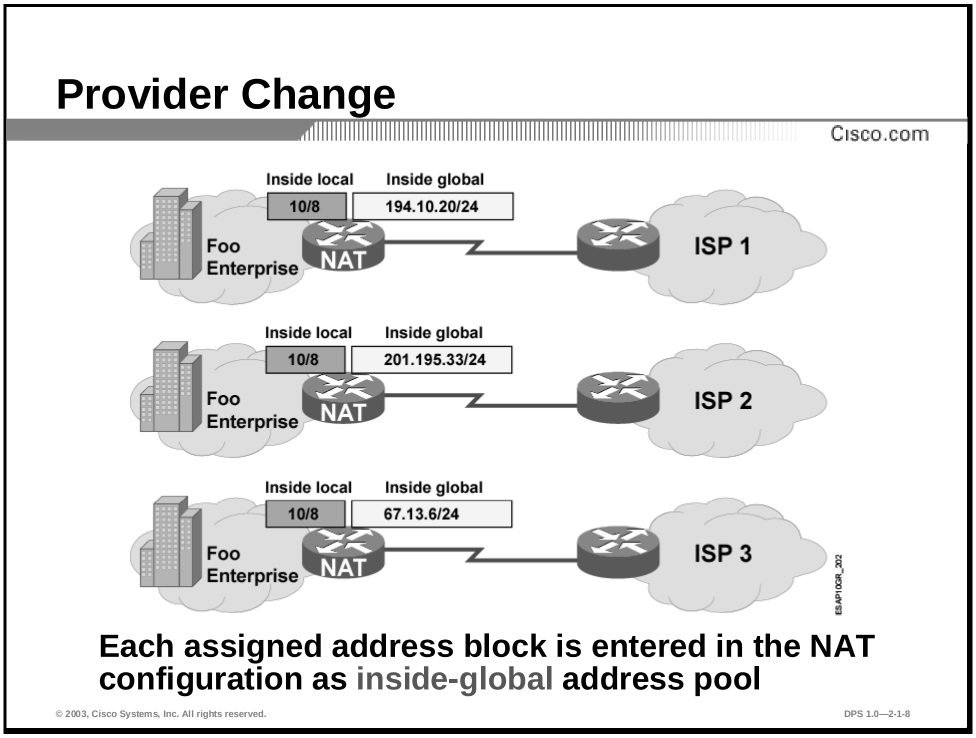
- **Session distinction via different port numbers**

DPS 1.0—2-1-7

## Sharing Inside Global Addresses

Typically, an enterprise network receives only a small number of addresses from its ISP, while the number of inside hosts is much higher. To resolve this situation, configure port address translation (PAT), which is an enhancement of NAT.

Using PAT, multiple connections originating from different hosts on the inside networks can be multiplexed by a single inside global IP address. The multiplexing identifier is the source port number. By default, the PAT router only changes the source port numbers when a collision of these numbers occurs.



**Provider Change**

Cisco.com

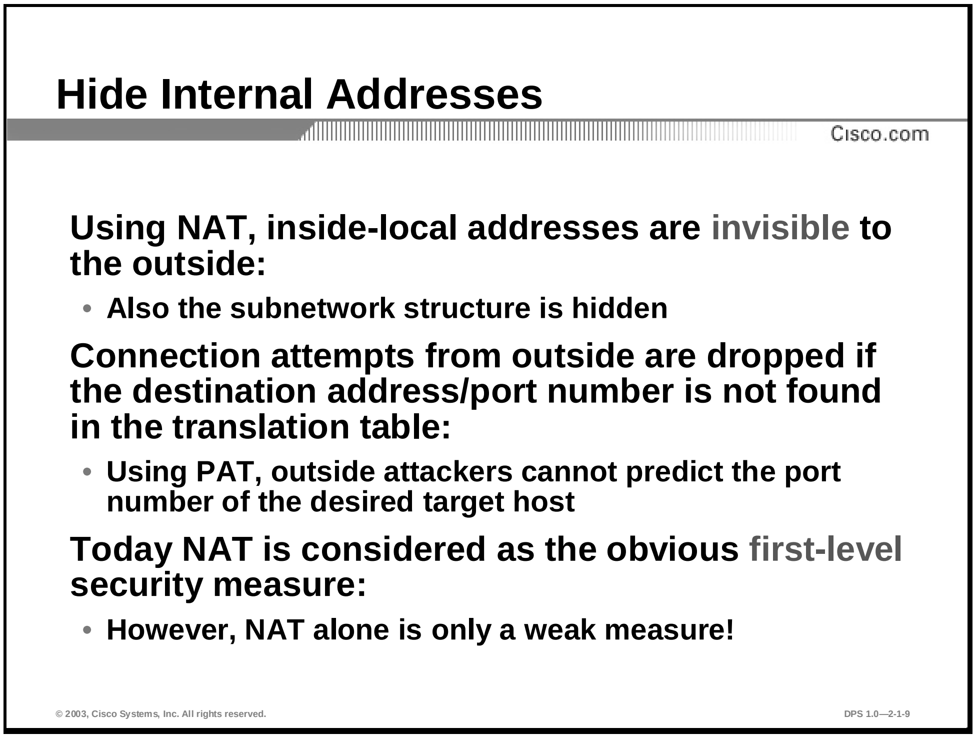
**Each assigned address block is entered in the NAT configuration as inside-global address pool**

DPS 1.0—2-1-8

## Save Time and Unnecessary Work

Typically, each ISP owns a dedicated classless interdomain routing (CIDR) address block (prefix). The ISP splits the block into many sub-address blocks and assigns them to customers. This means every ISP change requires a customer to completely renumber their inside networks and hosts. Instead, use NAT at the border to the ISP, to translate permanently assigned inside local addresses to ISP specific address prefixes.

RFC 1918 addresses are recommended as the inside local addresses. However, NAT gained greater importance when the registered addresses of large networks were released back into the global address space. For this scenario, the same addresses are used inside and outside, but NAT hides the inside ones and makes them only visible locally.



## Hide Internal Addresses

Cisco.com

Using NAT, inside-local addresses are invisible to the outside:

- Also the subnetwork structure is hidden

Connection attempts from outside are dropped if the destination address/port number is not found in the translation table:

- Using PAT, outside attackers cannot predict the port number of the desired target host

Today NAT is considered as the obvious first-level security measure:

- However, NAT alone is only a weak measure!

© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—2-1-9

## First-Level Security Measure

NAT is used as a “first-level” security measure, because it solves addressing problems, and also, by nature, hides inside addresses. Thus, an outside attacker who wants to harm hosts on the inside will not know the target addresses. Using PAT, the attacker will not know which combination of port number and IP address is currently assigned to a desired inside host. The NAT device drops any connection attempts to invalid sockets.

**Caution** NAT/PAT provides only weak security and sooner or later attackers discover the inside addresses, usually through trial and error testing. Additional security mechanisms such as advanced packet filtering, authentication, and encryption are therefore strongly recommended.

## Practice

Q1) Which of the following translations are realistic?

A) Outside Global to Inside Local

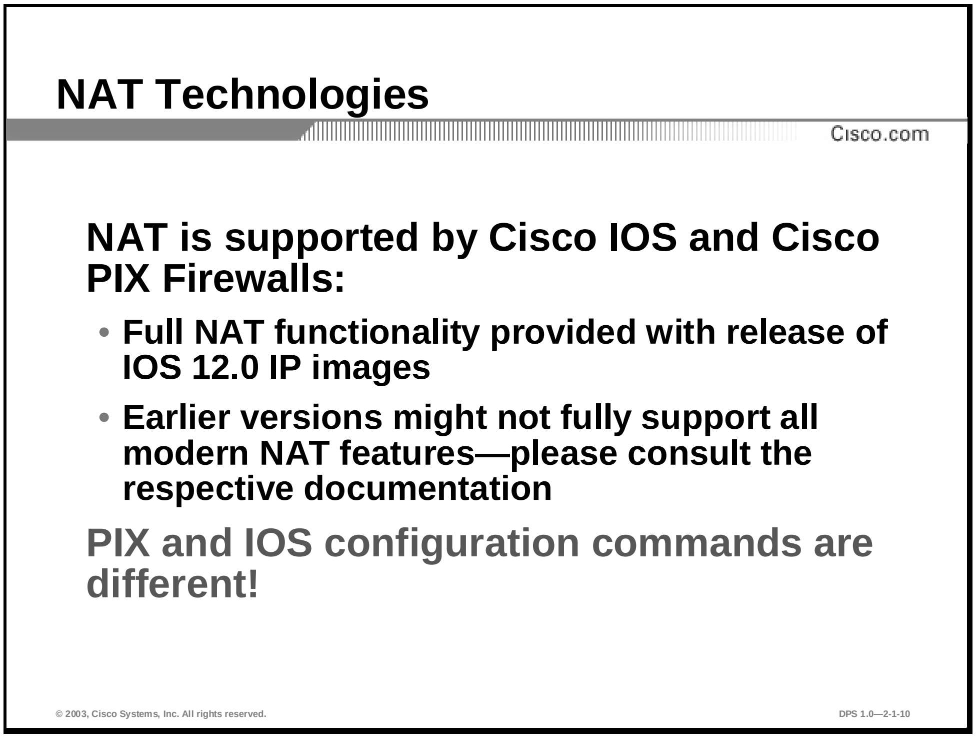
B) Inside Local to Outside Local

C) Outside Global to Outside Local

D) Inside Global to Inside Local

E) Inside Outside to Local Global

# NAT Technologies and Implementation



## NAT Technologies

NAT is supported by Cisco IOS and Cisco PIX Firewalls:

- Full NAT functionality provided with release of IOS 12.0 IP images
- Earlier versions might not fully support all modern NAT features—please consult the respective documentation

PIX and IOS configuration commands are different!

© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—2-1-10

## Objective

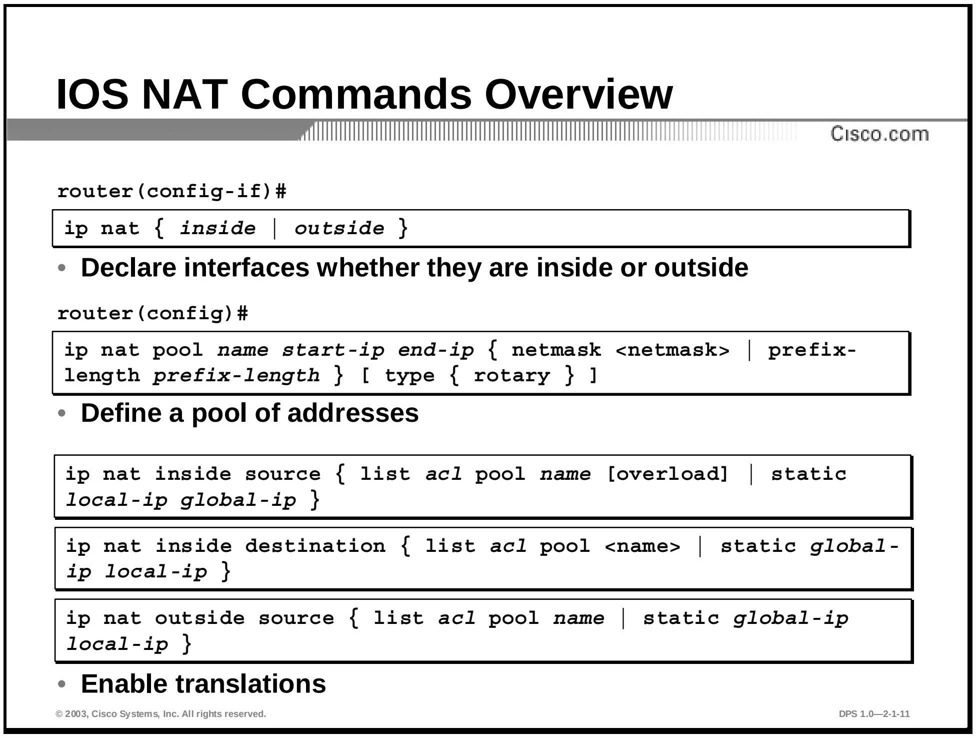
The section will enable the learner to identify technologies, used to perform NAT, explain their features and limitations, and select the appropriate technology for an organization's requirements.

## Introduction

NAT can be configured on Cisco routers running IOS and also on Cisco PIX Firewalls. This section highlights the differences in command syntax and explains the details of NAT operations necessary for troubleshooting. It also discusses features and limitations of NAT.

**Note** IOS versions prior to version 12 (IP) might not fully support all enhanced NAT features. For earlier versions consult the appropriate documentation.

**Note** When originally introduced in release 11.2 NAT was only available in the "Plus" images. With release 11.3, PAT was available in all IP images, with full NAT (1-1 and PAT) available only in "Plus" images. With release 12.0 all IP images provided full NAT functionality.



## IOS NAT Commands Overview

```
router(config-if)#
```

```
ip nat { inside | outside }
```

- Declare interfaces whether they are inside or outside

```
router(config)#
```

```
ip nat pool name start-ip end-ip { netmask <netmask> | prefix-
length prefix-length } [ type { rotary } ]
```

- Define a pool of addresses

```
ip nat inside source { list acl pool name [overload] | static
local-ip global-ip }
```

```
ip nat inside destination { list acl pool <name> | static global-
ip local-ip }
```

```
ip nat outside source { list acl pool name | static global-ip
local-ip }
```

- Enable translations

## Fundamental IOS NAT Commands

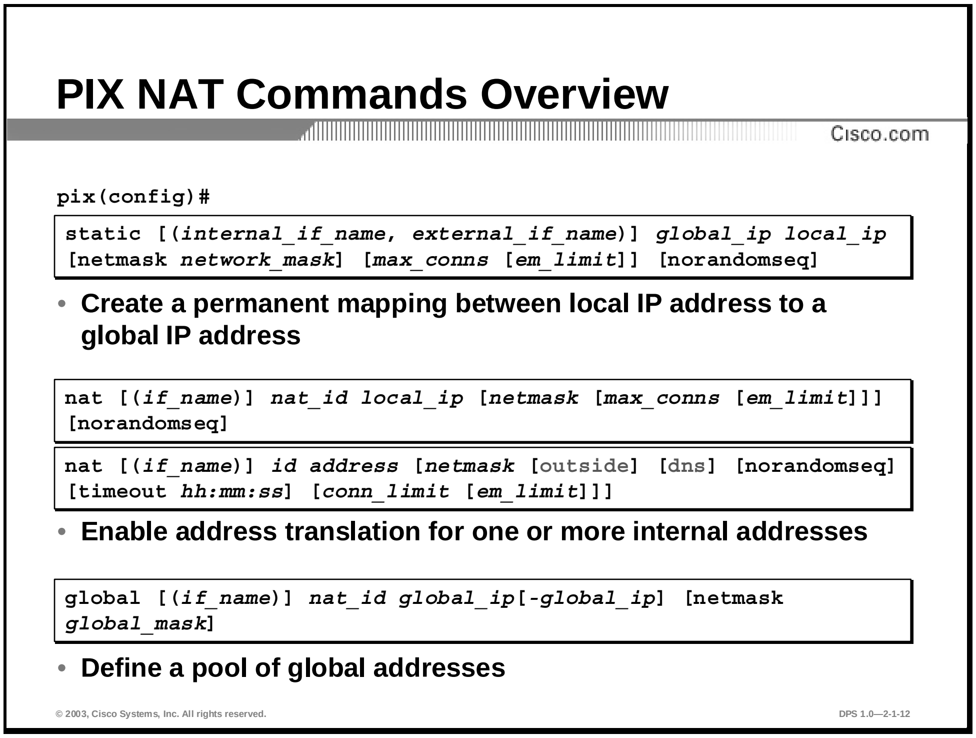
- The **ip nat** command marks interfaces identifying whether they are on the inside or the outside. Only packets arriving on a marked interface are subject to translation.

- The **ip nat pool** command defines a pool of addresses using the start address, end address, and netmask. These addresses will be allocated as needed.

- The **ip nat inside source** command enables dynamic translation. Packets from addresses that match those on the simple access list are translated using global addresses allocated from the named pool. The optional keyword **overload** enables port translation for User Datagram Protocol (UDP) and TCP. The second form of the command sets up a single static translation.

- The **ip nat inside destination** command is similar to the source translation command. The pool, however, needs to be a rotary-type pool for the dynamic destination translation to make any sense (rotary pool usage is not covered in this section).

- The first form of the **ip nat outside source** command enables dynamic translation. Packets from addresses that match those on the simple access list are translated using local addresses allocated from the named pool. The second form (static) of the command sets up a single static translation.



## PIX NAT Commands Overview

```
pix(config)#
```

```
static [(internal_if_name, external_if_name)] global_ip local_ip
[netmask network_mask] [max_conns [em_limit]] [norandomseq]
```

- **Create a permanent mapping between local IP address to a global IP address**

```
nat [(if_name)] nat_id local_ip [netmask [max_conns [em_limit]]]
[norandomseq]
```

```
nat [(if_name)] id address [netmask [outside] [dns] [norandomseq]
[timeout hh:mm:ss] [conn_limit [em_limit]]]
```

- **Enable address translation for one or more internal addresses**

```
global [(if_name)] nat_id global_ip[-global_ip] [netmask
global_mask]
```

- **Define a pool of global addresses**

DPS 1.0—2-1-12

# Fundamental PIX NAT Commands

## The static Command

The **static** command creates a permanent mapping between a local IP address *local_ip* and a global IP address *global_ip*:

- The *internal_if_name* is the inside (higher security level) network interface name.
- The *external_if_name* is the outside (lower security level) network interface name.
- The *network_mask* pertains to both *global_ip* and *local_ip*. For host addresses, use 255.255.255.255, except when subnetting is in effect; for example, 255.255.255.128. For network addresses, use the appropriate class mask; for example, for Class A networks, use 255.0.0.0.
- The *max_conns* value denotes the maximum number of connections permitted through the static at the same time.
- The value *em_limit* defines the so-called "embryonic connection limit". An embryonic connection is one that has started but not yet completed. Set this limit to prevent attack by a flood of embryonic connections. The default is 0, which means unlimited connections.
- The **norandomseq** statement disallows randomizing the TCP/IP packet's sequence number. Only use this option if another inline firewall is also randomizing sequence

numbers and the result is scrambling the data. Use of this option opens a security hole in the PIX Firewall.

**Note** Use static NAT to make inside hosts appear on the global network with a fixed address (use with servers accepting inbound connections).

## The nat Command

The **nat** command enables NAT for one or more inside addresses:

- The *if_name* defines the inside network interface name.
- Specify **0** for the *nat_id* to indicate that no address translation be used with *local_ip*. All **nat** command statements with the same *nat_id* are in the same nat group. Use the *nat_id* in the **global** command statement to bind a global address pool to this nat group, allowing dynamic NAT. The *nat_id* is an arbitrary positive number between 0 and two billion.
- The *local_ip* address specifies the inside network IP address to be translated. Use **0.0.0.0** to allow all hosts to start outbound connections. The **0.0.0.0** *local_ip* can be abbreviated as **0**.
- The *netmask* value defines a network mask for *local_ip*. Use **0.0.0.0** to allow all outbound connections to translate with IP addresses from the global pool.
- The *max_conns* value specifies the number pf maximum TCP connections permitted from the interface specified.
- The *em_limit* again sets the embryonic connection limit. The default is 0, which means unlimited connections. Set it lower for slower systems, higher for faster systems.
- The **norandomseq** again disallows randomization of the TCP packet's sequence number.

**Note** Starting with PIX version 6.2 outside NAT can be configured using the **nat** command together with the **outside** designator. This enhancement allows the definition of the outside global addresses to be translated to outside local addresses. The pool of outside local addresses can be defined using the **global** command applied on an inside interface. Additionally, the keyword **dns** has been introduced to the **nat** command to enable DNS interception.

## The global Command

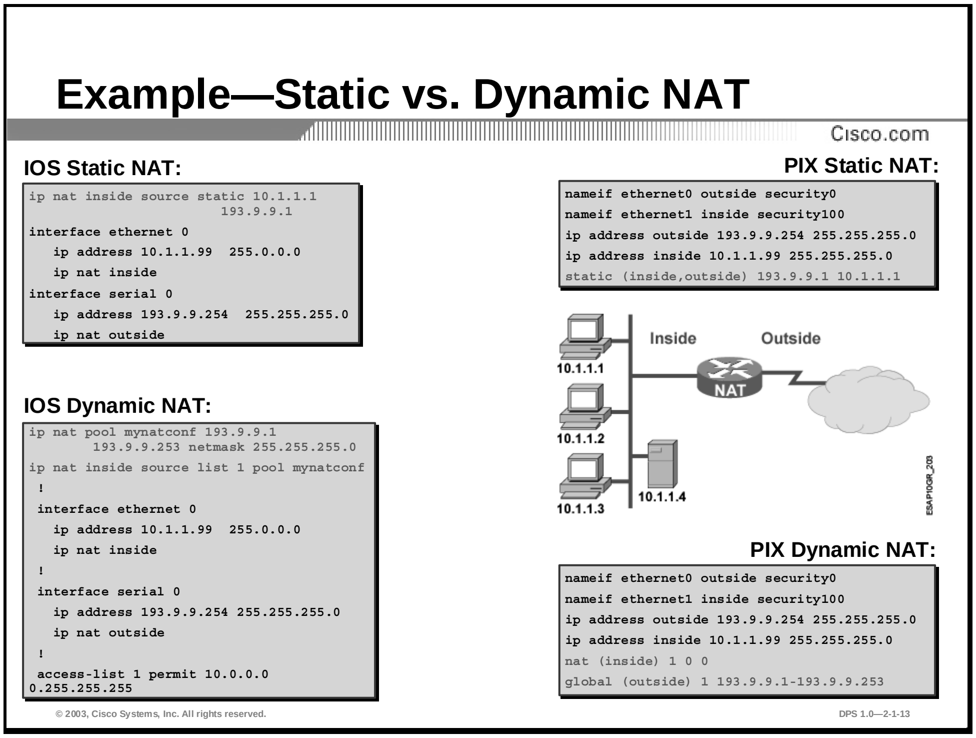
The **global** command defines a pool of global addresses bound to an outside interface *if_nam*:

- The *nat_id* is a positive number shared with the **nat** command that groups the **nat** and **global** command statements together. The valid ID numbers can be any positive number up to 2,147,483,647.

- The *global_ip* value defines one or more global IP addresses that the PIX Firewall shares among its connections. To specify a range of IP addresses, separate the addresses with a dash (-). To create a PAT **global** command statement, specify a single IP address. One PAT **global** command statement per interface is available. A PAT can support up to 65,535 xlate objects.

- The reserved word **netmask** prefaces the network *global_mask* variable, which sets a network mask for *global_ip*.

**Note** A “translation slot” (xlate slot) is a PIX/OS data structure used to describe active translations. A “connection slot” is a PIX/OS data structure used to describe an active connection. The “translation table” (xlate table) stores all active translation and connection slot objects. With the PIX, translation rules are always configured between pairs of interfaces. When a packet enters the PIX, the PIX determines the incoming and outgoing interface and translates the packet according to the translation rules between those interfaces. If a packet does not match a translation slot in the xlate table it cannot be switched across the PIX. If there is no translation slot the PIX will try to create a translation slot from its translation rules. If it fails to do so, the packet will be dropped.



## Example—Static vs. Dynamic NAT

Cisco.com

IOS Static NAT:

```
ip nat inside source static 10.1.1.1
                          193.9.9.1
interface ethernet 0
   ip address 10.1.1.99  255.0.0.0
   ip nat inside
interface serial 0
   ip address 193.9.9.254  255.255.255.0
   ip nat outside
```

IOS Dynamic NAT:

```
ip nat pool mynatconf 193.9.9.1
        193.9.9.253 netmask 255.255.255.0
ip nat inside source list 1 pool mynatconf
 !
 interface ethernet 0
   ip address 10.1.1.99  255.0.0.0
   ip nat inside
 !
 interface serial 0
   ip address 193.9.9.254 255.255.255.0
   ip nat outside
 !
 access-list 1 permit 10.0.0.0
0.255.255.255
```

PIX Static NAT:

```
nameif ethernet0 outside security0
nameif ethernet1 inside security100
ip address outside 193.9.9.254 255.255.255.0
ip address inside 10.1.1.99 255.255.255.0
static (inside,outside) 193.9.9.1 10.1.1.1
```

PIX Dynamic NAT:

```
nameif ethernet0 outside security0
nameif ethernet1 inside security100
ip address outside 193.9.9.254 255.255.255.0
ip address inside 10.1.1.99 255.255.255.0
nat (inside) 1 0 0
global (outside) 1 193.9.9.1-193.9.9.253
```

DPS 1.0—2-1-13

## Example Description

The picture illustrates an example of how to use the PIX/NAT commands. The hosts on the inside network are numbered with a 10.1.1/24 prefix. Any packets to the outside world should be translated to the inside global prefix 193.9.9/24. Both a static and a dynamic configuration example are given for both Cisco IOS routers and Cisco PIX Firewalls.

The **nat (inside) 1 0 0** permits all inside users to start outbound connections using the translated IP addresses from a global pool. Sometimes it is desirable to disable NAT for certain hosts and have all the IP addresses stay the same when traversing the PIX. Configure this option with the use of a *nat_id* of zero.

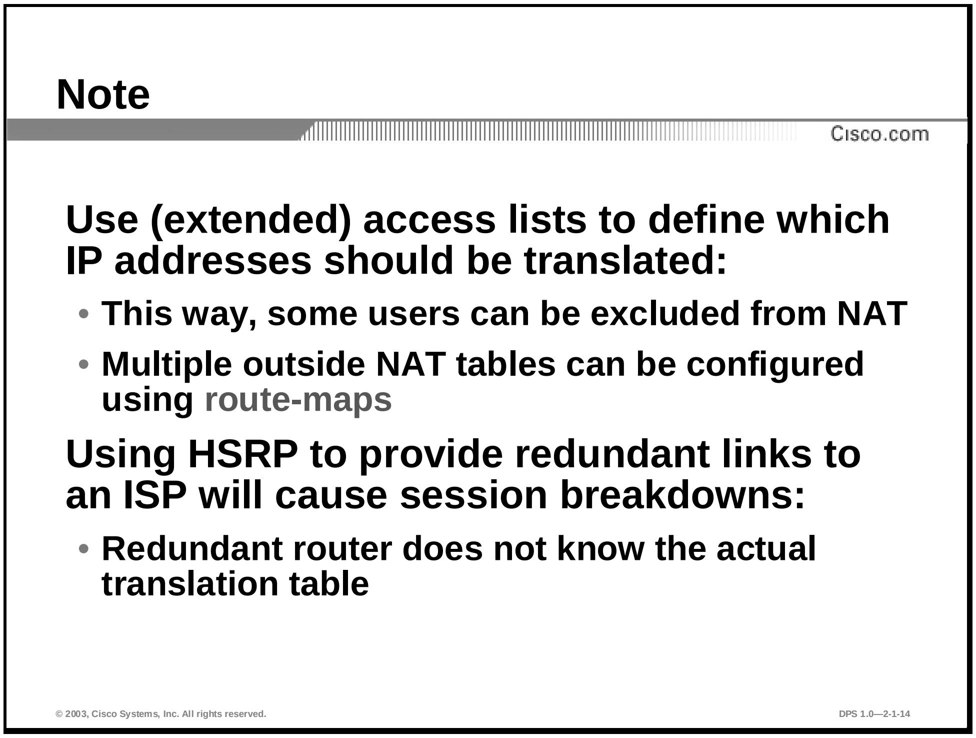
**Note** Where address specification is needed, the PIX/OS parser generally treats 0 as 0.0.0.0.

When configuring a NAT address pool, the subnet mask is used to sanity-check the addresses allocated from the pool. This way, NAT will not, for example, allocate the subnet broadcast address. The subnet mask must match the size of the subnet into which you are translating.

## Xlate Timeout

How long does a dynamically created translation slot remain active? If there is no traffic over this slot for a predefined idle-timeout period, the xlate slot is removed and the global address is returned to the pool. By default, the idle-timeout period is set to 24 hours, but can be configured using this command:

**timeout xlate** *hh:mm:ss*



## Note

Cisco.com

Use (extended) access lists to define which IP addresses should be translated:

- This way, some users can be excluded from NAT
- Multiple outside NAT tables can be configured using route-maps

Using HSRP to provide redundant links to an ISP will cause session breakdowns:

- Redundant router does not know the actual translation table

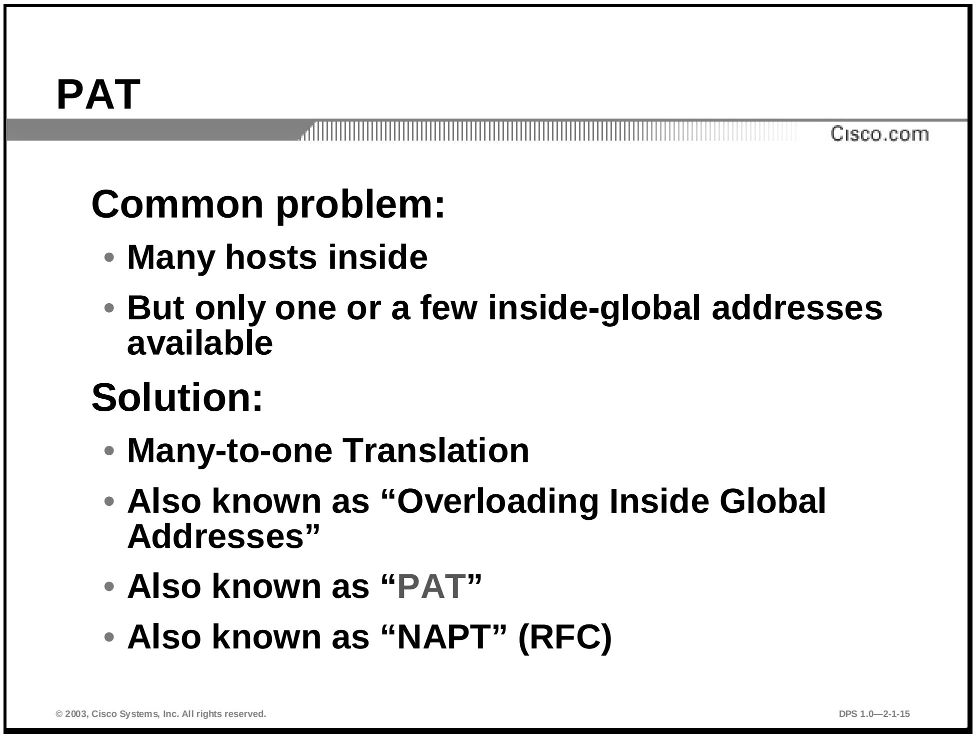
© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—2-1-14

# Translation Rules

To define the ‘rules’ for which IP device(s) gets translated use Access Lists, Extended Access Lists, and Route Maps.

| | |
|---|---|
| **Note** | Specify the network address and appropriate subnet mask instead of using the keyword “any” in place of the network address and subnet mask. This approach is highly recommended. |



## PAT

Cisco.com

**Common problem:**

- **Many hosts inside**
- **But only one or a few inside-global addresses available**

**Solution:**

- **Many-to-one Translation**
- **Also known as “Overloading Inside Global Addresses”**
- **Also known as “PAT”**
- **Also known as “NAPT” (RFC)**

DPS 1.0—2-1-15

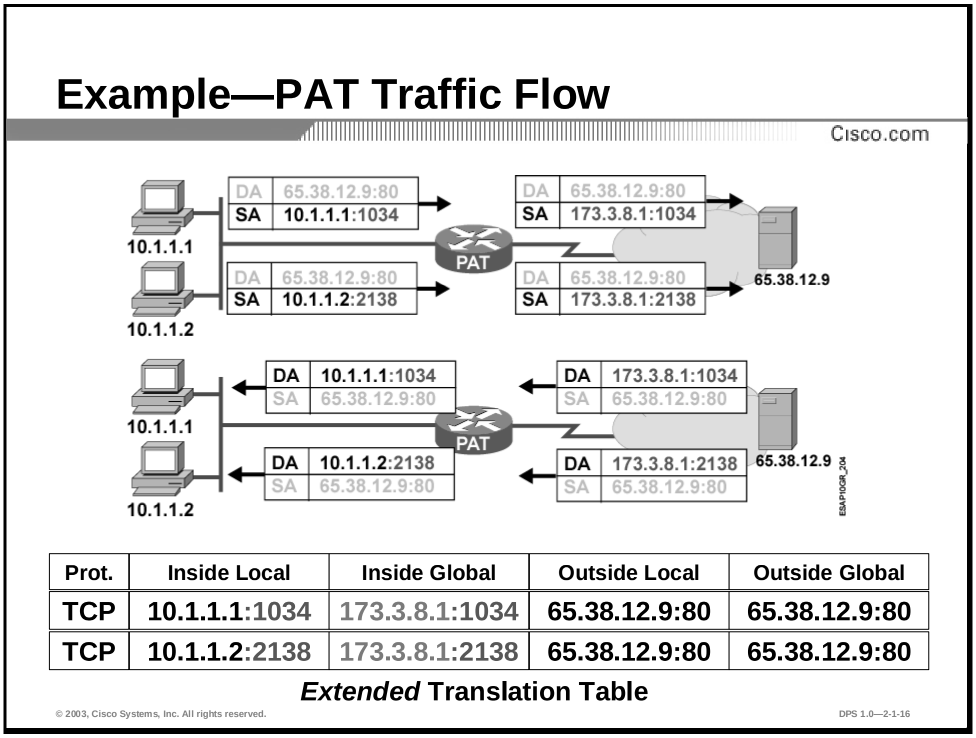
## Savior of the Internet Address Space

Traditional NAT limits the number of connections according to the number of assigned global addresses. Except for the early days of the Internet, most Intranet traffic is destined for outside destinations. Because of this a NAT solution must prevent blocking situations in case the number of hosts that want to establish a connection outside is greater than the number of assigned global addresses. This is achieved by a many-to-one translation.

## Definition

A many-to-one translation is accomplished by identifying each traffic according to the source port numbers. This method is commonly known as “address overloading” or PAT. The Internet Engineering Task Force (IETF) documents also use the abbreviation NAPT. In the Linux world this is known as “IP masquerading”.

**Note** This lesson only uses the term PAT to describe a many-to-one address translation.



## Example—PAT Traffic Flow

Cisco.com

| Prot. | Inside Local | Inside Global | Outside Local | Outside Global |
|---|---|---|---|---|
| TCP | 10.1.1.1:1034 | 173.3.8.1:1034 | 65.38.12.9:80 | 65.38.12.9:80 |
| TCP | 10.1.1.2:2138 | 173.3.8.1:2138 | 65.38.12.9:80 | 65.38.12.9:80 |

***Extended*** **Translation Table**

DPS 1.0—2-1-16

## PAT Mechanism

Traffic originating at different local hosts, but translated to the same inside global address, is differentiated using the source port number.

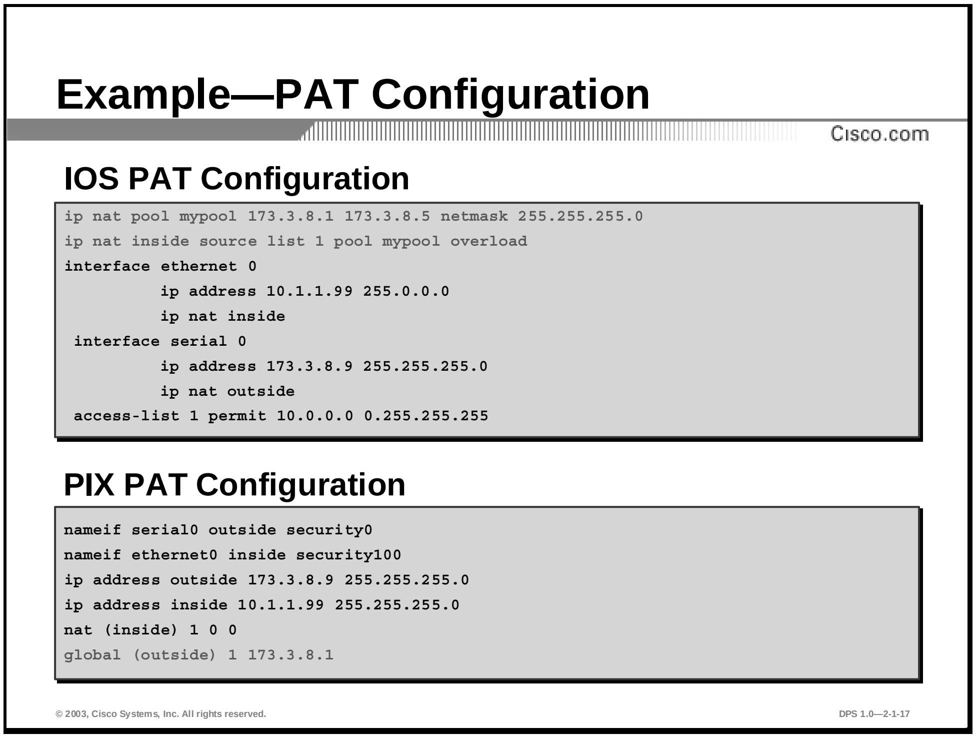
In the example both inside hosts (10.1.1.1 and 10.1.1.2) connect to the same outside server (65.38.12.9). Both connections appear on the outside as if they originated at the same source address (173.3.8.1), however, the port numbers (1034 and 2138) separate the sockets from each other.

The TCP and UDP port number range allows up to 65,536 number per IP address. This number is the upper limit for simultaneous transmissions per inside-global IP address.

If the port numbers run out, PAT moves to the next IP address and tries to allocate the original source port again. This continues until all available ports and IP addresses are utilized. Eventually, the PAT device runs out of IP addresses. An Internet Control Message Protocol (ICMP) “Host Unreachable message” is sent and the packets are dropped.

## Port Number Assignment Strategy

PAT divides the available ports per global IP Address into 3 ranges: 0 – 511, 512 – 1023, and 1024 – 65535. Cisco IOS and PIX/OS will attempt to assign the same port value of the original request. However, if the original source port has already been used it will start scanning from the beginning of the particular port range to find the first available port and assign it to the conversation.



## Example—PAT Configuration

Cisco.com

### IOS PAT Configuration

```
ip nat pool mypool 173.3.8.1 173.3.8.5 netmask 255.255.255.0
ip nat inside source list 1 pool mypool overload
interface ethernet 0
          ip address 10.1.1.99 255.0.0.0
          ip nat inside
 interface serial 0
          ip address 173.3.8.9 255.255.255.0
          ip nat outside
 access-list 1 permit 10.0.0.0 0.255.255.255
```

### PIX PAT Configuration

```
nameif serial0 outside security0
nameif ethernet0 inside security100
ip address outside 173.3.8.9 255.255.255.0
ip address inside 10.1.1.99 255.255.255.0
nat (inside) 1 0 0
global (outside) 1 173.3.8.1
```

DPS 1.0—2-1-17

# PAT Configuration Guidelines

Configuring PAT on an IOS device includes three important steps:

**Step 1** Create a NAT pool of one or more addresses using the **ip nat pool** command.

**Step 2** Specify network address translation for inside local addresses using the **ip nat inside source** command. After specifying the address pool (defined in step 1), use the keyword **overload** to enable PAT.

**Step 3** Define inside and outside interfaces as usual when configuring NAT.

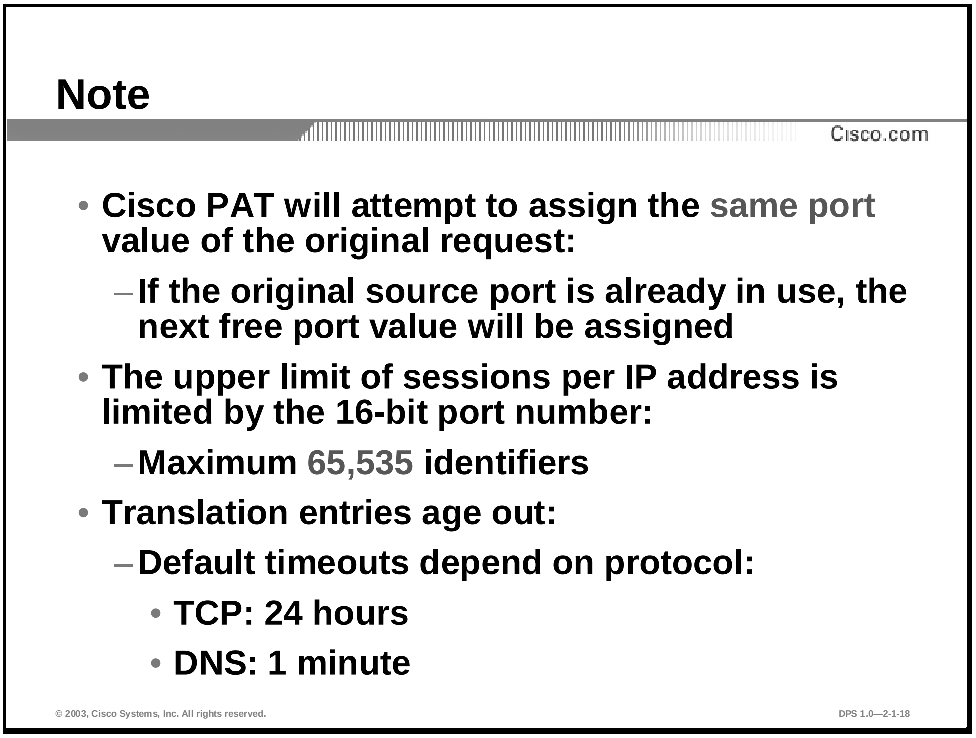
Configuring PAT on a PIX Firewall is very similar to configuring a traditional NAT. The only difference is to specify only one global address with the global command, instead of an address pool. There are three steps:

**Step 1** Define the inside and outside interfaces as usual when configuring NAT

**Step 2** Enable NAT using the **nat** command, hereby specifying the inside interface

**Step 3** Specify a single IP address with the **global** command applied on the outside interface to enable PAT

Dynamic NAT is performed if a global command uses a range of IP addresses. PAT is performed if a global command only specifies a single IP address. The example represents a typical, and recommended, configuration solution. The first global command specifies a limited IP range and a second global command is used for backup purposes. This allows PAT to handle additional outbound connections.



## Note

- Cisco PAT will attempt to assign the same port value of the original request:
  - If the original source port is already in use, the next free port value will be assigned
- The upper limit of sessions per IP address is limited by the 16-bit port number:
  - Maximum 65,535 identifiers
- Translation entries age out:
  - Default timeouts depend on protocol:
    - TCP: 24 hours
    - DNS: 1 minute

DPS 1.0—2-1-18

# Facts

Different vendors of PAT solutions have implemented different port assignment strategies. The Cisco PAT strategy is to keep the same port value during translation where possible. When N inside hosts use the same source port numbers the PAT-routers will increase N-1 of these identical source port numbers to the next free values.

The 16-bit port number used in the TCP or UDP header limits the maximum number of sessions per IP address. Therefore, each IP address can handle up to 65,535 sessions. The maximum number of configurable NAT IP pools is actually only limited by DRAM size available.

# PAT Timeouts

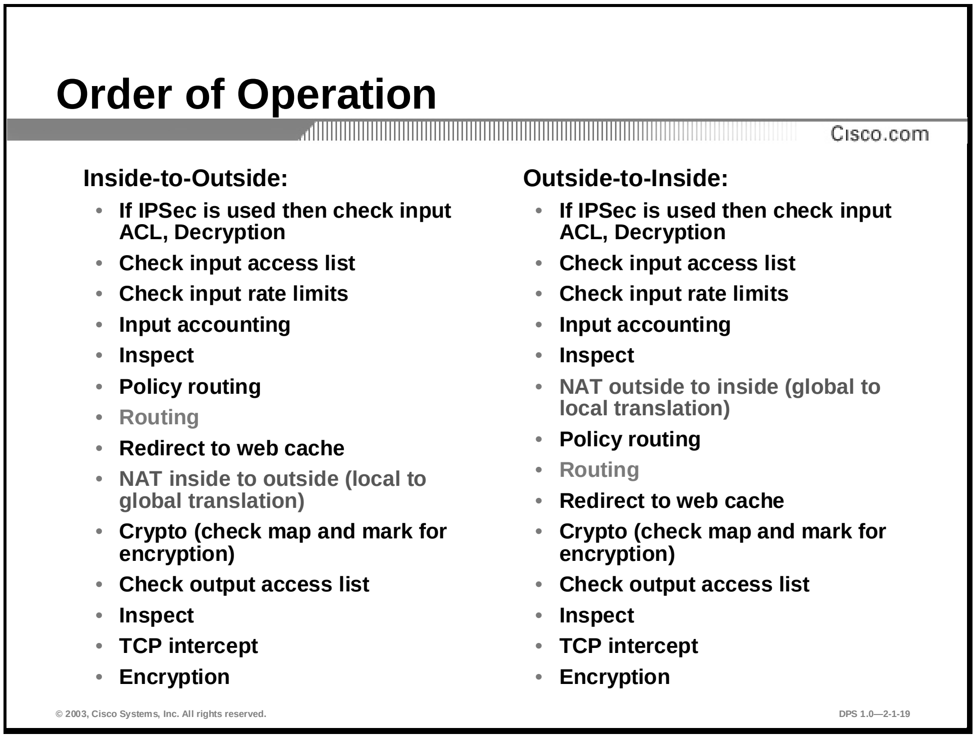
The dynamic translation table (or *translation matrix*) ages out after some time. The default timeouts are:

- **Non-DNS UDP—**5 minutes
- **DNS**—1 minute
- **TCP**—24 hours
- **TCP RST/FIN—**1 minute

To change these timeouts use the **ip nat translation** [***keyword***] command, where the ***keyword*** can be one of the following, according to the timeout types listed above:

- udp-timeout
- dns-timeout
- tcp-timeout
- finrst-timeout

The timeout period is 24 hours per default if overloading is not configured.



## Order of Operation

**Inside-to-Outside:**

- **If IPSec is used then check input ACL, Decryption**
- **Check input access list**
- **Check input rate limits**
- **Input accounting**
- **Inspect**
- **Policy routing**
- **Routing**
- **Redirect to web cache**
- **NAT inside to outside (local to global translation)**
- **Crypto (check map and mark for encryption)**
- **Check output access list**
- **Inspect**
- **TCP intercept**
- **Encryption**

**Outside-to-Inside:**

- **If IPSec is used then check input ACL, Decryption**
- **Check input access list**
- **Check input rate limits**
- **Input accounting**
- **Inspect**
- **NAT outside to inside (global to local translation)**
- **Policy routing**
- **Routing**
- **Redirect to web cache**
- **Crypto (check map and mark for encryption)**
- **Check output access list**
- **Inspect**
- **TCP intercept**
- **Encryption**

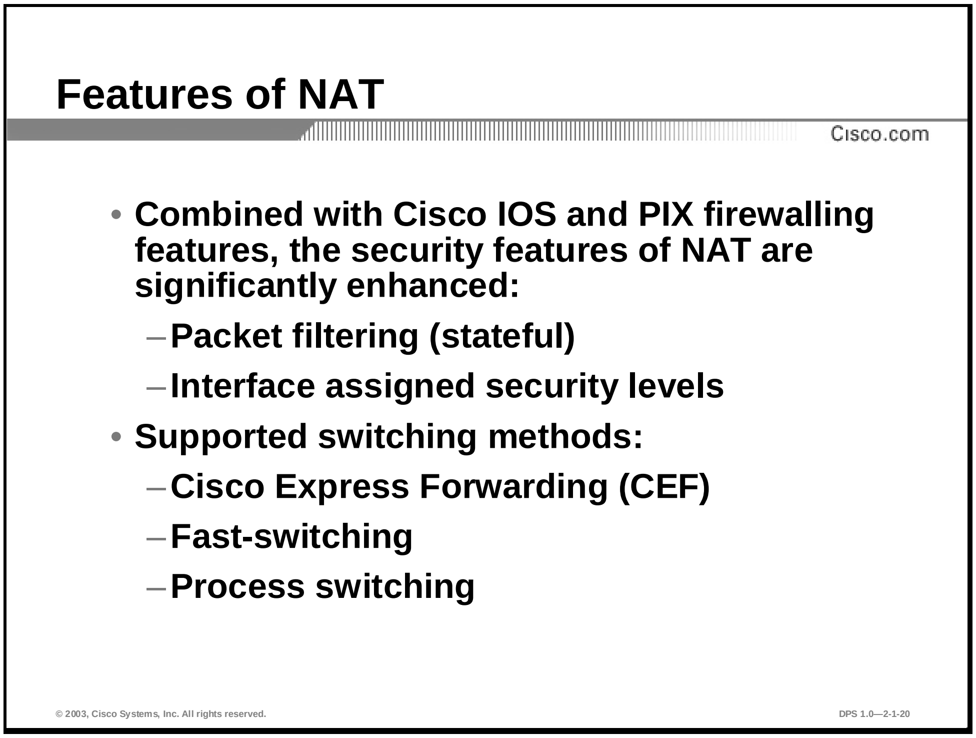
© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—2-1-19

## NAT and Routing

When a packet is traversing inside to outside, a NAT router checks its routing table for a route to the outside address before it continues to translate the packet. It is therefore important that the NAT router has a valid route to the outside network. The route to the destination network must be known through an interface that is defined as “NAT outside” in the router configuration.

It is important to note that the return packets are translated before they are routed. Therefore, the NAT router must also have a valid route for the inside local address in its routing table.



## Features of NAT

Cisco.com

- Combined with Cisco IOS and PIX firewalling features, the security features of NAT are significantly enhanced:
  - Packet filtering (stateful)
  - Interface assigned security levels
- Supported switching methods:
  - Cisco Express Forwarding (CEF)
  - Fast-switching
  - Process switching

© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—2-1-20

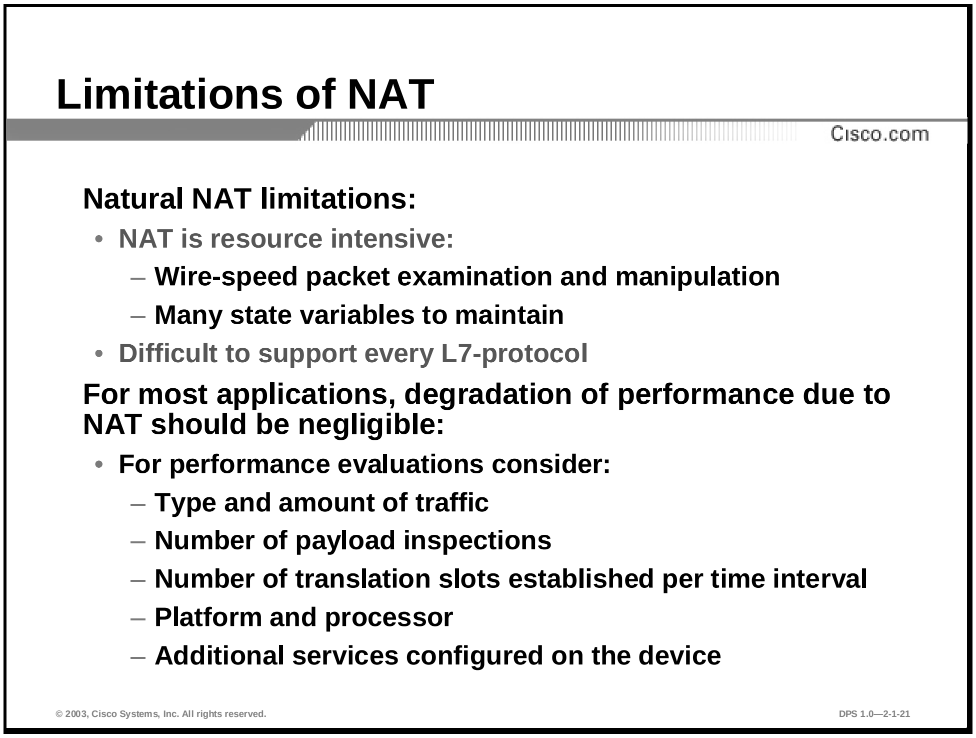
## Adding Value

NAT becomes more powerful when combined with modern packet filtering and security enhancements as implemented in Cisco IOS and PIX Firewalls. Of course, additional functionality increases complexity and an administrator must be aware of side effects and order of operation.

**Note** An inside-to-outside translation occurs *after* routing—that is, translation is not performed if there no valid route is found. Similarly, an outside-to-inside translation occurs *before* routing—that is, any output access lists are checked afterwards.

## Maximum Performance

Obviously, NAT decreases the performance of routers because it requires a large amount of additional processing power. The Cisco implementation of NAT is designed for maximum performance, thus packets are still switched using either Cisco Express Forwarding (CEF) or fast switching—process switching is also supported.



## Limitations of NAT

Cisco.com

**Natural NAT limitations:**

- NAT is resource intensive:
  - Wire-speed packet examination and manipulation
  - Many state variables to maintain
- Difficult to support every L7-protocol

**For most applications, degradation of performance due to NAT should be negligible:**

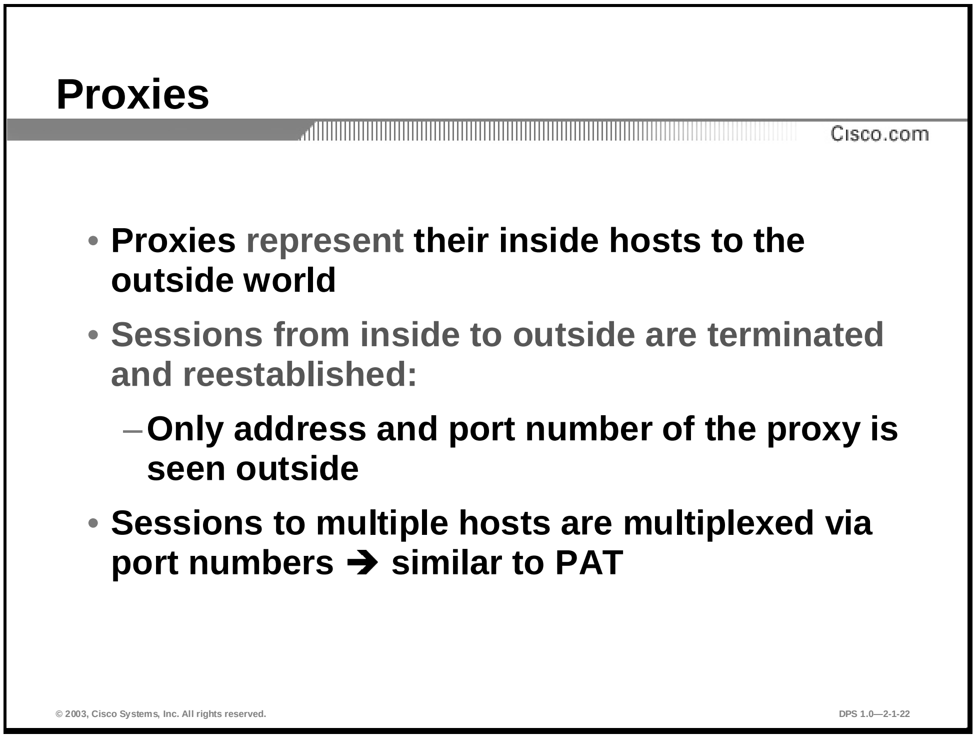
- For performance evaluations consider:
  - Type and amount of traffic
  - Number of payload inspections
  - Number of translation slots established per time interval
  - Platform and processor
  - Additional services configured on the device

DPS 1.0—2-1-21

## Natural Limits

Of course, there are natural NAT limitations, created by the amount and type of traffic. Many Layer-7 (L7) protocols transport address information, therefore NAT must also examine packet payloads and apply some manipulations to them. This means that some state variables have to be stored for each session.

The NAT session limit is bounded by the amount of available DRAM. Each NAT translation consumes about 160 bytes of DRAM. As a result, 10,000 translations would consume approximately 1.6 MB. A typical routing platform or a PIX have more than enough memory to support thousands of NAT translations.



## Proxies

- Proxies represent their inside hosts to the outside world
- Sessions from inside to outside are terminated and reestablished:
  - Only address and port number of the proxy is seen outside
- Sessions to multiple hosts are multiplexed via port numbers ➔ similar to PAT

DPS 1.0—2-1-22

## Definition

Another term frequently used for perimeter devices is “proxy”. The term proxy simply means “instead of”. In this context, a device that has been configured as proxy terminates session originated on the inside and reestablishes them on the outside—on behalf of the inside hosts. Sessions to multiple hosts are multiplexed via port numbers, very similar to PAT. However, unlike PAT, the sessions are terminated inside the proxy. That is, a proxy must be aware of TCP sequence and acknowledgement numbering. It must also maintain the TCP-associated timers, buffers, and algorithms.

Although this is an artificial separation, it increases security because only the address information (IP address and port number) of the proxy is seen outside. Additionally, invalid packets can never reach the inside hosts because each inbound packet is terminated and recreated by the proxy. If an attacker creates dangerous packets (for example ping-of-death) they would only affect the proxy. For this reason, design proxies for maximum stability.

## Practice

Q1) Set the following actions for an inside-to-outside packet flow in the correct order:

A) NAT

B) check output ACL

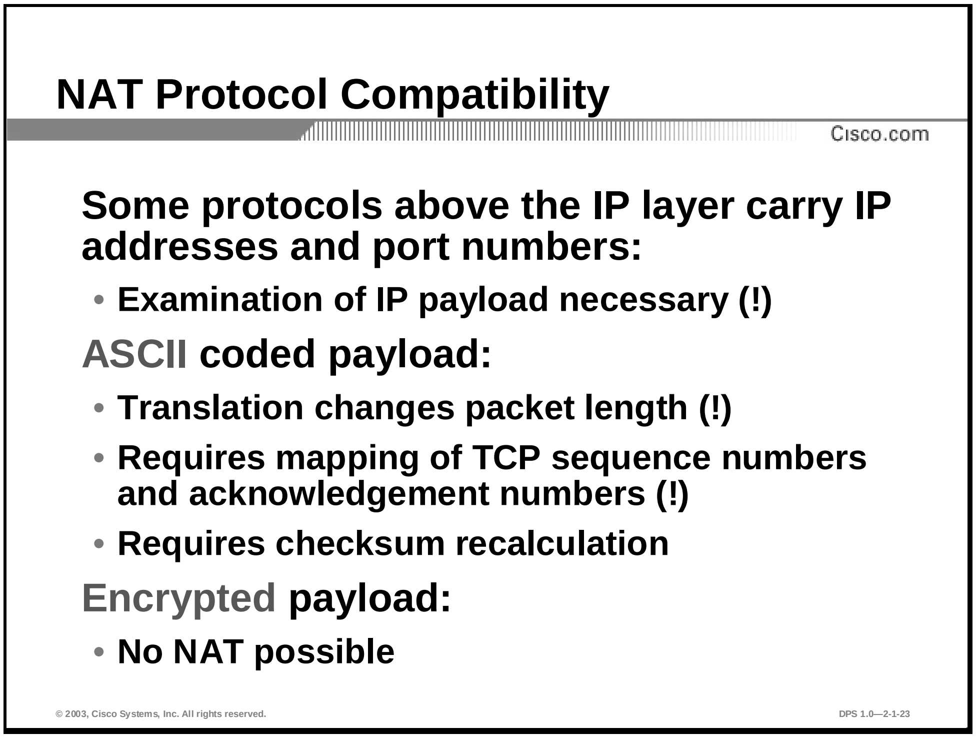
C) routing

D) policy routing

E) check input ACL

F) IPSec protection

# NAT Protocol Compatibility



## NAT Protocol Compatibility

Cisco.com

**Some protocols above the IP layer carry IP addresses and port numbers:**

- **Examination of IP payload necessary (!)**

**ASCII coded payload:**

- **Translation changes packet length (!)**
- **Requires mapping of TCP sequence numbers and acknowledgement numbers (!)**
- **Requires checksum recalculation**

**Encrypted payload:**

- **No NAT possible**

DPS 1.0—2-1-23

## Objective

This section will enable the learner to explain problems of compatibility between NAT and network applications.

## Introduction

Many application layer protocols carry address information. Therefore, NAT must also be applied to the packets’ payload. This section examines several of the important application protocols with regard to their NAT qualification.

## Payload Translation Challenge

Translating addresses carried by application protocols is more challenging than it appears, for the following reasons:

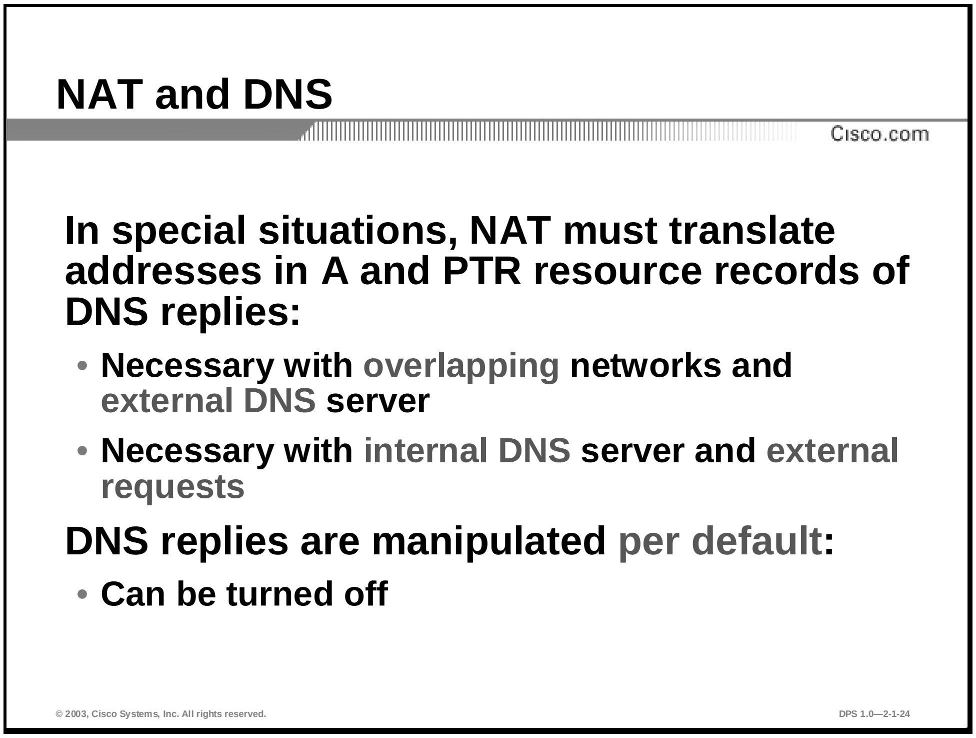
- Application protocols typically use a string-based segmentation, rather than byte-fixed fields. Therefore, IP addresses inside a payload are found in unpredictable positions.

- Internet application protocols typically use the ASCII code as the primitive presentation layer. As a result, the length of the address information might change when a translation is performed. In addition, the checksum, the TCP/UDP length field, and even more difficult, the TCP sequence and acknowledgement numbers must be adjusted—for each packet!

For all supported application protocols, Cisco NAT performs “stateful inspection” and maintains data structures for each session, where additional translation information is stored. For example, pointers to address fields and deltas for sequence numbering.

**Note** NAT cannot support applications that use encrypted payloads.

More and more new application protocols are appearing on the market, and in the Internet. NAT must therefore be updated frequently, or users inside the private network will be unable to use new applications.



## NAT and DNS

Cisco.com

**In special situations, NAT must translate addresses in A and PTR resource records of DNS replies:**

- **Necessary with overlapping networks and external DNS server**
- **Necessary with internal DNS server and external requests**

**DNS replies are manipulated per default:**

- **Can be turned off**

DPS 1.0—2-1-24

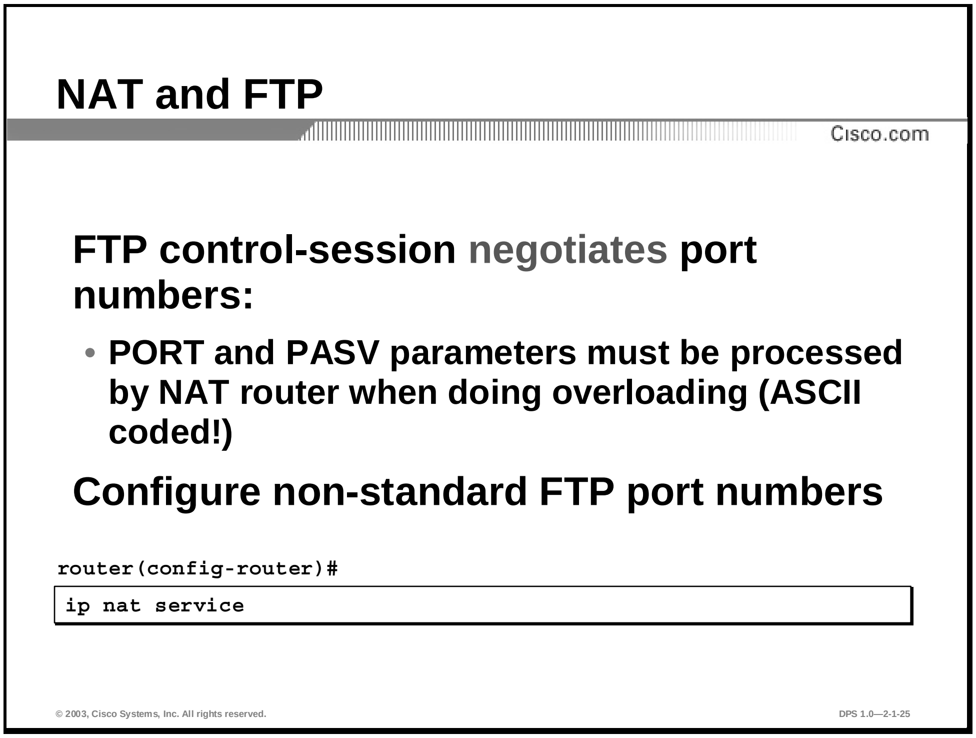
# Facts

The Domain Name System (DNS) protocol is perhaps the most important example of an L7 protocol that NAT has to intercept. Because DNS resolves hostnames into addresses, there are many situations where simple NAT might confuse the communication. Such examples include overlapping network addresses inside and outside, and scenarios with an internal DNS server that responses to external requests.

Cisco IOS and PIX Firewall NAT implementations translate the address(es) that appear in DNS responses to name lookups (A queries) and inverse lookups (PTR queries). Thus, if an outside host sends a name-lookup to a DNS server on the inside, and that server responds with a local address, the NAT code translates that local address to a global address. The opposite is also true, and is how IP addresses overlapping is supported: an inside host queries an outside DNS server, the response contains an address that matches the access list specified on the “outside source” command, so the code translates the outside global address to an outside local address.

All DNS resource records (RRs) that receive address translations in RR payloads are automatically set time-to-live (TTL) values to zero.

**Note** NAT is not applied on IP addresses embedded in DNS zone transfers.



## NAT and FTP

Cisco.com

**FTP control-session negotiates port numbers:**

- **PORT and PASV parameters must be processed by NAT router when doing overloading (ASCII coded!)**

**Configure non-standard FTP port numbers**

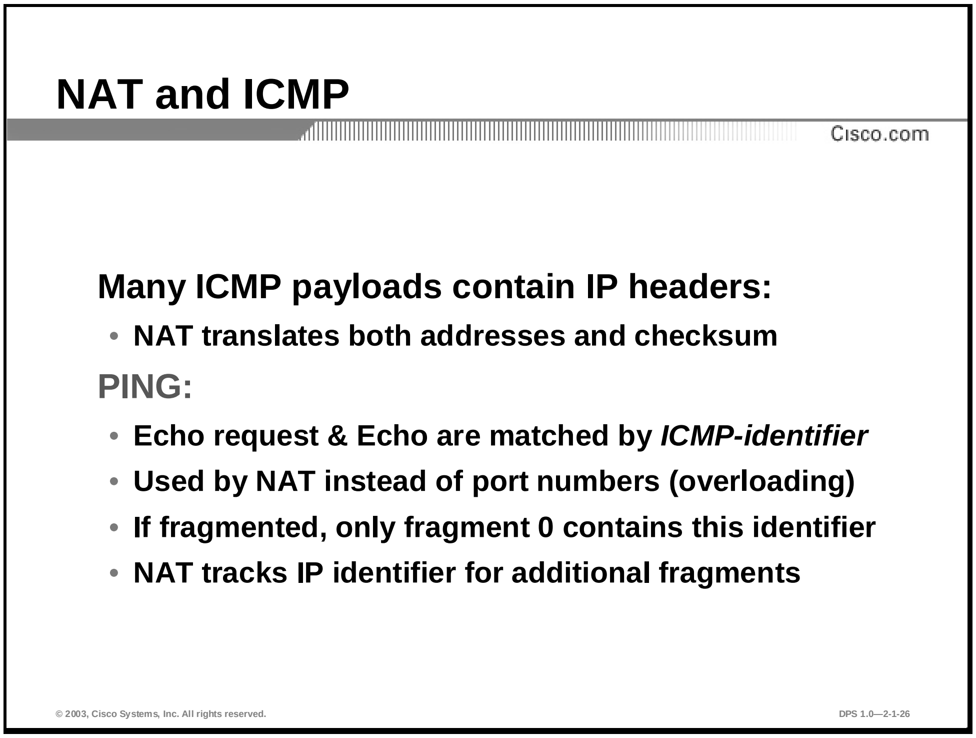
```
router(config-router)#
ip nat service
```

DPS 1.0—2-1-25

## Facts

FTP peers often negotiate port numbers to be used for the data TCP session. This causes problems for NAT.

FTP PORT and PASV parameters carry the IP addresses and port numbers. Unfortunately, this causes a problem because the addresses are in human readable ASCII format—the address length is variable! This affects the TCP segment length, and the sequence and acknowledgement numbers. Therefore, for the duration of the connection these parameters must be transformed.



## NAT and ICMP

Cisco.com

**Many ICMP payloads contain IP headers:**

- **NAT translates both addresses and checksum**

**PING:**

- **Echo request & Echo are matched by *ICMP-identifier***
- **Used by NAT instead of port numbers (overloading)**
- **If fragmented, only fragment 0 contains this identifier**
- **NAT tracks IP identifier for additional fragments**

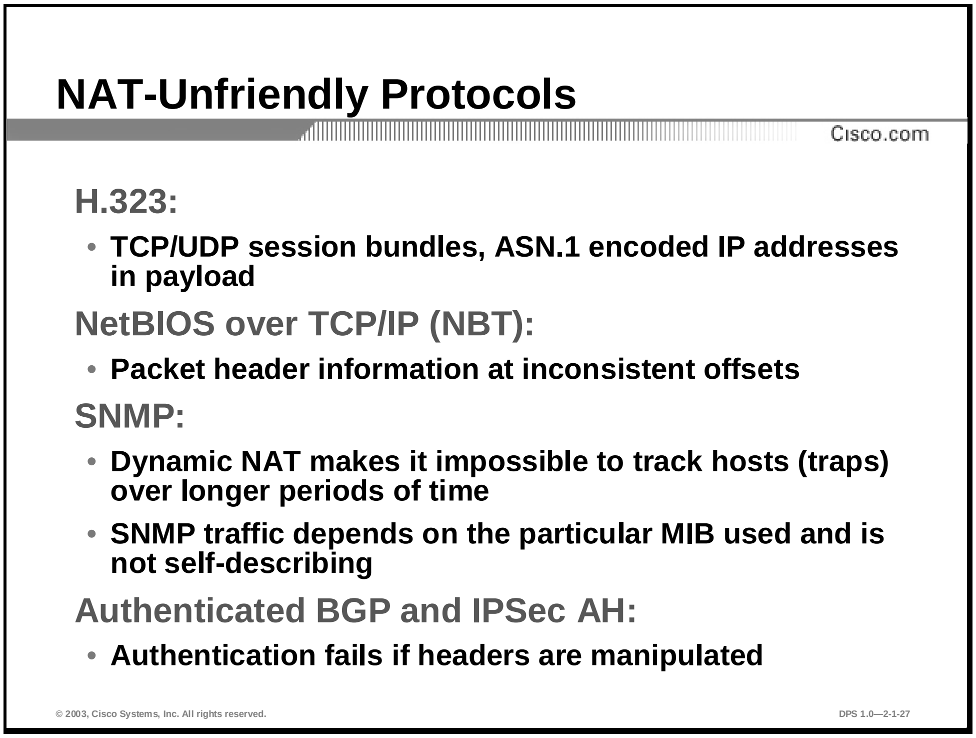
© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—2-1-26

# Facts

PAT requires some type of identifier in order to distribute incoming packets to the corresponding inside hosts. Because the Internet Control Message Protocol (ICMP) is carried directly within IP, NAT cannot utilize port numbers, so the ICMP identifier is used instead. This is only important for query messages such as PING, which uses echo request and echo ICMP messages. Both ICMP message types contain a 16-bit identifier field and a 16-bit sequence number field (according to RFC 792 both are only optional, but they are commonly used).

| | |
|---|---|
| **Note** | Only fragment 0 *creates* the translation entry. If a fragment N with N>0 arrives first at the router, it is dropped. |



## NAT-Unfriendly Protocols

Cisco.com

H.323:

- TCP/UDP session bundles, ASN.1 encoded IP addresses in payload

NetBIOS over TCP/IP (NBT):

- Packet header information at inconsistent offsets

SNMP:

- Dynamic NAT makes it impossible to track hosts (traps) over longer periods of time
- SNMP traffic depends on the particular MIB used and is not self-describing

Authenticated BGP and IPSec AH:

- Authentication fails if headers are manipulated

DPS 1.0—2-1-27

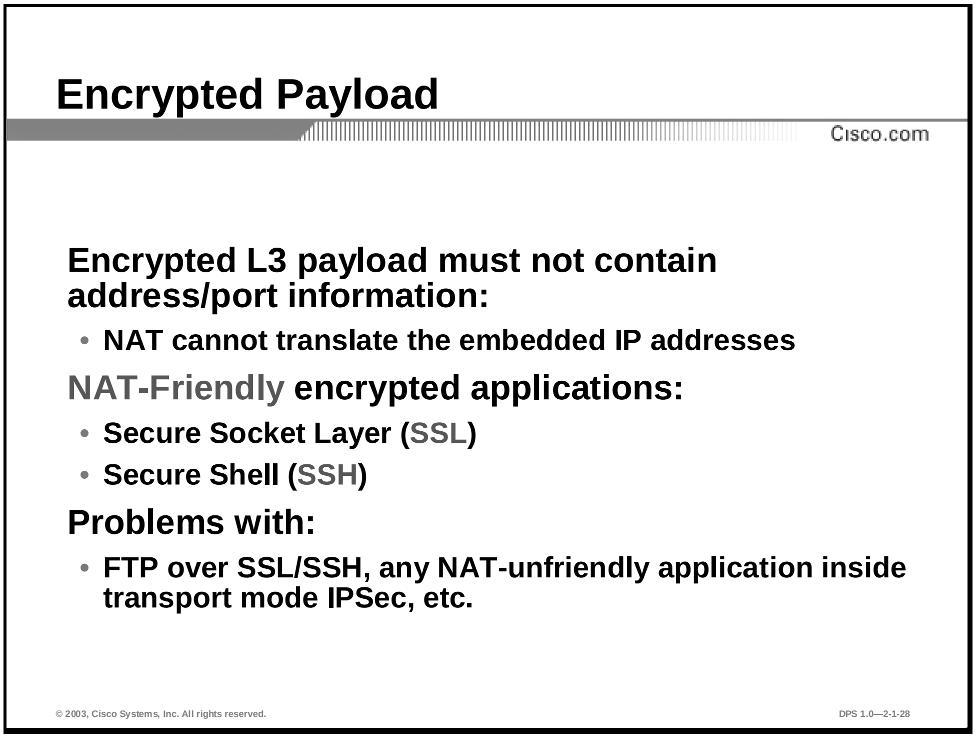
## Demanding Protocols

Several application layer protocols, for example, Simple Network Management Protocol (SNMP) and H.323, hide address information by using ASN.1 as a presentation layer. In addition, depending on the number of Management Information Bases (MIBs), there may be a large number of different SNMP messages. There is no single format for SNMP requests, so responses are processed in a general fashion.

SNMP trap messages are always inbound UDP packets and occur at unpredictable times. Sometimes these intervals are too large for a NAT/PAT device to track. NetBIOS over TCP/IP (NBT) transports packet header information at inconsistent offsets. These protocols are demanding for NAT and require excessive processing resources. Many vendors do not even support these protocols together with NAT.

## Authentication and Encryption

Obviously, protocols that authenticate the IP header fail with NAT, and NAT cannot deal with encrypted payloads—otherwise the authentication and encryption algorithms would be too weak to trust. Examples of such authenticated protocols include BGP with authentication, and IPSec in AH mode.



## Encrypted Payload

Cisco.com

Encrypted L3 payload must not contain address/port information:

- NAT cannot translate the embedded IP addresses

NAT-Friendly encrypted applications:

- Secure Socket Layer (SSL)
- Secure Shell (SSH)

Problems with:

- FTP over SSL/SSH, any NAT-unfriendly application inside transport mode IPSec, etc.

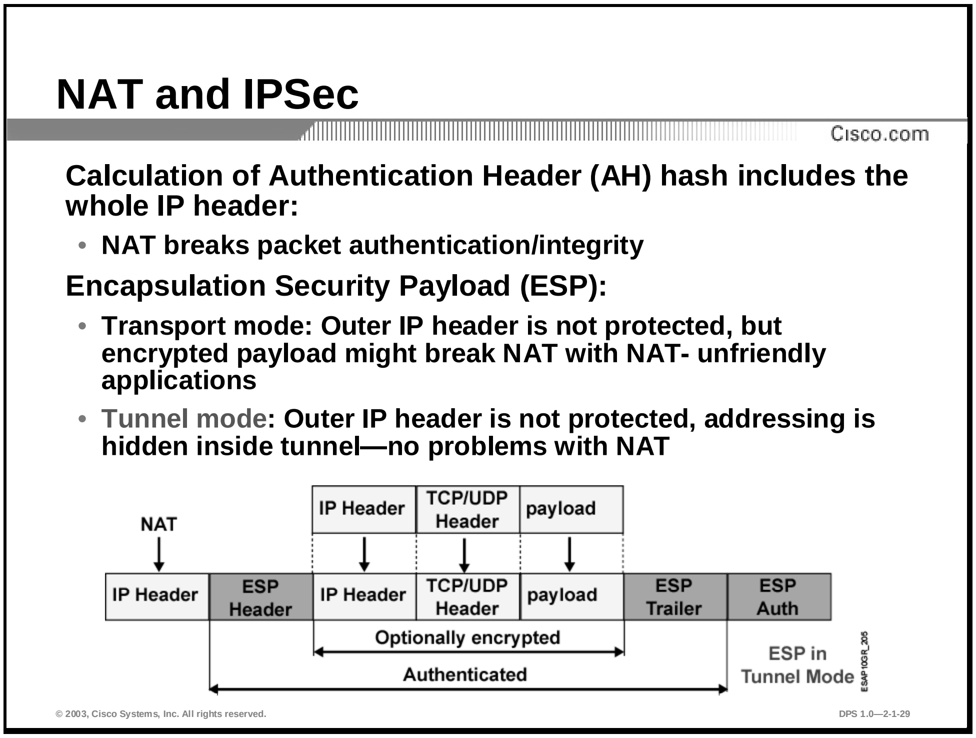
DPS 1.0—2-1-28

## Encryption of the Address Information

NAT *cannot* translate payload address information if the payload is encrypted.

Secure Socket Layer (SSL) and Secure Shell (SSH) are implemented as encrypted TCP payload, but the TCP header is not encrypted. Thus, NAT can handle SSL and SSH without problems.

However, problems may occur with Kerberos, X-Windows, Session Initiation Protocol (SIP), remote shell (RSH), and other NAT-sensitive protocols.



## NAT and IPSec

Cisco.com

**Calculation of Authentication Header (AH) hash includes the whole IP header:**

- **NAT breaks packet authentication/integrity**

**Encapsulation Security Payload (ESP):**

- **Transport mode: Outer IP header is not protected, but encrypted payload might break NAT with NAT- unfriendly applications**
- **Tunnel mode: Outer IP header is not protected, addressing is hidden inside tunnel—no problems with NAT**

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—2-1-29

# NAT and IPSec

IPSec supports two types of headers: the authentication header (AH) and the Encapsulated Security Payload (ESP) header. AH only supports authentication, ESP supports authentication and, optionally, encryption. Both the AH and the ESP support transport mode and tunnel mode.

Using NAT in the path of an AH-protected packet will break IPSec, because the AH encapsulation (transport or tunnel mode) uses the whole IP packet as an input to calculate the authentication hash. This causes the authentication check to fail due to hash mismatches when NAT is used.

Using NAT in the path of an ESP-protected packet can generally work with the following caveats

- The ESP encapsulation always excludes the outer IP header for the authentication hash calculation. Therefore, NAT can change the addresses in the outer header (the original IP header in transport mode, the tunnel header in tunnel mode) without breaking IPSec authentication. If tunnel mode is used, the only addresses of IPSec peers are translated – such a setup should always work.

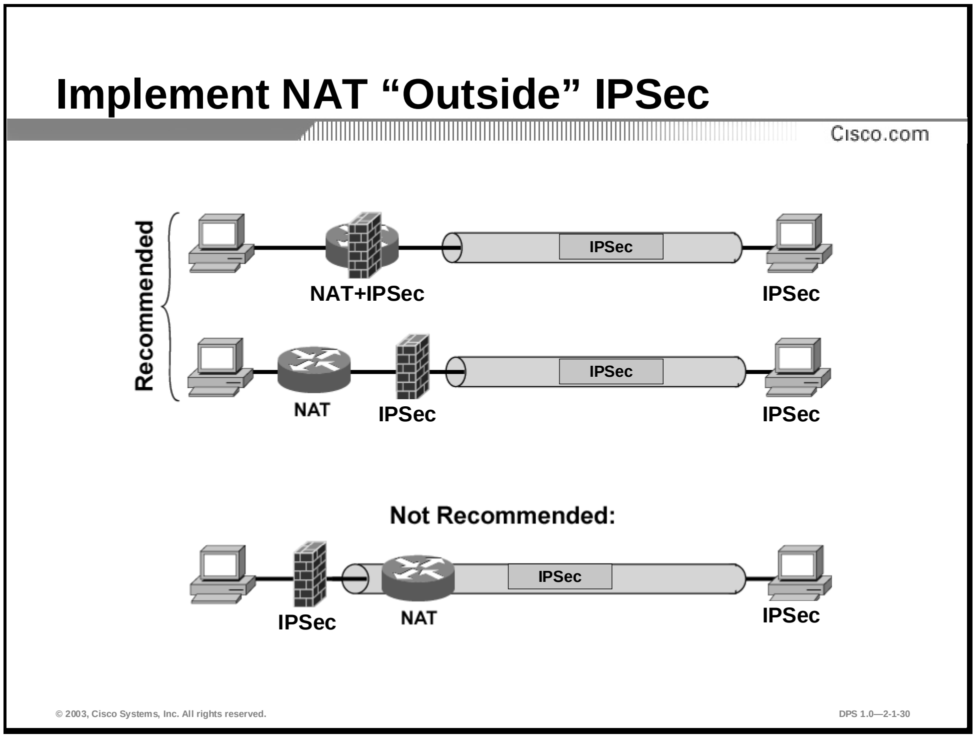
- If ESP transport mode is used, NAT unfriendly applications will embed IP addresses on the application layer, and the NAT device will have no insight into the application stream, as all payloads are encrypted. Therefore, NAT-unfriendly applications will break when protected inside the ESP transport mode encapsulation.

Therefore, the simplest and best solution is to use ESP tunnel mode, if NAT needs to be performed somewhere in the packet path. Because the outer IP header is neither encrypted not authenticated, the use of NAT causes no problems.

## NAT and IKE

The Internet Key Exchange (IKE) protocol is a simple UDP session with a source and destination port of 500. It is NAT friendly, and works over classic one-to-one NAT translation with no problems.

If pre-shared keys are used for authentication, the keys must be based on the global, not the local address of the peer behind the NAT device.



## Implement NAT “Outside” IPSec

Cisco.com

© 2003, Cisco Systems, Inc. All rights reserved.

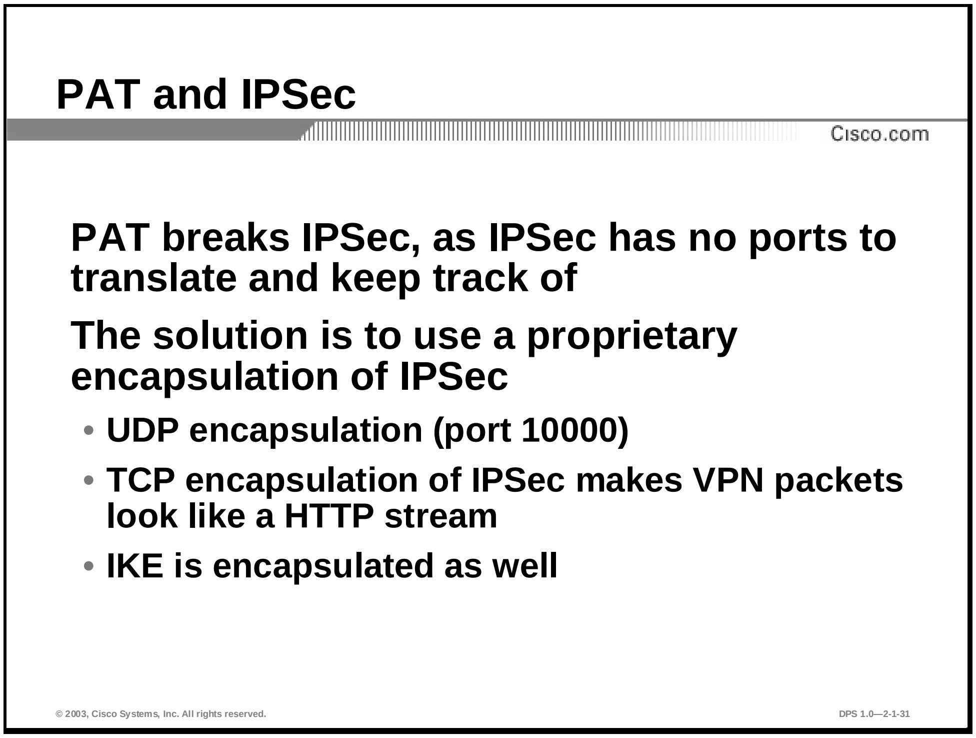
DPS 1.0—2-1-30

## Guidelines

NAT is usually used to translate packets, which are tunneled inside an IPSec connection. The simplest method for NAT to work inside an IPSec VPN is to terminate IPSec before initiating NAT. The general recommendations are:

- Either:
  - Enable NAT and IPSec upon the same gateway, then the operating system (IOS or PIX/OS) will take care for a proper order of processing the packets
  - Perform NAT “outside” the IPSec tunnel on a dedicated device, so that the incoming IPSec tunnel is terminated before packets are address-translated

If translation of the tunnel (IPSec) packets is required,  NAT can be performed in the packet path on ESP tunnel mode packets, taking into account the aforementioned limitations.



## PAT and IPSec

Cisco.com

**PAT breaks IPSec, as IPSec has no ports to translate and keep track of**

**The solution is to use a proprietary encapsulation of IPSec**

- **UDP encapsulation (port 10000)**
- **TCP encapsulation of IPSec makes VPN packets look like a HTTP stream**
- **IKE is encapsulated as well**

DPS 1.0—2-1-31

# PAT and IPSec

PAT performs many-to-one translation for a range of internal IP addresses. PAT is generally supported with most TCP and UDP applications automatically, and ICMP requires some special handling.

Using IPSec over a PAT device will break the tunnel, as the PAT device has no algorithm to associate an incoming IPSec packet to the single global address with an inside VPN peer (there are no “ports” to remember with an IPSec connection). For that reason, Cisco has developed two proprietary solutions, which enable an IPSec tunnel to be established over PAT devices

- Encapsulation of IPSec packets inside UDP, where the entire IPSec packet is additionally encapsulated with an UDP header, with the destination port of 10000. Such a session now looks like a plain UDP session to the PAT device, and can be bi-directionally routed over it.

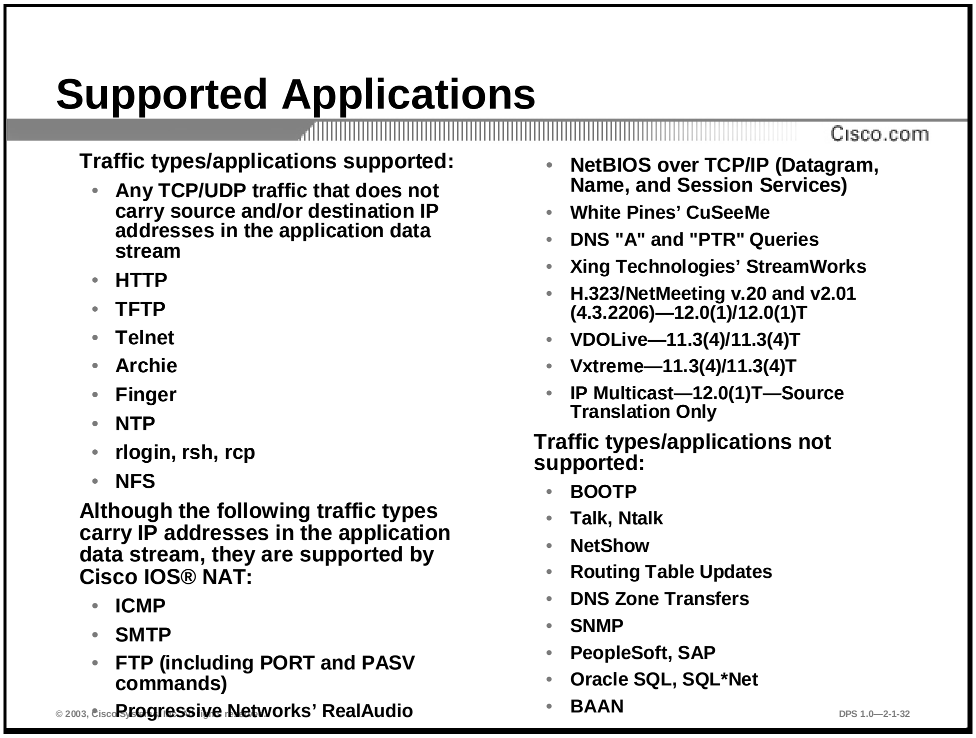
- Encapsulation of IPSec packets inside TCP, where the entire IPSec packet is additionally encapsulated with an TCP header, with the destination port of 80. Such a session now looks like a plain HTTP session to the PAT device, and can be bi-directionally routed over it.

With both encapsulations, the IKE protocol is encapsulated together with IPSec packets inside the same encapsulation session.

Any of the two encapsulations can be used to overcome difficulties with PAT. It must be noted, that the TCP (HTTP) encapsulation currently is not a “correct” TCP session in terms of sequence/acknowledgement numbers, and is dropped by any good stateful firewall, if one exists in the packet path. Therefore, if the encapsulated IPSec session has to cross a PAT device AND a stateful firewall, the UDP encapsulation is required.

The IETF is currently working to standardize the UDP encapsulation of IPSec packets. More information can be found in the published draft

- UDP Encapsulation of IPSec Packets, http://www.ietf.org/internet-drafts/draft-ietf-ipsec-udp-encaps-03.txt.



## Supported Applications

Cisco.com

Traffic types/applications supported:

- Any TCP/UDP traffic that does not carry source and/or destination IP addresses in the application data stream
- HTTP
- TFTP
- Telnet
- Archie
- Finger
- NTP
- rlogin, rsh, rcp
- NFS

Although the following traffic types carry IP addresses in the application data stream, they are supported by Cisco IOS® NAT:

- ICMP
- SMTP
- FTP (including PORT and PASV commands)
- Progressive Networks' RealAudio
- NetBIOS over TCP/IP (Datagram, Name, and Session Services)
- White Pines' CuSeeMe
- DNS "A" and "PTR" Queries
- Xing Technologies' StreamWorks
- H.323/NetMeeting v.20 and v2.01 (4.3.2206)—12.0(1)/12.0(1)T
- VDOLive—11.3(4)/11.3(4)T
- Vxtreme—11.3(4)/11.3(4)T
- IP Multicast—12.0(1)T—Source Translation Only

Traffic types/applications not supported:

- BOOTP
- Talk, Ntalk
- NetShow
- Routing Table Updates
- DNS Zone Transfers
- SNMP
- PeopleSoft, SAP
- Oracle SQL, SQL*Net
- BAAN

DPS 1.0—2-1-32

The following is a list of protocols supported by Cisco's NAT implementation:

- Any TCP/UDP traffic that does not carry source and/or destination IP addresses in the application data stream
- HTTP
- TFTP
- Telnet
- Archie
- Finger
- NTP
- rlogin, rsh, rcp
- NFS

Although the following traffic types carry IP addresses in the application data stream, they are supported by Cisco's NAT implementation:

- ICMP
- SMTP

- FTP (including PORT and PASV commands)Progressive Networks' RealAudioNetBIOS over TCP/IP (Datagram, Name, and Session Services)White Pines' CuSeeMe
- DNS "A" and "PTR" Queries
- Xing Technologies' StreamWorks
- H.323/NetMeeting v.20 and v2.01 (4.3.2206)—12.0(1)/12.0(1)T
- VDOLive—11.3(4)/11.3(4)T
- Vxtreme—11.3(4)/11.3(4)T
- IP Multicast—12.0(1)T—Source Translation Only

Note that this list continues to grow, so network professionals who need to verify whether some specific protocol, not mentioned in the list, is supported or not are strongly encouraged to consult the Cisco web site.

## Practice

Q1) Which secure protocols cause problems with NAT?

A) SSH

B) IPSec AH Transport Mode

C) IPSec AH Tunnel Mode

D) IPSec ESP Transport Mode, no encryption

E) IPSec ESP Transport Mode, encryption

F) IPSec ESP Tunnel Mode, no encryption

G) IPSec ESP Tunnel Mode, encryption

H) HTTPS

Q2) Which of the following NAT manipulations can be applied to DNS messages?

A) manipulation of “PTR” (address-to-host mapping) resource records

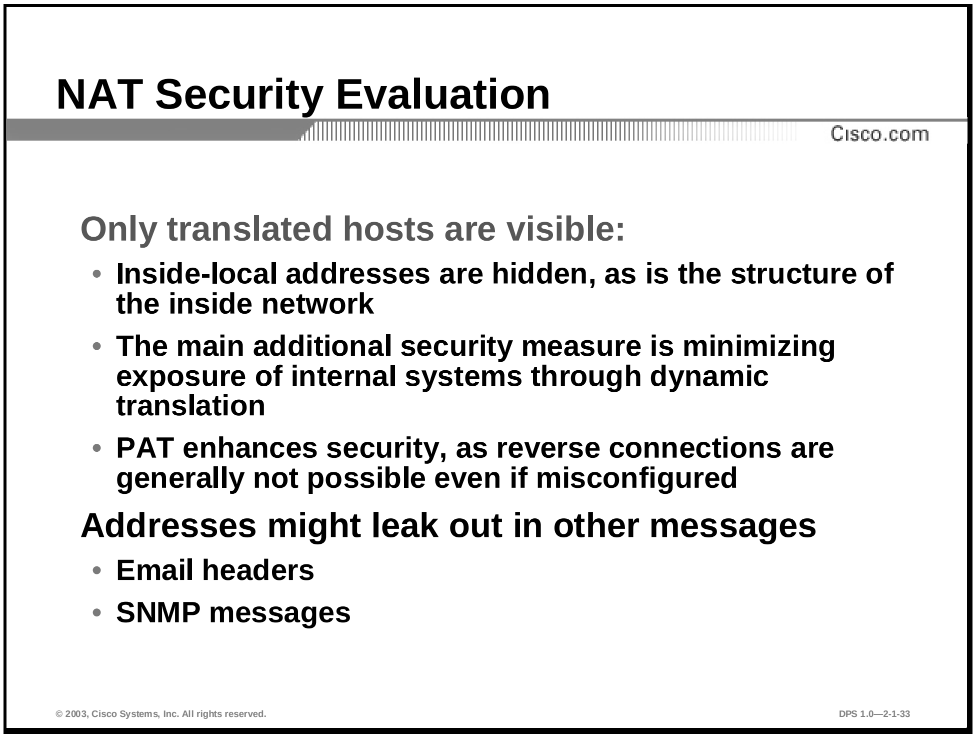
B) “A” (host-to-address mapping) resource records in zone transfers

C) “A” (host-to-address mapping) resource records in DNS responses and overlapping networks

D) “HINFO” resource records in responses of an inside DNS server for external request

E) DNS requests to external DNS servers and overlapping networks

# NAT Security Evaluation



## NAT Security Evaluation

Only translated hosts are visible:

- Inside-local addresses are hidden, as is the structure of the inside network
- The main additional security measure is minimizing exposure of internal systems through dynamic translation
- PAT enhances security, as reverse connections are generally not possible even if misconfigured

Addresses might leak out in other messages

- Email headers
- SNMP messages

DPS 1.0—2-1-33

## Objective

This section will enable the learner to explain how NAT influences the security of a perimeter security system.

## Introduction

NAT/PAT hides inside addresses and the subnet structure. NAT is therefore often regarded as a “first-level” security measure: only translated hosts are visible to the outside. However, NAT alone is a too weak security measure. This section investigates how NAT qualifies as building stone in typical security concepts.

## How NAT/PAT Enhance Security

Most people regard NAT a security measure simply because it hides the internal addresses of hosts behind a NAT device. As assignment of global addresses is usually not related to the actual network structure behind the network device (i.e. all inside hosts use the same global pool), NAT also hides the structure of the internal network. Both measures can be put in the “security through obscurity” class of security measures, as they simply try to withhold information from the attacker.

A more important security feature of NAT is the minimization of exposure, if dynamic NAT is used. With dynamic NAT, an inside system leases a global address of the pool on-demand, and returns it after an idle period. Therefore (depending on the NAT device implementation), the internal system is only visible on the external network when it needs to talk to it, and remains hidden behind the NAT device the rest of time.

PAT, on the other hand, provides additional security by essentially being a one-way connection engine. Outbound connections through a PAT device are all multiplexed over a single global IP address. If a new inbound connection is made to that exposed IP address, the PAT device does not know, to which inside host to forward the connection. Even if access rules permit such connections by mistake, PAT by design cannot support such connections without specific “static PAT port forwarding” rules and therefore provides an additional layer of security (defense in depth).

Even though NAT and PAT are used, some addressing information might still leak out of the inside network. Addresses embedded in email messages (the list of servers a message has passed through), or inside SNMP, are often not translated and might reveal internal addressing information and network structure to an attacker.

## Practice

Q1) What is the main security benefit when using NAT on the network perimeter? (Choose one.)

A) hiding of internal host addresses

B) hiding of internal network addresses

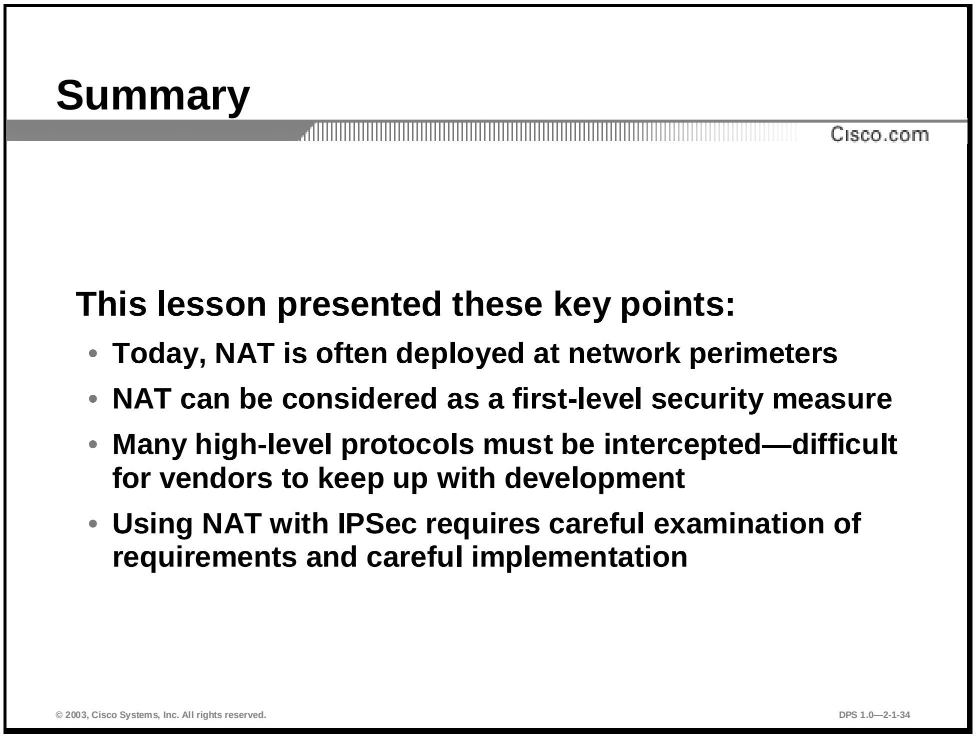
C) minimizing exposure of internal hosts through dynamic NAT

D) simplification of routing

E) minimizing exposure of internal hosts through static NAT

# Summary

This section summarizes the key points discussed in this lesson.



**Summary**

**This lesson presented these key points:**

- **Today, NAT is often deployed at network perimeters**
- **NAT can be considered as a first-level security measure**
- **Many high-level protocols must be intercepted—difficult for vendors to keep up with development**
- **Using NAT with IPSec requires careful examination of requirements and careful implementation**

DPS 1.0—2-1-34

## Next Steps

After completing this lesson, go to:

- Design Using a NAT/PAT Solution lesson

# Quiz: NAT Overview

Complete the quiz to assess what you have learned in this lesson.

## Objectives

This quiz tests your knowledge on how to:

- Identify and compare NAT technologies
- Select an appropriate NAT technology for an organization's requirements

## Instructions

Answer these questions:

1. Which protocols are generally problematic with NAT?
2. Which protocols are generally problematic with PAT?
3. How does NAT impact packet IPSec authentication and integrity?
4. How does IPSec encryption impact NAT?

## Scoring

You have successfully completed the quiz for this lesson when you earn a score of 80 percent or better.

# Design Using a NAT/PAT Solution

## Overview

### Introduction

This lesson describes how to configure bidirectional Network Address Translation (NAT) situations that often occur in private enterprise networks. The lesson initially analyzes scenarios, such as overlapping network address spaces and multihoming, where bidirectional NAT is required in order to maintain connectivity. After providing Cisco IOS and PIX/OS configuration guidelines and examples, the lesson closes with some in-depth considerations about multihoming.

### Importance

Designing bidirectional NAT is critical for many enterprise-ISP connections. Address translation “both ways” requires an understanding of perimeter policies, security demands, DNS interception, and routing principles. This lesson provides the necessary facts and guidelines to configure bidirectional NAT tailored to an organization’s requirement.

### Lesson Objective

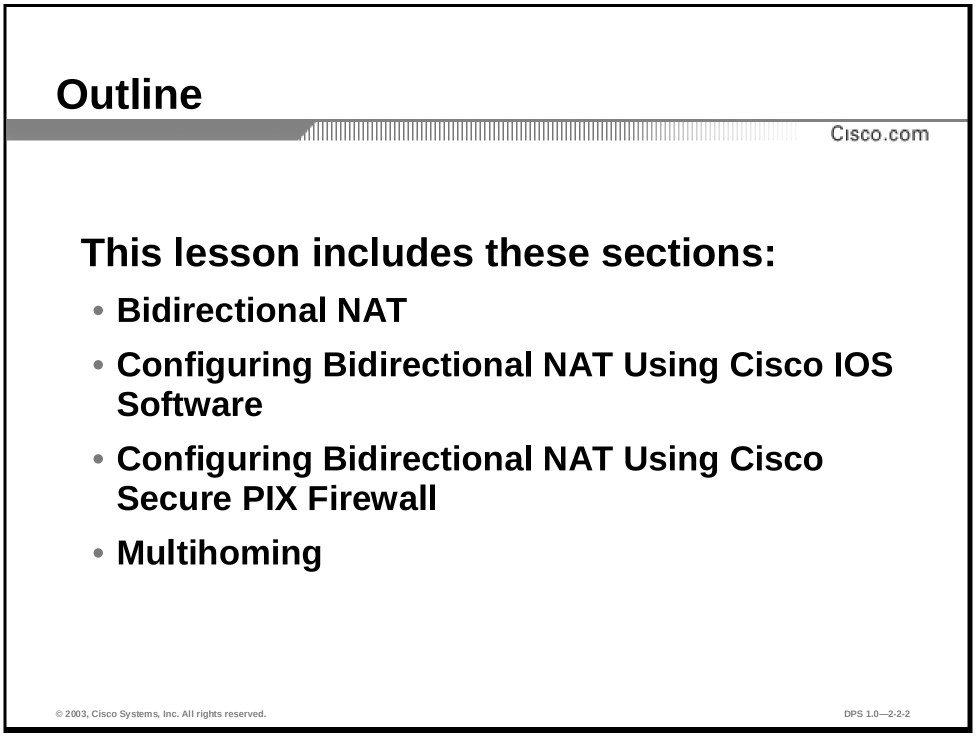
The lesson will enable the learner to design advanced NAT solutions for some common enterprise connectivity scenarios.

## Learner Skills and Knowledge

To fully benefit from this lesson, you must have these prerequisite skills and knowledge:

- A solid understanding of basic NAT
- A solid understanding of the Domain Name System (DNS)
- Fundamental knowledge about Internet (BGP) routing

## Outline



Outline

Cisco.com

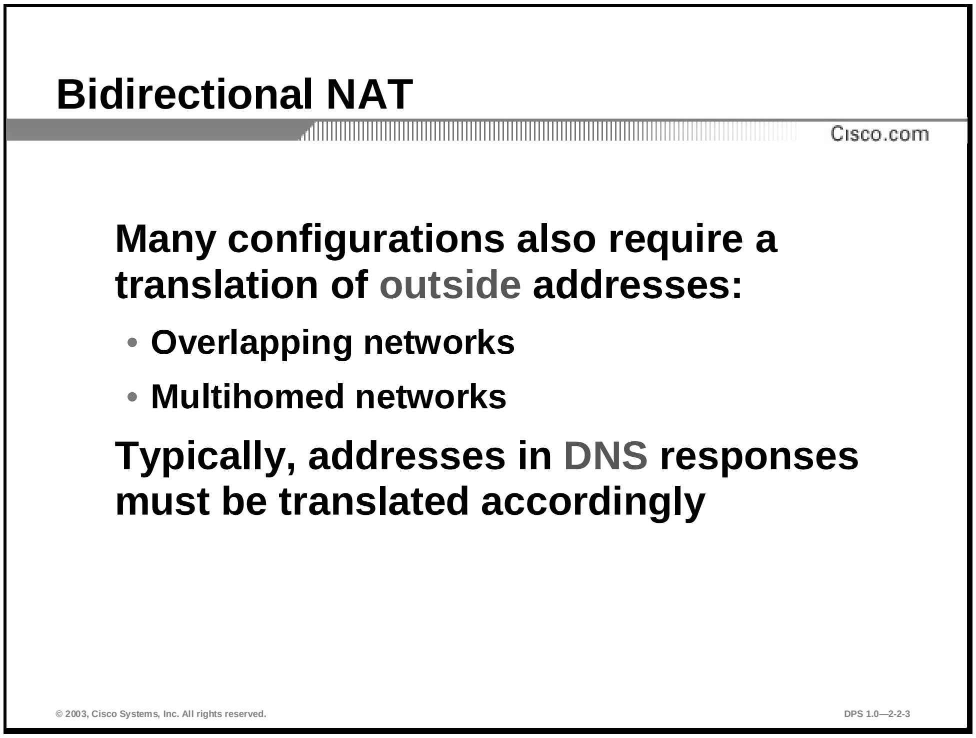
This lesson includes these sections:

- Bidirectional NAT
- Configuring Bidirectional NAT Using Cisco IOS Software
- Configuring Bidirectional NAT Using Cisco Secure PIX Firewall
- Multihoming

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—2-2-2

# Bidirectional NAT



## Bidirectional NAT

Cisco.com

**Many configurations also require a translation of outside addresses:**

- **Overlapping networks**
- **Multihomed networks**

**Typically, addresses in DNS responses must be translated accordingly**

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—2-2-3

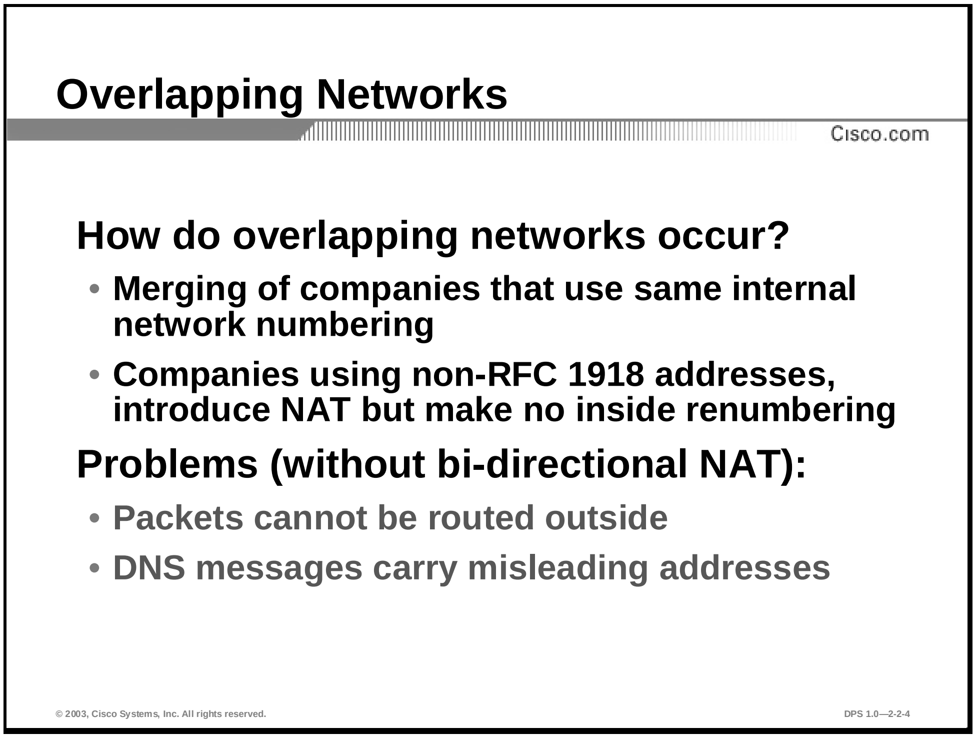
## Objective

This section will enable the learner to explain the concept of bidirectional NAT and choose it in appropriate design situations.

## Introduction

Although typical NAT applications only include an inside-to-outside translation of addresses, several very important scenarios also require an outside-to-inside translation. Other typical situations, excluding security reasons, where bi-directional NAT is absolutely necessary, are overlapping networks and multihomed networks. These situations are described in more detail later in this lesson.

Nearly all bidirectional NAT scenarios require an interception of DNS responses because either a response of an outside DNS server contains an inside-invalid IP address, or the DNS server is inside but questioned from outside.



## Overlapping Networks

Cisco.com

How do overlapping networks occur?

- Merging of companies that use same internal network numbering
- Companies using non-RFC 1918 addresses, introduce NAT but make no inside renumbering

Problems (without bi-directional NAT):

- Packets cannot be routed outside
- DNS messages carry misleading addresses

DPS 1.0—2-2-4

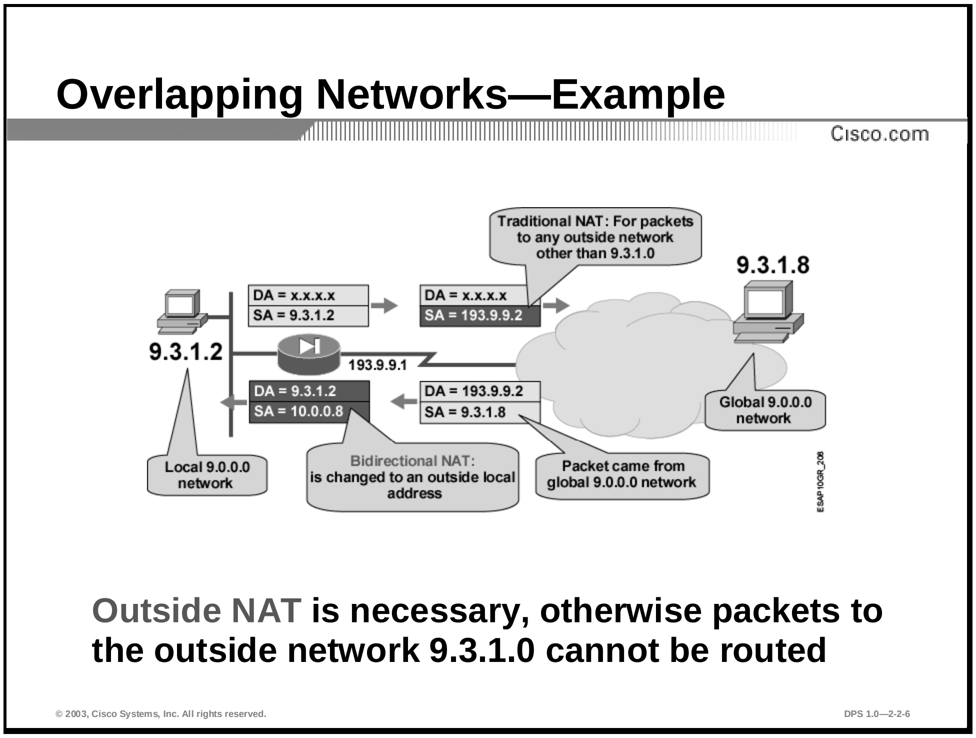
## Reasons for Overlapping

Consider two companies merging with each other who both use the same RFC 1918 addresses for their internal network numbering. If they are connected via tunnels only a bidirectional NAT setup at the perimeter will assure that their routing will still work. If they utilize an inside DNS, instead of tunnels, a bidirectional NAT configuration might be necessary to avoid confusion by DNS responses containing local IP addresses.

Overlapping network address space also occurs when large companies introduce NAT and return their sizeable registered IP address block to the Internet community, without changing to a RFC 1918 address strategy. By not changing to a RFC 1918 address strategy, the network of the company returning the IP address block may not be able to communicate with the network that is now assigned the returned addresses.

## Routing Ambiguity

Because some packets may have identical source and destination addresses, traffic between overlapping networks cannot be routed correctly without a clean bidirectional NAT configuration. DNS responses may also relate to addresses existing on the other side of a NAT router and again be told that the address is not reachable. This ambiguity is a connectivity problem that can only be solved with bidirectional NAT, including DNS interception.



## Overlapping Networks—Example

Cisco.com

**Outside NAT is necessary, otherwise packets to the outside network 9.3.1.0 cannot be routed**

DPS 1.0—2-2-6

## Example

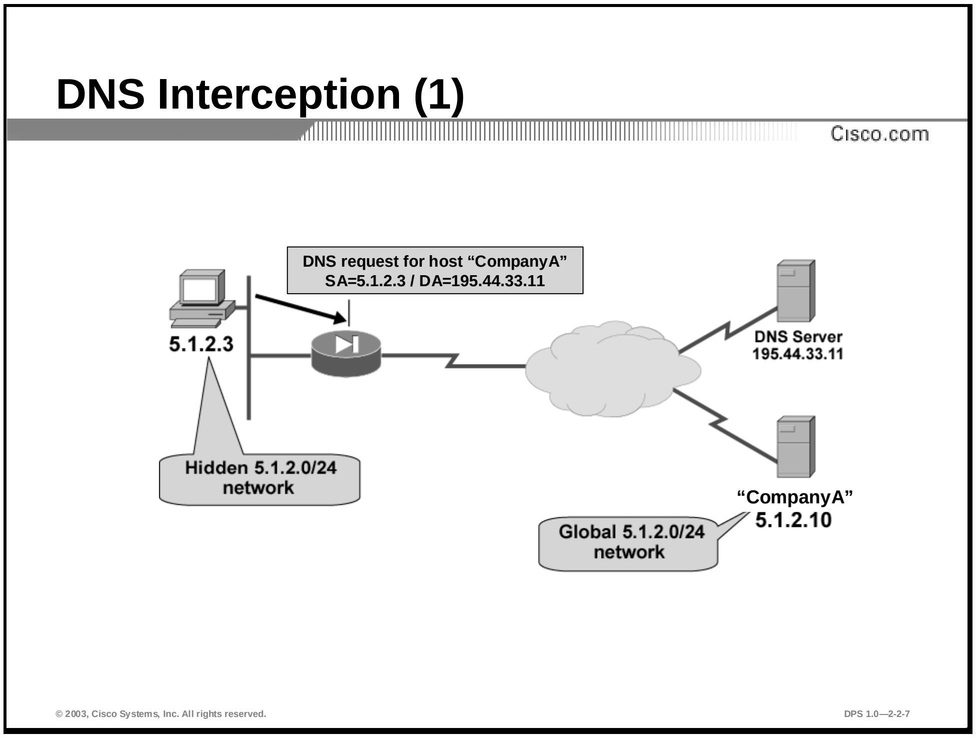
In the simple example illustrated, the left-hand network has used a class A network 9.0.0.0 for several years and now wants to return the address space to the Internet.

The left-hand network now presents itself to the outside world through NAT and the class A range they have returned will be used by other customers. Incoming packets therefore may have the same source addresses still used by the network's inside devices. Thus, the owner of the left-hand network should renumber their hosts with RFC1918 private addresses.

However, the left-hand network may have a large number of hosts and may not be prepared to renumber all devices at the same time. In this case, the NAT device (PIX Firewall in the figure) translates the incoming packet source addresses from the registered class A network 9.0.0.0 source addresses to RFC 1918 addresses (the 10.0.0.8 address in the example). Thus, by mapping to an outside-local source address, the return packets can be routed outside, as the outside 9.0.0.0 network appears to be reachable through the firewall as a RFC 1918 address.

## Definition

Bidirectional NAT is performed if **outside NAT** is added to the existing inside NAT configuration.



## DNS Interception (1)

Cisco.com

DNS request for host “CompanyA”
SA=5.1.2.3 / DA=195.44.33.11

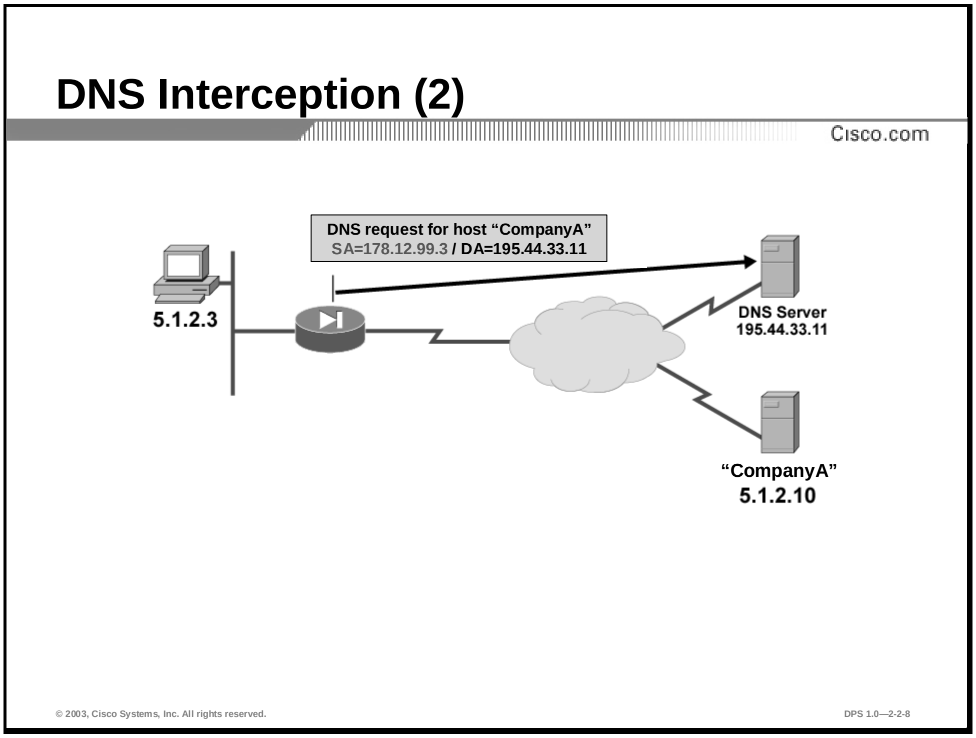
DPS 1.0—2-2-7

## Example

DNS interception is a more difficult issue.

Because IP addresses of outside hosts are usually unknown to inside hosts, the network queries an outside DNS server for name resolution. In the example, the inside host 5.1.2.3 (left) wants to connect to the outside host named “CompanyA”, which also has the same IP address:

**Step 1** The network sends a DNS request to the outside DNS server, 195.44.33.11.



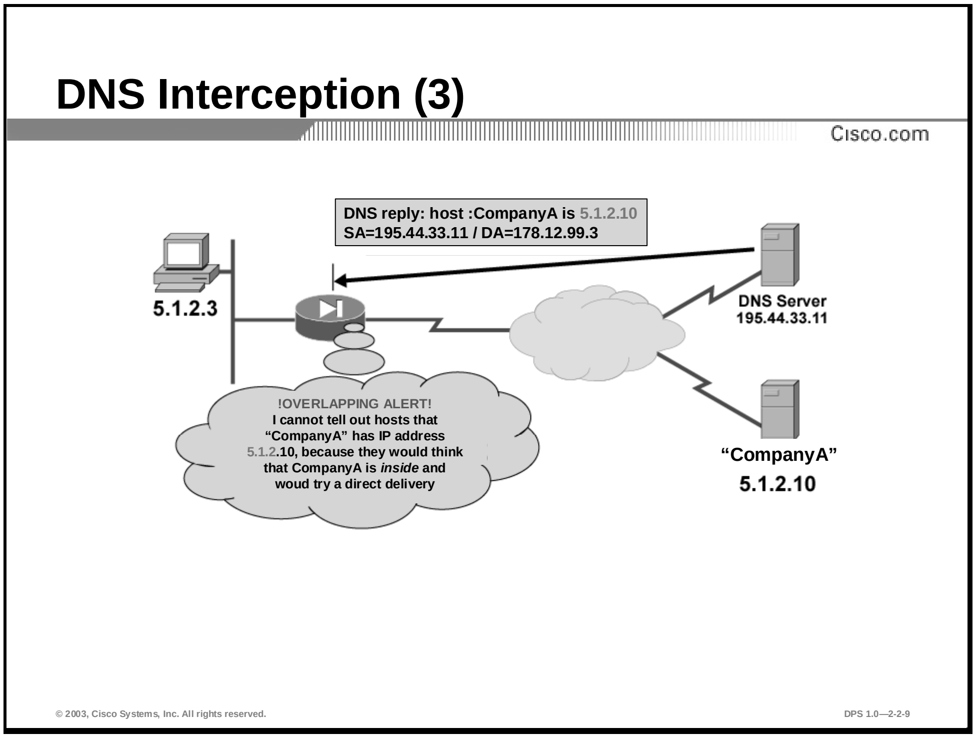
## DNS Interception (2)

Cisco.com

DNS request for host “CompanyA”
SA=178.12.99.3 / DA=195.44.33.11
5.1.2.3
DNS Server
195.44.33.11
“CompanyA”
5.1.2.10

DPS 1.0—2-2-8

**Step 2** The source address of the DNS request is translated to an inside global address as usual.



## DNS Interception (3)

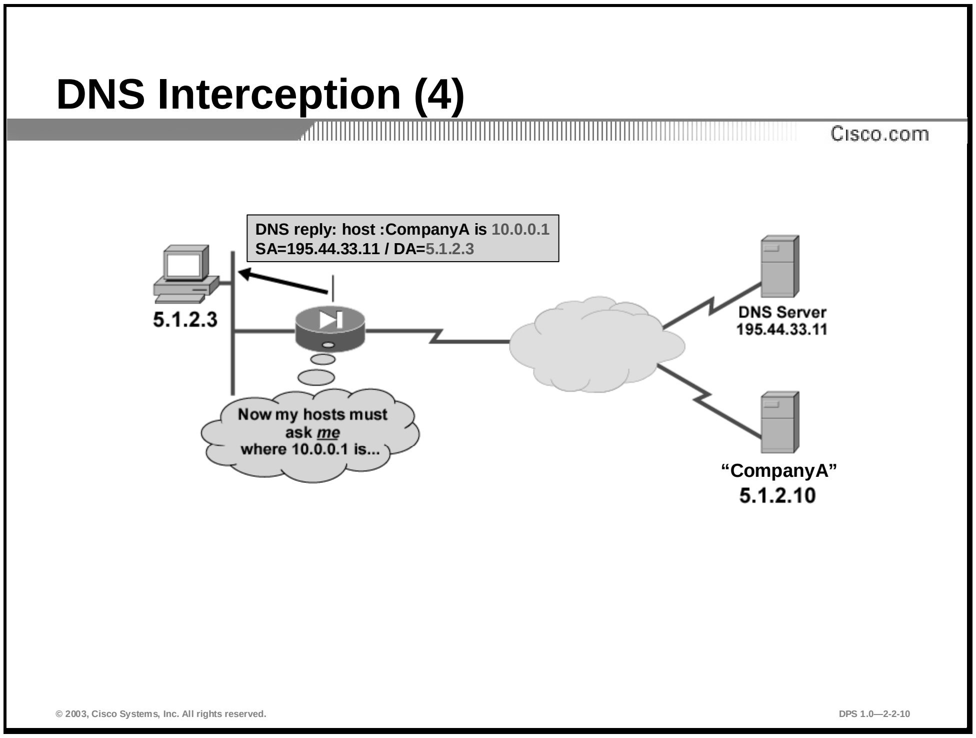
DNS reply: host :CompanyA is 5.1.2.10
SA=195.44.33.11 / DA=178.12.99.3

!OVERLAPPING ALERT!
I cannot tell out hosts that “CompanyA” has IP address 5.1.2.10, because they would think that CompanyA is *inside* and woud try a direct delivery

DPS 1.0—2-2-9

**Step 3** The DNS server performs an address resolution and sends a response message to the address 178.12.99.3, advising that “CompanyA” has the address 5.1.2.10.

However, this IP address is supposed to be inside the left-hand network. The host will assume that CompanyA is local and try a direct delivery, which will fail. In this situation the NAT router must manipulate the Layer 7 DNS information and translate the global-outside addresses to any other address not used inside the left-hand network. The safest plan is to choose a RFC 1918 address at this stage.



## DNS Interception (4)

DNS reply: host :CompanyA is 10.0.0.1
SA=195.44.33.11 / DA=5.1.2.3

Now my hosts must ask *me* where 10.0.0.1 is...

DPS 1.0—2-2-10

**Step 4** The router examines every DNS reply to ensure that the resolved address is not also used on the inside. If the router finds an overlapping address, it will translate the address to an RFC 1918 address.

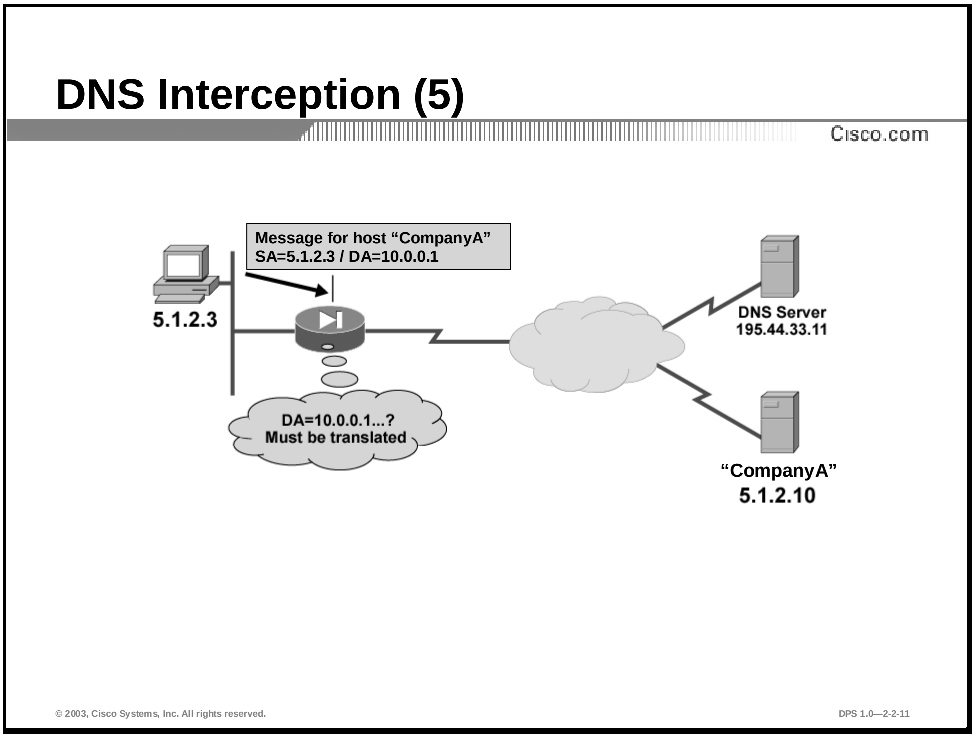
---

**Note** Always choose a RFC 1918 address for the outside local pool of addresses. Otherwise, you would block connectivity to another part of the Internet, which you would choose as the outside local pool.

---

**Note** Cisco IOS and PIX Firewall NAT are able to inspect and perform address translation on *A* (Address) and *PTR* (Pointer) DNS Resource Records.

---



## DNS Interception (5)

Cisco.com

Message for host “CompanyA”
SA=5.1.2.3 / DA=10.0.0.1

5.1.2.3

DA=10.0.0.1...?
Must be translated

DNS Server
195.44.33.11

“CompanyA”
5.1.2.10

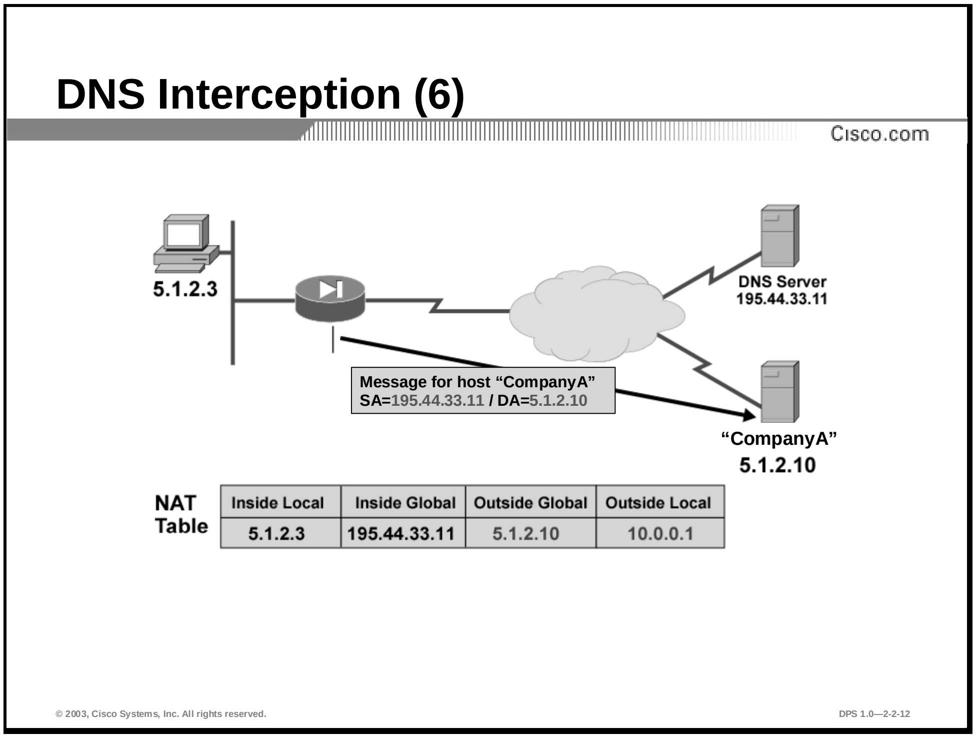
© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—2-2-11

**Step 5** If the destination address of outgoing packets matches a previously introduced outside-local address, it is translated into an associated outside-global address.

In the converse situation where the DNS server is inside and a DNS request is sent by an outside host, the same activity is performed. If the name resolution results in an inside local address the NAT router has to translate this address.

**Note** Cisco IOS and the PIX Firewall do *not* translate addresses inside DNS zone transfers.



## DNS Interception (6)

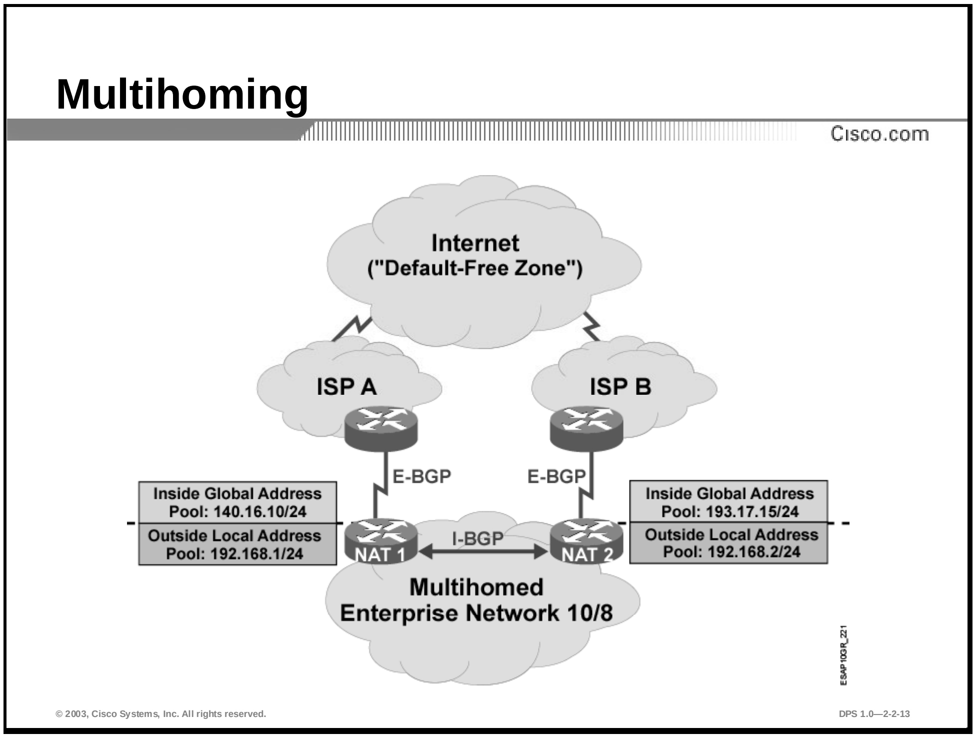
Message for host "CompanyA"
SA=195.44.33.11 / DA=5.1.2.10

NAT Table

| Inside Local | Inside Global | Outside Global | Outside Local |
|---|---|---|---|
| 5.1.2.3 | 195.44.33.11 | 5.1.2.10 | 10.0.0.1 |

DPS 1.0—2-2-12

**Step 6** The outside local address 10.0.0.1 is translated to the original outside global address 5.1.2.10 again, and the packet reaches CompanyA.



## Multihoming

DPS 1.0—2-2-13

## Multihoming NAT Issues

Using more than one ISP provides more reliable Internet connectivity and better routing flexibility.

In the example, the multihomed enterprise network is assigned ISP A prefixes (140.16.10/24) from ISP A’s address block and ISP B prefixes (193.17.15/24) from ISP B’s address block. In order to avoid later network renumbering, the enterprise network uses the inside local prefix 10/8, while the assigned ISP prefixes are used as inside global address pools.

The network maps the outside global addresses to outside local address pools (192.168.1/24 and 192.168.2/24) and only advertises these prefixes as outside destinations in the local network by routing protocols. Thus, the number of prefixes advertised is minimized. Note that the outside local address must not overlap with the inside local addresses nor with any outside global address.

## Practice

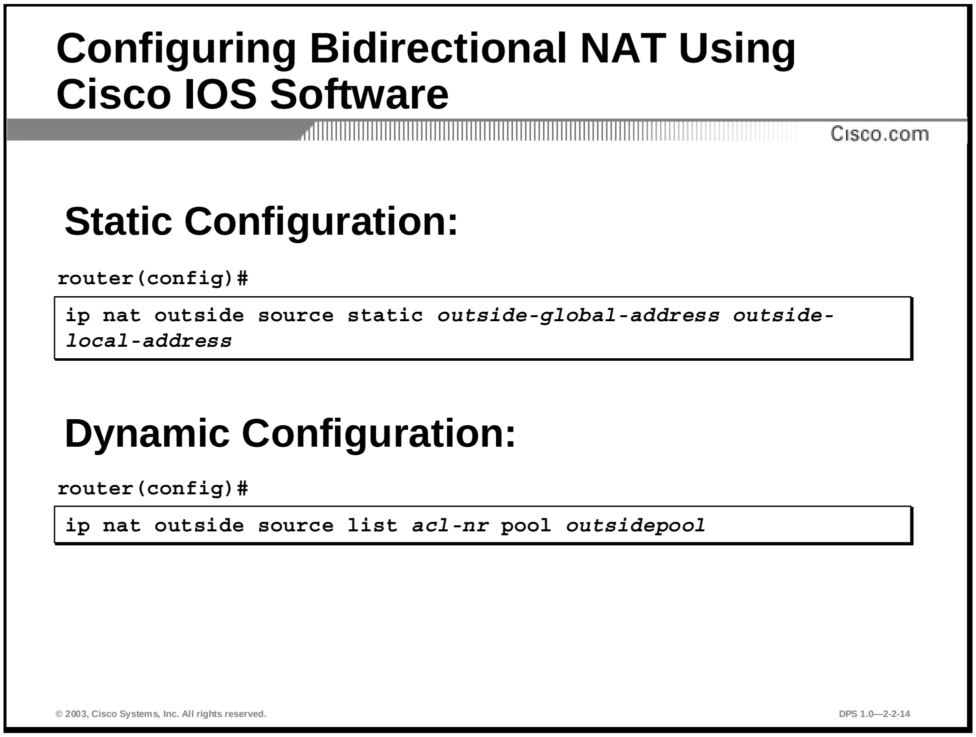
Q1) What might happen if there is no interception of external DNS responses with overlapping networks?

A) packets might be routed to other outside destinations

B) the DNS response would not reach the local host that sends the associate DNS request

C) the inside receiver of the DNS response would try a direct delivery of the packet

# Configuring Bidirectional NAT Using Cisco IOS Software



## Configuring Bidirectional NAT Using Cisco IOS Software

### Static Configuration:

```
router(config)#
ip nat outside source static outside-global-address outside-
local-address
```

### Dynamic Configuration:

```
router(config)#
ip nat outside source list acl-nr pool outsidepool
```

DPS 1.0—2-2-14

## Objective

This section will enable the learner to configure bidirectional NAT using Cisco IOS software.

## Introduction

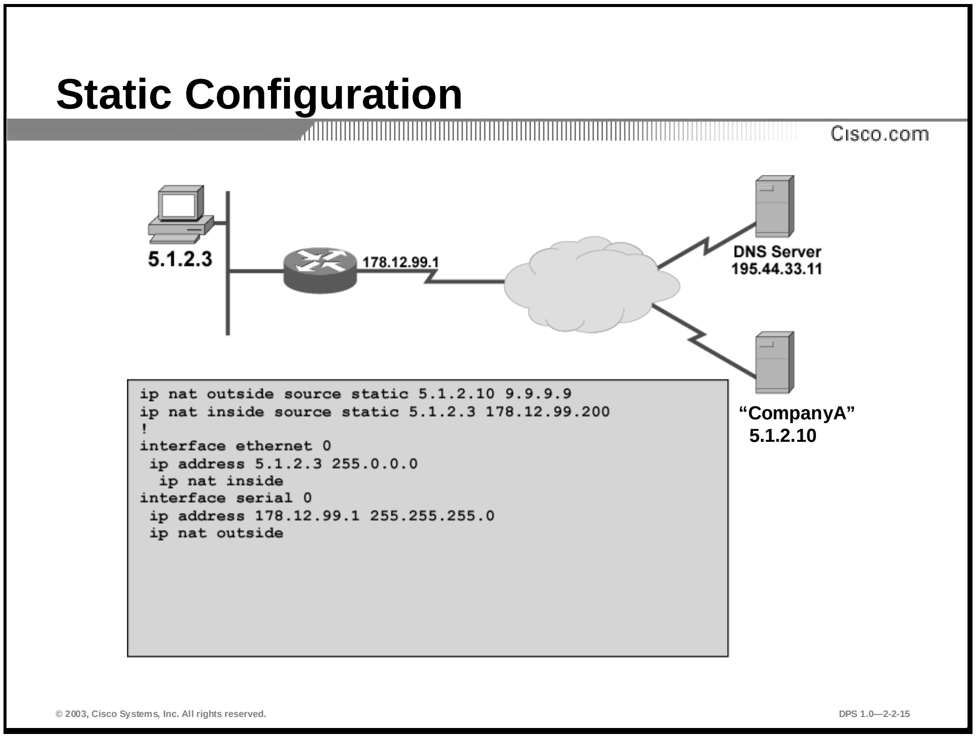
The configuration of NAT is different between Cisco IOS devices and PIX/OS firewalls. This section focuses on IOS NAT commands only.

## IOS Outside NAT

Cisco IOS routers can easily be configured for outside NAT by the well-known **ip nat** command using the additional keywords **outside source,** as presented in the figure. Use the keyword **static** before specifying the addresses to statically map an outside global address and an outside local address. If a dynamic translation is needed, use the keyword **list** to specify an access list that defines the outside global addresses to which the translation should be applied. Finally, a previously defined outside local address pool identifier must be specified.

The DNS interception is enabled automatically when enabling outside NAT.

The following examples include sample configurations.



## Static Configuration

```
ip nat outside source static 5.1.2.10 9.9.9.9
ip nat inside source static 5.1.2.3 178.12.99.200
!
interface ethernet 0
 ip address 5.1.2.3 255.0.0.0
  ip nat inside
interface serial 0
 ip address 178.12.99.1 255.255.255.0
 ip nat outside
```

DPS 1.0—2-2-15

## Example: Overlapping Networks

"CompanyA" uses the registered address 5.1.2.10, which is the same as used in the left-hand local network, therefore:

- The outside NAT must be configured
- A DNS interception will be required

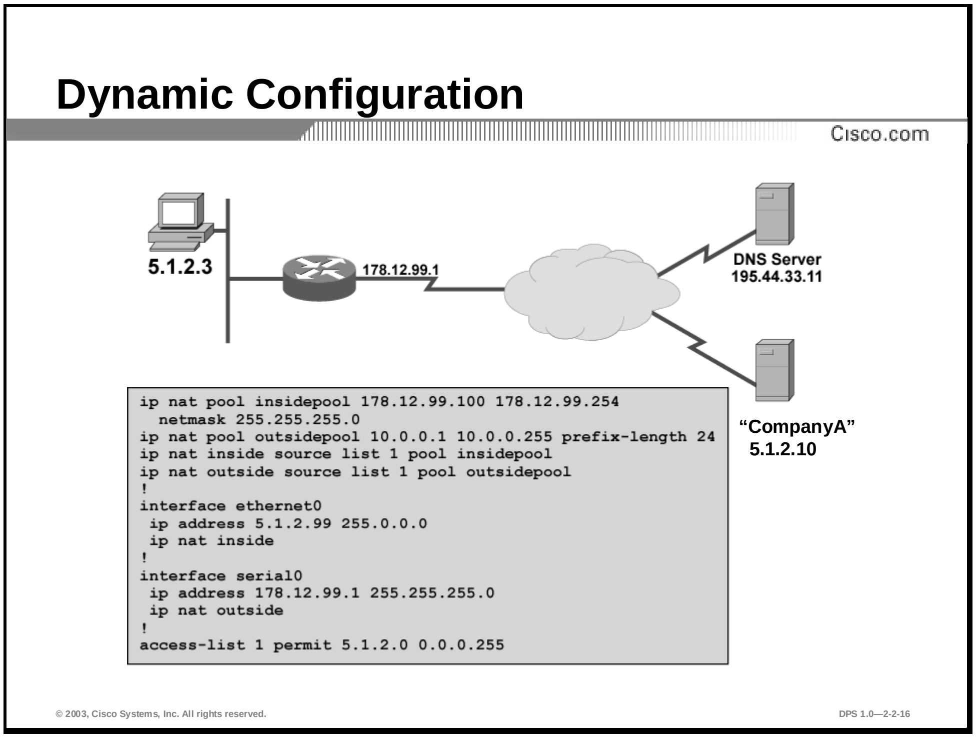
## Guidelines for Static NAT

The example configuration illustrated consists of the following basic steps:

**Step 1** Use the **ip nat outside source static** command to define the static outside NAT to be applied to inbound packets, originated by the outside host with the overlapping address. Specify the single outside global address to be translated with the single outside local address.

**Step 2** Use the **ip nat inside source static** command to define the static inside NAT to be applied to outbound packets, originated by the inside host with the overlapping address. Specify the inside local address and the inside global address to be used for the translation.

**Step 3** In the interface configuration mode, use the **ip nat** command to specify the inside and outside areas.



## Dynamic Configuration

```
ip nat pool insidepool 178.12.99.100 178.12.99.254
  netmask 255.255.255.0
ip nat pool outsidepool 10.0.0.1 10.0.0.255 prefix-length 24
ip nat inside source list 1 pool insidepool
ip nat outside source list 1 pool outsidepool
!
interface ethernet0
 ip address 5.1.2.99 255.0.0.0
 ip nat inside
!
interface serial0
 ip address 178.12.99.1 255.255.255.0
 ip nat outside
!
access-list 1 permit 5.1.2.0 0.0.0.255
```

DPS 1.0—2-2-16

# Guidelines for Dynamic NAT

The example configuration illustrated consists of the following basic steps:

**Step 1** Use the command **ip nat pool** to define an inside global address pool.

**Step 2** Use the command **ip nat pool** to define an outside local address pool.

**Step 3** Use the command **ip nat inside source** to enable inside NAT. Specify an access list, defining the inside local addresses to be translated using the predefined inside-pool

**Step 4** Use the command **ip nat outside source** to enable outside NAT. Specify an access list, defining the outside global addresses to be translated using the predefined outside-pool.

**Step 5** In the interface configuration mode, use the **ip nat** command to specify the inside and outside areas.

**Step 6** Create the access list specifying the overlapping address.

## The ip nat outside source Command Reference

To enable NAT of the outside source address, use the **ip nat outside source** global configuration command. To remove the static entry or the dynamic association, use the **no** form of this command.

**ip nat outside source** {**list** {*access-list-number* | *name*} **pool** *name* | **static** *global-ip local-ip*}

**no ip nat outside source** {**list** {*access-list-number* | *name*} **pool** *name* | **static** *global-ip local-ip*}

### Table 1: ip nat outside source Parameters

| Parameter | Description |
|---|---|
| **list** *access-list-number* | Standard IP access list number. Packets with source addresses that pass the access list are translated using global addresses from the named pool. |
| **list** *name* | Name of a standard IP access list. Packets with source addresses that pass the access list are translated using global addresses from the named pool. |
| **pool** *name* | Name of the pool from which global IP addresses are allocated. |
| **static** *global-ip* | Sets up a single static translation. This argument establishes the globally unique IP address assigned to a host on the outside network by its owner. It was allocated from globally routable network space. |
| *local-ip* | Sets up a single static translation. This argument establishes the local IP address of an outside host as it appears to the inside world. The address was allocated from address space routable on the inside (RFC 1918, *Address Allocation for Private Internets*). |

## Practice

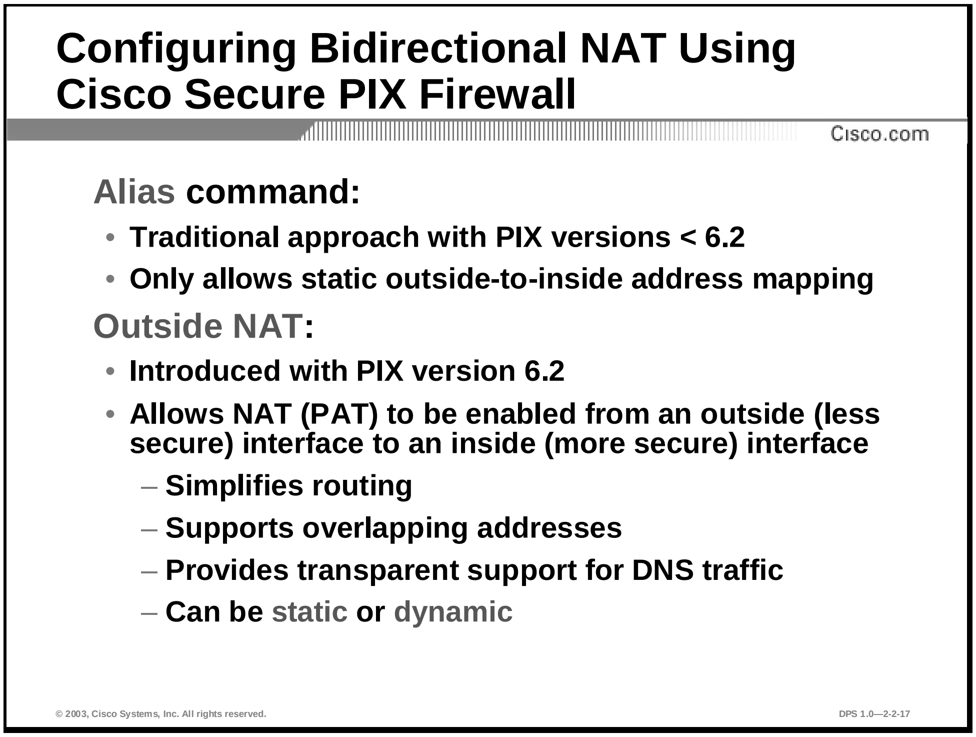
Q1) Which command enables outside NAT?

A) **ip nat pool**

B) **ip nat outside**

C) **ip nat outside source**

# Configuring Bidirectional NAT Using Cisco Secure PIX Firewall



## Configuring Bidirectional NAT Using Cisco Secure PIX Firewall

Cisco.com

**Alias command:**

- **Traditional approach with PIX versions < 6.2**
- **Only allows static outside-to-inside address mapping**

**Outside NAT:**

- **Introduced with PIX version 6.2**
- **Allows NAT (PAT) to be enabled from an outside (less secure) interface to an inside (more secure) interface**
  - **Simplifies routing**
  - **Supports overlapping addresses**
  - **Provides transparent support for DNS traffic**
  - **Can be static or dynamic**

DPS 1.0—2-2-17

## Objective

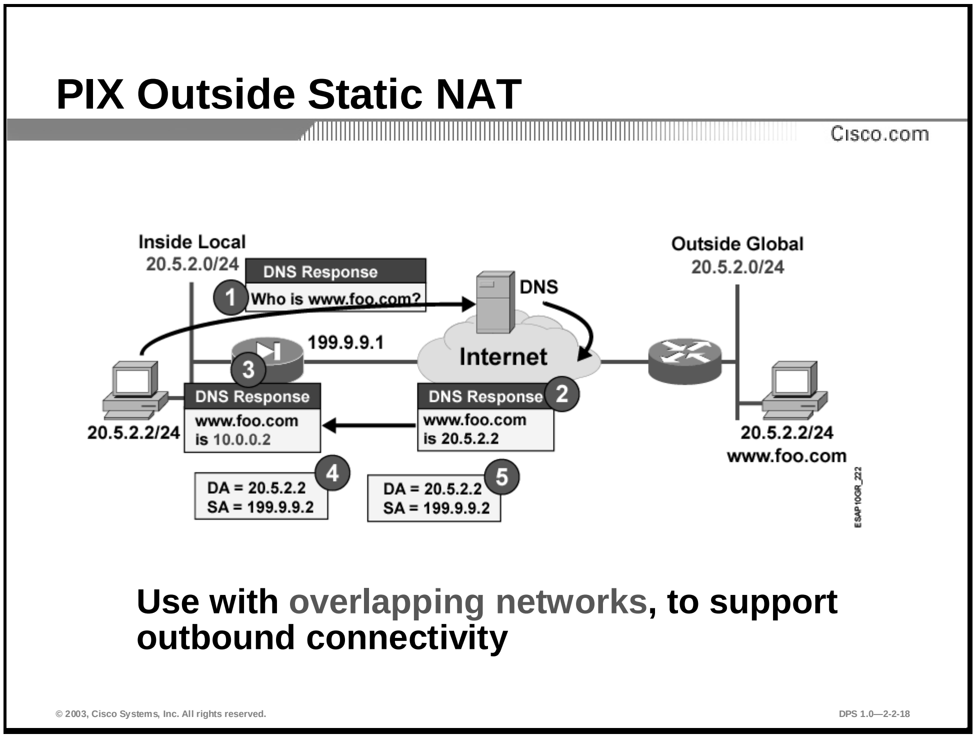
This section will enable the learner to configure bidirectional NAT using the Cisco Secure PIX Firewall.

## Introduction

Configuration of bidirectional NAT on PIX Firewall platforms is fundamentally different to Cisco IOS devices. This section presents the configuration philosophy for outside NAT on the PIX Firewall, as well as the caveats involved when configuring outside NAT.

## Command Summary

In PIX/OS releases prior to 6.2, the PIX, without the **alias** command, only allows the translation of the inside addresses. With the **alias** command, the PIX also allows translation of the outside addresses. However, translation of outside addresses is limited to static translation.



## PIX Outside Static NAT

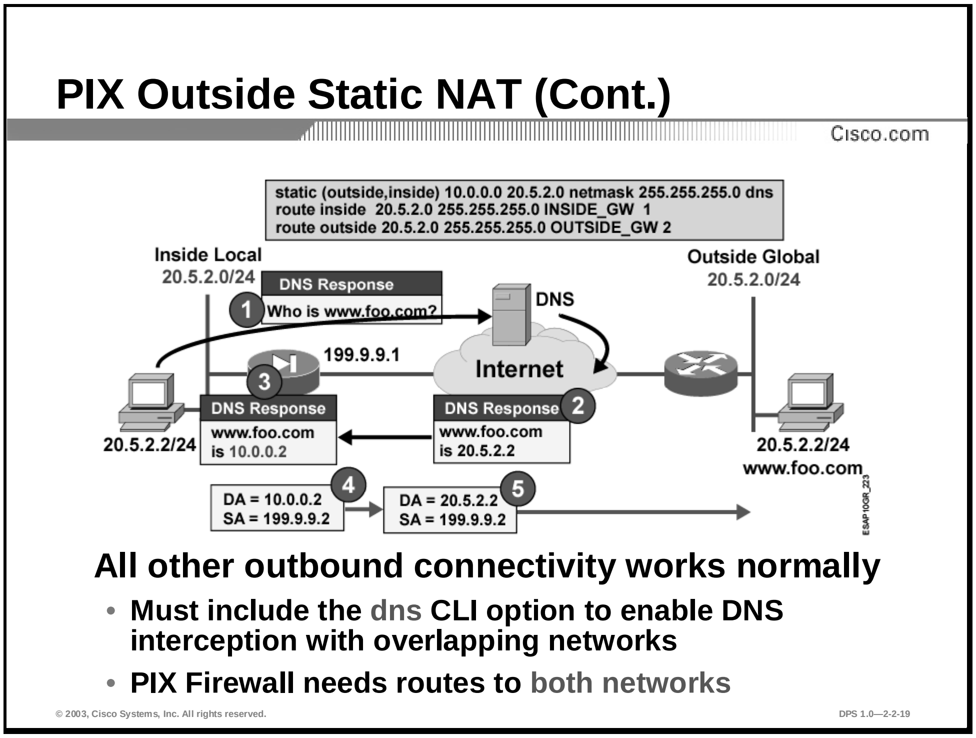
Cisco.com

**Use with overlapping networks, to support outbound connectivity**

DPS 1.0—2-2-18

# PIX Firewall Outside Static NAT

PIX/OS 6.2 enhances the static command by providing outside static NAT capability. Outside static NAT should be used to support overlapping networks, where full bidirectional connectivity is required. The static command presents an outside overlapping network to the inside networks using a static one-to-one translation. Network statics can be used to simplify configuration by configuring a single rule for the whole overlapping network.



## PIX Outside Static NAT (Cont.)

Cisco.com

```
static (outside,inside) 10.0.0.0 20.5.2.0 netmask 255.255.255.0 dns
route inside  20.5.2.0 255.255.255.0 INSIDE_GW  1
route outside 20.5.2.0 255.255.255.0 OUTSIDE_GW 2
```

**All other outbound connectivity works normally**

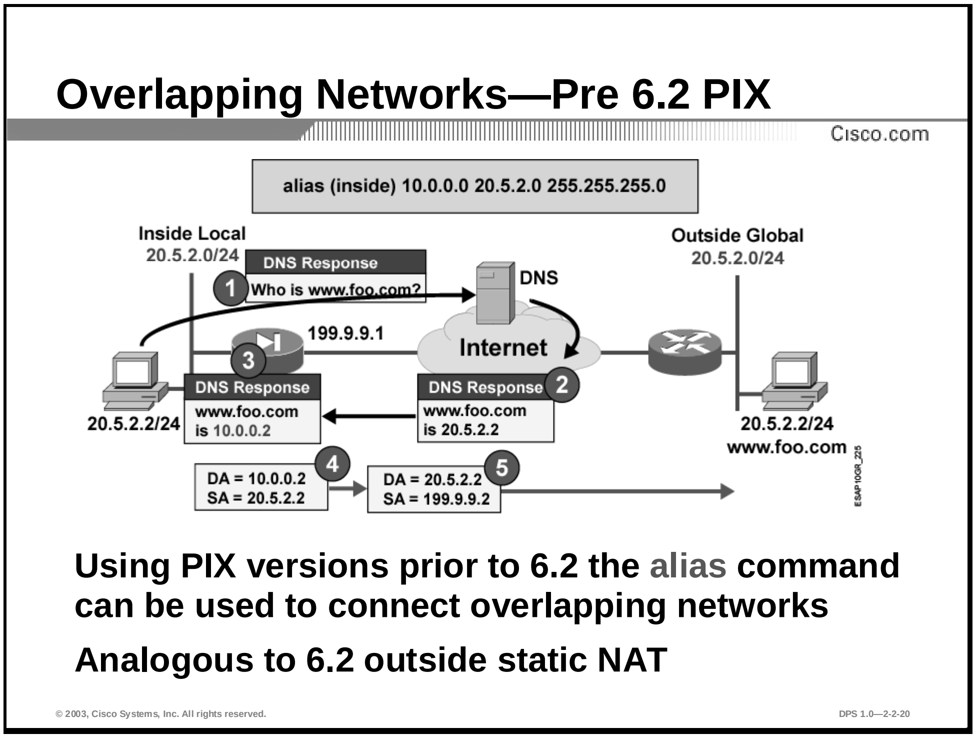
- **Must include the dns CLI option to enable DNS interception with overlapping networks**
- **PIX Firewall needs routes to both networks**

DPS 1.0—2-2-19

The example presented in the picture illustrates two networks with overlapping addresses, and the associated PIX Outside Static NAT configuration. The outside overlapping network is presented to the inside networks as the 10.0.0.0/24 network. By including the **dns** keyword, the PIX will intercept all DNS replies, and if the address within the reply is from the 20.5.2.0/24 network, the PIX will replace it with an address from the 10.0.0.0/24 range.

Routes to the overlapping network must exist on the PIX, pointing to both interfaces, where the overlapping networks are reachable. As the PIX Firewall is not a router, it will not attempt to route the packet back on the same interface, on which it was received, but will instead route the packet to the other available route through another interface.

| | |
|---|---|
| **Note** | When using outside static NAT, all other outbound connectivity still works as originally configured. Contrast this do outside dynamic NAT, when outbound connectivity is not possible, except to the global pool’s addresses on the inside interface. |



Overlapping Networks—Pre 6.2 PIX

Cisco.com

alias (inside) 10.0.0.0 20.5.2.0 255.255.255.0

Using PIX versions prior to 6.2 the alias command can be used to connect overlapping networks

Analogous to 6.2 outside static NAT

DPS 1.0—2-2-20

# Outside NAT using Pre-6.2 PIX/OS

Using the **alias** command of the pre-6.2 versions of PIX/OS, only a static outside NAT mapping can be configured. The alias command is equivalent to the 6.2 outside static NAT command.

## The nat Command Reference

To associate a network with a pool of global IP addresses, use the **nat** gloabal configuration command. There are two possible usages of the **nat** command:

**nat** [(*if_name*)] *id address* [*netmask* [**outside**] [**dns**] [**norandomseq**] [**timeout** *hh:mm:ss*] [*conn_limit* [*em_limit*]]]

**no nat** [(*if_name*)] *id address* [*netmask* [**outside**]

and

**nat** [(*if_name*)] **0 access-list** *acl_name*

**no nat** [(*if_name*)] **0** [**access-list** *acl_name*]

### Table 2: nat Parameters

| Parameter | Description |
|---|---|
| **access-list** | Associates **access-list** command statements to the **nat 0** command and exempts traffic that matches the access-list from NAT processing. |
| *acl_name* | The access list name. |
| **clear nat** | Removes **nat** command statements from the configuration. |

| | |
|---|---|
| *conn_limit* | The connection time limit. |
| **dns** | Specifies that DNS replies that match the xlate are translated. |
| *em_limit* | The embryonic connection limit. The default is 0, which means unlimited connections. Set it lower for slower systems, higher for faster systems. |
| *hh:mm:ss* | The timeout interval for the translation slot. However, timeout only occurs if no TCP or UDP connection is actively using the translation. |
| *id* | The id number to match with the global address pool. |
| *if_name* | The internal network interface name. |
| *local_ip* | Internal network IP address to be translated. You can use **0.0.0.0** to allow all hosts to start outbound connections. The **0.0.0.0** *local_ip* can be abbreviated as **0**. |
| *max_cons* | The maximum TCP connections permitted from the interface you specify. |
| *nat_id* | *nat_id v*alues can be **0**, **0 access list** *acl_name*, or a number greater than zero (0).<br>A *nat_id* that is **0** specifies the inside hosts for identity translation. Identity translations are translations that map an address to itself. The restriction is that the traffic must initiate from an inside host.<br>A *nat_id* that is **0 access list** *acl_name* specifies the traffic to exempt from NAT processing, based on the access list specified by *acl_name*. This is useful in Virtual Private Network (VPN) configuration where traffic between private networks should be exempted from NAT.<br>A *nat_id* that is a number greater than zero (0) specifies the inside hosts for dynamic address translation. The dynamic addresses are chosen from a global address pool created with the **global** command, so the *nat_id* number must match the *global_id* number of the global address pool you want to use for dynamic address translation. |
| *netmask* | Network mask for *local_ip*. You can use **0.0.0.0** to allow all outbound connections to translate with IP addresses from the global pool. The netmask **0.0.0.0** can be abbreviated as **0**. |
| **norandomseq** | Do not randomize the TCP packet's sequence number. Only use this option if another inline firewall is also randomizing sequence numbers and the result is scrambling the data. Using this option disables TCP Initial Sequence Number (ISN) randomization protection. Without this protection, inside hosts with weak self-ISN protection become more vulnerable to TCP connection hijacking. |
| **outside** | Specifies that the **nat** command apply to the outside interface address. For access control, IPSec, and AAA use the real outside address. |
| **timeout** | Sets the idle timeout value for the translation slot. |

## The static Command Reference

To configure a persistent one-to-one address translation rule by mapping a local IP address to a global IP address use the **static** global configuration command. This is also known as Static Port Address Translation (Static PAT). There are two possible usages of the **static** command:

**static** [**(***prenat_interface***,** *postnat_interface***)**] {**mapped_address**| **interface**} *real_address* [**dns**] [**netmask** *mask*] [**norandomseq**] [*connection_limit* [*em_limit*]]

**no static** [**(***prenat_interface***,** *postnat_interface***)**] {**mapped_address**| **interface**} *real_address* [**dns**] [**netmask** *mask*] [**norandomseq**] [*max_conns* [*em_limit*]]

and

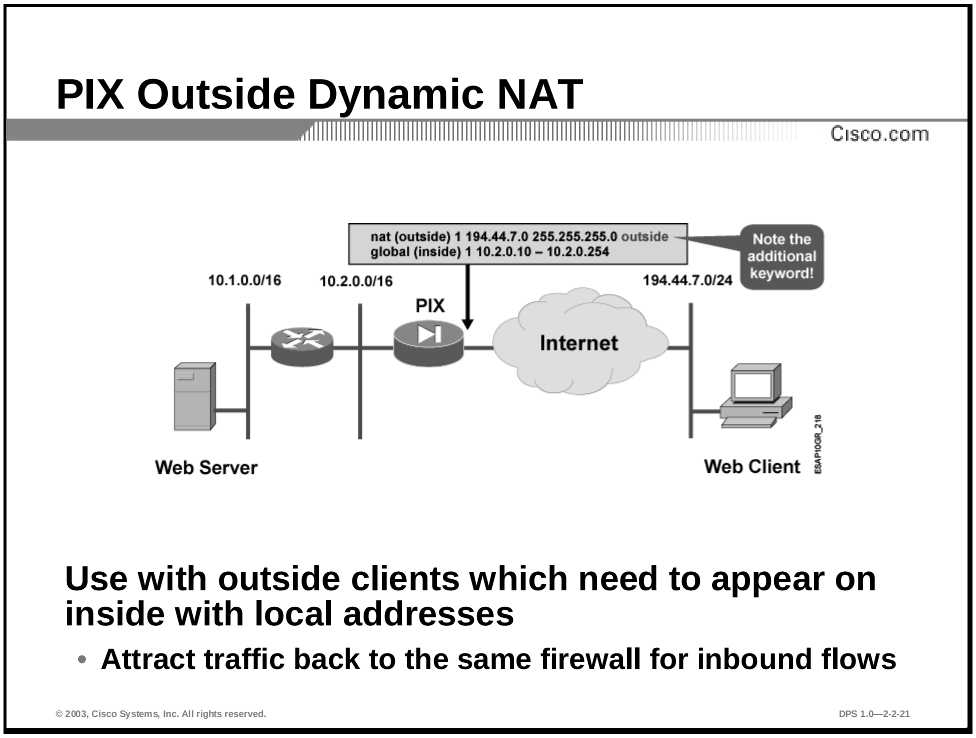
**static [(***internal_if_name, external_if_name***)] {tcp** | **udp**}{*global_ip* | **interface**} *global_port local_ip local_port* [**netmask** *mask*][*max_conns* [*emb_limit* [**norandomseq**]]]

**no static [(***internal_if_name, external_if_name***)] {tcp** | **udp**}{*global_ip* | **interface**} *global_port local_ip local_port* [**netmask** *mask*][*max_conns* [*emb_limit* [**norandomseq**]]]

### Table 3: static Parameters

| Parameter | Description |
|---|---|
| **dns** | Specifies that DNS replies that match the xlate are translated. |
| *em_limit* | The embryonic connection limit. An embryonic connection is one that has started but not yet completed. Set this limit to prevent attack by a flood of embryonic connections. The default is 0, which means unlimited connections. |
| *external_if_name* | The external network interface name. The lower security level interface you are accessing. |
| *global_ip* | A global IP address. This address cannot be a PAT IP address. The IP address on the lower security level interface you are accessing. |
| **interface** | Specifies to overload the global address from interface. |
| *internal_if_name* | The internal network interface name. The higher security level interface you are accessing. |
| *local_ip* | The local IP address from the inside network. The IP address on the higher security level interface you are accessi. |
| *mapped_address* | The address *real_address* is translated into. |
| *mapped_port* | The port *real_port* is translated into. |
| *mask or network_mask* | The network mask pertains to both *global_ip* and *local_ip*. For host addresses, always use 255.255.255.255. For network addresses, use the appropriate class mask or subnet mask; for example, for Class A networks, use 255.0.0.0. An example subnet mask is 255.255.255.224. |
| *max_conns* | The maximum number of connections permitted through the static at the same time. |
| **netmask** | Reserve word required before specifying the network mask. |
| **norandomseq** | Do not randomize the TCP/IP packet's sequence number. Only use this option if another inline firewall is also randomizing sequence numbers and the result is scrambling the data. Use of this option opens a security hole in the PIX Firewall. |
| *postnat_interface* | The outside interface when *prenat_interface* is the inside interface. However, if the outside interface is used for *prenat_interface*, then the translation is applied to the outside address and the *postnat_interface* is the inside interface. |
| *prenat_interface* | Usually the inside interface, in which case the translation is applied to the inside address. |

| | |
|---|---|
| *real_address* | The address to be mapped. |
| *real_port* | The port to be mapped. |



## PIX Outside Dynamic NAT

```
nat (outside) 1 194.44.7.0 255.255.255.0 outside
global (inside) 1 10.2.0.10 – 10.2.0.254
```

**Use with outside clients which need to appear on inside with local addresses**

- **Attract traffic back to the same firewall for inbound flows**

DPS 1.0—2-2-21

# PIX Outside Dynamic NAT

For outside dynamic translation, outside dynamic NAT can be configured with version 6.2, and static outside NAT is now available using normal PIX NAT syntax (the *static* command instead of the *alias* command). With dynamic outside translation in PIX/OS 6.2, the **nat** command includes two additional keywords: "**dns**" and "**outside**".

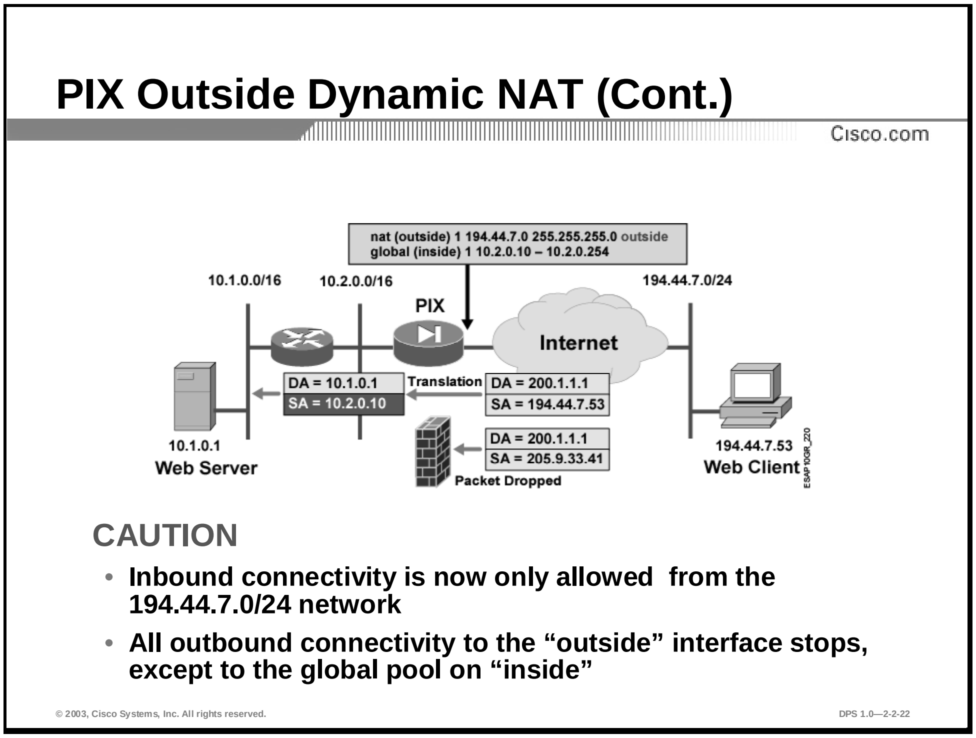
The **dns** keyword specifies that DNS replies should be intercepted and the addresses inside them translated to appropriate outside local addresses. The **outside** keyword specifies that the **nat** statement applies to the *outside* address.

| | |
|---|---|
| **Note** | Outside NAT does not work with application inspection ("fixup") for Internet Locator Service (ILS). |

Once the outside NAT is configured, when a packet arrives at the outer (less secure) interface of the PIX Firewall, the PIX Firewall attempts to locate an existing xlate in the connections database. If no xlate exists, it searches the NAT policy from the running configuration. If a NAT policy is located, the PIX Firewall creates an xlate and inserts it into the database. The PIX Firewall then rewrites the source address to the mapped or global address and transmits the packet on the inside interface. Once the xlate is established, the addresses of any subsequent packets can be quickly translated by consulting the entries in the translation table.

In the example presented in the figure, hosts from a dedicated outside network are allowed to access the local Web server because an outside NAT configuration makes them appear as the local host. Therefore, the local (internal) routers treat them accordingly, applying the same rules as for inside hosts.

**Caution** Using the **nat outside** PIX Firewall command stops any other outbound connectivity from the inside network, except to the global pool on “inside”. If other outbound connectivity is required, outside static translations need to be setup from outside to inside (which is a very non-scalable solution). Therefore, outside dynamic NAT should be used primarily when outside clients need to be translated to the inside, and no other outbound connectivity is desired.



## PIX Outside Dynamic NAT (Cont.)

Cisco.com

```
nat (outside) 1 194.44.7.0 255.255.255.0 outside
global (inside) 1 10.2.0.10 – 10.2.0.254
```

**CAUTION**

- **Inbound connectivity is now only allowed from the 194.44.7.0/24 network**
- **All outbound connectivity to the "outside" interface stops, except to the global pool on "inside"**

DPS 1.0—2-2-22

All types of packets from 194.44.7.0/24 are translated and forwarded, except those filtered by an access list.

An **xlate** entry is created dynamically in order to quickly translate all the packets associated with the session.

## Practice

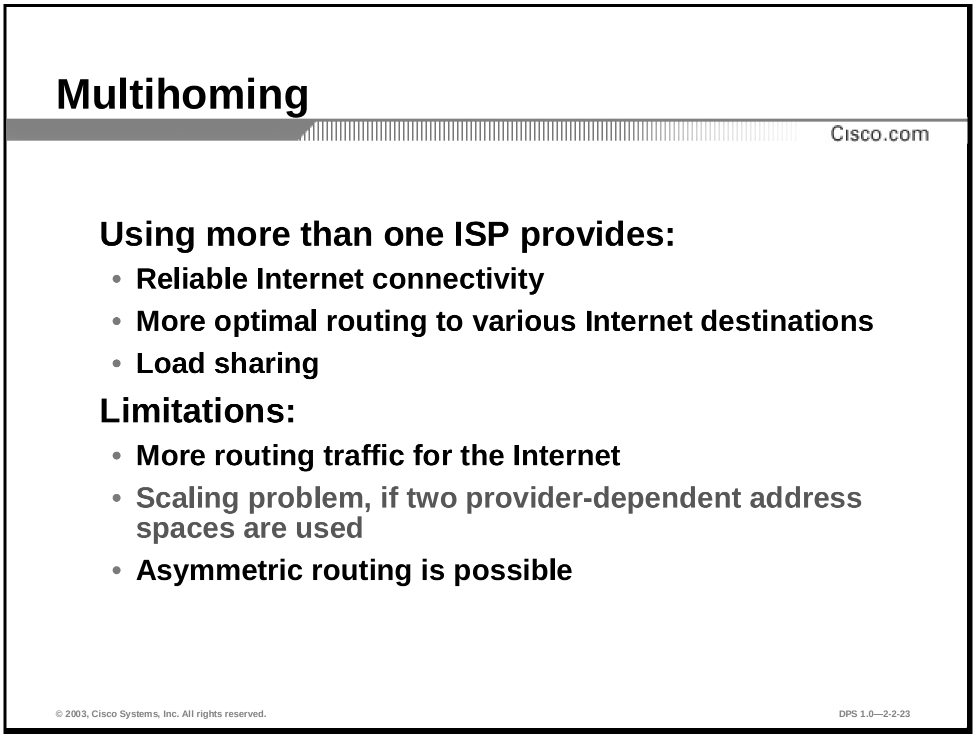
Q1) When is DNS interception used with PIX/OS version 6.2?

A) DNS interception is automatically enabled when specifying the additional keyword **outside** for the NAT commands

B) DNS interception is only possible with the **alias** command

C) DNS interception can be enabled by specifying the additional keyword **dns** for the NAT commands

# Multihoming



## Multihoming

Using more than one ISP provides:

- Reliable Internet connectivity
- More optimal routing to various Internet destinations
- Load sharing

Limitations:

- More routing traffic for the Internet
- Scaling problem, if two provider-dependent address spaces are used
- Asymmetric routing is possible

© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—2-2-23

## Objective

This section will enable the learner to explain the concept of multi-homing using NAT and choose it in appropriate design situations.

## Introduction

Enterprise networks are often attached to more than one ISP for performance and reliability. This concept is called (ISP) multihoming, and requires specific scaling considerations and advanced NAT solutions.

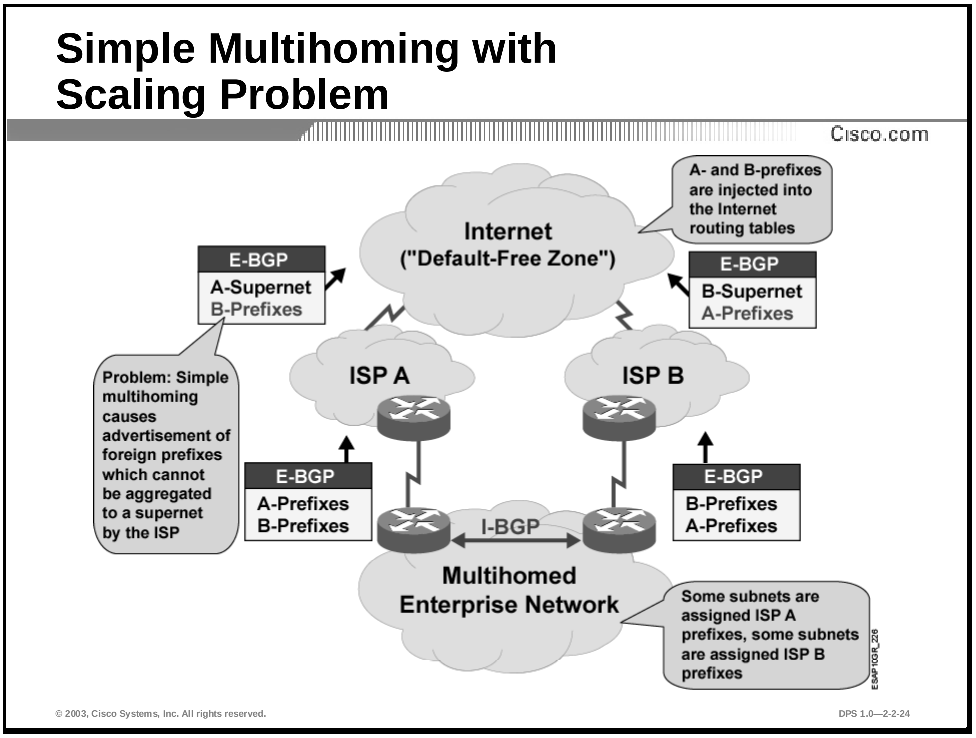
Multihoming provides the following features:

- Redundancy of the Internet connection, which is needed for mission-critical Internet applications
- More optimal routing to various Internet destinations
- Load-sharing over multiple connections to provide higher performance

The limitations of multihoming are the following:

- The routing table in the Internet increases, as there are multiple paths to the multihomed network's (possible multiple, if provider-dependent address space is used) prefixes

- Traffic flow can be asymmetric, impacting performance



## Simple Multihoming with Scaling Problem

A- and B-prefixes are injected into the Internet routing tables

Problem: Simple multihoming causes advertisement of foreign prefixes which cannot be aggregated to a supernet by the ISP

Some subnets are assigned ISP A prefixes, some subnets are assigned ISP B prefixes

DPS 1.0—2-2-24

# Example

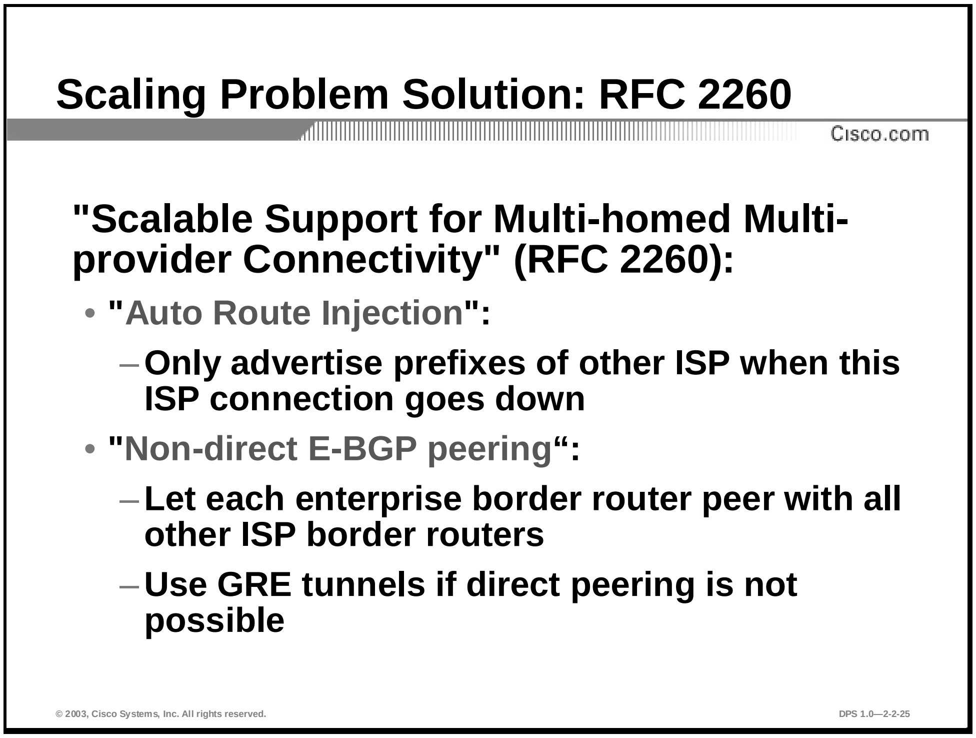
This figure illustrates an example of ISP multihoming without any NAT configuration. The provider network employs two border gateways, each connected to an ISP. E-BGP peering is configured from each border gateway directly to its connected ISP, and I-BGP is configured between the two border gateways.

Each ISP is assigned a block of its own prefixes, part of which it assigns to each customer. In our case, a multihomed customer gets two prefixes: one from ISP A (A-prefixes) and one from ISP B (B-prefixes). For redundancy, the customer advertises BOTH prefixes to BOTH ISPs.

**Note** If a customer can obtain provider-independent address space, use of classic BGP can provide robust multihoming, without the need for dual-NAT.

ISP A cannot aggregate the advertised prefixes assigned by ISP B, and ISP B cannot aggregate the ISP A prefixes. Therefore, both ISP A and B will advertise these prefixes into the "default-free" zone of the Internet, thereby increasing the amount of routes.

Thus, multihoming raises scalability questions for the Internet, because the default-free zone becomes increasingly polluted with non-aggregated prefixes.



## Scaling Problem Solution: RFC 2260

**"Scalable Support for Multi-homed Multi-provider Connectivity" (RFC 2260):**

- **"Auto Route Injection":**
  - **Only advertise prefixes of other ISP when this ISP connection goes down**
- **"Non-direct E-BGP peering“:**
  - **Let each enterprise border router peer with all other ISP border routers**
  - **Use GRE tunnels if direct peering is not possible**

DPS 1.0—2-2-25

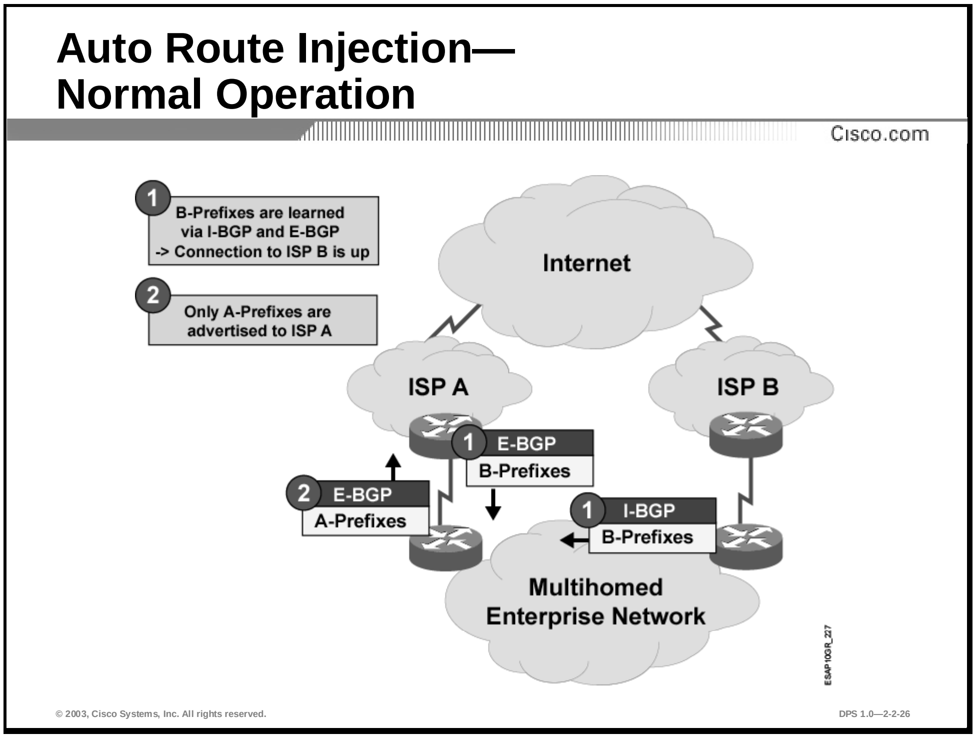
## Solution

The RFC 2260, entitled “Scalable Support for Multi-homed Multi-provider Connectivity” suggests two solutions to the scalability problem:

- **Auto Route Injection:** Requires a mechanism to determine if the “other” ISP connection is up or down. A router should only advertise its foreign prefixes to its ISP if the other router losses its connection to the “foreign” ISP.

- **Non-direct E-BGP peering:** Requires an additional non-direct back-up E-BGP peering from each enterprise border gateway to each “foreign” ISP. Thus, the prefixes can still advertise to the correct ISP if the main peering connection is lost.

To establish non-direct peering use generic routing encapsulation (GRE) tunnels to simulate a direct peering connection.



## Auto Route Injection— Normal Operation

Cisco.com

1 B-Prefixes are learned via I-BGP and E-BGP -> Connection to ISP B is up

2 Only A-Prefixes are advertised to ISP A

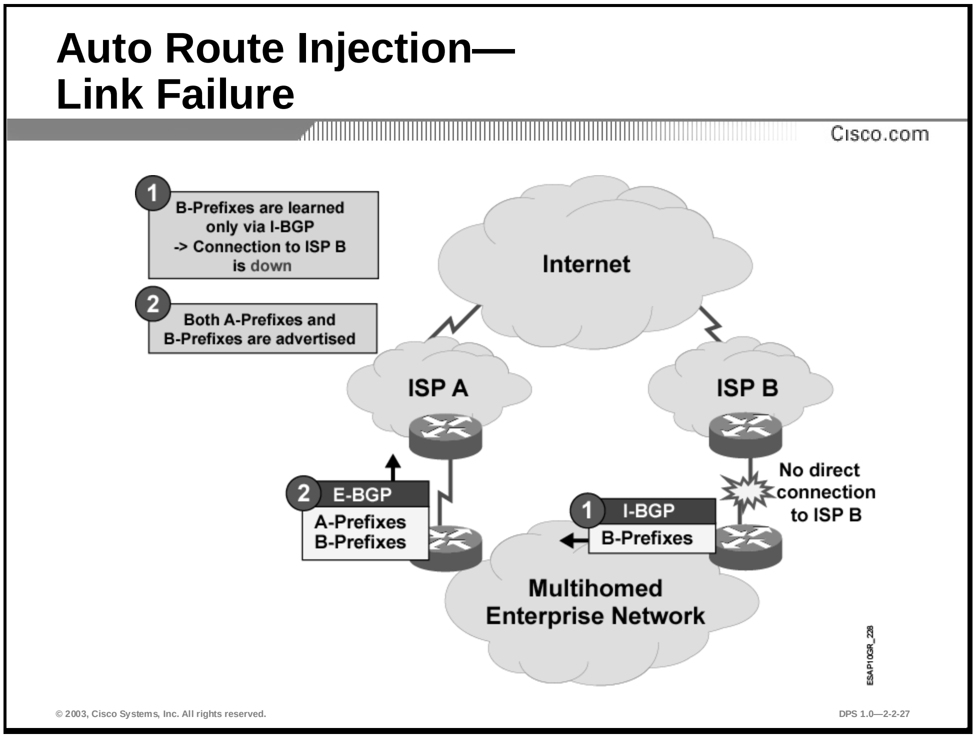
DPS 1.0—2-2-26

## Example

The example illustrated describes auto route injection:

**Step 1** B-prefixes of the enterprise network are advertised via ISP A. Thus, the left border router knows that the right border router has the correct connection to ISP B.

**Step 2** The left border router advertises the A-prefixes only to ISP A.



## Auto Route Injection— Link Failure

Cisco.com

1 B-Prefixes are learned only via I-BGP -> Connection to ISP B is down

2 Both A-Prefixes and B-Prefixes are advertised

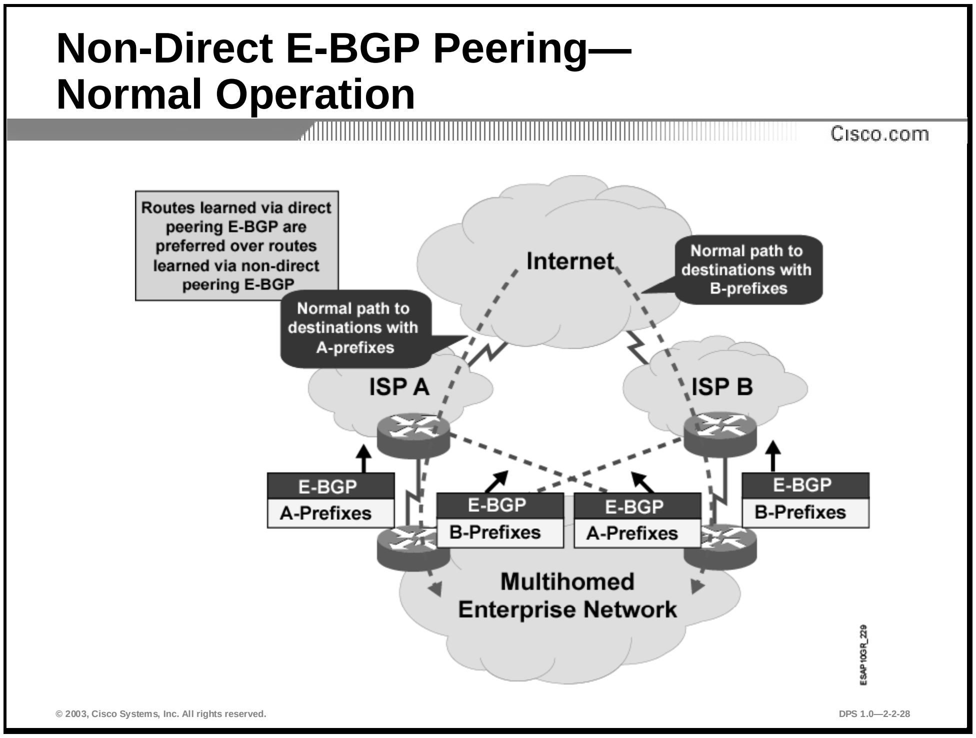
DPS 1.0—2-2-27

**Step 3** When ISP A does not advertise B-prefixes any longer, the left border router advertises to ISP A both A-prefixes and B-prefixes.

Although strictly an implementation detail, determining the outstanding prefixes can potentially be a costly operation for a large set of routes. An alternate solution is to:

**Step 1** Use a selected single, or more, address prefix received from an ISP (the ISP's backbone route for example).

**Step 2** Configure the enterprise border router to perform auto route injection if the selected prefix is not present via IBGP.



## Non-Direct E-BGP Peering—Normal Operation

Cisco.com

Routes learned via direct peering E-BGP are preferred over routes learned via non-direct peering E-BGP

Normal path to destinations with A-prefixes

Normal path to destinations with B-prefixes

DPS 1.0—2-2-28

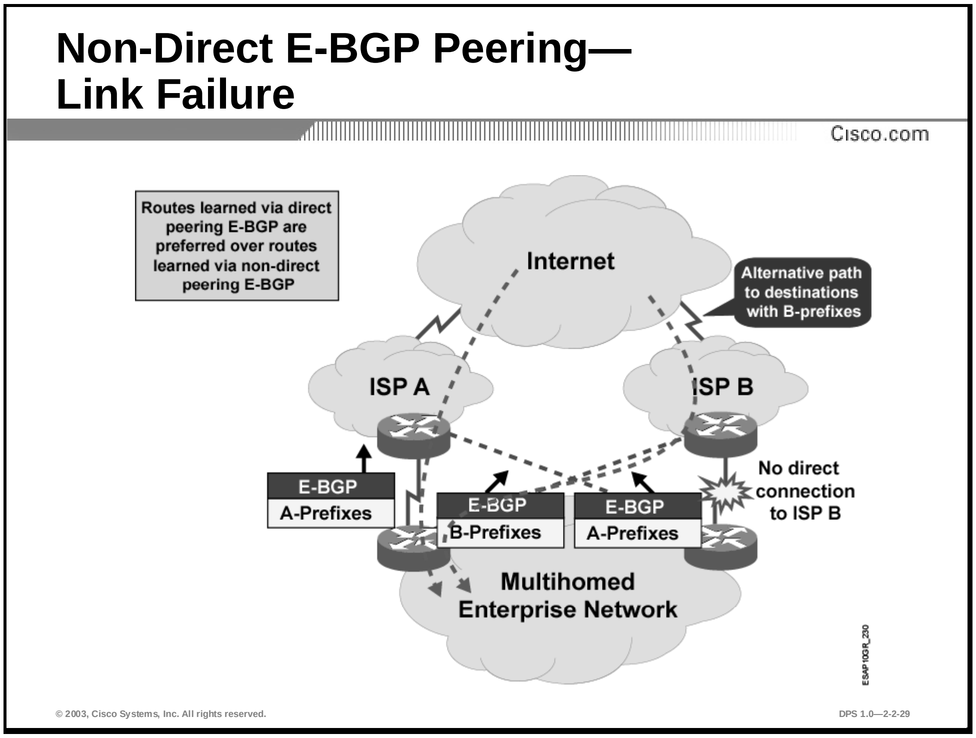
## Example

This example illustrates non-direct E-BGP peering. Both enterprise border gateways not only maintain an E-BGP session to the directly connected ISPs, but also to the non-directly connected “foreign” ISPs. The non-direct peering is established using GRE tunnels.

Because routes learned via direct E-BGP peering can be preferred over routes learned via non-direct E-BGP peering, during a normal operation:

- A-prefixes in the enterprise network are reached via ISP A

- B-prefixes are reached via ISP B



# Non-Direct E-BGP Peering— Link Failure

Cisco.com

Routes learned via direct peering E-BGP are preferred over routes learned via non-direct peering E-BGP

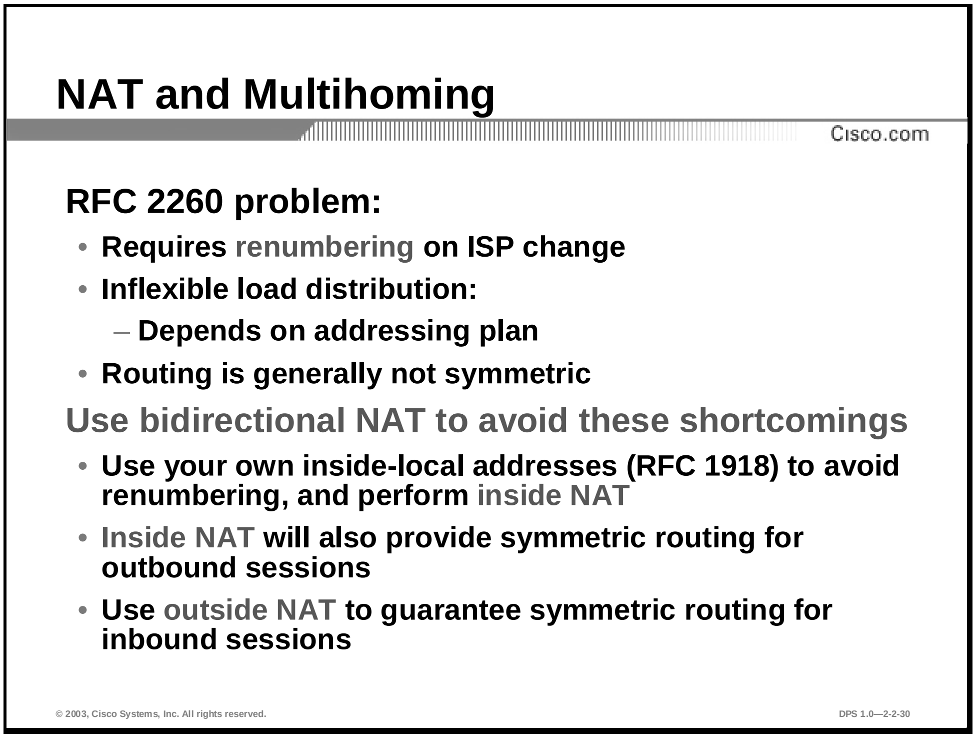
Alternative path to destinations with B-prefixes

No direct connection to ISP B

DPS 1.0—2-2-29

If the direct connection to ISP B goes down, the enterprise network routes all traffic to B-prefixes over the GRE tunnel.

Non-direct E-BGP peering completely eliminates the overhead in the “default-free” zone. In this example both enterprise border routers have established E-BGP peerings with all ISPs.



NAT and Multihoming

RFC 2260 problem:

- Requires renumbering on ISP change
- Inflexible load distribution:
  - Depends on addressing plan
- Routing is generally not symmetric

Use bidirectional NAT to avoid these shortcomings

- Use your own inside-local addresses (RFC 1918) to avoid renumbering, and perform inside NAT
- Inside NAT will also provide symmetric routing for outbound sessions
- Use outside NAT to guarantee symmetric routing for inbound sessions

DPS 1.0—2-2-30

## Why NAT Is Needed with Multihoming?

The simple multihoming idea has some limitations, when it comes to addressing:

- The entire enterprise network must be renumbered with the new ISP prefixes whenever there is an ISP change
- The load distribution for outbound is not flexible and depends on the addressing plan implemented in the enterprise network (i.e. the part of the inside network, which uses addresses from ISP-A, must always exit through ISP-A)
- Routing is not symmetric, i.e. incoming sessions to a host might arrive over the ISP-A connection, but the reverse traffic flows over the ISP-B connection, possibly impacting performance

## Solution—Bidirectional NAT

The solution for the above issues is to use dual (bidirectional) NAT on the border routers. Bidirectional NAT involves inside NAT (translation of inside addresses) and outside NAT (translation of outside addresses).

Inside NAT will:

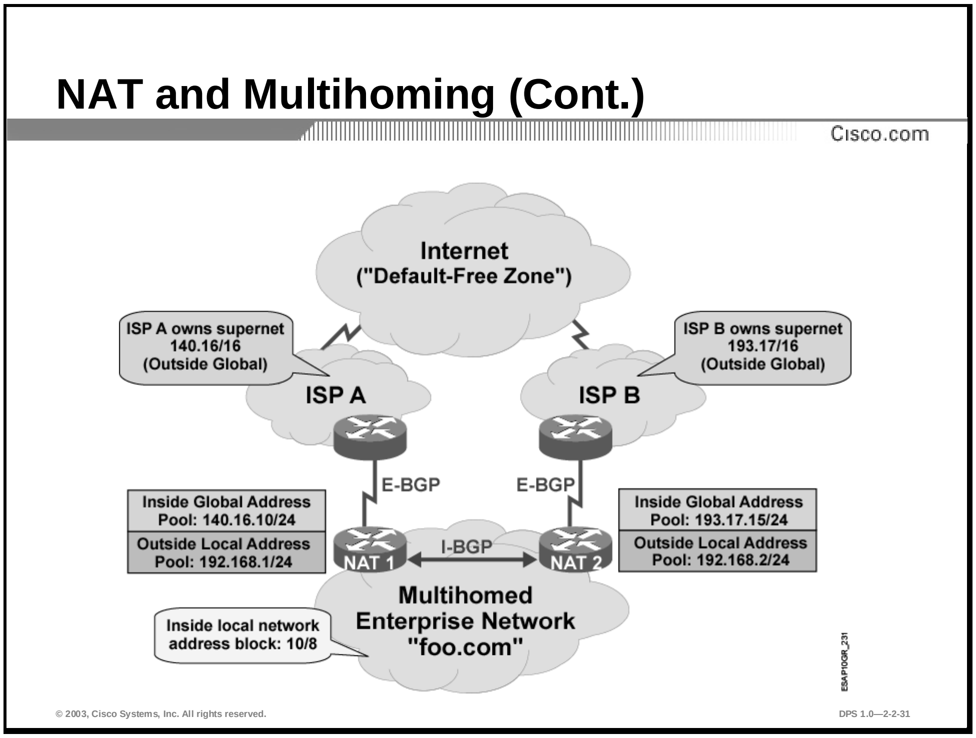
- Enable the enterprise to use RFC 1918 address space internally, eliminating the need to renumber on ISP change – only the address pools of inside global addresses need to be changed on the NAT device

- Guarantee symmetric routing for outbound connections – traffic flowing out to ISP-A will be translated to ISP-A prefixes, attracting reverse traffic over ISP-A as well.

Outside NAT will:

- Guarantee symmetric routing for inbound connections – traffic flowing in from ISP-A will be translated to a pool of outside local addresses by the NAT device, and the reverse (outbound) flow will be attracted to the same NAT device, routing traffic back to ISP-A.

**Note** The multi-homing examples shown using BGP all require an IOS router rather than a PIX Firewall since the PIX Firewall doesn't support BGP routing. For a full head-end design, you would then need to use the PIX in a non-NAT firewall role (since NAT is done on the access router), or alternatively perform NAT again on the router.



NAT and Multihoming (Cont.)

DPS 1.0—2-2-31

## Example

The figure illustrates an example where an enterprise, “foo.com”, is connected to two ISPs, ISP A and ISP B:

- ISP A allocates out of its 140.16/16 address block, a sub-block 140.16.10/24 to the enterprise

- ISP B allocates out of its 193.17/16 address block, a sub-block 193.17.15/24 to the enterprise

Both 140.16.10/24 and 193.17.15/24 are inside global addresses of the enterprise:

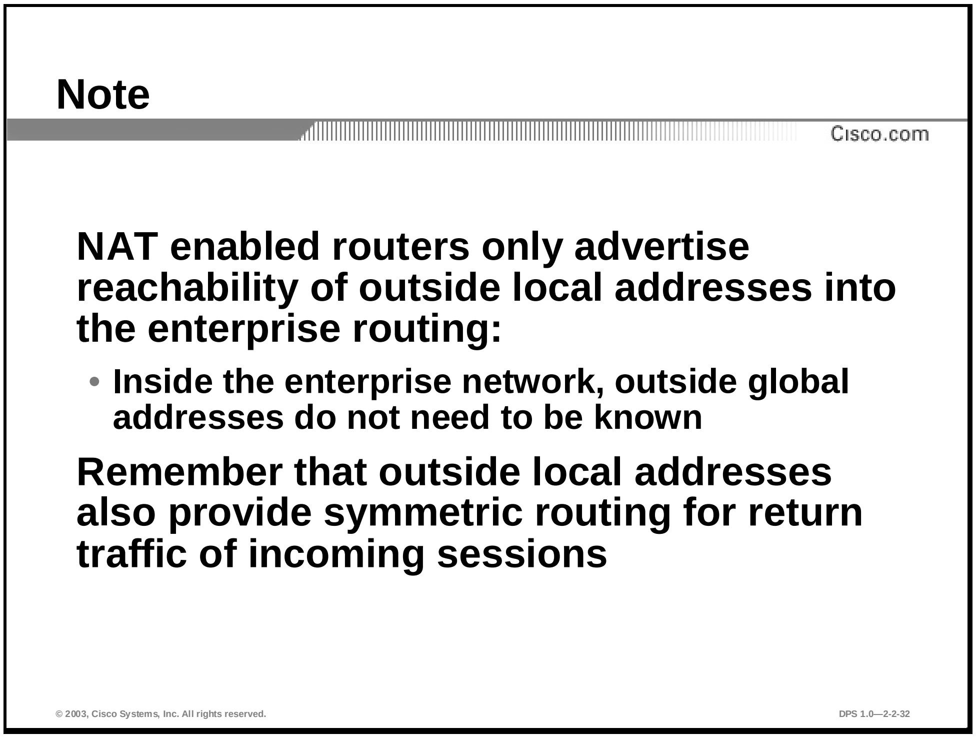
- NAT 1, which connects the enterprise to ISP A, advertises to ISP A direct reachability to 140.16.10/24

- NAT 2, which connects the enterprise to ISP B, advertises to ISP B direct reachability to 193.17.15/24

For its outside local addresses the enterprise uses addresses out of the private address space. For NAT 1 the enterprise allocates 192.168.1/24 block and for NAT 2 the enterprise allocates 192.168.2/24 block:

- NAT 1 advertises into the enterprise routing direct reachability to 192.168.1/24

- NAT 2 advertises into the enterprise routing direct reachability to 192.168.2/24



## Note

NAT enabled routers only advertise reachability of outside local addresses into the enterprise routing:

- Inside the enterprise network, outside global addresses do not need to be known

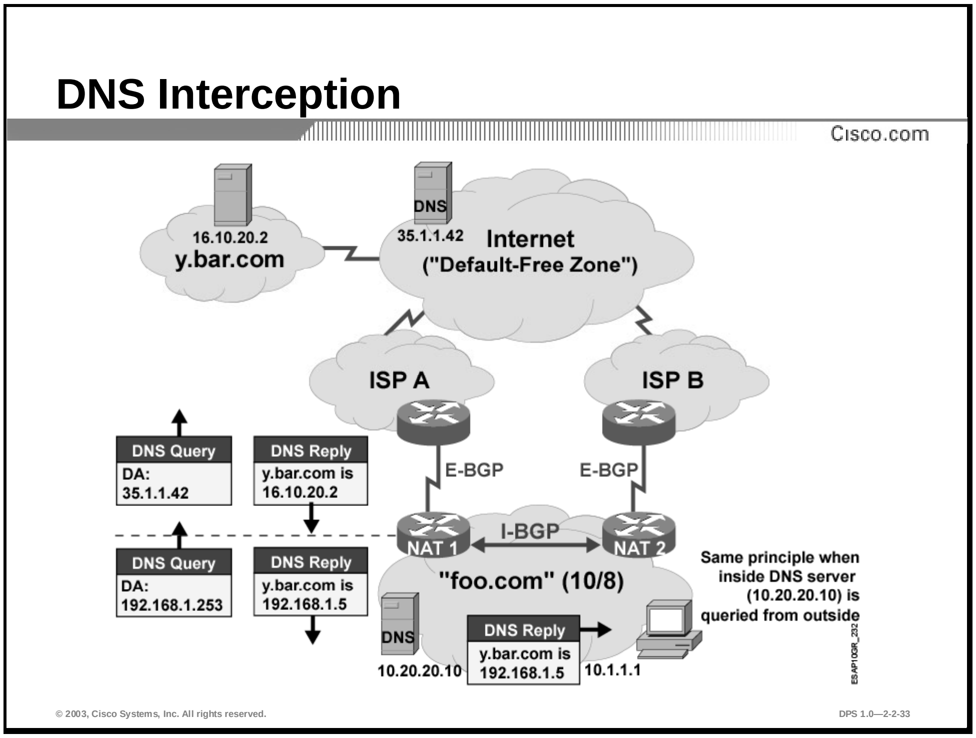
Remember that outside local addresses also provide symmetric routing for return traffic of incoming sessions

DPS 1.0—2-2-32

## Reduced Internal Routing Table Size

Because outside NAT is enabled, no outside global address is visible to the enterprise network. Therefore, the border routers only advertise a route to the outside local addresses, representing any current translation slot.

Hence, the network only advertises the inside local routes and outside local routes in the enterprise routing.



## DNS Interception

DNS Query
DA:
35.1.1.42

DNS Reply
y.bar.com is
16.10.20.2

DNS Query
DA:
192.168.1.253

DNS Reply
y.bar.com is
192.168.1.5

DNS Reply
y.bar.com is
192.168.1.5

Same principle when inside DNS server (10.20.20.10) is queried from outside

DPS 1.0—2-2-33

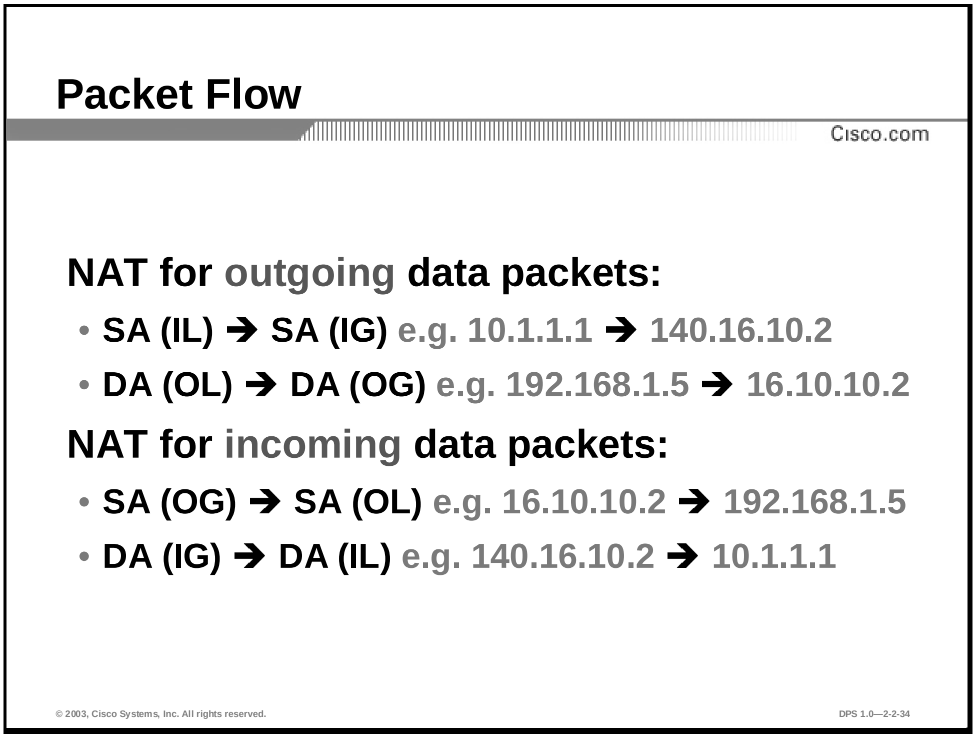
## Accessing an Outside Server via Hostname

This figure illustrates the translations involved for a DNS request issued by the local host 10.1.1.1 for the outside host “y.bar.com”:

**Step 1** The NAT router maintains a static translation slot for the outside DNS server so that an outbound DNS request is mapped from the outside local address 192.168.1.253 to the outside global address 35.1.1.42.

**Step 2** The A records of the DNS reply are translated into an outside local address, that is from 16.10.20.2 to 192.168.1.5.

**Step 3** The enterprise host 10.1.1.1 uses the latter address as the destination address for y.bar.com.



## Packet Flow

Cisco.com

**NAT for outgoing data packets:**

- **SA (IL) ➔ SA (IG)** e.g. 10.1.1.1 ➔ 140.16.10.2
- **DA (OL) ➔ DA (OG)** e.g. 192.168.1.5 ➔ 16.10.10.2

**NAT for incoming data packets:**

- **SA (OG) ➔ SA (OL)** e.g. 16.10.10.2 ➔ 192.168.1.5
- **DA (IG) ➔ DA (IL)** e.g. 140.16.10.2 ➔ 10.1.1.1

DPS 1.0—2-2-34

## Processing a Packet Originating Inside an Enterprise Network

When a NAT receives a packet that originated inside the enterprise network:

**Step 1** NAT searches its address translation table for the outside address translation type entry whose OL address is equal to the destination IP address in the packet

**Step 2** *If no such entry is found*, the packet is discarded

*If such an entry is found*, the NAT replaces the destination address in the packet with the OG address from the found entry

**Step 3** NAT searches the address translation table for the inside address translation type entry whose IL address is equal to the source IP address in the packet

**Step 4** *If such an entry is found*, the NAT replaces the source address in the packet with the IG address from the found entry

*If no such entry is found*, the NAT:

**Step 5** Creates a new inside address translation type entry

**Step 6** Sets the IL address in the entry to the source address in the packet

**Step 7** Allocates an address out of the inside global addresses block allocated to the NAT

**Step 8** Sets the IG address in the entry to the allocated address

**Step 9** Replaces the source address in the packet with the IG address from the newly created entry

### Processing a Packet Originating Outside an Enterprise Network

When a NAT receives a packet originated outside the enterprise network:

**Step 1** NAT searches its address translation table for the inside address translation type entry whose IG address is equal to the destination IP address in the packet

**Step 2** *If no such entry is found*, the packet is discarded

*If such an entry is found*, the NAT replaces the destination address with the IL address from the found entry

**Step 3** NAT searches the address translation table for the outside address translation type entry whose OG address is equal to the source IP address in the packet

**Step 4** *If such an entry is found*, the NAT replaces the source address with the OL address from the found entry

*If no such entry is found*, the NAT:

**Step 5** Creates a new outside address translation type entry

**Step 6** Sets the OG address in the entry to the source address in the packet

**Step 7** Allocates an address out of the outside local addresses block allocated to the NAT

**Step 8** Sets the OL address in the entry to the allocated address

**Step 9** Replaces the source address in the packet with the OL address from the newly created entry

## Practice

Q1) Why can ISP multihoming introduce Internet routing scalability problems?

A) because the NAT translation tables become quite large

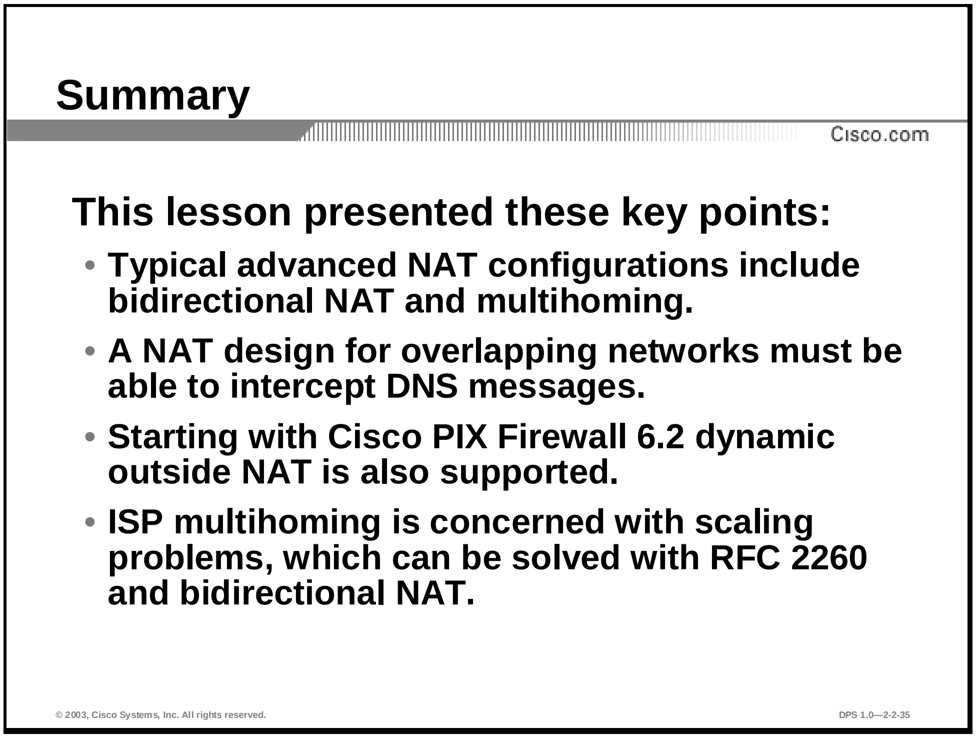
B) because the number of providers is limited

C) because the number of prefixes of the providers are limited

D) because one ISP cannot aggregate the prefixes of other ISPs learned via I-BGP sessions and the Internet becomes polluted with “long” prefixes

# Summary

This section summarizes the key points discussed in this lesson.



## Summary

Cisco.com

**This lesson presented these key points:**

- **Typical advanced NAT configurations include bidirectional NAT and multihoming.**
- **A NAT design for overlapping networks must be able to intercept DNS messages.**
- **Starting with Cisco PIX Firewall 6.2 dynamic outside NAT is also supported.**
- **ISP multihoming is concerned with scaling problems, which can be solved with RFC 2260 and bidirectional NAT.**

© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—2-2-35

## Next Steps

After completing this lesson, go to:

- Firewall Functionality module, Firewall Function lesson

## References

For additional information, refer to these resources:

- Cisco PIX Firewall and VPN Configuration Guide Version 6.2, http://www.cisco.com/univercd/cc/td/doc/product/iaabu/pix/pix_sw/v_62/config/index.htm
- Cisco IOS 12.2 Configuration Guide: Configuring IP Addressing, http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fipr_c/ipcprt1/1cfipadr.htm
- Enabling Enterprise Multihoming with Cisco IOS NAT, http://www.cisco.com/warp/public/cc/pd/iosw/ioft/ionetn/tech/emios_wp.htm

# Quiz: Design Using a NAT/PAT Solution

Complete the quiz to assess what you have learned in this lesson.

## Objectives

This quiz test your knowledge on how to:

- Design advanced NAT solutions for some common enterprise connectivity scenarios

## Instructions

Answer these questions:

1. What DNS information is translated by the Cisco IOS and PIX Firewall bidirectional NAT functionality?
2. How is outside NAT performed in PIX Firewall prior to version 6.2?
3. What are the caveats of PIX Firewall outside NAT?
4. When is bidirectional NAT needed with enterprise multihoming?

## Scoring

You have successfully completed the quiz for this lesson when you earn a score of 80 percent or better.

# Firewall Function

## Overview

Firewalls have become a de-facto standard for perimeter access control, but many of their features and limitations remain unknown to their users. This lesson introduces the function of a firewall system, and addresses its features and limitations in real-life deployments.

### Importance

This lesson provides the learner with the fundamental philosophy of network firewalls, which needs to be universally understood when firewall design is required.

### Lesson Objective

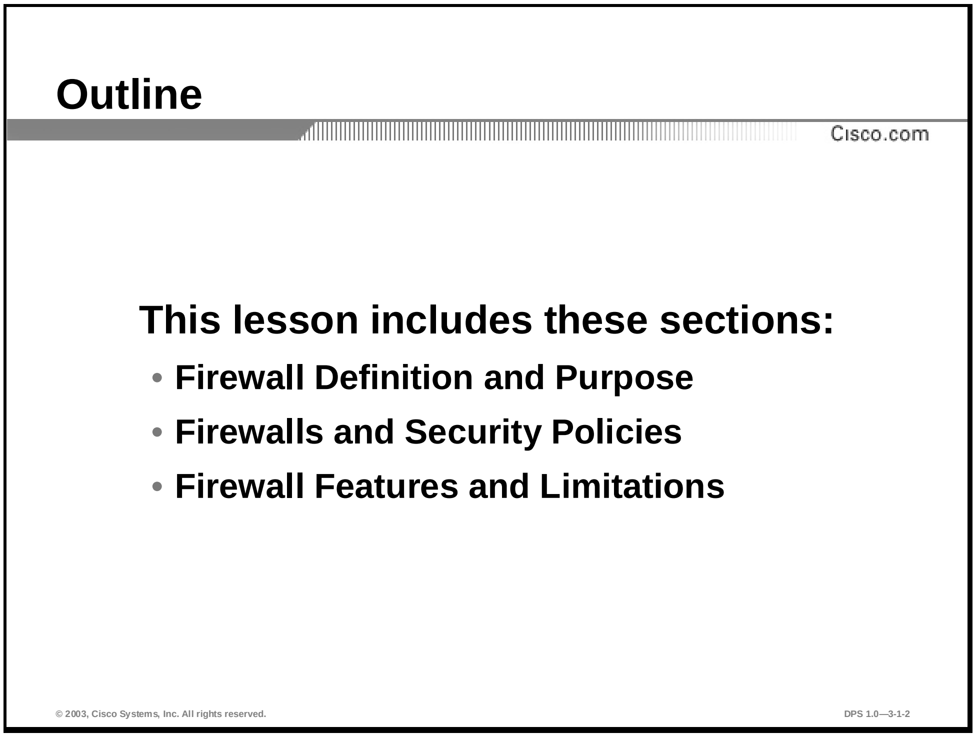
The lesson will enable the learner to explain the function of a firewall and to identify its features and limitations.

## Learner Skills and Knowledge

To fully benefit from this lesson, you must have these prerequisite skills and knowledge:

- Describe the basic enterprise perimeter connectivity options
- Describe the concept of security policy enforcement and access control

## Outline



### Outline

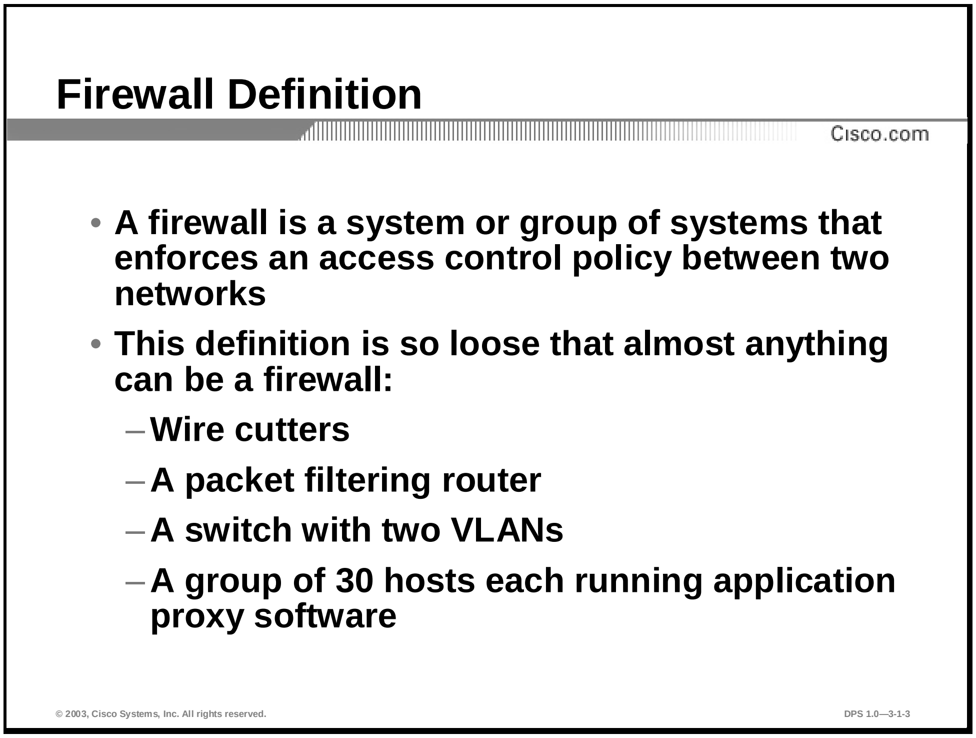
Cisco.com

**This lesson includes these sections:**

- **Firewall Definition and Purpose**
- **Firewalls and Security Policies**
- **Firewall Features and Limitations**

DPS 1.0—3-1-2

# Firewall Definition and Purpose



## Firewall Definition

- A firewall is a system or group of systems that enforces an access control policy between two networks
- This definition is so loose that almost anything can be a firewall:
  - Wire cutters
  - A packet filtering router
  - A switch with two VLANs
  - A group of 30 hosts each running application proxy software

© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—3-1-3

## Objective

The section will enable the learner to explain the function of a firewall.

## Introduction

This section introduces the definition of a firewall, and provides insight into what can or cannot be considered a firewall today. Some common properties that should be present in every firewall system are described.

## Definition

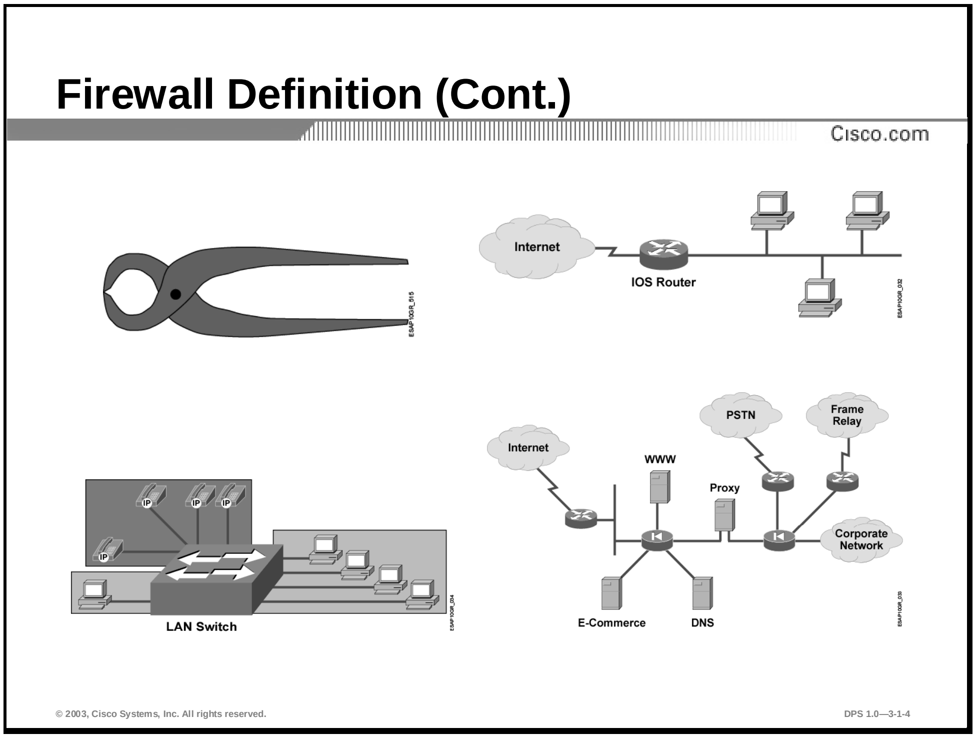
A firewall may be defined as: “a system or group of systems that enforces an access control policy between two networks.” As this definition is very generic, almost anything can be considered to be a firewall. This section explores this definition, and suggests various interpretations of the firewalling concept.

## Firewall Examples

There are many network access technologies that can be used to build a firewall. These include:

- Simple wire cutters
- Packet filtering routers

- LAN switches
- Complex systems integrating tens of hosts into a firewall system



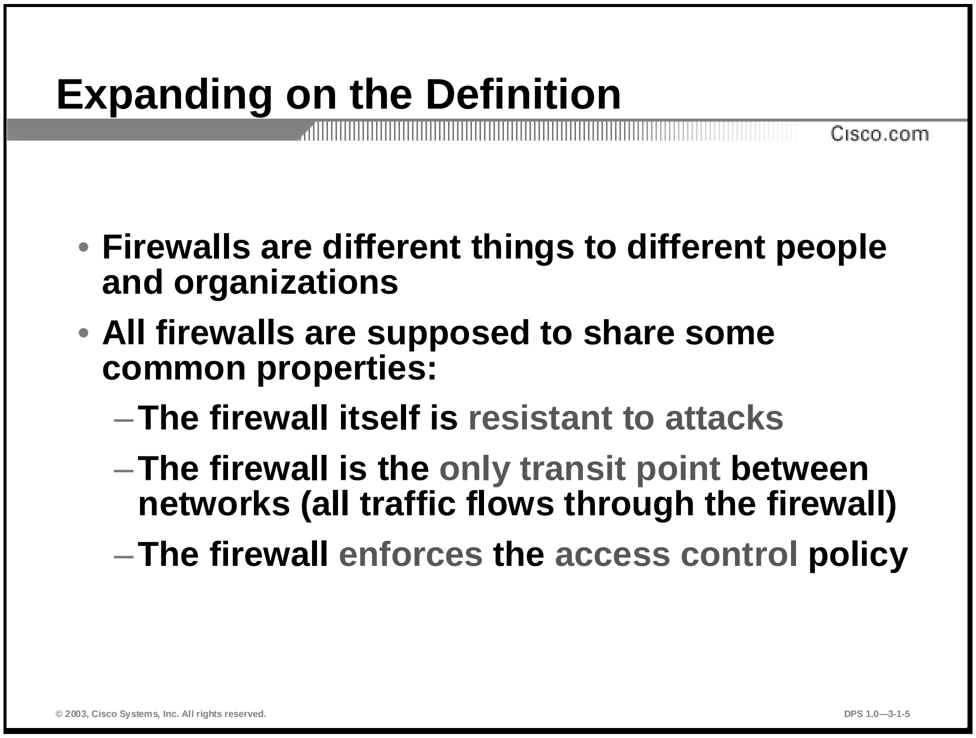
## Firewall Definition (Cont.)

DPS 1.0—3-1-4

This figure presents some implementations of the firewall concept—all of the systems can be easily classified as firewalls:

- A simple router, protecting a small network by enforcing access control for packets inbound/incoming from the Internet
- A LAN switch separating the voice and data network
- A system interconnecting multiple business partners and connecting an enterprise to the Internet

Any device, according to the definition, which performs network access control, may be called a firewall.



## Expanding on the Definition

Cisco.com

- Firewalls are different things to different people and organizations
- All firewalls are supposed to share some common properties:
  - The firewall itself is resistant to attacks
  - The firewall is the only transit point between networks (all traffic flows through the firewall)
  - The firewall enforces the access control policy

DPS 1.0—3-1-5

As loose as the firewall concept might be, this is easily understood as there are many policies, which need to be implemented using network access controls, hence the many definitions of a firewall. Firewalls mean different things to different organizations, and each organization has unique requirements. Nevertheless, all firewalls usually share some common properties. A firewall:

- **Must be resistant to attacks:** That is, compromise of the firewall system should be very unlikely, as it would enable an attacker to disable the firewall or change its access rules
- **Must be the only transit point between networks:** In other words, all traffic between networks must flow through the firewall. This prevents a backdoor connection being used to bypass the firewall, violating the network access policy.
- **Enforces an organization’s access control policy:** This defines what the firewall permits or denies.

## Practice

Q1) What should be the three properties of all firewalls?

A) a firewall in impenetrable

B) a firewall only allows trusted users to connect through it

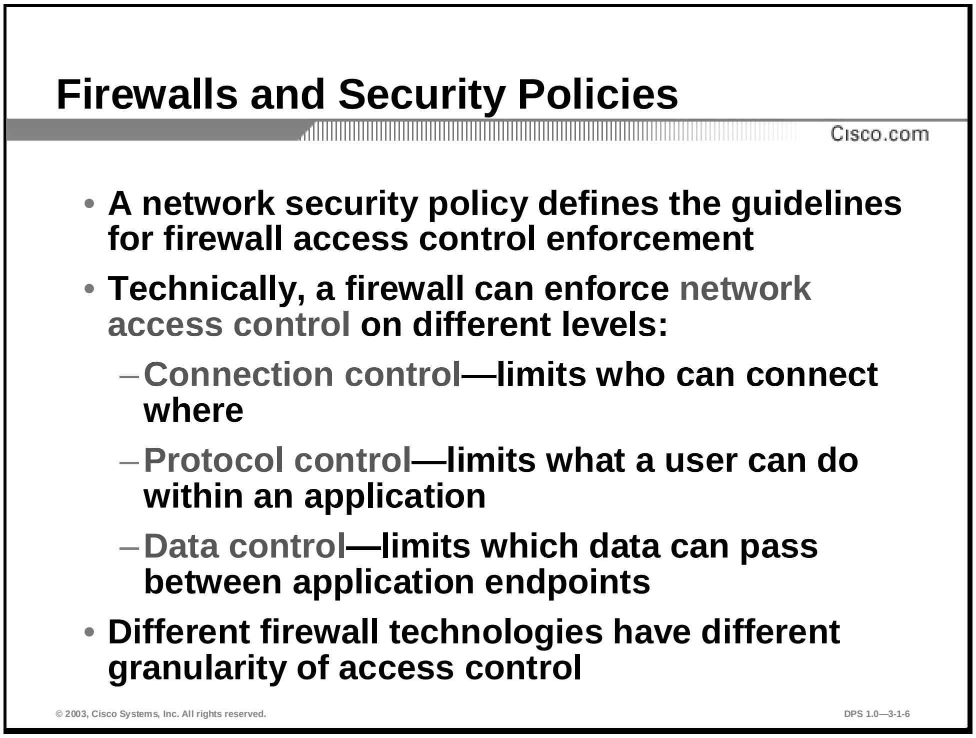
C) a firewall is the only transit point between networks

D) a firewall is resistant to penetration

E) a firewall enforces the required access control policy

F) a firewall fully protects all the hosts behind it on all layers

# Firewalls and Security Policies



## Firewalls and Security Policies

- A network security policy defines the guidelines for firewall access control enforcement
- Technically, a firewall can enforce network access control on different levels:
  - Connection control—limits who can connect where
  - Protocol control—limits what a user can do within an application
  - Data control—limits which data can pass between application endpoints
- Different firewall technologies have different granularity of access control

DPS 1.0—3-1-6

## Objective

The section will enable the learner to explain how firewalls enforce security policies.

## Introduction

This section defines the relationship between security policies, access control, and firewalls. Firewalls can provide multiple access control levels, which are described and discussed.

## Firewalls and Security Policies

A network access policy defines which network connectivity is allowed under the security policy of an organization. Under the umbrella of connectivity, many aspects of communication are covered, including:

- Network sessions between clients and servers
- Applications using the network sessions
- Data that is transported inside the application sessions

A more technical definition of a firewall can be stated as a system that enforces network access control in a network. The firewall, depending on its abilities, performs this enforcement on different levels. A firewall can perform:

- **Connection control:** Controls which application endpoints can intercommunicate. An example of which is a firewall that permits all inside users to open web connections to all web servers on the Internet.

- **Protocol control:** Controls what a user can do within an application. An example is a firewall that allows users to view web pages, but prohibits them from posting data to untrusted servers.

- **Data control:** Limits the data, passing inside the application stream. An example is a firewall that can block viruses in email messages

This granularity of filtering control depends on the technology used by the firewall system.

## Practice

Q1) When we refer to "connection control" by a firewall, we usually refer to the ability to restrict connectivity on which OSI layer?

A) Layer 3 (network layer)

B) Layer 4 (transport layer)

C) Layer 5 (session layer)

D) Layer 6 (presentation layer)

E) Layer 7 (application layer)

Q2) What is the most granular filtering available in modern firewall systems?

A) filtering of application data inside the application protocol

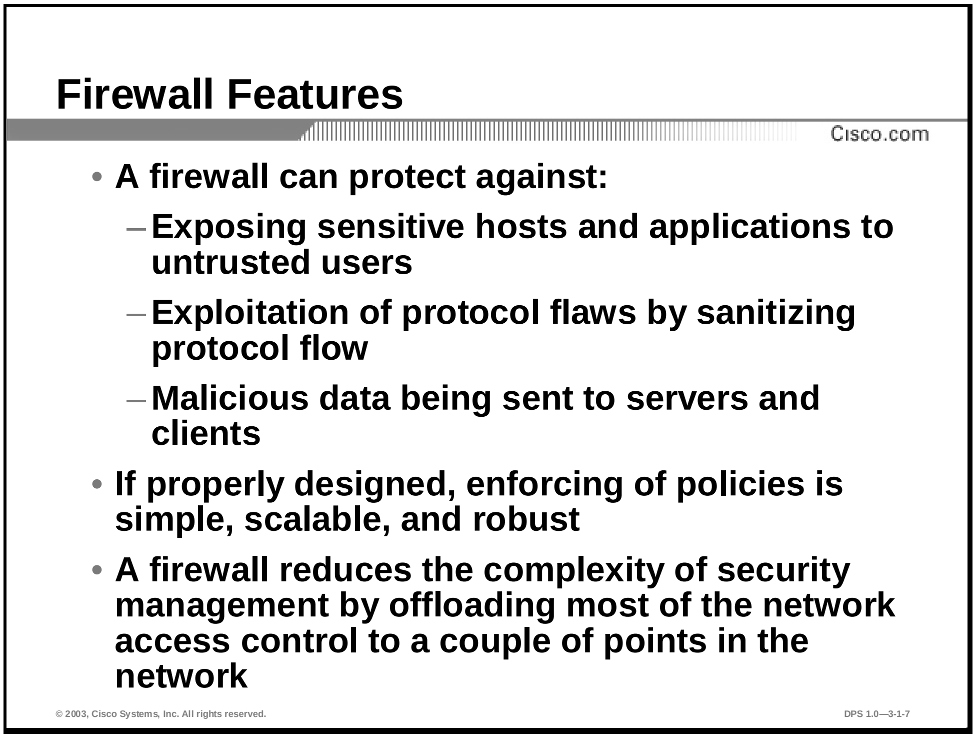
B) filtering of TCP and UDP connections (ports)

C) filtering of the application protocol

D) filtering of transport layer sessions

E) filtering on the OSI presentation layer

# Firewall Features and Limitations



## Firewall Features

- A firewall can protect against:
  - Exposing sensitive hosts and applications to untrusted users
  - Exploitation of protocol flaws by sanitizing protocol flow
  - Malicious data being sent to servers and clients
- If properly designed, enforcing of policies is simple, scalable, and robust
- A firewall reduces the complexity of security management by offloading most of the network access control to a couple of points in the network

DPS 1.0—3-1-7

## Objective

The section will enable the learner to describe general firewall features and limitations.

## Introduction

Today, firewalls are such a mainstream technology, that they are often considered a panacea for many security issues. This section attempts to clarify the features of the firewall model, and make the learner aware of the many limitations firewalls have, and how to mitigate some of the described limitations.

## Firewall Features

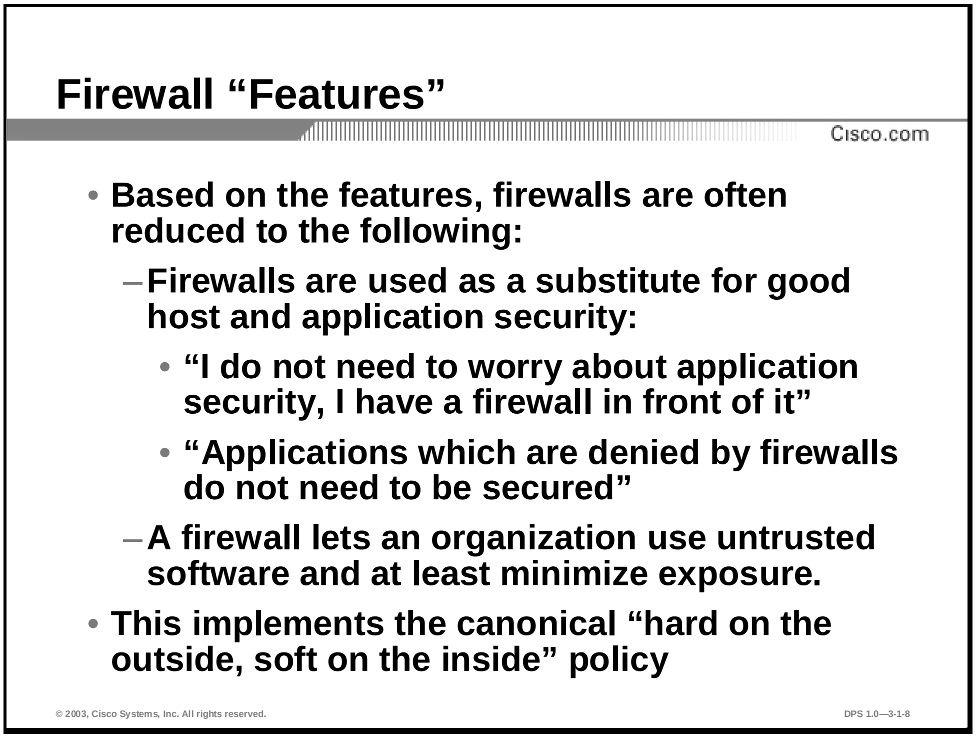
By performing network access control, a firewall can be used as a protective measure against:

- **Exposure of sensitive hosts and applications to untrusted users:** A firewall hides most of a host's functionality and only permits the minimum required connectivity to a host. Complexity is thus reduced, and many possible vulnerabilities are not exposed.

- **Exploitation of protocol flaws:** A firewall can be programmed to inspect protocol messages and verify their compliance with the protocol, be it Layer 3 (L3), Layer 4 (L4), or a higher layer application protocol. The firewall limits what attackers can send to their target, preventing the delivery of malformed packets used in an attempt either to crash, or to gain access to an application.

- **Malicious data:** A firewall can detect and block malicious data sent to clients or servers inside the application stream, thereby stopping it from infecting the server or the client.

As firewalls are located on critical interconnection points of the network, enforcing the network access policies is simple, scalable, and robust—sometimes, a small number of firewalls can handle most of an organization's network access control needs.



## Firewall “Features”

- **Based on the features, firewalls are often reduced to the following:**
  - **Firewalls are used as a substitute for good host and application security:**
    - **“I do not need to worry about application security, I have a firewall in front of it”**
    - **“Applications which are denied by firewalls do not need to be secured”**
  - **A firewall lets an organization use untrusted software and at least minimize exposure.**
- **This implements the canonical “hard on the outside, soft on the inside” policy**

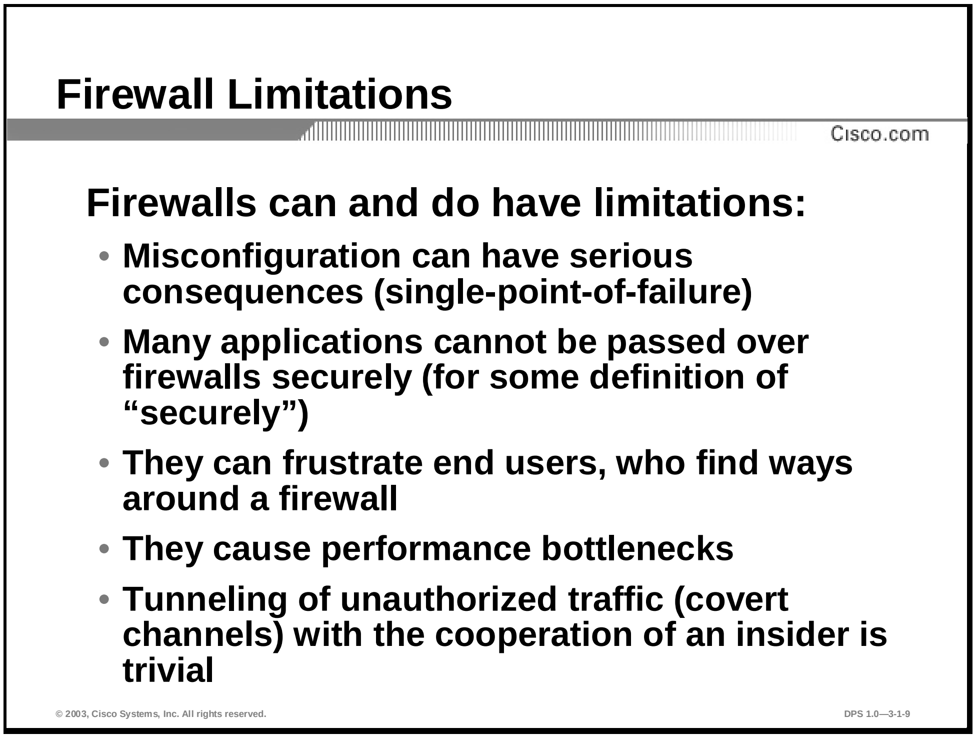
DPS 1.0—3-1-8

## Misconceptions about Firewall Functionality

Firewalls are often misunderstood, and false assumptions can be made about their capabilities. While it is true that firewalls would not be necessary if host/application security could be made extremely robust, many organizations use firewalls as a replacement for host or application security. Such an attitude is extremely dangerous, as it can completely ignore host and application security even in extreme cases, such as connecting a sensitive server inside an Internet firewall.

A cynical view of firewalls might be summarized: “a firewall lets an organization use untrusted applications and minimize their exposure to attack”. This frequent real-life scenario should be avoided at all costs. It is essential that the interdependence of application security and firewalls is understood and implemented correctly.

With respect to securing enterprise networks from the Internet, such a mindset has spawned the canonical “hard on the outside, soft on the inside” result, which is still very true in the majority of networks.



## Firewall Limitations

Firewalls can and do have limitations:

- Misconfiguration can have serious consequences (single-point-of-failure)
- Many applications cannot be passed over firewalls securely (for some definition of “securely”)
- They can frustrate end users, who find ways around a firewall
- They cause performance bottlenecks
- Tunneling of unauthorized traffic (covert channels) with the cooperation of an insider is trivial

DPS 1.0—3-1-9

# Firewall Limitations

In general, firewalls have the following limitations:

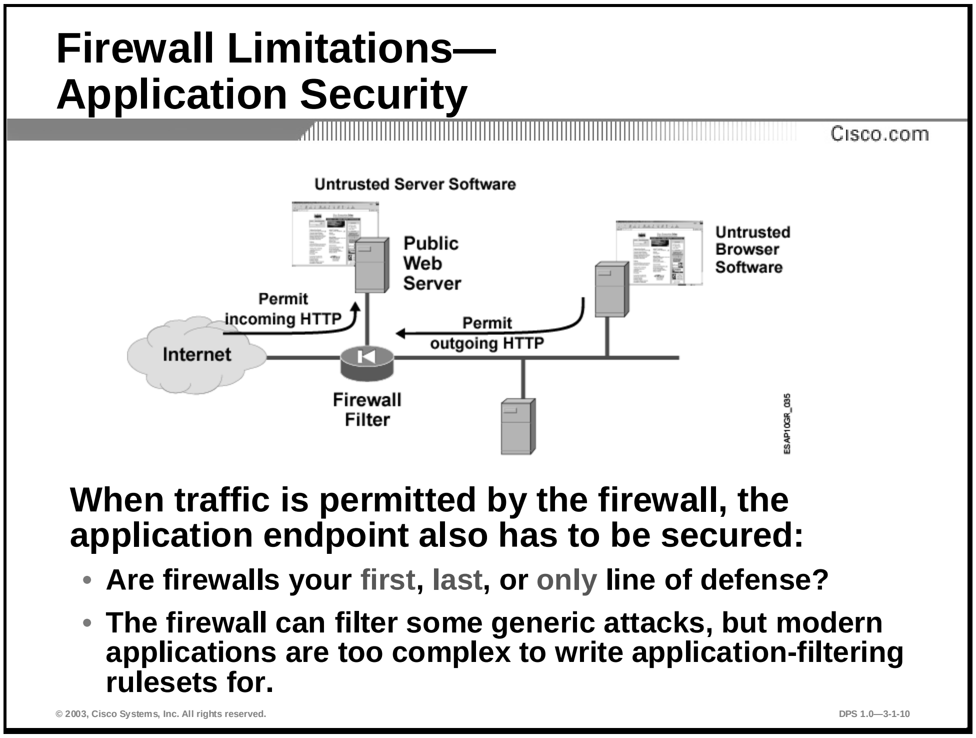
- As firewalls are used in critical points of the network, their misconfiguration can have disastrous consequences. Firewalls are often a single-point-of-failure security wise, and a single mistake in either a configuration rule or firewall code can compromise the network access policy.

- Many of the modern applications are firewall-unfriendly, as they are difficult to inspect properly. Compromises in rule design and inspection depth have to be made to support such applications, which might violate an organization’s policy.

- End-users, when faced with a restrictive firewall, might find their own methods of bypassing it. For example, inside users can dial out of the protected network to an Internet service provider (ISP), creating a backdoor connection to the protected network.

- Firewalls are placed at choke points, and can significantly impact performance if they inspect all the traffic.

- Tunneling unauthorized data over authorized connections (covert channels) is simple and generally impossible to detect. This activity usually requires the help of someone on the trusted side of the firewall.



## Firewall Limitations— Application Security

Cisco.com

**When traffic is permitted by the firewall, the application endpoint also has to be secured:**

- **Are firewalls your first, last, or only line of defense?**
- **The firewall can filter some generic attacks, but modern applications are too complex to write application-filtering rulesets for.**

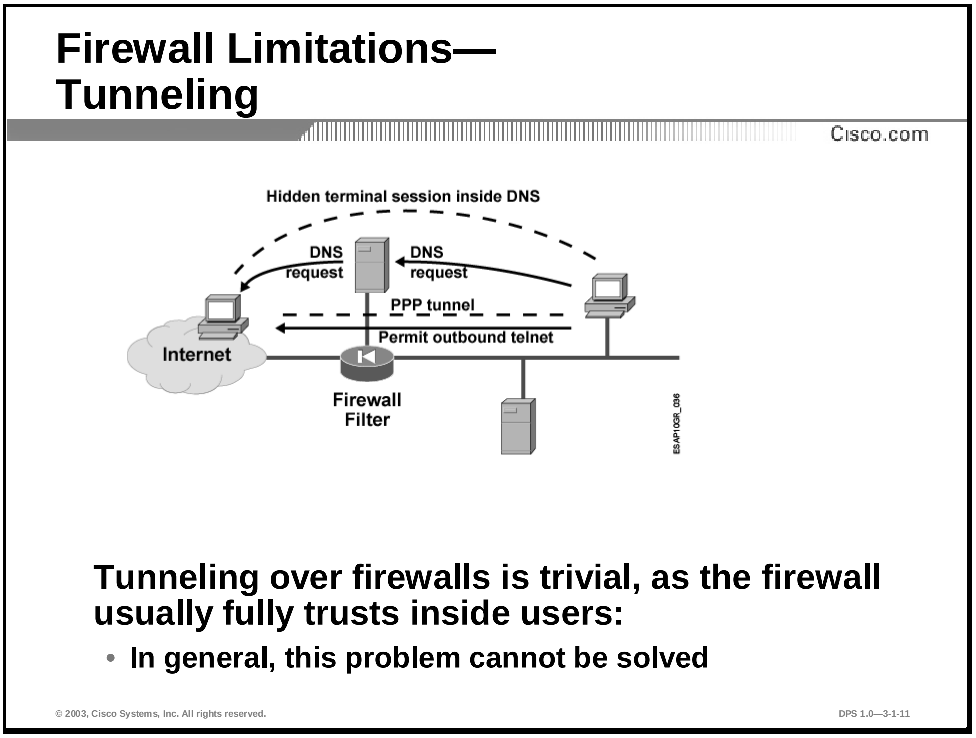
DPS 1.0—3-1-10

# Firewall Limitations in Application Security

This figure illustrates the concept of application security, when firewalls are used. A firewall can protect a vulnerable web server, but all the firewall might do is pass all web sessions to the server, and deny all other sessions. An attacker can compromise the exposed host if the permitted web sessions contain malicious data. The firewall may limit data flow on the application layer, but most firewalls on the Internet do not.

While some firewalls are able to filter traffic with fine granularity, which gives them control over all the data in an application session, configuring them to protect a custom application is practically impossible. An organization may not be able to deploy firewall rules, which would shield the application from possible threats, because modern applications are so complex, and often their internal structures are not disclosed. This impossibility forces organizations to configure firewalls with less inspection capabilities, and to provide the additional security within the application itself. This requires secure design and programming practices, which are still not widely accepted or available.

This brings up the issue of firewall adequacy. Firewalls augment network security by hiding potentially vulnerable services, permitting only the minimum allowed access (least privilege concept), and inspecting connections, protocols and applications. Firewalls however, should never be the only line of defense against a modern attacker, and their limitations must be understood. Firewalls are one of the most effective tools of network access control, and will continue to be used as networks and applications become more and more complex.



### Firewall Limitations— Tunneling

**Tunneling over firewalls is trivial, as the firewall usually fully trusts inside users:**

- **In general, this problem cannot be solved**

DPS 1.0—3-1-11

## Firewall Limitations with Tunneling

Another limitation of firewalls is their blind trust in the protected network. Firewalls are usually deployed to allow connectivity from a protected, to an untrusted network, and assume that users in the protected network are trusted. If an inside user collaborates with an outside user, they can establish a seemingly legitimate (permitted) connection over the firewall, and over this connection, tunnel unauthorized traffic.

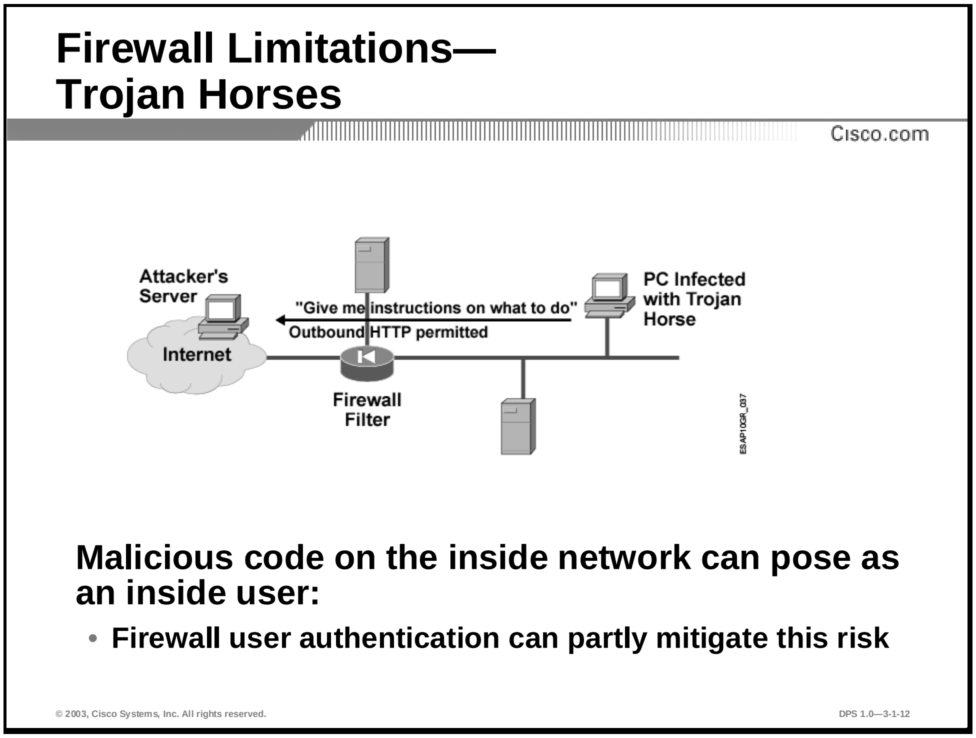
## Examples

Many firewall tunneling examples exist, and for several of them software is freely available. This figure illustrates two such possibilities of tunneling:

- Running the Point-to-Point Protocol (PPP) over an outbound telnet session. This establishes a point-to-point IP link between the perimeters (such as a leased line), while the firewall only sees a telnet session.

- Using specially crafted Domain Name System (DNS) servers and resolvers, where a terminal session is hidden in the DNS payload (for example, one of the bytes of the Pointer [PTR] record can be used to transport the terminal session).

Well-known examples of tunneling tools include GNU HTTP Tunnel and Internet Control Message Protocol (ICMP) Loki.



Firewall Limitations—
Trojan Horses

Cisco.com

Malicious code on the inside network can pose as an inside user:

- Firewall user authentication can partly mitigate this risk

DPS 1.0—3-1-12

## Firewall Limitations with Blind Trust

The firewall’s trust of the inside network can also be abused by software masquerading as a trusted inside user. An inside user can download malicious code (a “Trojan Horse”), which secretly opens connections to the untrusted network, masquerading as the user. The “Trojan Horse” then accepts and executes instructions, performing malicious actions on the user’s system.

Firewall authentication of users might reduce this risk somewhat, or at least reduce the window of exploitation.

## Practice

Q1) Which are some limitations of firewalls? (Choose two.)

A) tunneling over permitted connections

B) inability to filter on arbitrary application data for any application

C) all firewalls always impact performance

D) there is no user authentication

E) protocol control is unreliable

Q2) What does “covert channel” tunneling through firewalls require? (Choose one)

A) specially crafted tools, as no tools are available on the Internet

B) cooperation of an insider (or a compromised inside system)

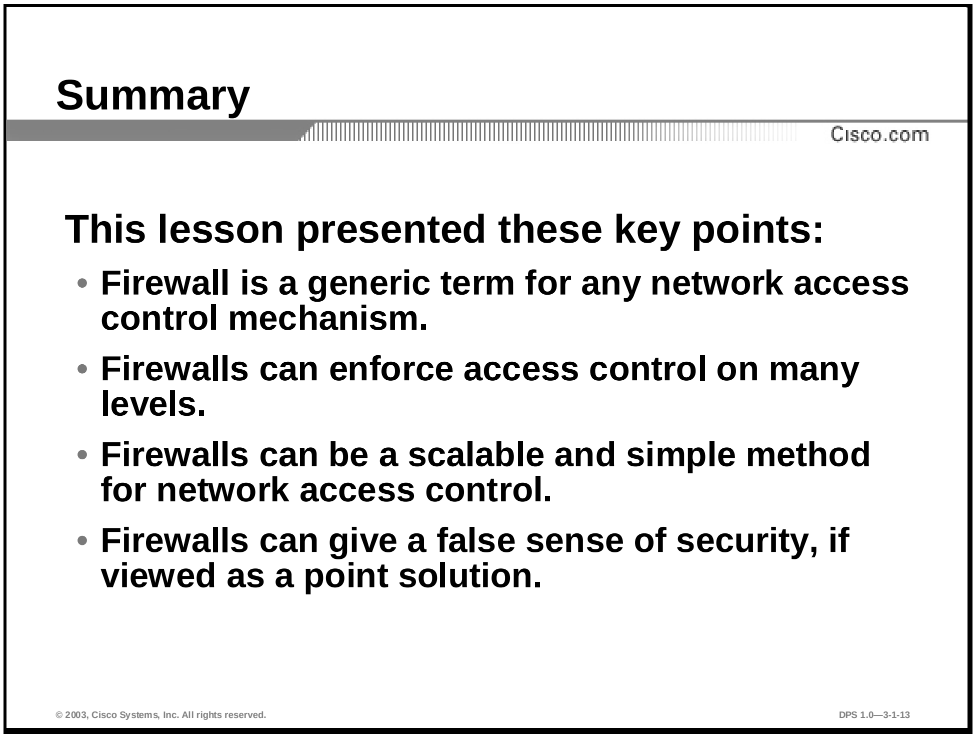
C) permitted inbound connections in the firewall rules

D) permitted IPSec tunneling on the firewall

E) permitted GRE tunneling on the firewall

# Summary

This section summarizes the key points discussed in this lesson.



## Summary

Cisco.com

This lesson presented these key points:

- Firewall is a generic term for any network access control mechanism.
- Firewalls can enforce access control on many levels.
- Firewalls can be a scalable and simple method for network access control.
- Firewalls can give a false sense of security, if viewed as a point solution.

DPS 1.0—3-1-13

## Next Steps

After completing this lesson, go to:

- Firewall Technologies lesson

# Quiz: Firewall Function

Complete the quiz to assess what you have learned in this lesson.

## Objectives

This quiz test your knowledge on how to:

- Explain the function of a firewall
- Identify its features and limitations

## Instructions

Answer these questions:

1. What is the definition of a firewall?
2. What are the common properties of all firewalls?
3. List some firewall features.
4. List some firewall limitations.

## Scoring

You have successfully completed the quiz for this lesson when you earn a score of 80 percent or better.

# Firewall Technologies

## Overview

This lesson introduces various firewall technologies, which are used to pass data between networks with different levels of trust. The features and limitations of each technology are addressed, as well as their suitability for use to support various customer requirements and applications.

### Importance

The need to understand firewall technologies is paramount when different requirements of performance and filtering granularity are presented to a designer.

### Lesson Objective

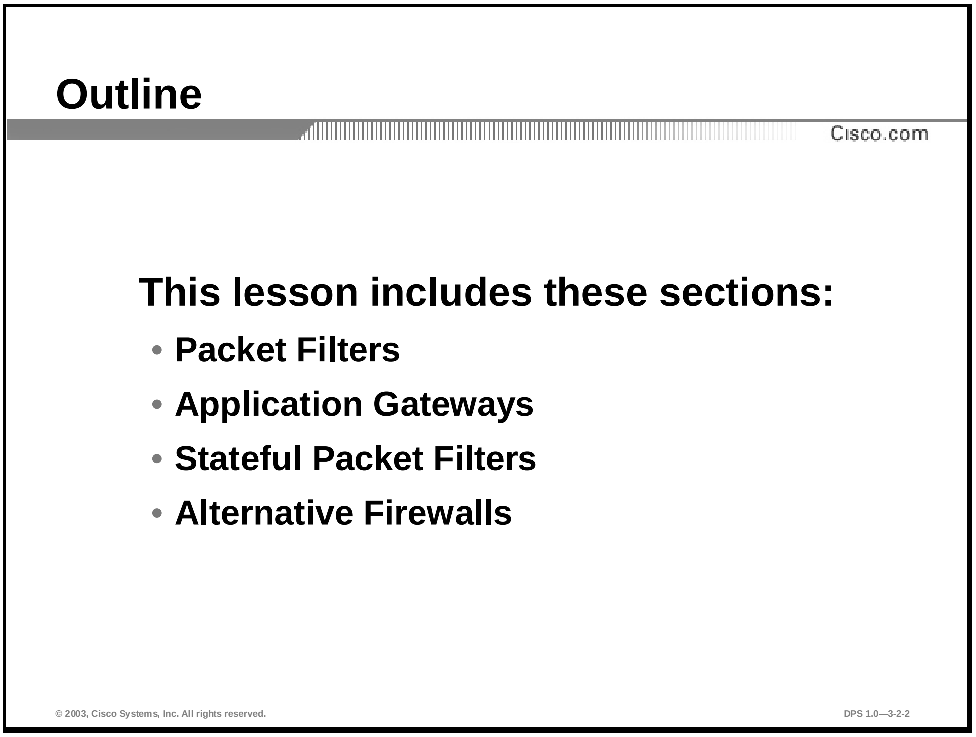
The lesson will enable the learner to compare several common firewall technologies with respect to access control and identify their features and limitations.

## Learner Skills and Knowledge

To fully benefit from this lesson, you must have these prerequisite skills and knowledge:

- Understand the firewall function
- Describe common access control methods in computer security

## Outline



Outline

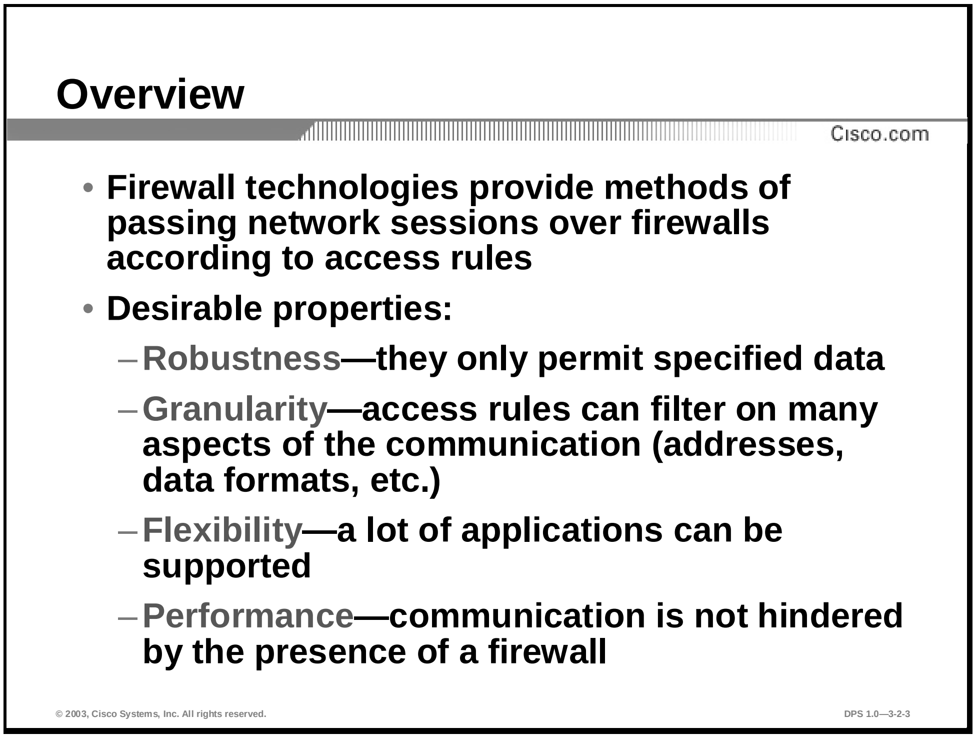
Cisco.com

This lesson includes these sections:

- Packet Filters
- Application Gateways
- Stateful Packet Filters
- Alternative Firewalls

© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—3-2-2

# Overview



## Overview

Cisco.com

- Firewall technologies provide methods of passing network sessions over firewalls according to access rules
- Desirable properties:
  - Robustness—they only permit specified data
  - Granularity—access rules can filter on many aspects of the communication (addresses, data formats, etc.)
  - Flexibility—a lot of applications can be supported
  - Performance—communication is not hindered by the presence of a firewall

© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—3-2-3

This lesson describes mainstream firewall technologies, which enable secure communication over firewalls. Secure communication describes the ability of the firewall to pass a network session between two network endpoints according to the defined access policy.

Firewall technologies differ in their ability to perform network access control by filtering data passing through it. This ability can be described from various aspects, such as:

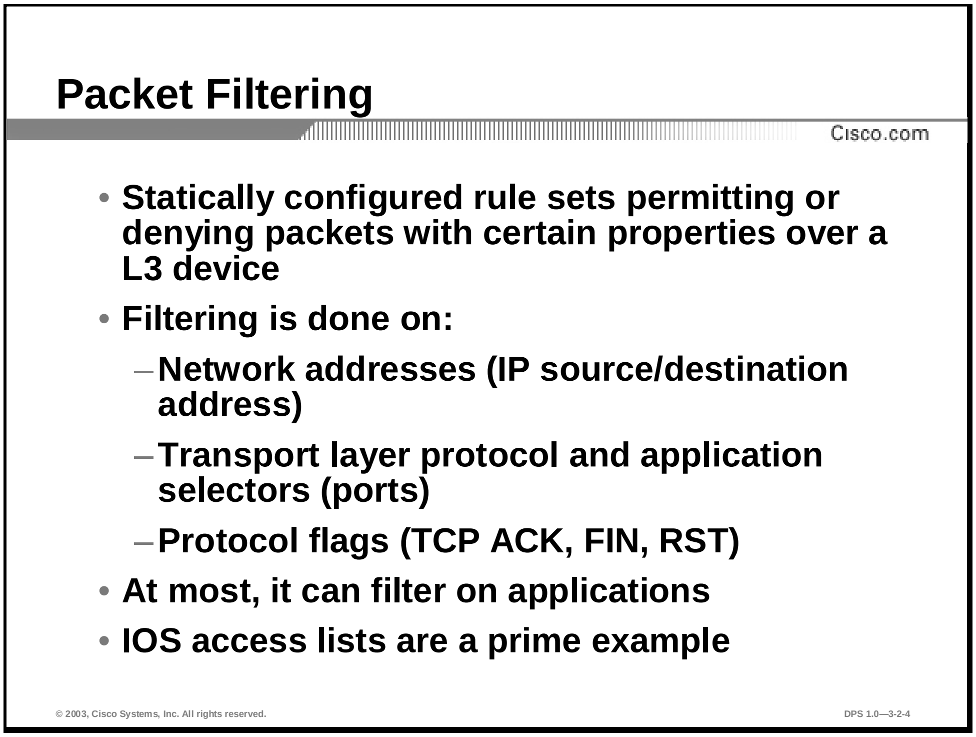
- **Robustness of filtering:** A robust filtering technology only permits the specified data over a firewall. A less robust technology might also leak other information through/past the firewall. As an example, a simple packet filter designed to only permit HTTP sessions actually permits all traffic that looks like HTTP. It passes spoofed packets with the proper port numbers through the firewall although they do not belong to a valid HTTP session.

- **Granularity of filtering:** A granular technology is able to filter on many aspects of the communication—endpoint addresses, transport protocol, application selectors (ports), application protocol commands, and application data itself. An example of a granular mechanism is an application gateway, which can look into the application protocol and attempt to eliminate suspicious data.

- **Flexibility of filtering:** A flexible technology is able to support many applications in many communication scenarios. A good example of a flexible technology is stateful packet filtering, which can be made application aware with reasonable effort. An example of an inflexible technology is an application-layer gateway, as they only filter a specific application protocol, and development of custom gateways is complex.

- **Performance of filtering:** A high-performance filtering solution provides flexibility in many scenarios that require high-performance connectivity.

# Packet Filters



## Packet Filtering

Cisco.com

- Statically configured rule sets permitting or denying packets with certain properties over a L3 device
- Filtering is done on:
  - Network addresses (IP source/destination address)
  - Transport layer protocol and application selectors (ports)
  - Protocol flags (TCP ACK, FIN, RST)
- At most, it can filter on applications
- IOS access lists are a prime example

© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—3-2-4

## Objective

The section will enable the learner to explain the security properties, features, and limitations of packet filtering used as an access control method, and identify their features and limitations.

## Introduction

Packet filtering is one of the earliest, and still widely used, firewalling technologies. This section describes the operation, features, and limitations of packet filtering technology when used as a firewall building block.

## Packet Filtering Technology

Packet filtering is usually employed by a Layer 3 (L3) device to permit or deny specific packets from being routed across it. Packet filters use statically defined sets of rules (rulesets, access lists), to define which traffic is to be permitted or denied.

## Packet Filtering Granularity

Packet filtering only looks at protocol headers up to the transport layer. The header fields used in access rules to match packets are:

- **Network-layer addresses:** For example, the IP source and destination address of the communication.

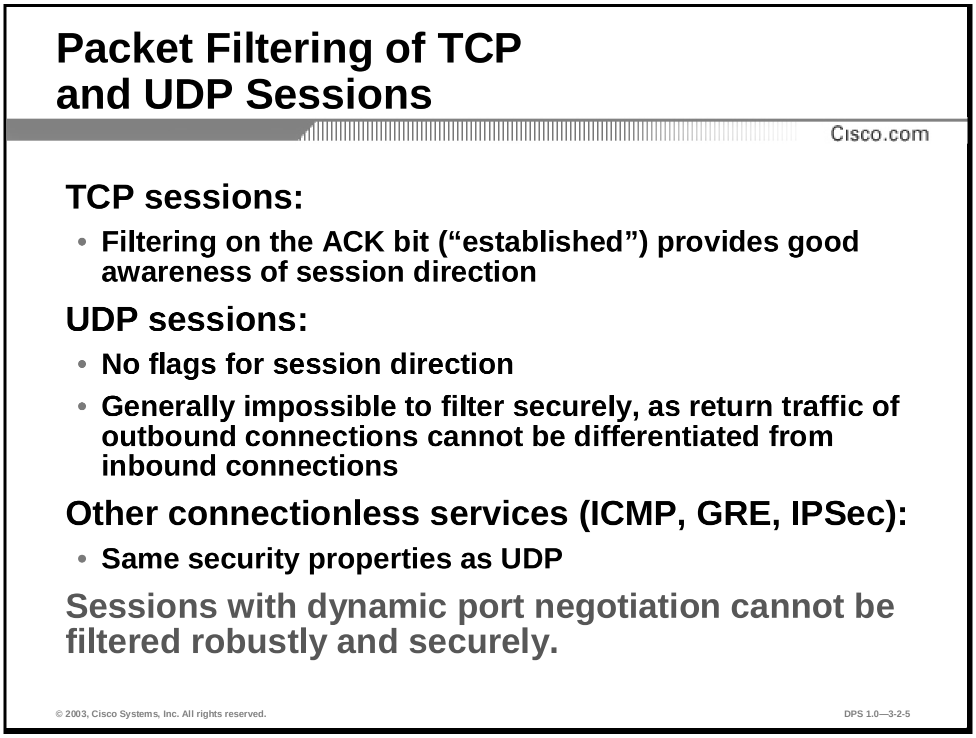
- **Transport-layer protocol and application selectors (ports):** To permit only a specific protocol or application between endpoints.

- **Flags inside the transport protocol:** These define specific per-connection properties, such as the connection direction. With the TCP protocol the ACK, FIN, RST, and other flags could be matched by the packet filter.

The most packet filtering can do is filter applications by either permitting or denying a specific application running between two network endpoints (based on port numbers, etc). Anything inside the application protocol is contained within the packet payloads, which are normally not inspected by packet filtering.

## Example

Cisco IOS extended Access Control Lists (ACLs) are a good example of a state-of-the-art packet filtering language. An extended ACL can filter on network addresses, protocols, ports, and specific pre-protocol flags, such as TCP flags or Internet Control Message Protocol (ICMP) types and codes.



**Packet Filtering of TCP and UDP Sessions**

TCP sessions:

- Filtering on the ACK bit (“established”) provides good awareness of session direction

UDP sessions:

- No flags for session direction
- Generally impossible to filter securely, as return traffic of outbound connections cannot be differentiated from inbound connections

Other connectionless services (ICMP, GRE, IPSec):

- Same security properties as UDP

Sessions with dynamic port negotiation cannot be filtered robustly and securely.

DPS 1.0—3-2-5

## Packet Filtering of TCP and UDP Sessions

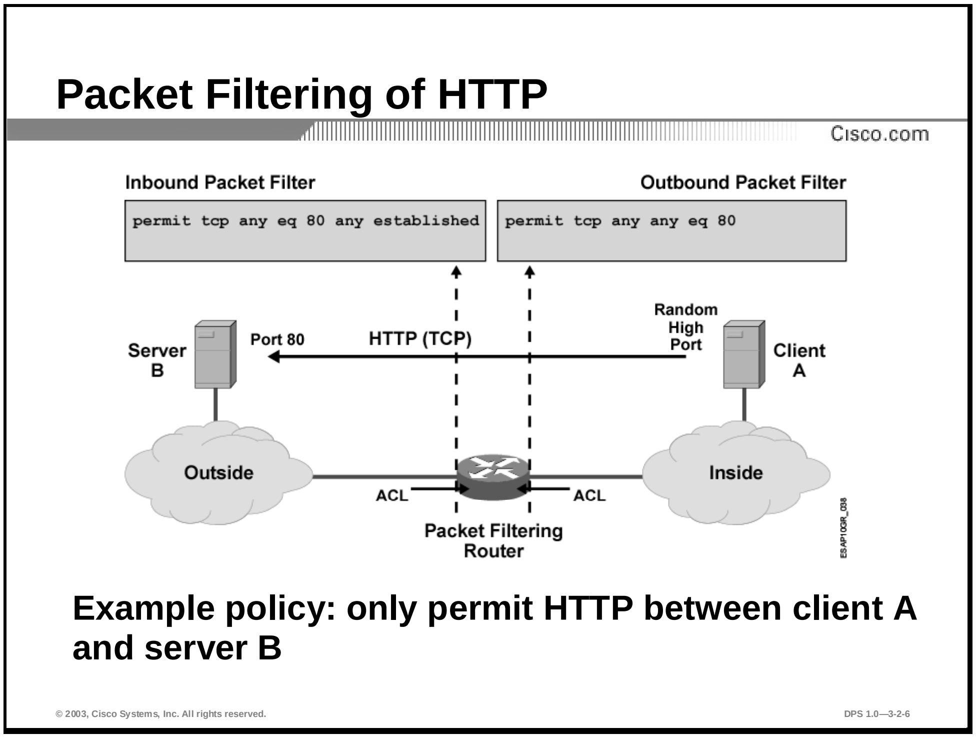
Packet filtering handles different protocols in different ways. It is desirable to have as much information in the protocol header as possible, as more specific rules can be applied to more accurately describe an application being filtered.

In TCP sessions, every TCP segment carries a great deal of information that can be used to describe an application. Besides containing flow information (IP addresses, ports), each TCP segment contains sequence numbers, not used by packet filters, as they do not keep track of them, and special TCP flags. A packet filter can use those flags to determine the direction of a session:

- If a TCP segment has the SYN flag set, but no ACK flag, this is the initiating (first) packet of a connection. The packet filtering firewall can use this information to either permit the packet (allow the application to start) or deny it.

- If a TCP segment has the ACK flag set, the segment belongs to an already established TCP flow. The packet is usually permitted, as the packet filter is already configured/defined to either permit or deny the SYN packet originally.

The second bullet illustrates the “intelligence” of packet filters. A packet filter cannot “know” that an ACK segment actually belongs to an established session. An attacker could be sending spoofed ACK segments in an effort to elicit responses from inside hosts and map the network. The firewall operator, who understands that TCP connections begin with a SYN segment, configures the packet filtering intelligence into the ruleset, and the packet filter blindly follows the defined static rules.



## Packet Filtering of HTTP

Cisco.com

**Inbound Packet Filter**

```
permit tcp any eq 80 any established
```

**Outbound Packet Filter**

```
permit tcp any any eq 80
```

**Example policy: only permit HTTP between client A and server B**

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—3-2-6

# Packet Filtering of HTTP

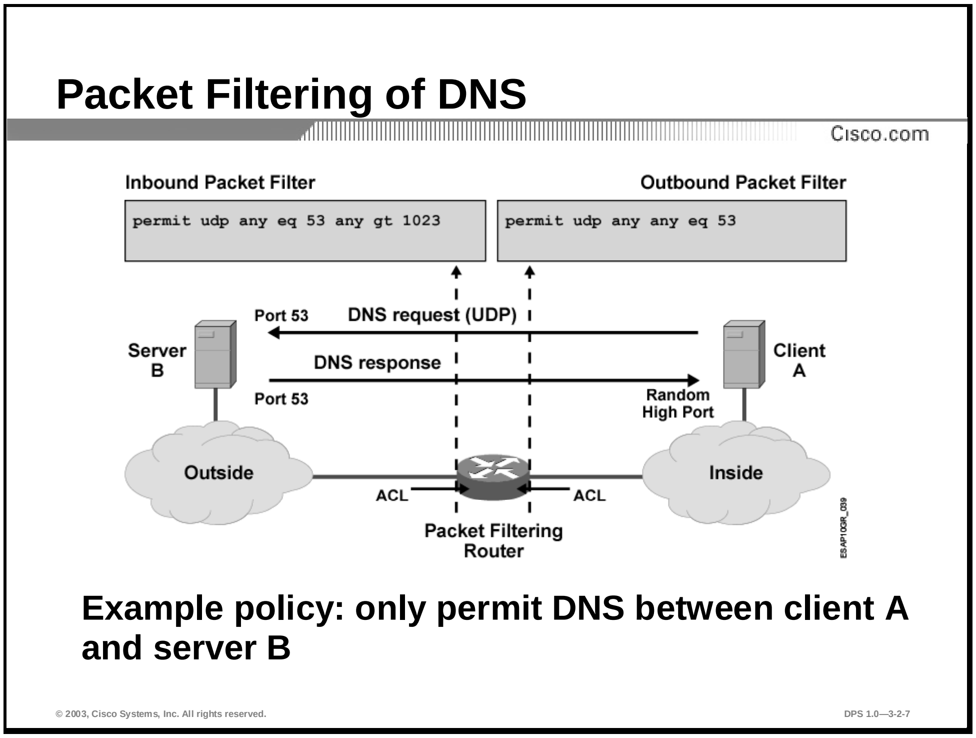
This example illustrates two Cisco IOS access lists programmed to only to allow an HTTP session between a client and a server. The access lists permit TCP segments with a server port of 80 and a random client port. The designer of the firewall should always be aware of the traffic flow in both directions and configure the two access rules:

- **Permit traffic from the client to the server:** The outbound access list permits traffic from the client to the server, and from any high (>1023) client port to the server port of 80. All flags are allowed, including the SYN flag, which will permit the establishment of the session.

- **Permit traffic from the server to the client:** The access list permits traffic from the server to the client, from the server port of 80, and to any client high port. Only segments with the ACK/RST/FIN flags are allowed (the “established” keyword) to prevent from any sessions establishing **from the server port to a high client port—**in effect, only return traffic of the HTTP session is permitted.

Most TCP filtering applications are able to use these two access rules because they are sufficiently secure.

However, an attacker on the outside network can send any packets to the inside client with an arbitrary (possibly spoofed) address of the server, from port 80, to any client high port, as long as they have the ACK/RST/FIN flags set. Malformed packets may, for example, trigger a bug and perhaps crash the client.



## Packet Filtering of DNS

Cisco.com

Inbound Packet Filter

```
permit udp any eq 53 any gt 1023
```

Outbound Packet Filter

```
permit udp any any eq 53
```

Port 53

DNS request (UDP)

Server B

DNS response

Client A

Port 53

Random High Port

Outside

Inside

ACL

ACL

Packet Filtering Router

ESAP10GR_039

**Example policy: only permit DNS between client A and server B**

DPS 1.0—3-2-7

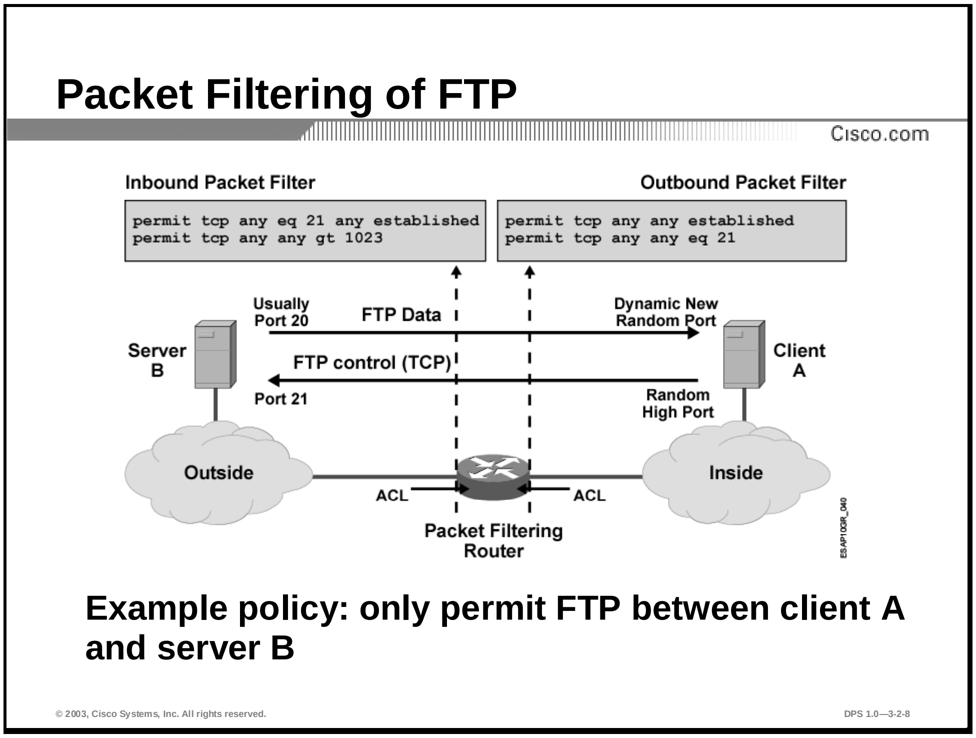
# Packet Filtering of DNS

This example illustrates two Cisco IOS access lists programmed to only to allow a Domain Name System (DNS) session between a client and a server. The access lists permits User Datagram Protocol (UDP) segments with a server port of 53 and a random client port. The designer of the firewall should always be aware of the traffic flow in both directions and configure two access rules:

- **Permit traffic from the client to the server:** The access list permits traffic from the client to the server, and from any high (>1023) client port to the server port of 53. There are no flags in UDP, therefore this access list is not “direction aware”.

- **Permit traffic from the server to the client:** The access list permits traffic from the server to the client, from the server port of 53, and to any client high port. As there are no flags in UDP (no equivalent of “established”), the rules are again not direction aware. An attacker spoofing the address of the server, or perhaps breaking into the server can INITIATE UDP sessions to any application on the client, which listens on high ports (for example, NFS). The only restriction is that the outside attacker needs to use a source port of 53, which is trivial to set in UDP packets.

This setup obviously has security issues and is not as robust as TCP filtering. Generally, packet filters do not filter UDP flows well as return traffic to the client application always requires opening all high ports towards the client. This exposes much more than the application, which should be permitted. Some UDP applications can specify a range of client ports they use, which might limit exposure. However, such applications are rare and do not conform to the classic Internet client implementation recommendations.



Packet Filtering of FTP

Cisco.com

Inbound Packet Filter

```
permit tcp any eq 21 any established
permit tcp any any gt 1023
```

Outbound Packet Filter

```
permit tcp any any established
permit tcp any any eq 21
```

Example policy: only permit FTP between client A and server B

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—3-2-8

## Packet Filtering of FTP

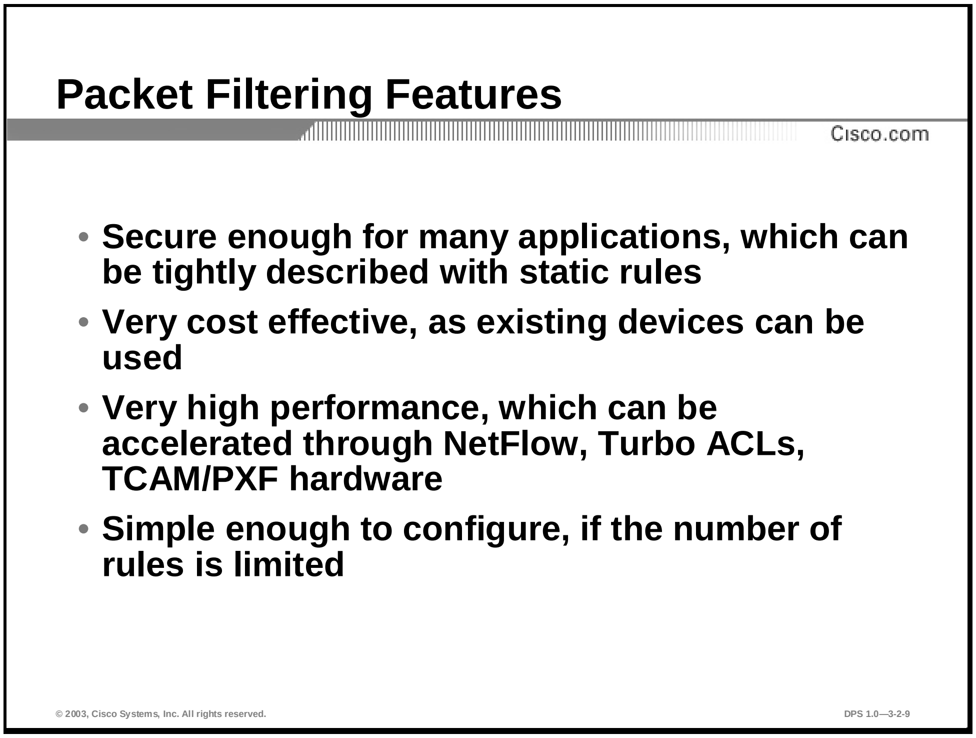
This example shows two Cisco IOS access lists programmed to only to allow a File Transfer Protocol (FTP) session between a client and a server. FTP is a protocol using dynamically negotiated ports for its data sessions; therefore the designer of an access list, the access list being a static ruleset, cannot adequately describe FTP using static rules. The access lists in the example permits TCP segments between the server's port of 21, and any high client port, to permit the FTP's control connection. The access lists must permit connections initiated from the server to the client, from the server's port 20 and to a high client port, which are the dynamic data connections FTP requires by design. The designer of the firewall needs to be aware of the traffic flow in both directions and of all dynamic sessions, and configure two access rules:

- **Permit traffic from the client to the server (on the outside interface):** The access list permits traffic from the client to the server, from any high (>1023) client port, and to the server port of 21. All flags are allowed, including the SYN flag that will permit the establishment of the session. This rule also must permit return traffic for the data sessions, which are initiated by the server—packets from any high client port to the server port of 20, carrying the ACK bit.

- **Permit traffic from the server to the client (on the inside interface):** The access list permits traffic from the server to the client, from the server port of 21, and to any client high port. Only segments with the ACK/RST/FIN flags are allowed (the "established" keyword) for the control session. The server may also establish TCP connections towards the client (the FTP data connections). Therefore, all TCP segments are allowed from the server’s port of 20 to any high client port. This is a tremendous security risk, as it allows the server to access any application running on the client, listening on a high port.

**Note** In general, FTP, in active mode, can use any server port for the data session. Port 20 is usually used, but not required by the RFC.

Dynamic applications, which are common in modern IP networks, marked the demise of packet filters as a robust filtering mechanism for most sessions. Compromises, such as opening huge holes in firewalls, have to be made to transport dynamic applications over firewalls. Developed to address these issues, application gateways and stateful packet filters are application aware and understand the behavior of dynamic protocols.



## Packet Filtering Features

- Secure enough for many applications, which can be tightly described with static rules
- Very cost effective, as existing devices can be used
- Very high performance, which can be accelerated through NetFlow, Turbo ACLs, TCAM/PXF hardware
- Simple enough to configure, if the number of rules is limited

DPS 1.0—3-2-9

# Packet Filtering Features

Packet filters are robust enough for a number of applications that do not require tracking of applications and tight filtering of every application packet. Such applications include firewalls where:

- Only static TCP protocols pass between perimeters.
- Access control is only done based on network layer addresses or transport layer protocols. A good example of this is ingress/egress filtering used by most service providers (SPs) and enterprises on the edge of their network.

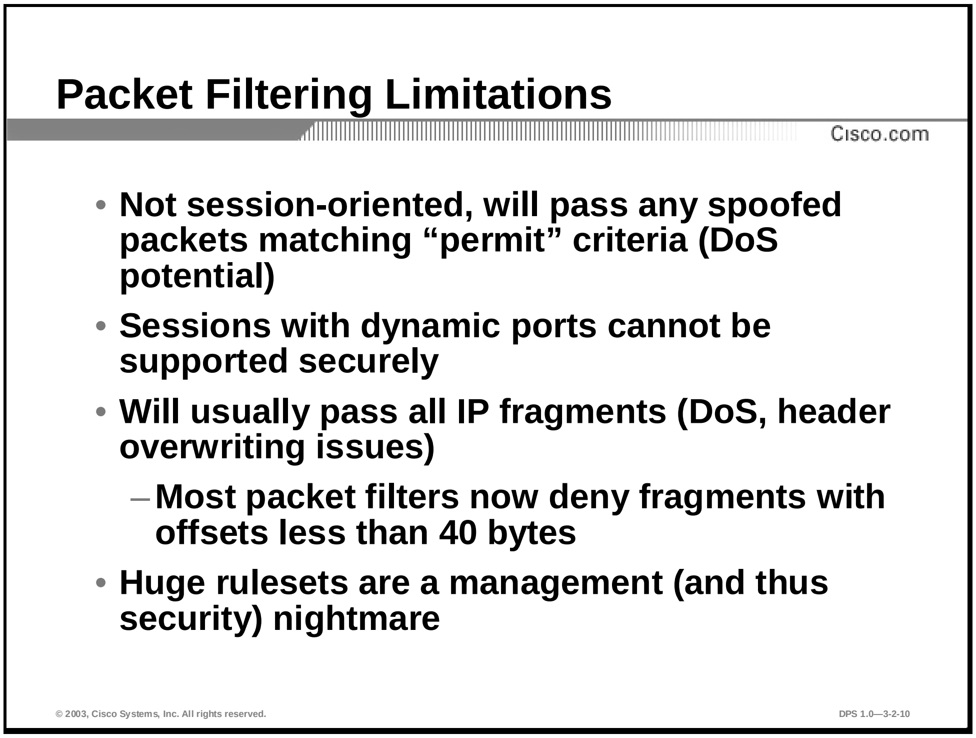
Packet filters are also very cost effective to deploy, as they are generally present in existing network software and do not require software changes.

Packet filtering firewalls easily achieves high performance. Cisco software uses a variety of techniques to speed up processing of packet filtering:

- **Manual optimization of rules:** The operator uses his/her knowledge of traffic patterns to optimize the order or first-match rules in the ruleset, which decreases per-packet search time when an ACL is evaluated
- **Turbo ACLs:** Build an evaluation tree of the rule list to reduce the number of lookups in comparison to the linear evaluation of a normal access list.
- **NetFlow switching:** Can speed up ACL evaluation by processing only the first packet of a flow, either permitting or denying it, and then apply the same decision on the rest of the flow's packets.

- **Hardware implementation of access lists:** Either using network processors (PXF) or constant-speed rule matching memory, Ternary Content Addressable Memory (TCAM), to incur little or no performance penalty even for huge access lists.

Packet filtering rules can also be simple to configure if the number of applications and endpoints is limited, and if the designer is proficient in the rule language and knowledge of protocols.



## Packet Filtering Limitations

- Not session-oriented, will pass any spoofed packets matching “permit” criteria (DoS potential)
- Sessions with dynamic ports cannot be supported securely
- Will usually pass all IP fragments (DoS, header overwriting issues)
  - Most packet filters now deny fragments with offsets less than 40 bytes
- Huge rulesets are a management (and thus security) nightmare

DPS 1.0—3-2-10

# Packet Filtering Limitations

Packet filtering has serious limitations, and therefore warrants the use of more advanced methods in most access control scenarios:

- **Packet filters are not session oriented:** They rely on the ability of the designer to set up the rules according to his knowledge of protocols. An attacker can still send arbitrary packets through the filter, even though those packets do not belong to a valid session. Such packets can, for example, trigger a software bug on a target system and make it unavailable. An example of this is packets that are too big or have malformed header fields.

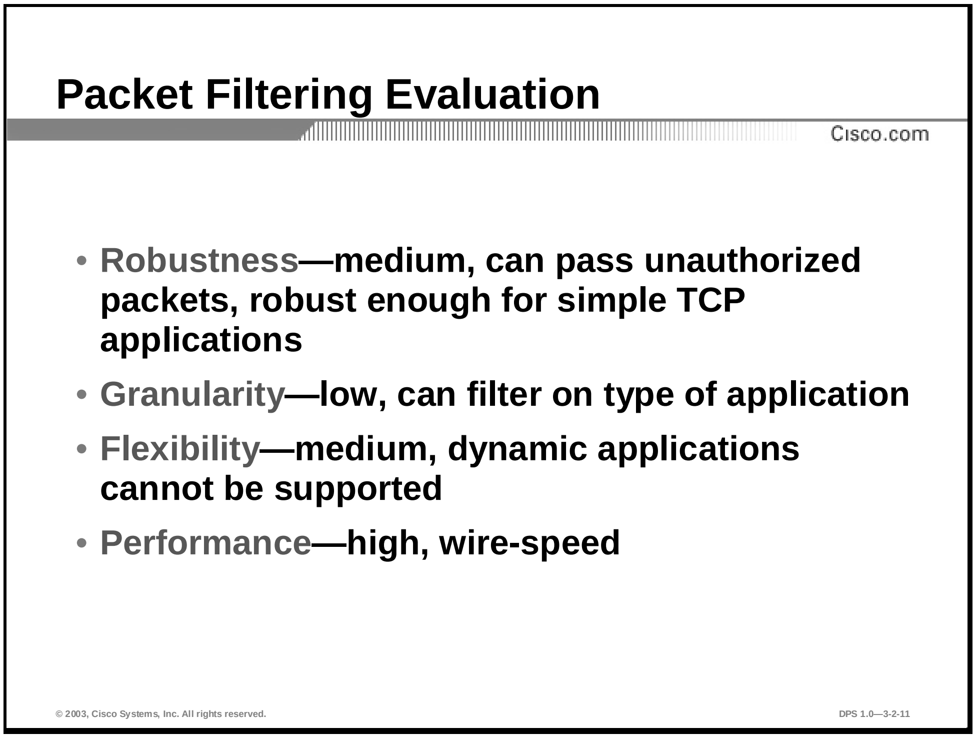
- **It is difficult to securely filter sessions with dynamic port negotiations:** To permit other (unauthorized) traffic, very open access rules are required.

- **Packet filters do not handle IP fragments strictly:** If a packet filter filters on TCP header information, fragmented IP packets only carry the TCP header in the first fragment. Packet filters pass all non-first fragments unconditionally, relying on the filtering of the first fragment to enforce a policy. This can open an inside host to denial-of-service (DoS) attacks, or header overwriting attacks. As the designer needs to be aware of bidirectional traffic flow and any additional sessions opened by applications, the rules can become complex and unmanageable. With the drive for simplicity to guarantee correctness, packet filtering rules can become too complex to be trusted.

## Example

A known attack against packet filters was the header overwriting method, where the attacker sent the TCP segment as two IP fragments. The following sequence of events takes place at the packet filter:

1. The packet filter generally permitted the first fragment, which had the TCP header, where the ACK bit was set.
2. The packet filter unconditionally passed the second fragment, as it assumed it did not contain TCP header (port) information. The second fragment in fact had a very low fragmentation offset of 20.
3. The end host interpreted the second fragment's low offset as overlapping the first fragment and overwriting the first fragments TCP header.
4. By setting the beginning of the second fragment's payload precisely, the first 20 bytes of payload would overwrite the first packet's TCP header, setting the SYN bit and clearing the ACK bit, effectively creating a new unauthorized connection to the inside host.



## Packet Filtering Evaluation

- **Robustness—medium, can pass unauthorized packets, robust enough for simple TCP applications**
- **Granularity—low, can filter on type of application**
- **Flexibility—medium, dynamic applications cannot be supported**
- **Performance—high, wire-speed**

DPS 1.0—3-2-11

# Packet Filtering Evaluation

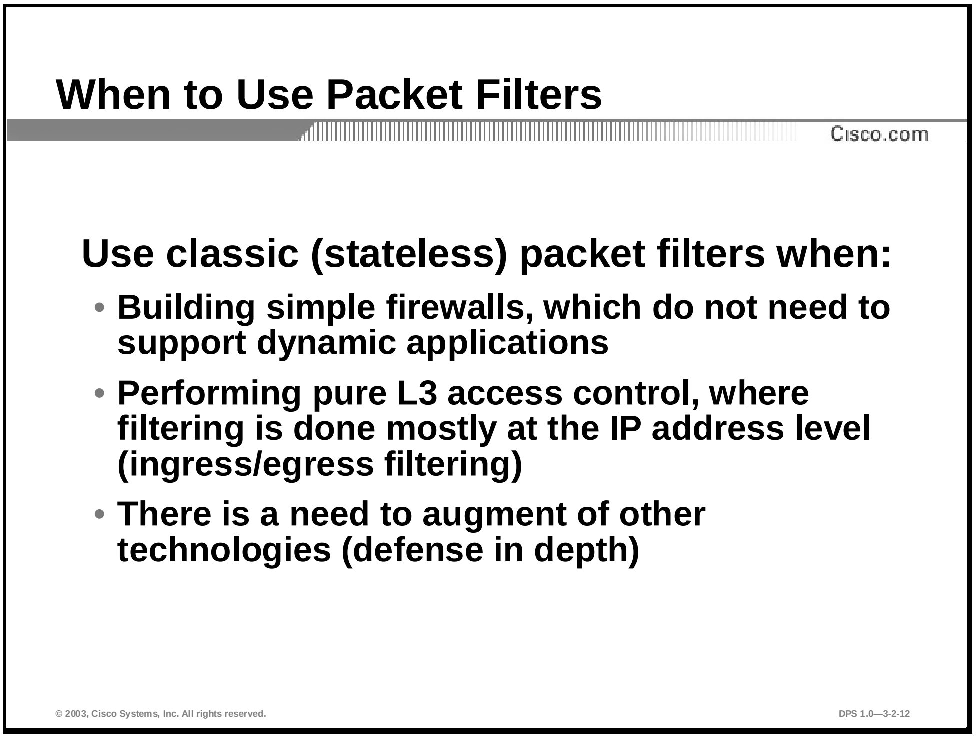
Packet filters can be evaluated against criteria for firewall technology including:

- **Robustness of filtering:** Medium. Packet filters are only robust enough for L3 (IP) filtering, and Layer 4 (L4) filtering of simple, single-channel TCP sessions. Any other sessions require the rules to allow more traffic than necessary, and expose hosts to additional risks.

- **Granularity of filtering:** Low. Packet filters can filter traffic based on L3 addresses, L4 protocol, and application selectors (ports). Therefore, the most they can do is filter based on the type of application running between endpoints.

- **Flexibility of filtering:** Low. Packet filters' flexibility is low, as UDP applications and modern dynamic applications cannot be filtered securely.

- **Performance:** High. Packet filter performance is very high, making them ideally suited for high-throughput/low-latency requirements.



When to Use Packet Filters

Use classic (stateless) packet filters when:

- Building simple firewalls, which do not need to support dynamic applications
- Performing pure L3 access control, where filtering is done mostly at the IP address level (ingress/egress filtering)
- There is a need to augment of other technologies (defense in depth)

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—3-2-12

## Packet Filtering Deployment Guidelines

Packet filters are still deployed in a variety of scenarios:

- When simple firewalls are built, and there is no need to support dynamic TCP or UDP applications.
- When filtering is performed strictly on L3. An example is classic ingress/egress filtering at the network edge, where anti-spoofing rules are often installed.
- When another technology needs to be augmented for defense-in-depth. For example, an application-layer gateway (ALG)-based firewall, is protected using a packet filtering router, which limits connectivity to the ALG host.

## Practice

Q1) What is the security issue in classic packet filtering of active FTP sessions?

A) the control session cannot be adequately filtered

B) allowing data sessions to the client opens up all the high ports on the client

C) performance of data transfer is low

D) allowing control sessions to the client opens up all the high ports on the client

E) the “established” keyword cannot be used for control or data sessions

Q2) How do packet filters handle IP fragments, when filtering on Layer 4 ports?

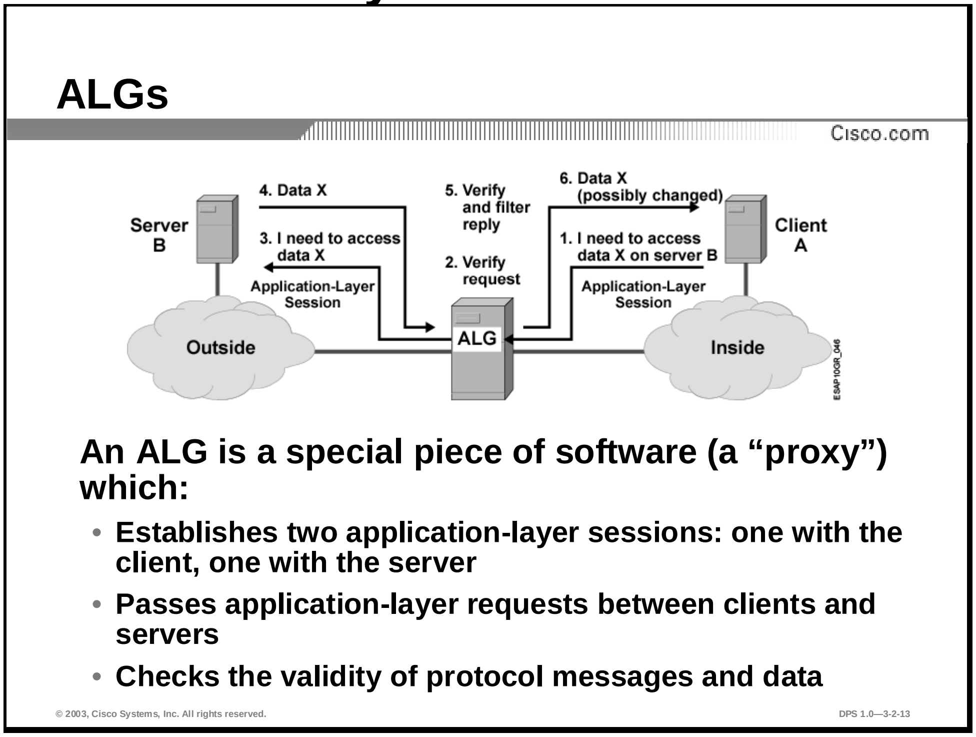
A) all fragments are buffered on the router, and then reassembled

B) all fragments are permitted by default after some basic fragmentation offset checks

C) all fragments are denied by default

D) all fragments are inspected to ensure they belong to the same IP packet

# Application Gateways


Cisco.com
Server B
4. Data X
3. I need to access data X
Application-Layer Session
Outside
5. Verify and filter reply
2. Verify request
ALG
6. Data X (possibly changed)
1. I need to access data X on server B
Application-Layer Session
Client A
Inside
ESAP10GR_046

## ALGs

**An ALG is a special piece of software (a "proxy") which:**

- **Establishes two application-layer sessions: one with the client, one with the server**
- **Passes application-layer requests between clients and servers**
- **Checks the validity of protocol messages and data**

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—3-2-13

## Objective

The section will enable the learner to explain the security properties, features, and limitations of application gateways used as an access control method, and identify their features and limitations.

## Introduction

Application-layer gateways are a legacy technology, which is still used in organizations, which require the most granularity of filtering in their firewall systems. This section describes the operation, features, and limitations of application-layer gateway technology when used as a firewall building block.

## ALG Technology

An ALG is a special piece of software designed to relay application-layer requests and responses between endpoints. An ALG acts as an intermediary between an application client and a server, acting as a virtual server to the client, and as a virtual client to the real server.

## ALG Operation

When using an ALG to pass application-layer traffic:

**Step 1** The client connects to the ALG and submits an application-layer request, indicating the true destination of the request, and the request data itself.

**Step 2** The ALG analyzes the request and may filter or change its contents, and then opens a session to the destination server, posing as the client.

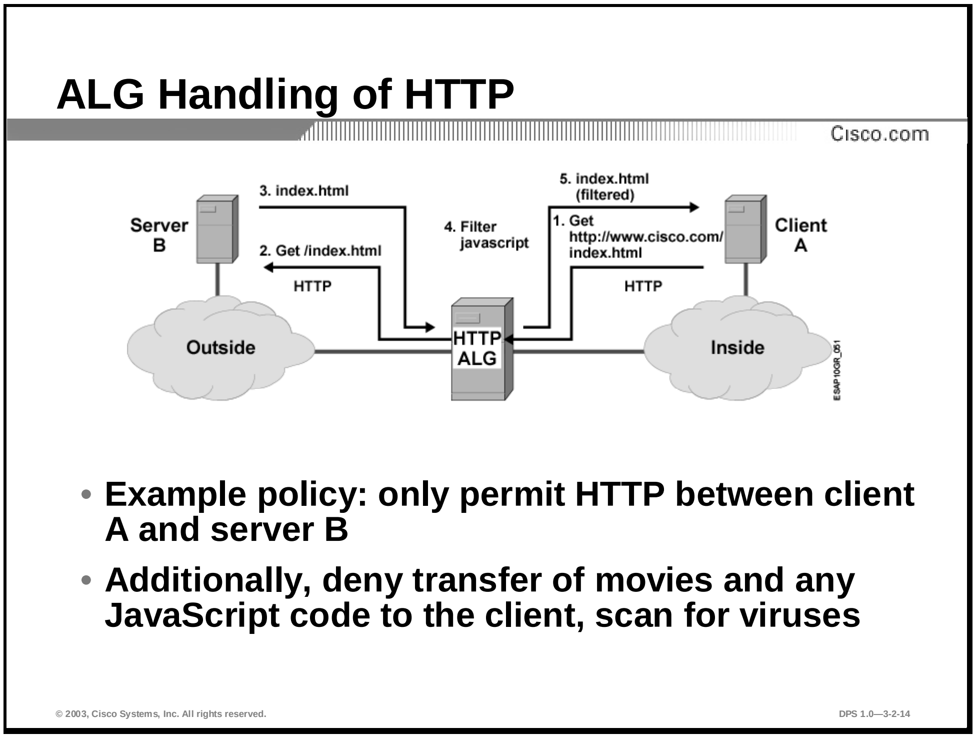
**Step 3** The destination server replies to the ALG.

**Step 4** The ALG passes the response, which may be filtered and changed, back to the client.

Each ALG is designed to support a particular application. For example, an FTP ALG usually only passes the FTP protocol between endpoints. For a pure ALG-based firewall, each application passing over it requires its own ALG.

## Example

A well-known example of an ALG is an HTTP proxy, such as Squid. HTTP proxies, while not necessarily security-focused or firewall-based, pass the HTTP protocol between HTTP clients (browsers) and servers (web servers). An HTTP proxy might be used for its HTTP request/response filtering capabilities, or only to provide acceleration of web access using caching methods.



ALG Handling of HTTP

- Example policy: only permit HTTP between client A and server B
- Additionally, deny transfer of movies and any JavaScript code to the client, scan for viruses

© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—3-2-14

## ALG Handling of HTTP

This figure illustrates an HTTP ALG used in the context of a firewall.

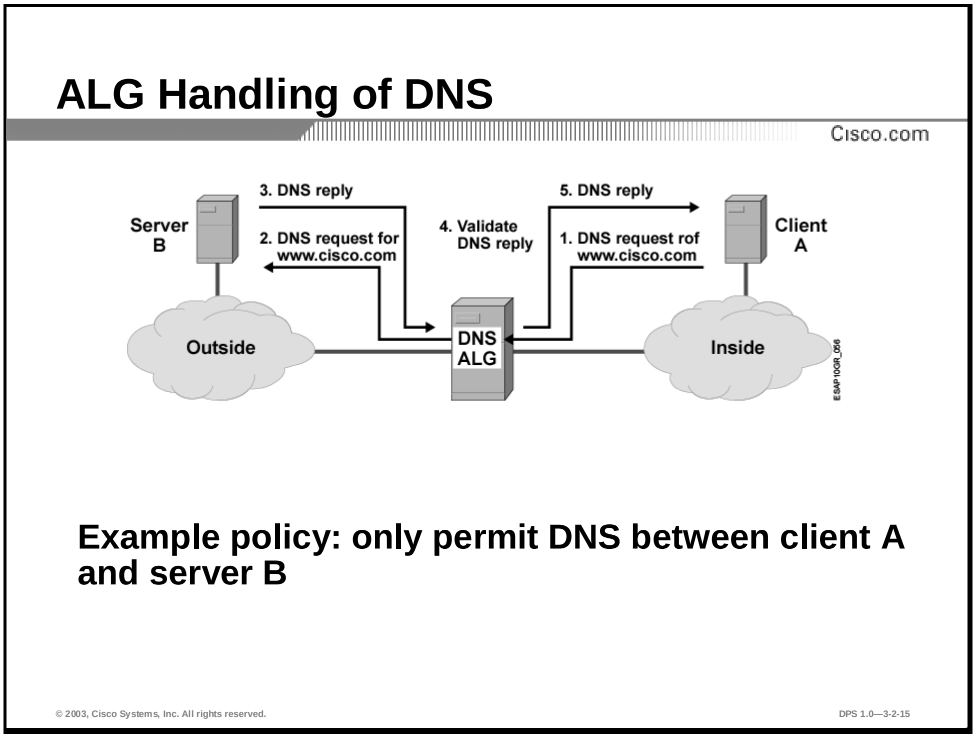
**Step 1** A client in the protected network opens an HTTP session to the proxy and submits a HTTP request. For example, the client might submit the uniform resource identifier (URI) http://www.cisco.com/univercd/index.html to the ALG, expecting the ALG to retrieve the object for the client.

**Step 2** The ALG examines the request, verifies its validity and conformance to the HTTP protocol, it then contacts the destination server (www.cisco.com), and retrieves the object "/univercd/index.html".

**Step 3** The ALG sends the response, which can be filtered (JavaScript stripping, virus scanning) to the client.

Configuration of the access rules occurs in the ALG's configuration. The ALG can be configured only to allow access from specific clients to specific destination HTTP servers. Because of ALG's application awareness, the access rules can filter on any part of the client request or server response, such as the URL, file types, and HTTP request types.

**Note** The client in the example is proxy-aware and uses an extension of the HTTP protocol (proxy URIs inside the HTTP session) to talk to the ALG. The ALG uses pure HTTP to talk to the destination servers.



## ALG Handling of DNS

Cisco.com

**Example policy: only permit DNS between client A and server B**

DPS 1.0—3-2-15

# ALG Handling of DNS

This figure illustrates a DNS ALG passing traffic over an ALG-based firewall.

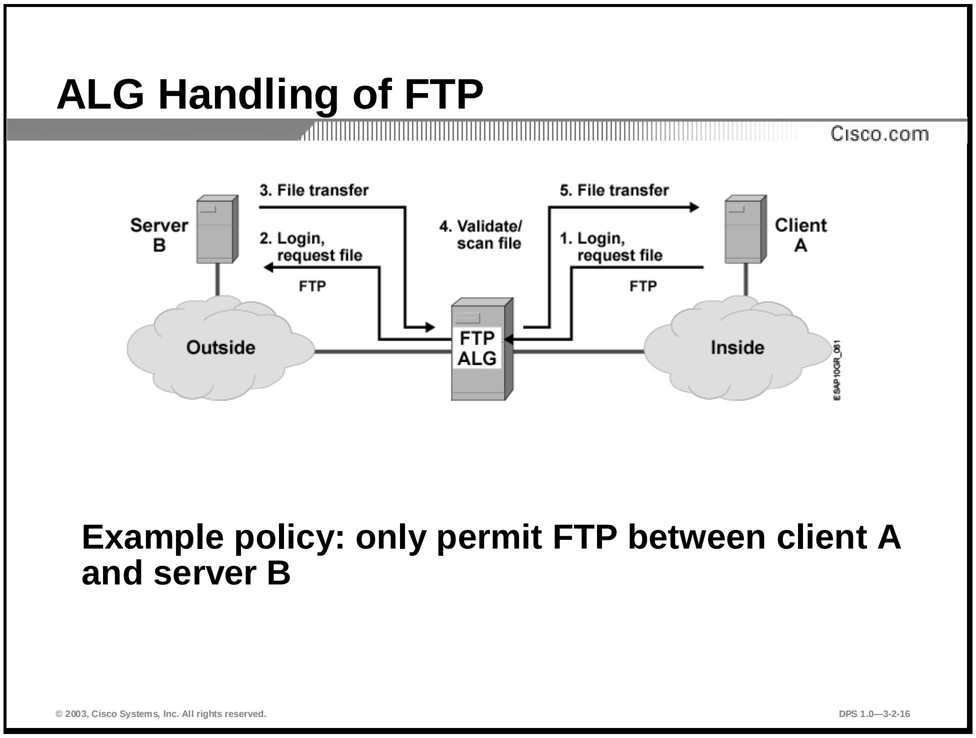
**Step 1** The inside client submits a DNS request to the ALG running on the firewall host.

**Step 2** The ALG, who appears to be a DNS server to the inside client, accepts the request, validates it, and passes it to an appropriate destination DNS server.

**Step 3** The destination DNS server replies to the DNS ALG, which validates the reply, and returns it to the client.

# Example

A common example of a DNS ALG is a caching nameserver, which runs off-the-shelf DNS software configured only to proxy DNS. A caching name server may or may not be authoritative for a domain, which is a separate role it may be used for.



## ALG Handling of FTP

Cisco.com

**Example policy: only permit FTP between client A and server B**

DPS 1.0—3-2-16

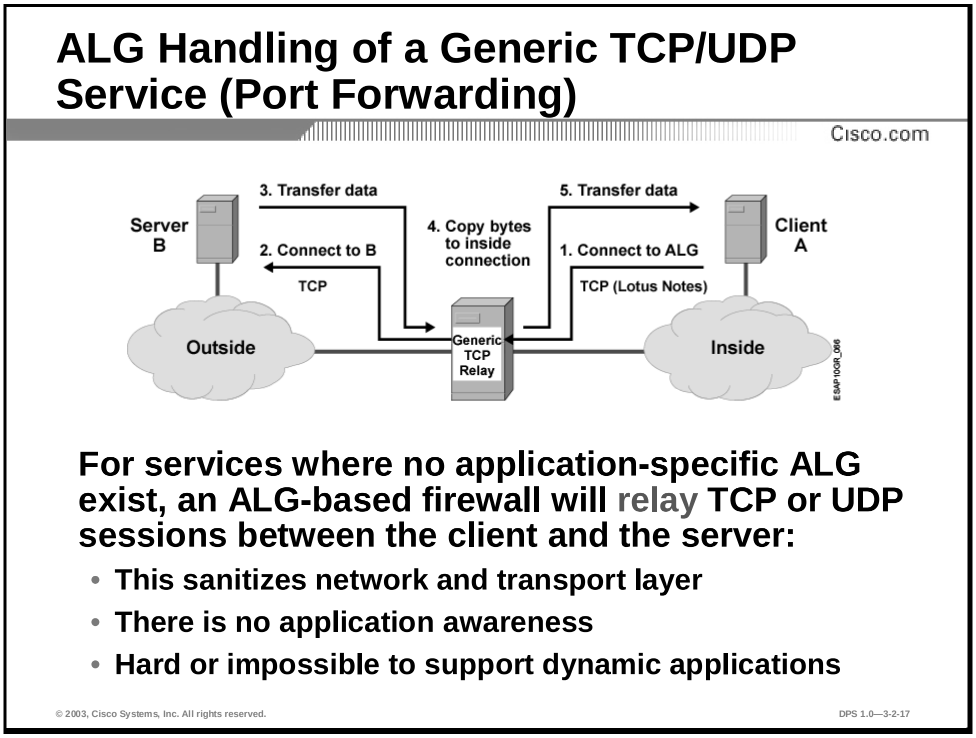
## ALG Handling of FTP

This figure illustrates an FTP ALG passing traffic over an ALG-based firewall.

**Step 1** The inside client starts an FTP session with the FTP ALG, authenticating and passing a request for a remote file.

**Step 2** The FTP ALG poses as the destination server to the client. After receiving the client request, the FTP ALG opens a new FTP session to the destination server, and proxies the client's request to it.

**Step 3** The destination server sends the file to the FTP ALG, which filters the response with, for example, a virus scanner, and passes the file to the client over the client-ALG FTP session.



## ALG Handling of a Generic TCP/UDP Service (Port Forwarding)

Cisco.com

For services where no application-specific ALG exist, an ALG-based firewall will relay TCP or UDP sessions between the client and the server:

- This sanitizes network and transport layer
- There is no application awareness
- Hard or impossible to support dynamic applications

DPS 1.0—3-2-17

## ALG Handling of Generic TCP/UDP Services

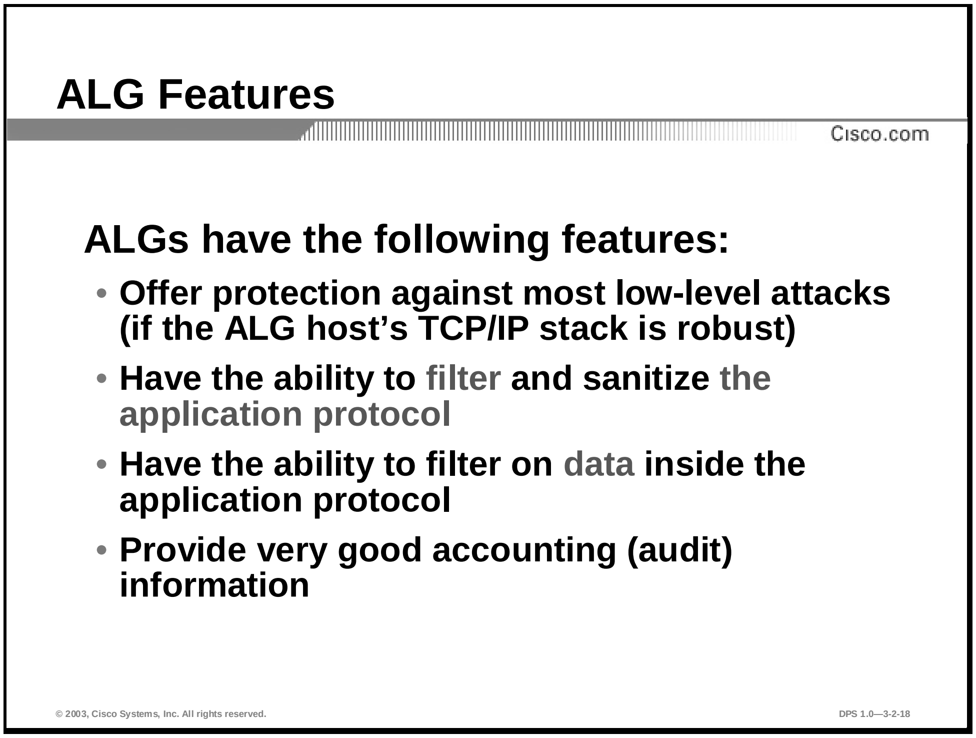
Each application should have its own ALG, which can filter on all aspects of its operation. Often however, ALG-based-firewall vendors have only developed a few ALGs, as the number of applications exploded with the Internet's expansion. Because of this, a significant percentage of applications were difficult or even impossible to proxy. Examples of these are complex multimedia protocols.

For such applications, ALG-based firewalls often resort to two solutions:

- **Usage of port-forwarding TCP or UDP relays:** These are simple agents, which can pass a TCP or UDP session between a client and a server without application-layer filtering. This approach enables connectivity, but runs opposite to the very idea of an ALG—to be able to verify and filter the application protocol. A benefit of this approach is to, at a minimum, sanitize the network and transport layer protocol, if the ALG runs on a host with a robust TCP/IP stack.

- **Usage of (stateful) packet filtering for unsupported applications:** This approach again negates the benefits of an ALG, and is a tradeoff. Therefore, it might compromise the security of the entire firewall.

With this in mind, the biggest weakness of ALG-based firewalls is that, for a significant percentage of mainstream applications, no ALG software exists. ALG-based firewalls therefore have to resort to port forwarding or packet filtering, and the firewall designer has to focus even more on hardening the application endpoints themselves to make up for the lack of an ALG.



## ALG Features

**ALGs have the following features:**

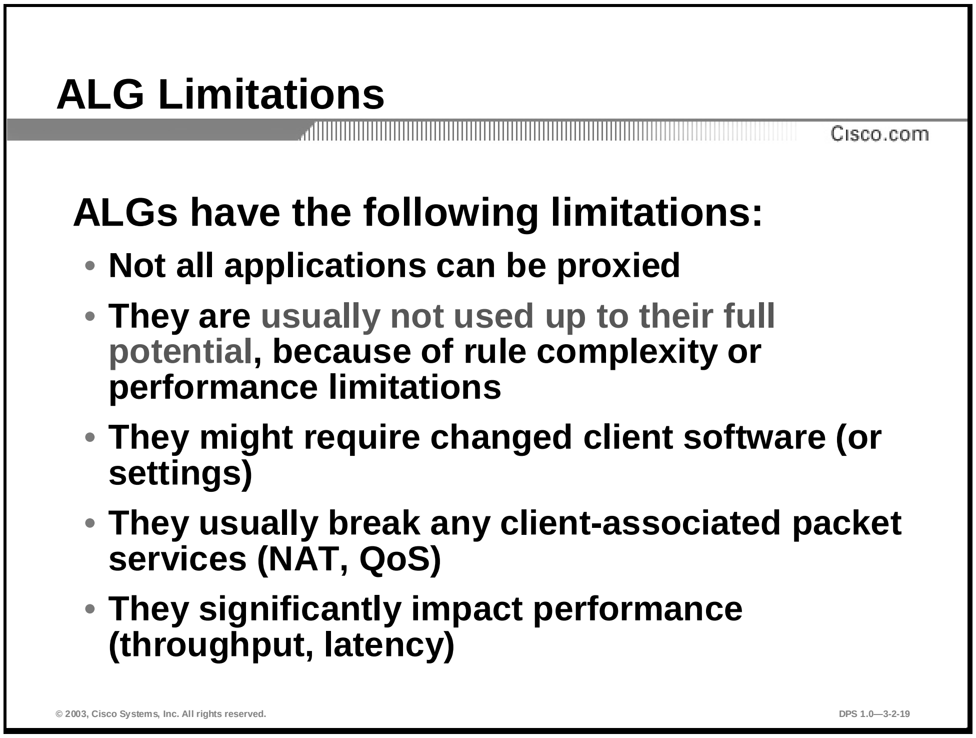
- **Offer protection against most low-level attacks (if the ALG host's TCP/IP stack is robust)**
- **Have the ability to filter and sanitize the application protocol**
- **Have the ability to filter on data inside the application protocol**
- **Provide very good accounting (audit) information**

DPS 1.0—3-2-18

# ALG Features

From the perspective of a firewall designer, an ALG-based firewall has the following features:

- As all application sessions terminate on the ALG, the ALG's TCP/IP stack can protect against network and transport-layer attacks (for example, TCP Loopback DoS Attack [land.c], source routing and TCP SYN flooding)
- Ability to filter and sanitize the application protocol, to prevent the majority of protocol-level attacks, and to resist basic attempts of tunneling
- Ability to filter data inside the application protocol, to prevent data-driven attacks, and leaking of sensitive information
- Very good accounting, as it looks at all data from the application perspective



## ALG Limitations

Cisco.com

**ALGs have the following limitations:**

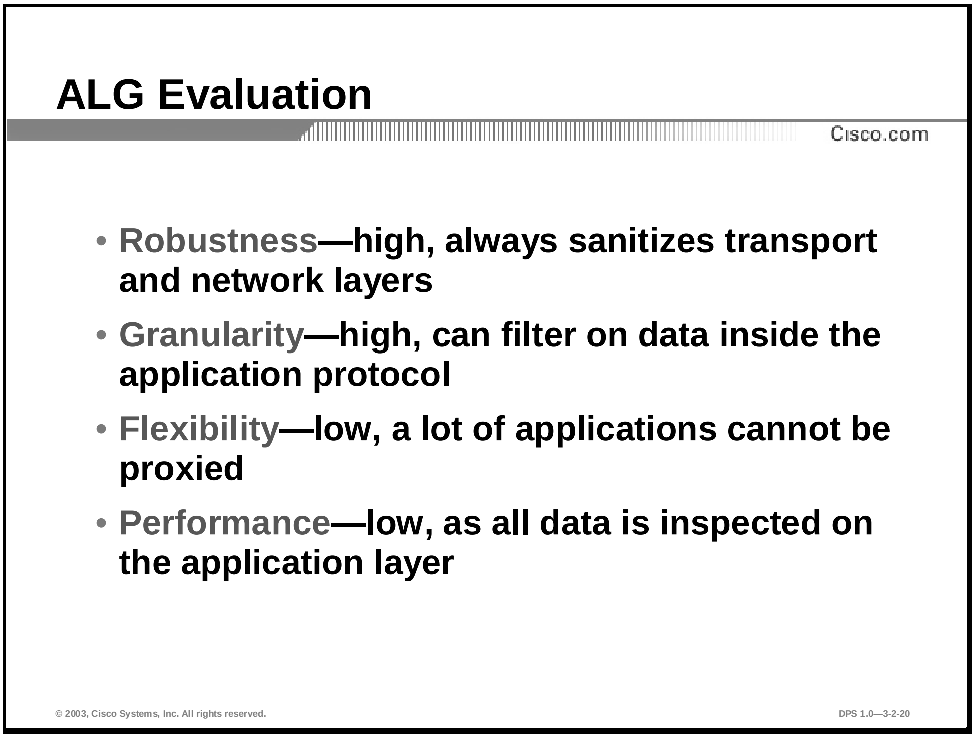
- **Not all applications can be proxied**
- **They are usually not used up to their full potential, because of rule complexity or performance limitations**
- **They might require changed client software (or settings)**
- **They usually break any client-associated packet services (NAT, QoS)**
- **They significantly impact performance (throughput, latency)**

DPS 1.0—3-2-19

# ALG Limitations

The ALG approach has the following major weaknesses:

- A relatively small number of ALGs exist to support modern applications, forcing a designer to make unwelcome compromises.
- ALGs are frequently not used to their full potential, as many applications are too complex to describe their details to the ALG. For example, it would be beneficial for an ALG protecting a custom web application, to check all sensitive parameters passed between the client and the server. However, this would require extensive customization of the ALG. Such customization is often not practical or may be impossible due to poor communication with developers or non-disclosure of the application protocol, rendering the ALGs as robust as stateful packet filters.
- ALGs might require clients to use modified proxy-aware software or modified client settings.
- As ALGs terminate application sessions, any packet service (header marking, translation), associated with the client is lost outside the ALG, because the ALG sanitizes the IP protocol and hides the client’s identity. For example, router Network Address Translation (NAT) and client-specific quality of service (QoS) cannot be deployed with an ALG in path.
- ALG processing can significantly impact throughput and latency of a firewall system.



## ALG Evaluation

- Robustness—high, always sanitizes transport and network layers
- Granularity—high, can filter on data inside the application protocol
- Flexibility—low, a lot of applications cannot be proxied
- Performance—low, as all data is inspected on the application layer

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—3-2-20

## ALG Evaluation

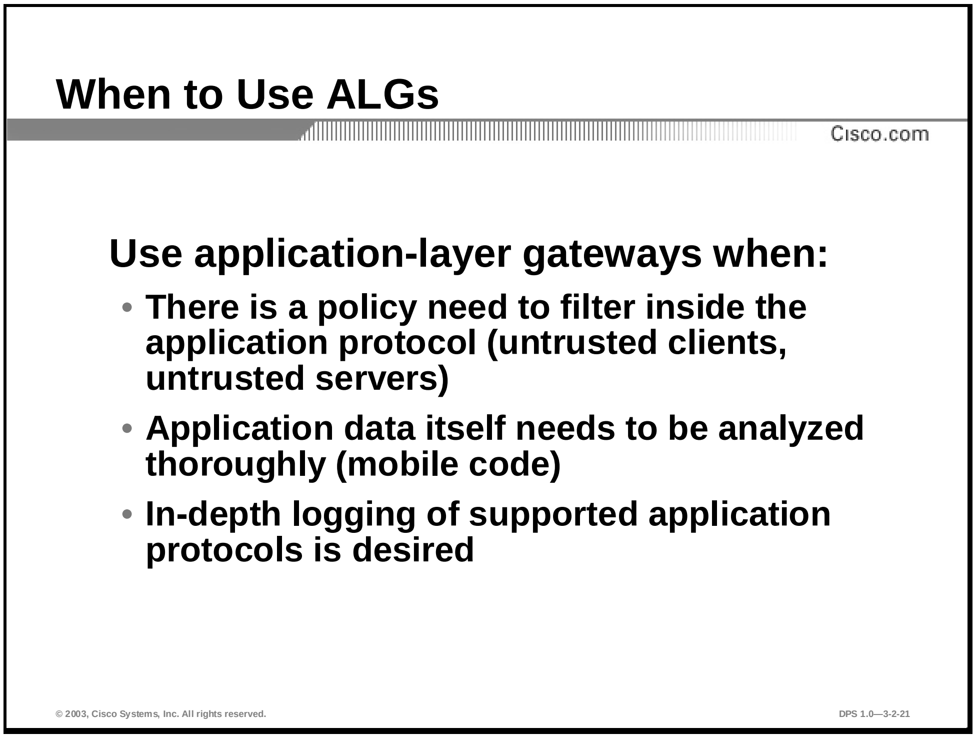
ALGs can be evaluated against the criteria for firewall technology evaluation, which includes:

- **Robustness of filtering:** High. ALGs have very robust filters. They sanitize the network and transport protocols, and as they speak to the application protocol, they have the ability to block any suspicious protocol messages between the endpoints.

- **Granularity of filtering:** High. ALGs can theoretically filter on any aspect of the application protocol.

- **Flexibility of filtering:** Low. Each application requires its own ALG to be developed.

- **Performance:** Low. ALG processing is extremely demanding on the host system.



## When to Use ALGs

Cisco.com

**Use application-layer gateways when:**

- **There is a policy need to filter inside the application protocol (untrusted clients, untrusted servers)**
- **Application data itself needs to be analyzed thoroughly (mobile code)**
- **In-depth logging of supported application protocols is desired**

DPS 1.0—3-2-21

## ALG Deployment Guidelines

In modern networks, ALGs are deployed when the following policy requirements need to be addressed:

- Filtering inside the application protocol is required to either protect trusted clients from untrusted servers, or to filter data from untrusted clients to trusted servers.
- With regard to application protocol filtering, ALGs can easily inspect application-layer data in detail. For example, when tight control over mobile code is desired, an ALG is used to pass and analyze data between perimeters.
- When extensive logging of all application-layer transactions is required to maintain a detailed audit trail.

# Practice

Q1) What are the two benefits of ALG technology, when compared to packet filters? (Choose two.)

A) flexibility in application support

B) granularity of filtering

C) performance

D) robust filtering

E) support for real-time traffic

Q2) How do ALGs handle services, for which there is no specific proxy code available? (Choose one.)

A) by using a generic TCP/UDP forwarder

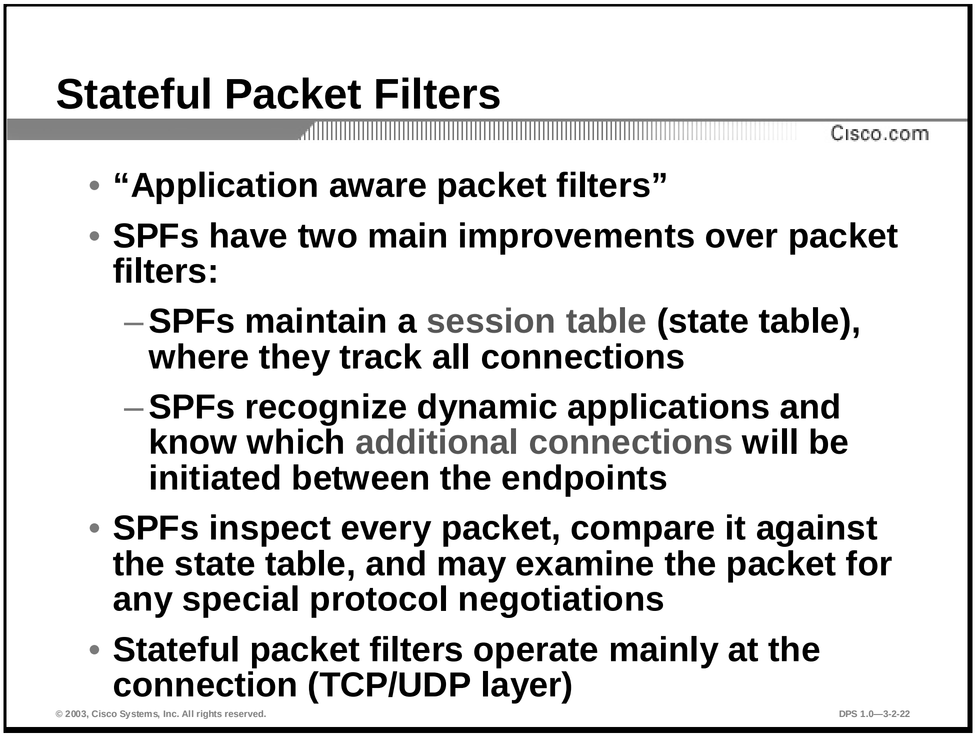
B) ALGs cannot handle such services

C) by always reverting to packet filtering

D) by using another (existing) proxy to support such a service

E) by simply routing such packets on Layer 3

# Stateful Packet Filters



## Stateful Packet Filters

Cisco.com

- "Application aware packet filters"
- SPFs have two main improvements over packet filters:
  - SPFs maintain a session table (state table), where they track all connections
  - SPFs recognize dynamic applications and know which additional connections will be initiated between the endpoints
- SPFs inspect every packet, compare it against the state table, and may examine the packet for any special protocol negotiations
- Stateful packet filters operate mainly at the connection (TCP/UDP layer)

DPS 1.0—3-2-22

## Objective

The section will enable the learner to explain the security properties, features, and limitations of stateful packet filtering used as an access control method, and identify their features and limitations.

## Introduction

Stateful packet filtering is currently the most widely used firewalling technology. This section describes the operation, features, and limitations of stateful packet filtering technology when used as a firewall building block.

## Stateful Packet Filtering Technology

In the mid-nineties, packet filters and ALGs were the two technologies used to build firewall systems. As the number of applications that needed to pass through firewalls increased, ALG-based firewall vendors could not keep up with the development of new ALGs. On the other hand, packet filtering also could not support the dynamic nature of the many modern applications. Thus, a new technology was born.

# Stateful Packet Filtering Definition

Stateful packet filtering is an application aware method of packet filtering that works on the connection (flow) level. A stateful packet filter (SPF):

- Maintains a state table (or connection table), where it keeps track of all the active sessions over the firewall

- Is application aware—a SPF is able to recognize all session of a dynamic application

## The State Table

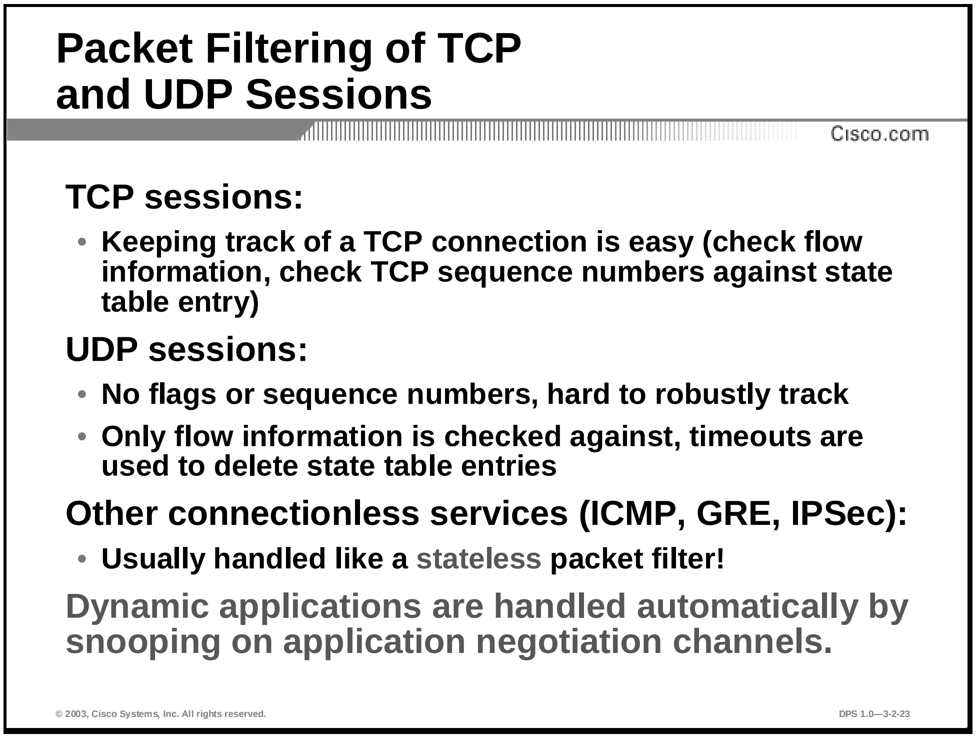
The state table is part of the internal data structure of a SPF. It tracks all the sessions, and inspects all the packets passing over the SPF-based firewall. The packets only pass if they have the expected properties that the state table predicts. The state table dynamically changes and adapts with the traffic flow. If no state exists, one is created and entered into the state table if meeting the rules allowed in the firewall.

## Application Awareness

SPFs are application-aware through additional inspection of passing traffic. By inspecting the session more closely, usually on the application layer, a SPF is able to associate any dynamic channels of the application with the application's initial session.

The concept of a session in the SPF world is mainly connected to the TCP and UDP notion of a session. Some SPF implementations though, can keep state of other protocols, such as the ICMP or generic routing encapsulation (GRE).

**Note** Stateful packet filters do not usually change packet headers or payloads in any way. Packets are only compared against the state table and, if permitted, transmitted in their original form.



## Packet Filtering of TCP and UDP Sessions

TCP sessions:

- Keeping track of a TCP connection is easy (check flow information, check TCP sequence numbers against state table entry)

UDP sessions:

- No flags or sequence numbers, hard to robustly track
- Only flow information is checked against, timeouts are used to delete state table entries

Other connectionless services (ICMP, GRE, IPSec):

- Usually handled like a stateless packet filter!

Dynamic applications are handled automatically by snooping on application negotiation channels.

© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—3-2-23

## SPF Handling of TCP Sessions

When a SPF-based firewall permits a TCP session, the session creates an entry in the state table. SPFs check every subsequent packet against the state table to verify that each packet is the next expected packet in the session. SPFs robustly filter TCP sessions. They check each packet’s flow information (network addresses and transport layer ports) to find a matching entry in the state table, and verify that the TCP sequence and acknowledgement numbers are within the expected range. There is a window of allowed values to allow minor reordering of packets, which is legal in IP networks.

SPFs usually process TCP flags to ensure that a session starts with a proper three-way handshake. The SPFs then remove the state table entry after the session has closed with a connection close, or with a forceful teardown using the RST flag. Timeouts delete half-open, half-closed, and idle TCP sessions.

## SPF Handling of UDP Sessions

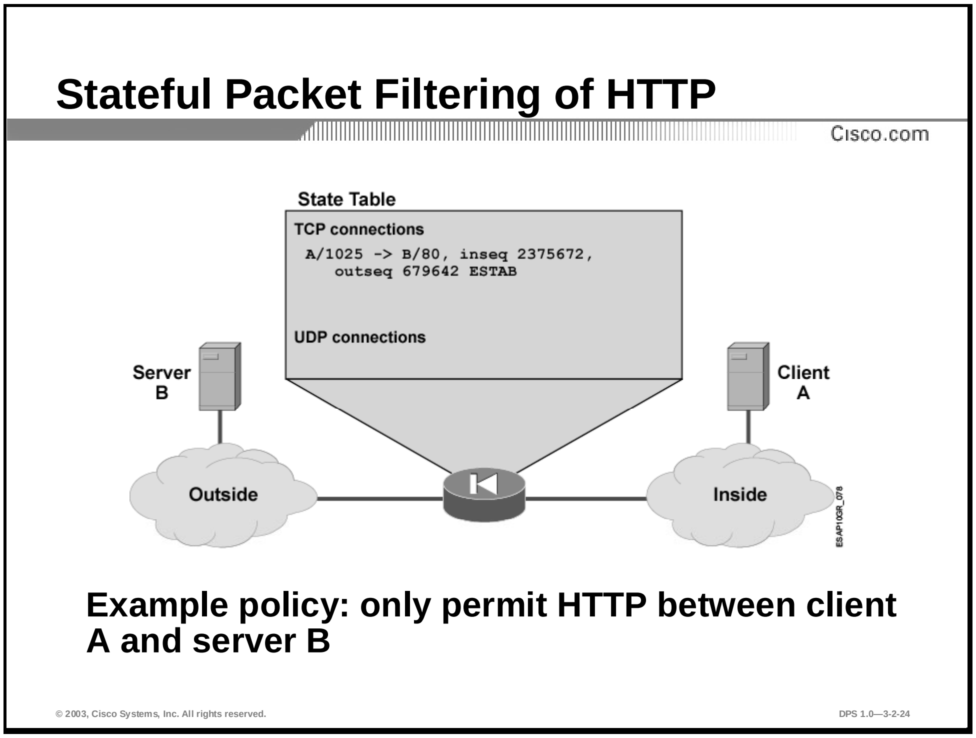
The UDP protocol does not contain sufficient information in each packet robustly to verify the integrity of the UDP session, or its opening or closing. A stateful filter, when permitting a UDP application, creates a state table entry when the first UDP packet is permitted. The state table will contain flow information (network addresses and transport layer ports), and an idle timer. The SPF permits all packets of the session if they match the flow description, and the state table entry is deleted when the idle timer expires.

## SPF Handling of Other IP Sessions

SPFs do not usually track other protocol sessions, such as ICMP and GRE, but handles them statelessly, similar to a classic packet filter. If stateful support is provided for other protocols, it is usually similar to that of UDP. When a protocol flow is initially permitted, all packets matching the flow are permitted until an idle timer expires.

## SPF Handling of Dynamic Applications

Dynamic applications open a channel on a well-known port (such as FTP), and then negotiate additional channels through the initial session. SPFs support these dynamic applications through SPF snooping of the initial session, and parsing the application protocol enough to learn about the additional negotiated channels. Then SPF usually enforces the policy that if the initial session was permitted, any additional channels of that application should be permitted as well.



### Stateful Packet Filtering of HTTP

State Table

TCP connections

```
A/1025 -> B/80, inseq 2375672,
   outseq 679642 ESTAB
```

UDP connections

**Example policy: only permit HTTP between client A and server B**

DPS 1.0—3-2-24

## SPF Handling of HTTP

This figure illustrates an SPF filtering an HTTP session between a client and a server. The operator, who configures the firewall, does not have to be aware of the bidirectional flow of packets or any dynamic channels opened by the application—this is automatically handled by the stateful intelligence. The operator simply permits an application between two endpoints.

When the client initiates the HTTP session to the server:

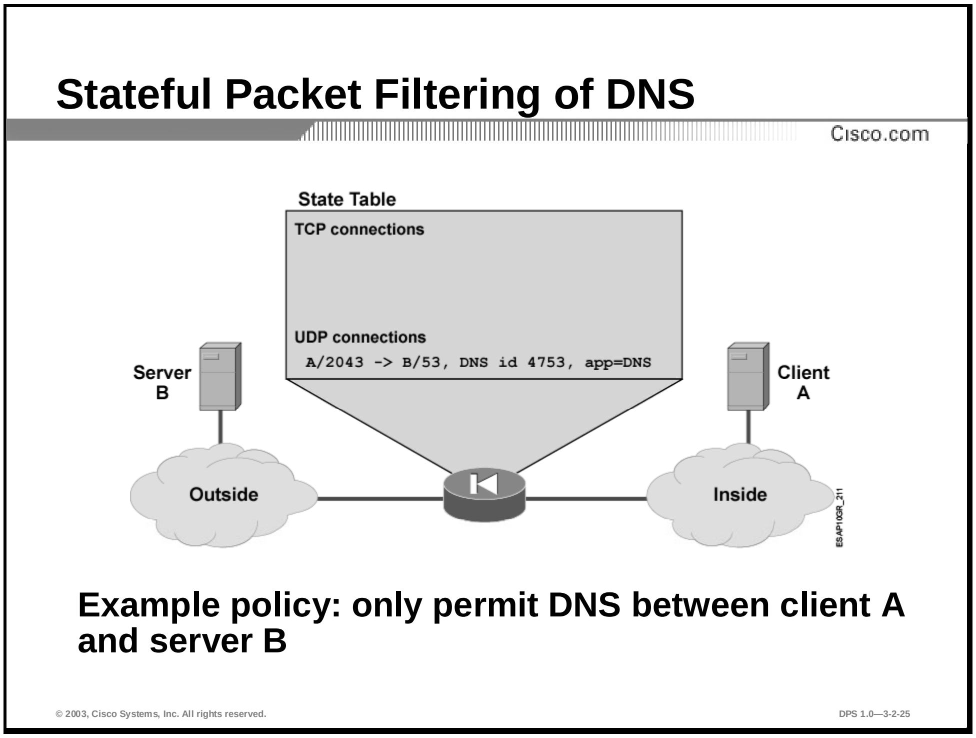
**Step 1** A SPF compares the initial TCP segment against the SPF-based firewall access rules and is permitted, or denied, per the access rules.

**Step 2** The TCP segment with the SYN flag creates a state table entry where the flow information and initial sequence number is recorded. The session then dispatches the packet to the server unchanged (unless the firewall performs NAT).

**Step 3** The server replies with a SYN/ACK segment in the three-way handshake, which the SPF verifies against the state table. The packet passes to the client once the flow information, flags, and sequence numbers agree with the predicted values.

**Step 4** The client completes the handshake and sends a request to the server. The server replies directly to the client.

SPFs verify every packet's headers in the application session against the state table to ensure the packets are not spoofed. When the connection closes, the state table entry is removed from the state table.



## Stateful Packet Filtering of DNS

Cisco.com

**Example policy: only permit DNS between client A and server B**

DPS 1.0—3-2-25

# SPF Handling of DNS

This figure illustrates a SPF filtering a DNS session between a client and a server.

When the client initiates the DNS request to the outside server:

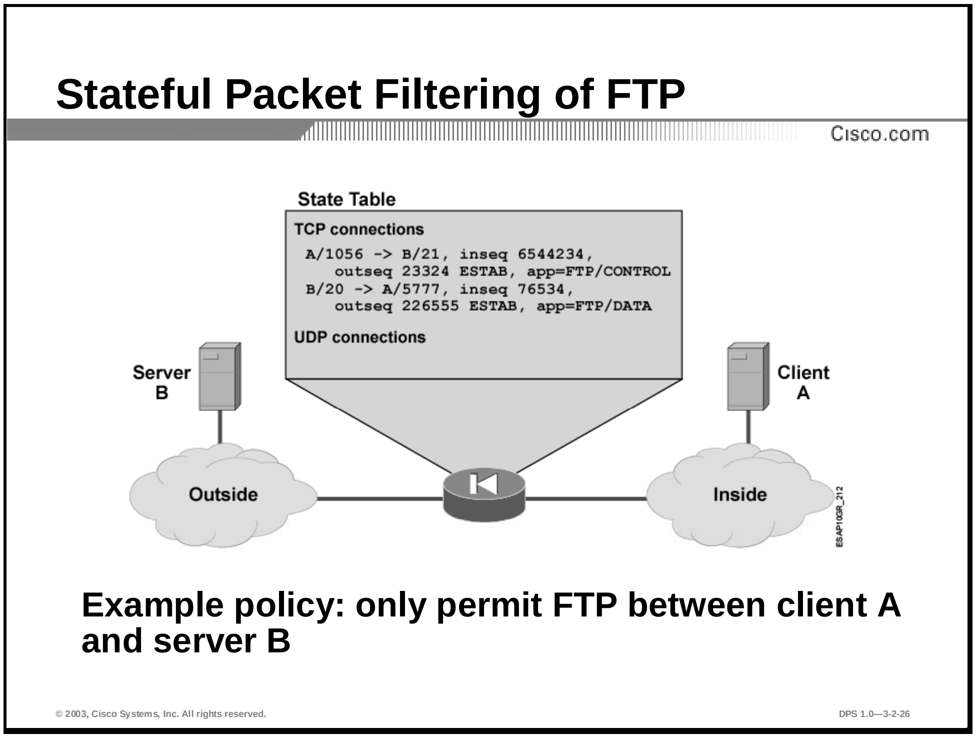
**Step 1** A SPF verifies the DNS request packet against theaccess rules.

**Step 2** If the packet is permitted, a UDP "connection" entry is created in the state table, remembering the addresses and ports of the flow.

**Step 3** When the DNS server replies, the state table verifies the reply packet. The reply to the client is permitted only if the flow information of the request packet exactly matches the state table entry.

The state table slot may be cleared after an idle timeout.

**Note** Many SPF implementations recognize DNS as a special UDP protocol and remove the state table entry as soon as the first response is received. The Cisco Secure PIX Firewall will also track the DNS ID field, giving an additional sequence-number-like protection to the DNS protocol.



## Stateful Packet Filtering of FTP

Cisco.com

State Table

TCP connections

```
A/1056 -> B/21, inseq 6544234,
   outseq 23324 ESTAB, app=FTP/CONTROL
B/20 -> A/5777, inseq 76534,
   outseq 226555 ESTAB, app=FTP/DATA
```

UDP connections

**Example policy: only permit FTP between client A and server B**

DPS 1.0—3-2-26

## SPF Handling of FTP

This figure illustrates a SPF filtering a FTP session between a client and a server. FTP is a dynamic protocol, which opens multiple sessions between the client and the server. An FTP control session authenticates the client user to the server and allows the client to browse the server and specify which files are to be transferred. This session connects to the server port of 21.

The inside client initiates the FTP’s control session to the destination server. If this session is permitted in the firewall rules:

**Step 1** A state table entry is created and the session is inspected as any other TCP session.

**Step 2** By looking at the destination port number, the SPF recognizes this as a FTP session and focuses more intensively on the FTP control session.

**Step 3** When the client is ready to receive data, it will signal this to the server by sending a “PORT” command to the server. This indicates on which local port it is listening, so that the server can open a new connection to the client, and transfer a file over it.

The syntax of the port command is:

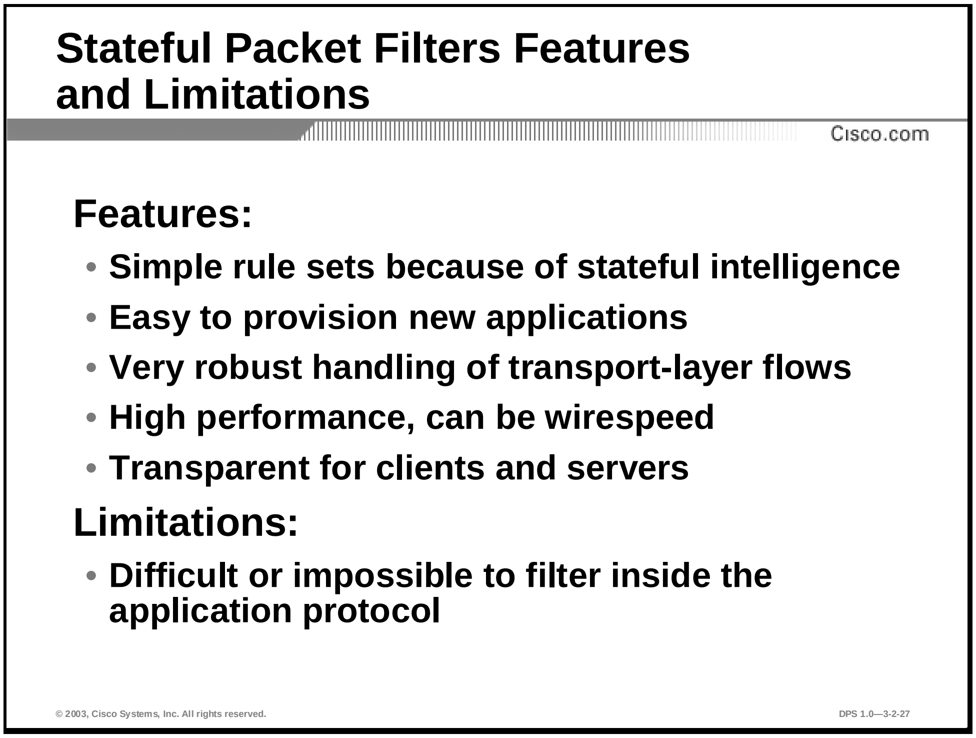
```
PORT A,A,A,A,L,H
```

Where the symbol “A” represents individual bytes of the client’s IP address, and the symbol L and H represent the low- and high-order byte of the client port. If, for example, the client 193.77.3.133 starts listening on port 1025, it will send the following command to the server over the control session:

```
PORT 193,77,3,133,4,1
```

The SPF intercepts this command by snooping on the control session, and now knows there will be a session opening from the server, usually from port 20, to the client on port 1025

(4*256 + 1*1). When this session arrives inbound to the SPF, it permits it as a part of the overall FTP application session and performs standard TCP stateful filtering on it.



## Stateful Packet Filters Features and Limitations

Cisco.com

**Features:**

- **Simple rule sets because of stateful intelligence**
- **Easy to provision new applications**
- **Very robust handling of transport-layer flows**
- **High performance, can be wirespeed**
- **Transparent for clients and servers**

**Limitations:**

- **Difficult or impossible to filter inside the application protocol**

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—3-2-27

# SPF Features and Limitations

The features of SPF technology are:

- **Simple configuration:** The firewall operator does not need to be aware of the application protocol internals—the stateful intelligence handles any exceptional behavior and hides it from the user.

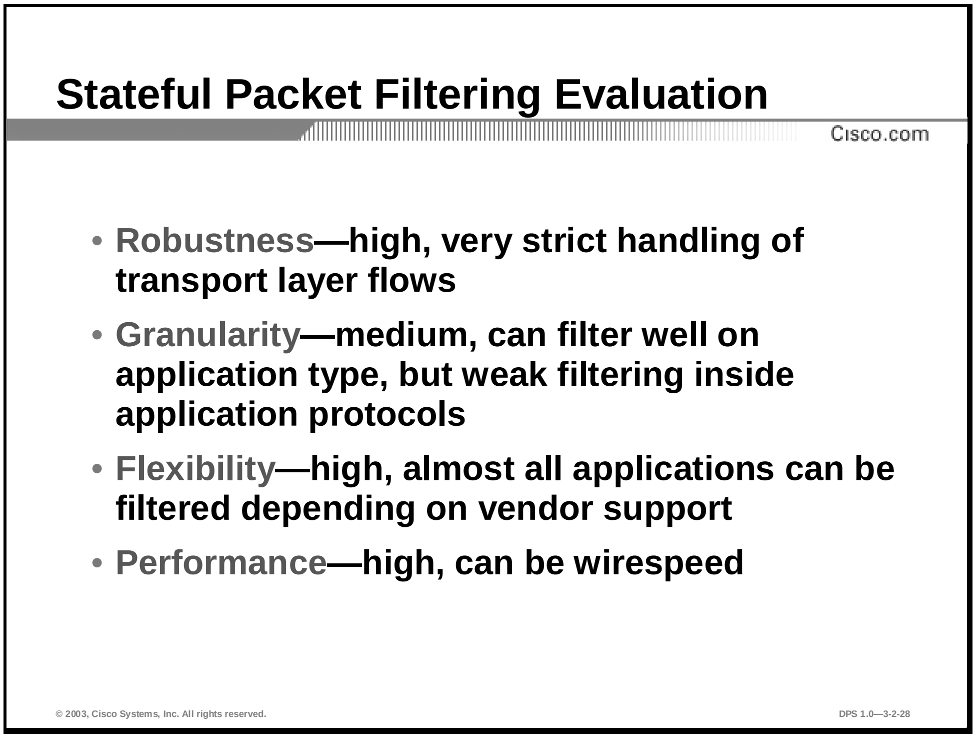
- **Easy enough to provision new applications:** Vendors can develop the "intelligence" needed to provision new applications in a much shorter timeframe compared to application-layer gateways.

- **Very robust filtering:** Especially for TCP flows, where a lot of information is checked against the state table.

- **Very high performance:** SPF performance is high, and is comparable to packet filtering performance.

- **Full transparency for clients and servers:** No application change is necessary to run an application over a SPF-based firewall.

Stateful packet filtering has one major disadvantage—the inability to robustly filter inside the application session. A SPF might have insight in the application protocol necessary to convey its sessions securely over a firewall, but it does not validate every protocol message and does not terminate an application session. Some SPFs have the ability to peek inside application-layer protocols and look for specific malicious messages, but such filtering can sometimes be bypassed.



## Stateful Packet Filtering Evaluation

Cisco.com

- Robustness—high, very strict handling of transport layer flows
- Granularity—medium, can filter well on application type, but weak filtering inside application protocols
- Flexibility—high, almost all applications can be filtered depending on vendor support
- Performance—high, can be wirespeed

DPS 1.0—3-2-28

## SPF Evaluation

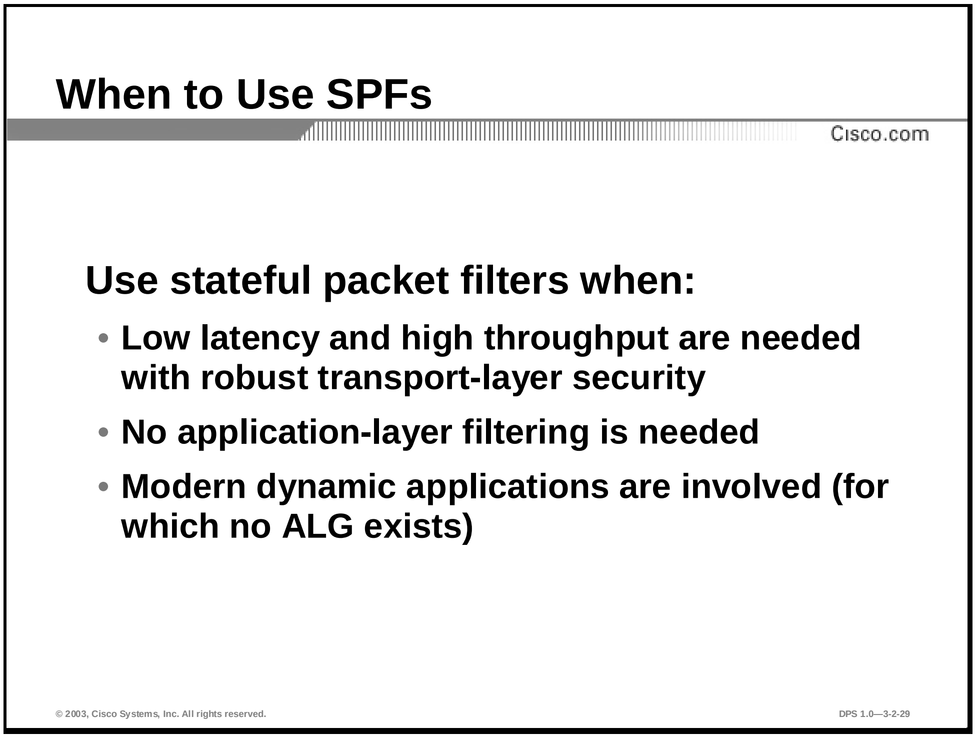
SPFs can be evaluated against the following criteria for firewall technology evaluation:

- **Robustness of filtering:** High. Generally, SPFs strictly verify if a packet is allowed to pass through the filter. This is especially true for TCP-based applications, where an attackers chance of sending spoofed packets through a SPF is extremely small.

- **Granularity of filtering:** Medium. SPFs usually filter up to the transport layer, and can permit specific applications between hosts. Any filtering on the application-layer is usually attempted on a per-packet basis. This means they may be vulnerable to fragmentation/segmentation attacks, where an attacker is able to split malicious data over multiple packets.

- **Flexibility of filtering:** High. SPF vendors can quickly develop inspection intelligence for almost any application.

- **Performance:** High. SPF performance is high, and is comparable to packet filtering performance.



## When to Use SPFs

Cisco.com

**Use stateful packet filters when:**

- **Low latency and high throughput are needed with robust transport-layer security**
- **No application-layer filtering is needed**
- **Modern dynamic applications are involved (for which no ALG exists)**

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—3-2-29

## SPF Deployment Guidelines

SPFs are currently the most common firewall technology and are deployed when:

- Low latency/high throughput connectivity is desired, with much more robust transport-layer filtering when compared to classic packet filters
- No in-depth application-layer filtering is needed at the firewall
- Modern dynamic applications are used, for which there is no application-layer gateway available

SPFs offer a high level of filtering robustness and high performance in a single package. Application security though, often needs to be handled at the application endpoint. In modern applications, performing application security at the firewall is often too complex or even impossible. Because of this, SPFs are now the preferred firewalling method for most applications, with ALGs usually used as a point-solution for a specific application.

# Practice

Q1) What is a major benefit of stateful packet filtering?

A) support for many modern applications

B) application-layer filtering

C) resistance against application attacks

D) prevention of covert channels

E) integration of virus scanning

Q2) How does a SPF usually determine the end of an UDP flow?

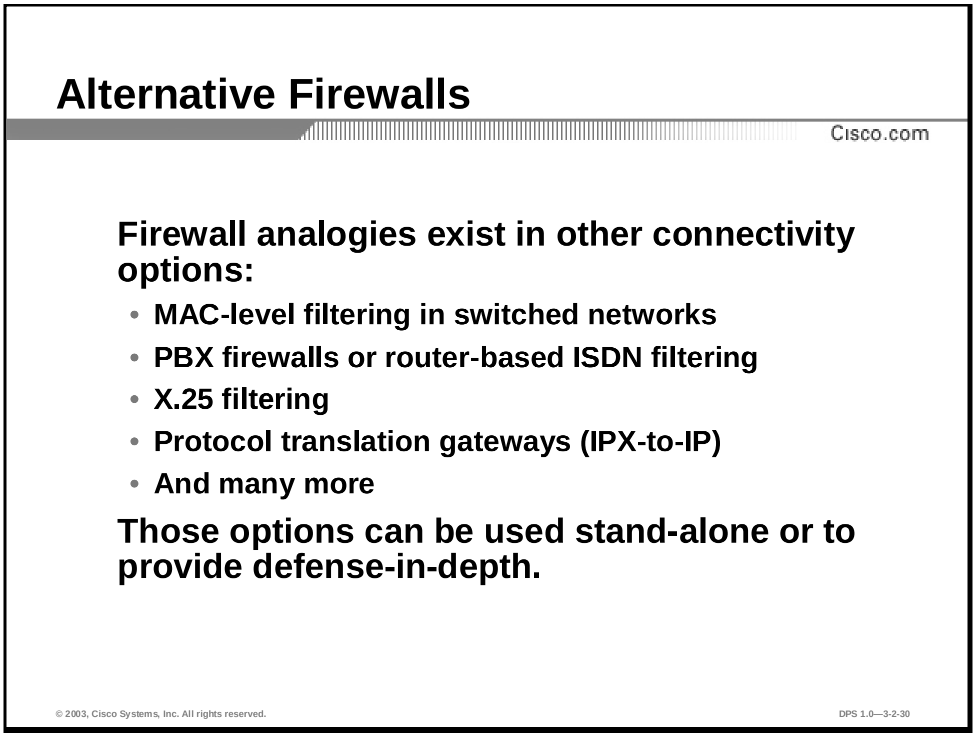
A) through the UDP connection termination flags in packets

B) using an idle timer

C) using an absolute timer

D) by looking at the FIN flag

# Alternative Firewalls



## Alternative Firewalls

Cisco.com

**Firewall analogies exist in other connectivity options:**

- **MAC-level filtering in switched networks**
- **PBX firewalls or router-based ISDN filtering**
- **X.25 filtering**
- **Protocol translation gateways (IPX-to-IP)**
- **And many more**

**Those options can be used stand-alone or to provide defense-in-depth.**

DPS 1.0—3-2-30

## Objective

The section will enable the learner to explain the security properties of other common methods used for network access control.

## Introduction

Many other technologies can be designated as firewalling technologies, if they can be used in the context of access control. This section identifies some such technologies, and describes how they could be used in a firewall system.

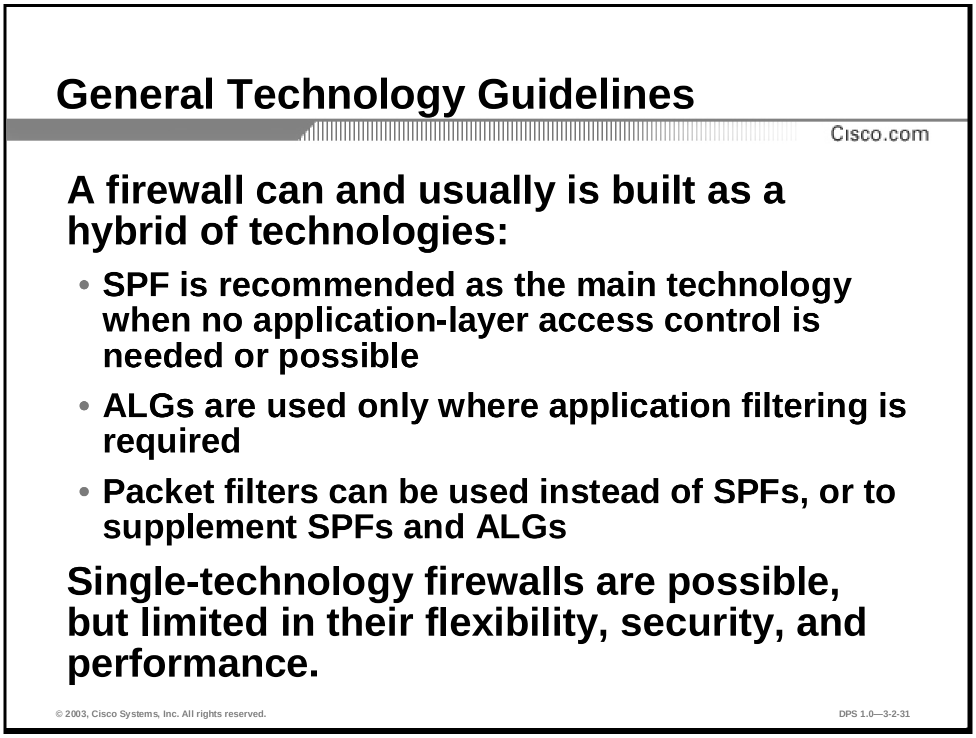
## Alternative Firewall Technologies

Besides filtering of IP applications, other technologies can easily be classified as firewalls, if they perform any access control between networks. Examples of such technologies include:

- Filtering of Layer 2 (L2) frames, using a L2 device such as a dedicated switch or bridged router interfaces
- Setting of static ARP entries or switch CAM entries, which effectively only enables communication between selected hosts
- Filtering of voice/data calls on a PBX
- Filtering of incoming ISDN data calls based on the Caller ID

- Filtering of X.25 sessions based on caller or called party addresses
- Translation of IPX/SPX protocols into TCP/IP, using a gateway that also restricts access

All these options can be used as standalone access control mechanisms or to complement existing methods to provide defense-in-depth.



## General Technology Guidelines

**A firewall can and usually is built as a hybrid of technologies:**

- **SPF is recommended as the main technology when no application-layer access control is needed or possible**
- **ALGs are used only where application filtering is required**
- **Packet filters can be used instead of SPFs, or to supplement SPFs and ALGs**

**Single-technology firewalls are possible, but limited in their flexibility, security, and performance.**

DPS 1.0—3-2-31

## General Technology Guidelines

Modern firewalls are usually built as hybrids using packet filtering, application-layer gateways, and stateful packet filtering. The core technology that provides basic access control is often stateful packet filtering. It is the most extensible and simple-to-use method, and offers the most flexibility and room to grow in the future.

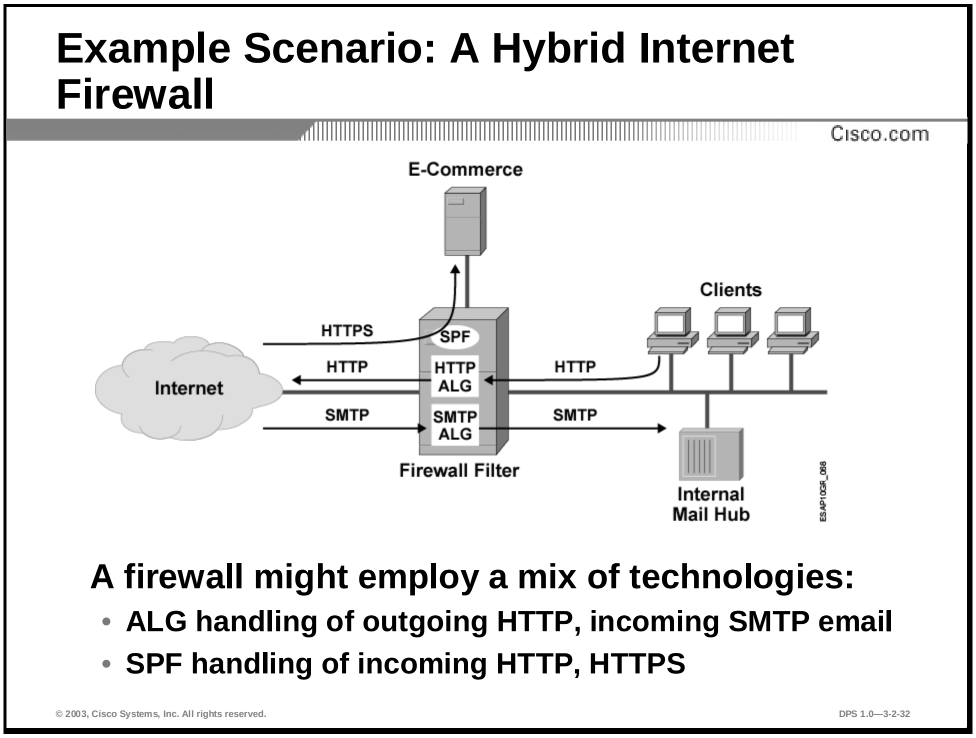
Application-layer gateways are used to augment basic access control. Traffic, which needs application-layer inspection, is redirected to the ALGs.

Packet filters still play an important role:

- As a defense-in-depth mechanism, which duplicates some of the access control on routers. The routers are a part of the firewall, a standalone filtering method.

- To only provide much needed ingress/egress filtering on the network layer to resist spoofing and filter other unnecessary traffic before reaching the firewall as policy dictates and provide a front line of defense.

A single modern technology cannot provide a solution for all organizations. Organizations, who impose more restrictive policies on internetwork access, generally prefer to process most of their data using ALGs, and use SPFs only for specific applications. On the other hand, most organizations find the flexibility of a SPF acceptable for the majority of their applications, and only process a few select applications using ALGs. This hybrid approach does result in firewalls that may be more complex overall, but their functionality is often well-separated on individual systems, and hence easier to control security-wise. The art of firewall design explores how, where, and why to apply a particular filtering technology to a particular application need.



## Example Scenario: A Hybrid Internet Firewall

**A firewall might employ a mix of technologies:**

- **ALG handling of outgoing HTTP, incoming SMTP email**
- **SPF handling of incoming HTTP, HTTPS**

DPS 1.0—3-2-32

## Hybrid Firewalls

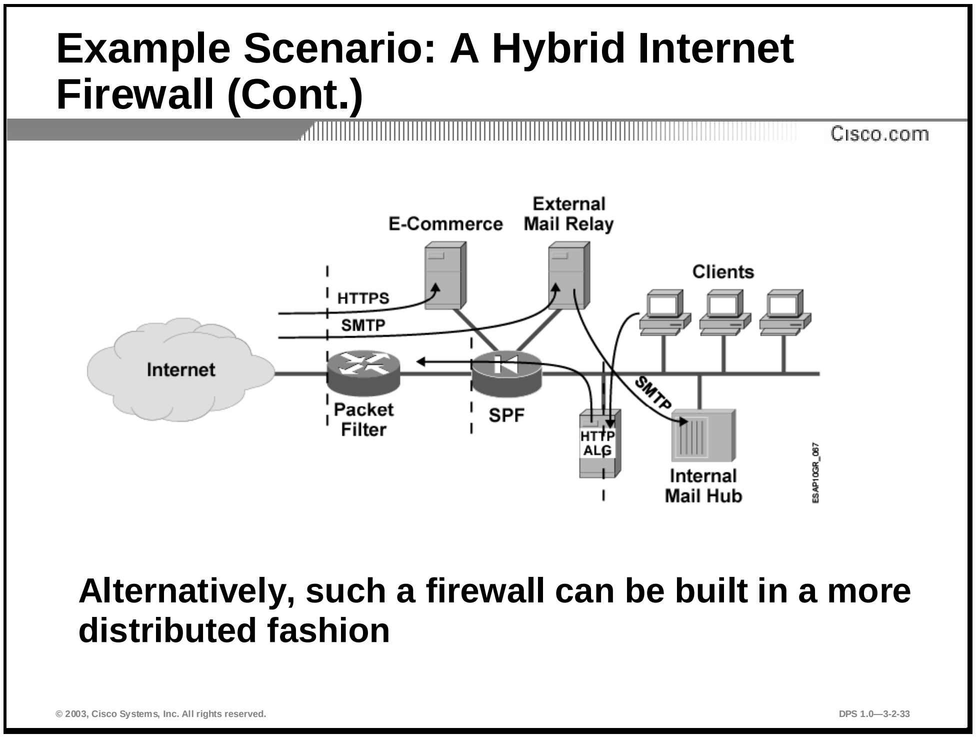
This figure illustrates a hybrid Internet firewall that passes along HTTP and SMTP traffic. Where needed, ALGs provide application-layer filtering for sessions, where malicious content and protocol attacks are likely. For example:

- Outbound HTTP access is only possible by using an ALG, which might strip all JavaScript, VBScript, Java, and ActiveX objects from the HTTP stream

- Exchange of mail is only possible over a dedicated SMTP ALG, which scans all email messages and removes ANY attachments

SPFs provide access control only to specific application end-points if application-layer filtering is not needed:

- Incoming HTTP to public web servers is permitted, and the web server is well-secured against application-layer attacks.



Example Scenario: A Hybrid Internet Firewall (Cont.)

Cisco.com

**Alternatively, such a firewall can be built in a more distributed fashion**

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—3-2-33

This figure shows the same firewall system, built in a more distributed fashion using several dedicated systems to achieve the same goals. The advantages of distributed systems are better security as each piece of the system is less complex in itself, and there is less possibility of unexpected interaction between components, as well as better performance. The disadvantage of distributed systems is primarily the non-centralized management.

## Practice

Q1) Which two technologies can be considered as “alternative” firewalling technologies? (Choose two.)

A) Layer 2 filtering (for example, private VLANs)

B) routing protocols

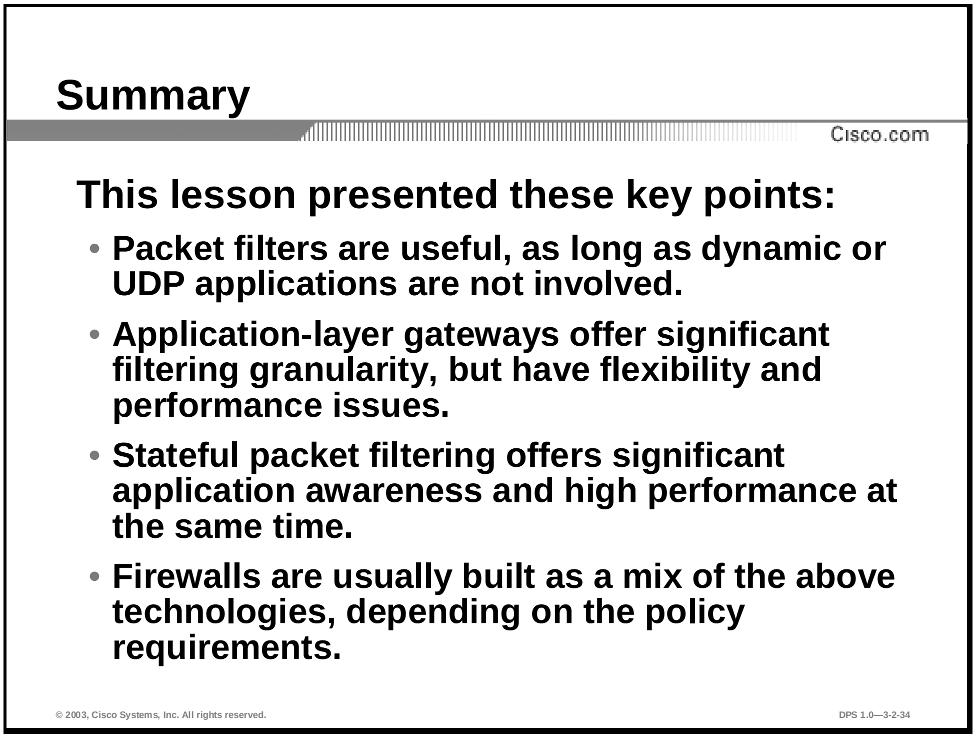
C) call filtering based on Caller-ID

D) NAT

E) process switching

# Summary

This section summarizes the key points discussed in this lesson.



## Summary

This lesson presented these key points:

- Packet filters are useful, as long as dynamic or UDP applications are not involved.
- Application-layer gateways offer significant filtering granularity, but have flexibility and performance issues.
- Stateful packet filtering offers significant application awareness and high performance at the same time.
- Firewalls are usually built as a mix of the above technologies, depending on the policy requirements.

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—3-2-34

## Next Steps

After completing this lesson, go to:

- Firewall Architectures lesson

# Quiz: Firewall Technologies

Complete the quiz to assess what you have learned in this lesson.

## Objectives

This quiz tests your knowledge on how to:

- Compare several common firewall technologies with respect to access control granularity and their limitations

## Instructions

Answer these questions:

1. What are the weaknesses of packet filtering?
2. What is the major benefit of application-layer gateways?
3. Why are SPFs the most popular technology?
4. What are the two options for building a hybrid firewall?

## Scoring

You have successfully completed the quiz for this lesson when you earn a score of 80 percent or better.

# Firewall Architectures

## Overview

This lesson introduces various firewall architectures, which are used to build firewall systems of various functionalities. The features and limitations of each architecture are addressed, as well as their suitability for use to support various customer requirements and applications.

### Importance

The need to understand firewall architectures is paramount when different requirements of separation, performance and filtering capability are presented to a designer.

### Lesson Objective

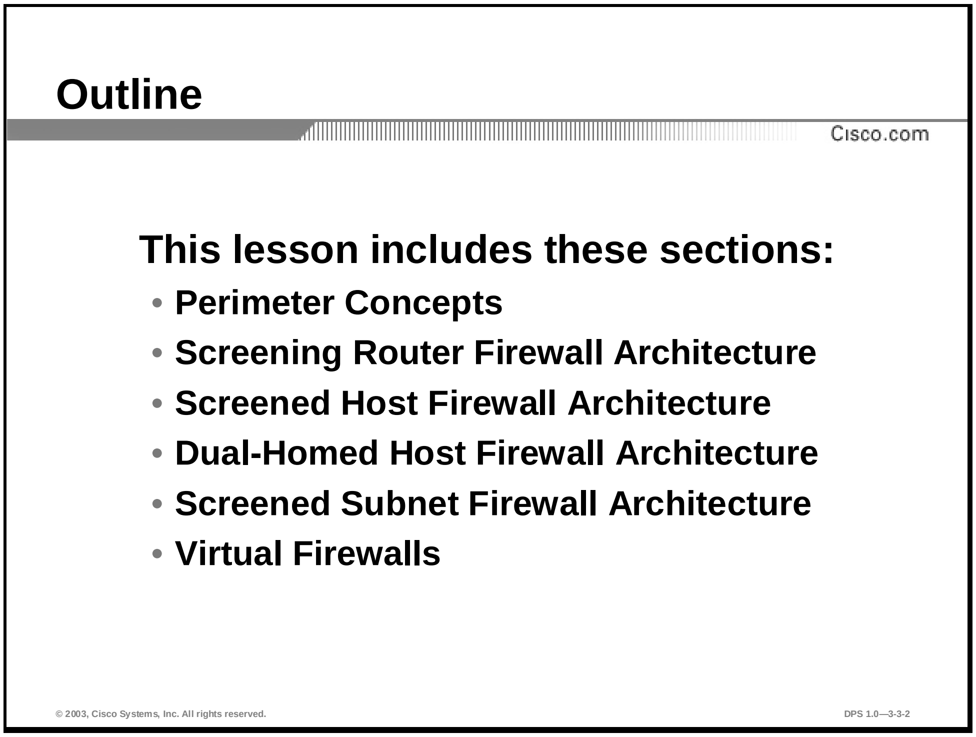
The lesson will enable the learner to compare different basic firewall architectures and to select the proper architecture for an organization’s requirements

## Learner Skills and Knowledge

To fully benefit from this lesson, you must have these prerequisite skills and knowledge:

- Understand the concept of firewall function
- Describe and select firewall technologies based on an organization's requirements

## Outline



Outline

Cisco.com

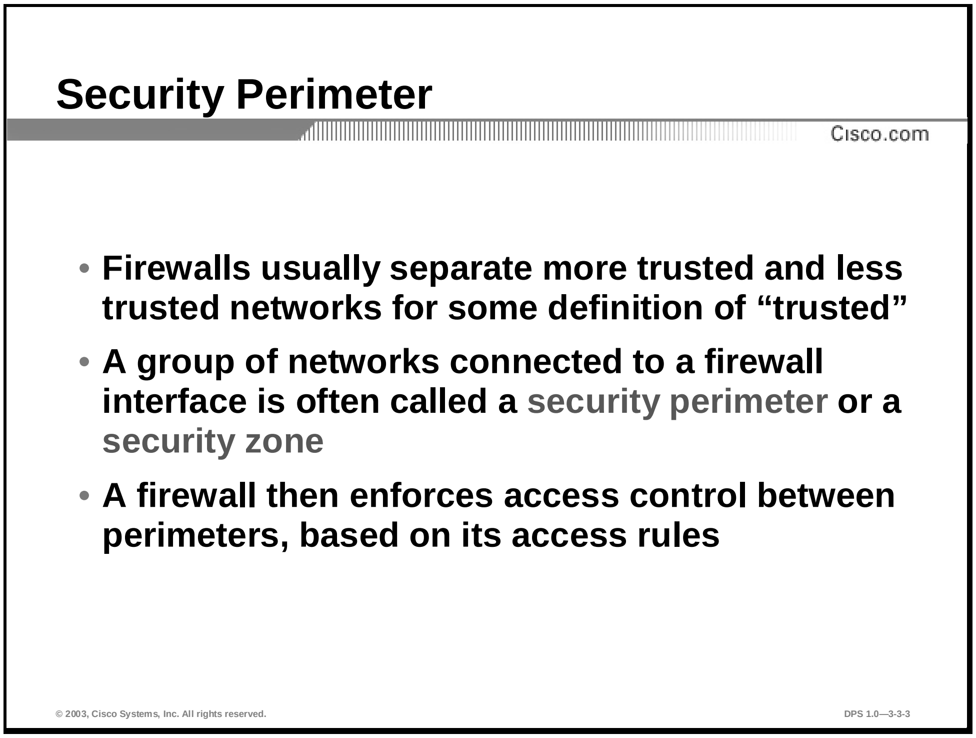
**This lesson includes these sections:**

- **Perimeter Concepts**
- **Screening Router Firewall Architecture**
- **Screened Host Firewall Architecture**
- **Dual-Homed Host Firewall Architecture**
- **Screened Subnet Firewall Architecture**
- **Virtual Firewalls**

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—3-3-2

# Perimeter Concepts



## Security Perimeter

Cisco.com

- Firewalls usually separate more trusted and less trusted networks for some definition of “trusted”
- A group of networks connected to a firewall interface is often called a security perimeter or a security zone
- A firewall then enforces access control between perimeters, based on its access rules

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—3-3-3

## Objective

The section will enable the learner to explain the concept of the security perimeter.

## Introduction

Security perimeters define network areas, between which a firewall will enforce access control. The definition and design of such perimeters is vital in firewall design, if proper levels of separation are to be achieved.

## Definition

Firewalls enforce access control between networks, which can be of different types and levels of trust. A common name for a group of networks reachable over a single firewall network interface, is a **security perimeter** or **security zone**. A perimeter is therefore an administratively separate domain, to or from which a firewall can filter incoming or outgoing connections.

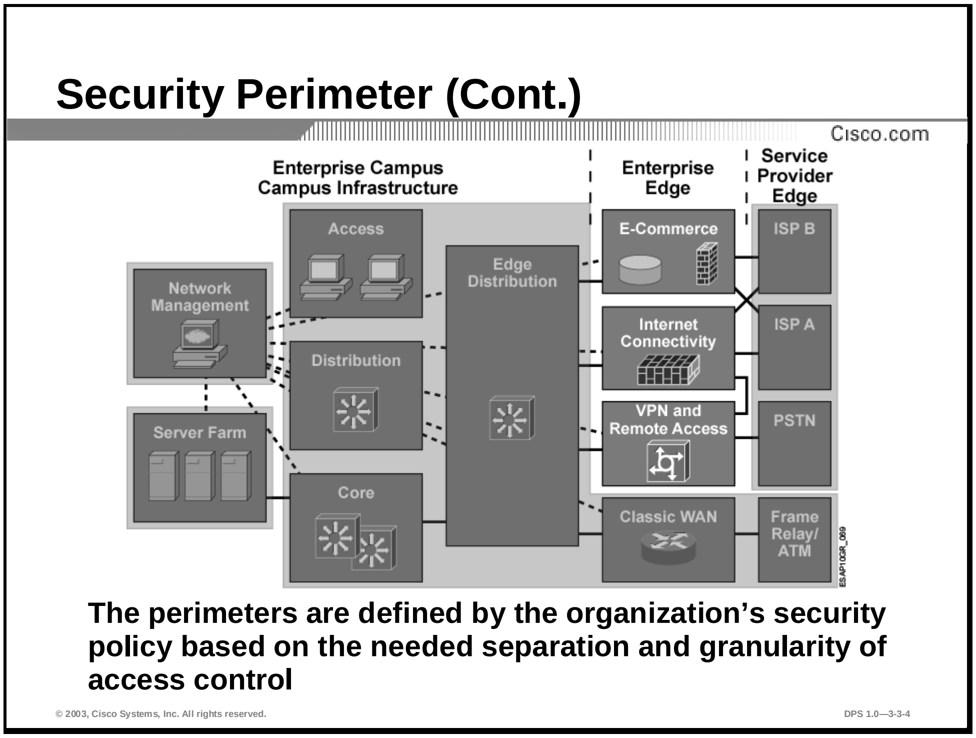
## Examples

When connecting a home network to the Internet, there are two perimeters separated by a home (SOHO) firewall: the outside perimeter, which encompasses all networks of the Internet, and the inside perimeter, containing the trusted home network.

When building a corporate firewall, which connects business partners to an enterprise, the designer can define the perimeters in several ways, depending on the separation requirements. There are two main examples:

- The firewall can simply separate the perimeter of “business partners” from the perimeter of “corporate network”. This approach, however, cannot enforce robust access control between individual business partners, which may be a requirement.

- The firewall can consider each business partner network as a perimeter, the corporate network being the most trusted perimeter. A firewall can therefore connect to each business partner over a dedicated interface, improving its access control capability.



### Security Perimeter (Cont.)

Cisco.com

**The perimeters are defined by the organization’s security policy based on the needed separation and granularity of access control**

DPS 1.0—3-3-4

Perimeters are usually defined based on an organization’s policy requirements: the required granularity of access control and the level of separation between networks. Besides access control policies, several defense-in-depth techniques in firewall design rely on the creation of new perimeters to improve overall firewall robustness.

## Example

A good practice in firewall design is to host exposed services on small perimeter networks. This allows them to be well contained them and limit damage in the case of a break in. This is a simple and effective method for providing defense in-depth for modern multi-tiered applications.

## Practice

Q1) Which of the following cannot belong to a security perimeter of a firewall?

A) the Internet

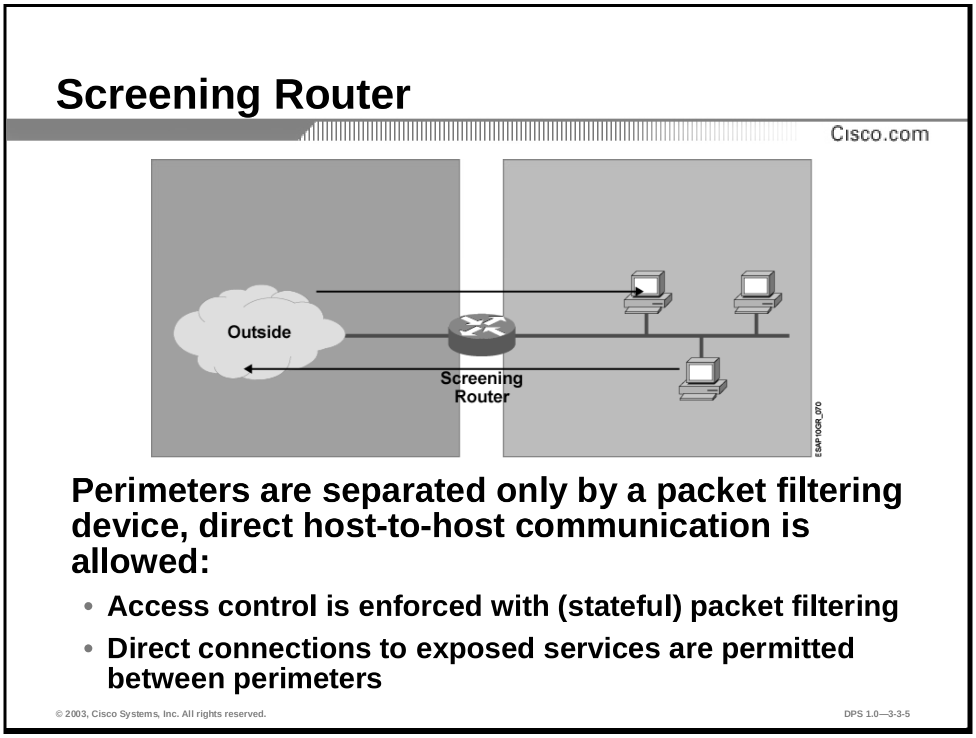
B) a router

C) the firewall itself

D) the whole inside enterprise network

E) another firewall

# Screening Router Firewall Architecture



## Screening Router

**Perimeters are separated only by a packet filtering device, direct host-to-host communication is allowed:**

- **Access control is enforced with (stateful) packet filtering**
- **Direct connections to exposed services are permitted between perimeters**

© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—3-3-5

## Objective

The section will enable the learner to explain the security properties, features, and limitations of a screening router architecture used as firewall architecture.

## Introduction

The screening router architecture is the most basic architecture used in firewall design, and one of the most simple to understand. It is still heavily used when simple internetwork connectivity is desired.

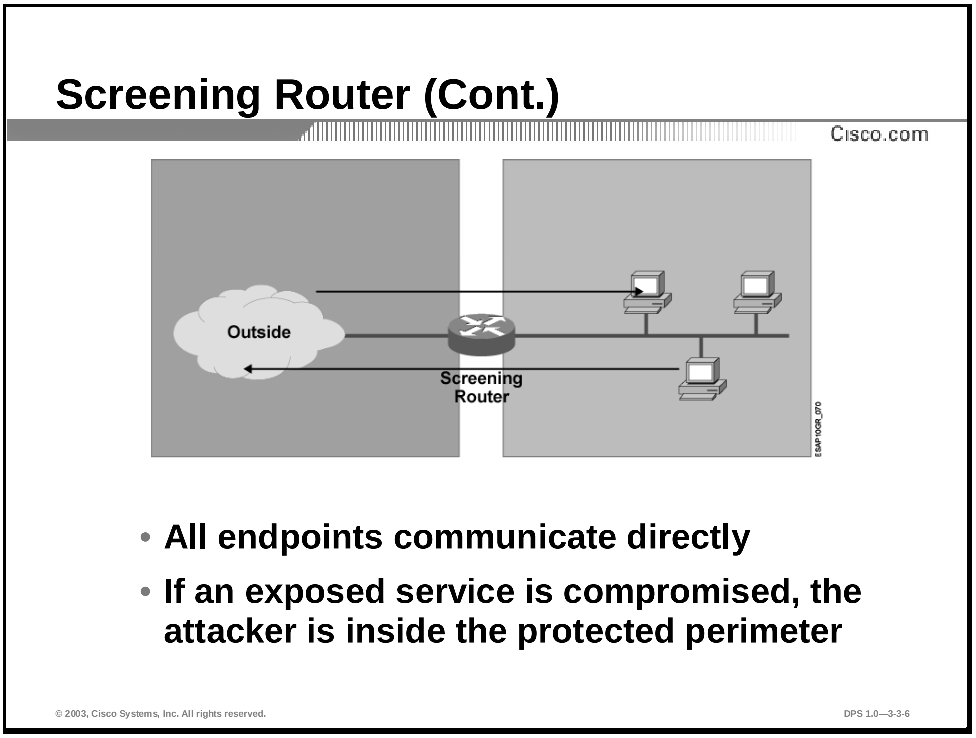
## Definition

The simplest of all firewall architectures is the screening router. In the screening router architecture, a single packet filtering device is located between networks, enforcing access control. The device can be a:

- **Classic (stateless) packet filtering router:** Uses access lists to control the flow of traffic between connected networks

- **Stateful packet filtering device (an enhanced router or a dedicated SPF appliance):** Uses application-aware packet filtering to control the flow of traffic between connected networks

The packet filter permits any exposed services and terminates on an exposed host in the protected network.



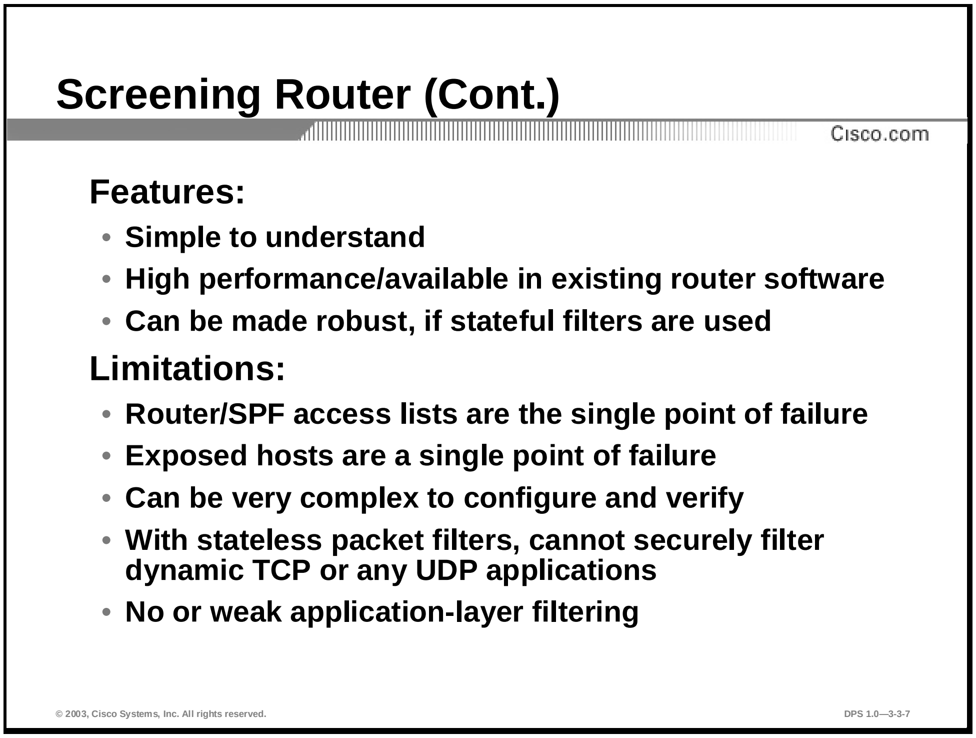
## Screening Router (Cont.)

- **All endpoints communicate directly**
- **If an exposed service is compromised, the attacker is inside the protected perimeter**

DPS 1.0—3-3-6

## Example

This figure illustrates an example of the screening router architecture. The organization in the figure is connected to the Internet and to a business partner using routers. It enforces its access control using stateful packet filtering on the Internet firewall device, and classic packet filtering on the device connected to the partner network. In both scenarios, the permitted applications establish connections directly between clients and servers, without any intermediary inside the firewall system.



## Screening Router (Cont.)

Cisco.com

**Features:**

- **Simple to understand**
- **High performance/available in existing router software**
- **Can be made robust, if stateful filters are used**

**Limitations:**

- **Router/SPF access lists are the single point of failure**
- **Exposed hosts are a single point of failure**
- **Can be very complex to configure and verify**
- **With stateless packet filters, cannot securely filter dynamic TCP or any UDP applications**
- **No or weak application-layer filtering**

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—3-3-7

# Features and Limitations

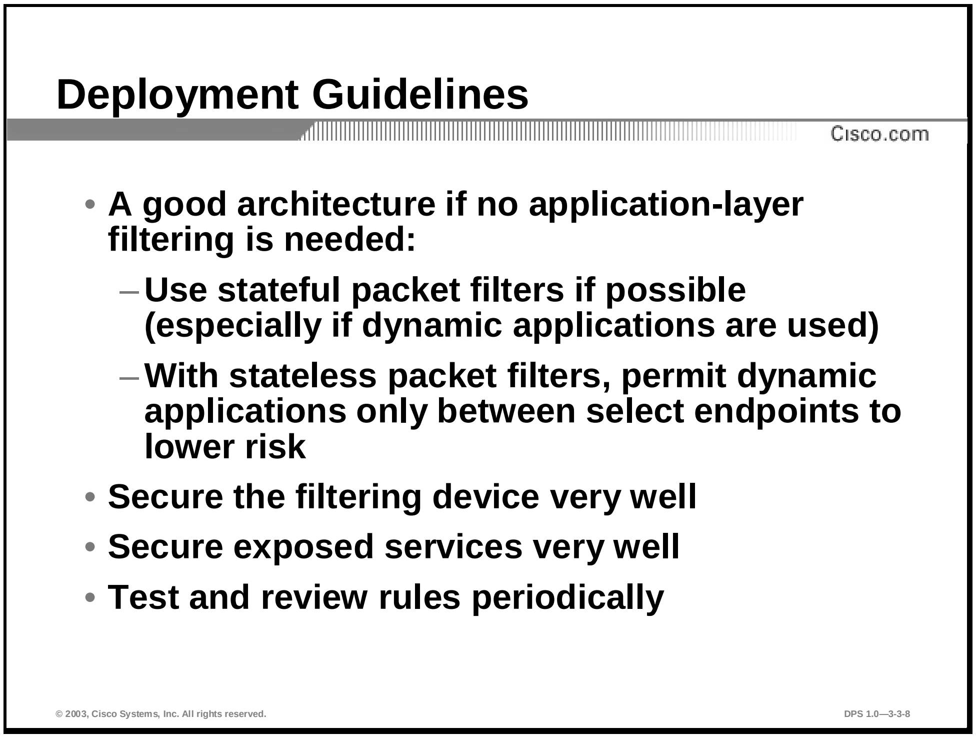
The features of the screening router architecture include:

- Simplicity in design. Multiple perimeters are separated by a single device, which enforces access control.
- High, router-like performance.
- Availability in existing software sets, therefore no upgrade is necessary and the functionality is available anywhere.
- Robust access control, if application aware (stateful) filtering is used.

The limitations of the screening router architecture include:

- The filtering device is a single point of failure, should a bug or misconfiguration of access rules occur.
- Incoming connections from the less trusted, to the more trusted network, terminate directly on hosts inside the most protected network. Compromising an exposed host results in an attacker being on the most protected network, where no further firewalls separate other sensitive resources.
- Extremely complex configuration and management of rules, if stateless packet filtering is used.

- Weak access control, if stateless packet filters (SPFs) are used.
- Only reliably filter up to the transport layer (no application-layer access control).



## Deployment Guidelines

Cisco.com

- A good architecture if no application-layer filtering is needed:
  - Use stateful packet filters if possible (especially if dynamic applications are used)
  - With stateless packet filters, permit dynamic applications only between select endpoints to lower risk
- Secure the filtering device very well
- Secure exposed services very well
- Test and review rules periodically

DPS 1.0—3-3-8

# Guidelines

A screening router is a viable architecture for building a firewall if the policy requirements do not include:

- Application-layer filtering. The screening router enables direct contact between application endpoints

- Dynamic applications, if only stateless (classic) packet filters are available

The screening device (router or special-purpose device) must be resistant against attacks. Use router and generic operating system (OS) hardening techniques to fortify the filtering device against compromise.

Give special attention to the application endpoints because there is no application-layer filtering. If allowing connections inbound from a less trusted to a more trusted perimeter, secure the exposed host/service against application-layer attacks. If an attacker is able to compromise the exposed host, the attacker gains control of a system on the trusted network and can attack other hosts from the compromised host, as there is no firewall inside the trusted perimeter.

The screening device performs all access control between perimeters; therefore it presents a single-point-of-failure from the configuration and software issues' perspective. To address the configuration issues, review and test the access rules periodically to ensure their correct behavior.

Access control is usually performed using address-based rules, and hosts in the protected perimeters might rely on IP addresses to grant or deny access. Therefore, the packet filtering device needs to prevent identity spoofing by filtering out addresses, which should not appear on a particular firewall interface.

## Practice

Q1) What does the performance of a screening router firewall architecture depend on most?

A) the filtering element (screening router) only

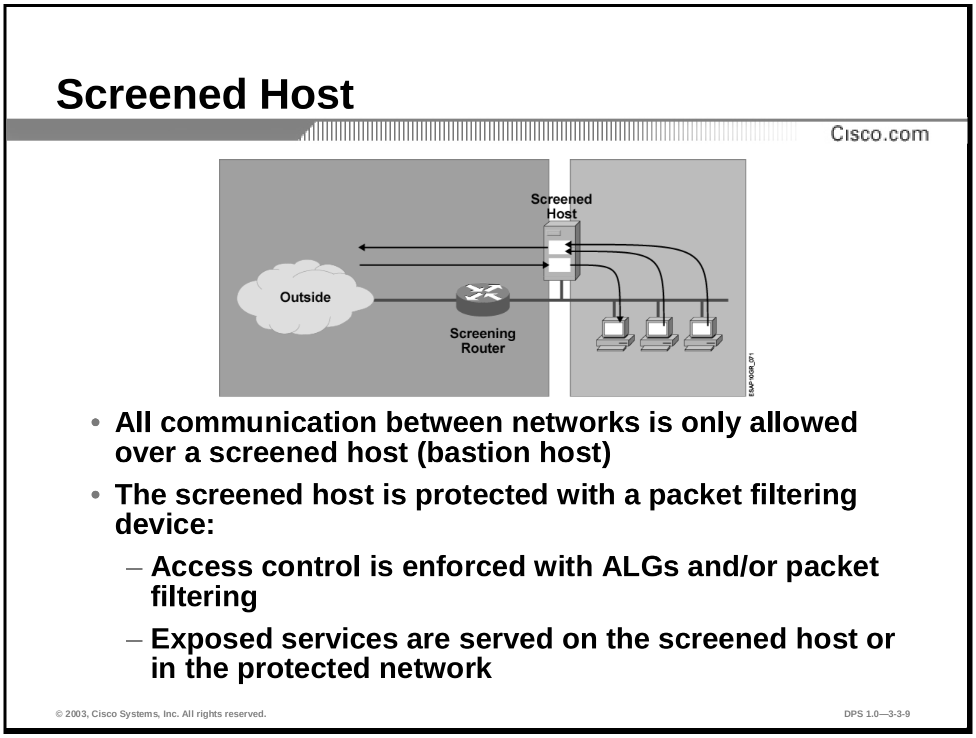
B) application-layer gateways within the firewall system

C) the end hosts

D) the server of the session

E) session-level gateways within the firewall system

# Screened Host Firewall Architecture



## Screened Host

Cisco.com

- All communication between networks is only allowed over a screened host (bastion host)
- The screened host is protected with a packet filtering device:
  - Access control is enforced with ALGs and/or packet filtering
  - Exposed services are served on the screened host or in the protected network

DPS 1.0—3-3-9

## Objective

The section will enable the learner to explain the security properties, features, and limitations of a screened host architecture used as firewall architecture.

## Introduction

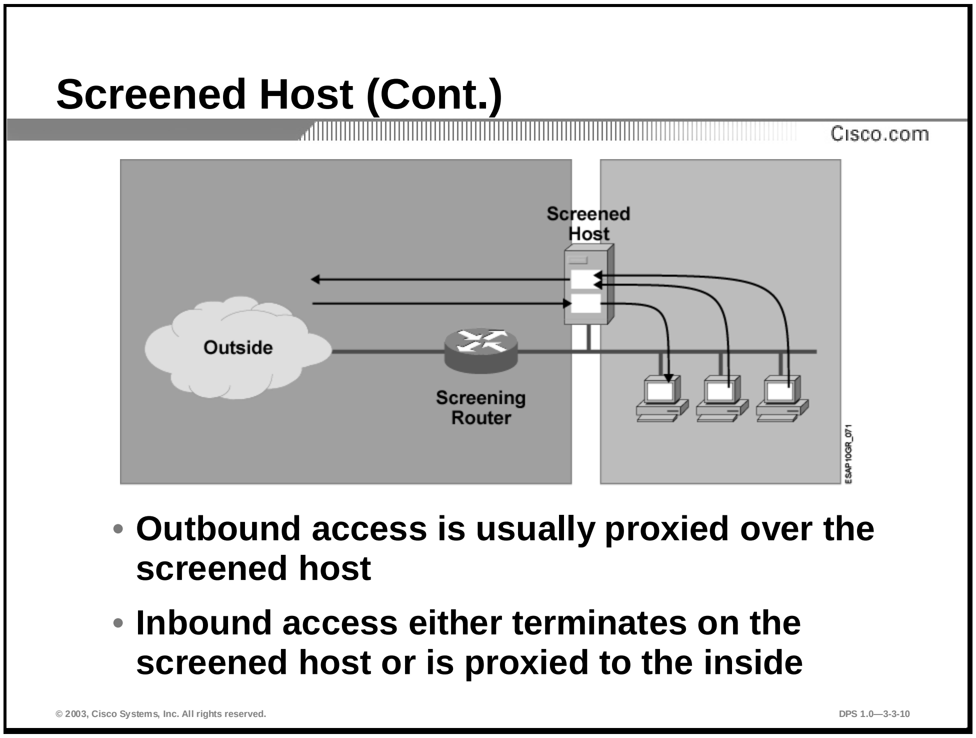
The screened host architecture is a basic architecture used in firewall design. It provides additional functionality compared to the simple screening router architecture, and is still used in some specific simpler connectivity scenarios.

## Definition

Compared to screening router architecture, the screened host architecture adds the concept of a bastion host (or multiple bastion hosts), which serve as the only reachable systems in the protected network.

A packet-filtering device protects (screens) the bastion host because it only allows the necessary access to and from it. This approach simplifies access control on the packet-filtering device because it allows access to and from only one host, and enables application-layer filtering using the bastion host’s application proxies.

Public services can be served on the bastion host(s) or on other inside systems, with a bastion host proxying requests to the inside, using application-layer gateway (ALG) technology.



## Screened Host (Cont.)

Cisco.com

- Outbound access is usually proxied over the screened host
- Inbound access either terminates on the screened host or is proxied to the inside

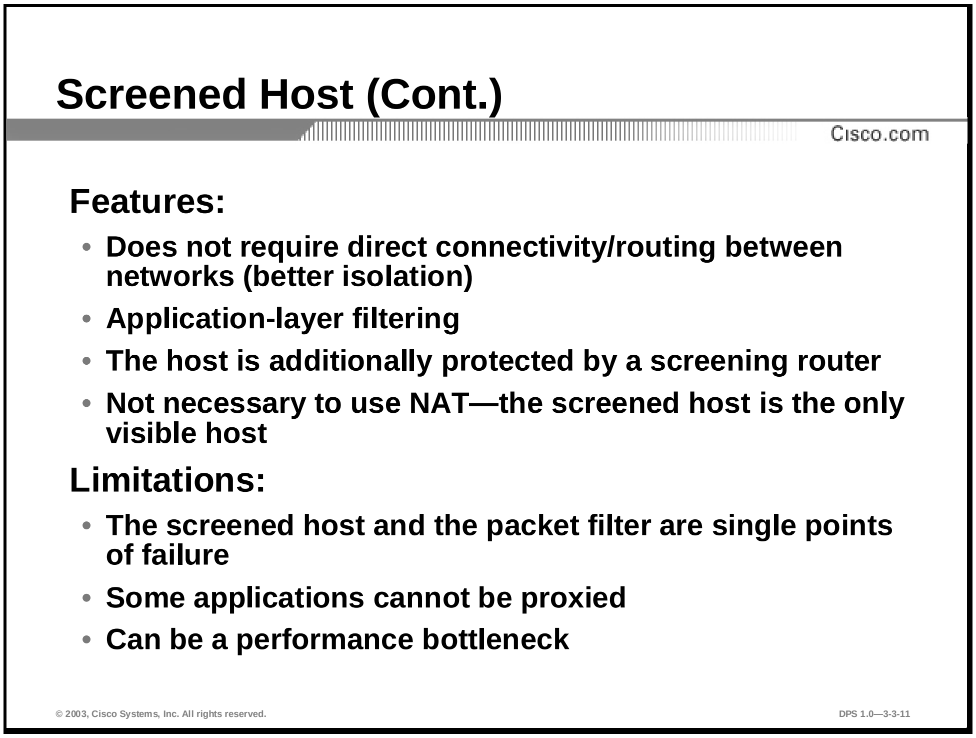
DPS 1.0—3-3-10

## Example

This figure illustrates a typical data flow in classic screened host architecture:

- Outbound connections are only allowed over application proxies on the bastion host(s), as the bastion host is the only point of contact between the protected perimeter and the outside network

- Inbound connections to exposed services either terminate on a bastion host or are relayed to the inside network via application proxies on the bastion hosts



## Screened Host (Cont.)

Cisco.com

**Features:**

- **Does not require direct connectivity/routing between networks (better isolation)**
- **Application-layer filtering**
- **The host is additionally protected by a screening router**
- **Not necessary to use NAT—the screened host is the only visible host**

**Limitations:**

- **The screened host and the packet filter are single points of failure**
- **Some applications cannot be proxied**
- **Can be a performance bottleneck**

DPS 1.0—3-3-11

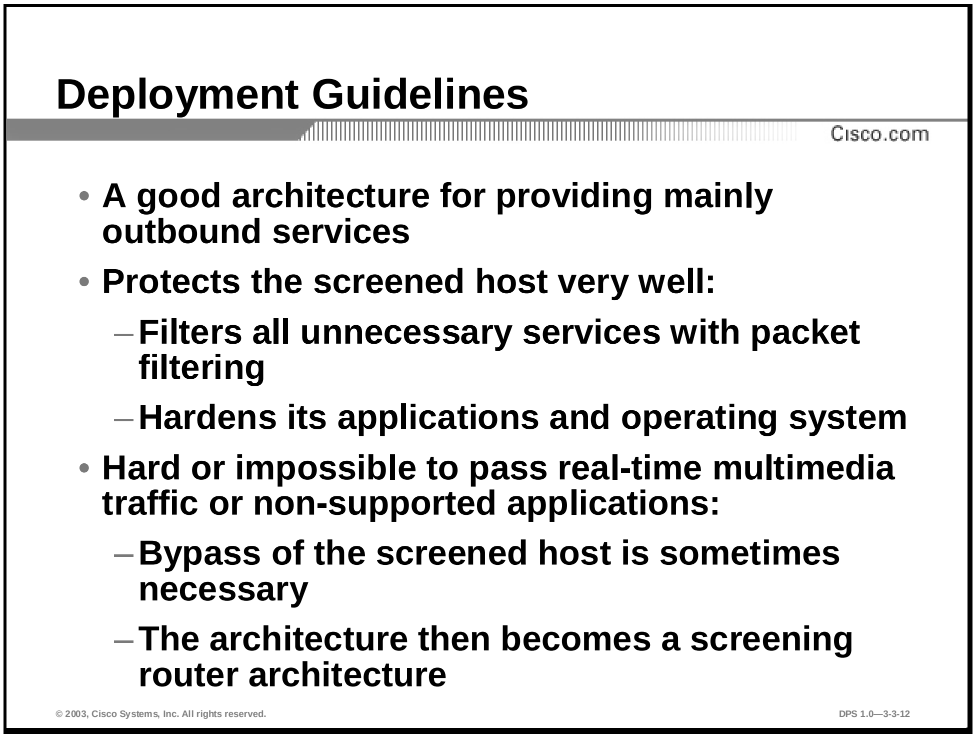
# Features and Limitations

The features of the screening host architecture include:

- Better isolation of networks. Because the screened host handles all communication, there is no need for direct routing between the protected network and the untrusted network.
- Availability of application-layer filtering on the bastion host's application proxies.
- A screening router additionally protects the host, decreasing its exposure.
- As the bastion host is the only reachable host, it can have a globally unique IP address, therefore Network Address Translation (NAT) can be avoided.

The limitations of the screening router architecture include:

- In security terms, the exposed host is a single-point-of-failure. As it is exposed it can be broken into, placing an attacker squarely in the trusted network.
- It relies on ALG technology to send requests outside. Applications that cannot be proxied are either patched via TCP/UDP relays, or can BYPASS the bastion host. They are then packet filtered on the router, which diminishes the added security of this architecture.
- Firewall performance depends on the bastion host's performance, which can present a bottleneck.



### Deployment Guidelines

- A good architecture for providing mainly outbound services
- Protects the screened host very well:
  - Filters all unnecessary services with packet filtering
  - Hardens its applications and operating system
- Hard or impossible to pass real-time multimedia traffic or non-supported applications:
  - Bypass of the screened host is sometimes necessary
  - The architecture then becomes a screening router architecture

## Guidelines

The screening host architecture is beneficial when outbound services need to be provided with application filtering. This is provided by ALGs on the screened host.

To stop any unnecessary connection to the host, fortify it inside its OS and applications, and protect it with the screening router. The major limitation of this approach is that all traffic has to pass through the screened host. This limits its usefulness for real-time traffic or applications, for which no ALG exists. A designer can solve this problem by allowing some applications to bypass the screened host and only be handled by the packet filtering router. This mutates this architecture towards the screening router architecture for the bypassing applications.

## Practice

Q1) How are outbound connections handled by the screened host architecture?

A) by an application-layer gateway (ALG) host

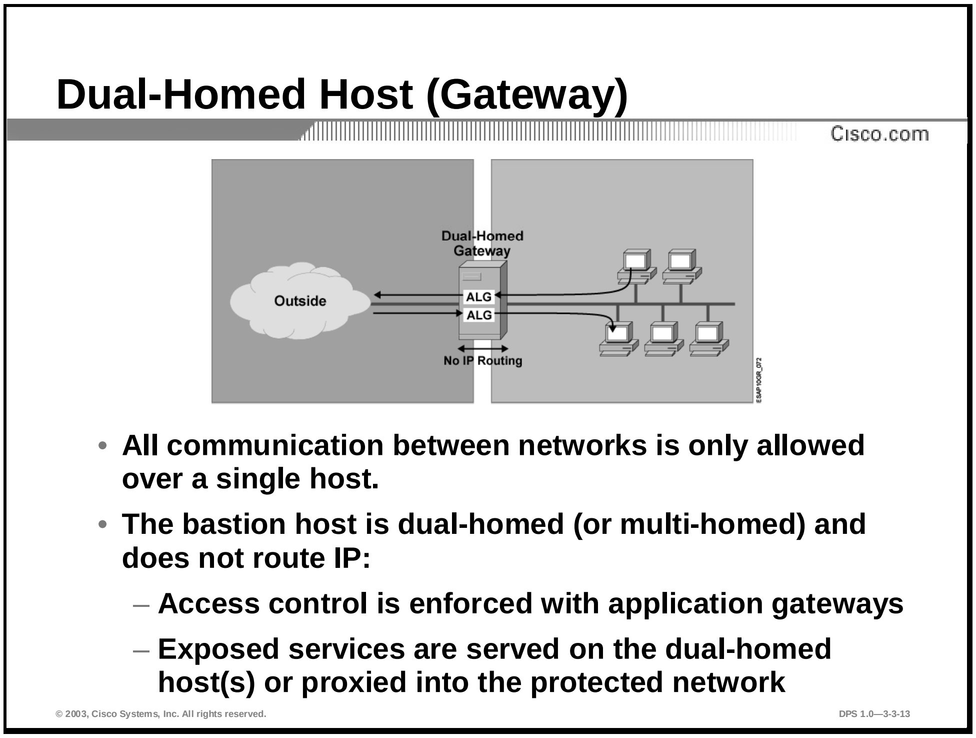
B) by the screening router only

C) by a proxy in a DMZ

D) by the stateful filter only

E) by a session-layer gateway host

# Dual-Homed Host Firewall Architecture



## Dual-Homed Host (Gateway)

Cisco.com

- **All communication between networks is only allowed over a single host.**
- **The bastion host is dual-homed (or multi-homed) and does not route IP:**
  - **Access control is enforced with application gateways**
  - **Exposed services are served on the dual-homed host(s) or proxied into the protected network**

© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—3-3-13

## Objective

The section will enable the learner to explain the security properties, features, and limitations of a screened host architecture used as firewall architecture.

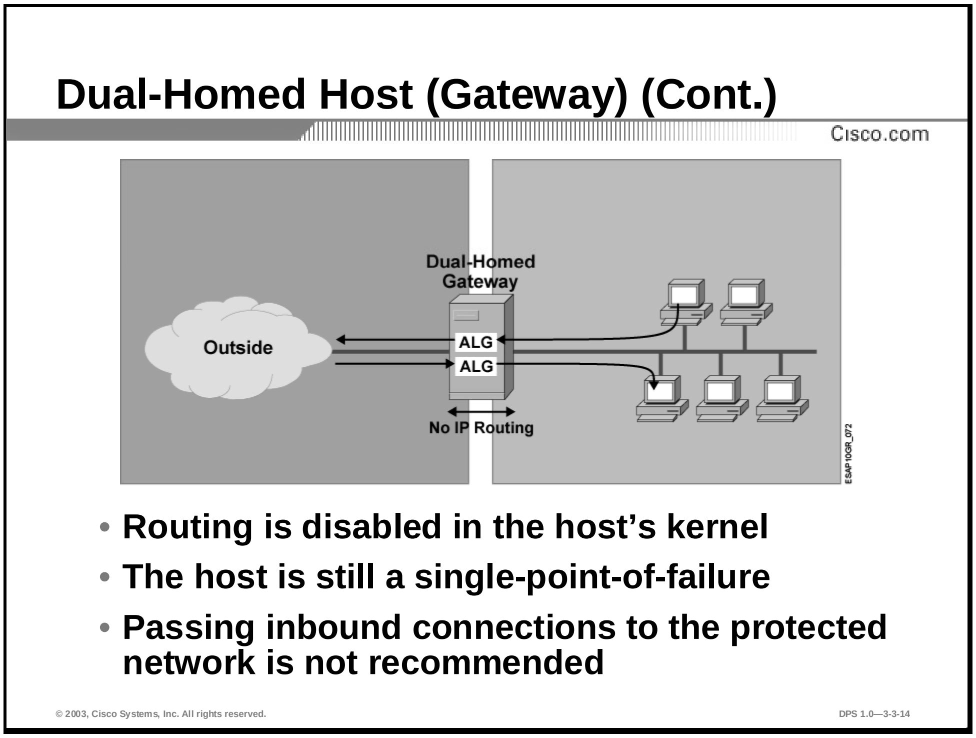
## Introduction

The dual-homed host architecture is another basic architecture used in firewall design. It provides additional separation between networks and a very granular access control method, which made is suitable in the past to be used in environments with high security requirements.

## Definition

The dual-homed host is a popular alternative to the screened host architecture. The dual-homed host does not rely on a router to limit communication over a single host, but uses a host multi-homed (usually dual-homed) to multiple perimeters. The host does not route IP, but terminates sessions from any of the connected perimeters. This ensures that no traffic can pass the firewall without first connecting to its application-layer services (ALGs).

This architecture might still employ a router to protect the dual-homed host against some attacks, but does not require it. Some products contain packet filtering functionality within the dual-homed host's kernel, eliminating the need for an outside router for additional protection.



## Dual-Homed Host (Gateway) (Cont.)

- Routing is disabled in the host's kernel
- The host is still a single-point-of-failure
- Passing inbound connections to the protected network is not recommended

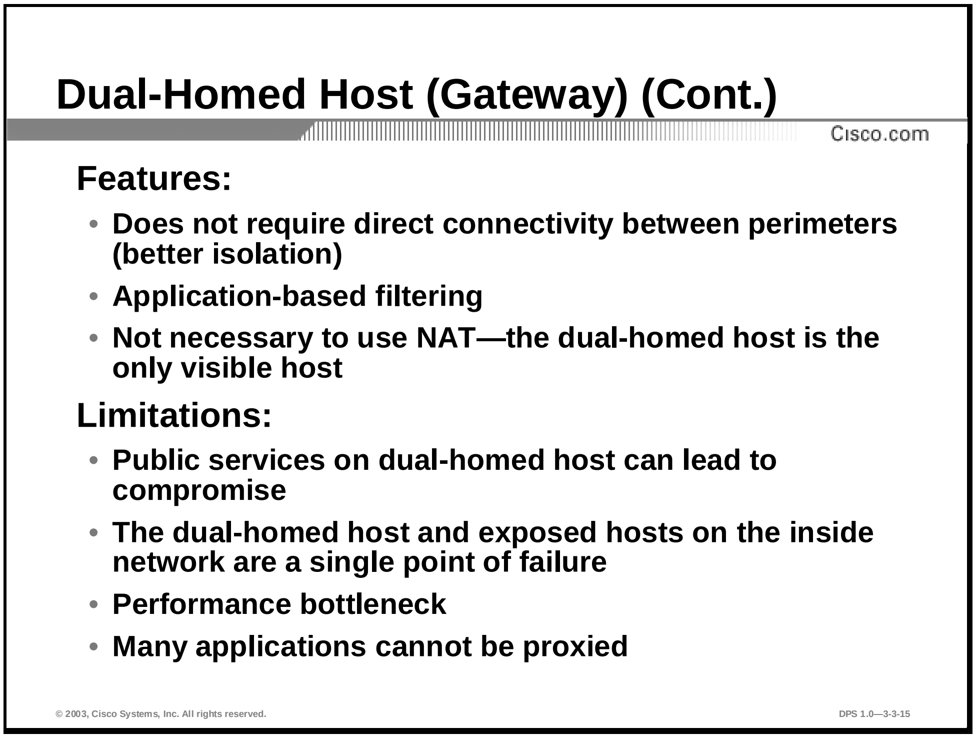
DPS 1.0—3-3-14

## Example

This figure illustrates typical data flow in classic dual-homed host architecture:

- Outbound connections are only allowed over application proxies on the dual-homed gateway
- Inbound connections to exposed services either terminate on a bastion host, or are relayed to the inside network via application proxies on the bastion hosts

**Note** Passing inbound connections to the inside network can result in an inside host being compromised, and an attacker immediately entering the secure perimeter.



## Dual-Homed Host (Gateway) (Cont.)

**Features:**

- **Does not require direct connectivity between perimeters (better isolation)**
- **Application-based filtering**
- **Not necessary to use NAT—the dual-homed host is the only visible host**

**Limitations:**

- **Public services on dual-homed host can lead to compromise**
- **The dual-homed host and exposed hosts on the inside network are a single point of failure**
- **Performance bottleneck**
- **Many applications cannot be proxied**

DPS 1.0—3-3-15

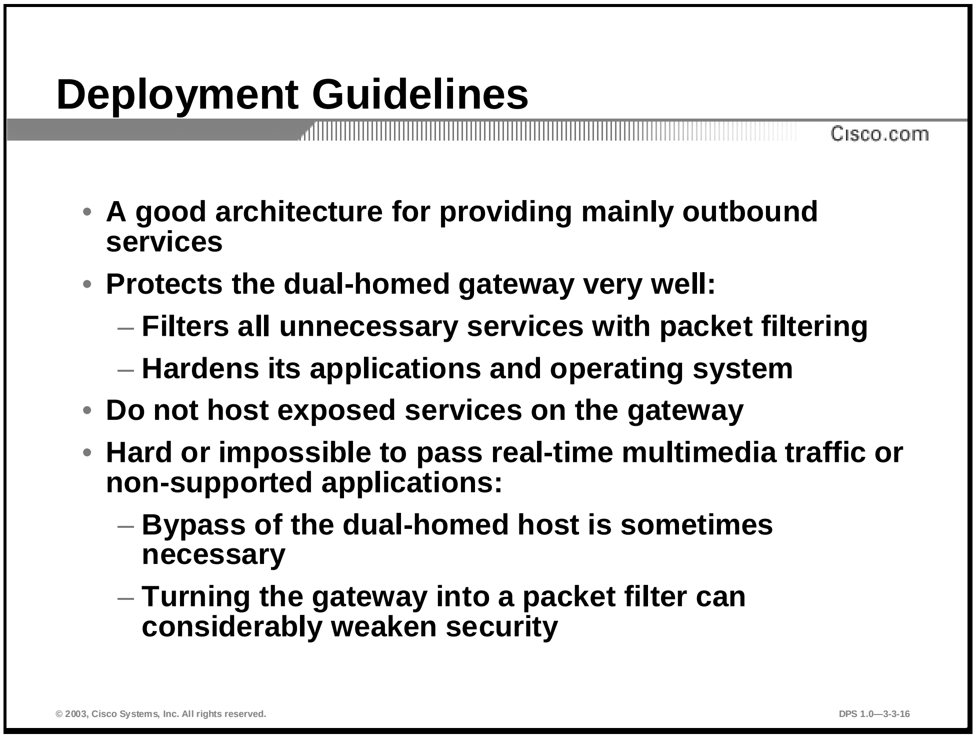
## Features and Limitations

The features of the screening host architecture include:

- Better isolation of networks. As the screened host handles all communication, there is no need for direct routing between the protected network and the untrusted network.
- Availability of application-layer filtering on the bastion host's application proxies.
- As the bastion host is the only reachable host, it can have a globally unique IP address, therefore NAT can be avoided.

The limitations of the screening router architecture include:

- Hosting public services on the dual-homed host increases the complexity and vulnerability of the dual-homed host.
- The dual-homed host, and any exposed host on the inside network to which the dual-homed host relays incoming requests, are single-points-of-failure. As they are exposed they can be broken into, placing an attacker in the trusted network.
- It relies on ALG technology to relay requests between perimeters.
- Firewall performance depends on the dual-homed host's performance, which can present a bottleneck.



## Deployment Guidelines

- A good architecture for providing mainly outbound services
- Protects the dual-homed gateway very well:
  - Filters all unnecessary services with packet filtering
  - Hardens its applications and operating system
- Do not host exposed services on the gateway
- Hard or impossible to pass real-time multimedia traffic or non-supported applications:
  - Bypass of the dual-homed host is sometimes necessary
  - Turning the gateway into a packet filter can considerably weaken security

DPS 1.0—3-3-16

# Guidelines

If an organization requires mainly outbound services, the dual-homed host is a robust architecture. As the security of the firewall relies on the security of the host itself, the dual-homed gateway needs to be hardened, similar to the screened host approach.

Exposed services (such as web, email, or DNS servers supporting inbound connections) are often terminated and served on the host itself. This can be extremely risky, as a compromised exposed application will probably compromise the whole gateway, therefore compromising the firewall. Exposed services can be hosted either in front of the dual-homed host, therefore having no protection from it, or on the inside, where requests to them are relayed through the gateway's ALGs. This is not recommended as it exposes the most secure perimeter to a single-point-of-failure if the internal server is compromised.

# Practice

Q1) How are inbound connections handled by the dual-homed host architecture?

A) by an application-layer gateway (ALG) host

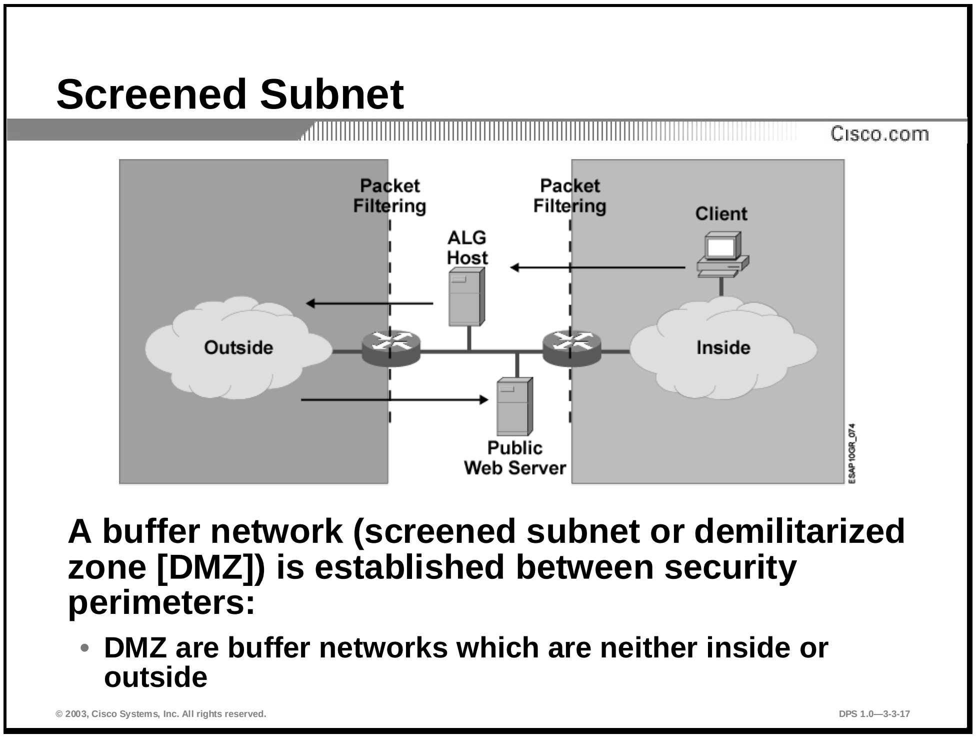
B) by the screening router only

C) by a proxy in a DMZ

D) by the stateful filter only

E) by a session-layer gateway host

# Screened Subnet Firewall Architecture



## Screened Subnet

**A buffer network (screened subnet or demilitarized zone [DMZ]) is established between security perimeters:**

- **DMZ are buffer networks which are neither inside or outside**

DPS 1.0—3-3-17

## Objective

The section will enable the learner to explain the security properties, features, and limitations of a screened subnet architecture used as firewall architecture.

## Introduction

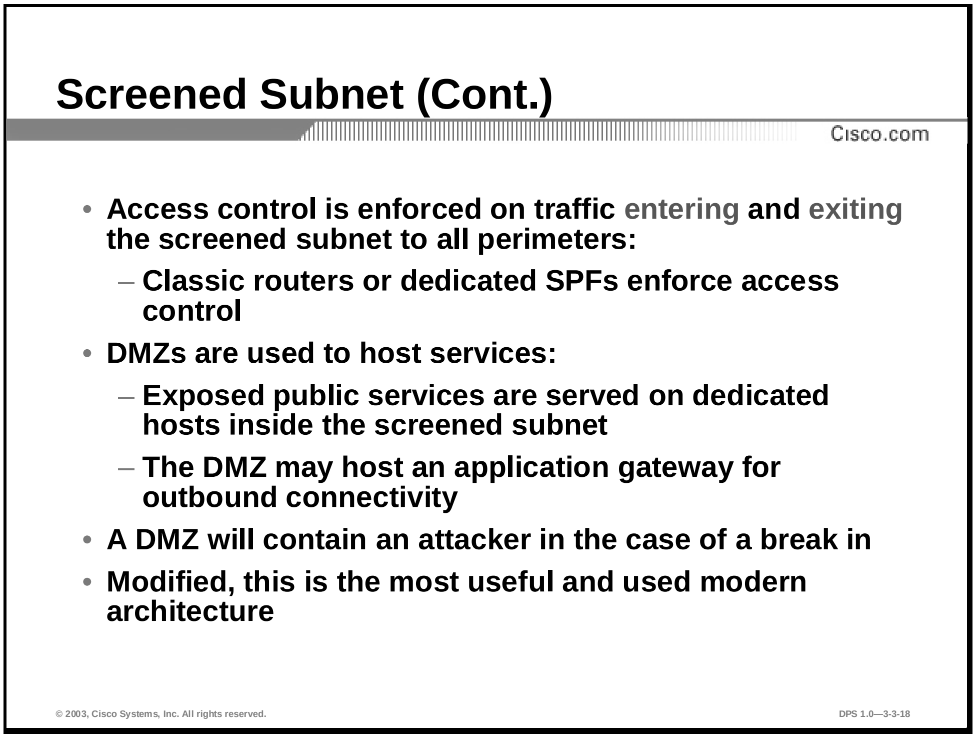
The evolution of firewall architectures provided more assurance in the filtering capability of firewall systems. However, it did little to provide multiple layers of security or to improve the ability to tolerate integrity failures, such as an attacker compromising a device in the firewall system.

## Example

If either the screening router or the screened host is compromised the screened host architecture can fail. By changing the packet filtering rules on the screening router, an attacker can bypass the screened host and enter the private network. By compromising the screened host, an attacker is on the private network and can continue working from the compromised screened host.

## Screened Subnet

In order to provide a layered approach, the idea of the screened subnet was developed. The idea is based on a creation of a “buffer” network, which is situated between perimeters, and actually represents a miniature perimeter itself. This small network, often called the demilitarized zone (DMZ), is neither an inside, nor an outside network. It acts as “no-man’s land”, and access to it is permitted from inside and outside, although no traffic can ever directly cross the DMZ. Filtering points, set up on DMZ edges to connect it to the inside and outside perimeter, enforce access control for traffic entering or exiting the DMZ. These filtering points are usually implemented with classic or stateful packet filters, or a dual-homed ALG host.



## Screened Subnet (Cont.)

- **Access control is enforced on traffic entering and exiting the screened subnet to all perimeters:**
  - **Classic routers or dedicated SPFs enforce access control**
- **DMZs are used to host services:**
  - **Exposed public services are served on dedicated hosts inside the screened subnet**
  - **The DMZ may host an application gateway for outbound connectivity**
- **A DMZ will contain an attacker in the case of a break in**
- **Modified, this is the most useful and used modern architecture**

DPS 1.0—3-3-18

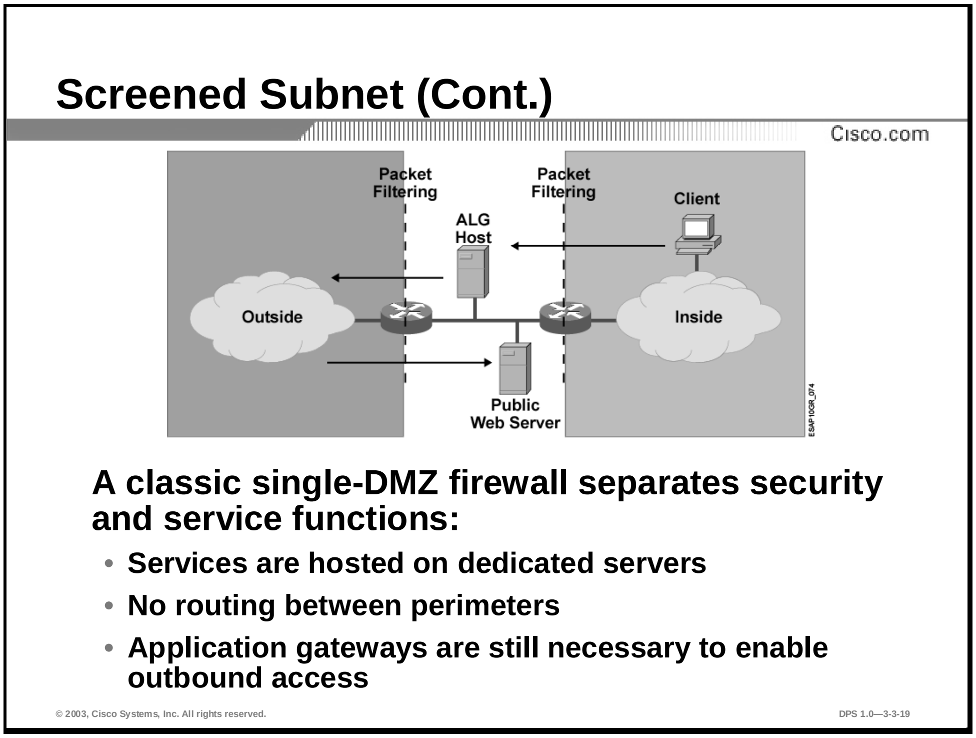
## Screened Subnet Access Control and Service Hosting

The DMZ is an ideal place to host services—either public, exposed servers, which untrusted users connect to, or hosts running ALG software—to enable inside users to connect to the outside perimeter. The DMZ contains the attacker, and the DMZ filtering points limit his action, if either of these hosts or services is compromised

**Note** Because of its ability to contain an attacker, and limit damage in the case of a break in, the screened subnet (DMZ) approach is the most popular and commonly used modern architecture.

The multiple layers of security a DMZ offers are distributed between the services and filtering points:

- The filtering points initially protect the services and, if the services are compromised, limit an attacker's ability to proceed further into the system
- The services are hardened, making it hard for an attacker to compromise them



## Screened Subnet (Cont.)

**A classic single-DMZ firewall separates security and service functions:**

- **Services are hosted on dedicated servers**
- **No routing between perimeters**
- **Application gateways are still necessary to enable outbound access**

DPS 1.0—3-3-19

This figure illustrates a traditional approach to building a screened-subnet firewall. Two routers, used as packet filtering devices, separate the DMZ network from the outside and inside perimeters. The DMZ network hosts both exposed public servers, and hosts running ALG software, to pass data between perimeters.

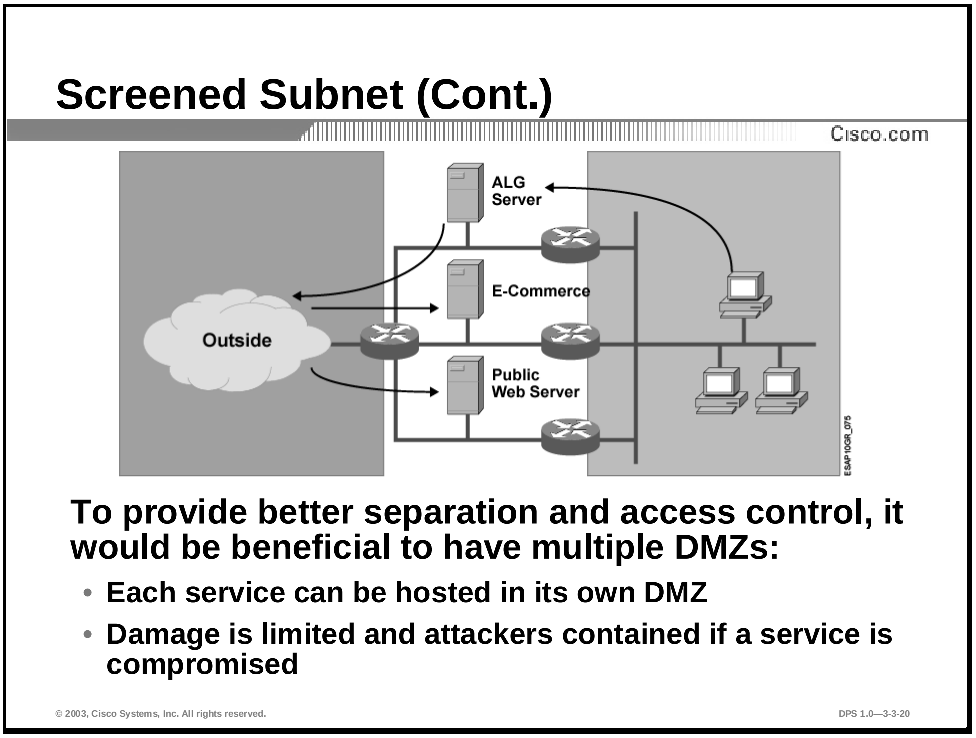
## A Classic Screened Subnet "Breaks" Routing

A notable feature of the classic approach is in its routing design. As no traffic can flow directly between the inside and outside perimeter, no routing needs to be set up to support it. The architecture assumes that all connections will terminate inside the DMZ, therefore only limited routing is needed:

- On the outside router, only a default route to the outside and a directly connected route to the DMZ are needed

- On the inside of the router, only a route to the inside networks and a directly connected route to the DMZ are needed

If all the packet filtering rules on both routers by mistake, an attacker on the outside of the network cannot reach the inside network, as the outside router has no route to it. Conversely, an inside user cannot open a direct connection to the outside perimeter, as the inside router does not have a default route installed. This is an additional layer of security, which augments both packet filtering and ALG-based passing of data between perimeters.



## Screened Subnet (Cont.)

**To provide better separation and access control, it would be beneficial to have multiple DMZs:**

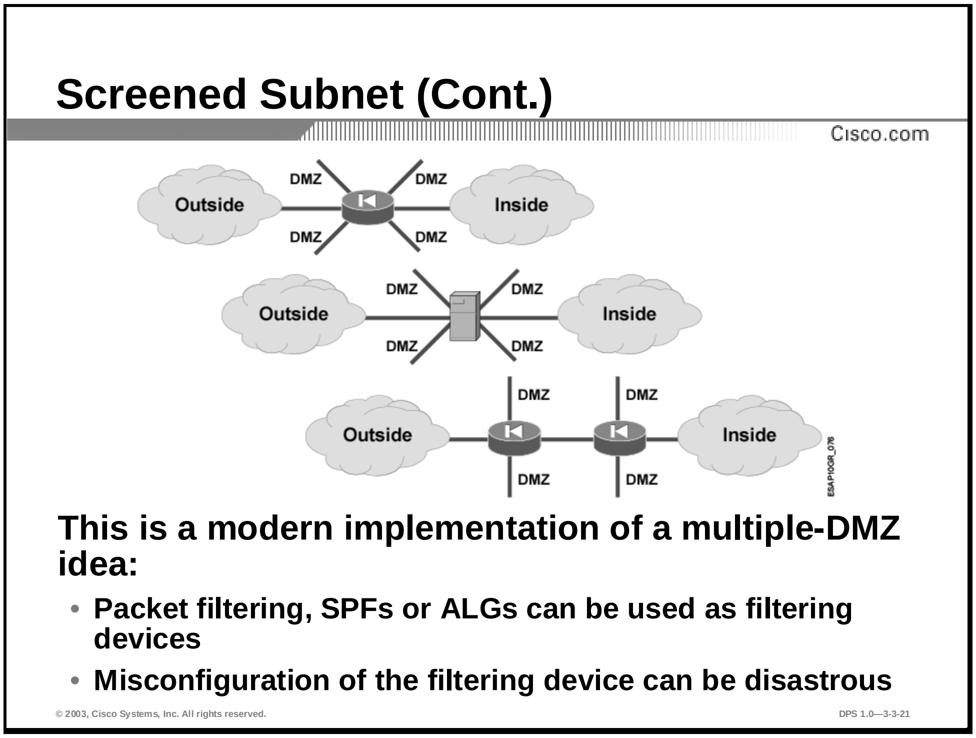
- **Each service can be hosted in its own DMZ**
- **Damage is limited and attackers contained if a service is compromised**

DPS 1.0—3-3-20

## Multiple DMZs

The screened subnet (DMZ), which was introduced to host services and gateways, is a single perimeter, nested between the inside and outside perimeters. There is no access control available to perform access control between hosts inside the DMZ. If a host is broken into, it is likely that other hosts in the same DMZ can be compromised if their operating systems and applications are not properly hardened. For security reasons modern applications are often multi-tiered, and separating the web server from the application server, as well as the database server, is required in a robust system.

This raises the issue of multiple DMZ networks, where each DMZ would host a particular service. This figure illustrates a possible implementation of a multi-DMZ where each new DMZ creates a new perimeter, with filtering points controlling traffic entering and exiting in each single DMZ. A web server can now be isolated from the application server. A compromise of one server will leave an attacker in an extremely restricted environment, with only a few carefully chosen services available, in accordance with the least privilege philosophy. Simplify this approach if possible, because it can introduce additional network elements and configuration complexity.



## Screened Subnet (Cont.)

Cisco.com

**This is a modern implementation of a multiple-DMZ idea:**

- **Packet filtering, SPFs or ALGs can be used as filtering devices**
- **Misconfiguration of the filtering device can be disastrous**

DPS 1.0—3-3-21

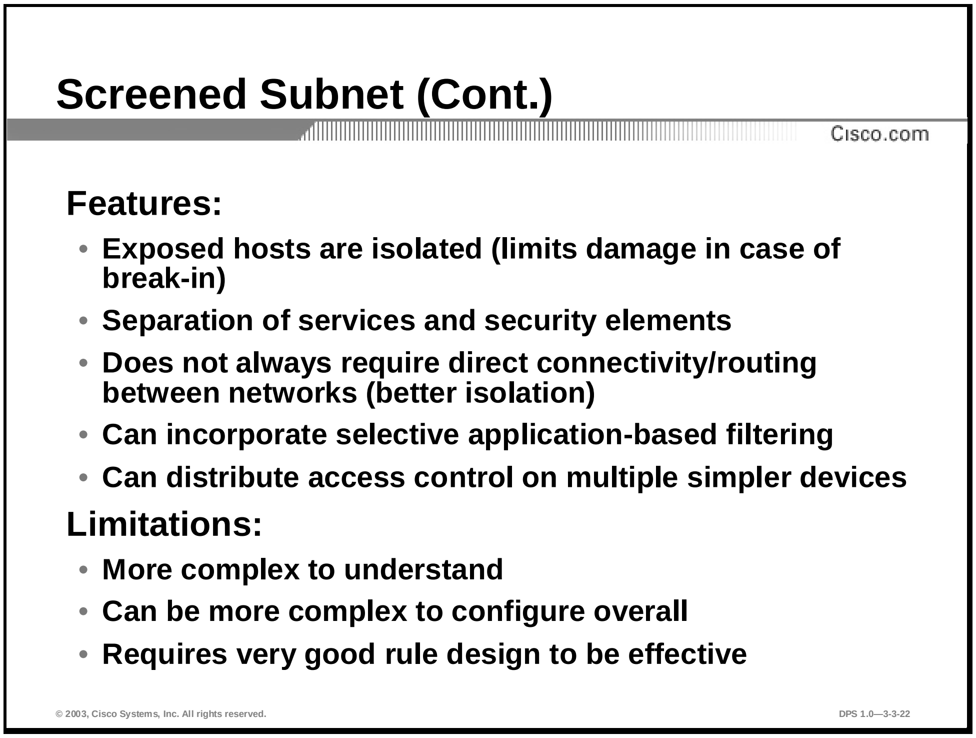
# Modern Architectures with Multiple DMZs

This is a simplified version of the multi-DMZ configuration. A firewall device with multiple “legs” creates multiple DMZs, each “leg network” (a standalone perimeter) being separated from others via the single filtering device. The single device substitutes the “outside” and “inside” routers of a classic DMZ, providing the same level of ingress and egress filtering.

Such a setup has the benefit of being simple, manageable, and very cost effective, although it also has several limitations.

- All traffic to or from a single DMZ, as well as the cumulative traffic of all the DMZs, cross the same device. The device must provide enough bandwidth to satisfy application requirements for expected traffic patters.

- The filtering device is a single-point-of-failure. If misconfigured, access control can break down with disastrous consequences. However, services such as public servers are extremely well-contained and very strict access control can be configured for every perimeter network.



## Screened Subnet (Cont.)

Cisco.com

**Features:**

- **Exposed hosts are isolated (limits damage in case of break-in)**
- **Separation of services and security elements**
- **Does not always require direct connectivity/routing between networks (better isolation)**
- **Can incorporate selective application-based filtering**
- **Can distribute access control on multiple simpler devices**

**Limitations:**

- **More complex to understand**
- **Can be more complex to configure overall**
- **Requires very good rule design to be effective**

DPS 1.0—3-3-22

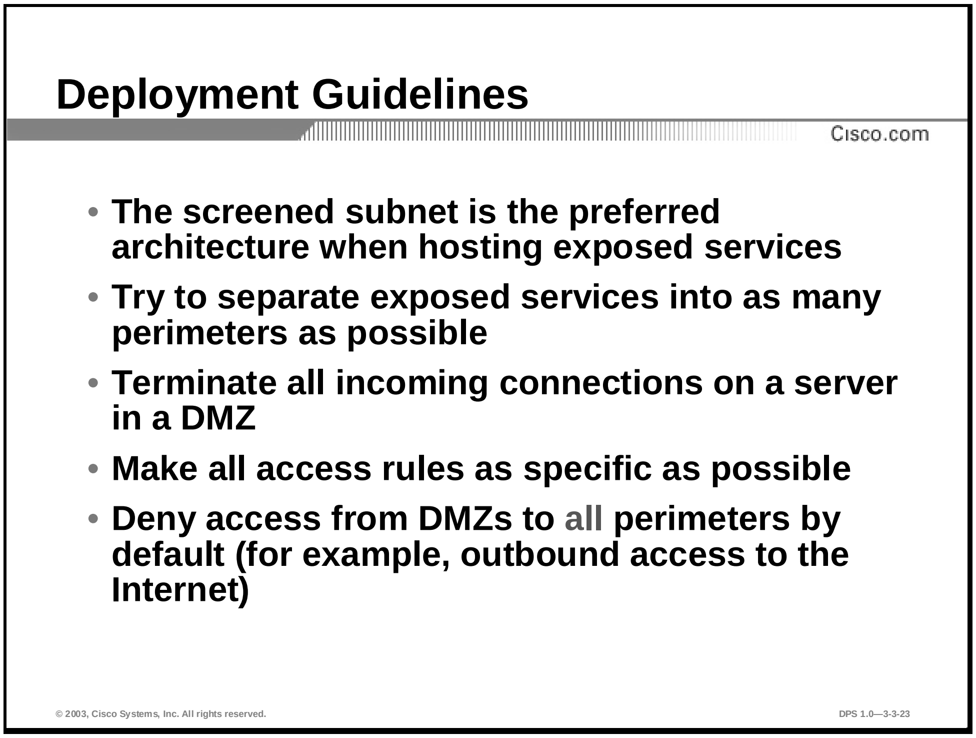
## Features and Limitations

The features of the screened subnet (DMZ) architecture are:

- Isolation of services (exposed hosts) to limit damage in the case of a break-in.
- Separation of services and security. Exposed servers inside the DMZ can host exposed services. While the filtering devices and perhaps an ALG host, provide security filtering.
- Minimal routing can be used to provide an additional layer of protection.
- Application-based filtering can be enabled selectively using an ALG host. Some non-proxiable traffic may bypass the host and be directly exchanged between the perimeters, if the policy allows it.
- Distribution of access control on multiple devices. The devices can back each other up and have a simpler configuration compared to a single-device system.

The limitations of the screened subnet architecture are:

- More complex to understand when compared to simple, two-perimeter systems
- More complex to configure, as multiple devices are used
- Requires a very good design of access rules in the filtering devices to provide multiple layers of protection and robust filtering between perimeters



## Deployment Guidelines

- The screened subnet is the preferred architecture when hosting exposed services
- Try to separate exposed services into as many perimeters as possible
- Terminate all incoming connections on a server in a DMZ
- Make all access rules as specific as possible
- Deny access from DMZs to all perimeters by default (for example, outbound access to the Internet)

© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—3-3-23

## Guidelines

Use the screened subnet architecture as a preferred architecture when public services are deployed using exposed hosts. Use the multiple-DMZ idea to separate different exposed services into multiple DMZs. Use the firewall filters to control access among those DMZs in a very granular fashion. This very effectively limits damage if an incident occurs, and can considerably slow down or stop an attacker from penetrating the network further.

Ideally, all incoming connections from untrusted networks should terminate on an exposed host within a DMZ network. This host usually either serves the request itself, or is an application-gateway, filtering and relaying data to another system.

With such a complex system it is recommended to use:

- An extremely conservative access policy to maximize robustness and control over all flows inside the system
- A default “deny any” stance, with extremely specific rules exactly describing the minimal required access between any two perimeters

## Example

If a system administrator is lenient and allows all access from a DMZ to the Internet, which does not seem dangerous, this can have important implications. An attacker, who compromises a DMZ system might use the compromised system to break into other systems on the Internet. Allowing all outbound access also enables an attacker to download tools from the outside, which might be easier than uploading them through a restrictive inbound filter. Outbound filtering addresses both threats and prevents or resists such exploitation.

## Practice

Q1) What is the main benefit of the screened subnet architecture?

A) it allows hosting of services in a “buffer” network

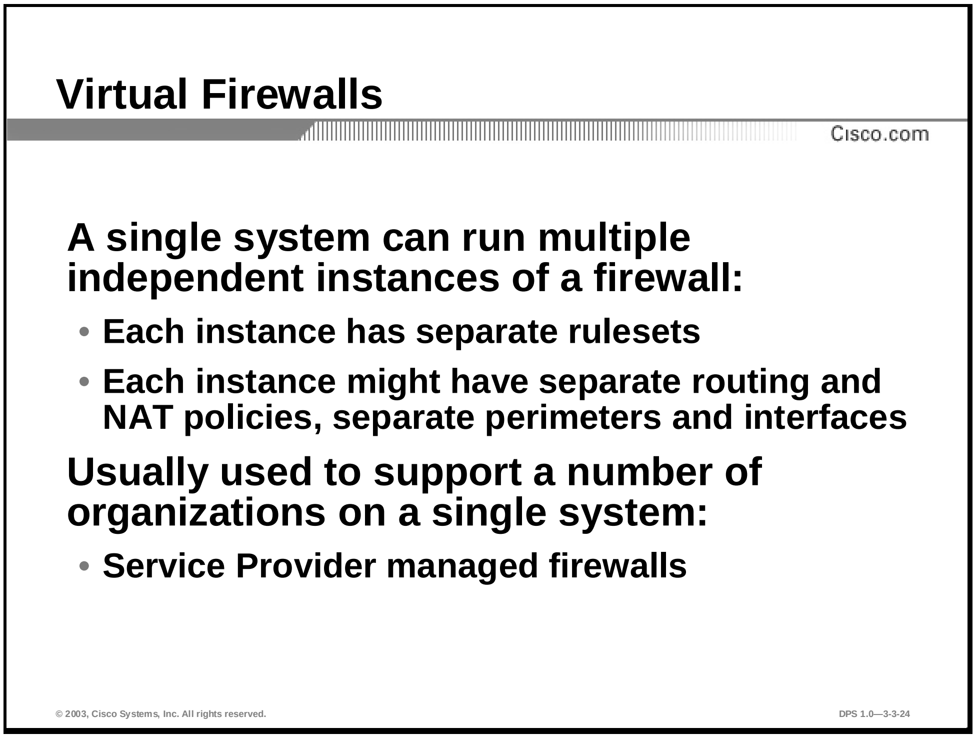
B) it has the highest performance

C) it has the simplest routing configuration

D) it can provide application-layer filtering

E) it is the most transparent for the end users

# Virtual Firewalls



## Virtual Firewalls

A single system can run multiple independent instances of a firewall:

- Each instance has separate rulesets
- Each instance might have separate routing and NAT policies, separate perimeters and interfaces

Usually used to support a number of organizations on a single system:

- Service Provider managed firewalls

© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—3-3-24

## Objective

The section will enable the learner to explain the concept of virtual firewalling.

## Introduction

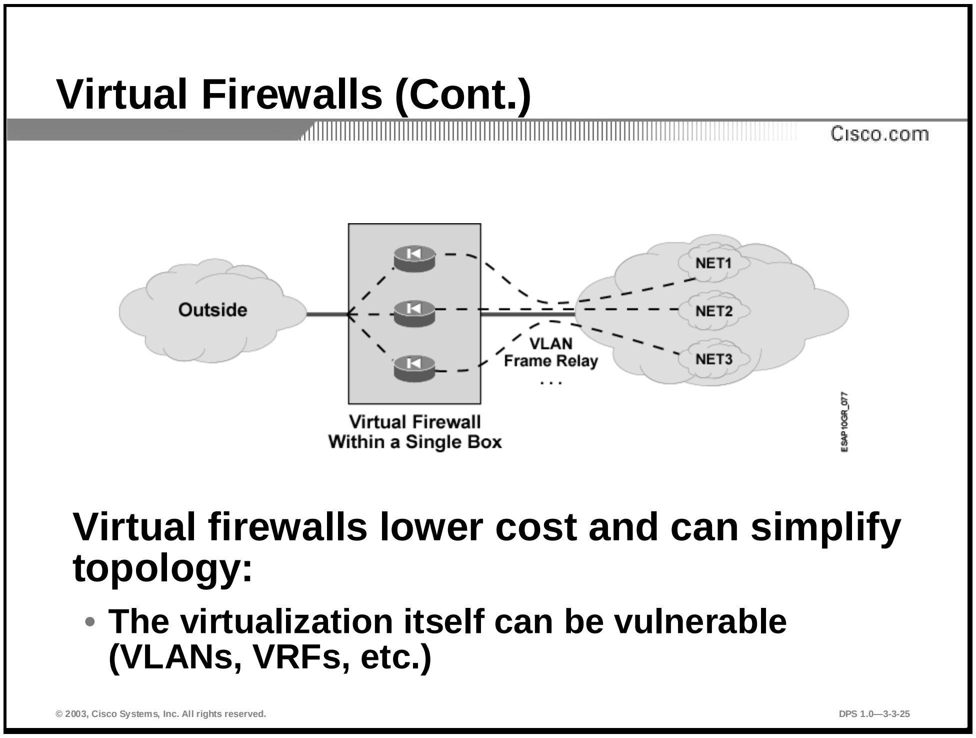
Deploying multiple firewalls for multiple organizations has proven to be difficult for the (security) service providers, as the cost of such firewalls is high, and their flexibility and manageability can be problematic. Virtual firewalls were created to address those limitations and provide for large-scale deployment of firewalling in such scenarios.

## Definition

Virtual firewalls are not a specific architecture, but more of an extension to existing filtering devices. A filtering device is by default under a single administrative domain, filtering between perimeters according to the organization’s policy.

Virtual firewalls introduce multiple firewall instances inside one device. Each firewall instance might have its own interfaces, perimeter definitions, and access control policy. They may also separate routing and NAT information. Such a system creates the illusion of multiple firewalls, each connected to its own set of perimeters.

Virtual firewalls are normally used to support a number of organizations using a single firewall system employing virtualization. This is especially cost-effective in managed firewall solutions, offered by service providers to customers, who do not have a need or the means to run their own firewall system.



## Virtual Firewalls (Cont.)

**Virtual firewalls lower cost and can simplify topology:**

- **The virtualization itself can be vulnerable (VLANs, VRFs, etc.)**

DPS 1.0—3-3-25

## Features and Limitations

The main features of virtual firewalls are:

- Lower cost of firewalling
- Simplified network topology

The main disadvantage is that the method of virtualization might be vulnerable, leaking data between domains. Virtualization is not only achieved in the firewall engine, but often also on the firewall’s interfaces. These might be connected to a technology, which is less robust in providing traffic separation, such as bad VLAN implementations.

## Practice

Q1) What is the main difference in the filtering capability of virtual firewalls compared to any classic firewalls?

A) the filtering granularity is necessarily lower

B) the filtering granularity is necessarily higher

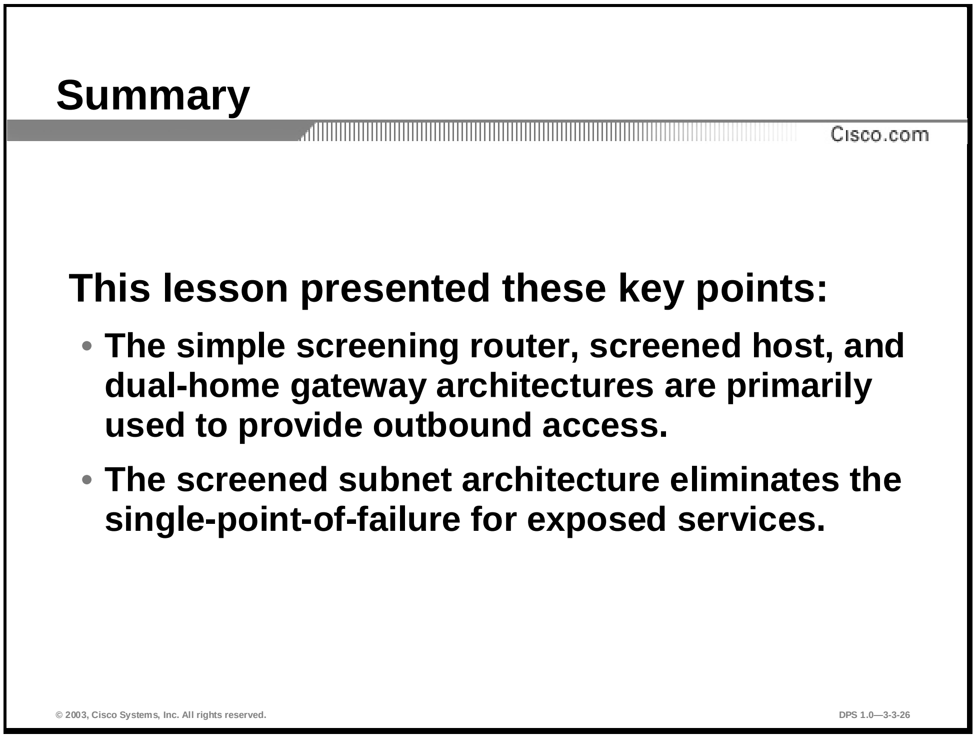
C) there is no difference—the same technologies and architectures are used

D) the filtering impacts performance more significantly

E) virtual firewalls perform virtually no filtering on their own

# Summary

This section summarizes the key points discussed in this lesson.



## Summary

This lesson presented these key points:

- The simple screening router, screened host, and dual-home gateway architectures are primarily used to provide outbound access.
- The screened subnet architecture eliminates the single-point-of-failure for exposed services.

# Next Steps

After completing this lesson, go to:

- Protocol Handling in Firewalls lesson

# Quiz: Firewall Architectures

Complete the quiz to assess what you have learned in this lesson.

## Objectives

This quiz tests your knowledge on how to:

- Compare different basic firewall architectures
- Select the proper architecture for an organization’s requirements

## Instructions

Answer these questions:

1. What is a major security weakness of the screened host approach?
2. How are services, for which no ALG exists, passed over a dual-homed host firewall?
3. How does a multi-tiered application benefit from a screened subnet firewall?

## Scoring

You have successfully completed the quiz for this lesson when you earn a score of 80 percent or better.

# Protocol Handling in Firewalls

## Overview

Firewall systems today handle an unusually high number of diverse applications and their protocols, which have been developed in the Internet boom. Some such protocols are simple, and are passed over firewalls with ease, while others are exceedingly complex and much effort is required to analyze their operation and securely handle in sensitive environments. This lesson presents the protocol handling of many well-known applications by the three main firewall technologies: packet filters, stateful packet filters (SPFs), and application-layer gateways (ALGs).

### Importance

The support and robustness of application protocol handling is of the utmost importance to the firewall designer, when an organization's security requirements need to be met, and faced with a set of application needs by the same organization. Therefore, this lesson contains crucial information needed to strike the best balance between security and functionality in the modern network application world.

### Lesson Objectives

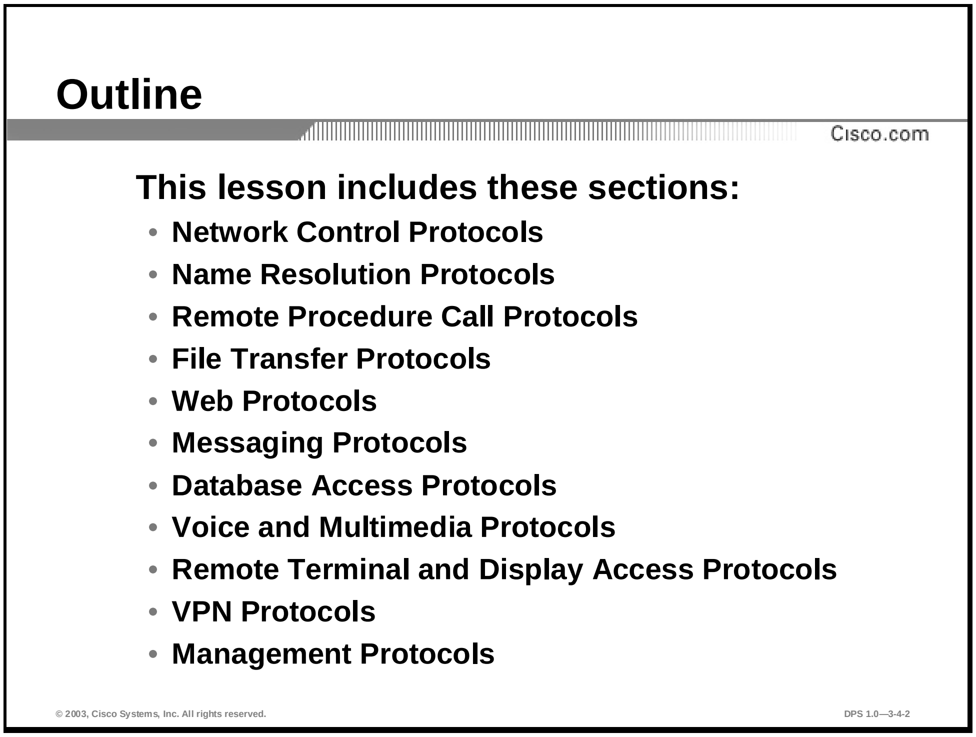
The lesson will enable the learner to select an appropriate firewall technology for an organization's application needs, and provide guidelines to an organization on how to integrate an application with a particular firewall technology.

## Learner Skills and Knowledge

To fully benefit from this lesson, you must have these prerequisite skills and knowledge:

- Understand of the concept of a firewall
- Describe and select appropriate firewall technologies
- Describe and select appropriate firewall architectures

## Outline



### Outline

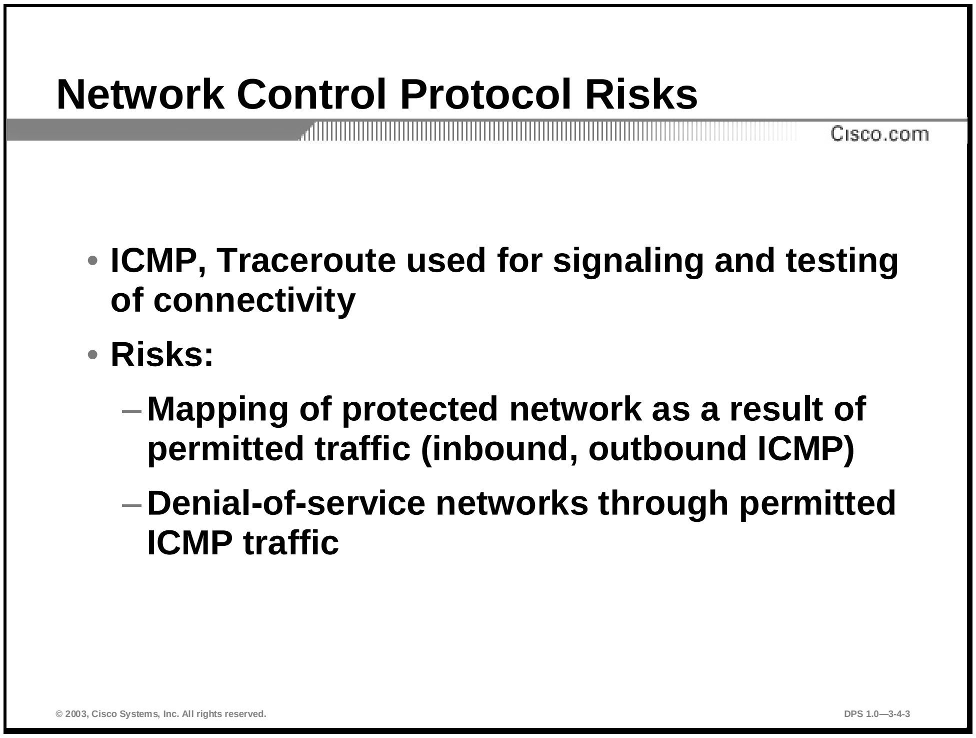
Cisco.com

**This lesson includes these sections:**

- **Network Control Protocols**
- **Name Resolution Protocols**
- **Remote Procedure Call Protocols**
- **File Transfer Protocols**
- **Web Protocols**
- **Messaging Protocols**
- **Database Access Protocols**
- **Voice and Multimedia Protocols**
- **Remote Terminal and Display Access Protocols**
- **VPN Protocols**
- **Management Protocols**

DPS 1.0—3-4-2

# Network Control Protocols



## Network Control Protocol Risks

- ICMP, Traceroute used for signaling and testing of connectivity
- Risks:
  - Mapping of protected network as a result of permitted traffic (inbound, outbound ICMP)
  - Denial-of-service networks through permitted ICMP traffic

DPS 1.0—3-4-3

## Objective

The section will enable the learner to explain the security properties of network control protocols to select an appropriate technology to securely pass them over firewalls.

## Introduction

The network control protocols include the Internet Control Message Protocol (ICMP) and the traceroute application. Both are heavily used in enterprise networks, and conveying them securely over Internet firewalls has been a long-standing issue in the security community.

## ICMP Refresher

IP hosts and routers use the ICMP protocol to provide basic error signaling and notifications, such as:

- Reachability information (echo, echo-reply, unreachable messages)
- Resource quality (source quench messages)
- Information (mask-request, mask-reply, timestamp messages)
- Generic error reporting (parameter problem messages)

Usually, IP hosts do not rely on ICMP information and can, in most cases, operate with ICMP filtered out of the network.

Many network administrators use the traceroute application to provide diagnostics about a path between two endpoints. There are two ways of implementing the conveyance of traceroute across firewalls.

Network endpoints and routers use ICMP to signal network control messages among themselves. Over a firewall, the following ICMP services are often required:

- **PING:** The PING protocol is a simple request-reply protocol, using the ICMP ECHO message as a request, and the ICMP ECHO-REPLY message as a reply from the reachable endpoint.

- **ICMP "Destination Unreachable" Family:** Contains several critical and optional messages, which the IP stack uses to determine possible path problems. The only ICMP "unreachable" message required for an IP stack operation is the "Fragmentation Needed by DF is set" message. Routers along the path between the end systems use this message to inform the end systems about a low maximum transmission unit (MTU), to which they should adjust their packet size. If a firewall blocks this message, the hosts endlessly send large packets, which are discarded by intermediate routers. This results in traffic blackholing.

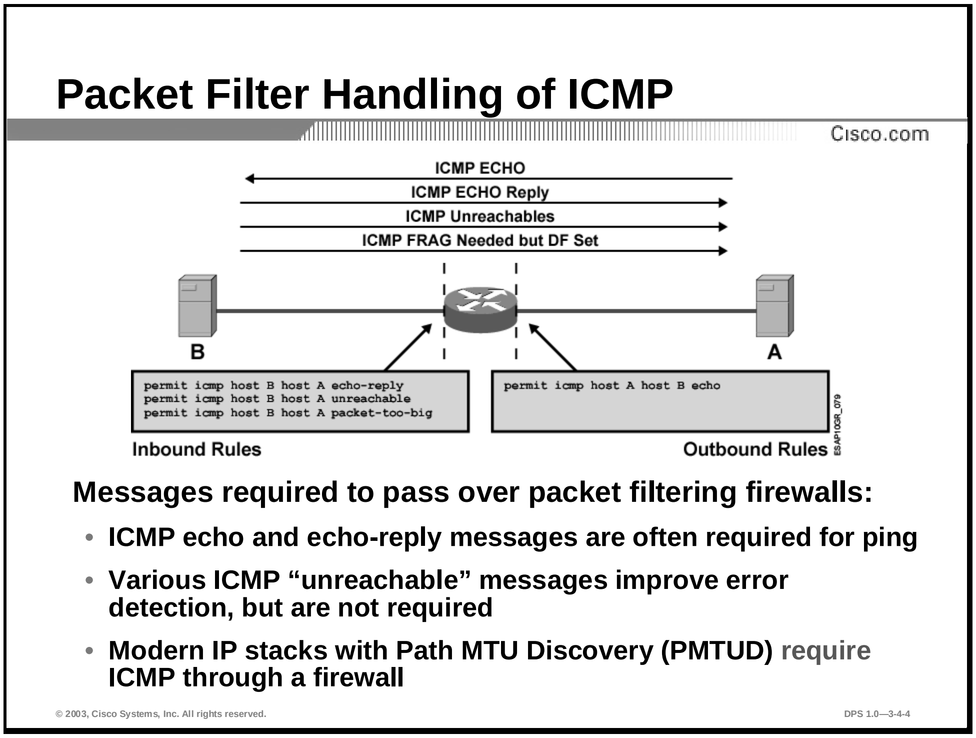
  The following ICMP "Destination Unreachable" messages, which are optional for an IP stack to work, can be filtered by firewalls:

  - ICMP port unreachable

  - ICMP network unreachable

  Other ICMP messages are usually not required for proper IP stack operation and, using the guideline of permitting only the necessary services, are not of concern with firewalling.

- **"Time-to-live (TTL) exceeded" message:** Can be used to allow proper operation of traceroute from the protected network



## Packet Filter Handling of ICMP

Inbound Rules

```
permit icmp host B host A echo-reply
permit icmp host B host A unreachable
permit icmp host B host A packet-too-big
```

Outbound Rules

```
permit icmp host A host B echo
```

**Messages required to pass over packet filtering firewalls:**

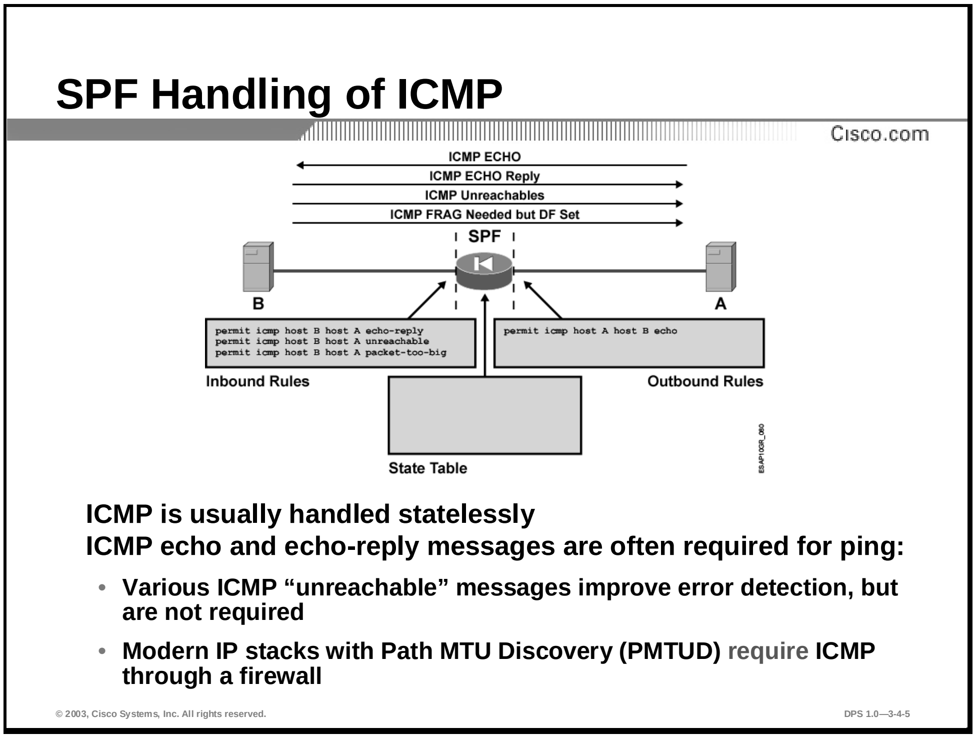
- **ICMP echo and echo-reply messages are often required for ping**
- **Various ICMP "unreachable" messages improve error detection, but are not required**
- **Modern IP stacks with Path MTU Discovery (PMTUD) require ICMP through a firewall**

DPS 1.0—3-4-4

# Packet Filter Handling of ICMP

Each firewall technology handles ICMP messages differently. A pure packet-filtering router passes ICMP messages between networks statelessly. Therefore, the static rules must describe the messages allowed to pass between networks.

A minimal configuration would only permit Path MTU Discovery (PMTUD) ICMP messages to the protected network's hosts. Any other permission for ICMP packets would allow legitimate and illegitimate packets to enter.



## SPF Handling of ICMP

```
permit icmp host B host A echo-reply
permit icmp host B host A unreachable
permit icmp host B host A packet-too-big
```

```
permit icmp host A host B echo
```

**ICMP is usually handled statelessly**

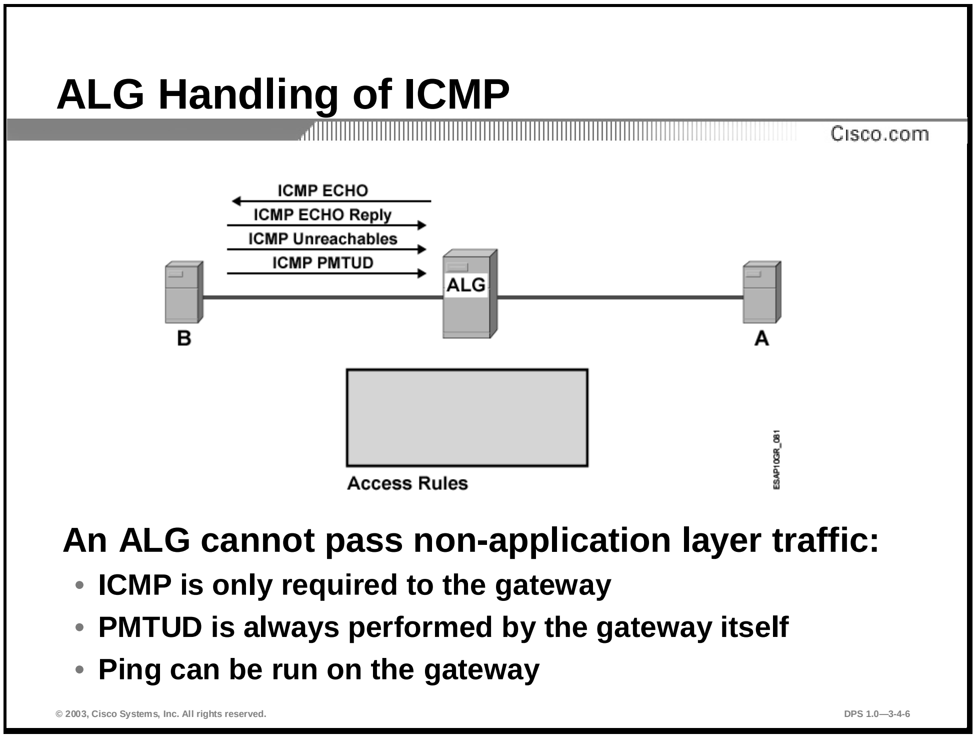
**ICMP echo and echo-reply messages are often required for ping:**

- **Various ICMP “unreachable” messages improve error detection, but are not required**
- **Modern IP stacks with Path MTU Discovery (PMTUD) require ICMP through a firewall**

DPS 1.0—3-4-5

# SPF Handling of ICMP

A stateful filtering device may enforce some stateful intelligence on ICMP traffic. For example, it could allow the ICMP echo-reply to enter the protected network only if the packet filter saw the related ICMP echo request previously. All other ICMP messages, such as unreachables, occur asynchronously. Therefore, the stateful firewall must be configured to accept them at any time, and to any host that is communicating over the firewall.



## ALG Handling of ICMP

Cisco.com

**An ALG cannot pass non-application layer traffic:**

- **ICMP is only required to the gateway**
- **PMTUD is always performed by the gateway itself**
- **Ping can be run on the gateway**

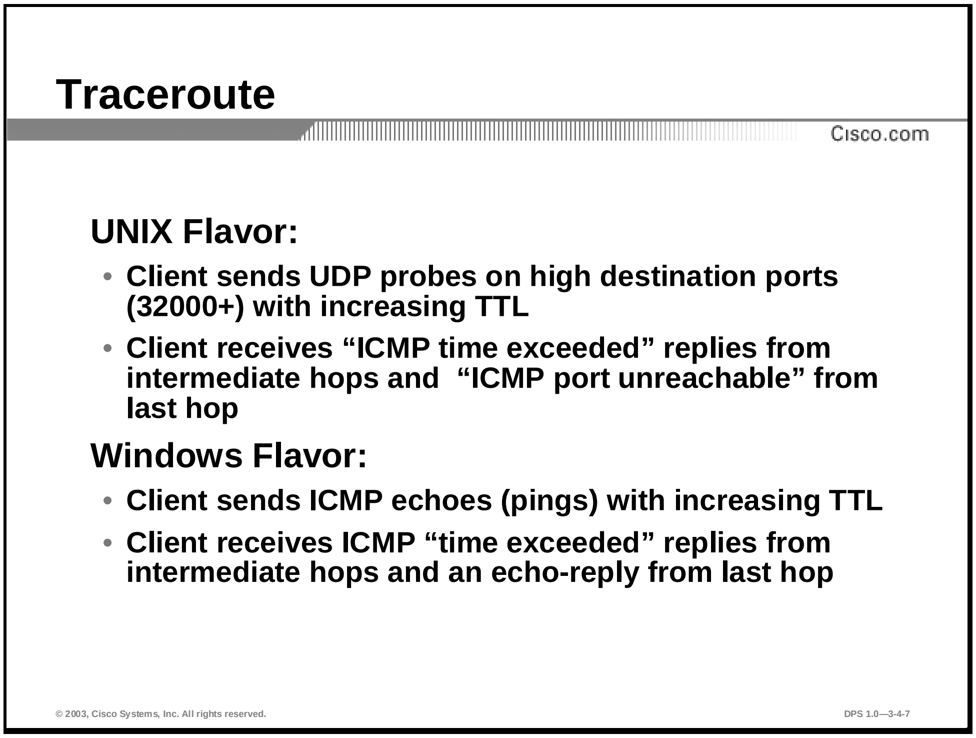
DPS 1.0—3-4-6

# ALG Handling of ICMP

An ALG establishes two network sessions between:

- The gateway and the client
- The gateway and the target server

All ICMP control traffic therefore does not need to pass between the client and the server, but rather terminates on the gateway. Thus, a firewall can only permit ICMP to and from the gateway, minimizing exposure of the protected network.



## Traceroute

Cisco.com

**UNIX Flavor:**

- **Client sends UDP probes on high destination ports (32000+) with increasing TTL**
- **Client receives "ICMP time exceeded" replies from intermediate hops and "ICMP port unreachable" from last hop**

**Windows Flavor:**

- **Client sends ICMP echoes (pings) with increasing TTL**
- **Client receives ICMP "time exceeded" replies from intermediate hops and an echo-reply from last hop**

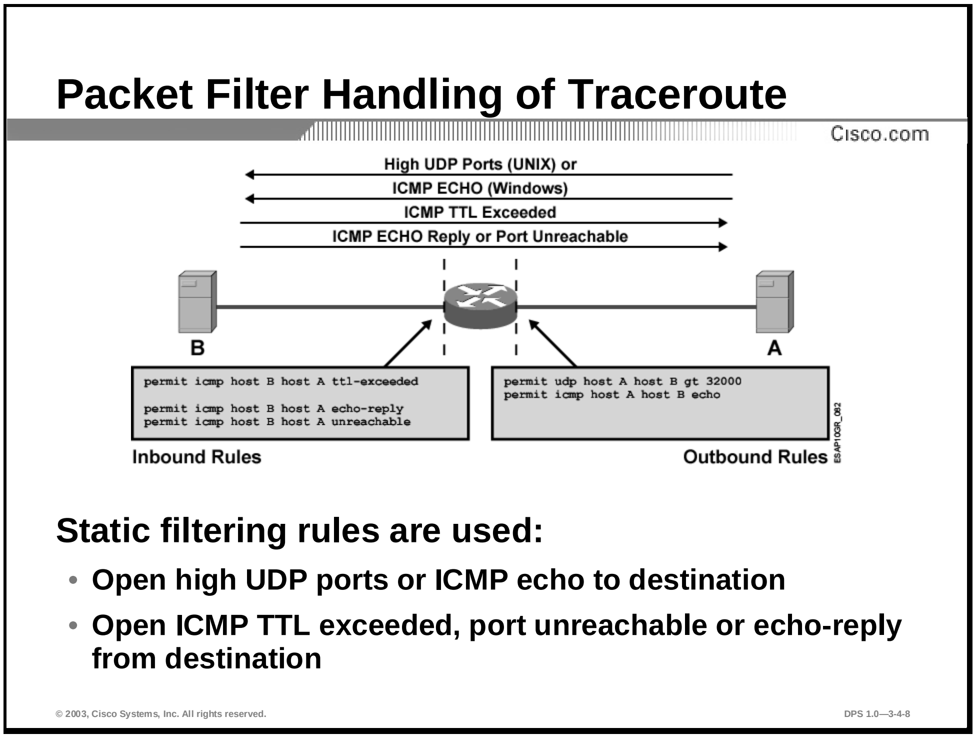
DPS 1.0—3-4-7

# Traceroute Refresher

The traceroute program attempts to determine the path between two network endpoints. There are two flavors of traceroute

- **UNIX flavor:** A public domain program running on virtually all UNIX platforms. The UNIX flavor of traceroute works by sending User Datagram Protocol (UDP) packets to high destination ports (usually higher than 32000) in phases. Each phase has an increased TTL IP parameter, starting from 1 in the first phase. Routers along the path route the UDP packets and discard them as soon as the TTL value reaches 0. At that point, the router sends a "TTL time exceeded" message to the source, informing the source about the router in the path. When the TTL is increased sufficiently to reach the last hop, the destination host receives the UDP packet and, because it is not listening on the random high UDP port, returns an "ICMP port unreachable" message to the source.

- **Windows flavor:** Runs on Microsoft Windows platforms. The Windows flavor of traceroute works similarly to the UNIX version. The difference is that it sends out ICMP ECHO messages with an increasing TTL value, and waits for "TTL time exceeded" and a final ICMP ECHO REPLY message. In terms of firewall filtering, permitting Windows traceroute is similar to permitting the PING service, with the addition of permitting "TTL time exceeded" messages to the source. A side effect of this behavior is that permitting traceroute also permits PING between the networks.



## Packet Filter Handling of Traceroute

Cisco.com

```
permit icmp host B host A ttl-exceeded

permit icmp host B host A echo-reply
permit icmp host B host A unreachable
```

```
permit udp host A host B gt 32000
permit icmp host A host B echo
```

**Static filtering rules are used:**

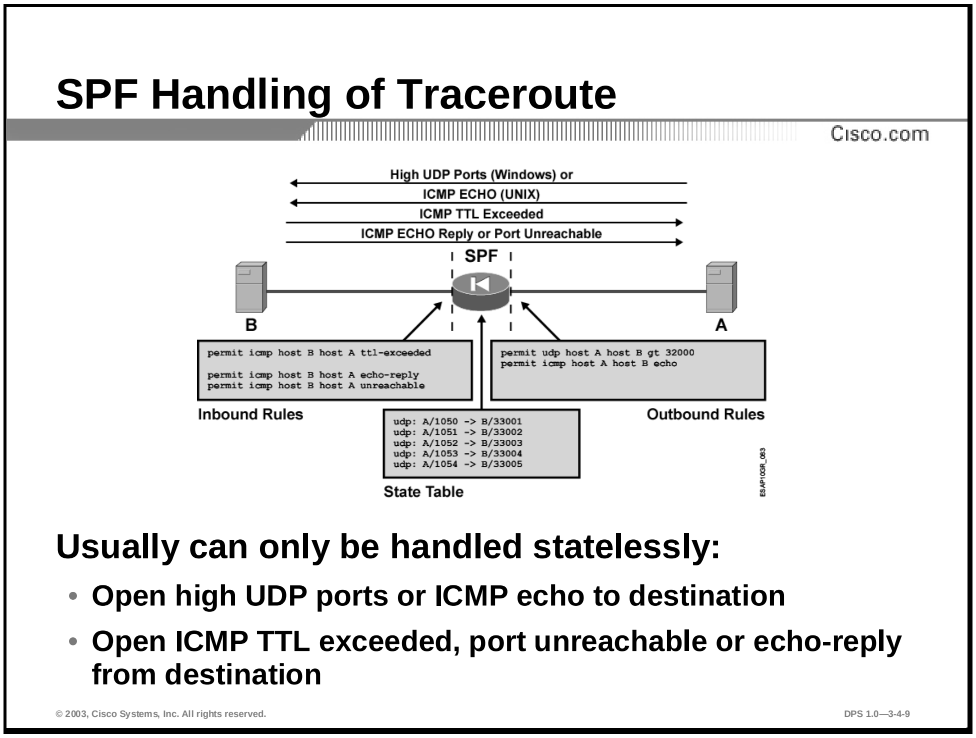
- **Open high UDP ports or ICMP echo to destination**
- **Open ICMP TTL exceeded, port unreachable or echo-reply from destination**

DPS 1.0—3-4-8

# Packet Filter Handling of Traceroute

Permitting traceroute through a firewall requires the firewall to pass the UDP and ICMP messages directly between the source and destination of the traceroute session, permitting all routers in between to answer as well.

On a packet filtering router, this simply involves allowing high-port UDP traffic out, and ICMP “TTL time exceeded” and “port unreachable” messages to the inside hosts. Static access rules can describe this quite tightly and do not open significant windows into the protected network.



## SPF Handling of Traceroute

```
permit icmp host B host A ttl-exceeded

permit icmp host B host A echo-reply
permit icmp host B host A unreachable
```

```
permit udp host A host B gt 32000
permit icmp host A host B echo
```

```
udp: A/1050 -> B/33001
udp: A/1051 -> B/33002
udp: A/1052 -> B/33003
udp: A/1053 -> B/33004
udp: A/1054 -> B/33005
```

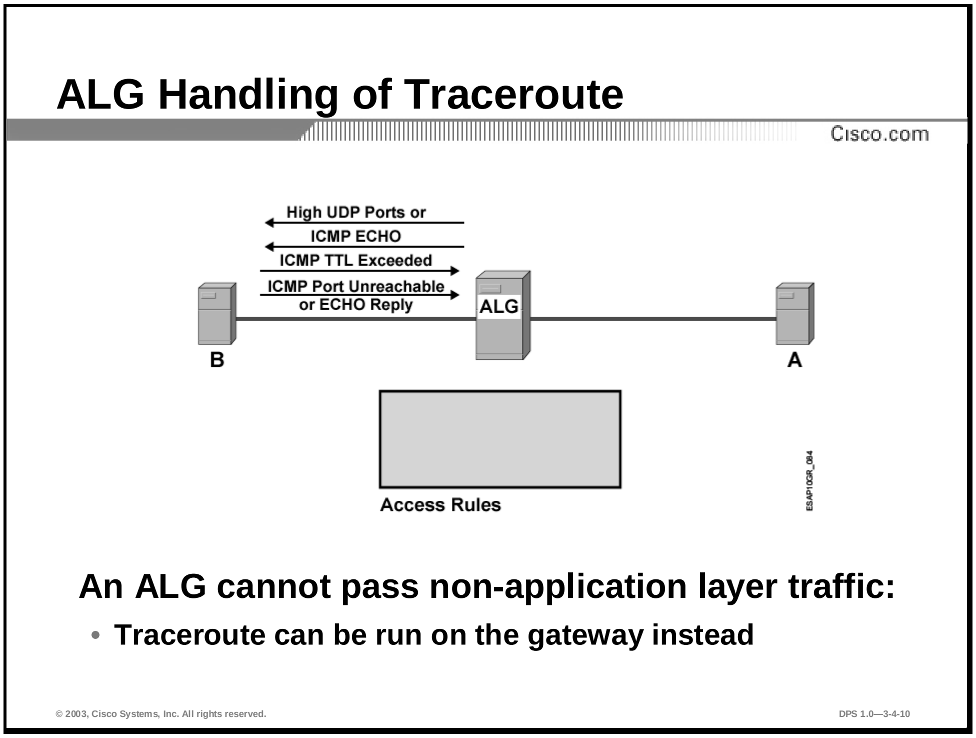
**Usually can only be handled statelessly:**

- **Open high UDP ports or ICMP echo to destination**
- **Open ICMP TTL exceeded, port unreachable or echo-reply from destination**

DPS 1.0—3-4-9

# SPF Handling of Traceroute

Using a stateful filtering router, the access rules and risk are the same as with the normal packet filter.



## ALG Handling of Traceroute

Cisco.com

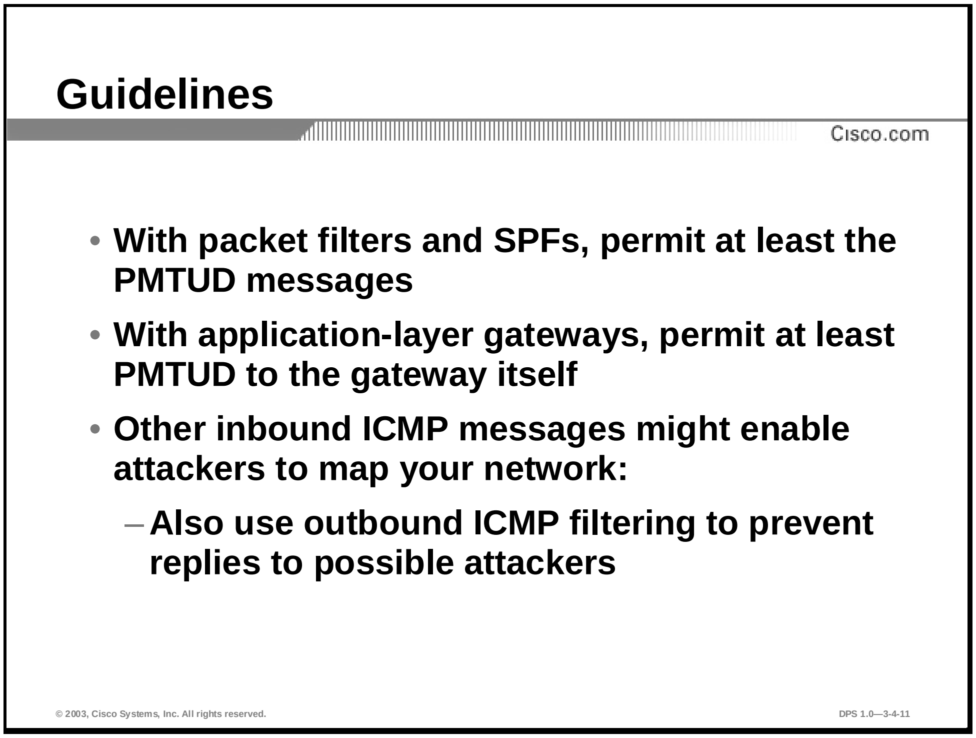
**An ALG cannot pass non-application layer traffic:**

- **Traceroute can be run on the gateway instead**

© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—3-4-10

# ALG Handling of Traceroute

By design, an ALG does not pass any raw packets, such as UDP or ICMP, between the protected and outside networks. Therefore, traceroute breaks where there is an ALG in the traceroute path. A solution for this would be to install traceroute on the application gateway, allow select users restricted login to the application gateway, and then run traceroute.



## Guidelines

- With packet filters and SPFs, permit at least the PMTUD messages
- With application-layer gateways, permit at least PMTUD to the gateway itself
- Other inbound ICMP messages might enable attackers to map your network:
  - Also use outbound ICMP filtering to prevent replies to possible attackers

DPS 1.0—3-4-11

# Guidelines

The main risk associated with permitting *inbound* ICMP to a protected network is associated with network mapping. ICMP can be used to:

- Elicit various responses from remote hosts using ICMP ECHO, ICMP MASK REQUEST, ICMP TIMESTAMP, and similar messages
- Verify reachability of destination hosts

The most important guideline for *outbound* ICMP access is not to allow outgoing ICMP traffic, which might reveal information about the inside network. For example, if an attacker manages to send a probe into the protected network, outbound rules should prevent the response from reaching the attacker.

It is good practice to block all ICMP traffic outbound, except the required messages. An example of this could be PMTUD messages, which might be sent by the protected networks' routers to outside hosts. ICMP ECHOs might be allowed out to allow pinging of the outside network. Other outgoing messages, such as port and network unreachables or mask/timestamp replies, may provide valuable reachability information to an attacker.

In the context of an Internet firewall, where the probability of mapping attacks is very high, the most conservative stance with ICMP is suggested. A solution that permits the minimal set of ICMP messages should be used. This includes only the messages needed for PMTUD (ICMP "Fragmentation needed but DF set" unreachable message). Often, an ICMP ECHO REPLY packet will be permitted inbound to allow pinging of the outside network, but this is not necessary.

# Example

If inbound ICMP “TTL exceeded” messages are allowed into a protected network, the following could occur:

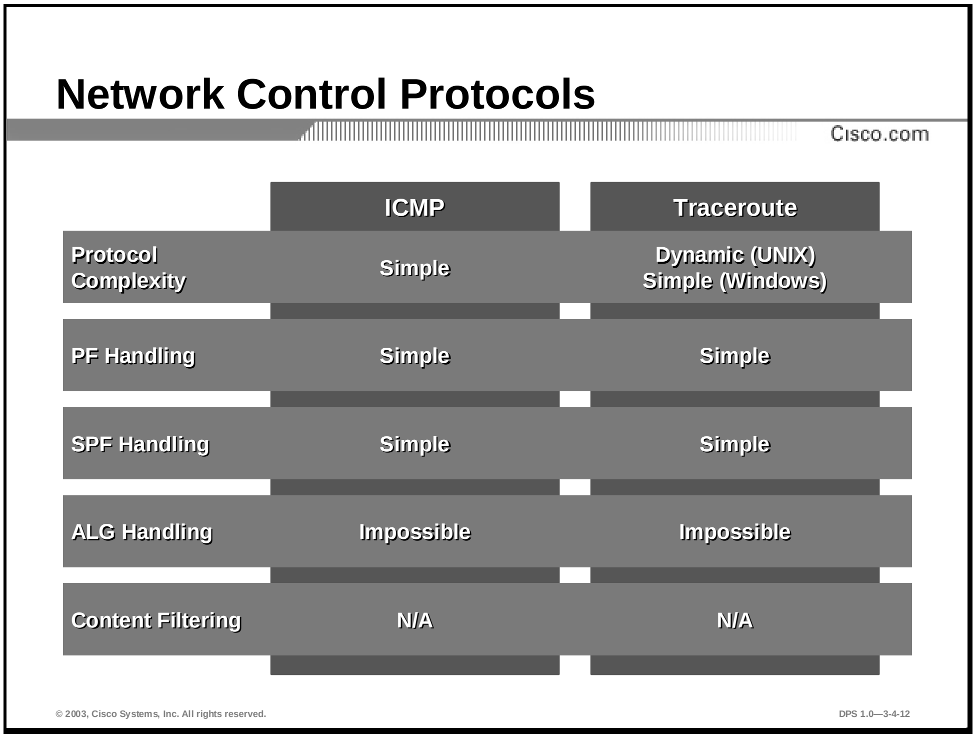
**Step 1** An attacker uses a network mapping technique to send ICMP “TTL exceeded” messages to every single host behind the firewall.

**Step 2** The firewall passes the message to the inside network.

**Step 3** If the host does not exist or is unreachable, the router nearest to the destination returns an ICMP “Host Unreachable” message, informing the attacker about a host NOT present.

**Step 4** If there is no reply, the host has processed and discarded the message, and is therefore alive.

The use of NAT and PAT mitigates this risk by not exposing all inside hosts permanently, in spite of firewall access rules.



## Network Control Protocols

| | ICMP | Traceroute |
|---|---|---|
| Protocol Complexity | Simple | Dynamic (UNIX)<br>Simple (Windows) |
| PF Handling | Simple | Simple |
| SPF Handling | Simple | Simple |
| ALG Handling | Impossible | Impossible |
| Content Filtering | N/A | N/A |

DPS 1.0—3-4-12

# Practice

Q1) How does an ALG natively pass ICMP between perimeters?

A) using a specialized proxy program

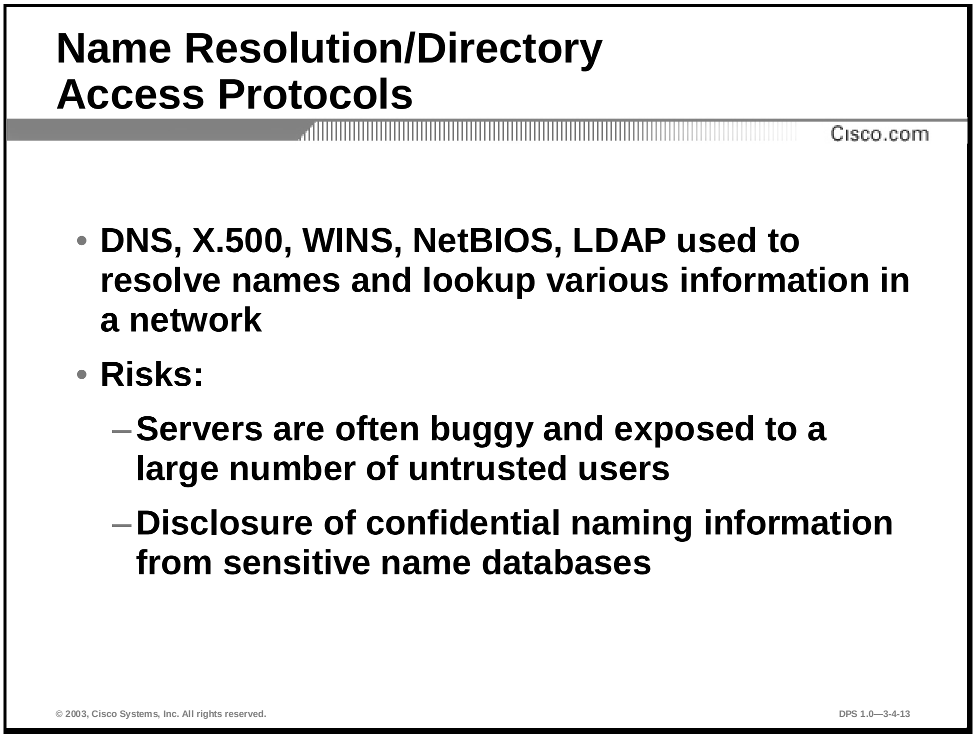
B) using packet filtering

C) it does not—all ICMP traffic terminates at the ALG host

D) using stateful packet filtering, if available

E) using a connection relay

# Name Resolution Protocols



**Name Resolution/Directory Access Protocols**

- **DNS, X.500, WINS, NetBIOS, LDAP used to resolve names and lookup various information in a network**
- **Risks:**
  - **Servers are often buggy and exposed to a large number of untrusted users**
  - **Disclosure of confidential naming information from sensitive name databases**

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—3-4-13

## Objective

The section will enable the learner to explain the security properties of network control protocols to select an appropriate technology to securely pass them over firewalls.

## Introduction

Directory and name resolution protocols provide the lookup of information in network-accessible databases. Firewalls often pass name resolution/directory protocols between trusted and untrusted network, and secure handling of those protocols is required.

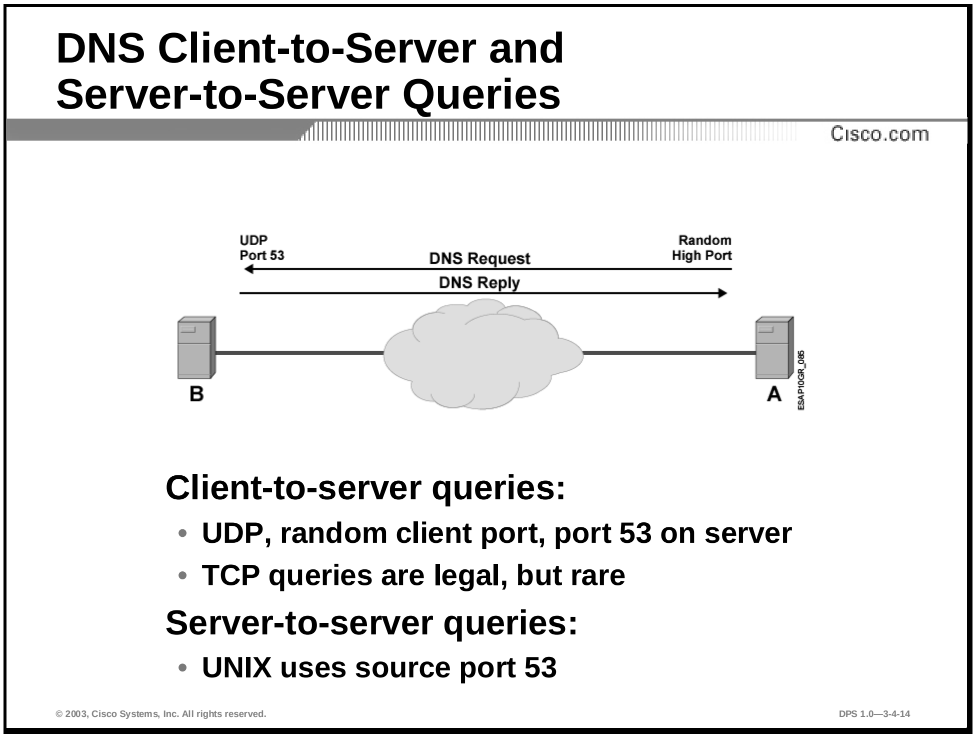
## Name Resolution Refresher

The main risk associated with name resolution/directory servers is that they are often buggy, and combined with their exposure to an extremely large user population, the threat of their compromise is very high. Often, those servers also run with high privileges on the host system, and, if exploited, allow an attacker to fully compromise the host.

Some directories may also contain confidential information. Windows 2000 Active Directory, for example, contains user account information which, if disclosed, would enable the attacker to compromise arbitrary user accounts.

## Example

The canonical example of a security-relevant risk is the use of the Berkeley Internet Name Domain (BIND) Domain Name System (DNS) server on the Internet. While respected for its stability, BIND has been infected with some security vulnerabilities that could crash it and either cause a denial-of-service (DoS) attack, or give the attacker an opportunity to access the host system with the privileges of the BIND software. Unfortunately, BIND software has historically been run with the highest system administrator privileges, thus enabling an attacker to fully control the host operating system.



## DNS Client-to-Server and Server-to-Server Queries

Cisco.com

UDP
Port 53
DNS Request
Random
High Port
DNS Reply
B
A
ESAP10GR_085

**Client-to-server queries:**

- **UDP, random client port, port 53 on server**
- **TCP queries are legal, but rare**

**Server-to-server queries:**

- **UNIX uses source port 53**

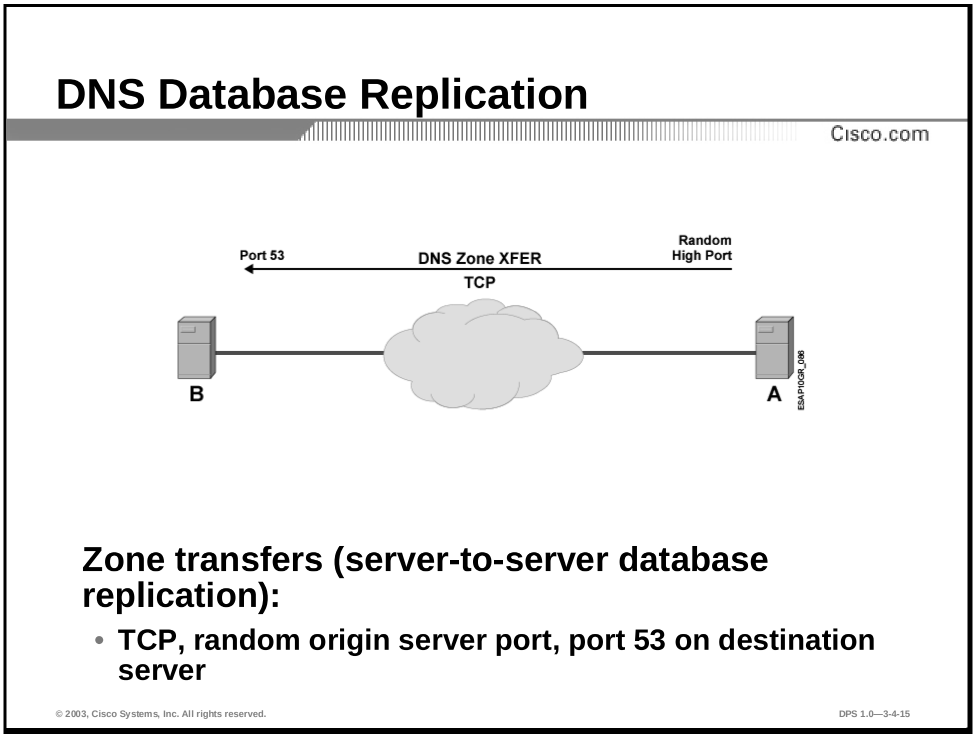
DPS 1.0—3-4-14

## DNS Queries

The DNS protocol specifies port 53 as the well-known port for client-server and server-server queries, as well as for database replication. The general rule is to use the UDP protocol, and to only use TCP when a large reply is expected (over 512 bytes). Database replication (zone transfers) strictly uses TCP on the well-known server port of 53, while almost all other transactions occur over the UDP protocol.

The DNS client-to-server protocol is a simple request-reply (ping-pong) protocol, where the client uses a high random UDP port to send a request to port 53 on the server. The server replies from port 53 to the high client port. The standard also specifies that TCP can be used to make the queries, but this behavior is rarely seen. IBM AIX operating system is one example, which uses both UDP and TCP querying.

The DNS server-to-server transactions (request forwarding) occur almost exclusively over UDP, with a destination port of 53. The source port varies by implementation. The classic UNIX BIND nameserver also uses 53 as the source port, but most modern name servers use a random high UDP port at the source of the request.



## DNS Database Replication

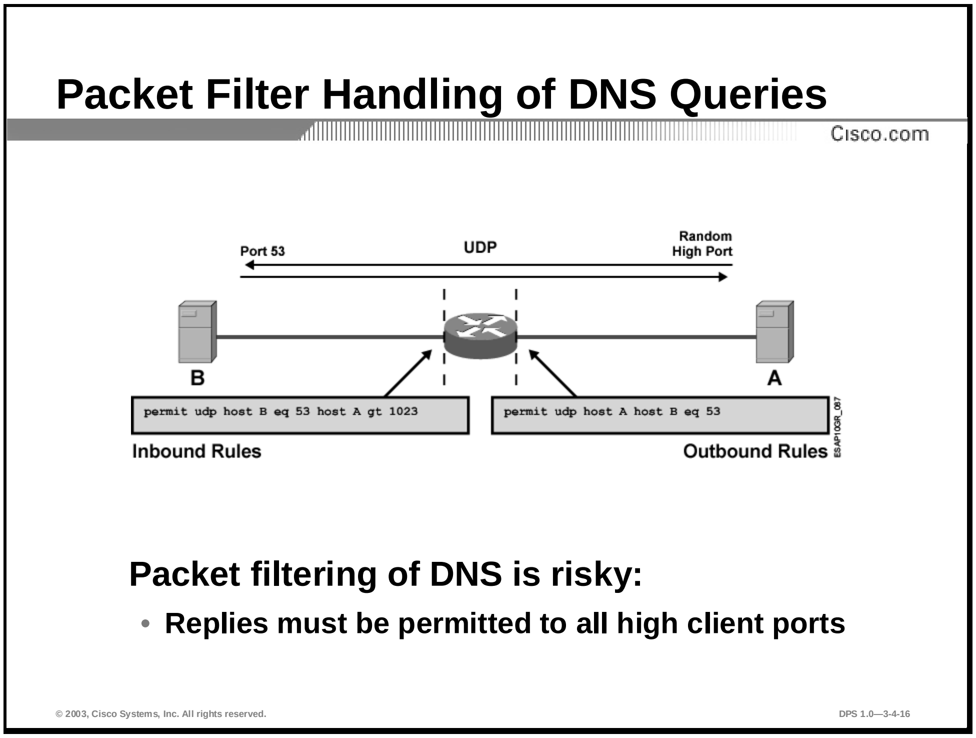
**Zone transfers (server-to-server database replication):**

- **TCP, random origin server port, port 53 on destination server**

DPS 1.0—3-4-15

## DNS Database Replication

Zone transfers are used to replicate databases between primary and secondary servers for a zone (domain). They occur exclusively over TCP with a random port at the originator, and port 53 at the destination DNS server.



## Packet Filter Handling of DNS Queries

Cisco.com
Port 53
UDP
Random
High Port
B
A
Inbound Rules
Outbound Rules
ESAP10GR_087

```
permit udp host B eq 53 host A gt 1023
```

```
permit udp host A host B eq 53
```

**Packet filtering of DNS is risky:**

- **Replies must be permitted to all high client ports**

## Packet Filter Handling of DNS Queries

The DNS client-to-server queries across packet filters can be reasonably secure or completely insecure, depending on the location of the server and client. As with the majority of UDP applications, the client will choose a random high UDP source port to send its request to the server.

The packet filter is configured with two access lists: one on the outside interface (enforcing inbound rules), and one on the inside interface (enforcing outbound rules).

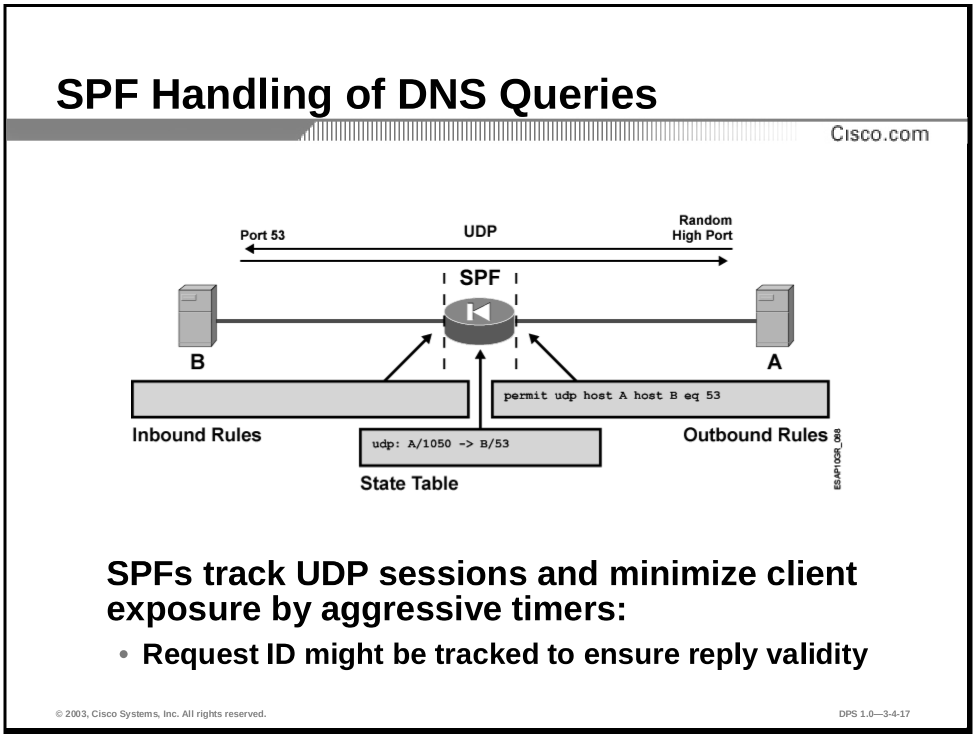
To permit the outbound request on the inside interface, permit all UDP packets from any client source port, to the server destination port of 53. The packet filter cannot “remember” this packet, therefore, when the server replies another rule needs to be in place on the outside interface to explicitly permit return traffic.

As the client port is random, the packet-filtering rule on the outside interface (inbound traffic filter) has to include all possible client ports. The rule therefore permits UDP packets from server source port of 53 to all high client destination ports.

Such a configuration effectively opens all the ports on the client machine for attack, if the attacker uses packets with source port of 53. An attacker might exploit this and connect to a client service, which listens on a high UDP port, for example, a Network File System (NFS). Therefore, if the client needs to be well protected, a packet filter cannot securely filter the DNS queries. Packet filtering is very effective in protecting the server, if there is no need to protect the client. It only allows packets with a destination port of 53 to the server and blocks all other server applications.

## Example

A small enterprise is using a packet filtering router on their Internet connection as the only firewall. On a separate router LAN interface, they have set up a public external DNS server, which answers requests from the Internet. The packet filtering router can now be configured with a rule that permits UDP packets from any source UDP port to port 53 on the server, and denies all other UDP traffic. Another rule for return traffic can be also set up, permitting all packets from the server’s port 53 to all high ports on the clients. Because the client does not belong to the enterprise and the server is an unlikely source of attack, this setup is robust.



**SPF Handling of DNS Queries**

Cisco.com

Port 53

UDP

Random
High Port

SPF

B

A

Inbound Rules

permit udp host A host B eq 53

Outbound Rules

udp: A/1050 -> B/53

State Table

ESAP10GR_088

**SPFs track UDP sessions and minimize client exposure by aggressive timers:**

- **Request ID might be tracked to ensure reply validity**

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—3-4-17

## SPF Handling of DNS Queries

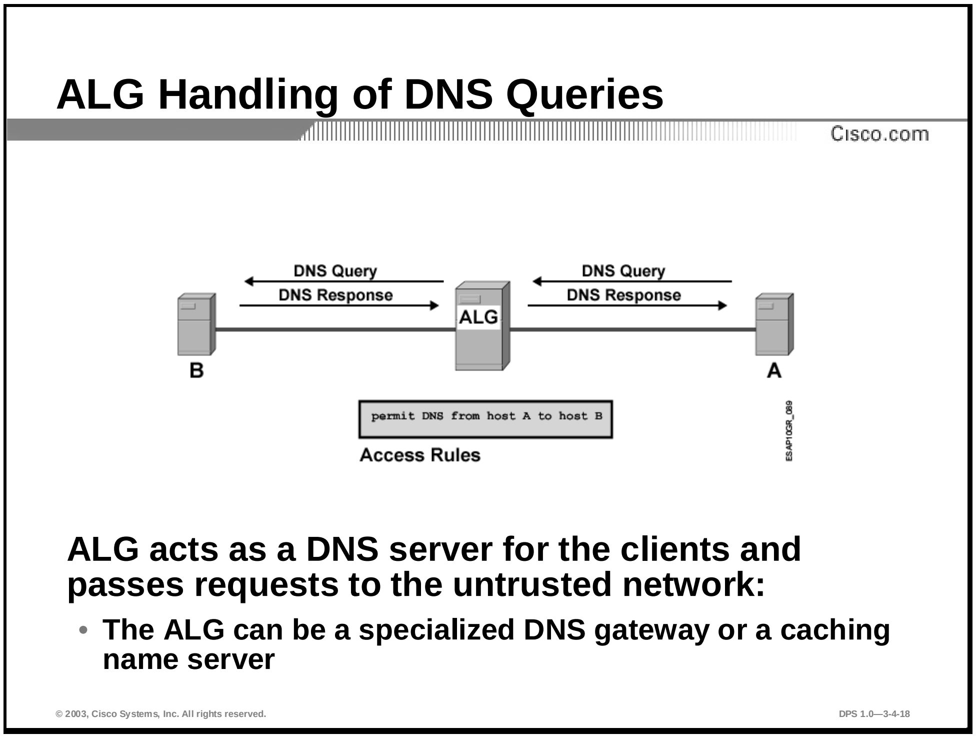
SPFs remove the need for rules permitting return traffic, as they are flow and are application aware. A SPF on two levels usually handles DNS:

- When a SPF firewall rule permits a DNS query it creates a connection entry in the state table. This connection entry permits all return traffic from the server back to the client. Normally, the connection entry will be closed when an idle timeout expires.

- Some SPF implementations such as the Cisco Secure PIX Firewall or IOS Firewall are more intelligent and perform an application-layer inspection of the DNS request. When the DNS request arrives at the firewall, they remember the DNS transaction ID inside the packet, and check the reply packet's ID against the stored information. As DNS is a request-reply (ping-pong) protocol, the connection entry closes as soon as the reply is received, without waiting for the default idle timeout.

**Note** The PIX Firewall only allows the first DNS response to pass inbound and then closes the connection. The IOS Firewall allows for a window of 5 seconds of pass any DNS replies to the client after a valid request was seen going outbound.

Used together, both methods enforce a very strict mechanism for passing DNS queries directly through a firewall. However, they cannot defend against malicious replies, which can contain reply data that somehow infects the client, for example, crash it, execute on the stack. The use of ALGs helps protect against these application-level attacks on DNS.



## ALG Handling of DNS Queries

Cisco.com

DNS Query
DNS Response
ALG
DNS Query
DNS Response
B
A
permit DNS from host A to host B
Access Rules
ESAP10GR_089

**ALG acts as a DNS server for the clients and passes requests to the untrusted network:**

- **The ALG can be a specialized DNS gateway or a caching name server**

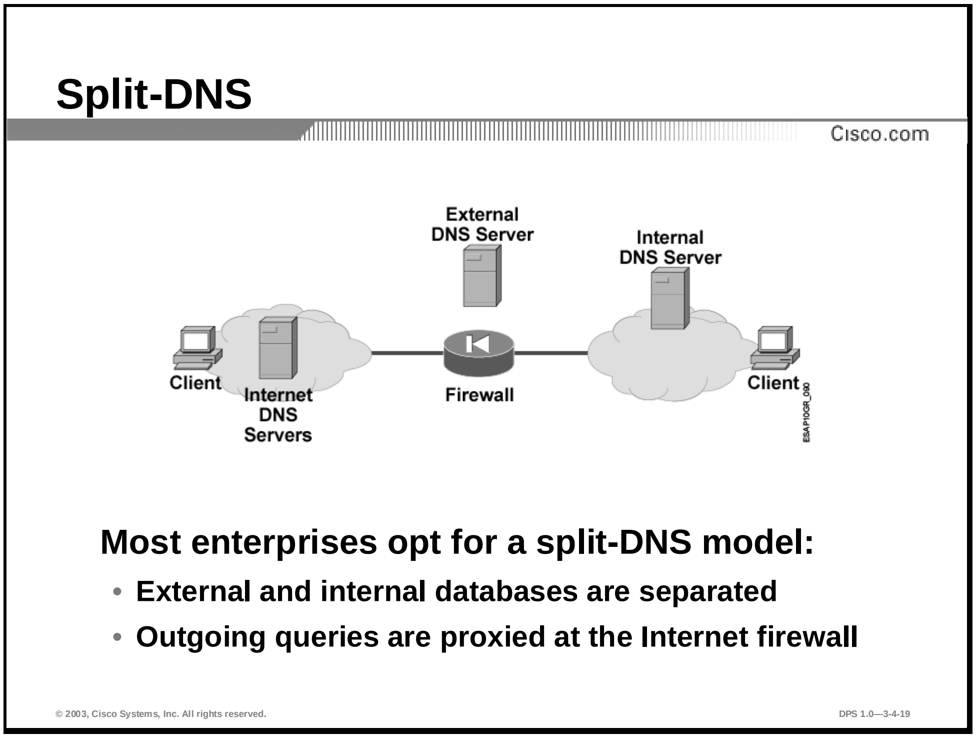
DPS 1.0—3-4-18

## ALG Handling of DNS Queries

An ALG for the DNS protocol can be a specialized software package, running in the context of a larger product, or an off-the-shelf caching name server, such as BIND or Cisco Network Registrar (CNR).

The ALG passes DNS traffic over the firewall by posing as a DNS server to the inside clients, accepting their requests, and forwarding those requests to outside DNS servers. The outside DNS servers send their replies to the ALG, which inspects the reply on the application layer, and creates a new reply to which it copies the received information and forwards it to the inside client.

The ALG will typically perform some sanity checks on the packet payload, and can be configured to filter DNS information according to a policy. All this sanitizes information inside the DNS protocol as it is passed between the two networks.



Split-DNS

**Most enterprises opt for a split-DNS model:**

- **External and internal databases are separated**
- **Outgoing queries are proxied at the Internet firewall**

DPS 1.0—3-4-19

## Split-DNS

This figure illustrates the concept of split-DNS, which most enterprises use to separate their public and private DNS directories.

When using split-DNS, there are two DNS zones (domains), which are maintained by the enterprise:

- **The external DNS zone:** For example, cisco.com. This is served by a public external DNS server, which is located outside the protected network, usually in one of the DMZ networks on the firewall or on the firewall host itself.

- **The internal DNS zone:** The same name, cisco.com. This is served by the internal DNS servers.

Users in external networks can only access the external DNS servers. These only serve information about publicly reachable servers, such as the corporate web or mail servers.

Users in internal networks can only access the internal DNS servers, which provide name-to-address mapping for all hosts within the enterprise.

When a user in an internal network needs to obtain DNS information from the external network:

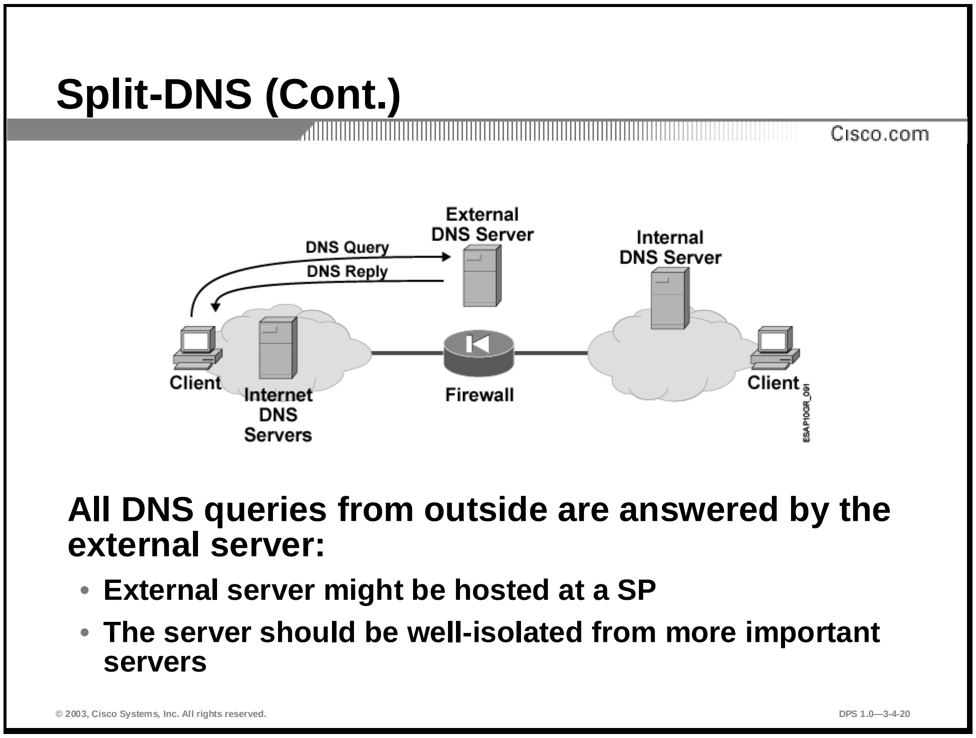
**Step 1** The user asks an internal DNS server for the information

**Step 2** The internal DNS server forwards the query to the external DNS server at the firewall

**Step 3** The external DNS server forwards the query to the Internet DNS servers.

**Step 4** The reply comes back to the external DNS server, which forwards in to the internal DNS server

**Step 5** The internal DNS server returns the reply to the client



## Split-DNS (Cont.)

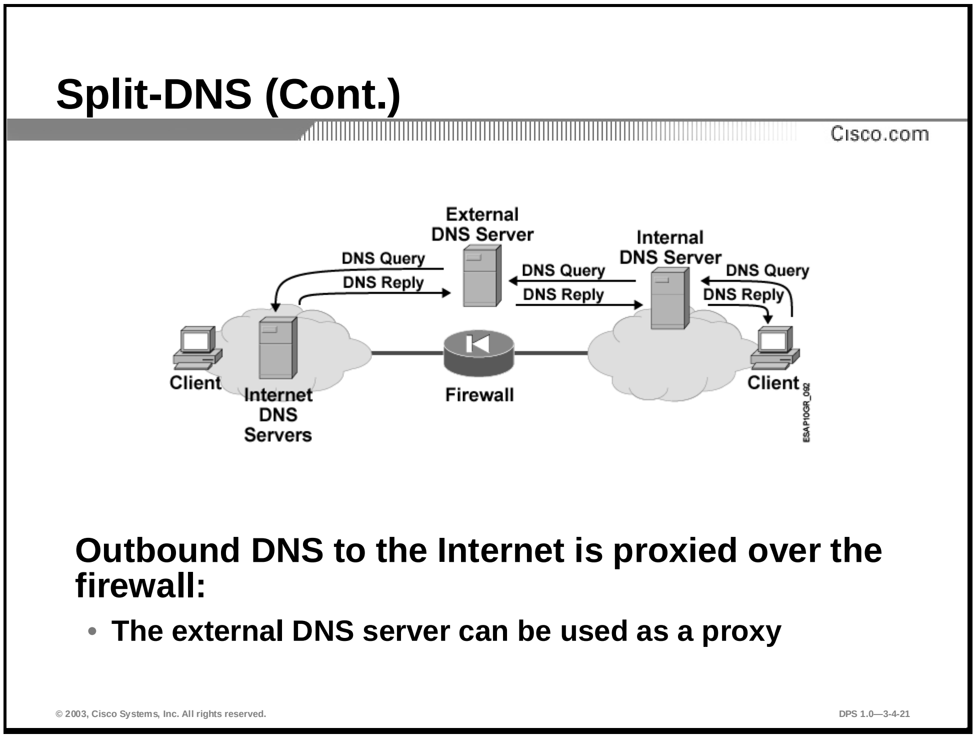
Cisco.com

**All DNS queries from outside are answered by the external server:**

- **External server might be hosted at a SP**
- **The server should be well-isolated from more important servers**

DPS 1.0—3-4-20

This figure illustrates how queries from the untrusted network terminate at the external DNS server. This server should be well protected and use robust DNS server software. Optionally, a hosting service provider (SP) hosts this server, and replicates geographically for highest availability.



## Split-DNS (Cont.)

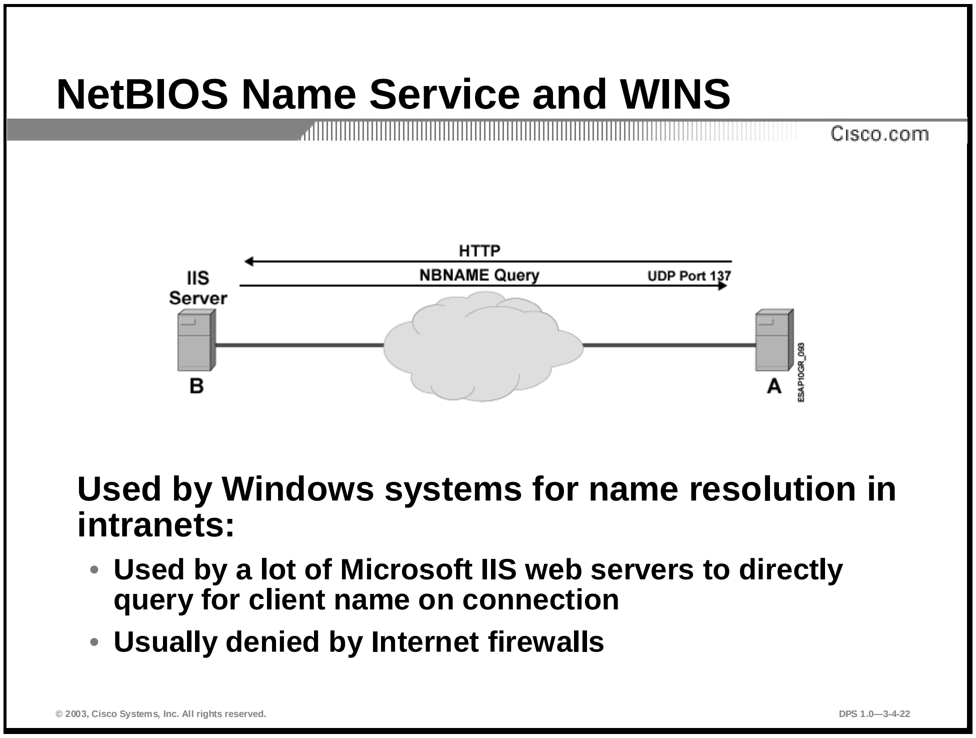
Cisco.com

**Outbound DNS to the Internet is proxied over the firewall:**

- **The external DNS server can be used as a proxy**

DPS 1.0—3-4-21

This figure illustrates how queries from the trusted network pass from the internal servers to the external server, who forwards (proxies) them to the Internet. The external DNS server’s software should stop any attack attempted inside the DNS protocol.


Cisco.com
HTTP
NBNAME Query
UDP Port 137
IIS
Server
B
A
ESAP10GR_093

## NetBIOS Name Service and WINS

**Used by Windows systems for name resolution in intranets:**

- **Used by a lot of Microsoft IIS web servers to directly query for client name on connection**
- **Usually denied by Internet firewalls**

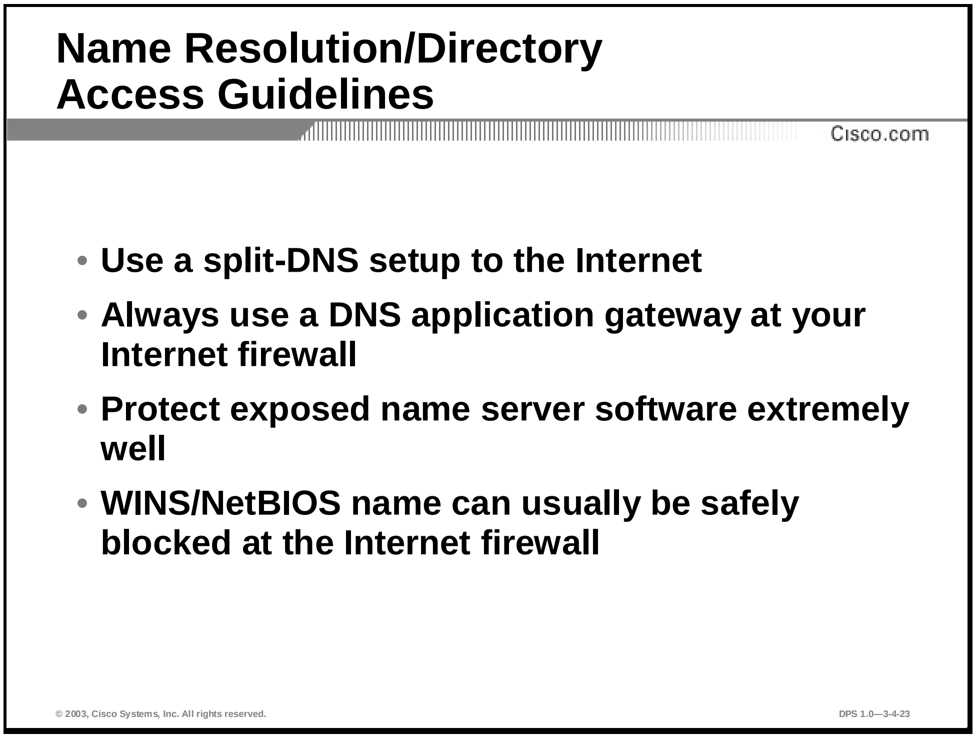
DPS 1.0—3-4-22

## NetBIOS Name and WINS

Microsoft clients and servers for name resolution often use the WINS and NetBIOS name services. Unfortunately, many Internet servers running Microsoft IIS software still try to use them for name resolution in the Internet when they attempt to resolve the client’s name. This usually results in those queries, which are sent directly to the client, being denied by firewalls, significantly increasing the audit trail.

## LDAP

Lightweight Directory Access Protocol (LDAP) is a simple, single-TCP-session application (it uses TCP port 389), used to access X.500 directories. The main risks lie in possible software bugs in the LDAP server, and disclosure of sensitive information in the directory, if the access control lists governing access to directory data are set incorrectly. From the protocol perspective, LDAP is easily passed through any firewall filtering technology. If application-layer control is required, a replica X.500 directory can be set up with only partial data available to untrusted clients (the same concept as split-DNS).



## Name Resolution/Directory Access Guidelines

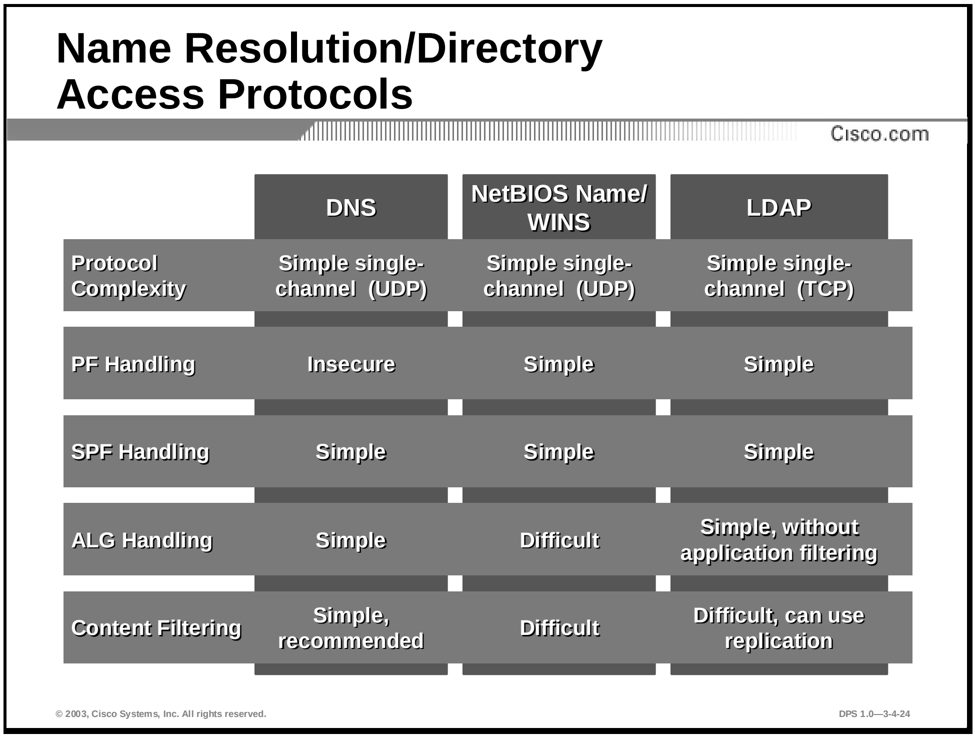
- Use a split-DNS setup to the Internet
- Always use a DNS application gateway at your Internet firewall
- Protect exposed name server software extremely well
- WINS/NetBIOS name can usually be safely blocked at the Internet firewall

DPS 1.0—3-4-23

# Guidelines

It is important to protect directory and name resolution servers as they are exposed to a large user population. For DNS, it is suggested to always proxy it over an ALG. For example, a caching DNS server to the Internet, as DNS is historically vulnerable to malformed requests. Split-DNS is a proven method used to deploy DNS forwarding over the Internet firewall.

WINS and NetBIOS name resolution can be safely blocked on the Internet firewall as no functionality is broken if requests are filtered.



## Name Resolution/Directory Access Protocols

| | DNS | NetBIOS Name/ WINS | LDAP |
|---|---|---|---|
| Protocol Complexity | Simple single-channel (UDP) | Simple single-channel (UDP) | Simple single-channel (TCP) |
| PF Handling | Insecure | Simple | Simple |
| SPF Handling | Simple | Simple | Simple |
| ALG Handling | Simple | Difficult | Simple, without application filtering |
| Content Filtering | Simple, recommended | Difficult | Difficult, can use replication |

DPS 1.0—3-4-24

## Practice

Q1) What does split-DNS refer to?

A) the separation of an organization’s external and internal DNS databases outside and inside the firewall

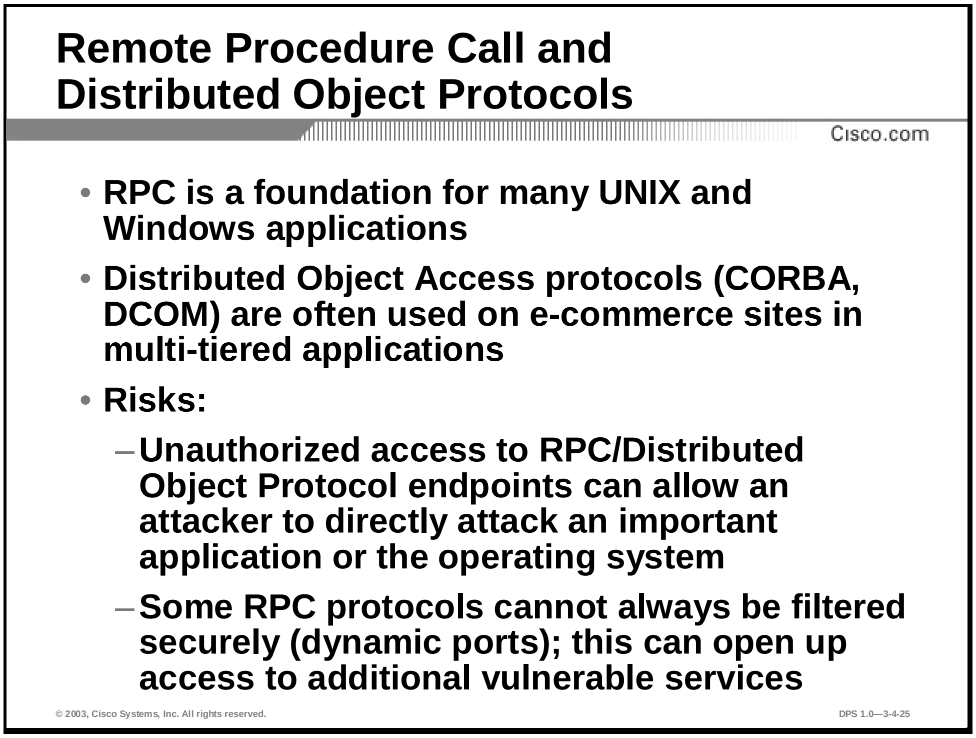
B) the separation of an organization’s external and internal DNS databases outside the firewall

C) the answering of all DNS client queries by two distinct name servers

D) the splitting of DNS packets by the firewall to ensure validity

E) the separation of name service so that DNS is never used inside a firewall

# Remote Procedure Call Protocols



## Remote Procedure Call and Distributed Object Protocols

Cisco.com

- RPC is a foundation for many UNIX and Windows applications
- Distributed Object Access protocols (CORBA, DCOM) are often used on e-commerce sites in multi-tiered applications
- Risks:
  - Unauthorized access to RPC/Distributed Object Protocol endpoints can allow an attacker to directly attack an important application or the operating system
  - Some RPC protocols cannot always be filtered securely (dynamic ports); this can open up access to additional vulnerable services

© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—3-4-25

## Objective

The section will enable the learner to explain the security properties of Remote Procedure Call (RPC) protocols to select an appropriate technology to securely pass them over firewalls.

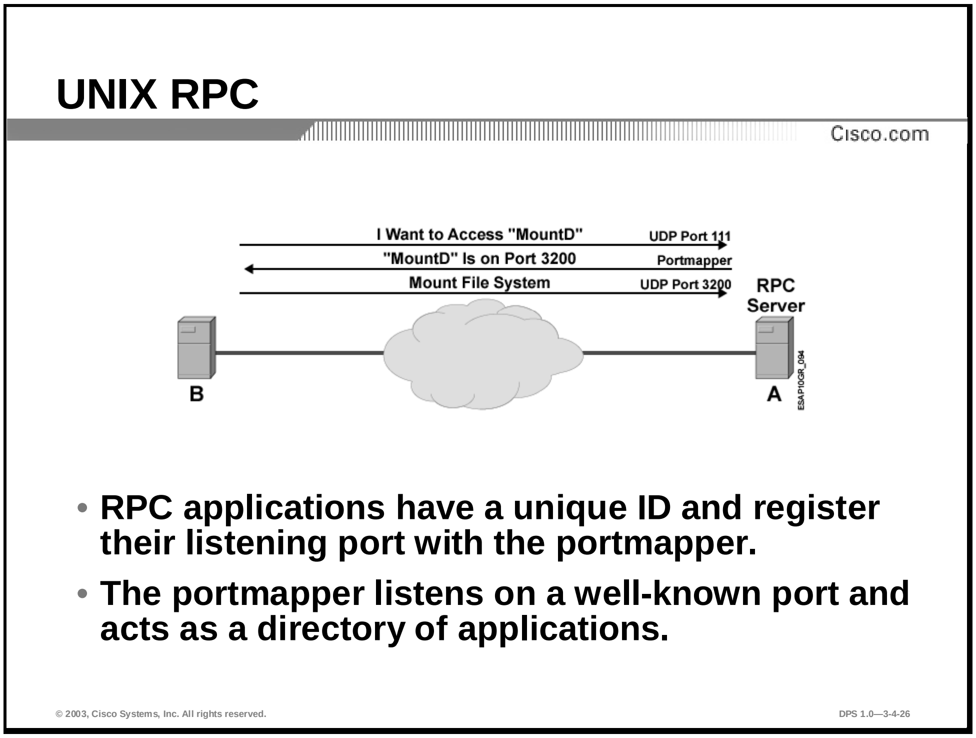
## Introduction

RPCs are the foundation of many UNIX and Windows applications, which run distributed over the network. Enterprise applications also often use Common Object Request Broker Architecture (CORBA) and Distributed Component Object Model (DCOM), two related distributed object models, which may use existing RPC functionality. Such protocols are often permitted through firewalls, and a need for secure handling is paramount.

## RPC and Distributed Object Protocols' Refresher

The main risk of RPCs, and distributed object protocols, lays in the possible vulnerabilities of the endpoints. These allow a remote attacker to invoke functions on the exposed server. As RPC and distributed objects are often a gateway to core enterprise applications, the data retrieved by an attacker from a compromised RPC or distributed object protocol endpoint can be extremely sensitive.

Some firewall technologies also cannot securely filter certain RPC protocols, forcing an organization to permit more than the minimal possible access, violating the least privilege concept.



## UNIX RPC

- RPC applications have a unique ID and register their listening port with the portmapper.
- The portmapper listens on a well-known port and acts as a directory of applications.

## UNIX RPC

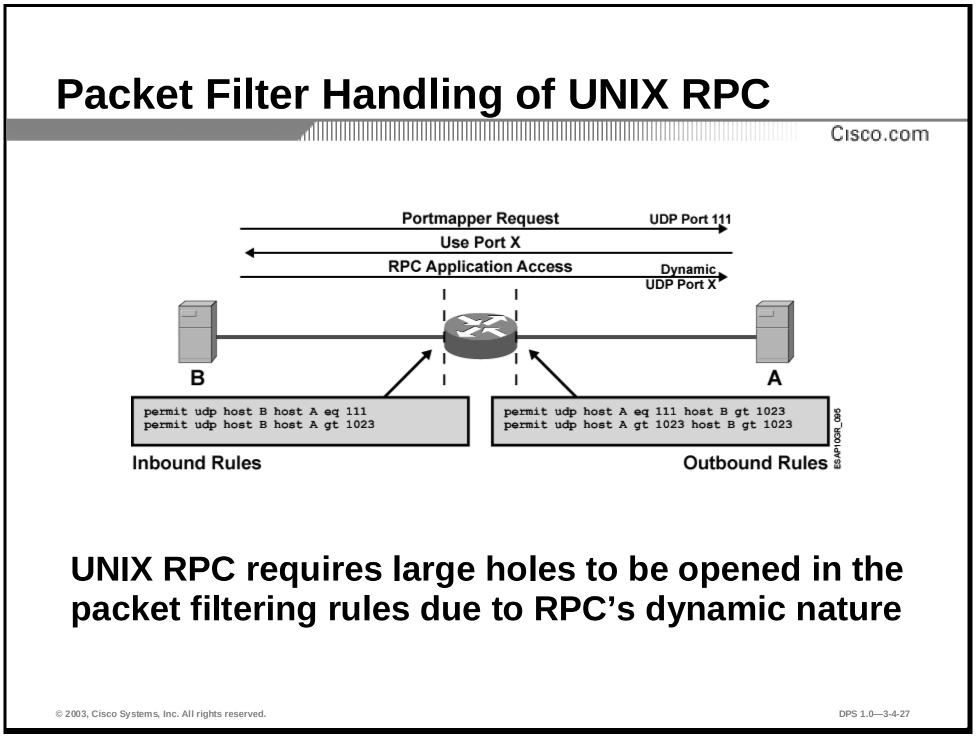
The UNIX RPC system consists of RPC-enabled applications, such as the mountd, lockd, and pcnfsd, as well as the “portmapper” program. As there are many RPC applications, they do not have officially assigned port numbers. Instead, each application has a standardized “Application ID”, and chooses any port number it likes, and listens on it. Additionally, the application registers itself with the “portmapper” program, which acts as a directory of applications, providing the clients with the applications’ port numbers.

A client connecting to a RPC application:

**Step 1** Connects to the portmapper

**Step 2** Requests the port number for a certain application, identified by its “Application ID”

**Step 3** The portmapper returns the application’s port number to the client, which can then directly connect to the RPC application



## Packet Filter Handling of UNIX RPC

```
permit udp host B host A eq 111
permit udp host B host A gt 1023
```

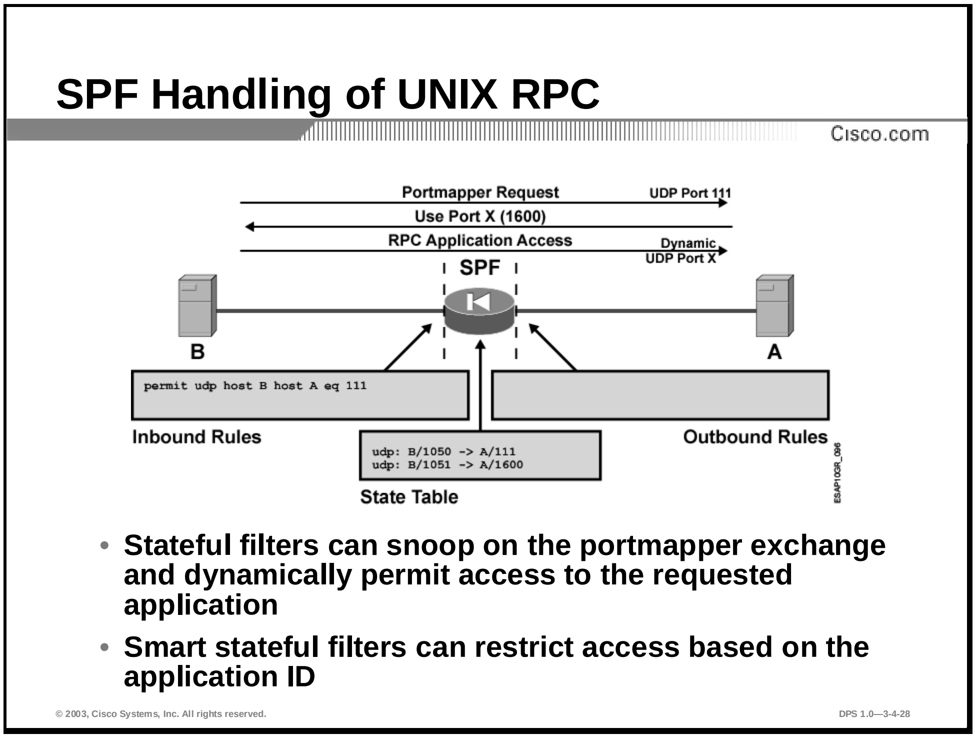
```
permit udp host A eq 111 host B gt 1023
permit udp host A gt 1023 host B gt 1023
```

**UNIX RPC requires large holes to be opened in the packet filtering rules due to RPC’s dynamic nature**

DPS 1.0—3-4-27

## Packet Filter Handling of UNIX RPC

As the application’s port numbers are dynamic and use any high UDP or TCP port number, a classic packet filter cannot filter UNIX RPC traffic securely. To access a RPC application, the packet filtering rules need to permit access to UDP port 111 (portmapper) and all high UDP and TCP ports. This presents an obvious risk, as any applications on high UDP and TCP ports can be accessed and exploited.



## SPF Handling of UNIX RPC

```
permit udp host B host A eq 111
```

```
udp: B/1050 -> A/111
udp: B/1051 -> A/1600
```

- Stateful filters can snoop on the portmapper exchange and dynamically permit access to the requested application
- Smart stateful filters can restrict access based on the application ID

DPS 1.0—3-4-28

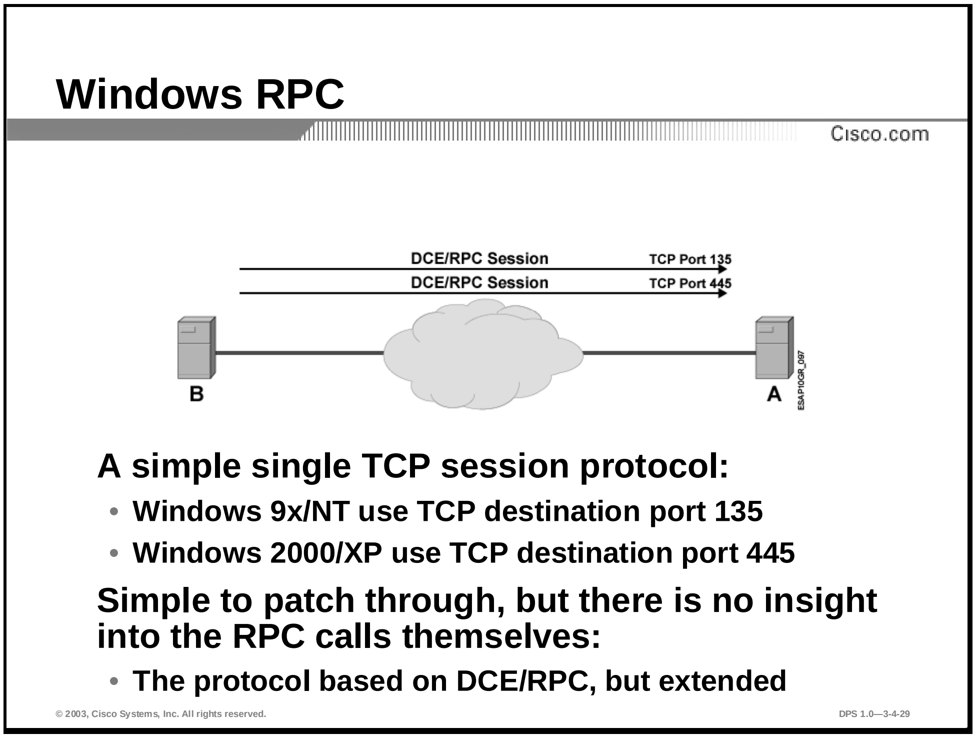
## SPF Handling of UNIX RPC

A stateful packet filter adds intelligence in the handling of the UNIX RPC protocols. There are two levels of protection:

- **Basic support:** Consists of snooping on all portmapper transactions. When the portmapper returns a port number, it dynamically reconfigures the firewall rules to support connections to that port. This only allows access to the requested RPC application. The Cisco Secure PIX Firewall performs such snooping.

- **Advanced support:** Also involves filtering on the “Application IDs”, which the client is allowed to access. The SPF can only allow requests for specific applications by looking deeper into the packet. For example, access to mountd is allowed, but access to lockd is not. The Cisco IOS Firewall performs such filtering.

There are no common ALGs for UNIX RPC.



## Windows RPC

Cisco.com

**A simple single TCP session protocol:**

- **Windows 9x/NT use TCP destination port 135**
- **Windows 2000/XP use TCP destination port 445**

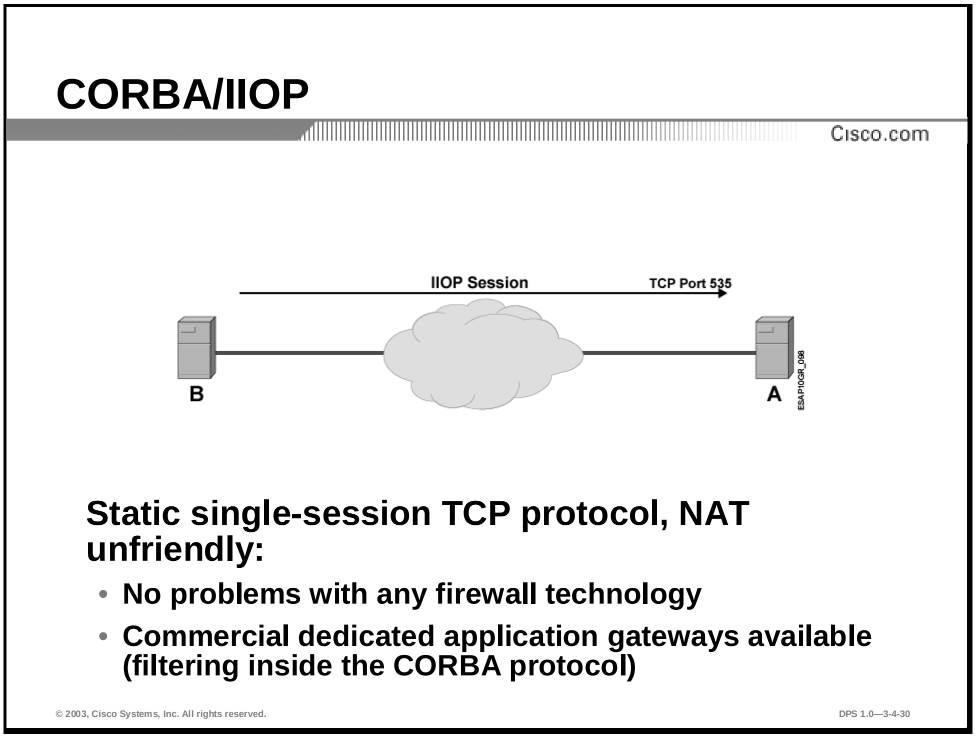
**Simple to patch through, but there is no insight into the RPC calls themselves:**

- **The protocol based on DCE/RPC, but extended**

DPS 1.0—3-4-29

# Windows RPC

Windows RPC is a simple network protocol that uses a single TCP session between the client and the server. The protocol is undocumented and no filtering mechanisms exist to look inside the RPC session to provide more access control. Some SPF vendors have built Microsoft RPC (extended DCE/RPC) in their stateful engines to support certain RPC applications. These open dynamic ports, negotiated over the RPC channel, such as Microsoft DCOM.



## CORBA/IIOP

Cisco.com

**Static single-session TCP protocol, NAT unfriendly:**

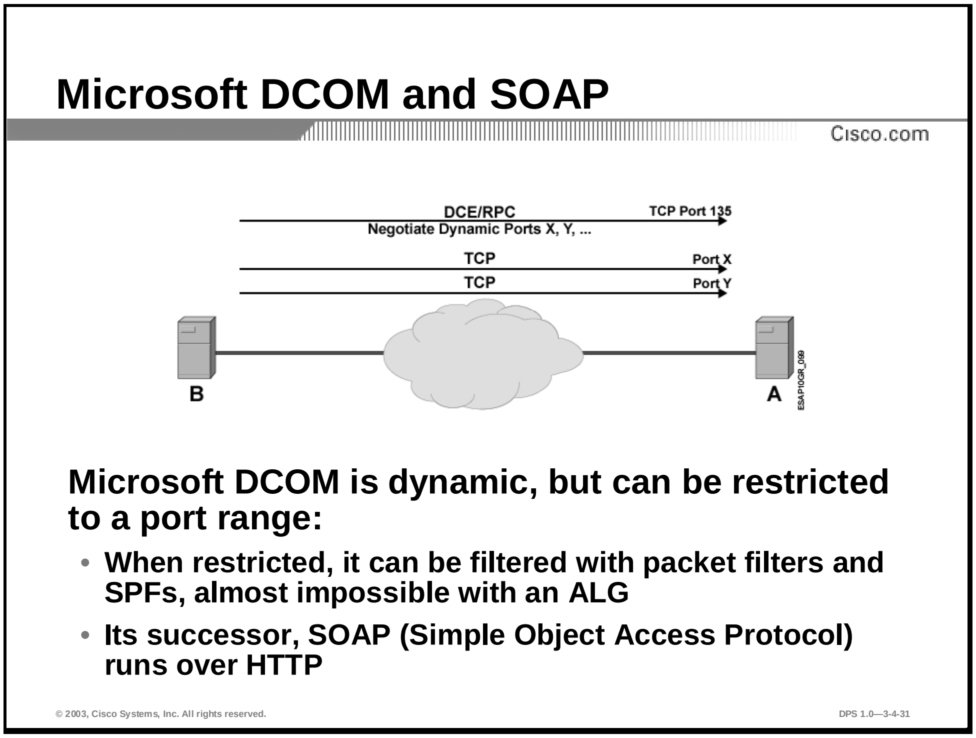
- **No problems with any firewall technology**
- **Commercial dedicated application gateways available (filtering inside the CORBA protocol)**

DPS 1.0—3-4-30

## CORBA and IIOP

The CORBA relies on the Internet Inter-Orb Protocol (IIOP) to access distributed objects on a network. UNIX environments often use CORBA as an equivalent to Windows-based DCOM. Various E-commerce and network management products use CORBA, often running over firewalls. The protocol itself is simple in terms of sessions (a single TCP channel), but is NAT-unfriendly.

Special application gateways for CORBA exist, which can provide granular CORBA access control, such as control over which user can invoke which remote object with which parameters.



## Microsoft DCOM and SOAP

Cisco.com

**Microsoft DCOM is dynamic, but can be restricted to a port range:**

- **When restricted, it can be filtered with packet filters and SPFs, almost impossible with an ALG**
- **Its successor, SOAP (Simple Object Access Protocol) runs over HTTP**

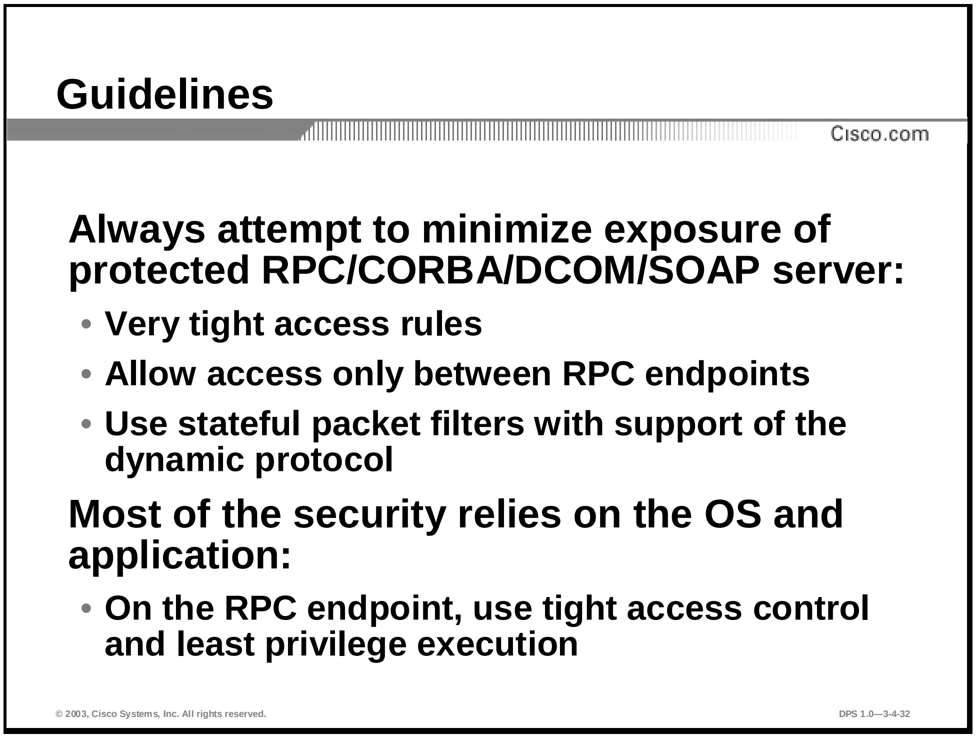
© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—3-4-31

# DCOM and SOAP

Microsoft systems use DCOM as the preferred distributed object access protocol, which in turn uses Microsoft RPC protocols. When establishing a session between two DCOM peers, the initiator first opens a normal Windows RPC channel (TCP, destination port 135) to the responder. Over that channel, the peers negotiate an additional session with a random high port for each DCOM instance. A network design can specify a port range for the negotiated connections by using the applicable published Windows registry key. Using such manually configured port ranges, a firewall designer can minimize the window of exposure by constructing firewall rules to permit connections in the configured port range, which should not be shared with any other applications. Many applications use DCOM, including Microsoft Transaction Server, Exchange, and many custom Internet applications.

Certain SPF vendors have built stateful intelligence for handling DCOM in their stateful engines, enabling them to snoop on the Microsoft RPC session in a similar way to UNIX RPC.

To address the problems that the dynamic nature of DCOM introduces with firewalls, Microsoft has developed a new method for distributed object access to be used within the .NET architecture. The new protocol, called Simple Object Access Protocol (SOAP) is implemented as XML messages inside a HTTP stream and can be tightly filtered by packet filtering firewalls. Some people believe that tunneling is harmful in the long run, as many things can be transferred inside HTTP, and there is a debate within the security community whether this is a good or a bad thing. Generally, tunneling should be used as a last resort, as no application control is possible within the tunneled protocol with generic application gateways.



## Guidelines

Cisco.com

**Always attempt to minimize exposure of protected RPC/CORBA/DCOM/SOAP server:**

- **Very tight access rules**
- **Allow access only between RPC endpoints**
- **Use stateful packet filters with support of the dynamic protocol**

**Most of the security relies on the OS and application:**

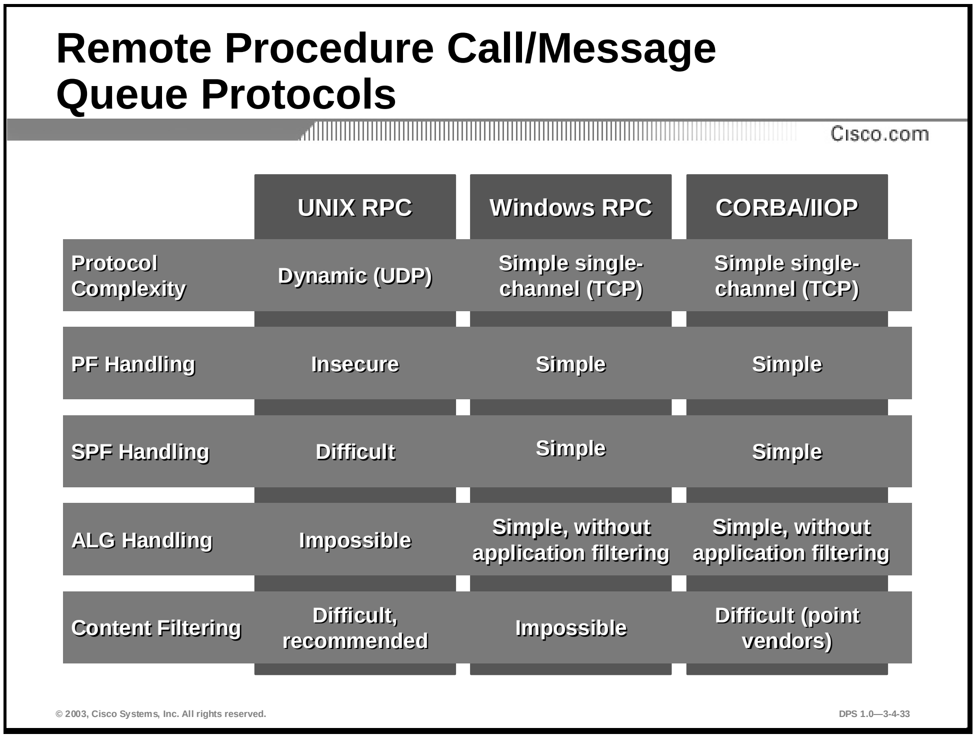
- **On the RPC endpoint, use tight access control and least privilege execution**

DPS 1.0—3-4-32

## Guidelines

Use extreme caution when passing RPC/distributed object protocols over a firewall. Always identify the authorized endpoints of communication, and always permit only minimal required connectivity. Packet filters are often not sufficient in this regard; therefore use SPFs, if they support the RPC/distributed object protocol in question.

However, most of the security needs to be deployed at the application layer. RPC/distributed object protocols usually implement some access control on their own, which should be used.

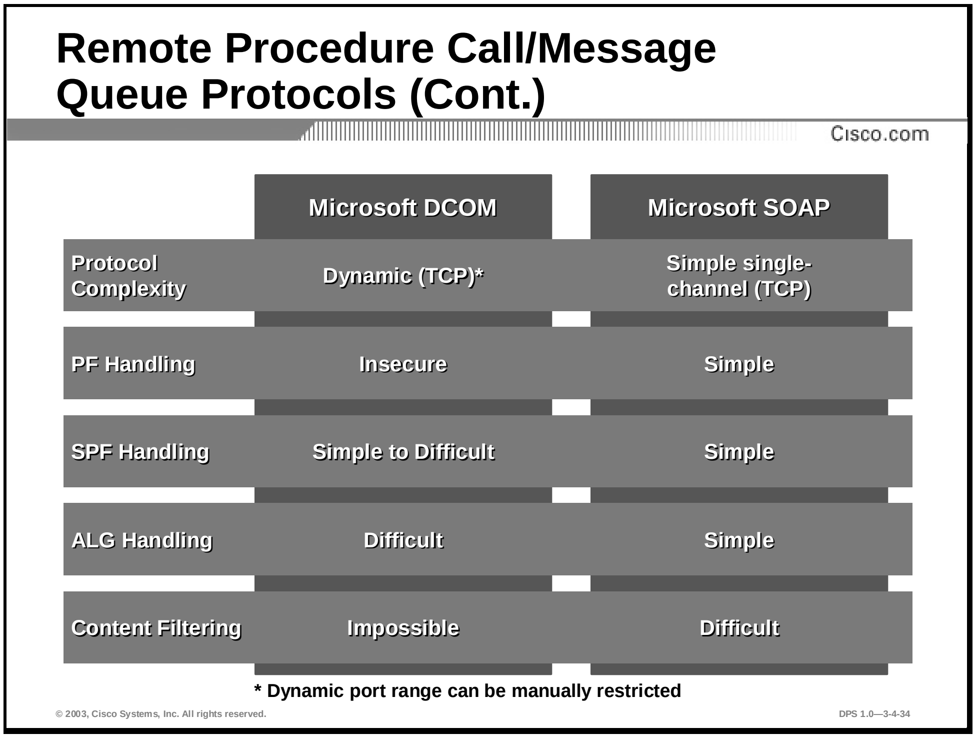


# Remote Procedure Call/Message Queue Protocols

Cisco.com

| | UNIX RPC | Windows RPC | CORBA/IIOP |
|---|---|---|---|
| Protocol Complexity | Dynamic (UDP) | Simple single-channel (TCP) | Simple single-channel (TCP) |
| PF Handling | Insecure | Simple | Simple |
| SPF Handling | Difficult | Simple | Simple |
| ALG Handling | Impossible | Simple, without application filtering | Simple, without application filtering |
| Content Filtering | Difficult, recommended | Impossible | Difficult (point vendors) |

DPS 1.0—3-4-33



## Remote Procedure Call/Message Queue Protocols (Cont.)

| | Microsoft DCOM | Microsoft SOAP |
|---|---|---|
| Protocol Complexity | Dynamic (TCP)* | Simple single-channel (TCP) |
| PF Handling | Insecure | Simple |
| SPF Handling | Simple to Difficult | Simple |
| ALG Handling | Difficult | Simple |
| Content Filtering | Impossible | Difficult |

* Dynamic port range can be manually restricted

DPS 1.0—3-4-34

## Practice

Q1) How is Microsoft DCOM handled by stateless packet filters?

A) by permitting Microsoft RPC and a range of dynamic high TCP ports

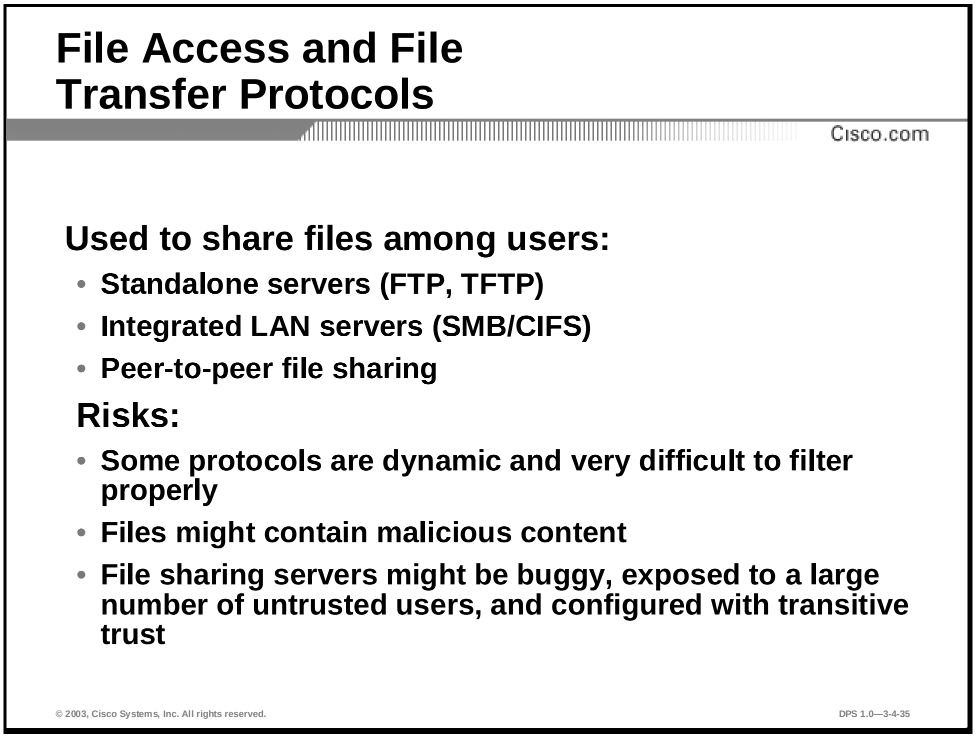
B) by permitting Microsoft RPC only

C) by permitting all high TCP ports only

D) by permitting SOAP access to TCP port 80

E) by permitting all TCP and UDP high ports

# File Transfer Protocols



## File Access and File Transfer Protocols

Used to share files among users:

- Standalone servers (FTP, TFTP)
- Integrated LAN servers (SMB/CIFS)
- Peer-to-peer file sharing

Risks:

- Some protocols are dynamic and very difficult to filter properly
- Files might contain malicious content
- File sharing servers might be buggy, exposed to a large number of untrusted users, and configured with transitive trust

DPS 1.0—3-4-35

## Objective

The section will enable the learner to explain the security properties of common file transfer protocols to select an appropriate technology to securely pass them over firewalls.

## Introduction

To transfer files between systems, file access and file transfer protocols are used. The file transfer can occur in a client-server fashion, standalone file transfer servers, such as FTP and TFTP servers, or integrated LAN servers for SMB/CIFS or NFS, or in a peer-to-peer network . Such protocols are often permitted through firewalls, and a need for secure handling is paramount.

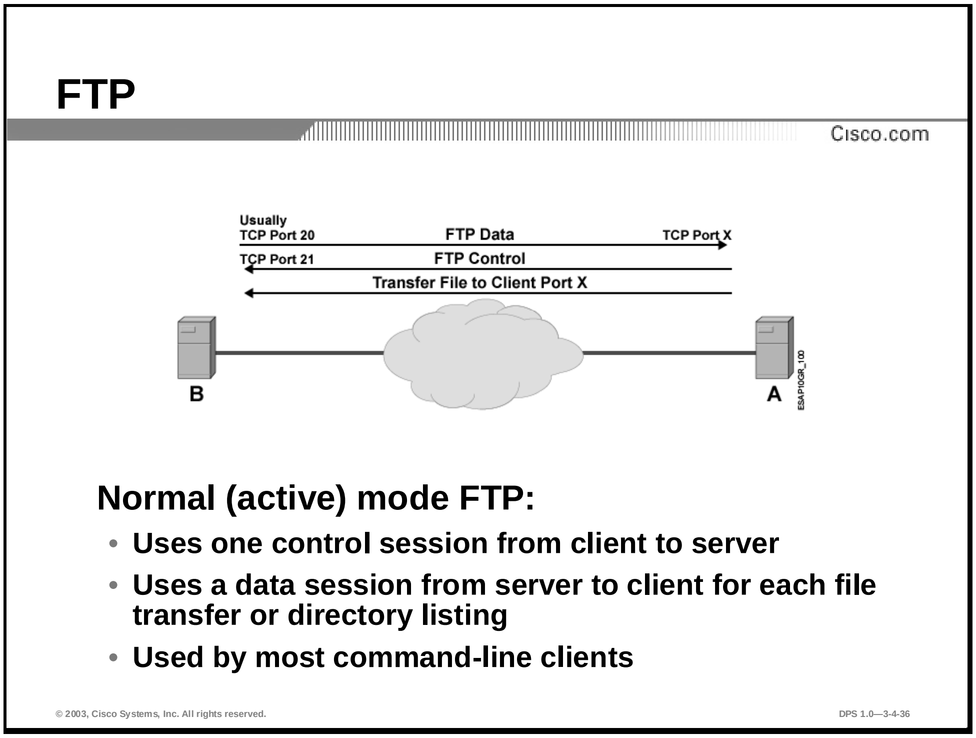
## File Transfer Protocols’ Refresher

The risks of file transfer are twofold:

- Exposed file servers might be buggy, allowing an attackers to access more that they should be allowed. In addition, the tight integration of file serving with the operating system might enable an attacker to escalate his privileges quickly.

- Files transferred between the application endpoints may contain malicious content, and may compromise the client, if executed.

Some file transfer/file access protocols are hard to filter by some firewall technologies. This forces an organization to permit more than the minimal possible access, violating the least privilege concept.



## FTP

**Normal (active) mode FTP:**

- **Uses one control session from client to server**
- **Uses a data session from server to client for each file transfer or directory listing**
- **Used by most command-line clients**

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—3-4-36

# FTP

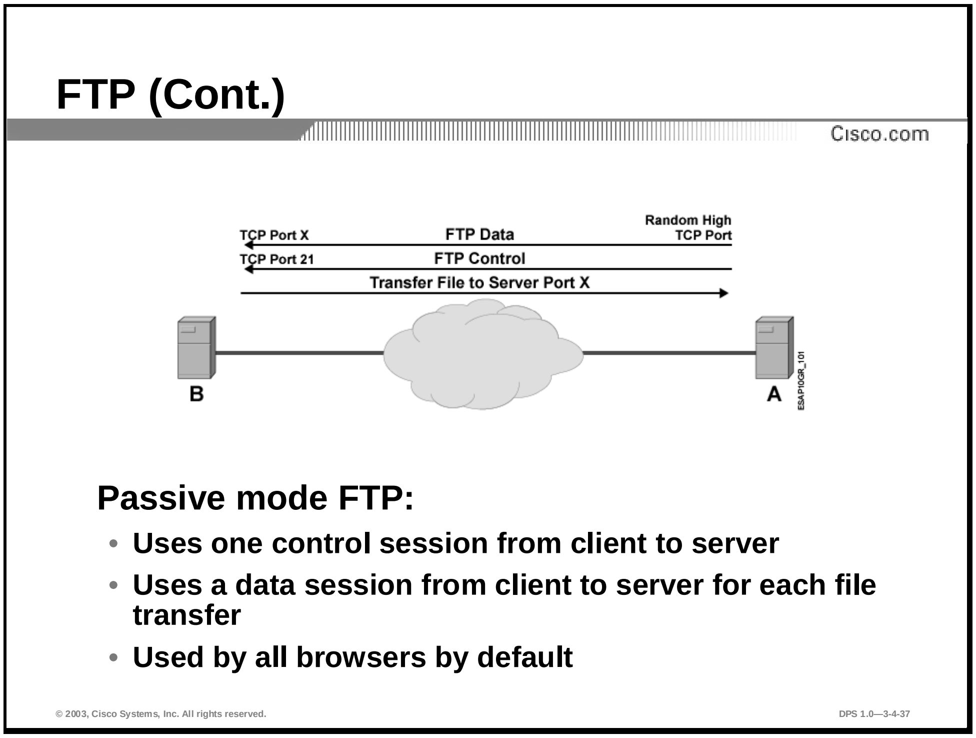
Originally designed in the 1970s, FTP is one of the legacy Internet protocols, and has caused significant problems with regard to firewalls. FTP could not complete its transactions over a single transport-layer session due to the limitations of ARPAnet transport protocol (then called NCP, not TCP). Instead, it opened a separate control session to authenticate the user, passed commands to the server, and used separate data sessions for any file or directory listing from the server.

FTP has two modes of operation, which differ in how the control and data sessions are opened.

In normal or active mode FTP, the client opens the control session to the server. The control session is always a TCP session with a random client port and port 21 on the server side. Over this session, the client authenticates and issues commands to transfer files.

The client initializes a local port and starts LISTENING on it, when a file is requested. Over the control connection, the client informs a server about its listening port, and the server opens a new (data) TCP connection to the client's listening port. The data connection has no fixed ports even though the server port is *often, but not always,* 20.

The classic packet filter cannot securely filter FTP as there is no static rule that can match data connections of FTP sessions. These connections have a random server port and a negotiated client port over the control session. To securely handle FTP, the firewall would need to snoop on the control connection to intercept the negotiations between the client and server, and only permit the necessary connections. Stateless packet filters, by design, do not offer such functionality.



## FTP (Cont.)

Cisco.com

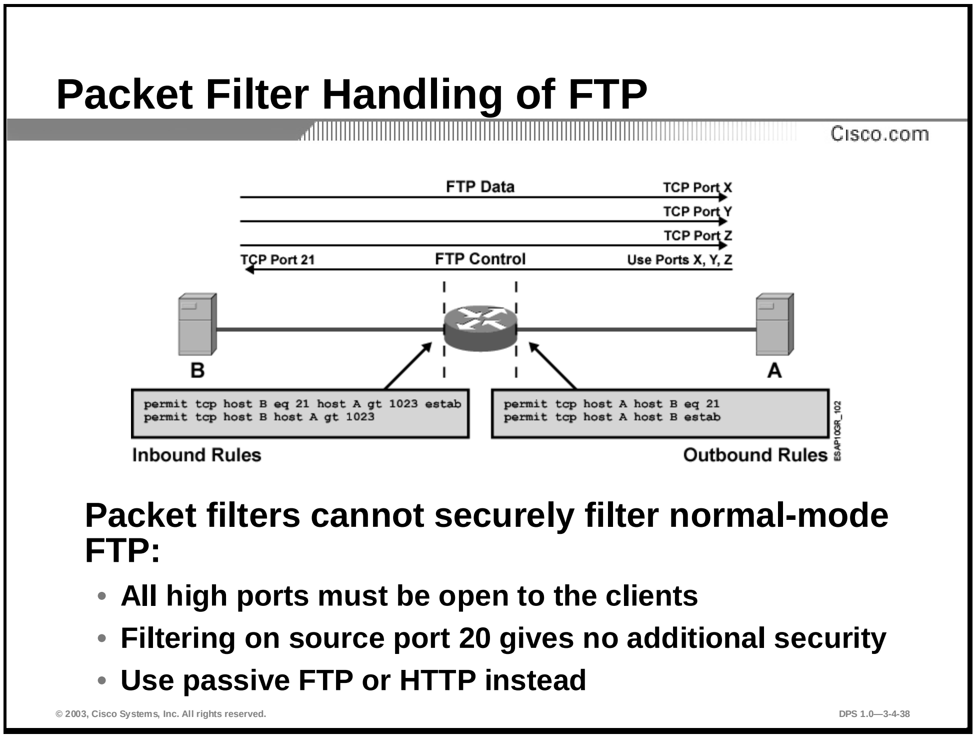
**Passive mode FTP:**

- **Uses one control session from client to server**
- **Uses a data session from client to server for each file transfer**
- **Used by all browsers by default**

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—3-4-37

To improve handling of FTP by stateless packet filters, passive FTP, or "firewall-friendly FTP", was designed. Passive FTP differs from the active version on the direction of the data sessions. In active FTP, the data sessions always open from the server to the client. In passive FTP, the data sessions open from the client to the server, in the same direction as the control connection. Although the ports are still negotiated, this behavior enables a packet filtering firewall to permit outbound FTP to the Internet with reasonable levels of security.



## Packet Filter Handling of FTP

Cisco.com

Inbound Rules

```
permit tcp host B eq 21 host A gt 1023 estab
permit tcp host B host A gt 1023
```

Outbound Rules

```
permit tcp host A host B eq 21
permit tcp host A host B estab
```

**Packet filters cannot securely filter normal-mode FTP:**

- **All high ports must be open to the clients**
- **Filtering on source port 20 gives no additional security**
- **Use passive FTP or HTTP instead**

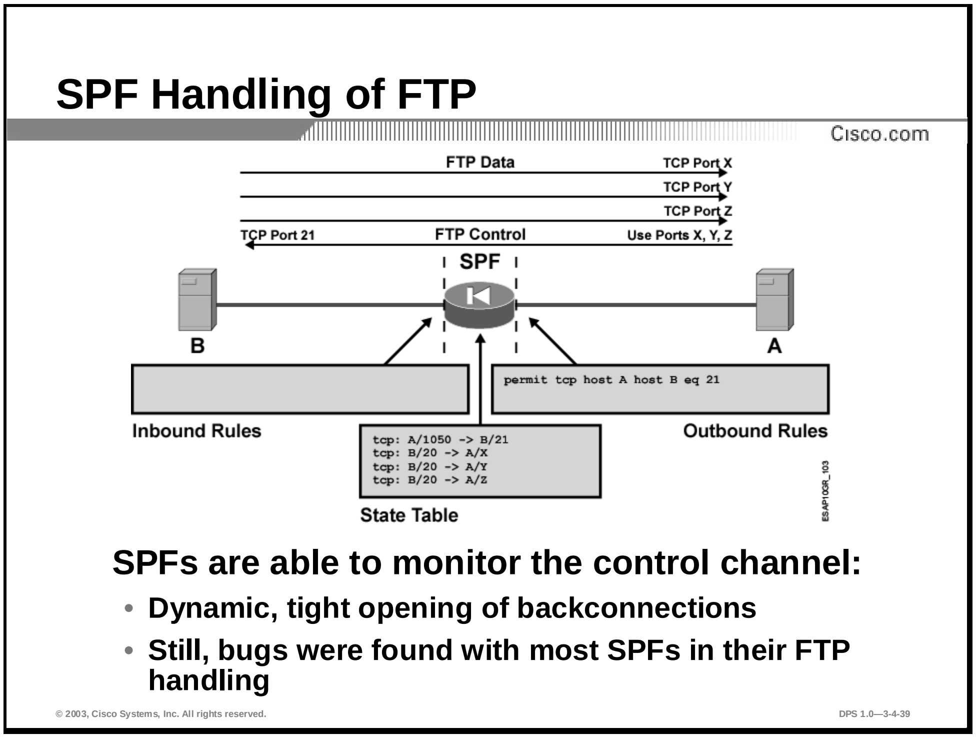
DPS 1.0—3-4-38

# Packet Filter Handling of FTP

Packet filters cannot filter active FTP properly as they require the following rules to support control and data connections:

- Permit all TCP packets from the client, on any high port, to the server, on port 21
- Permit all TCP packets from the server, on any port, to the client, on any high port (not only established traffic, as the data sessions initiate towards the client)

Such rules again open up a hole to the client, as an attacker can connect to any high port, where another sensitive and exploitable application might be running. In some cases the rules are tightened by permitting only the server port on the back-connection to be 20—this is NOT mandated by the FTP standard, and it provides no additional security. An attacker could initiate all connections from port 20 and succeed. Many scanning tools use source port 20 when scanning for TCP applications, to take advantage of misconfigured packet filtering firewalls.



## SPF Handling of FTP

```
permit tcp host A host B eq 21
```

```
tcp: A/1050 -> B/21
tcp: B/20 -> A/X
tcp: B/20 -> A/Y
tcp: B/20 -> A/Z
```

**SPFs are able to monitor the control channel:**

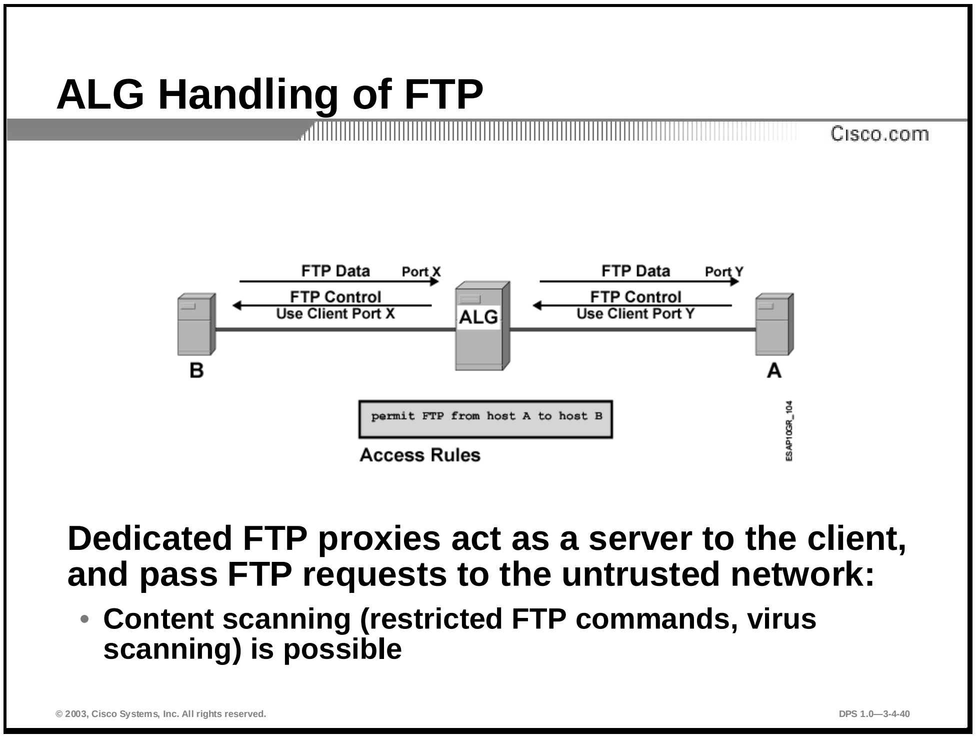
- **Dynamic, tight opening of backconnections**
- **Still, bugs were found with most SPFs in their FTP handling**

DPS 1.0—3-4-39

# SPF Handling of FTP

A stateful filter employs a great deal more intelligence to inspect the FTP session. When the control connection establishes, the SPF monitors the FTP protocol messages and looks for the port negotiation procedure inside them. The SPF then opens the firewall to allow the exact negotiated connection when the dynamic port is signaled between the client and the server.

Still, FTP is such a complex protocol that bugs were found in FTP handlers of almost all mainstream SPF implementations:



ALG Handling of FTP

Cisco.com

Dedicated FTP proxies act as a server to the client, and pass FTP requests to the untrusted network:

- Content scanning (restricted FTP commands, virus scanning) is possible

DPS 1.0—3-4-40

## ALG Handling of FTP

In a FTP, the ALG acts as a broker between the original FTP client, and the original FTP server, acting as an FTP server to the client, and as an FTP client to the server. The gateway terminates two control sessions—one with the client, and one with the server. Over the control session with the client, the gateway receives FTP commands and passes them, possibly changed, to the server. Because FTP ALGs implement the full FTP protocol to relay between the client and server, they can filter on objects inside the FTP protocol.

## Examples

An FTP ALG might deny certain users to upload files (the FTP protocol **put** command), but allow them to download files (the FTP protocol **retr** command). The gateway might also:

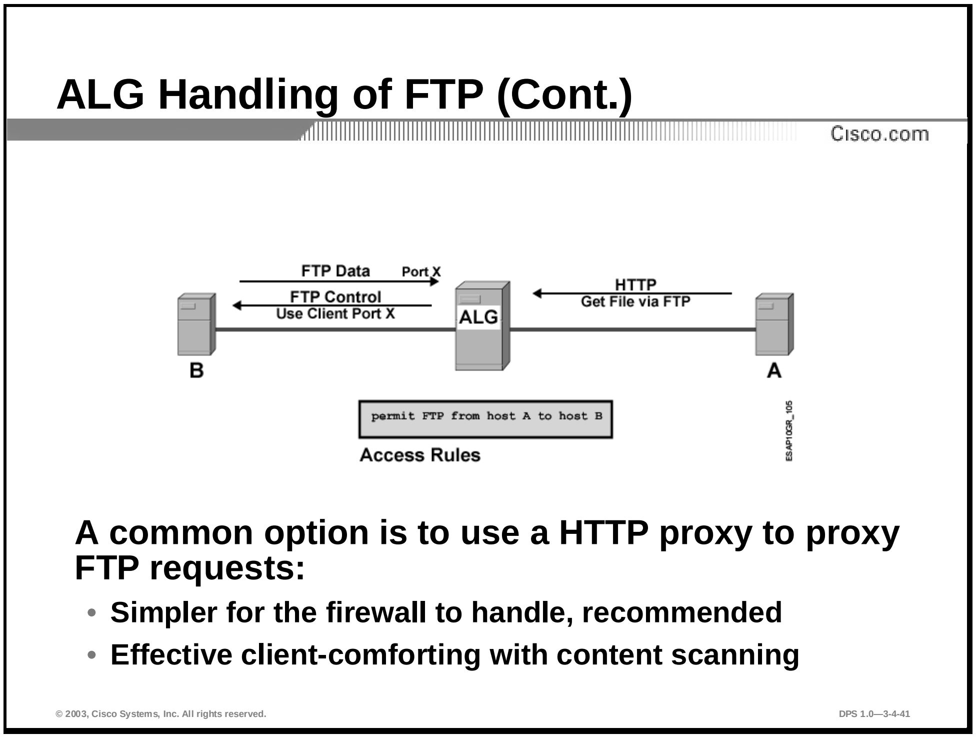
**Step 1** Relay the file download command from the client to the server

**Step 2** Wait for the server to open a data connection to the gateway

**Step 3** Receive the file

**Step 4** Scan the file for viruses

**Step 5** Relay the file to the client



## ALG Handling of FTP (Cont.)

```
permit FTP from host A to host B
```

Access Rules

**A common option is to use a HTTP proxy to proxy FTP requests:**

- **Simpler for the firewall to handle, recommended**
- **Effective client-comforting with content scanning**

HTTP-to-FTP protocol translation, which is supported by most HTTP proxies, is another option for application-layer proxying of FTP. In this case:

**Step 1** The client opens a connection to the ALG over HTTP

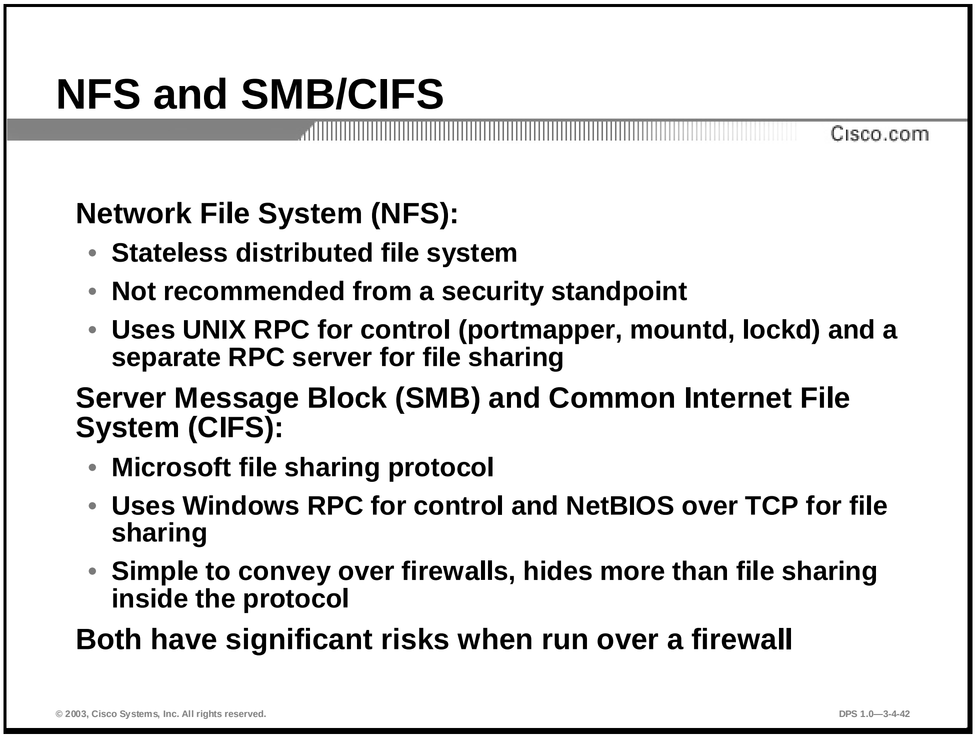
**Step 2** The client specifies that a FTP URL should be opened by the gateway

**Step 3** The gateway starts a FTP connection to the destination site

**Step 4** The gateway transfers the file

**Step 5** The gateway returns the file to the client over the HTTP connection

This method does not require any FTP functionality on the client, and is one of the preferred methods of FTP content filters such as virus scanners. While the application gateway downloads the file and checks it for viruses, the client is comforted by the application gateway over the web session until the download completes and is virus-scanned. The file is then transferred to the client over HTTP.



## NFS and SMB/CIFS

Network File System (NFS):

- Stateless distributed file system
- Not recommended from a security standpoint
- Uses UNIX RPC for control (portmapper, mountd, lockd) and a separate RPC server for file sharing

Server Message Block (SMB) and Common Internet File System (CIFS):

- Microsoft file sharing protocol
- Uses Windows RPC for control and NetBIOS over TCP for file sharing
- Simple to convey over firewalls, hides more than file sharing inside the protocol

Both have significant risks when run over a firewall

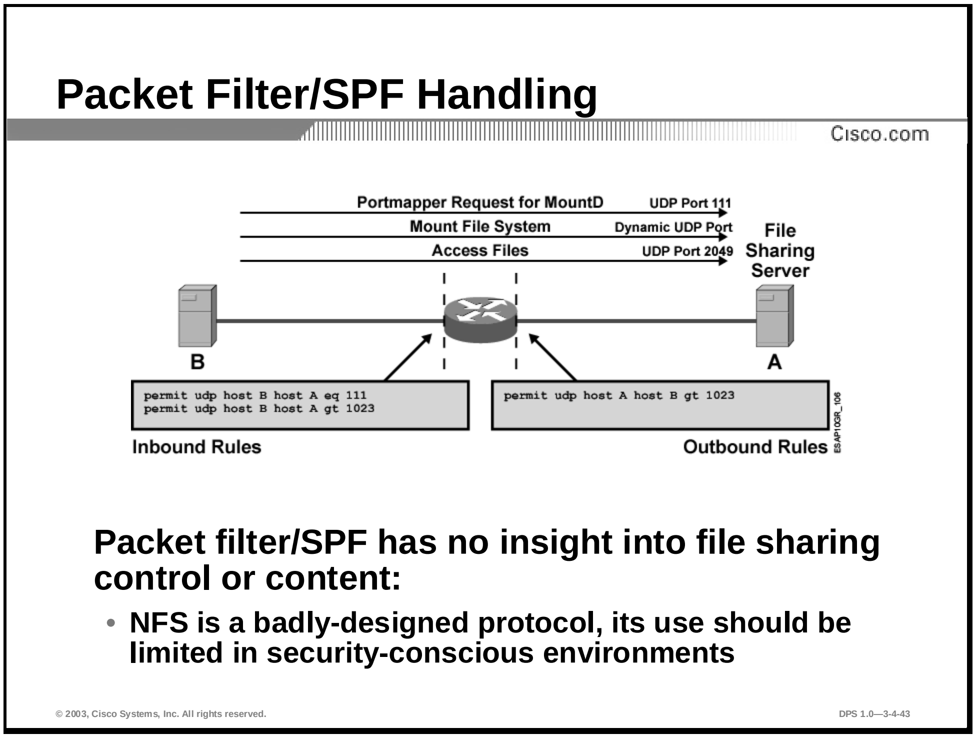
DPS 1.0—3-4-42

# NFS and SMB/CIFS File Access Protocols

The Network File System (NFS) is used primarily with UNIX systems, where it presents the preferred method of sharing files from or between UNIX systems. It is not recommended to use NFS over untrusted networks as it is not a well-designed protocol in terms of security.

On the network, NFS uses several RPC helper applications, for example, the mount daemon, the lock daemon, using UNIX RPC mechanisms. Additionally, the main NFS process listens on a well-known UDP (sometimes also TCP) port of 2049.

The Server Message Block (SMB) protocol, later renamed to Common Internet File System (CIFS), is the file sharing protocol used by Microsoft Windows platforms. It relies on NetBIOS over TCP/IP for transport and uses Windows RPC messages for control functions. SMB/CIFS looks simple on the network (multiple simple TCP sessions), but a lot of other functionality besides file sharing is available through SMB. Therefore, permitting SMB/CIFS opens up access to additional Windows server functionality, which can be compromised by an attacker.



## Packet Filter/SPF Handling

Cisco.com

```
permit udp host B host A eq 111
permit udp host B host A gt 1023
```

```
permit udp host A host B gt 1023
```

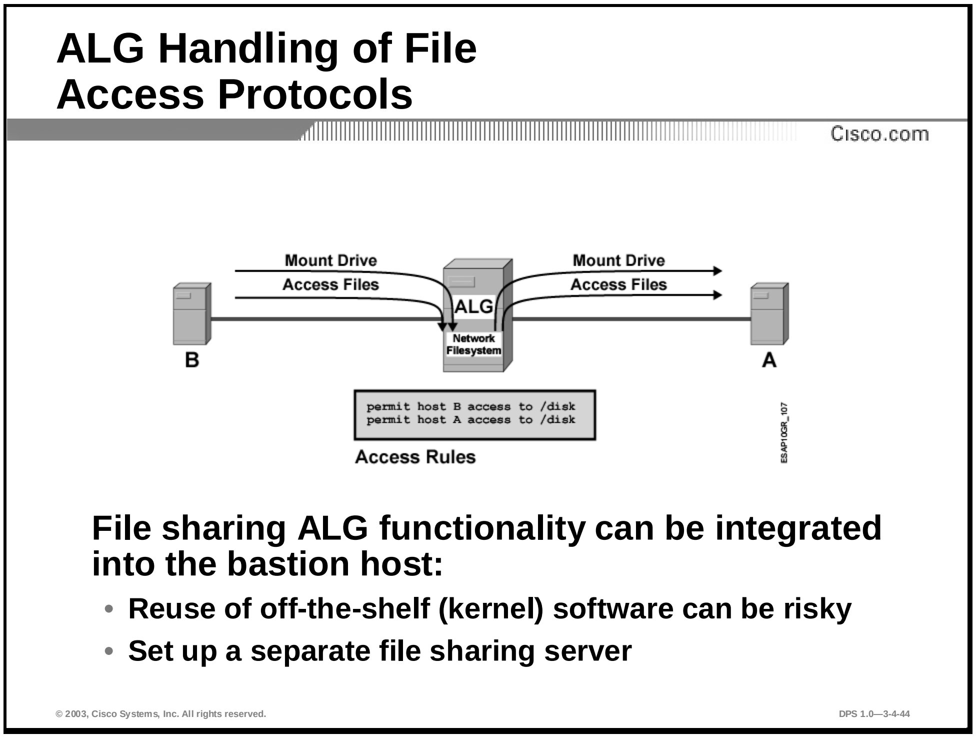
**Packet filter/SPF has no insight into file sharing control or content:**

- **NFS is a badly-designed protocol, its use should be limited in security-conscious environments**

DPS 1.0—3-4-43

## Packet Filter/SPF Handling of NFS

Over a firewall, SPFs, which support UNIX RPC, can support NFS. Classic packet filtering requires wide-open access rules, and pure NFS ALGs are not commercially available.



Mount Drive
Access Files
ALG
Network Filesystem
Mount Drive
Access Files
B
A
ESAP10GR_107

## ALG Handling of File Access Protocols

Cisco.com

```
permit host B access to /disk
permit host A access to /disk
```

Access Rules

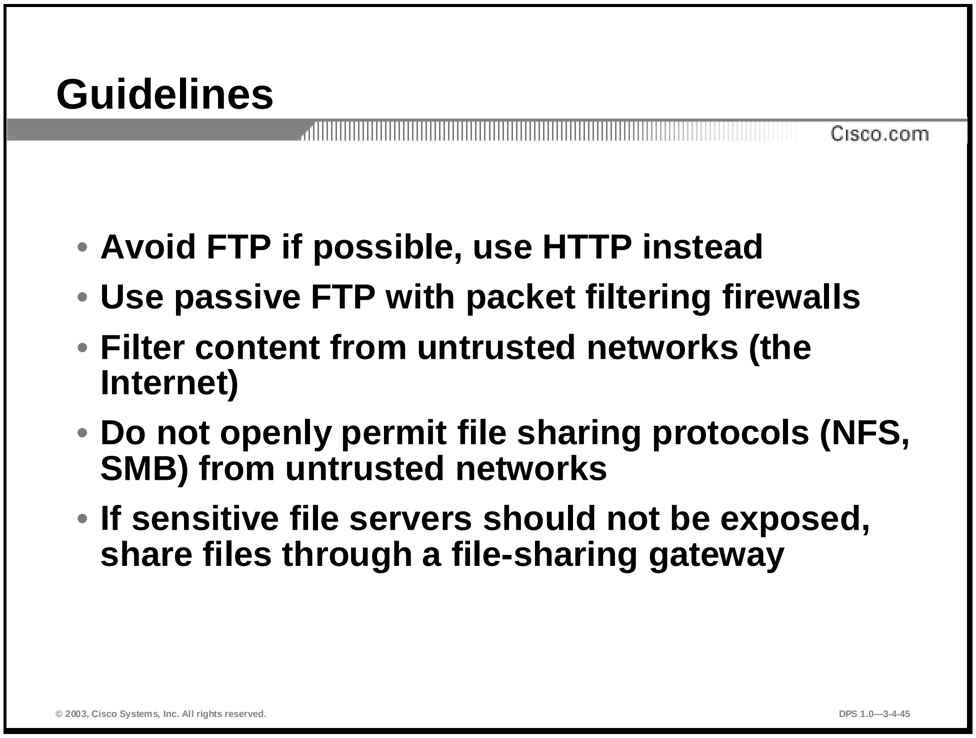
**File sharing ALG functionality can be integrated into the bastion host:**

- **Reuse of off-the-shelf (kernel) software can be risky**
- **Set up a separate file sharing server**

DPS 1.0—3-4-44

## ALG Handling of File Access Protocols

Both NFS and SMB/CIFS can be “proxied” using a simple homemade application gateway. A firewall designer can choose to place a “shared server” in the firewall system, the server being accessible by both the untrusted and the trusted side. Both sides can mount network volumes on the shared server, exchanging files between them. Optionally, the shared server can also be dual-homed to the two networks, and authenticate users from both networks.



## Guidelines

- Avoid FTP if possible, use HTTP instead
- Use passive FTP with packet filtering firewalls
- Filter content from untrusted networks (the Internet)
- Do not openly permit file sharing protocols (NFS, SMB) from untrusted networks
- If sensitive file servers should not be exposed, share files through a file-sharing gateway

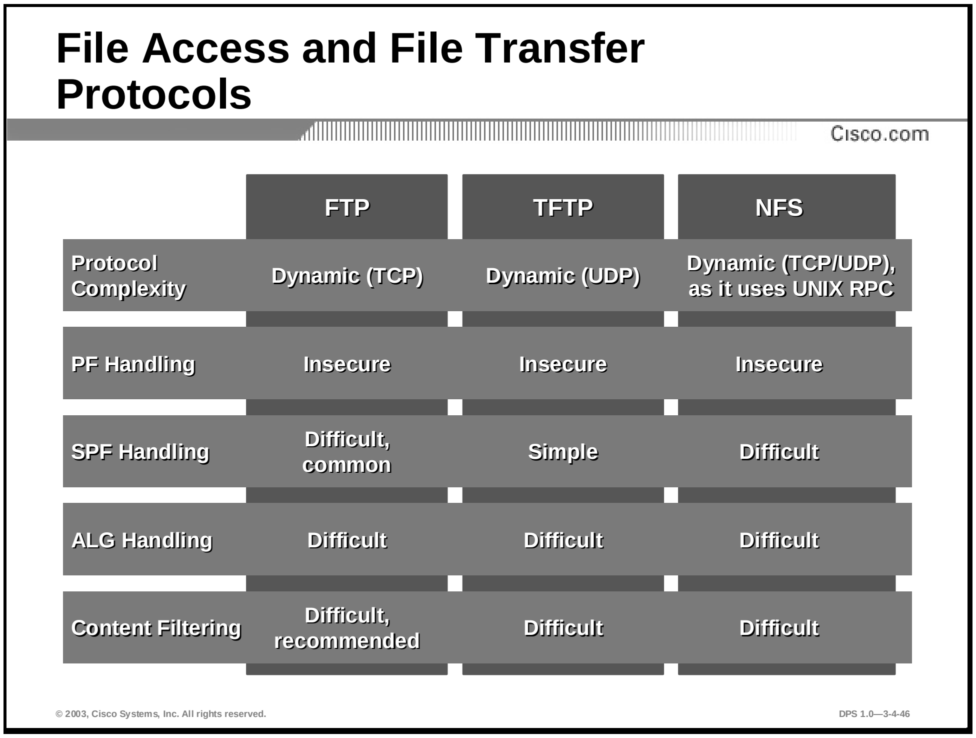
DPS 1.0—3-4-45

## Guidelines

In general, avoid FTP as a file transfer method in secure environments. It is too complex, and has traditionally been both buggy on the server side, and mishandled by firewalls. Use of HTTP is suggested instead. If the use of FTP is necessary, use passive FTP with packet filters to minimize internal network exposure.

Ensure the filtering of data inside file transfer/file access protocols, if it is coming from untrusted sources or unprotected over untrusted networks. Virus scanning, which can work with file transfer and file sharing, is a prime example of a content-filtering technology.

The two best-known file access protocols, NFS and SMB, are both too complex and historically too unpredictable to be trusted in a secure environment. Set up virtual private networks (VPNs) to secure their transmission, or use exposed untrusted file-sharing gateways to minimize damage in the case of compromise.

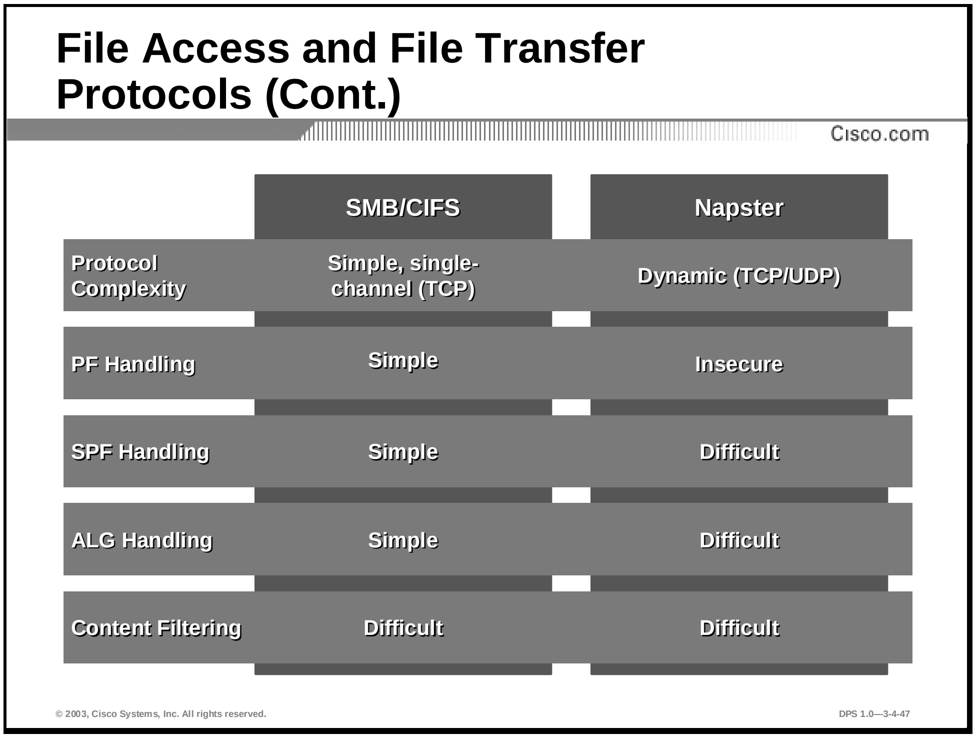


# File Access and File Transfer Protocols

Cisco.com

| | FTP | TFTP | NFS |
|---|---|---|---|
| Protocol Complexity | Dynamic (TCP) | Dynamic (UDP) | Dynamic (TCP/UDP), as it uses UNIX RPC |
| PF Handling | Insecure | Insecure | Insecure |
| SPF Handling | Difficult, common | Simple | Difficult |
| ALG Handling | Difficult | Difficult | Difficult |
| Content Filtering | Difficult, recommended | Difficult | Difficult |

DPS 1.0—3-4-46



## File Access and File Transfer Protocols (Cont.)

Cisco.com

| | SMB/CIFS | Napster |
|---|---|---|
| Protocol Complexity | Simple, single-channel (TCP) | Dynamic (TCP/UDP) |
| PF Handling | Simple | Insecure |
| SPF Handling | Simple | Difficult |
| ALG Handling | Simple | Difficult |
| Content Filtering | Difficult | Difficult |

DPS 1.0—3-4-47

# Practice

Q1) Which of the following FTP modes can be handled securely by a classic (stateless) packet filter?

A) normal FTP

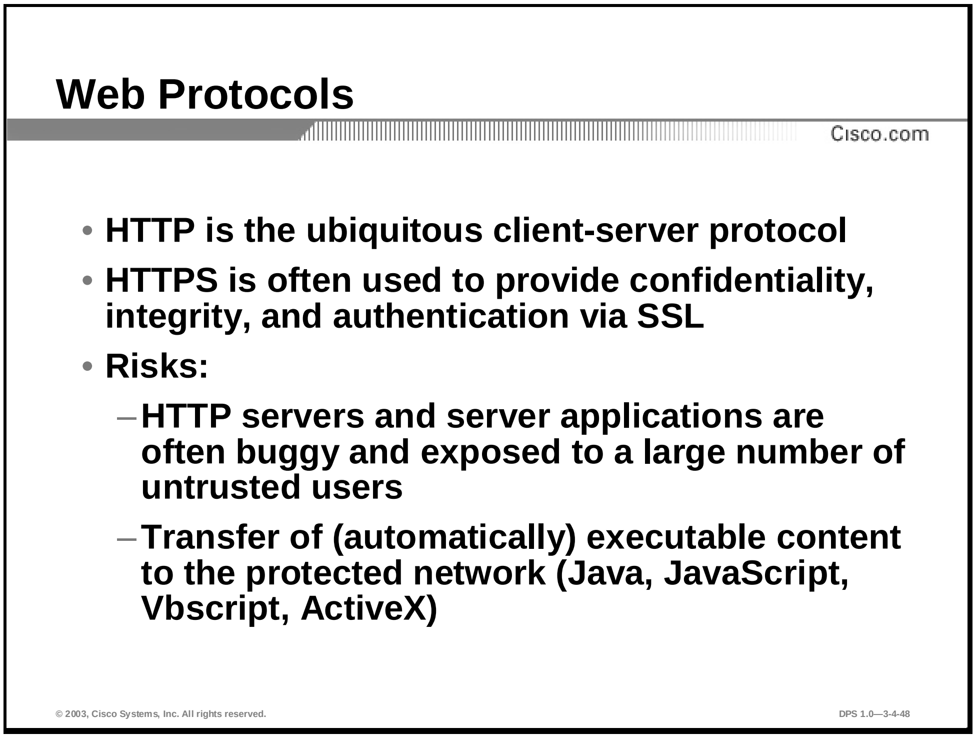
B) active FTP

C) passive FTP

D) any FTP mode

E) FTP can never be handled securely by a packet filter

# Web Protocols



## Web Protocols

Cisco.com

- HTTP is the ubiquitous client-server protocol
- HTTPS is often used to provide confidentiality, integrity, and authentication via SSL
- Risks:
  - HTTP servers and server applications are often buggy and exposed to a large number of untrusted users
  - Transfer of (automatically) executable content to the protected network (Java, JavaScript, Vbscript, ActiveX)

## Objective

The section will enable the learner to explain the security properties of web protocols to select an appropriate technology to securely pass them over firewalls.

## Introduction

Web protocols are ubiquitous in the firewall world, as many client-server and backend applications rely on them to pass data between application endpoints. Such protocols are often permitted through firewalls, and a need for secure handling is paramount.

## Web Protocols' Refresher

HTTP is a simple text protocol used to transfer web objects between web servers and web clients. The protocol has two main versions:

- **HTTP 1.0:** The basic protocol supported by all browsers and servers.

- **HTTP 1.1:** An extended protocol, which is already widely supported. It adds multiple performance enhancements to HTTP 1.0, such as pipelining of requests and persistent connections.

HTTP enables a client to request content from a server. On the network, the session is a simple single TCP session with the destination port of 80. The security risks associated with HTTP are:

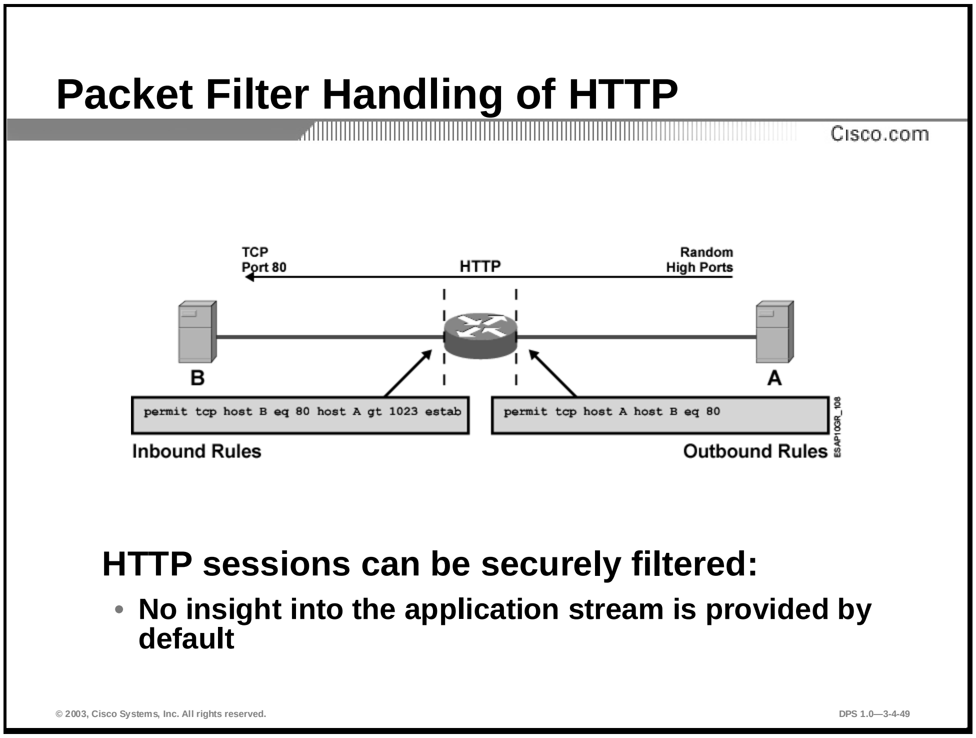
- Web servers and their applications are complex and are most often attacked through their HTTP interface

- Clients browsing untrusted web servers can download executable code, which might be malicious, when automatically executed on the client

A firewall, which performs strict content filtering on HTTP connections, can mitigate both these risks. Such a firewall examines every HTTP request and response, and evaluates whether it presents a violation of the defined policy.

The use of content filtering to protect web servers is not usually used: modern applications are much too complex to be able to agree on what the firewall would treat as acceptable. Many sites use Secure Socket Layer (SSL) cryptographic protection between clients and servers, whose encryption would defeat any application-layer filter in between.

Filtering of data flowing to clients is common. Organizations filter data inside HTTP, based on the type of data (movies, executables, etc.) or the URL of the requestor. Organizations also perform anti-virus checking.



## Packet Filter Handling of HTTP

```
permit tcp host B eq 80 host A gt 1023 estab
```

```
permit tcp host A host B eq 80
```

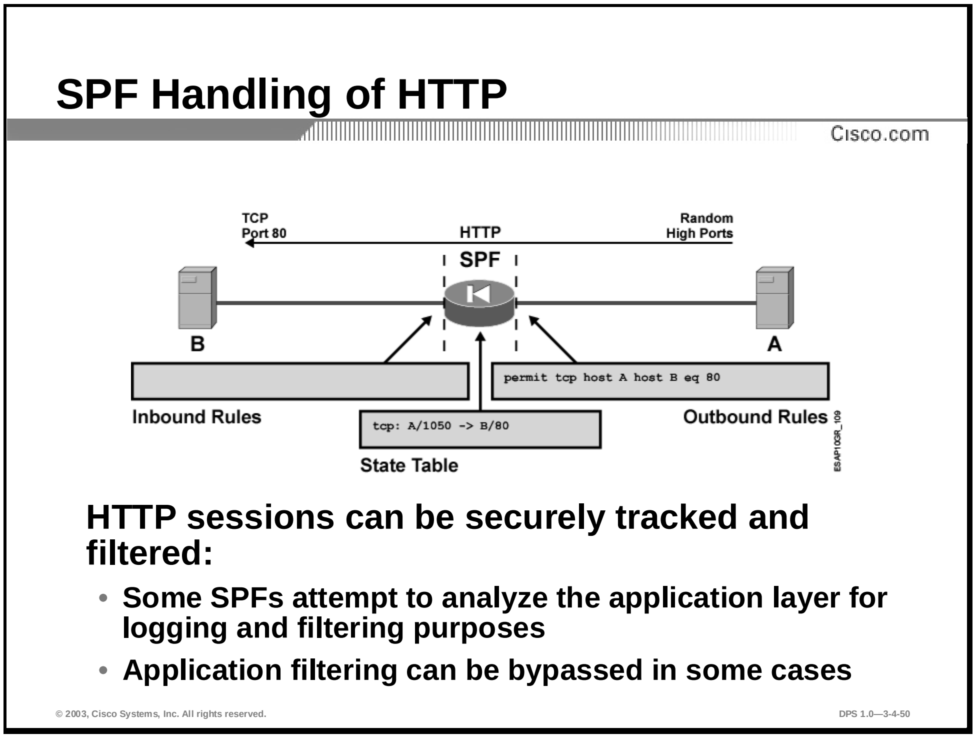
**HTTP sessions can be securely filtered:**

- **No insight into the application stream is provided by default**

DPS 1.0—3-4-49

## Packet Filter Handling of HTTP

A packet filter handles HTTP like any other single-channel TCP session. It can securely convey it between two networks with tight filtering but is unable to analyze the application layer data inside the HTTP stream. If application analysis is not required, a classic packet filter usually provides sufficiently robust filtering of HTTP between networks.



## SPF Handling of HTTP

**HTTP sessions can be securely tracked and filtered:**

- **Some SPFs attempt to analyze the application layer for logging and filtering purposes**
- **Application filtering can be bypassed in some cases**

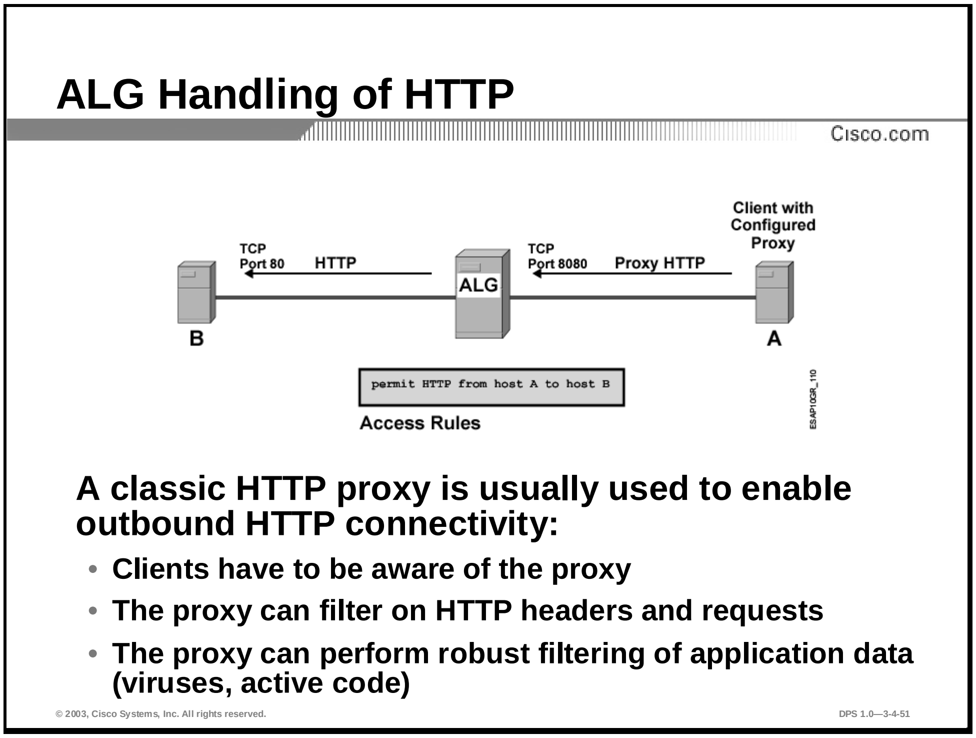
DPS 1.0—3-4-50

# SPF Handling of HTTP

SPFs provide additional inspection by being session aware, they can also track session state and protect against spoofed packets by checking TCP sequence numbers. Some SPFs have some insight in the HTTP application layer, as they can peek into the contents of individual packets. In this way, an SPF can look for specific commands or patterns in the application stream, and perform some basic application-layer access control, such as Java or ActiveX blocking, or URL filtering and accounting.

**Note** The PIX Firewall has Java and ActiveX blocking functionality. The IOS Firewall can block Java applets only.

**Note** Application-layer awareness of SPFs is usually limited to per-packet analysis. SPFs analyze each packet’s contents independently; therefore a command spanning multiple packets will not be processed properly. For a robust application-layer access control use pure application gateways.



## ALG Handling of HTTP

Cisco.com

**A classic HTTP proxy is usually used to enable outbound HTTP connectivity:**

- **Clients have to be aware of the proxy**
- **The proxy can filter on HTTP headers and requests**
- **The proxy can perform robust filtering of application data (viruses, active code)**

DPS 1.0—3-4-51

# ALG Handling of HTTP

The HTTP protocol also introduces the concept of HTTP proxies, which are application gateways, not necessarily used for the purpose of increasing security. Organizations also use HTTP proxies solely as a caching solution or a replacement for NAT.

An HTTP proxy is a software package that listens on a configured port for client requests. Such an HTTP proxy provides a service to local clients. The client knows about the proxy and configures itself to pass all its requests through the proxy. When the client needs to access a URL, it opens a connection to the HTTP proxy and submits a request for a URL, using a slightly modified HTTP protocol. The proxy then impersonates the client and connects to the outside server using HTTP, downloads the URL object, and passes the contents to the inside client, performing filtering in the process.

The non-transparent proxy requires the client to be aware of it, which means that the client explicitly configures the proxy parameters in its software. Modern browsers can automatically detect a proxy via an administrator-supplied configuration script. Stateful firewalls, such as the PIX Firewall, do not require any end-station reconfiguration, and the associated administrative effort.

In a classic HTTP proxy setup, the proxy

**Step 1** Accepts HTTP proxy sessions from the client, which contain the client’s request for an external object (uniform resource identifier [URI])

**Step 2** Initiates HTTP sessions to the external server, passes the clients request, possibly filtered, to the external server and receives the server response

**Step 3** Filters the received data and forwards the filtered response to the client

HTTP proxies can perform robust and detailed filtering of application data within the HTTP stream. An HTTP proxy might filter the following HTTP objects:

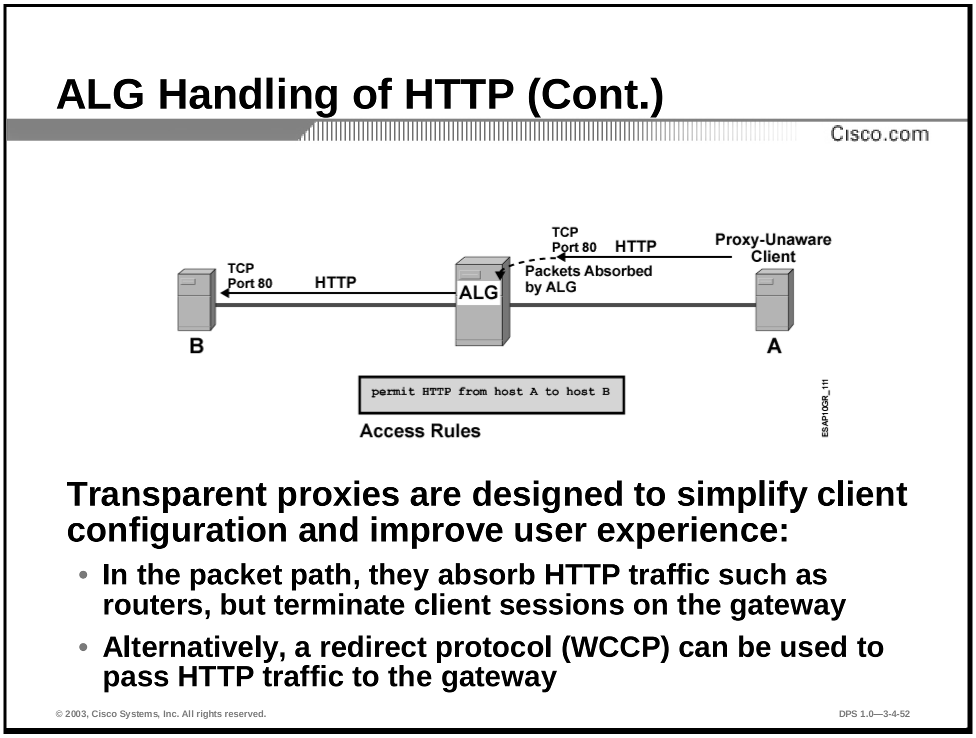
- The source or destination of the request, which is the simplest filtering method. The client's access to a specific server is denied, which might be identified by its IP address or HTTP hostname field.

- The client request, where the proxy might deny access to specific URIs (URL filtering) or specific patterns within a URI, for example, no access to .GIF files.

- The type of data returned by the server, where the proxy inspects the Multipurpose Internet Mail Extension (MIME) content type of the response, which indicates the data type. For example, the proxy might not allow any content with a MIME type of "movie/avi" to be forwarded to the clients to conserve bandwidth, or any content with a MIME type of "application/msword" to be forwarded to clients to avoid Word macro viruses.

- The data itself, where the proxy might perform content filtering to eliminate active content (JavaScript, VBScript, Java, ActiveX, ShockWave, etc) or scan for viruses inside data.

The benefit of a classic HTTP proxy, besides the possibility of granular access control inside HTTP, is that clients on the protected network do not need any DNS information or routing towards the Internet. All they need is to contact the proxy IP address and pass their request to the proxy.



## ALG Handling of HTTP (Cont.)

Cisco.com

**Transparent proxies are designed to simplify client configuration and improve user experience:**

- **In the packet path, they absorb HTTP traffic such as routers, but terminate client sessions on the gateway**
- **Alternatively, a redirect protocol (WCCP) can be used to pass HTTP traffic to the gateway**

DPS 1.0—3-4-52

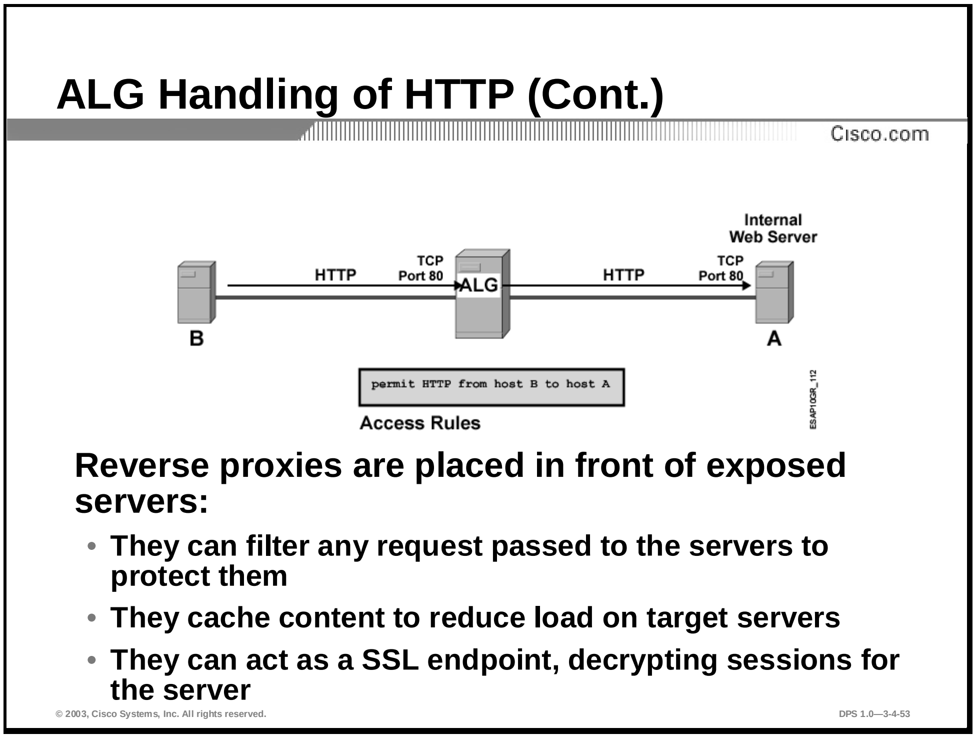
# Transparent HTTP ALGs

A transparent proxy (ALG) simplifies deployment by not requiring the clients to be aware of the proxy. The client opens a normal connection directly to the Internet, with the transparent HTTP proxy somewhere near the packet path. The proxy then automatically absorbs the session using one of two methods:

- The proxy might have a changed TCP/IP stack, where it appears as a router to the network. It terminates incoming HTTP sessions and passes them to the built-in HTTP ALG, which initiates a new HTTP session to the destination server. The operation of the HTTP engine is the same as the classic HTTP proxy: only the client connections are automatically terminated without the need for a proxy protocol between the client and the proxy. Although the proxy needs to be directly in the packet path or be assisted by a Layer 4 (L4) switch to redirect raw HTTP traffic to it.

- The proxy might cooperate with a network device using an offload protocol, such as Web Cache Control Protocol (WCCP) or Content Vectoring Protocol (CVP). In this case a network device redirects the original HTTP session to the proxy, which does need to be directly in the packet path.

The main benefit of the transparent proxy lies in its transparency for the end users. The downside of transparent proxies is that they require the clients to have full DNS information and routing to the Internet. Otherwise, it has the same functionality as a classic proxy.



## ALG Handling of HTTP (Cont.)

**Reverse proxies are placed in front of exposed servers:**

- **They can filter any request passed to the servers to protect them**
- **They cache content to reduce load on target servers**
- **They can act as a SSL endpoint, decrypting sessions for the server**

DPS 1.0—3-4-53

## Reverse HTTP Proxies

A reverse HTTP proxy is a proxy that provides services to nearby servers. Usually, the role of a reverse proxy is to lower the load of real servers by serving popular content instead of the main server. A reverse proxy is a system, to which all clients connect and then request data. If the reverse proxy has the data available locally, it answers the request without forwarding it. If the reverse proxy is configured to forward requests, it contacts the correct server, requests the file, and relays it to the client. The client in this case does not know about the existence of the proxy and does not have to be aware of it.

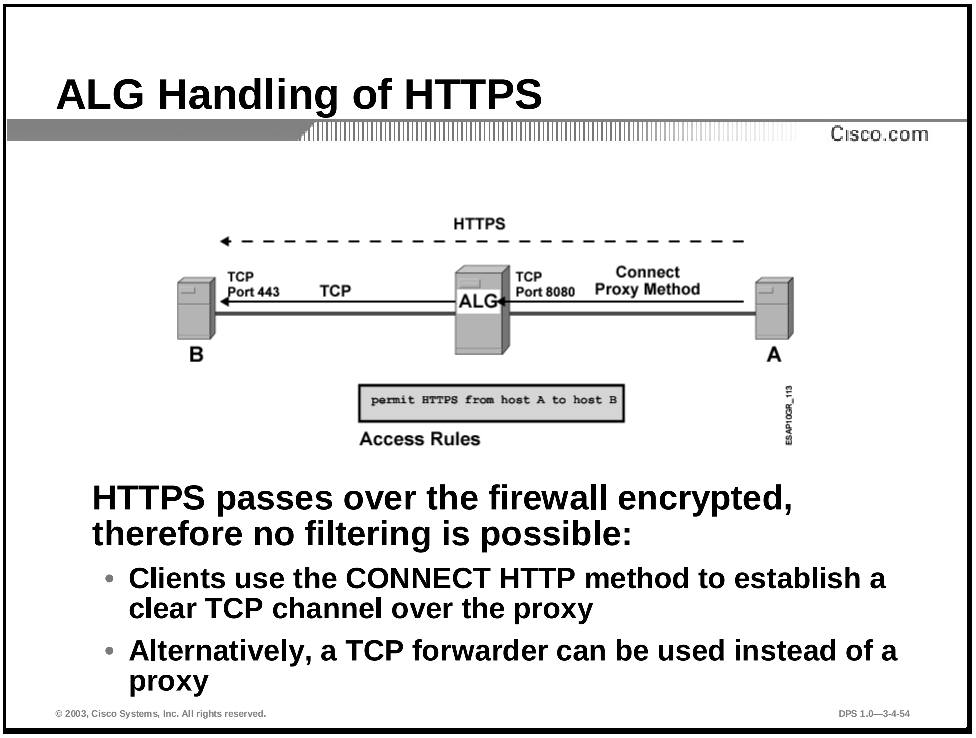
A reverse proxy handles incoming HTTP sessions to protected servers. The reverse proxy poses as an HTTP server to the client, and the client connects to the reverse proxy, believing it has connected to the target server. The client then passes an HTTP request, containing the URL and the “hostname” parameter, the latter indicating which server it believes it is connecting to.

The reverse proxy then inspects and perhaps filters the request, and forwards the request to the appropriate server using the URL and possibly also the “hostname” parameter. However, the “hostname” parameter does not need to be honored. The reverse proxy might elect to forward requests solely based on the URL of the request. For example, all URLs beginning with “/univercd/” might be forwarded to the documentation web server, while all other requests might be forwarded to the main web server.

The benefit of the reverse proxy is its ability to filter data from the clients, and hence protect the servers, if suitable application filtering rules can be set up. For example, the reverse proxy might now allow any content to be uploaded to the protected servers, or might deny the outside users to access specific protected areas of the protected HTTP server.

## Example

The Cisco Content engine can also act as a reverse proxy and be used in this scenario.



ALG Handling of HTTPS

Cisco.com

```
permit HTTPS from host A to host B
```

Access Rules

HTTPS passes over the firewall encrypted, therefore no filtering is possible:

- Clients use the CONNECT HTTP method to establish a clear TCP channel over the proxy
- Alternatively, a TCP forwarder can be used instead of a proxy

DPS 1.0—3-4-54

## ALG Handling of HTTPS

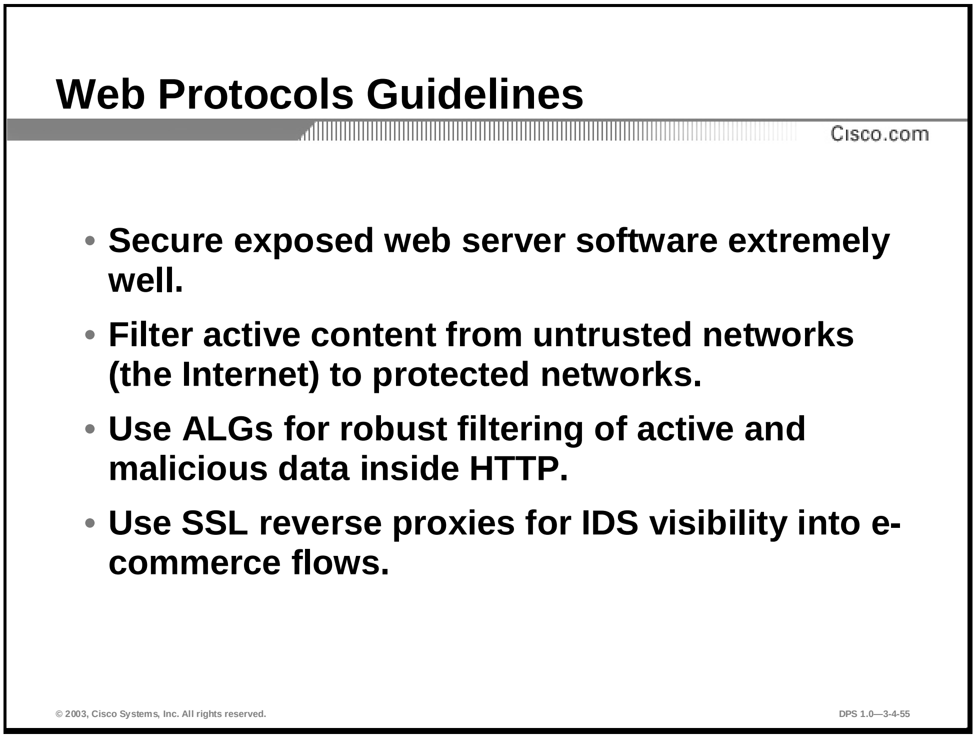
The HTTP over SSL (HTTPS) protocol requires encryption between the client and the server, and can present problems when running over HTTP ALGs. The problem is the encryption of the session, which makes the proxy blind to all application data within the session.

HTTPS is usually handled by application gateways in two different ways

- **Using a simple TCP forwarding tool:** The tool patches the HTTPS session from the client to the server. This negates all benefits of application-layer relaying and only sanitizes the TCP session with the gateway’s TCP stack.

- **Using the “CONNECT” method inside the HTTP proxy:** The client contacts the HTTP proxy and requests a clear channel to the destination server, bypassing all HTTP filtering mechanisms. From a security standpoint, this method is equivalent to the aforementioned TCP forwarding.

A reverse proxy however, can be very useful when handling HTTPS. By loading the private key and certificate of the protected server onto the reverse proxy, it can be configured to pose as the HTTPS endpoint. The client then connects to the reverse proxy using HTTPS, and the HTTPS session terminates there. The reverse proxy can then decrypt HTTP data, and forward it in clear-text to the destination server. This enables application-layer filtering on the proxy, as well as exposing the HTTP request to the network behind the proxy. This enables IDS systems to have full insight into HTTP sessions.



## Web Protocols Guidelines

- **Secure exposed web server software extremely well.**
- **Filter active content from untrusted networks (the Internet) to protected networks.**
- **Use ALGs for robust filtering of active and malicious data inside HTTP.**
- **Use SSL reverse proxies for IDS visibility into e-commerce flows.**

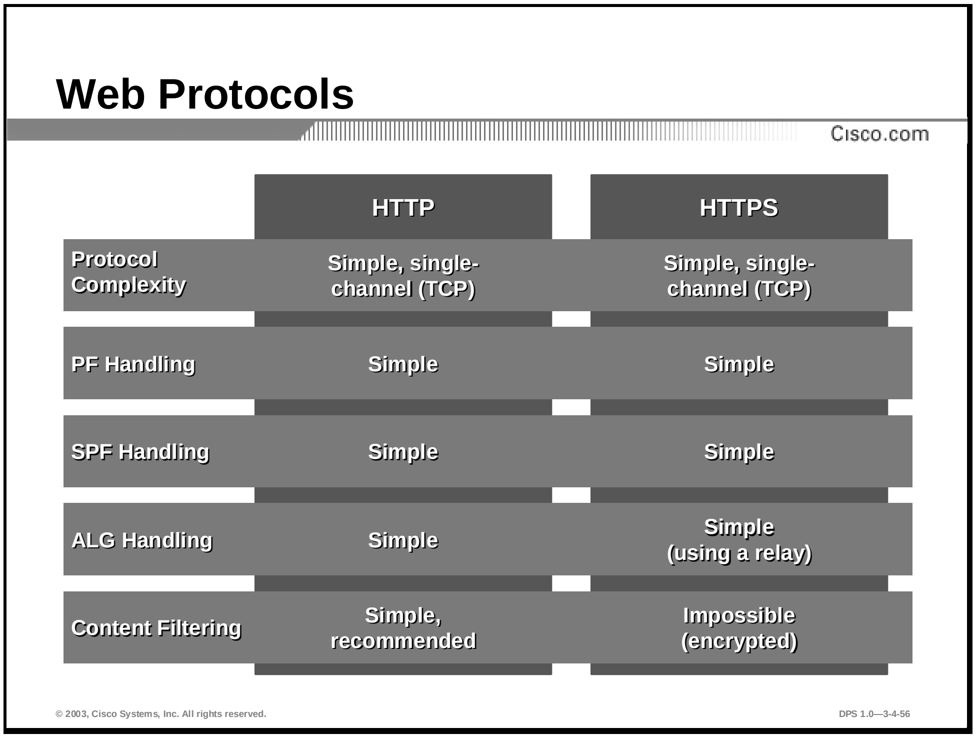
DPS 1.0—3-4-55

## Guidelines

Often used in trusted networks, modern web servers are the most common server software on the Internet. As vendors race to bring functionality to users, web server software has been the source of many security advisories. Securing an exposed web server’s software should be the first priority, when inbound web access is desired, and some vendors provide guidelines for server hardening.

Web client security is most often endangered by malicious code, which can take many forms: from scripting attacks, to infected downloaded files. Active content from untrusted sources such as the Internet should be filtered. Organizations that require the highest levels of security should also opt for complete blocking of any executable-like content from an untrusted network, and not rely solely on content-scanning tools. Use ALGs to provide this filtering functionality.

HTTPS can be passed over firewalls unencrypted, by first decrypting it when it enters the firewall system. Use SSL reverse proxies to terminate SSL connections and translate them into HTTP connections. An ALG or an intrusion detection system can then analyze the HTTP connection.



## Web Protocols

Cisco.com

| | HTTP | HTTPS |
|---|---|---|
| Protocol Complexity | Simple, single-channel (TCP) | Simple, single-channel (TCP) |
| PF Handling | Simple | Simple |
| SPF Handling | Simple | Simple |
| ALG Handling | Simple | Simple (using a relay) |
| Content Filtering | Simple, recommended | Impossible (encrypted) |

DPS 1.0—3-4-56

# Practice

Q1) Which are the three types of HTTP ALGs? (Choose three.)

A) classic (non-transparent) proxies, which the client needs to be aware of

B) transparent proxies

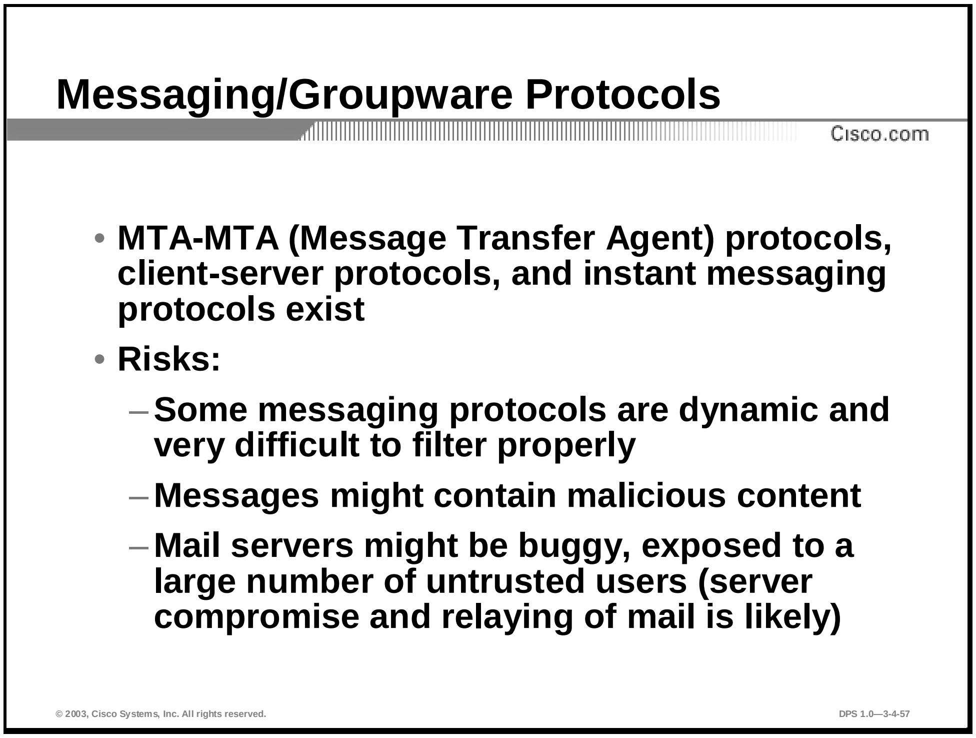
C) reverse proxies

D) passive proxies

E) chained proxies

F) HTTPS proxies

# Messaging Protocols



## Messaging/Groupware Protocols

Cisco.com

- MTA-MTA (Message Transfer Agent) protocols, client-server protocols, and instant messaging protocols exist
- Risks:
  - Some messaging protocols are dynamic and very difficult to filter properly
  - Messages might contain malicious content
  - Mail servers might be buggy, exposed to a large number of untrusted users (server compromise and relaying of mail is likely)

DPS 1.0—3-4-57

## Objective

The section will enable the learner to explain the security properties of messaging protocols to select an appropriate technology to securely pass them over firewalls.

## Introduction

Relaying of electronic mail has always been one of the most important data transfer mechanisms between networks with various security levels. New forms of messaging, such as instant messaging, provide new challenges for passing various messaging protocols over firewalls, and a need for secure handling is needed.

## Messaging Protocols' Refresher

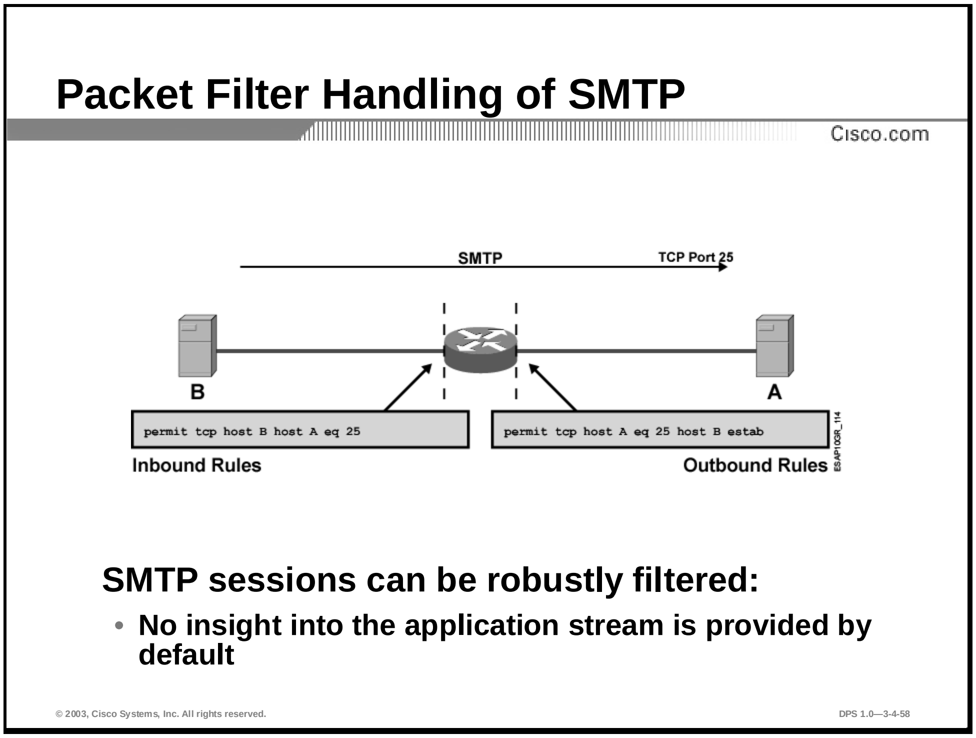
The main risks associated with passing Message Transfer Agent (MTA)-MTA, usually a mail server) messaging protocols between security perimeters over a firewall include:

- The complexity of mail servers makes them viable targets for attack. An attacker is likely to compromise a host through the mail server software.
- The content of email messages can contain confidential information, which can leak to the untrusted network.

- The content of email messages can contain malicious mobile code, such as viruses and “Trojans”, which infect the client system.

- Unauthorized relaying of messages is possible where an outside party uses the organization’s mail server to relay mail to other mail servers, in order to cover its tracks (spamming).

Due to the risks involved, many organizations give significant focus to the relaying of mail between trusted and untrusted networks. Some organizations even install highly specialized systems, called “mailguards”, to granularly control the flow of information between messaging systems.



## Packet Filter Handling of SMTP

```
permit tcp host B host A eq 25
```

```
permit tcp host A eq 25 host B estab
```

**SMTP sessions can be robustly filtered:**

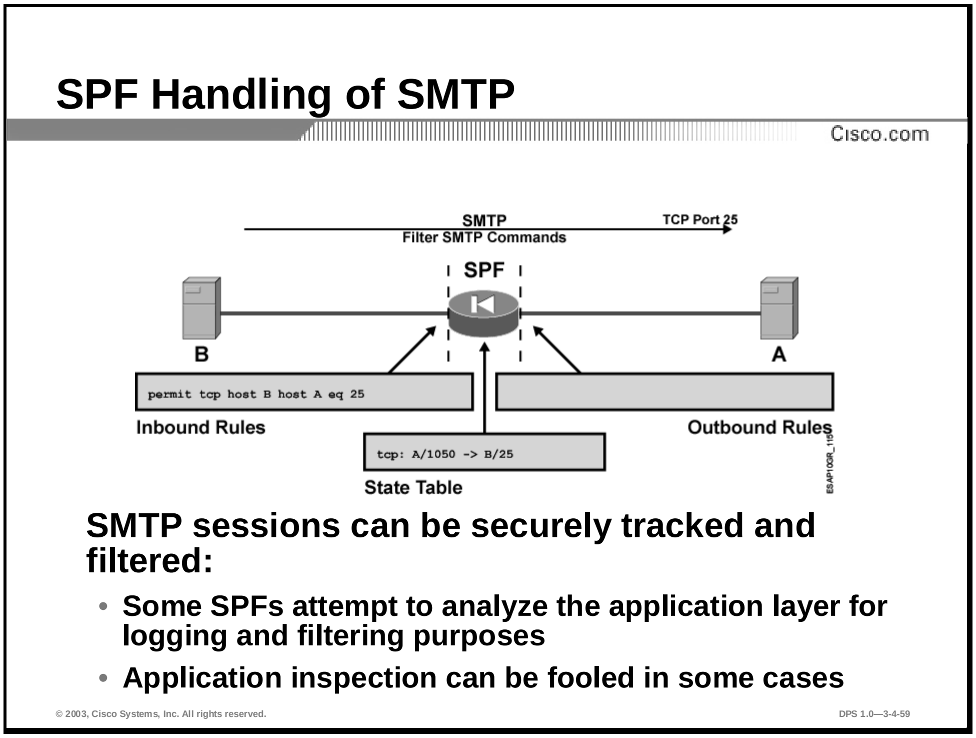
- **No insight into the application stream is provided by default**

DPS 1.0—3-4-58

# Packet Filter Handling of SMTP

Simple Mail Transfer Protocol/Extended SMTP (SMTP/ESMTP) are simple protocols for delivering mail between MTAs in IP networks, usually the Internet. A notable property of the SMTP protocol is that it only supports pushing mail from one MTA to the other—that is, it cannot open a connection to receive mail over it. Also, SMTP/ESMTP protocols do not support any strong authentication of MTAs, which is only provided if SSL is used as a security layer.

A packet filter passes a SMTP session as any other TCP session—that is, by using static filtering rules, the session can be adequately described. A packet filter has no insight into the application stream.



## SPF Handling of SMTP

```
permit tcp host B host A eq 25
```

```
tcp: A/1050 -> B/25
```

**SMTP sessions can be securely tracked and filtered:**

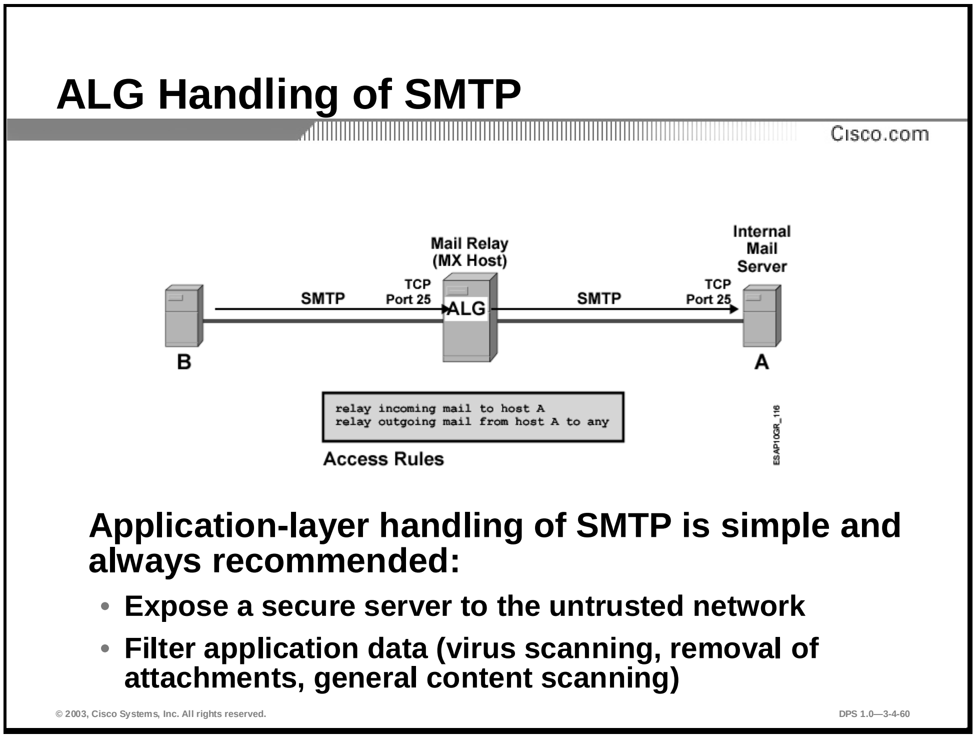
- **Some SPFs attempt to analyze the application layer for logging and filtering purposes**
- **Application inspection can be fooled in some cases**

# SPF Handling of SMTP

SPFs perform standard TCP connection tracking on SMTP sessions. Some SPFs attempt to analyze the application-layer protocol as well, usually filtering non-standard, possibly probably malicious, SMTP commands out of the session.

**Note** Application-layer awareness of SPFs is usually limited to per-packet analysis. SPFs analyze each packet's contents independently; therefore a command spanning multiple packets will not be processed properly. For a robust application-layer access control use pure application gateways.

**Note** The main risks of SMTP lie with buggy servers and malicious content of messages. Focus on securing the application endpoint (exposed mail server, user mail client) to provide the best protection against the most probably threats.



## ALG Handling of SMTP

Cisco.com

```
relay incoming mail to host A
relay outgoing mail from host A to any
```

Access Rules

**Application-layer handling of SMTP is simple and always recommended:**

- **Expose a secure server to the untrusted network**
- **Filter application data (virus scanning, removal of attachments, general content scanning)**

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—3-4-60

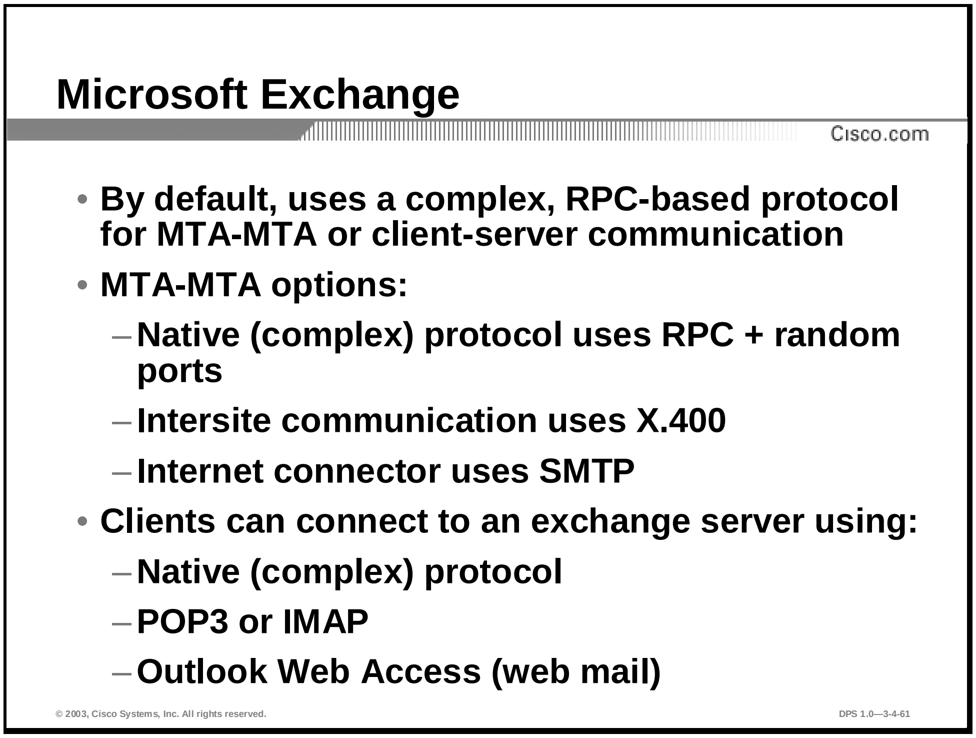
## ALG Handling of SMTP

SMTP is one of the easiest protocols to relay on the application layer as almost any mail server is capable of acting as a SMTP mail router, as well as the mailbox server. A mail-relaying SMTP server acts as an ALG to pass mail between security perimeters. It accepts all messages for the trusted perimeter, and forwards all messages from the trusted perimeter. The DNS concept of mail exchanger (MX) host greatly simplifies redirecting mail addressed to a domain to a specific mail gateway.

The SMTP relay server can be a standalone application, or a simple ALG contained within a firewall package. Depending on the implementation, a SMTP mail relay can impose impressive granularity of message filtering, such as filtering messages on:

- Sender or recipients mail address, domain, or gateway
- Any field in message headers (subjects, dates, message types)
- Any part of the message’s content (text, attachments using pattern matching, and virus scanning)

Mail gateways with rich filtering functionality are sometimes referred to as mailguards. Mailguards are used in environments where messaging is a core application, which needs to be filtered aggressively on network boundaries.



## Microsoft Exchange

- By default, uses a complex, RPC-based protocol for MTA-MTA or client-server communication
- MTA-MTA options:
  - Native (complex) protocol uses RPC + random ports
  - Intersite communication uses X.400
  - Internet connector uses SMTP
- Clients can connect to an exchange server using:
  - Native (complex) protocol
  - POP3 or IMAP
  - Outlook Web Access (web mail)

DPS 1.0—3-4-61

# Microsoft Exchange

Microsoft Exchange uses several protocols to transfer mail between users and gateways, and the availability of those protocols can vary in different versions of Exchange.

Mail transfer between Exchange MTAs (servers) can use two methods

- **Native Exchange protocol:** Uses Microsoft RPC (TCP/135) to negotiate dynamic ports. Using manual Windows registry settings, those ports can be limited to a range of ports, minimizing endpoint exposure. The native protocol is used by default for all Exchange intra-site (usually intra-enterprise) communication.

- **X.400:** Used for all inter-site communications, where Exchange servers in different administrative domains exchange messages. X.400 tunnels all traffic over a single TCP connection (well-known port 102), which is generally used for running OSI applications over TCP/IP.

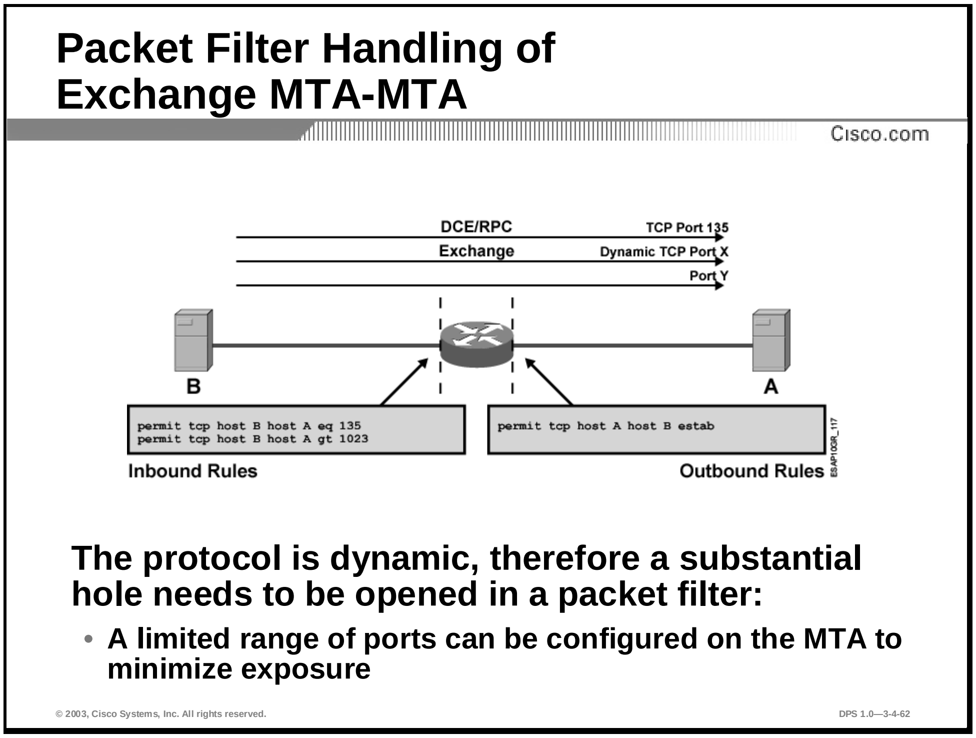
Additionally, SMTP transfer can be used when sending or receiving messages from servers on the Internet.

Clients connect to the Exchange servers using either:

- **Native Exchange protocol:** Similar to servers, this protocol uses Microsoft RPC (TCP/135) to negotiate dynamic ports to connect to the server. Using manual Windows registry settings, those ports can be limited to a range of ports, minimizing endpoint exposure.

- **POP3 or IMAP4 protocols:** Support any standard email client.

- **Outlook Web Access:** Provides mailbox access over a web interface, which can in turn be protected with SSL.



## Packet Filter Handling of Exchange MTA-MTA

```
permit tcp host B host A eq 135
permit tcp host B host A gt 1023
```

```
permit tcp host A host B estab
```

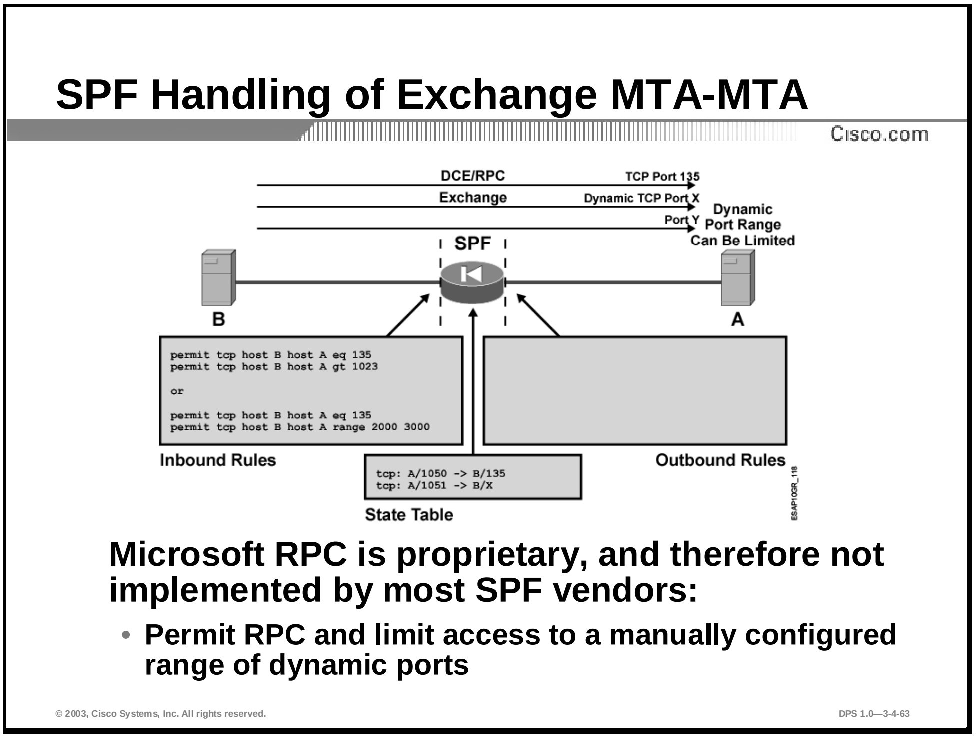
**The protocol is dynamic, therefore a substantial hole needs to be opened in a packet filter:**

- **A limited range of ports can be configured on the MTA to minimize exposure**

DPS 1.0—3-4-62

# Packet Filter Handling of Exchange MTA-MTA

Classic packet filters cannot filter Exchange robustly due to its dynamic nature. It is recommended to manually restrict the port range to a reserved range.



SPF Handling of Exchange MTA-MTA

```
permit tcp host B host A eq 135
permit tcp host B host A gt 1023

or

permit tcp host B host A eq 135
permit tcp host B host A range 2000 3000
```

```
tcp: A/1050 -> B/135
tcp: A/1051 -> B/X
```

**Microsoft RPC is proprietary, and therefore not implemented by most SPF vendors:**

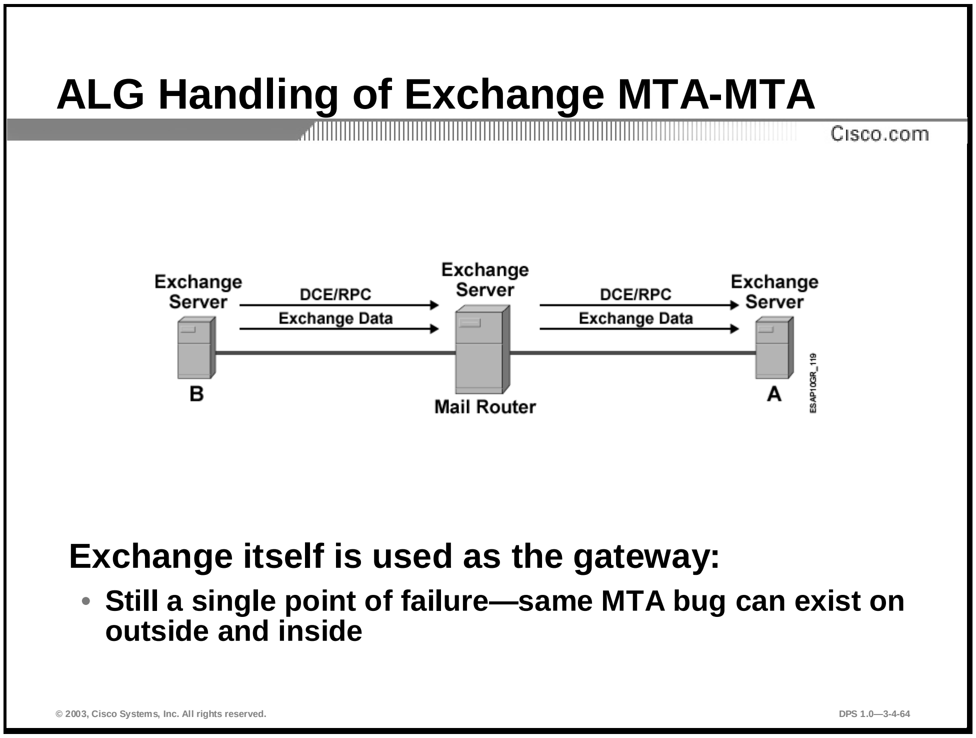
- **Permit RPC and limit access to a manually configured range of dynamic ports**

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—3-4-63

## SPF Handling of Exchange MTA-MTA

Microsoft RPC protocol is proprietary and never disclosed in public. Therefore few vendors have built approximate stateful intelligence to handle the Microsoft RPC protocol, on which native Exchange MTA-MTA sessions are based. A SPF will filter MTA-MTA sessions similarly to a classic packet filter, permitting a limited range of ports between communicating MTAs.



## ALG Handling of Exchange MTA-MTA

Cisco.com

**Exchange itself is used as the gateway:**

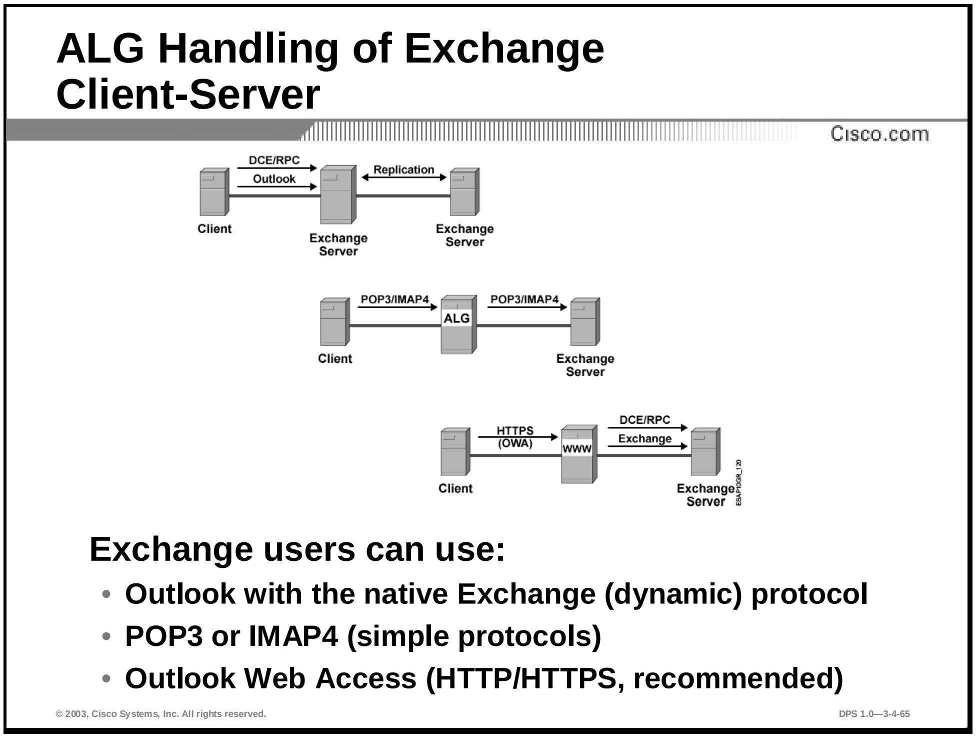
- **Still a single point of failure—same MTA bug can exist on outside and inside**

DPS 1.0—3-4-64

# ALG Handling of Exchange MTA-MTA

ALG handling of Exchange MTA-MTA sessions is possible by using another Exchange server as a relay between two MTAs. In this case, the relaying Exchange server can impose some limited filtering on messages.

| | |
|---|---|
| **Note** | If a software vulnerability exists in Exchange, it will be present on both the protected and relaying Exchange servers, possibly allowing an attacker to compromise both hosts with the same technique. |



## ALG Handling of Exchange Client-Server

Cisco.com

**Exchange users can use:**

- **Outlook with the native Exchange (dynamic) protocol**
- **POP3 or IMAP4 (simple protocols)**
- **Outlook Web Access (HTTP/HTTPS, recommended)**

DPS 1.0—3-4-65

# ALG Handling of Exchange Client-Server

To handle Exchange client-server communication, several options exist

- **Native client-server protocol:** Hard to filter securely using classic packet filters or SPFs. The same recommendations apply as with MTA-MTA communication. Microsoft Outlook uses this protocol by default.

- **POP3 or IMAP4 protocols:** Simple single-session TCP protocols.

- **Outlook Web Access:** A web front-end for Exchange mailboxes. This is the simplest, and the recommended method, for handling Exchange client access from untrusted networks using HTTPS. However, it can only be used if end-users are willing to use a web browser instead of Microsoft outlook.

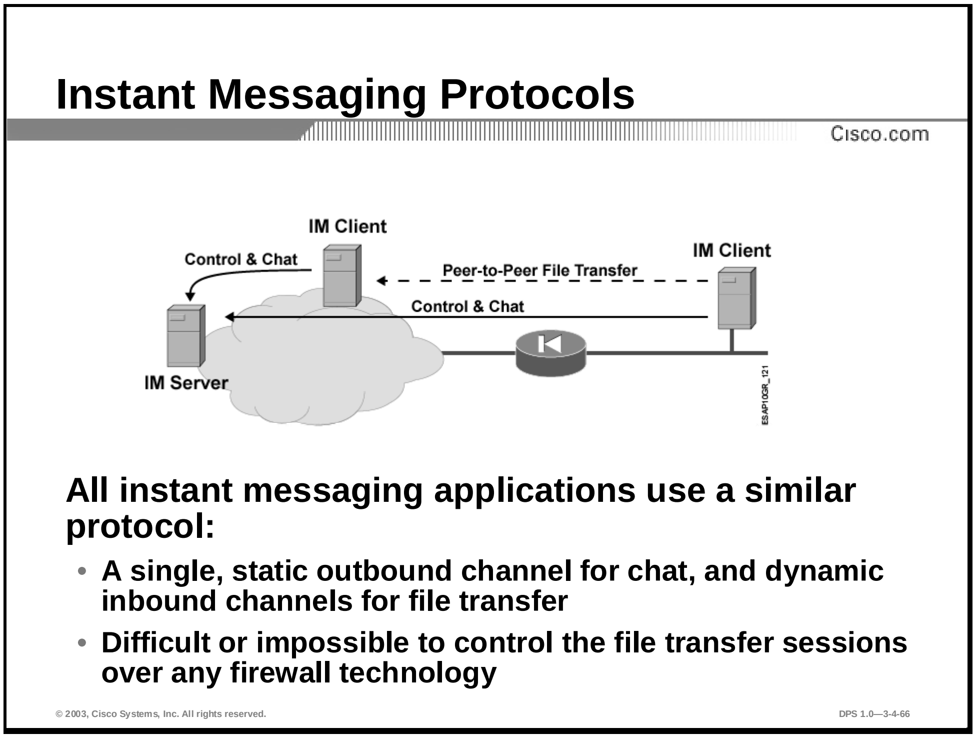
# Guidelines

Guidelines for Microsoft Exchange

- To run Microsoft Exchange MTA-MTA native connections, use fixed port ranges for Microsoft RPC and ensure no other applications live within those port ranges on the affected endpoints.

- X.400 and SMTP can be easily filtered or proxied using an ALG (a dedicated mail relay).

- For Exchange client-server communication, Outlook Web Access is preferred for its simplicity. Internet Message Access Protocol, version 4 (IMAP4) access uses a simple protocol, which is easy to filter. Native access uses dynamic ports, which is the least secure option for relaying over a firewall.



## Instant Messaging Protocols

Cisco.com

**All instant messaging applications use a similar protocol:**

- **A single, static outbound channel for chat, and dynamic inbound channels for file transfer**
- **Difficult or impossible to control the file transfer sessions over any firewall technology**

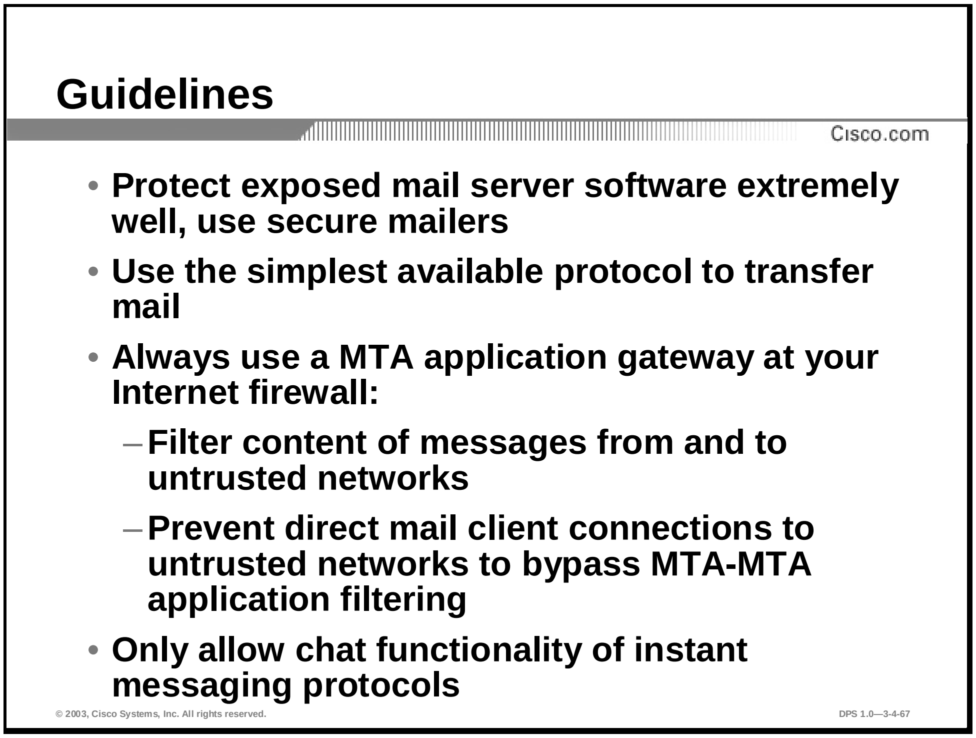
DPS 1.0—3-4-66

# Instant Messaging

All messaging protocols use a similar method for inter-user communication:

- The instant messaging client opens a simple session to a centralized server, which is used for control and chat traffic
- Additional sessions are opened on demand directly between users, and negotiated over the initial control session via the server

Any firewall technology can relay the first session between perimeters. The secondary, peer-to-peer sessions present a significant problem as they are dynamic, and a possible source of malicious content. Do not allow the peer-to-peer sessions to trusted perimeters, especially if there is no content control imposed on data exchanged inside those channels.



## Guidelines

- Protect exposed mail server software extremely well, use secure mailers
- Use the simplest available protocol to transfer mail
- Always use a MTA application gateway at your Internet firewall:
  - Filter content of messages from and to untrusted networks
  - Prevent direct mail client connections to untrusted networks to bypass MTA-MTA application filtering
- Only allow chat functionality of instant messaging protocols

DPS 1.0—3-4-67

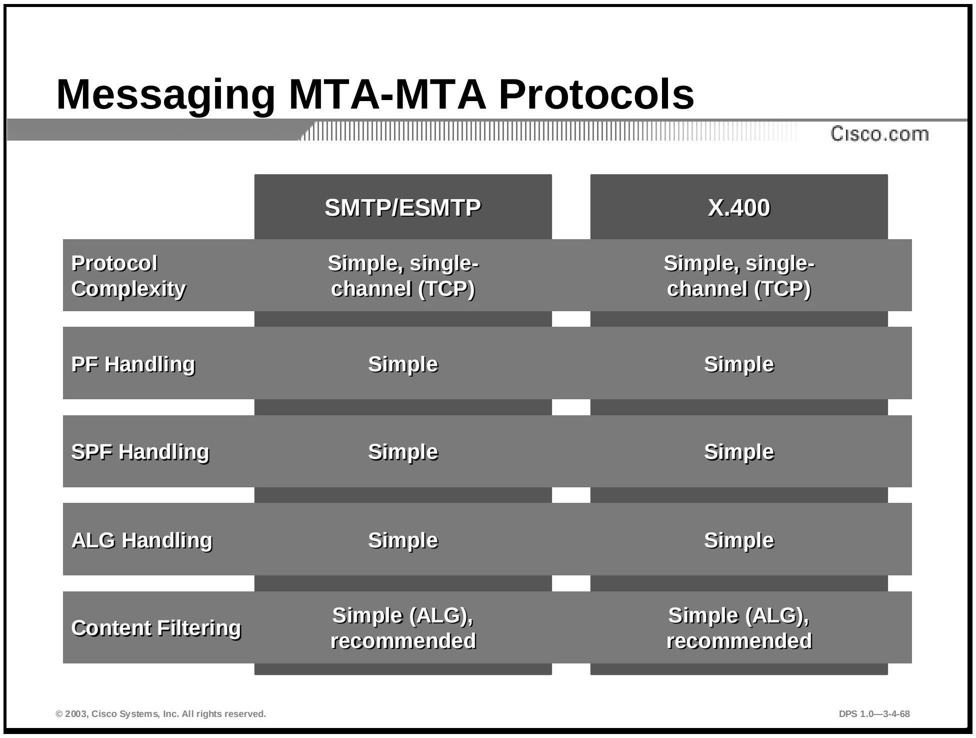
# Guidelines

Mail relaying is one of the most used and best understood firewall applications. As mail servers are usually complex, they have traditionally been a source of many vulnerabilities. Therefore, always relay mail from untrusted networks over ALGs (mail relays), which should be resistant to attacks. This is especially important on the Internet mail gateway. Well-known examples of mailers, designed to resist exploitation, are *postfix* and *qmail*. Multiple commercial mail gateways with extensive filtering capability are available.

Filter email from untrusted networks, as traditionally mail is one of the main channels for malicious code. In high-security environments, use extreme measures such as the stripping of all attachments.

Another issue of relaying mail over firewalls is that of indirect mail channels. If the policy of an organization is to protect email using centralized content control enforced on firewalls and mail servers, users must not be allowed to use clients to connect to servers in untrusted networks and retrieve their mail while bypassing centralized control. An example would be a user who transfers mail to his corporate PC from several outside POP3 accounts.

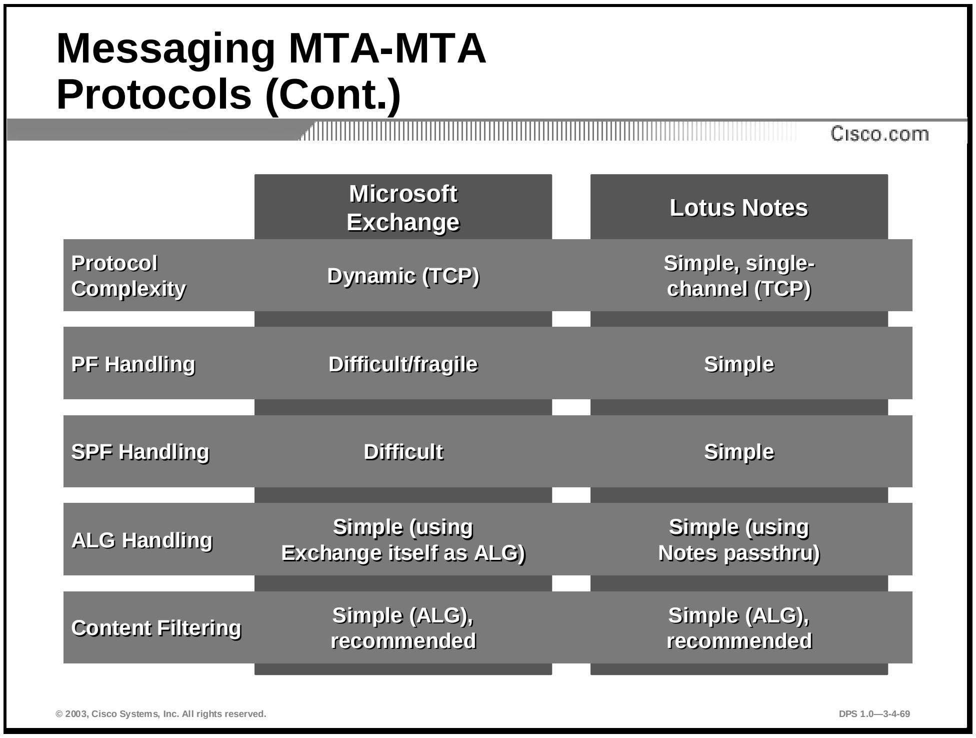
Instant messaging relies partly on a centralized model, which is easy to support over firewalls, as long as active content is not arbitrarily transferred over it. The other part of instant messaging relies on peer-to-peer communication, which is designed to transfer any content. However, firewalls have not yet been able to keep up with peer-to-peer aspects chat protocols. Therefore, do not allow peer-to-peer connections over firewalls, whose policy is to restrict executable content transferred into the protected perimeter.



## Messaging MTA-MTA Protocols

| | SMTP/ESMTP | X.400 |
|---|---|---|
| Protocol Complexity | Simple, single-channel (TCP) | Simple, single-channel (TCP) |
| PF Handling | Simple | Simple |
| SPF Handling | Simple | Simple |
| ALG Handling | Simple | Simple |
| Content Filtering | Simple (ALG), recommended | Simple (ALG), recommended |

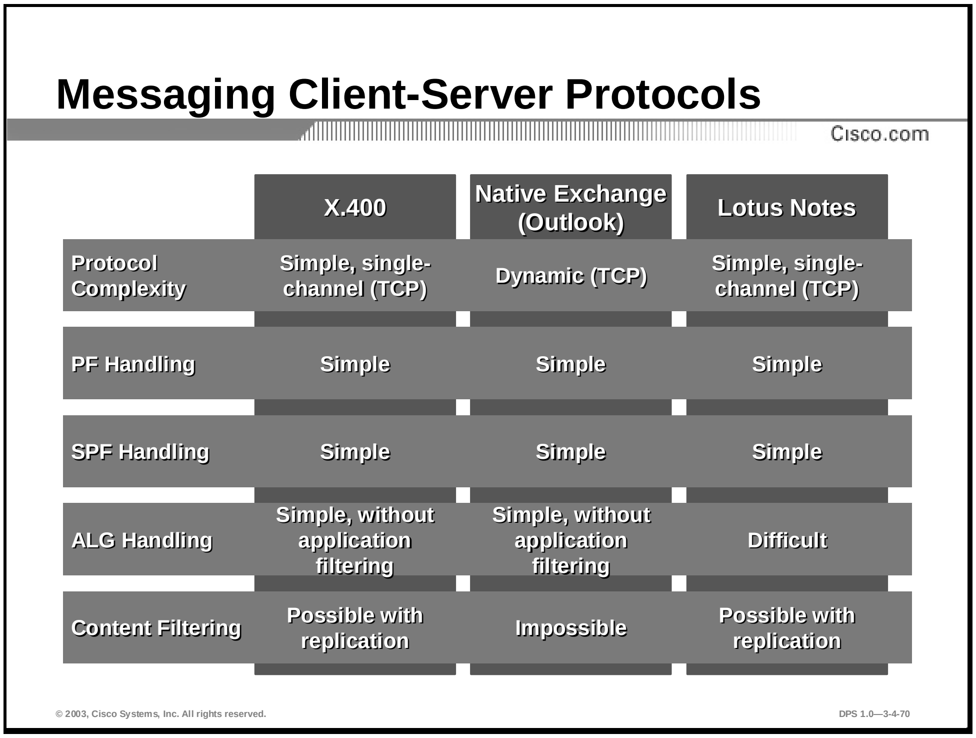
DPS 1.0—3-4-68



## Messaging MTA-MTA Protocols (Cont.)

| | Microsoft Exchange | Lotus Notes |
|---|---|---|
| Protocol Complexity | Dynamic (TCP) | Simple, single-channel (TCP) |
| PF Handling | Difficult/fragile | Simple |
| SPF Handling | Difficult | Simple |
| ALG Handling | Simple (using Exchange itself as ALG) | Simple (using Notes passthru) |
| Content Filtering | Simple (ALG), recommended | Simple (ALG), recommended |

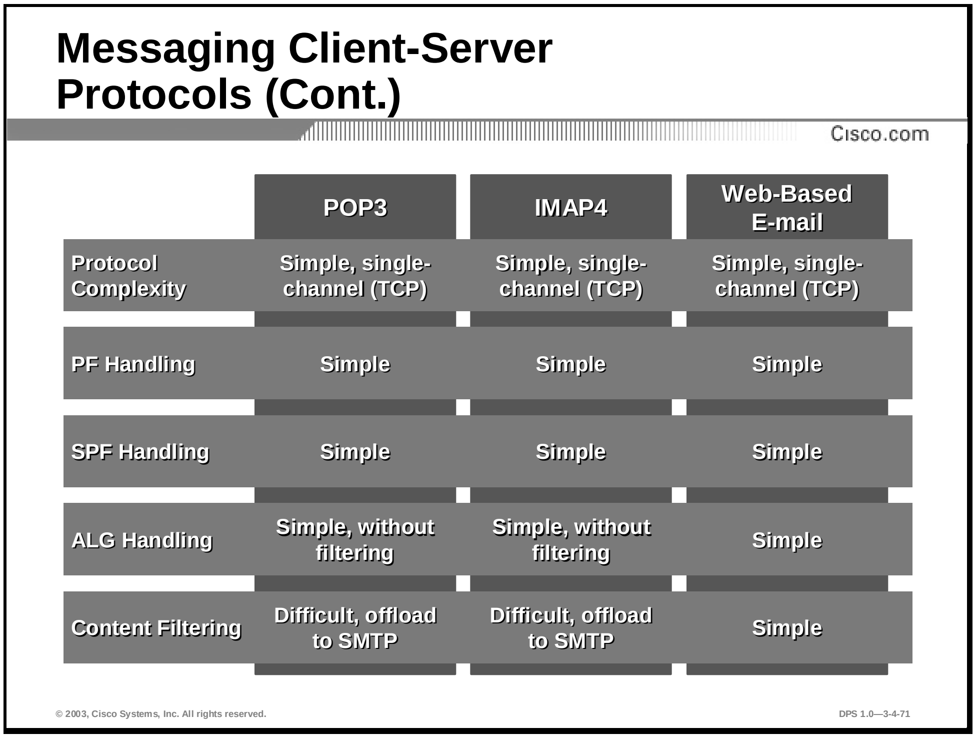
DPS 1.0—3-4-69



## Messaging Client-Server Protocols

| | X.400 | Native Exchange (Outlook) | Lotus Notes |
|---|---|---|---|
| Protocol Complexity | Simple, single-channel (TCP) | Dynamic (TCP) | Simple, single-channel (TCP) |
| PF Handling | Simple | Simple | Simple |
| SPF Handling | Simple | Simple | Simple |
| ALG Handling | Simple, without application filtering | Simple, without application filtering | Difficult |
| Content Filtering | Possible with replication | Impossible | Possible with replication |

DPS 1.0—3-4-70

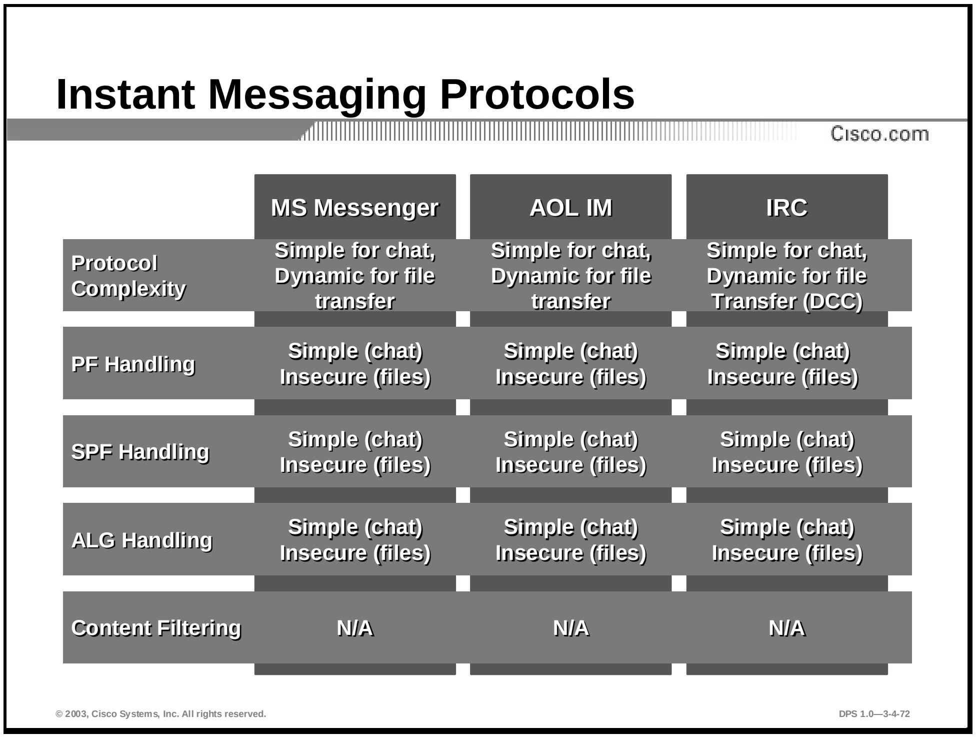


# Messaging Client-Server Protocols (Cont.)

Cisco.com

| | POP3 | IMAP4 | Web-Based E-mail |
|---|---|---|---|
| Protocol Complexity | Simple, single-channel (TCP) | Simple, single-channel (TCP) | Simple, single-channel (TCP) |
| PF Handling | Simple | Simple | Simple |
| SPF Handling | Simple | Simple | Simple |
| ALG Handling | Simple, without filtering | Simple, without filtering | Simple |
| Content Filtering | Difficult, offload to SMTP | Difficult, offload to SMTP | Simple |

DPS 1.0—3-4-71



## Instant Messaging Protocols

Cisco.com

| | MS Messenger | AOL IM | IRC |
|---|---|---|---|
| Protocol Complexity | Simple for chat, Dynamic for file transfer | Simple for chat, Dynamic for file transfer | Simple for chat, Dynamic for file Transfer (DCC) |
| PF Handling | Simple (chat)<br>Insecure (files) | Simple (chat)<br>Insecure (files) | Simple (chat)<br>Insecure (files) |
| SPF Handling | Simple (chat)<br>Insecure (files) | Simple (chat)<br>Insecure (files) | Simple (chat)<br>Insecure (files) |
| ALG Handling | Simple (chat)<br>Insecure (files) | Simple (chat)<br>Insecure (files) | Simple (chat)<br>Insecure (files) |
| Content Filtering | N/A | N/A | N/A |

DPS 1.0—3-4-72

# Practice

Q1) Which three of the following messaging protocols can be securely filtered by a classic packet filter or a generic stateful packet filter? (Choose three.)

A) SMTP

B) POP3

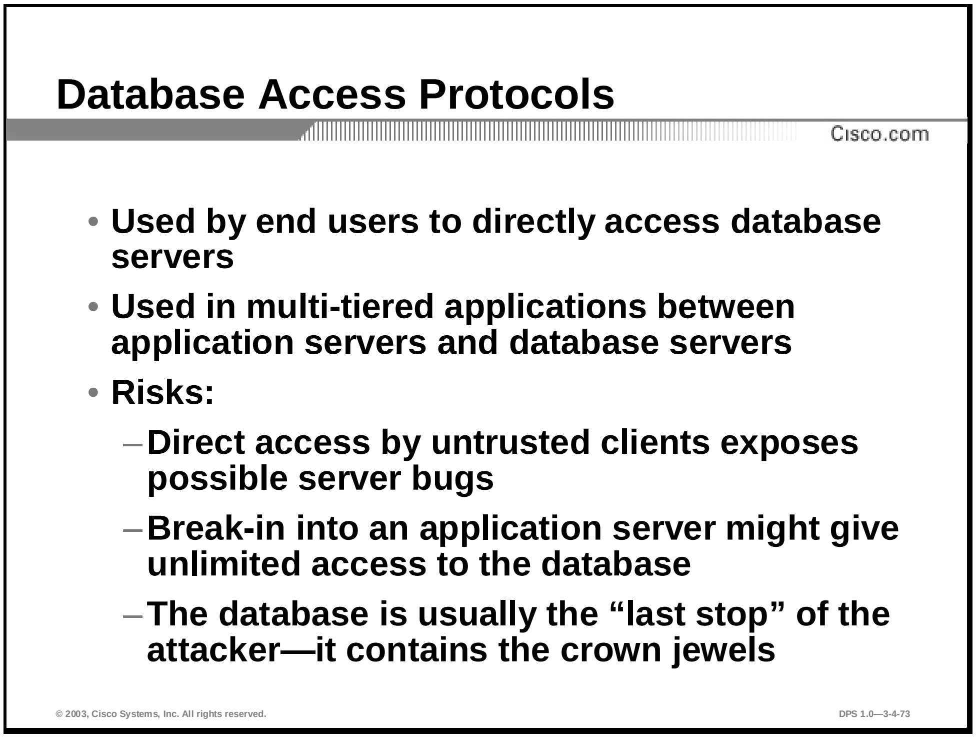
C) Native Exchange MTA-MTA

D) Exchange X.400

E) MS Messenger

F) Yahoo Messenger

# Database Access Protocols



## Database Access Protocols

- Used by end users to directly access database servers
- Used in multi-tiered applications between application servers and database servers
- Risks:
  - Direct access by untrusted clients exposes possible server bugs
  - Break-in into an application server might give unlimited access to the database
  - The database is usually the “last stop” of the attacker—it contains the crown jewels

DPS 1.0—3-4-73

## Objective

The section will enable the learner to explain the security properties of database access protocols to select an appropriate technology to securely pass them over firewalls.

## Introduction

Many mission critical IT system rely on database access protocols to handle the most sensitive data. In the e-commerce world, many systems exposed to untrusted networks will use a database access protocol to access their data back-ends. Firewalls are therefore required to understand and securely handle such protocols in a variety of firewall designs.

## Database Access Protocols’ Refresher

Database access protocols are used in two basic scenarios

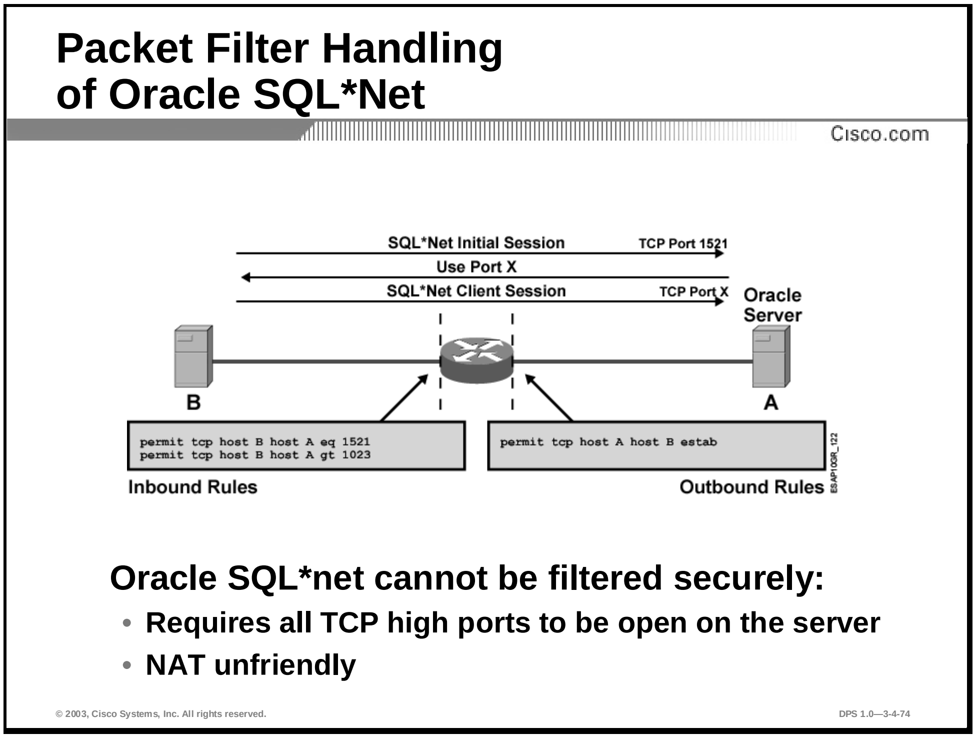
- Used by end users (PC clients) to directly access database servers
- Used in multi-tiered (E-commerce) applications between application servers and database servers

The risks associated with database access are

- Direct access by untrusted clients exposes possible server bugs, possibly allowing direct access to the database

- Break-in into an application server might give unlimited access to the database, if the application server runs with significant database access privileges

- The database is usually the “last stop” of the attacker—it contains the crown jewels of an enterprise, therefore the strongest protection is often required.



## Packet Filter Handling of Oracle SQL*Net

```
permit tcp host B host A eq 1521
permit tcp host B host A gt 1023
```

```
permit tcp host A host B estab
```

**Oracle SQL*net cannot be filtered securely:**

- **Requires all TCP high ports to be open on the server**
- **NAT unfriendly**

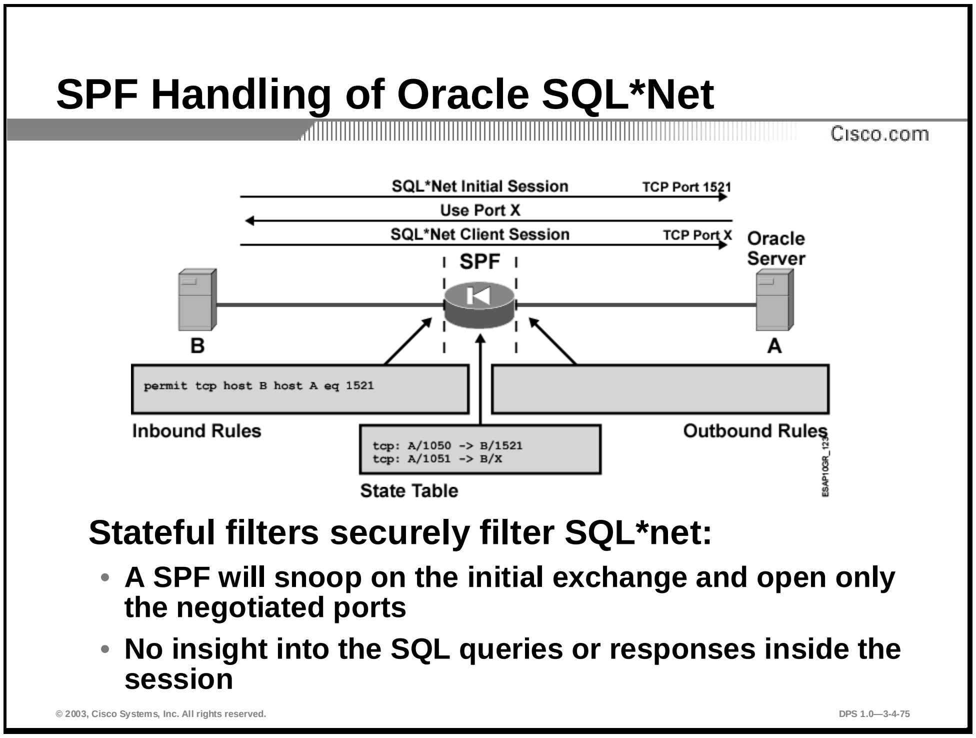
DPS 1.0—3-4-74

# Packet Filter Handling of Oracle SQL*Net

Oracle SQL*Net is a dynamic protocol, where the client initially connects to a well-known listening port on the Oracle server. The server then redirects the client to a new, random server port, and the client reconnects to it and proceeds with the database management system (DBMS) session. This redirect is a message on the application layer.

A packet filter cannot snoop on the client-server negotiation to see the redirect; therefore opening of all high TCP ports to the server is necessary. Packet filters are not suitable for minimizing the exposure of sensitive Oracle servers.

As well its dynamic nature, the SQL*net protocol also embeds the IP address on the application-layer redirect, and is therefore NAT unfriendly.



## SPF Handling of Oracle SQL*Net

```
permit tcp host B host A eq 1521
```

```
tcp: A/1050 -> B/1521
tcp: A/1051 -> B/X
```

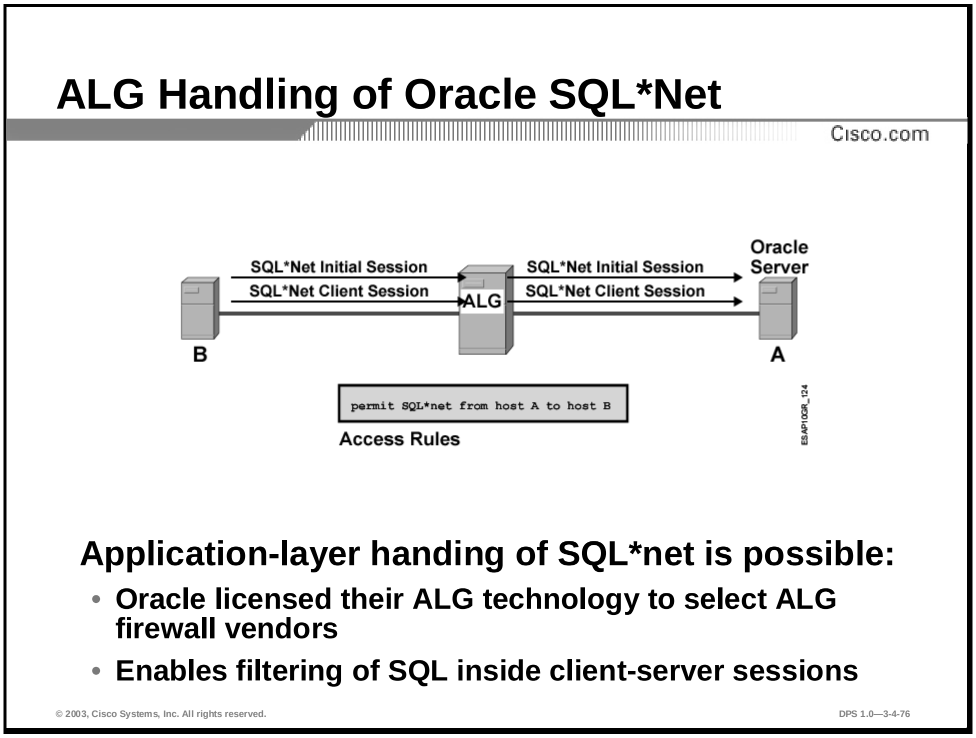
**Stateful filters securely filter SQL*net:**

- A SPF will snoop on the initial exchange and open only the negotiated ports
- No insight into the SQL queries or responses inside the session

DPS 1.0—3-4-75

# SPF Handling of Oracle SQL*Net

Modern SPFs are able to snoop on the initial exchange between the client and the server, and can only open the redirected, negotiated ports between the client and the server. This results in minimal server exposure, but no control over the application content is possible.



## ALG Handling of Oracle SQL*Net

Cisco.com

```
permit SQL*net from host A to host B
```

Access Rules

**Application-layer handing of SQL*net is possible:**

- **Oracle licensed their ALG technology to select ALG firewall vendors**
- **Enables filtering of SQL inside client-server sessions**

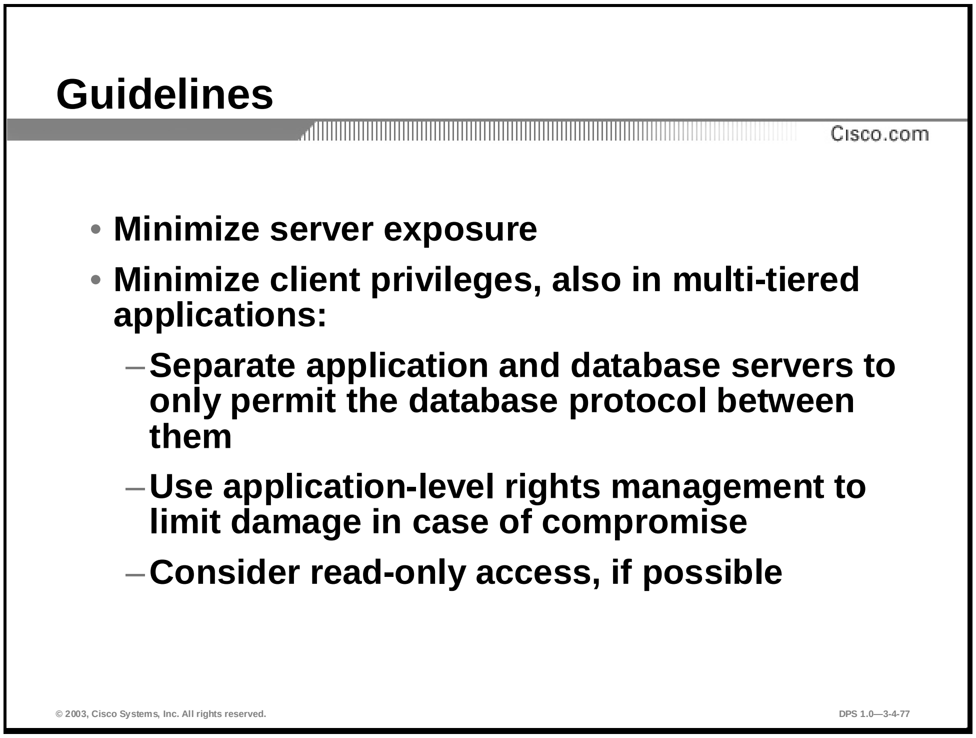
DPS 1.0—3-4-76

## ALG Handling of Oracle SQL*Net

Oracle developed an ALG code for the SQL*net protocol, and licensed it to select ALG vendors. These ALGs can filter inside the SQL*net protocol, enabling the organization to tightly control what SQL transactions are allowed between the application endpoints.

## Other Database Protocols

Other database protocols, such as IBM DB2, Sybase, Microsoft SQL Server, IBM Informix or IBM CICS all use single-TCP sessions to a well known port, therefore they do not present a challenge to any filtering technology. Filtering content inside the database sessions is usually impossible, as specialized database protocol proxies are rare.



## Guidelines

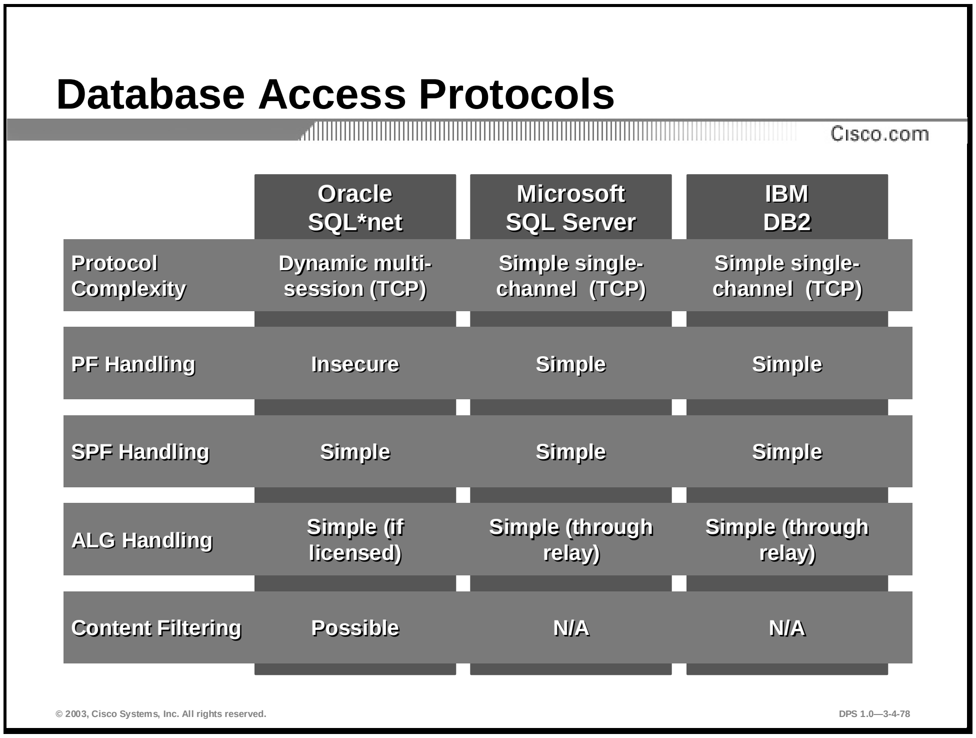
- Minimize server exposure
- Minimize client privileges, also in multi-tiered applications:
  - Separate application and database servers to only permit the database protocol between them
  - Use application-level rights management to limit damage in case of compromise
  - Consider read-only access, if possible

DPS 1.0—3-4-77

# Guidelines

Databases frequently contain extremely sensitive data; therefore strong protection of those servers is often required. Minimizing the exposure of the database servers is the first step. If the database protocol is dynamic, SPF technology is sometimes the only solution.

Minimize client privileges (least privilege concept) in a client-server database application. From an application perspective this is important, because they are often built in multiple tiers. The tiers separate the complex functionality of the application server (which acts as a client to the database) and the database server, which can be isolated from the application server by a firewall. Application rights and database rights can enforce defense-in-depth. However, the designer can sometimes use read-only access if no changes to the database are necessary in an application.

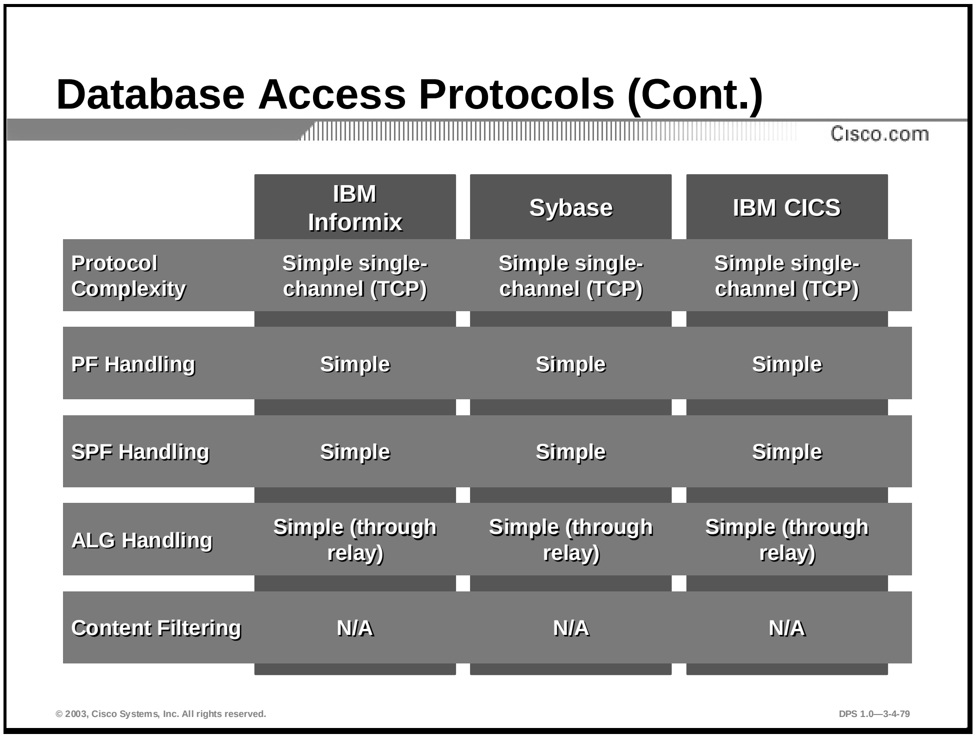


## Database Access Protocols

Cisco.com

| | Oracle SQL*net | Microsoft SQL Server | IBM DB2 |
|---|---|---|---|
| Protocol Complexity | Dynamic multi-session (TCP) | Simple single-channel (TCP) | Simple single-channel (TCP) |
| PF Handling | Insecure | Simple | Simple |
| SPF Handling | Simple | Simple | Simple |
| ALG Handling | Simple (if licensed) | Simple (through relay) | Simple (through relay) |
| Content Filtering | Possible | N/A | N/A |

DPS 1.0—3-4-78



## Database Access Protocols (Cont.)

Cisco.com

| | IBM Informix | Sybase | IBM CICS |
|---|---|---|---|
| Protocol Complexity | Simple single-channel (TCP) | Simple single-channel (TCP) | Simple single-channel (TCP) |
| PF Handling | Simple | Simple | Simple |
| SPF Handling | Simple | Simple | Simple |
| ALG Handling | Simple (through relay) | Simple (through relay) | Simple (through relay) |
| Content Filtering | N/A | N/A | N/A |

DPS 1.0—3-4-79

# Practice

Q1) Why is Oracle SQL*Net protocol peculiar in terms of firewall compatibility?

A) because it uses sessions with dynamic ports

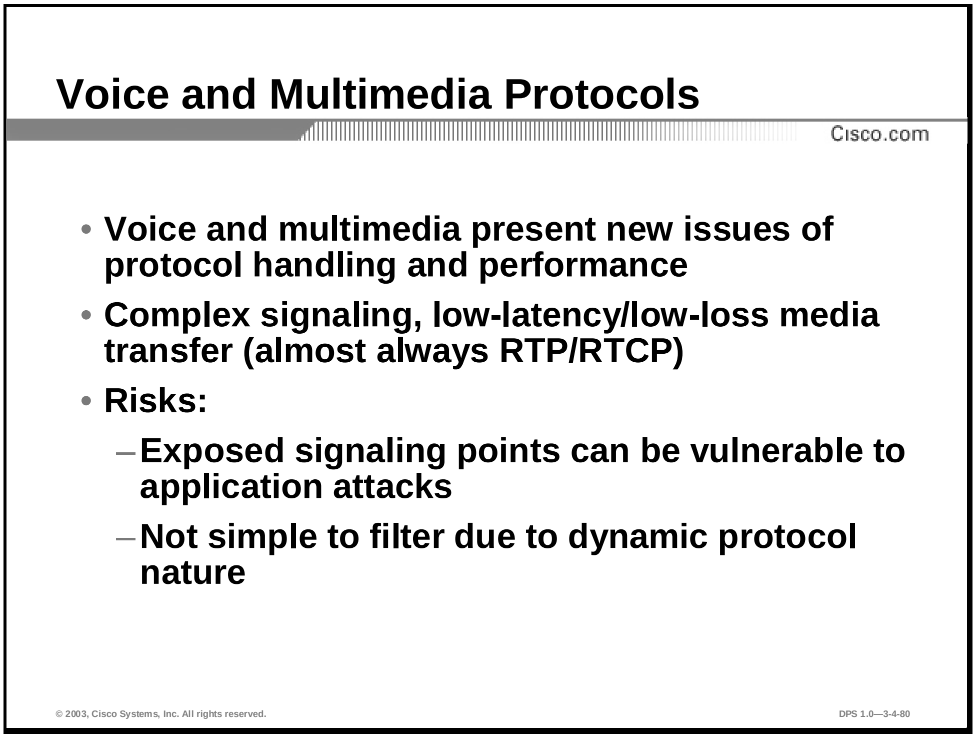
B) because the sessions are always encrypted

C) because it uses a syntax not understood by most firewalls

D) because it is UDP-based

E) because it requires the highest possible performance

# Voice and Multimedia Protocols



## Voice and Multimedia Protocols

Cisco.com

- Voice and multimedia present new issues of protocol handling and performance
- Complex signaling, low-latency/low-loss media transfer (almost always RTP/RTCP)
- Risks:
  - Exposed signaling points can be vulnerable to application attacks
  - Not simple to filter due to dynamic protocol nature

DPS 1.0—3-4-80

## Objective

The section will enable the learner to explain the security properties of multimedia protocols to select an appropriate technology to securely pass them over firewalls.

## Introduction

The consolidation of data and voice/multimedia requires the support of the security infrastructure in modern networks. Firewalls, as the main access control technology, are required to support those often complex protocols in a variety of firewall designs.

## Voice and Multimedia Protocols' Refresher

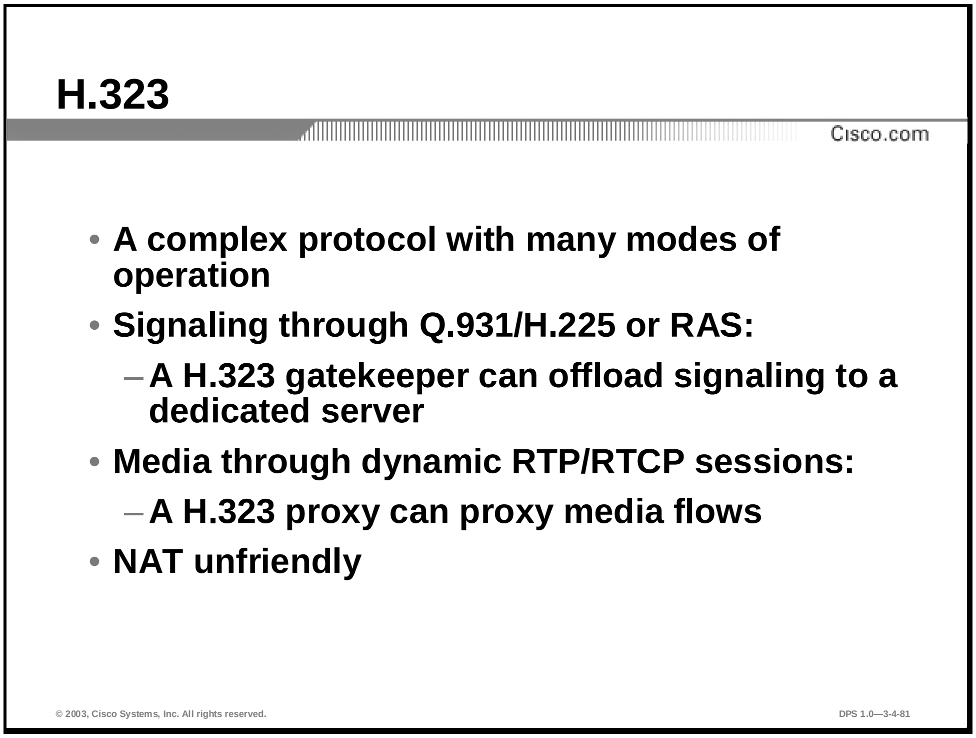
Running voice and multimedia protocols presents new challenges for firewalls, namely:

- The protocols are usually complex (dynamic)
- The applications require low latency handling and consistent throughput (no dropping)

Most of the protocols involve a very complex signaling part, where applications negotiate media channels, and simple dynamic media channels, which usually use the RTP (Real-Time Protocol) and RTCP (Real-Time Control Protocol) protocols.

Risks of voice/multimedia applications involve:

- Attacks against the signaling protocol, which might be exposed to a large number of potential users
- Difficult filtering, due to the dynamic nature of the majority of multimedia protocol



H.323

- A complex protocol with many modes of operation
- Signaling through Q.931/H.225 or RAS:
  - A H.323 gatekeeper can offload signaling to a dedicated server
- Media through dynamic RTP/RTCP sessions:
  - A H.323 proxy can proxy media flows
- NAT unfriendly

DPS 1.0—3-4-81

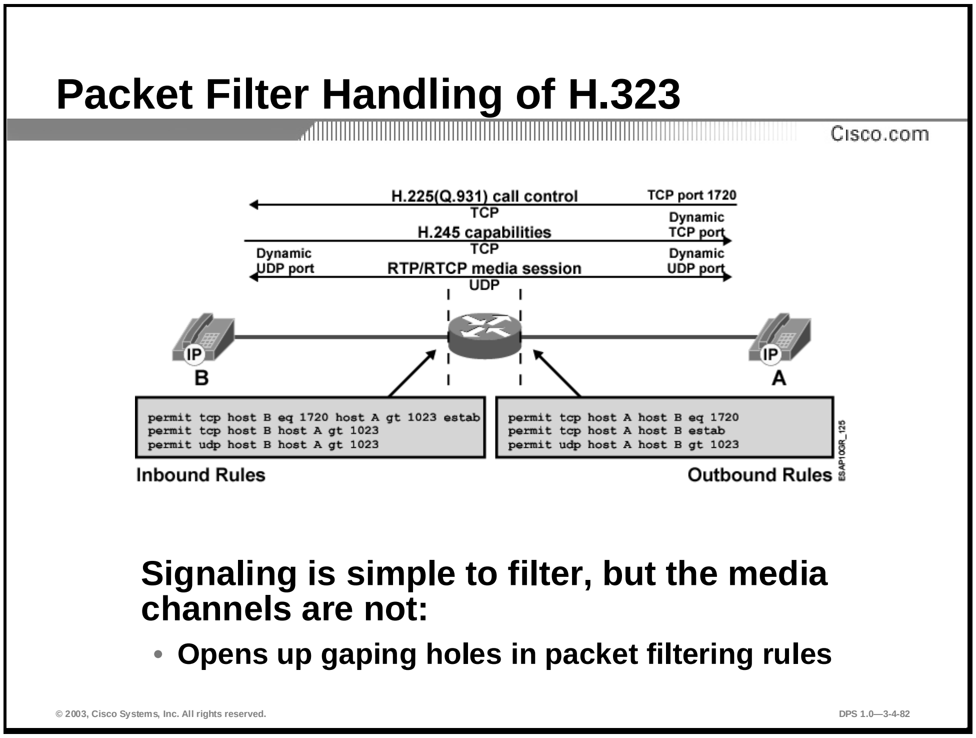
## H.323

H.323 is a complex protocol that has many modes of operation. The signaling channel uses the Q.931/H.225 protocols or, alternatively, the H.323 RAS protocol.

H.323 can run directly between endpoints, for example, IP phones, or a third party can mediate it. The name for the signaling proxy, or third party, in H.323 is gatekeeper. This proxy can provide application-layer relaying of the signaling protocol between two media endpoints, enabling an organization to control and filter signaling messages. The media sessions are established directly between the endpoints, with the gatekeeper only providing call control.

If direct end-to-end media connectivity is not desired, media sessions can also be forwarded through a H.323 proxy. The H.323 proxy accepts the media session on behalf on the endpoints, and forwards it between the two endpoints. This is a viable solution when a firewall filter is not able to process H.323 securely.

H.323 is also NAT unfriendly as it embeds IP addresses on the application layer.



Packet Filter Handling of H.323

Cisco.com

```
permit tcp host B eq 1720 host A gt 1023 estab
permit tcp host B host A gt 1023
permit udp host B host A gt 1023
```

Inbound Rules

```
permit tcp host A host B eq 1720
permit tcp host A host B estab
permit udp host A host B gt 1023
```

Outbound Rules

**Signaling is simple to filter, but the media channels are not:**

- **Opens up gaping holes in packet filtering rules**

© 2003, Cisco Systems, Inc. All rights reserved.

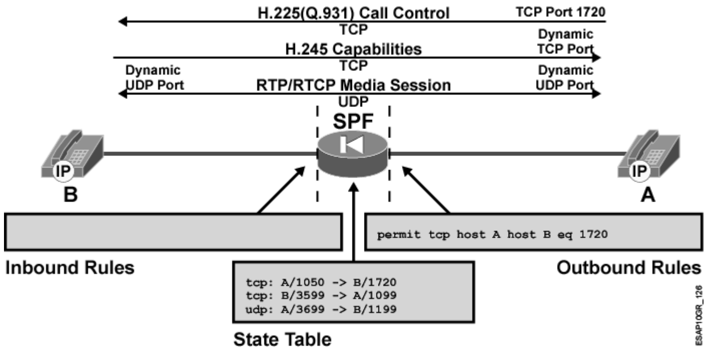
DPS 1.0—3-4-82

## Packet Filter Handling of H.323

A packet filter can securely pass the H.323 signaling protocol (H.225), as it uses a fixed well-known port. The signaling protocol then negotiates dynamic media sessions, which use random ports, and cannot be securely checked by a packet filter. Therefore, packet filtering is not a viable technology to securely support H.323 directly.

## SPF Handling of H.323

Cisco.com



```
permit tcp host A host B eq 1720
```

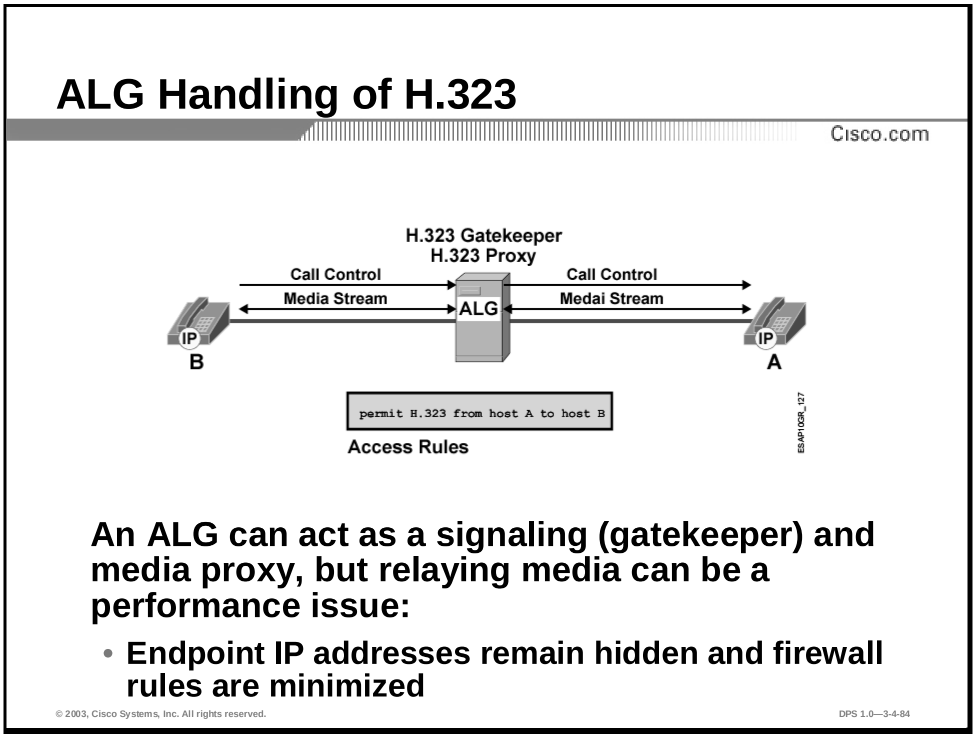
```
tcp: A/1050 -> B/1720
tcp: B/3599 -> A/1099
udp: A/3699 -> B/1199
```

**SPFs generally support snooping of signaling sessions:**

- **The signaling endpoint is exposed to application attacks**

## SPF Handling of H.323

SPFs generally provide support for H.323, and can securely convey both signaling and media sessions. However, this exposes the signaling endpoint to application layer attacks, which are generally not filtered by SPFs.



## ALG Handling of H.323

Cisco.com

H.323 Gatekeeper
H.323 Proxy
Call Control
Media Stream
ALG
Call Control
Medai Stream
IP
B
IP
A
ESAP10GR_127

```
permit H.323 from host A to host B
```

Access Rules

**An ALG can act as a signaling (gatekeeper) and media proxy, but relaying media can be a performance issue:**

- **Endpoint IP addresses remain hidden and firewall rules are minimized**

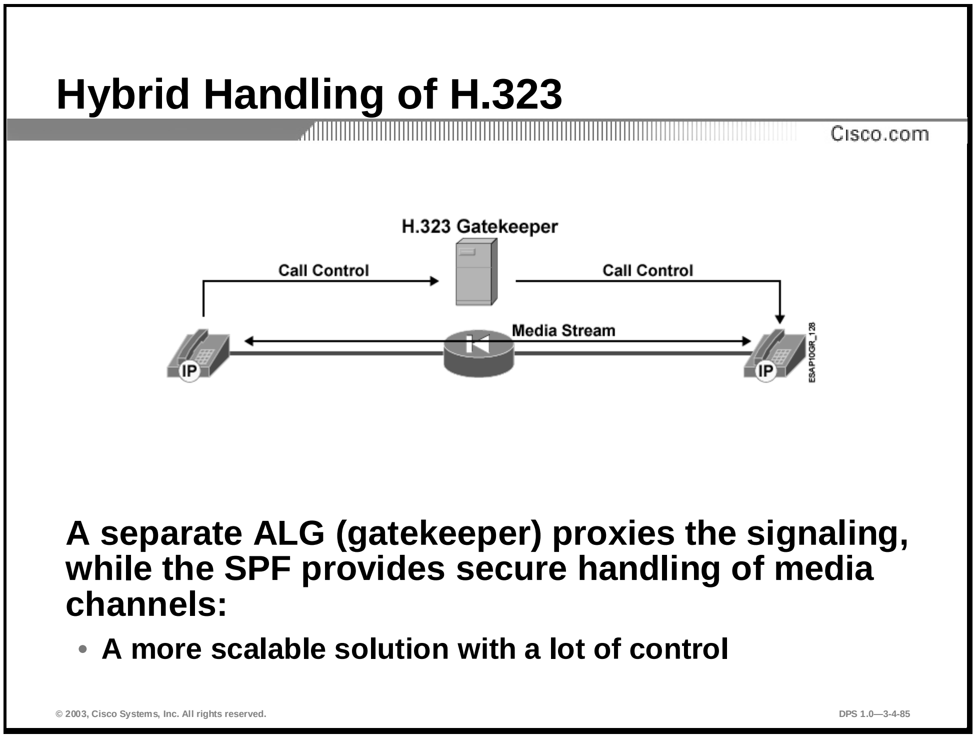
DPS 1.0—3-4-84

## ALG Handling of H.323

An H.323 ALG is a combination of a H.323 gatekeeper and a H.323 proxy. Such an ALG presents the only visible media endpoint, and all other endpoints communicate with it to route calls to their final destinations. The ALG, especially the gatekeeper, may deploy filtering rules, specifying which functionality is allowed within the H.323 network.

This is a solution that minimizes the exposure of a H.323 network and allows minimal connectivity from untrusted networks to the ALG itself.

Depending on the H.323 proxy implementation, latency and throughput of media sessions may become an issue.



## Hybrid Handling of H.323

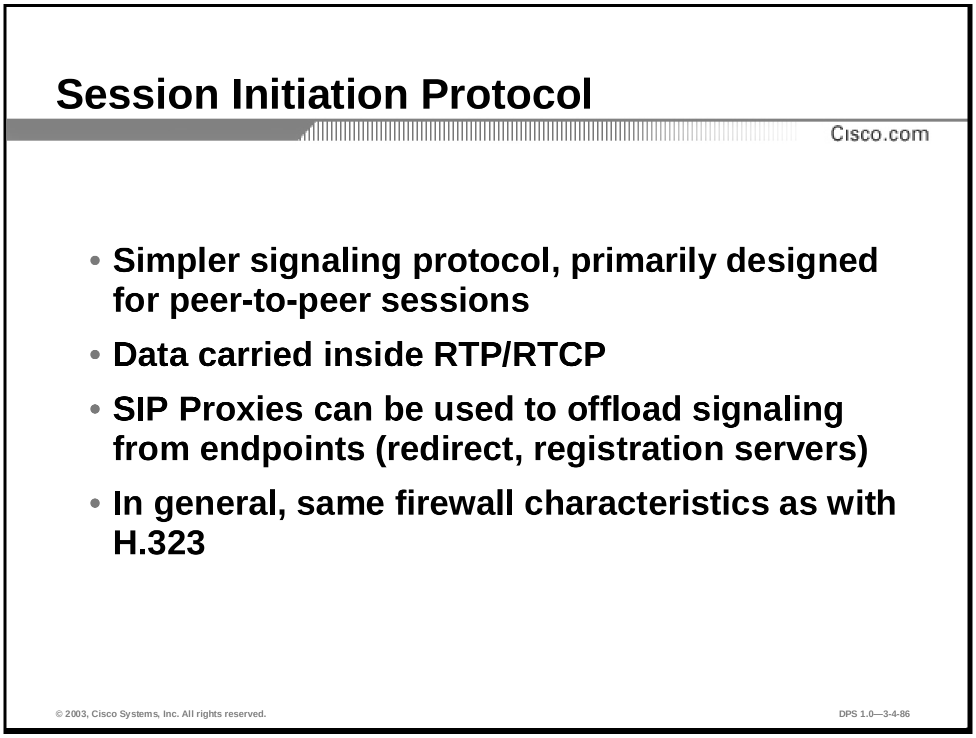
**A separate ALG (gatekeeper) proxies the signaling, while the SPF provides secure handling of media channels:**

- **A more scalable solution with a lot of control**

DPS 1.0—3-4-85

## Hybrid Handling of H.323

If the performance of media streams is an issue, ALG technology can be coupled with SPF technology to create a “best of both worlds” H.323 firewall gateway. Deploy a H.323 gatekeeper (ALG) to filter signaling, and if permitted, the SPF will permit a direct end-to-end media session between the H.323 endpoints.



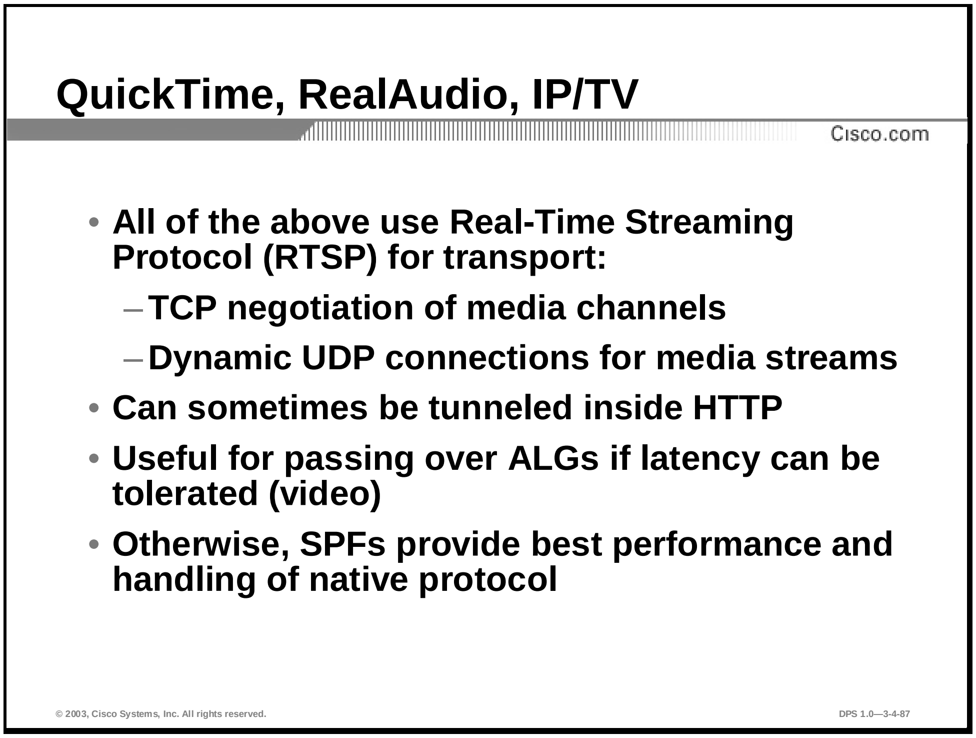
## Session Initiation Protocol

- Simpler signaling protocol, primarily designed for peer-to-peer sessions
- Data carried inside RTP/RTCP
- SIP Proxies can be used to offload signaling from endpoints (redirect, registration servers)
- In general, same firewall characteristics as with H.323

DPS 1.0—3-4-86

## Session Initiation Protocol

Session Initiation Protocol (SIP) is a simpler protocol, designed mainly for pure peer-to-peer connectivity. It is also based on TCP signaling and RTP (UDP) media sessions. SIP implementations can also provide a SIP proxy. Similar to H.323 gatekeeper, this is an ALG used to offload signaling from endpoints, optionally forwarding signaling messages to SIP redirect or registration servers for filtering. When running over the firewall, it generally has the same characteristics as H.323. It can run directly between endpoints using a stateful packet filter with SIP support, or run signaling over an ALG (SIP Proxy), and passing media directly between endpoints over a SPF.



## QuickTime, RealAudio, IP/TV

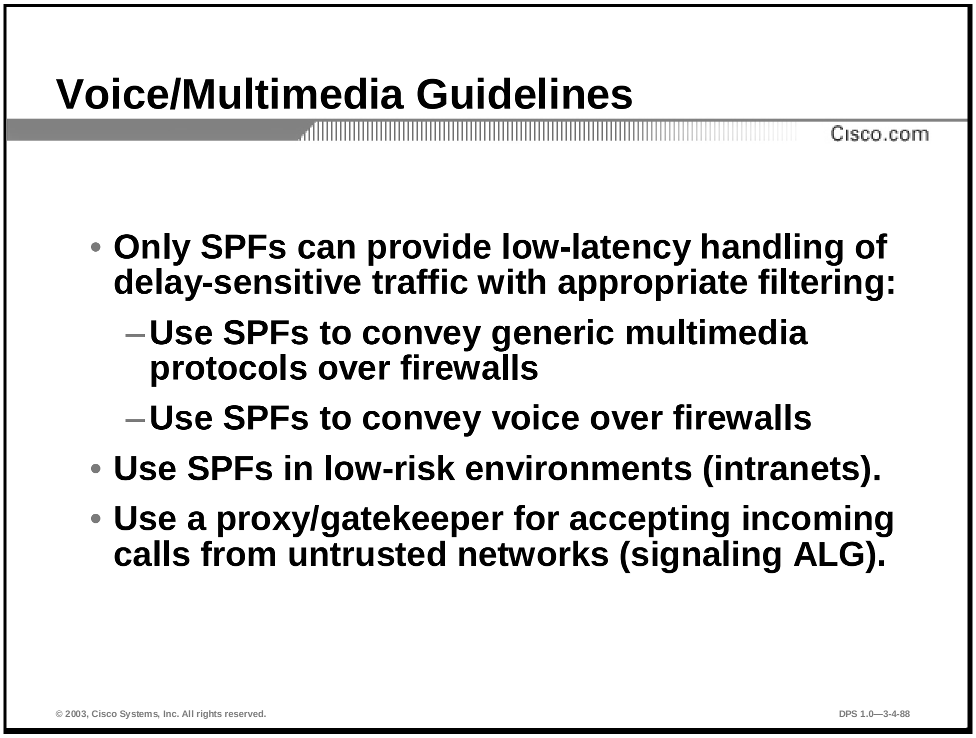
- All of the above use Real-Time Streaming Protocol (RTSP) for transport:
  - TCP negotiation of media channels
  - Dynamic UDP connections for media streams
- Can sometimes be tunneled inside HTTP
- Useful for passing over ALGs if latency can be tolerated (video)
- Otherwise, SPFs provide best performance and handling of native protocol

DPS 1.0—3-4-87

# QuickTime, RealAudio, IP/TV

Video streaming protocols, such as QuickTime, RealAudio, or IP/TV work in a similar way to voice protocols. They all rely on the Real-Time Streaming Protocol (RTSP) for transport of media steams. They initially establish a TCP signaling session between the client and the server, and then negotiate UDP media streams (RTSP) over that TCP session. Therefore, do not use classic packet filtering to securely filter video streaming in this native protocol form.

Some of the video streaming protocols can tunnel themselves inside HTTP. This tunneling has the unfortunate consequence of being able to pass firewalls, which permit HTTP without detailed filtering of the application stream. However, this can simplify a firewall, which cannot handle these protocols natively. Even ALGs can proxy videoconferencing inside HTTP, or natively. The reason for this is that streaming videos can usually tolerate latency. In general, if the SPF in question provides application support for the needed protocol, use SPF technology to pass video traffic.



## Voice/Multimedia Guidelines

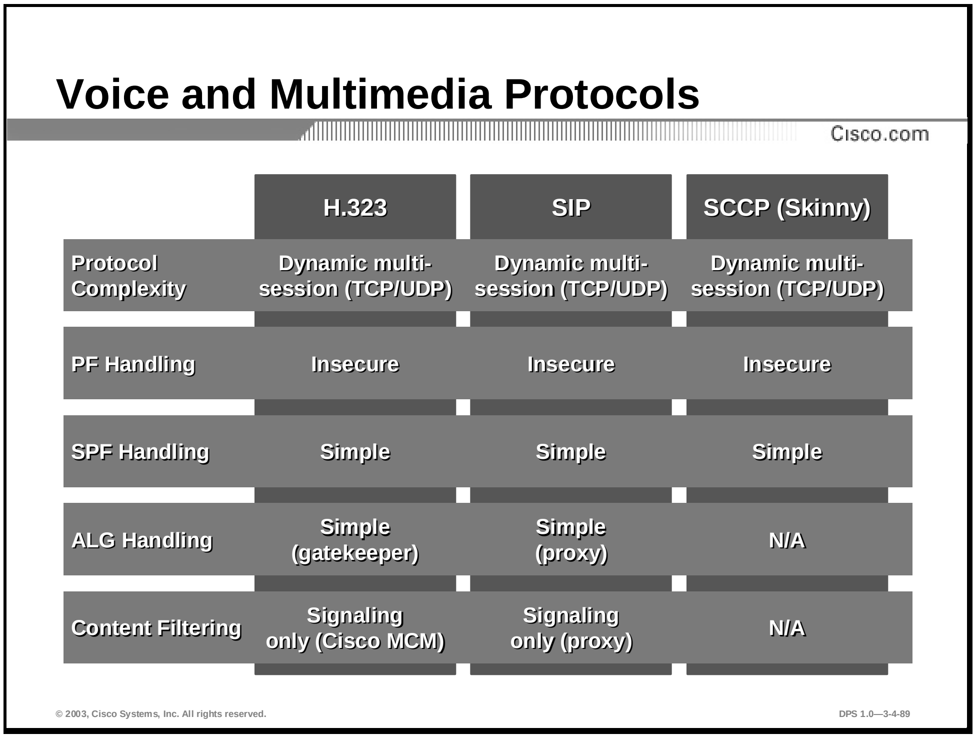
- Only SPFs can provide low-latency handling of delay-sensitive traffic with appropriate filtering:
  - Use SPFs to convey generic multimedia protocols over firewalls
  - Use SPFs to convey voice over firewalls
- Use SPFs in low-risk environments (intranets).
- Use a proxy/gatekeeper for accepting incoming calls from untrusted networks (signaling ALG).

DPS 1.0—3-4-88

# Guidelines

To support real-time multimedia and voice traffic, which are highly dynamic in nature, SPFs are the only technology that can filter it securely standalone. SPFs provide router-like (low) delay and high throughput, and are usually easily aware of multimedia protocol negotiation.

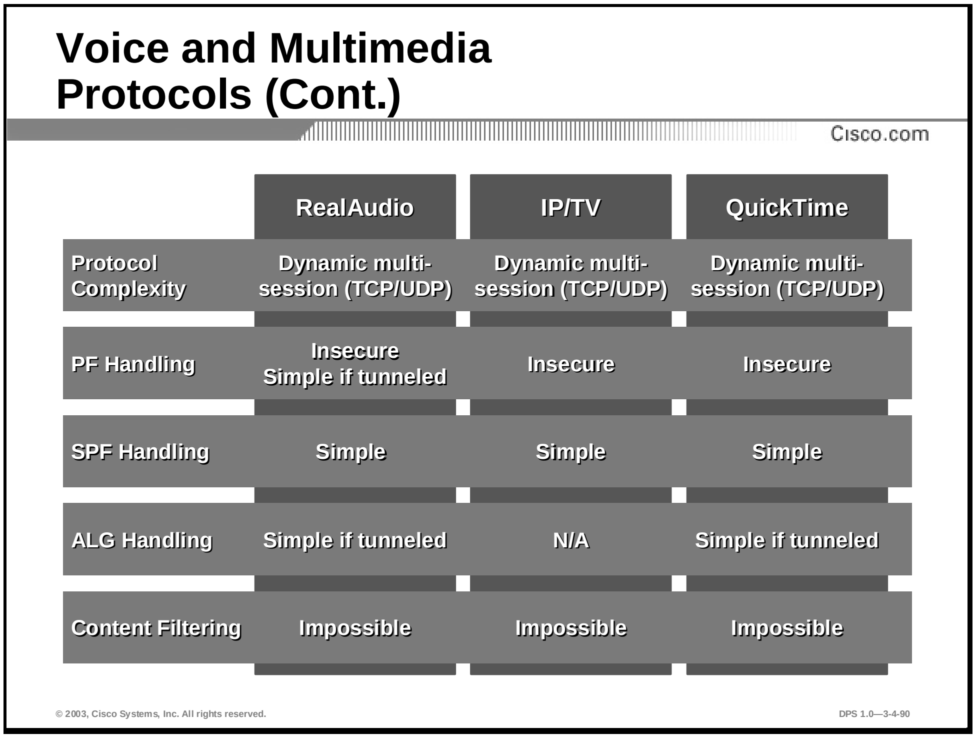
When supporting inbound voice calls from an untrusted network, deploy a signaling ALG (H.323 gatekeeper, SIP proxy) to limit exposure of inside endpoints. The ALG can then also filter incoming call data to provide “voice firewalling”.



# Voice and Multimedia Protocols

| | H.323 | SIP | SCCP (Skinny) |
|---|---|---|---|
| Protocol Complexity | Dynamic multi-session (TCP/UDP) | Dynamic multi-session (TCP/UDP) | Dynamic multi-session (TCP/UDP) |
| PF Handling | Insecure | Insecure | Insecure |
| SPF Handling | Simple | Simple | Simple |
| ALG Handling | Simple (gatekeeper) | Simple (proxy) | N/A |
| Content Filtering | Signaling only (Cisco MCM) | Signaling only (proxy) | N/A |

DPS 1.0—3-4-89



## Voice and Multimedia Protocols (Cont.)

Cisco.com

| | RealAudio | IP/TV | QuickTime |
|---|---|---|---|
| Protocol Complexity | Dynamic multi-session (TCP/UDP) | Dynamic multi-session (TCP/UDP) | Dynamic multi-session (TCP/UDP) |
| PF Handling | Insecure<br>Simple if tunneled | Insecure | Insecure |
| SPF Handling | Simple | Simple | Simple |
| ALG Handling | Simple if tunneled | N/A | Simple if tunneled |
| Content Filtering | Impossible | Impossible | Impossible |

DPS 1.0—3-4-90

# Practice

Q1) What does “hybrid” firewall handling of H.323 refer to?

A) handling of the control stream by an ALG, and the media stream by an SPF

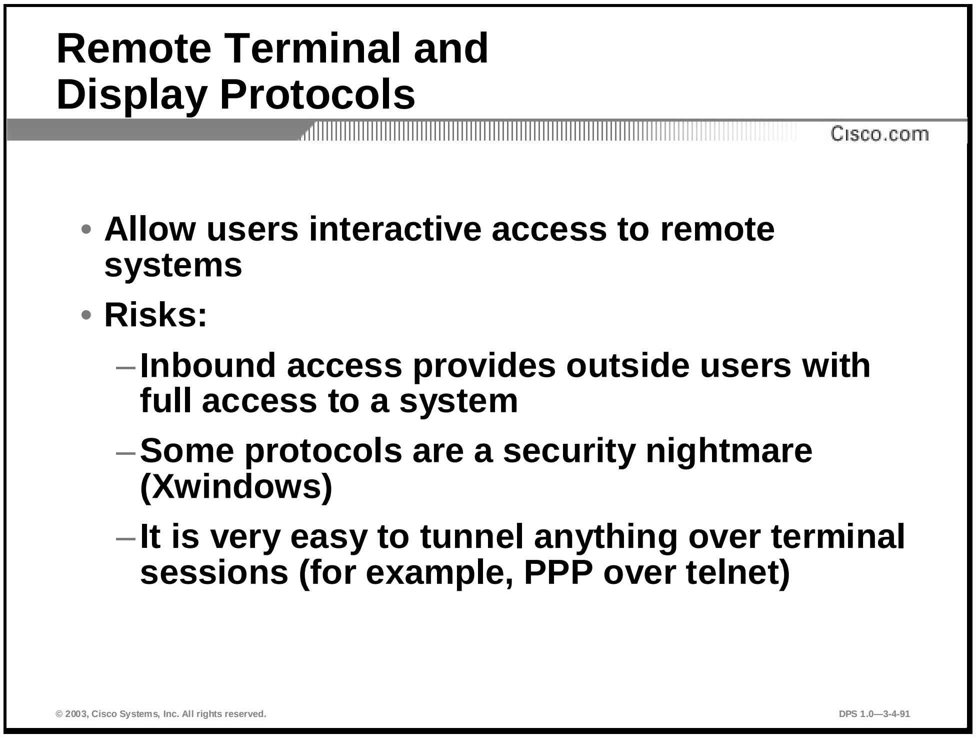
B) handling of the control stream by an SPF, and the media stream by an ALG

C) handling of the control stream and the media stream by an ALG only

D) handling of all data by the H.323 gatekeeper

E) handling of the control stream by an ALG, and the media stream by a H.323 gatekeeper

# Remote Terminal and Display Access Protocols



## Remote Terminal and Display Protocols

- Allow users interactive access to remote systems
- Risks:
  - Inbound access provides outside users with full access to a system
  - Some protocols are a security nightmare (Xwindows)
  - It is very easy to tunnel anything over terminal sessions (for example, PPP over telnet)

© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—3-4-91

## Objective

The section will enable the learner to explain the security properties of remote terminal and display access protocols to select an appropriate technology to securely pass them over firewalls.

## Introduction

Remote terminal and display protocols can be especially vulnerable to failures, as they provide full access to a networked host. Firewalls are often required to pass such sessions to trusted networks, which requires them to handle such protocols securely.

## Remote Terminal and Display Protocols' Refresher

Remote terminal and display protocols provide interactive access to a remote system over a character terminal or a graphical display. Such protocols are often relayed through firewalls, and their use involves several risks

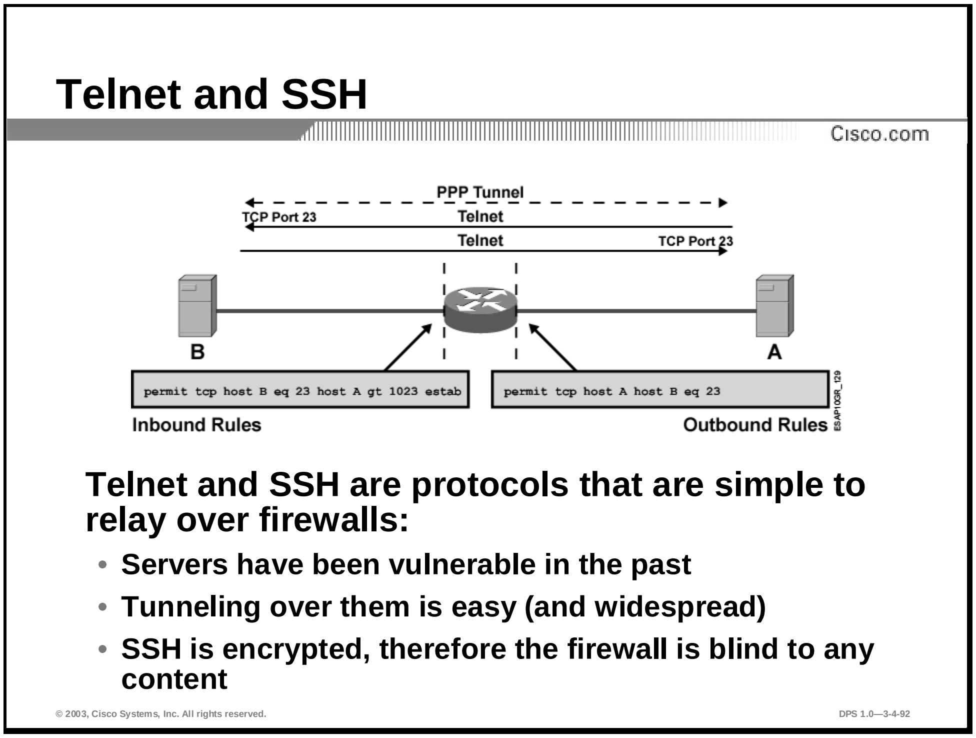
- When permitting a remote terminal/display session, a user receives full user-level access to a system behind the firewall. Therefore, permitting this session is equivalent to permitting all services to that exposed host, as the user fully logs on to the host and could attack it locally without any restrictions of a firewall.

- Some protocols for remote display connectivity are poorly designed and are hard to pass through firewalls, such as Xwindows.
- Character terminal sessions, such as telnet, are ideally suited to tunnel unauthorized protocols.

Fortunately, many terminal sessions can be configured to use strong authentication (for example, via some AAA functionality) or encryption to lower some of the aforementioned risks.

## Example

A well-known tunneling example is running the Point-to-Point Protocol (PPP) over an outgoing telnet session, which a firewall permits. This is functionally is equivalent to installing a leased line with full bidirectional connectivity between the telnet endpoints.



## Telnet and SSH

Cisco.com

```
permit tcp host B eq 23 host A gt 1023 estab
```

```
permit tcp host A host B eq 23
```

**Telnet and SSH are protocols that are simple to relay over firewalls:**

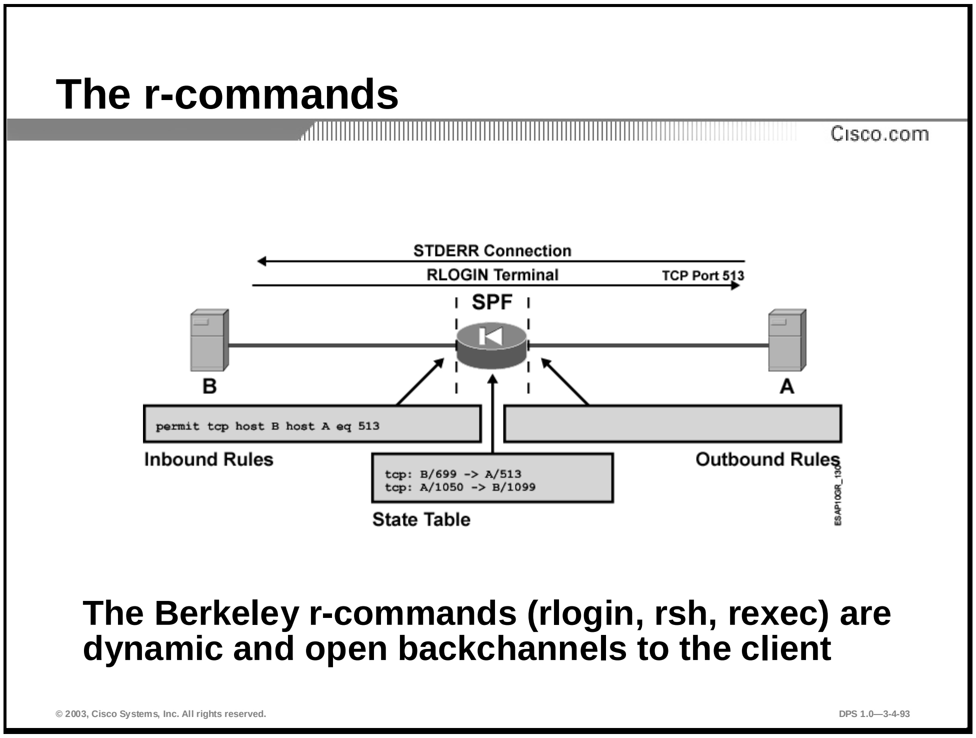
- **Servers have been vulnerable in the past**
- **Tunneling over them is easy (and widespread)**
- **SSH is encrypted, therefore the firewall is blind to any content**

DPS 1.0—3-4-92

# Telnet and SSH

Telnet (TCP well-known port 23) and Secure Shell (SSH) (TCP well-known port 22) are two widely used protocols for remote terminal access. Firewall technology can easily filter both of them, as they are single-channel TCP sessions. Both telnet and SSH servers have traditionally been vulnerable, therefore be cautious when permitting inbound terminal sessions.

Telnet and SSH are both ideally suited for insider tunneling to the inside. SSH has built-in features to forward any single-channel TCP application through the terminal session. Telnet can be used to run PPP over. SSH is also encrypted, therefore the firewall does not have any insight into the content, transferred over the SSH session.



## The r-commands

Cisco.com

```
permit tcp host B host A eq 513
```

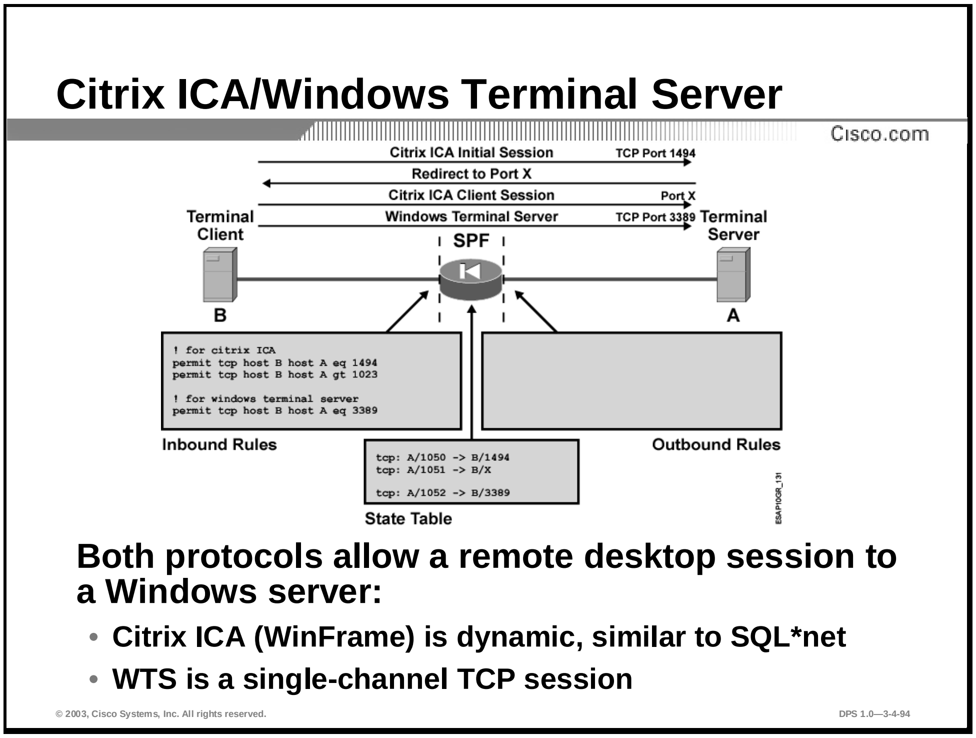
```
tcp: B/699 -> A/513
tcp: A/1050 -> B/1099
```

**The Berkeley r-commands (rlogin, rsh, rexec) are dynamic and open backchannels to the client**

DPS 1.0—3-4-93

# The r-commands

Berkeley UNIX r-commands (rlogin, rsh, rexec) open a session from the client to the server on a well-known port, over which a terminal session or remote execution functionality is established. To report various errors the server also opens a back connection to the client. While this secondary connection is not required for basic operations, classic packet filters cannot securely permit it. Stateful filters and ALGs generally have the intelligence to pass r-commands between perimeters.



## Citrix ICA/Windows Terminal Server

```
! for citrix ICA
permit tcp host B host A eq 1494
permit tcp host B host A gt 1023

! for windows terminal server
permit tcp host B host A eq 3389
```

```
tcp: A/1050 -> B/1494
tcp: A/1051 -> B/X

tcp: A/1052 -> B/3389
```

**Both protocols allow a remote desktop session to a Windows server:**

- **Citrix ICA (WinFrame) is dynamic, similar to SQL*net**
- **WTS is a single-channel TCP session**

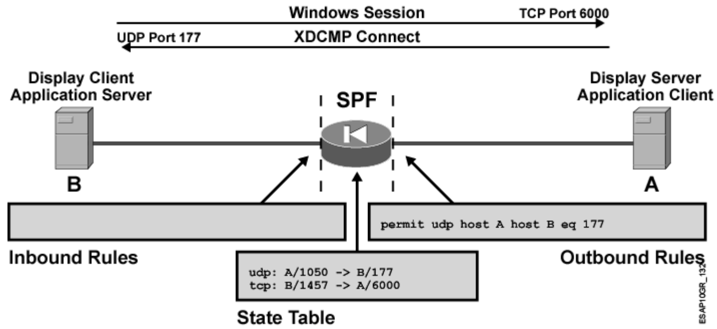
## Citrix ICA and Windows Terminal Server

Citrix ICA and Microsoft Windows Terminal server are both remote display protocols used to access Microsoft Windows servers. Citrix ICA is a dynamic application, similar to Oracle SQL*net as it opens a TCP session to a well-known port, and immediately negotiates a new server port and reconnects to it. This behavior makes it packet filter-unfriendly, while SPFs and ALGs may support it.

Windows Terminal Server uses a single-channel TCP session for each display connection. Therefore, any firewall technology can securely relay it.

## Xwindows

Cisco.com



```
permit udp host A host B eq 177
```

```
udp: A/1050 -> B/177
tcp: B/1457 -> A/6000
```

**Xwindows has a weak security model and offers a lot of access possibilities:**

- **Sharing of display, mouse, keyboard**
- **Unusual behavior—sessions open from the application server to the display server (application client)**

DPS 1.0—3-4-95

# Xwindows

Xwindows is not a particularly secure system and has been shown to have vulnerabilities in the past. Xwindows allows a user to share his/her display, mouse, and keyboard with remote systems, allowing an application running on a different host to display data over the network on the local display.

Xwindows is well known for its “reverse logic” of clients and servers. In Xwindows world, an Xwindows display server is actually a host, which shares its display. The Xwindows display client is the application. It is usually running on an application server, sending display data to the display server, which is the host sharing the display.

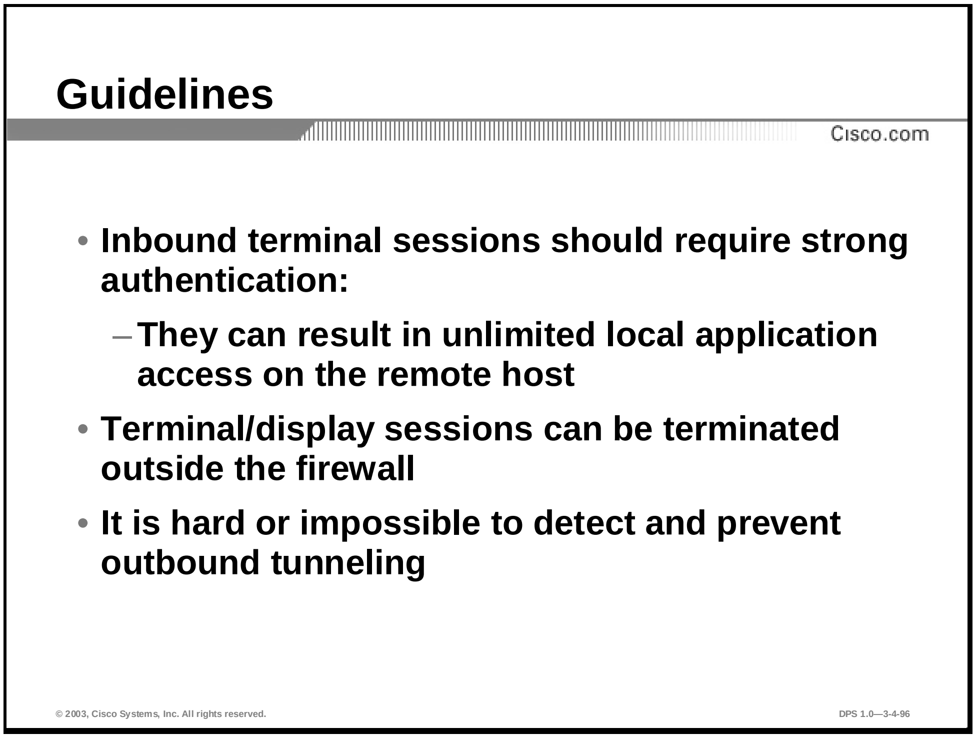
Xwindows opens a single connection from the application server to the display server. If a single display is available on the display server, the connection is usually made on TCP port 6000. Otherwise, other ports above 6000 (for example, 6000 – 6010) are used. Permitting Xwindows statically is therefore not a problem for any firewall.

Historically, Xwindows starts by the user telnetting to a remote server, and running an application that sends display data the users local host. Therefore, multiple connections need to be permitted—one from the user to the application server (telnet, ssh, rlogin), and one from the application server back to the client (Xwindows).

To simplify this procedure for end users, Xwindows sometimes uses an auxiliary protocol called X Display Manager Control Protocol (XDMCP). This protocol sends messages from the display server to the application server, and the application server immediately starts displaying an application on the display server. Some SPFs have built-in support for XDMCP /Xwindows interaction, therefore they only have to permit XDCMP, and the Xwindows channels open automatically.

## Example

The PIX Firewall now provides support for XDMCP to handle an XWindows TCP back connection. Therefore, the PIX Firewall immediately permits a back connection to the client when an XDMCP message goes from the client to the server.



## Guidelines

- Inbound terminal sessions should require strong authentication:
  - They can result in unlimited local application access on the remote host
- Terminal/display sessions can be terminated outside the firewall
- It is hard or impossible to detect and prevent outbound tunneling

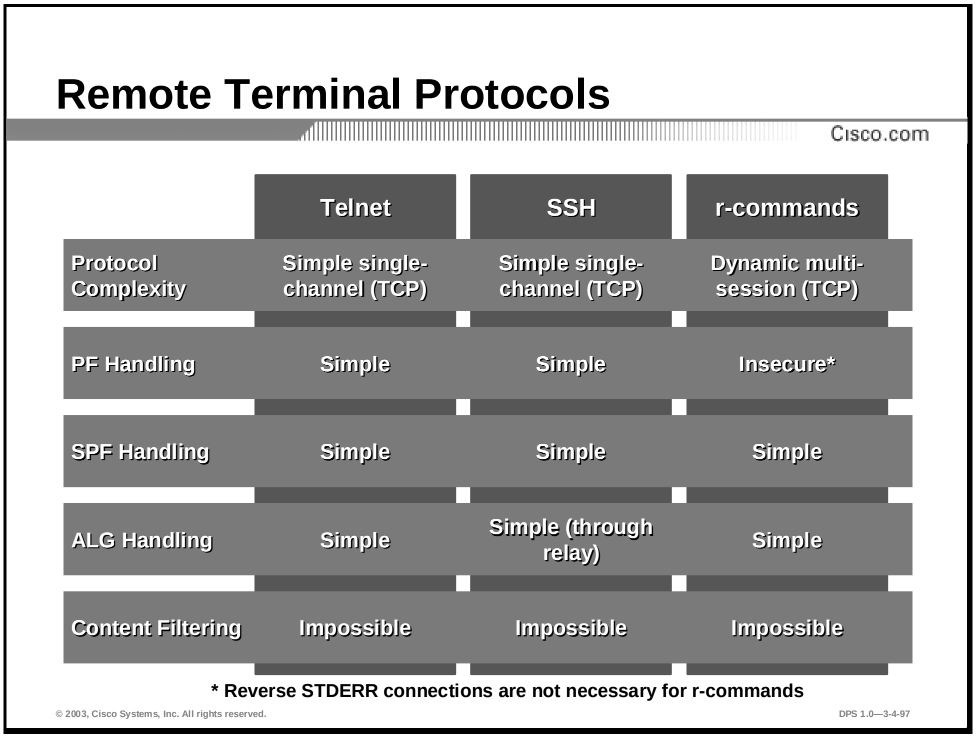
© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—3-4-96

# Guidelines

Running terminal or display sessions outbound, does not usually present a major risk, as sessions do not generally contain malicious data that might compromise the client. Tunneling is an issue, which is generally hard or impossible to detect, and requires the cooperation of an inside.

A terminal or display session can provide unlimited host access; therefore treat inbound sessions very conservatively. Either ensure a very strong authentication on the target server, or use firewall authentication before passing the connection to the firewall server. Alternatively, allow terminal/display access to a host inside the firewall, for example, on an isolated network, by not allowing such sessions to enter the protected network.



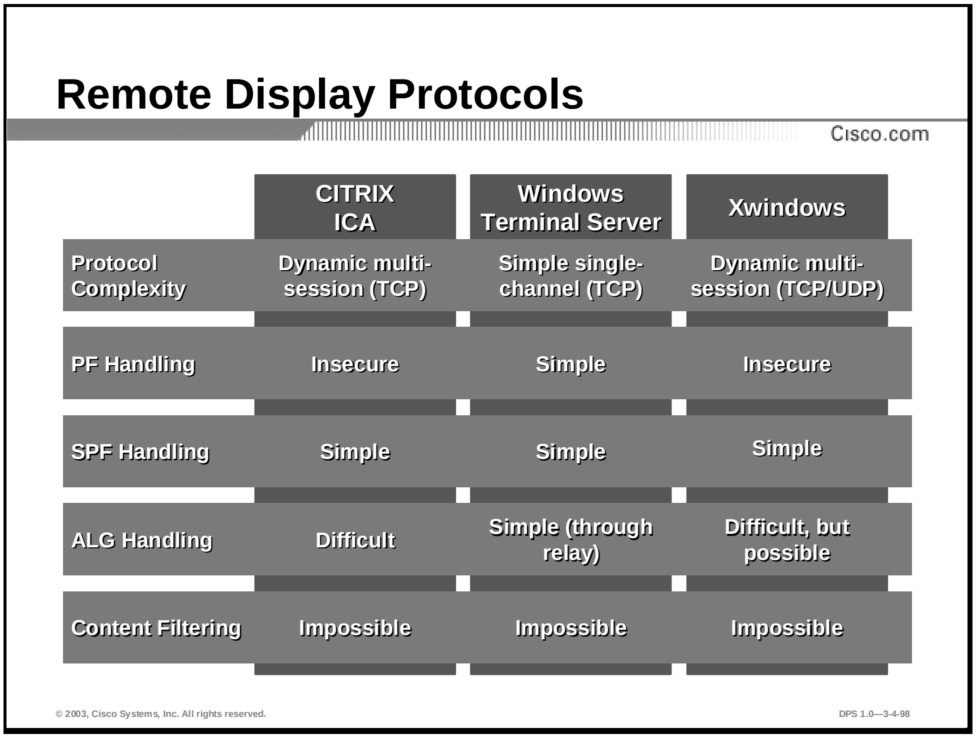
## Remote Terminal Protocols

Cisco.com

| | Telnet | SSH | r-commands |
|---|---|---|---|
| Protocol Complexity | Simple single-channel (TCP) | Simple single-channel (TCP) | Dynamic multi-session (TCP) |
| PF Handling | Simple | Simple | Insecure* |
| SPF Handling | Simple | Simple | Simple |
| ALG Handling | Simple | Simple (through relay) | Simple |
| Content Filtering | Impossible | Impossible | Impossible |

* Reverse STDERR connections are not necessary for r-commands

DPS 1.0—3-4-97



## Remote Display Protocols

| | CITRIX ICA | Windows Terminal Server | Xwindows |
|---|---|---|---|
| Protocol Complexity | Dynamic multi-session (TCP) | Simple single-channel (TCP) | Dynamic multi-session (TCP/UDP) |
| PF Handling | Insecure | Simple | Insecure |
| SPF Handling | Simple | Simple | Simple |
| ALG Handling | Difficult | Simple (through relay) | Difficult, but possible |
| Content Filtering | Impossible | Impossible | Impossible |

DPS 1.0—3-4-98

## Practice

Q1) Which of the following remote terminal/display protocols is static in nature (i.e. uses a single TCP or UDP channel)? (Choose two.)

A) Citrix ICA

B) telnet

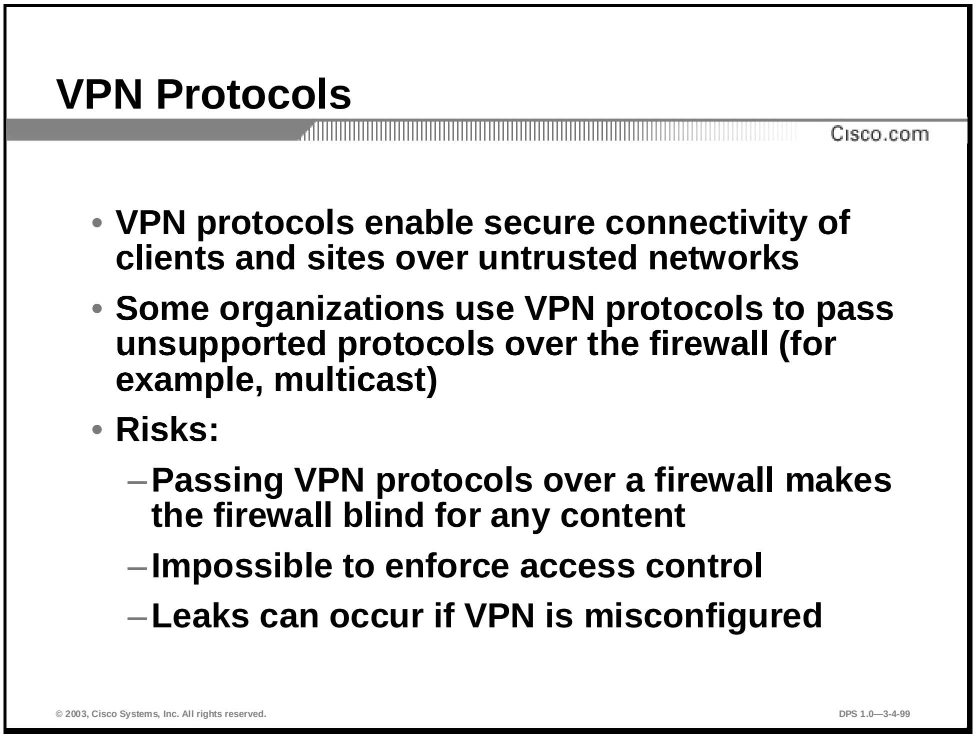
C) SSH

D) Xwindows with XDMCP

E) rlogin

F) rsh

# VPN Protocols



## VPN Protocols

- VPN protocols enable secure connectivity of clients and sites over untrusted networks
- Some organizations use VPN protocols to pass unsupported protocols over the firewall (for example, multicast)
- Risks:
  - Passing VPN protocols over a firewall makes the firewall blind for any content
  - Impossible to enforce access control
  - Leaks can occur if VPN is misconfigured

## Objective

The section will enable the learner to explain the security properties of VPN protocols to select an appropriate technology to securely pass them over firewalls.

## Introduction

VPN protocols are often used in conjunction with firewalls at network boundaries. Often, some integration of VPN technology with the firewall's access control technologies is needed, and modern firewalls must provide a method to securely pass and integrate VPN connections into the firewall system.

## VPN Protocols' Refresher

To provide secure connectivity over untrusted networks or to tunnel a foreign protocol over another network, for example, IPX over the IP Internet, VPN protocols are used.

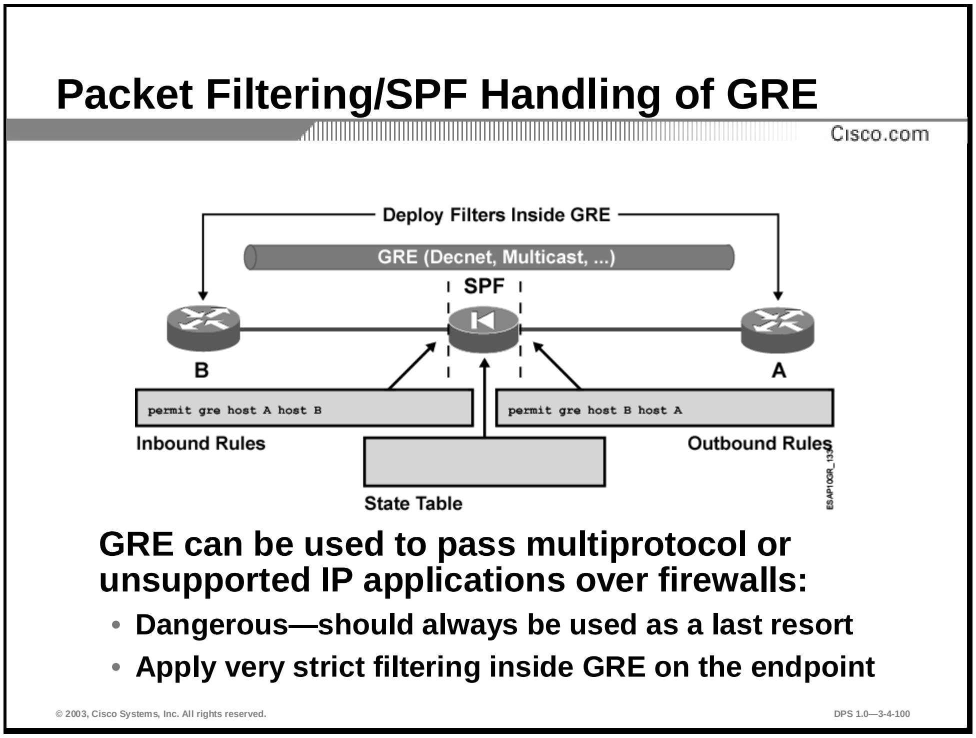
Some organizations pass VPN protocols through firewalls for two reasons:

- They terminate VPN connections on the inside or on a leg of a firewall
- They use the VPN protocol to pass an unsupported protocol over the firewall. For example, IP multicast, DECnet, IPX

Running a VPN over a firewall has significant disadvantages, which must be reviewed and perhaps mitigated:

- When a VPN protocol passes over the firewall it does not see the enveloped content, which is usually encrypted. Therefore, access control is extremely coarse if the firewall cannot filter traffic after it has been decapsulated.

- If the VPN allows unauthorized traffic to enter, it bypasses the firewall and violate the policy.



## Packet Filtering/SPF Handling of GRE

Cisco.com

**GRE can be used to pass multiprotocol or unsupported IP applications over firewalls:**

- **Dangerous—should always be used as a last resort**
- **Apply very strict filtering inside GRE on the endpoint**

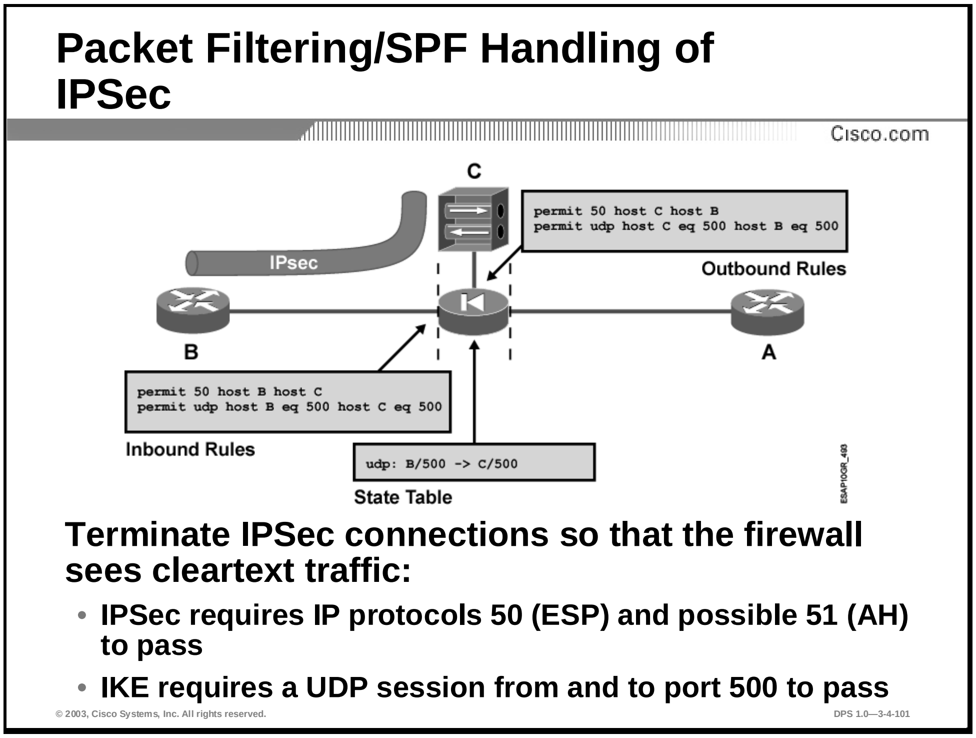
DPS 1.0—3-4-100

## (Stateful) Packet Filtering of GRE

This figure illustrates an example of running a foreign protocol such as IPX, over an IP firewall by using a generic routing encapsulation (GRE) tunnel. Such workarounds are always discouraged (but sometimes are required as the only means to extend non-IP protocols, etc), as a small misconfiguration on GRE endpoints can compromise the firewall. The two routers on both sides of the firewall create a GRE session, which the firewall permits. This is functionally equivalent to having a back-to-back cable or leased line between the two routers—the firewall is unable to enforce access control on tunneled traffic.

Such workarounds should always be used as a last resort, and defense-in-depth should be practiced. Additional filters should block any attempts to run IP unicast in the tunnel.

**Note** GRE can also be used inside IPSec tunnels to provide routing protocol functionality inside IPSec VPN tunnels. In such a setup, usually both sides of the VPN connection belong to trusted perimeters, and the aforementioned warnings do not apply.



## Packet Filtering/SPF Handling of IPSec

```
permit 50 host C host B
permit udp host C eq 500 host B eq 500
```

```
permit 50 host B host C
permit udp host B eq 500 host C eq 500
```

```
udp: B/500 -> C/500
```

**Terminate IPSec connections so that the firewall sees cleartext traffic:**

- **IPSec requires IP protocols 50 (ESP) and possible 51 (AH) to pass**
- **IKE requires a UDP session from and to port 500 to pass**

DPS 1.0—3-4-101

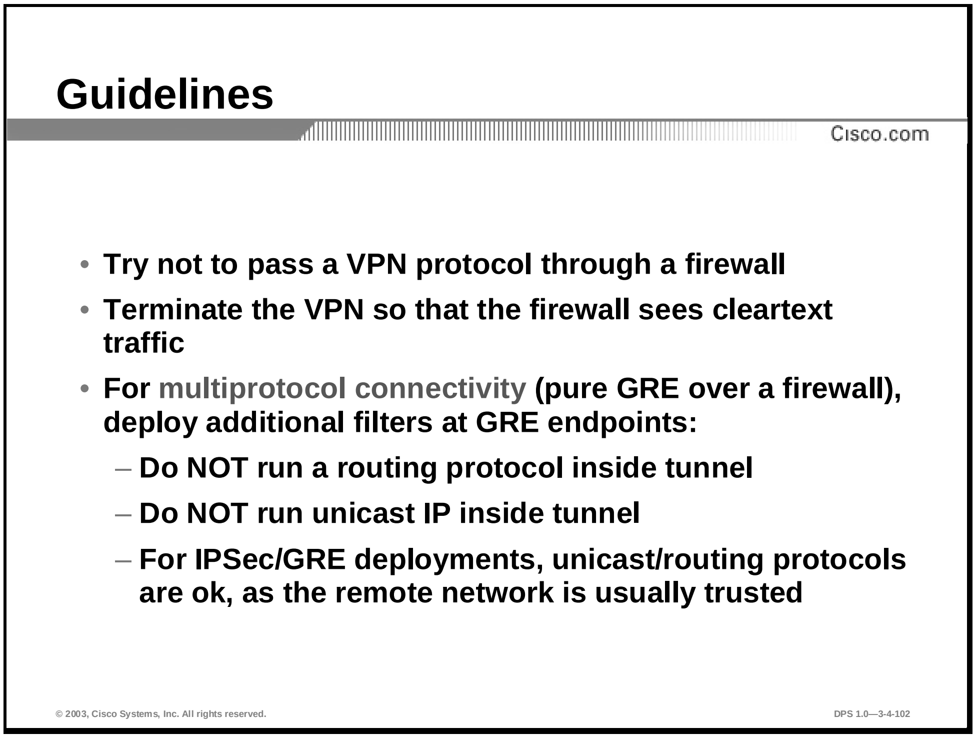
## (Stateful) Packet Filtering of IPSec Protocols

If an IPSec VPN needs to be integrated with a firewall system, the encrypted tunnels should ideally terminate at the firewall as to decrypt all traffic before it passes through the firewall for the most granular access control. This can be accomplished by either have the firewall filter (for example, the PIX Firewall) terminate IPSec, or a dedicated VPN system terminating it inside the firewall architecture.

Passing IPSec through a firewall filter (for example, to the dedicated termination device) requires the firewall to pass:

- IP protocol 50 if the ESP (Encapsulating Security Payload) IPSec encapsulation is used or
- IP protocol 51 if the AH (Authenticating Header) IPSec encapsulation is used

Besides those tunnel packets, the IKE (Internet Key Exchange) packets need to be permitted between IPSec peers. IKE uses a UDP session with the source and destination port of 500.



## Guidelines

- **Try not to pass a VPN protocol through a firewall**
- **Terminate the VPN so that the firewall sees cleartext traffic**
- **For multiprotocol connectivity (pure GRE over a firewall), deploy additional filters at GRE endpoints:**
  - **Do NOT run a routing protocol inside tunnel**
  - **Do NOT run unicast IP inside tunnel**
  - **For IPSec/GRE deployments, unicast/routing protocols are ok, as the remote network is usually trusted**

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—3-4-102

# Guidelines

If a VPN protocol runs through the firewall, it is best not to pass it through the whole firewall encapsulated. Terminate the VPN in such a way, so that the firewall will be able to perform access control on decapsulated (cleartext) traffic.

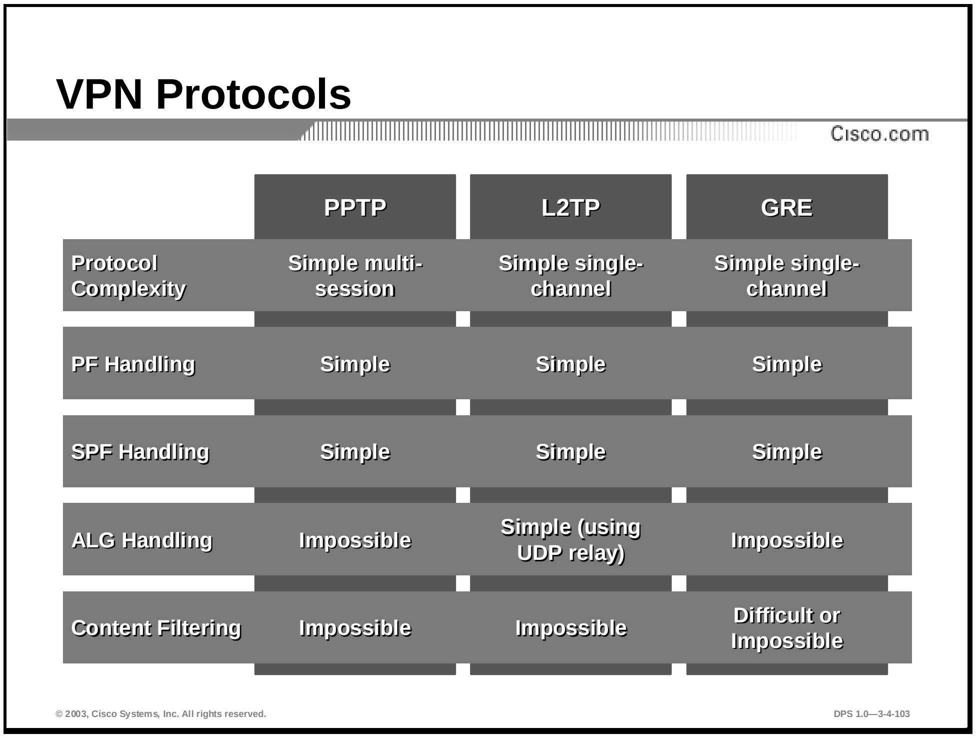
# Pure GRE Connectivity (Firewall Bypass)

If GRE is desired for multiprotocol connectivity over a firewall (and the GRE endpoints are on networks with different levels of trust), ensure that IP unicast can never run through the GRE tunnel. Never run an IP routing protocol inside the GRE tunnel, as the routing protocol may attract unwanted traffic into the tunnel automatically, bypassing the firewall.

**Note** Again, GRE can also be used inside IPSec tunnels to provide routing protocol functionality inside IPSec VPN tunnels. In such a setup, usually both sides of the VPN connection belong to a trusted perimeters, and the aforementioned warnings do not apply.

# Example

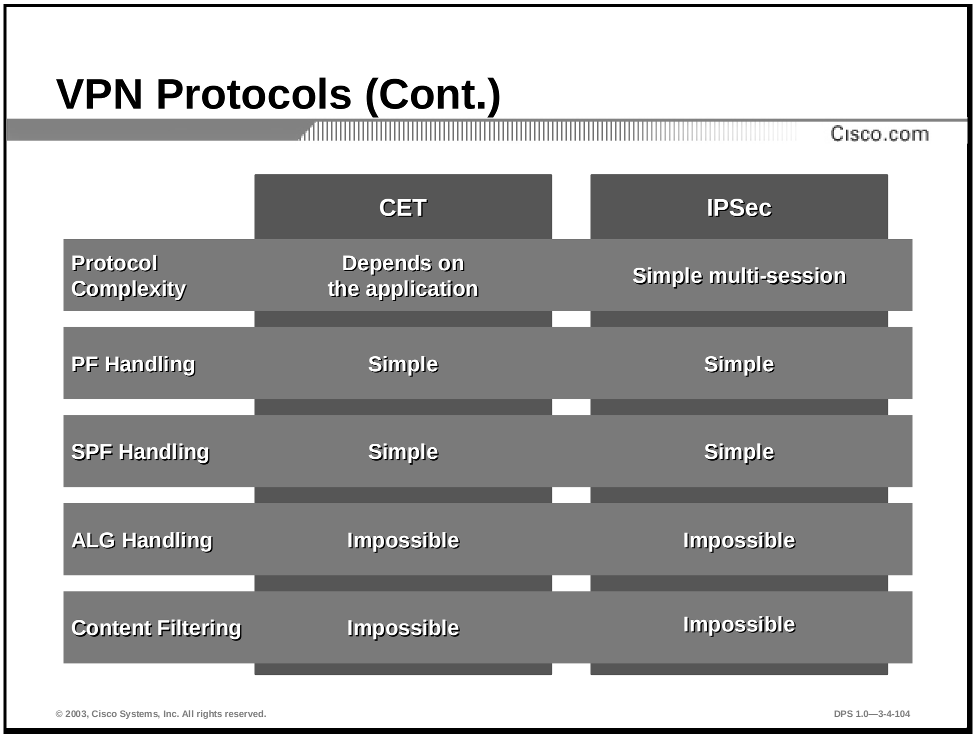
If the GRE tunnel is only intended to carry IPX, do not configure IP addresses on the tunnel to prevent IP routing through it. Deploy additional barriers such as access control lists (ACLs), denying all IP traffic, or policy routing of IP traffic to the null interface, which should disable unicast IP running through it if misconfigured.



## VPN Protocols

| | PPTP | L2TP | GRE |
|---|---|---|---|
| Protocol Complexity | Simple multi-session | Simple single-channel | Simple single-channel |
| PF Handling | Simple | Simple | Simple |
| SPF Handling | Simple | Simple | Simple |
| ALG Handling | Impossible | Simple (using UDP relay) | Impossible |
| Content Filtering | Impossible | Impossible | Difficult or Impossible |

DPS 1.0—3-4-103



## VPN Protocols (Cont.)

| | CET | IPSec |
|---|---|---|
| Protocol Complexity | Depends on the application | Simple multi-session |
| PF Handling | Simple | Simple |
| SPF Handling | Simple | Simple |
| ALG Handling | Impossible | Impossible |
| Content Filtering | Impossible | Impossible |

DPS 1.0—3-4-104

## Practice

Q1) Which protocols need to be permitted through a firewall for an IPSec tunnel to pass?

A) IP protocols 50 and 51

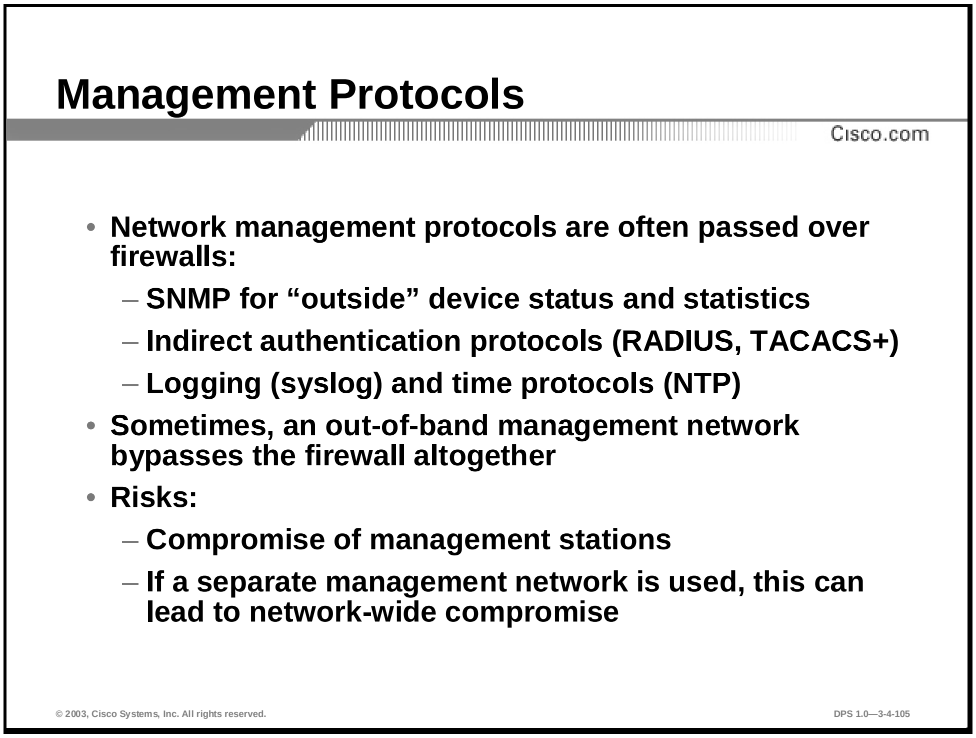
B) IP protocols 50 and 51, and a UDP session from port 500 to port 500

C) IP protocols 500 and 501, and a UDP session from port 500 to port 500

D) UDP ports 50, 51, and 500

E) IP protocols 47, 50, and 51

# Management Protocols



## Management Protocols

- Network management protocols are often passed over firewalls:
  - SNMP for “outside” device status and statistics
  - Indirect authentication protocols (RADIUS, TACACS+)
  - Logging (syslog) and time protocols (NTP)
- Sometimes, an out-of-band management network bypasses the firewall altogether
- Risks:
  - Compromise of management stations
  - If a separate management network is used, this can lead to network-wide compromise

© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—3-4-105

## Objective

The section will enable the learner to explain the security properties of management protocols and to choose an appropriate technology to securely pass them over firewalls.

## Introduction

Management protocols often pass through firewalls, either to manage outside (less trusted) devices, or inbound to manage devices on a more trusted network. If in-band management over the firewall is preferred, a variety of management and supporting protocols need to be supported by a firewall.

## Management Protocols’ Refresher

Firewalls are often required to pass management protocols between perimeters. Examples include:

- **Simple Network Management Protocol (SNMP):** To access devices outside the firewall to monitor their status and collect statistics.

- **Indirect authentication protocols (TACACS+, RADIUS):** Often run inbound through firewalls to allow outside devices to authenticate against an inside authentication server.

- **Logging protocols (for example, SYSLOG):** Often allowed so outside devices can to log onto an inside logging server. SYSLOG is an unreliable protocol, which has no authentication or integrity mechanisms, therefore a trusted path is necessary to pass it securely between devices and syslog servers.

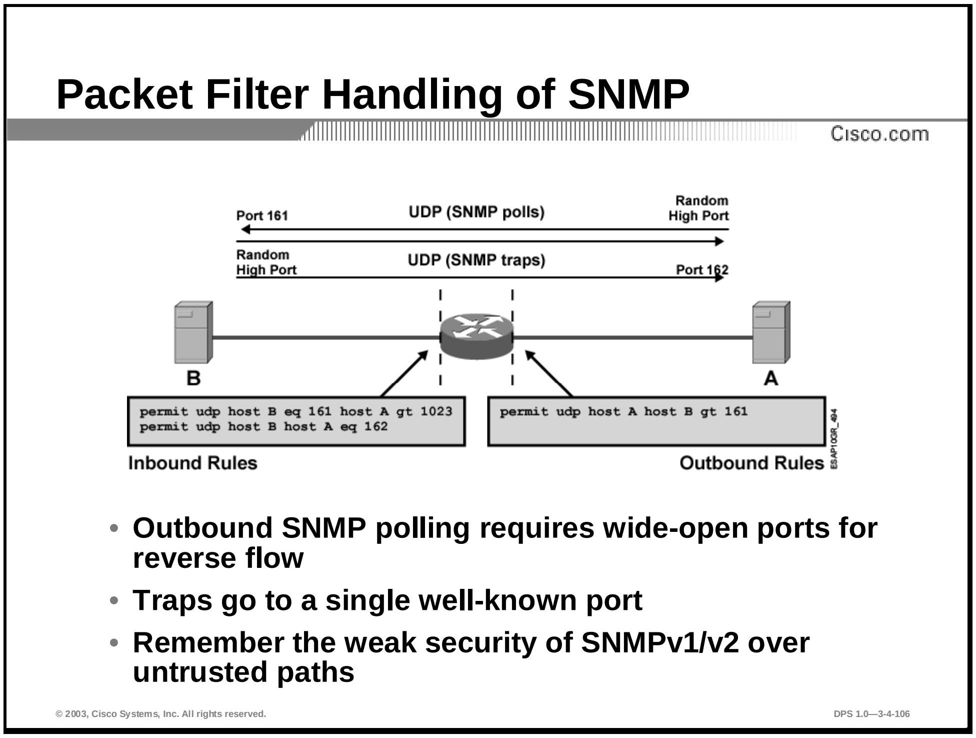
- **Time protocols:** Allowed outbound or inbound to enable time synchronization between devices in different perimeters.

Some organizations opt for an out-of-band management network, where every device is connected to a common management network. This greatly simplifies management but introduces a dangerous backdoor. If a device is compromised, all other devices, possibly in other perimeters, can be attacked over the out-of-band network. Possible solutions include having a separate out-of-band management network for every perimeter, or using private VLANs to disallow connectivity between devices.

The risks of using management protocols include:

- An allowed inbound session to an exposed management station might be used to break into the management station

- If a separate management network is used, an attacker can gain access to the management station, and from it access other parts of the network that are normally all accessible from the management network



## Packet Filter Handling of SNMP

Cisco.com

```
permit udp host B eq 161 host A gt 1023
permit udp host B host A eq 162
```

```
permit udp host A host B gt 161
```

- **Outbound SNMP polling requires wide-open ports for reverse flow**
- **Traps go to a single well-known port**
- **Remember the weak security of SNMPv1/v2 over untrusted paths**

DPS 1.0—3-4-106

## Packet Filter Handling of SNMP

The (in)security of the SNMP protocol is well-known, and can be summarized as:

- SNMP offers no confidentiality or integrity mechanisms, and therefore needs a trusted path if those two properties are desired
- SNMP authentication is weak in SNMPv1 and SNMPv2 (cleartext authentication), providing yet another reason not to run it over untrusted networks
- SNMPv3 adds the support for encrypted authentication, but its deployment and support in devices and management software is limited

## Example

Some examples of SNMP-related problems would be:

- If an attacker has compromised a system on the network and installed a packet-sniffer program, that person could compromise your entire network infrastructure provided SNMP read-write access was enabled.
- If you use in-band management over a firewall, and the firewall is under serious attackk, you may lose your management traffic altogether, if traffic stops passing through the firewall.

## SNMP on the Network

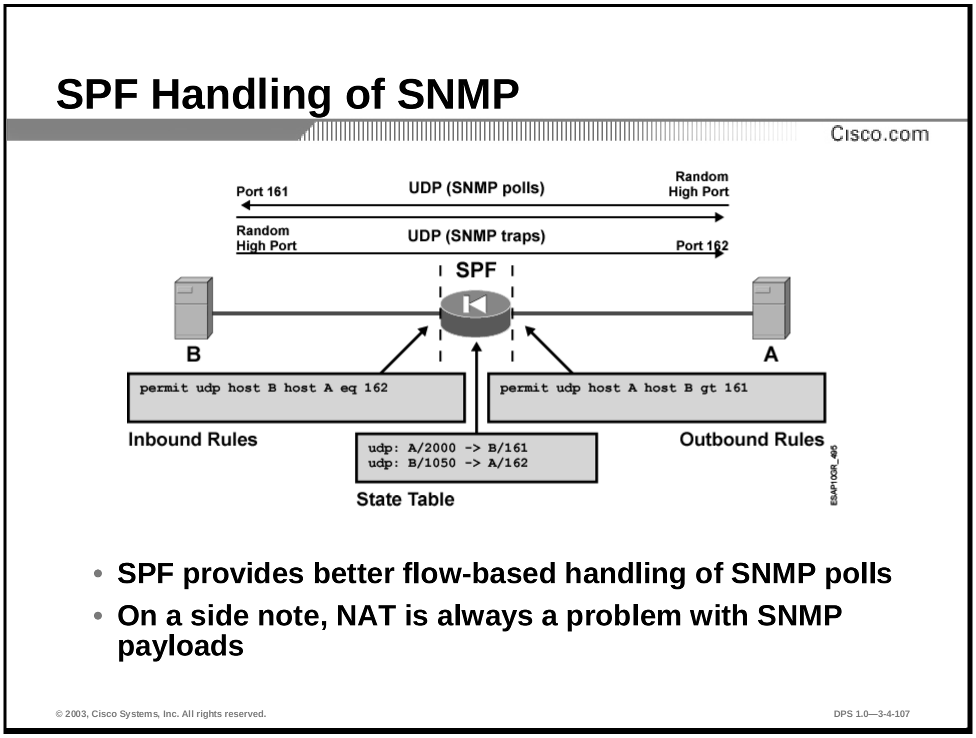
On the network, SNMP has two types of transactions:

- SNMP polling, where the NMS (network management system) sends UDP packets with a random source port and a destination port of 161 to the device, which responds to the poll to the server random port (a UDP ping-pong transaction, like DNS)

- Asynchronous sending of SNMP traps (exceptional events) to the NMS well-known UDP port 162

## Packet Filter Handling of SNMP

Packet filter therefore cannot support filtering of SNMP polls securely, as this would require opening of all UDP ports to the NMS. Traps can be filtered more robustly, as only a single port needs to be exposed on the NMS.



## SPF Handling of SNMP

Cisco.com

```
permit udp host B host A eq 162
```

```
permit udp host A host B gt 161
```

```
udp: A/2000 -> B/161
udp: B/1050 -> A/162
```

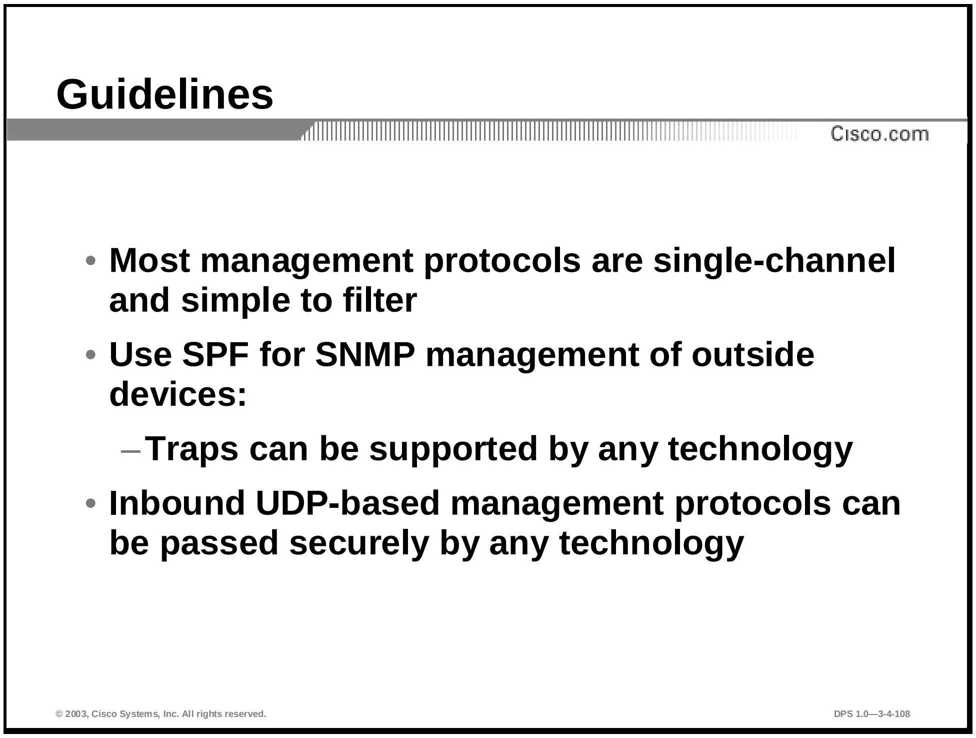
- **SPF provides better flow-based handling of SNMP polls**
- **On a side note, NAT is always a problem with SNMP payloads**

DPS 1.0—3-4-107

## SPF Handling of SNMP

SPFs provide much more robust handling of SNMP, as polls are treated as UDP flows, which the stateful engine will track; therefore, exposure of the NMS host is minimized.

On a side note, it is worth mentioning that the addresses inside SNMP messages are not translated by NAT engines, although many people would expect them to be.



## Guidelines

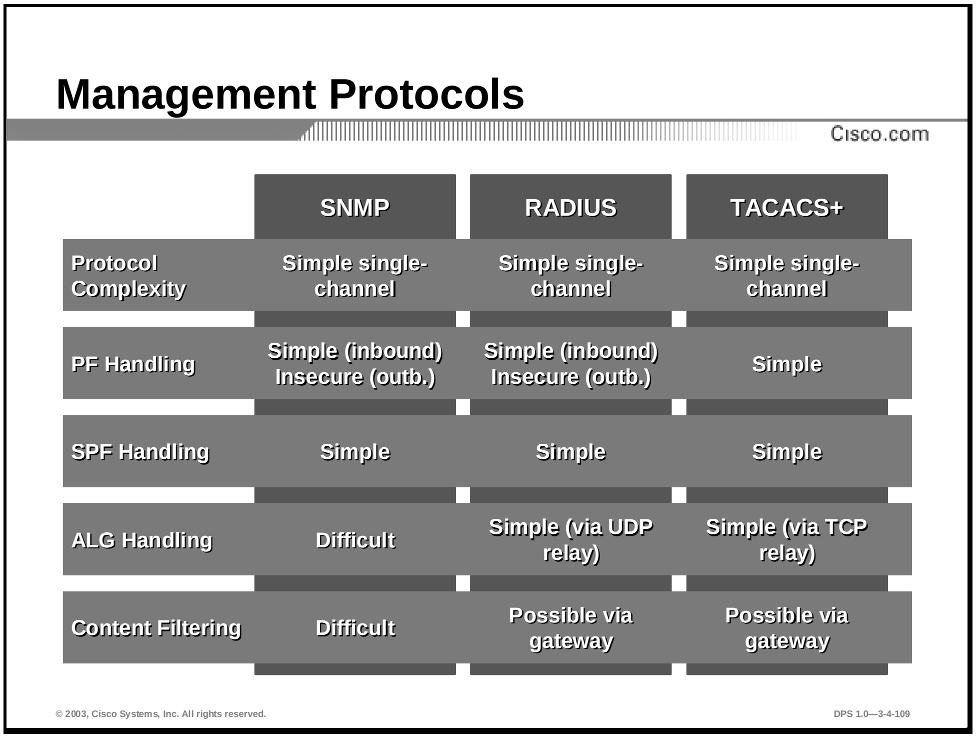
- Most management protocols are single-channel and simple to filter
- Use SPF for SNMP management of outside devices:
  - Traps can be supported by any technology
- Inbound UDP-based management protocols can be passed securely by any technology

DPS 1.0—3-4-108

# Guidelines

Most management protocols are single-channel TCP or UDP, and therefore simple to filter:

- Passing single-channel TCP management protocols inbound or outbound is possible using any firewall technology
- SPFs should handle outbound UDP-based applications
- Any technology can securely support inbound UDP connectivity, as only a single port needs to be permitted to the inside

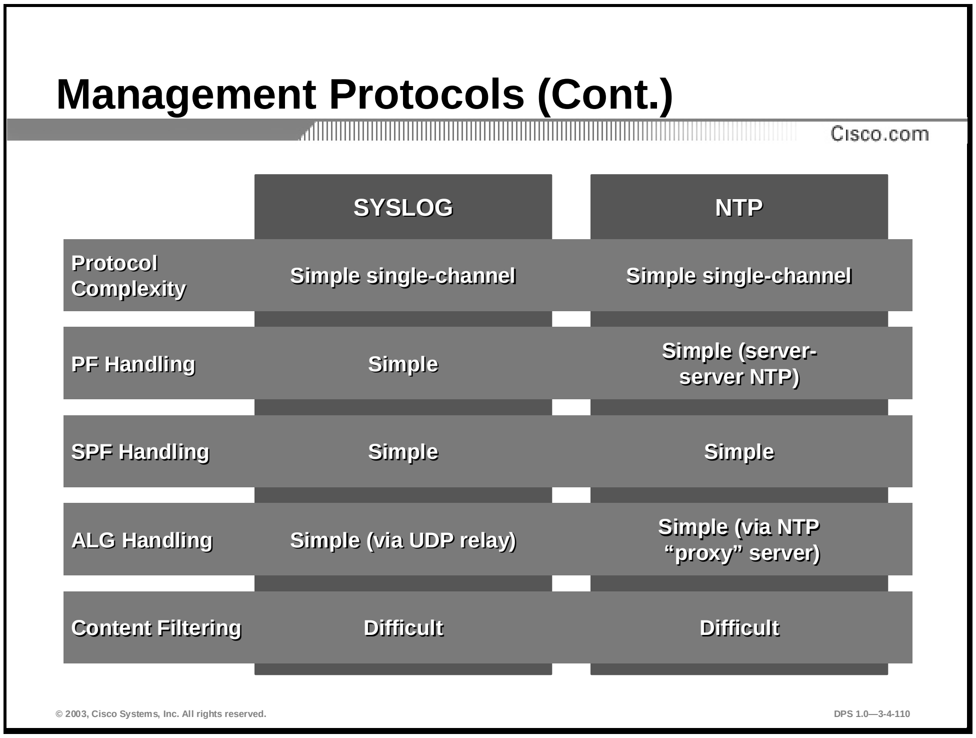


## Management Protocols

Cisco.com

| | SNMP | RADIUS | TACACS+ |
|---|---|---|---|
| Protocol Complexity | Simple single-channel | Simple single-channel | Simple single-channel |
| PF Handling | Simple (inbound)<br>Insecure (outb.) | Simple (inbound)<br>Insecure (outb.) | Simple |
| SPF Handling | Simple | Simple | Simple |
| ALG Handling | Difficult | Simple (via UDP relay) | Simple (via TCP relay) |
| Content Filtering | Difficult | Possible via gateway | Possible via gateway |

DPS 1.0—3-4-109



## Management Protocols (Cont.)

Cisco.com

| | SYSLOG | NTP |
|---|---|---|
| Protocol Complexity | Simple single-channel | Simple single-channel |
| PF Handling | Simple | Simple (server-server NTP) |
| SPF Handling | Simple | Simple |
| ALG Handling | Simple (via UDP relay) | Simple (via NTP “proxy” server) |
| Content Filtering | Difficult | Difficult |

DPS 1.0—3-4-110

# Practice

Q1) Which technology can pass SNMP traps securely enough (without allowing other traffic)?

A) packet filters only

B) SPFs only

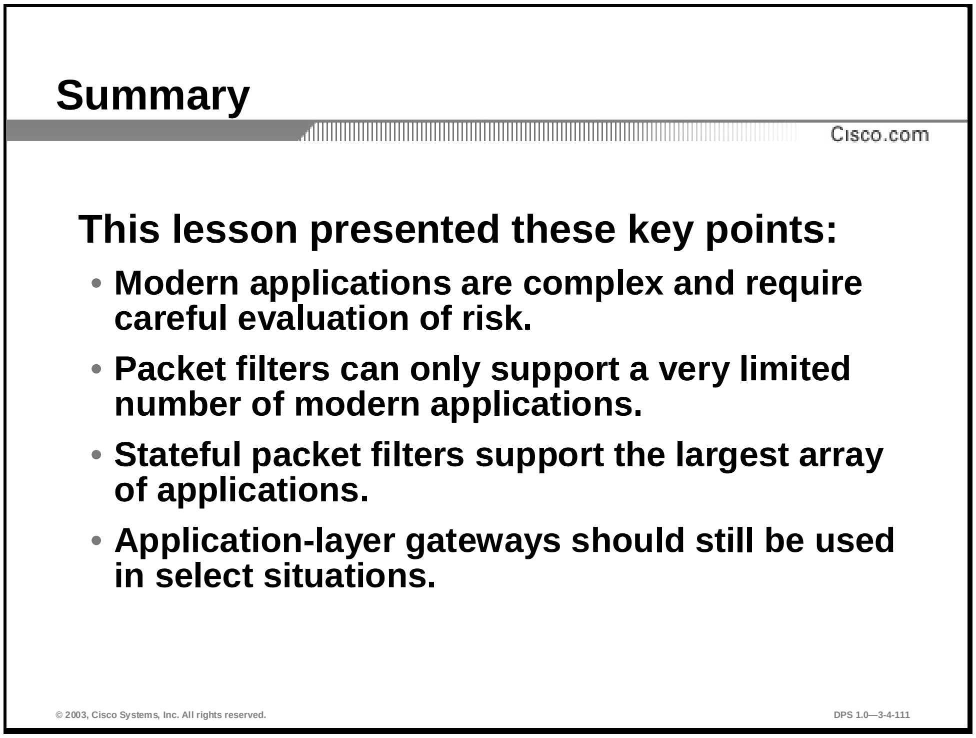
C) ALGs only

D) session-level gateways only

E) any technology (packet filters, SPFs, ALGs)

# Summary

This section summarizes the key points discussed in this lesson.



## Summary

Cisco.com

**This lesson presented these key points:**

- **Modern applications are complex and require careful evaluation of risk.**
- **Packet filters can only support a very limited number of modern applications.**
- **Stateful packet filters support the largest array of applications.**
- **Application-layer gateways should still be used in select situations.**

DPS 1.0—3-4-111

## Next Steps

After completing this lesson, go to:

- Perimeter Security Design module, Firewall Design General Guidelines lesson

# Quiz: Firewall Handling of Protocols

Complete the quiz to assess what you have learned in this lesson.

## Objectives

This quiz tests your knowledge on how to:

- Select an appropriate firewall technology for an organization’s application needs

## Instructions

Answer these questions:

1. Which ICMP messages are required to pass over firewalls?
2. How is voice traffic handled in a “hybrid” SPF/ALG firewall?
3. What is a major risk of inbound remote terminal connections?
4. Why is it not recommended for VPN protocols to cross firewalls?

## Scoring

You have successfully completed the quiz for this lesson when you earn a score of 80 percent or better.

# Firewall Design General Guidelines

## Overview

### Importance

Although firewall design is considered by many to be more an art than a science, some general guidelines exist and must be applied in any firewall design. This lesson presents those general guidelines and is the highest importance for a network security designer.

### Lesson Objective

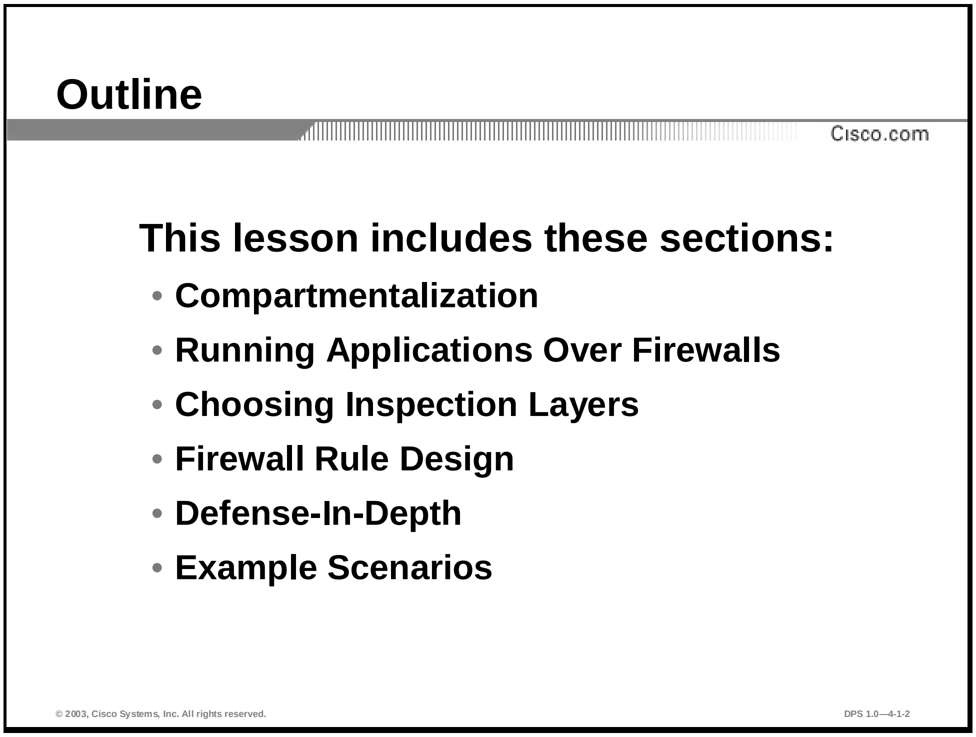
Upon completing this lesson, you will be able to design an abstract firewall system, enforcing a defined security policy, and using best practice design methods.

# Learner Skills and Knowledge

To fully benefit from this lesson, you must have these prerequisite skills and knowledge:

- Understand the concept of a firewall
- Select the appropriate technology and architecture of a firewall

# Outline



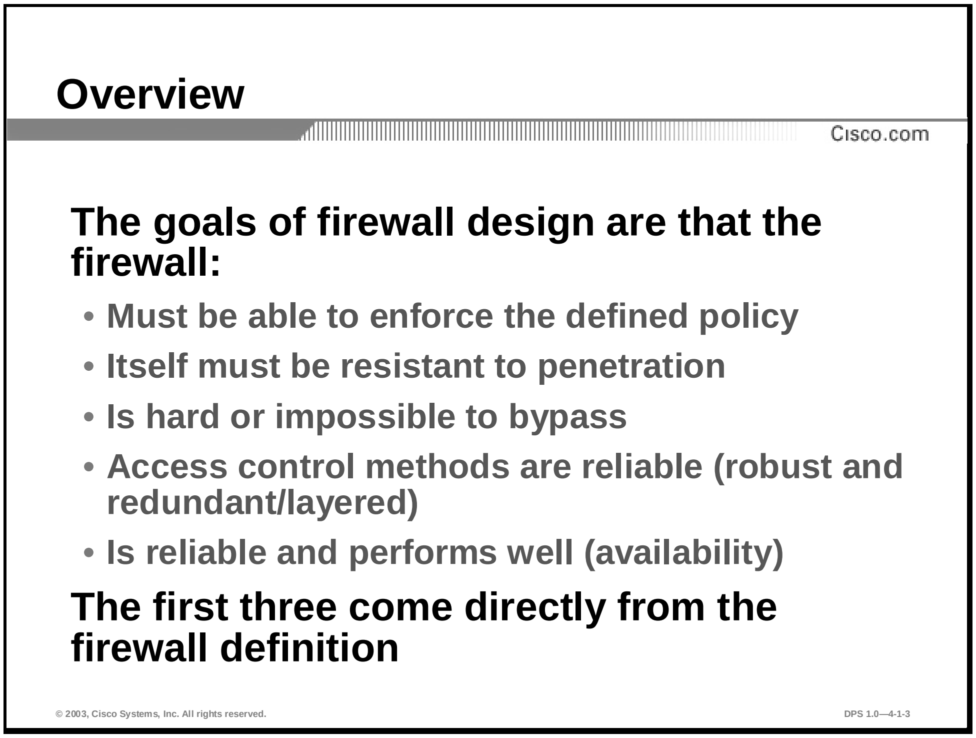
## Outline

Cisco.com

**This lesson includes these sections:**

- **Compartmentalization**
- **Running Applications Over Firewalls**
- **Choosing Inspection Layers**
- **Firewall Rule Design**
- **Defense-In-Depth**
- **Example Scenarios**

# Overview



## Overview

**The goals of firewall design are that the firewall:**

- **Must be able to enforce the defined policy**
- **Itself must be resistant to penetration**
- **Is hard or impossible to bypass**
- **Access control methods are reliable (robust and redundant/layered)**
- **Is reliable and performs well (availability)**

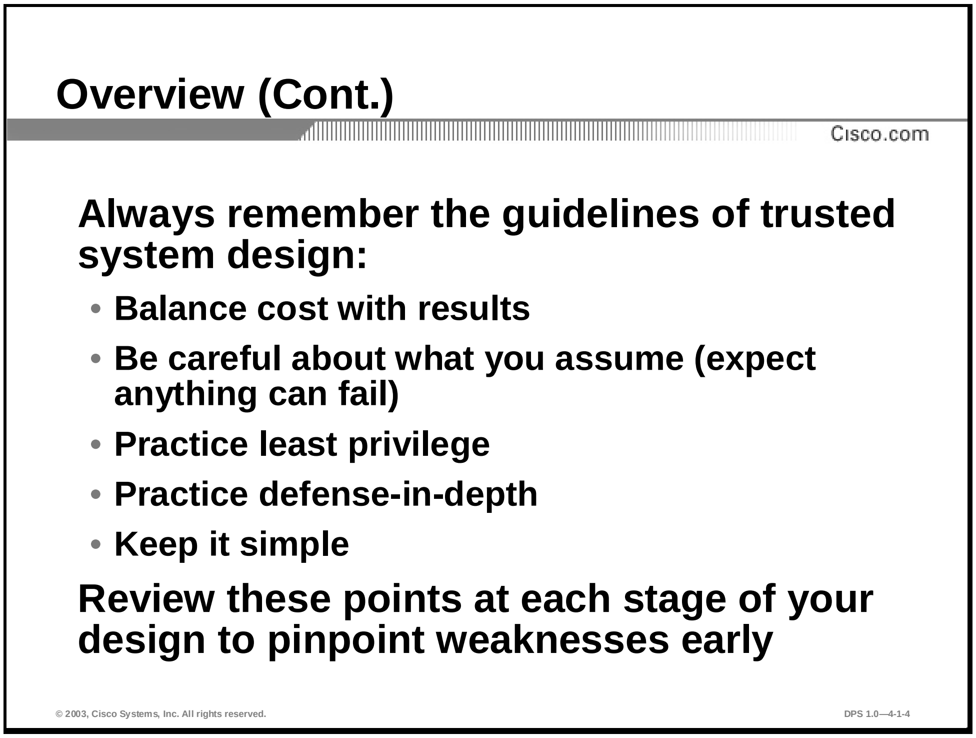
**The first three come directly from the firewall definition**

DPS 1.0—4-1-3

The goals of firewall design are that the firewall:

- Must be able to enforce the defined policy
- Itself must be resistant to penetration
- Is hard or impossible to bypass
- Access control methods are reliable (robust and redundant/layered)
- Is reliable and performs well (availability)

The first three goals are directly from the firewall definition, while the other two goals are provided by good firewall design techniques.



## Overview (Cont.)

Cisco.com

**Always remember the guidelines of trusted system design:**

- **Balance cost with results**
- **Be careful about what you assume (expect anything can fail)**
- **Practice least privilege**
- **Practice defense-in-depth**
- **Keep it simple**

**Review these points at each stage of your design to pinpoint weaknesses early**

DPS 1.0—4-1-4

Firewalls are security systems; therefore the main guidelines of trusted system design also apply to them. Those guidelines are:

- **Balance cost with results:** The investment in protection should never exceed the estimated cost of intrusion.

- **Be careful about what assumptions:** Always question the validity of assumptions, and remember that risks change in time. What is trusted today may be inadequate tomorrow. Additionally, expect anything can fail to provide different levels of defense-in-depth protection.

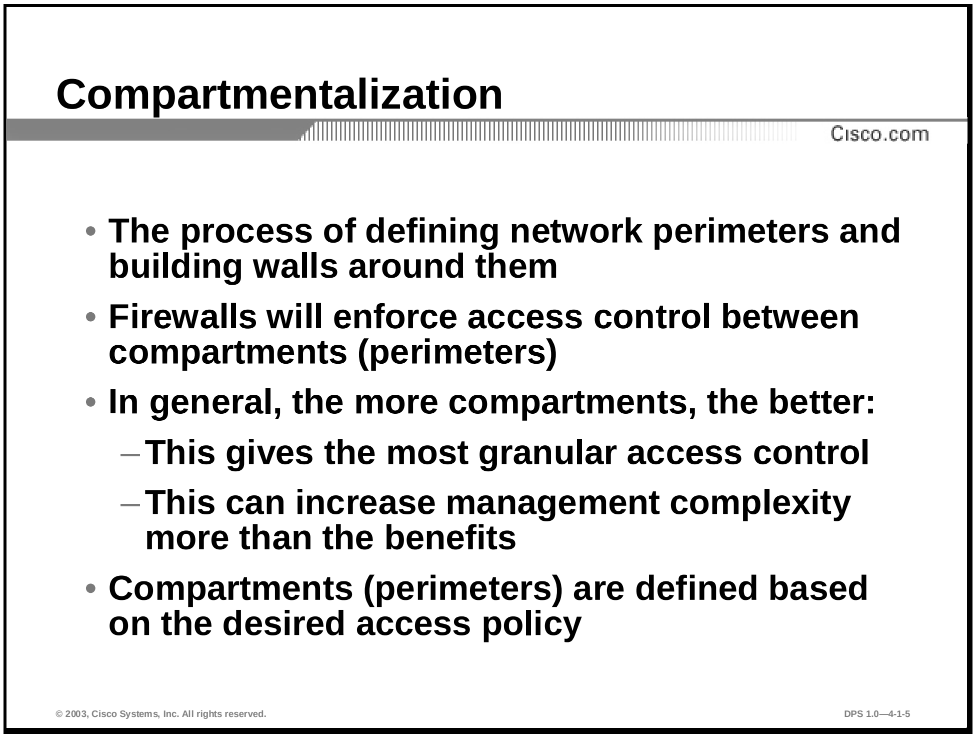
- **Practice least privilege:** For any task that needs to be performed by a human or a computer process, assign the lowest possible privileges that are just sufficient to perform that task.

- **Practice defense-in-depth:** Provide multiple independent protection mechanisms for critical security mechanisms.

- **Keep it simple:** Complexity is the worst enemy of security, and keeping things simple makes them verifiable and robust.

As the design progresses, keep checking these points to pinpoint any potential design weaknesses before system deployment.

# Compartmentalization



## Compartmentalization

Cisco.com

- The process of defining network perimeters and building walls around them
- Firewalls will enforce access control between compartments (perimeters)
- In general, the more compartments, the better:
  - This gives the most granular access control
  - This can increase management complexity more than the benefits
- Compartments (perimeters) are defined based on the desired access policy

© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—4-1-5

## Objectives

This section will enable the learner to explain the principle of compartmentalization and design a firewall system using it.

## Introduction

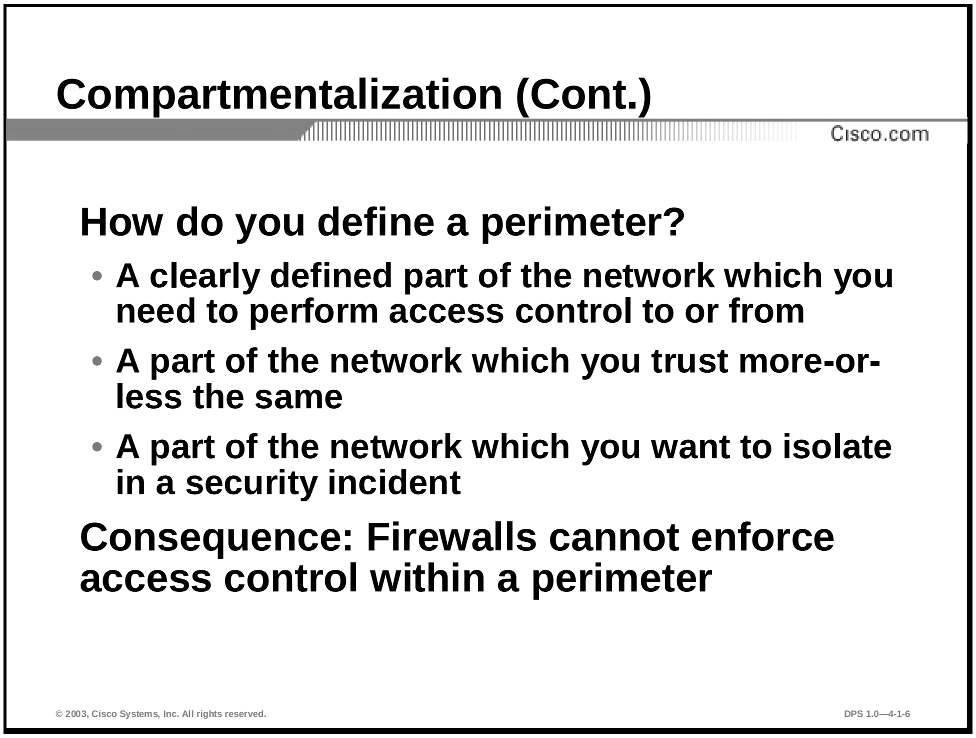
The first principle of firewall design is compartmentalization. The definition of compartmentalization is “the formation of perimeters”. A firewall enforces access control between perimeters; therefore define the perimeters correctly to fully implement an access policy.

## Compartmentalization

In general, the more the designer compartmentalizes a network, the more granular the access policy can be that is deployed between the compartments (perimeters). However, do not overdo compartmentalization—only divide the network into as many perimeters as necessary. A policy requirement that identifies the subjects to be grouped together and treated as a single entity in access control determines this division. Compartmentalization also increases management complexity (more firewall interfaces, larger access rules); therefore balance it with the benefits.

## Example

The simplest Internet firewall might require only two perimeters: the organization's trusted network, and the Internet. A firewall will then enforce access control between the two perimeters connected to its interfaces.



## Compartmentalization (Cont.)

How do you define a perimeter?

- A clearly defined part of the network which you need to perform access control to or from
- A part of the network which you trust more-or-less the same
- A part of the network which you want to isolate in a security incident

Consequence: Firewalls cannot enforce access control within a perimeter

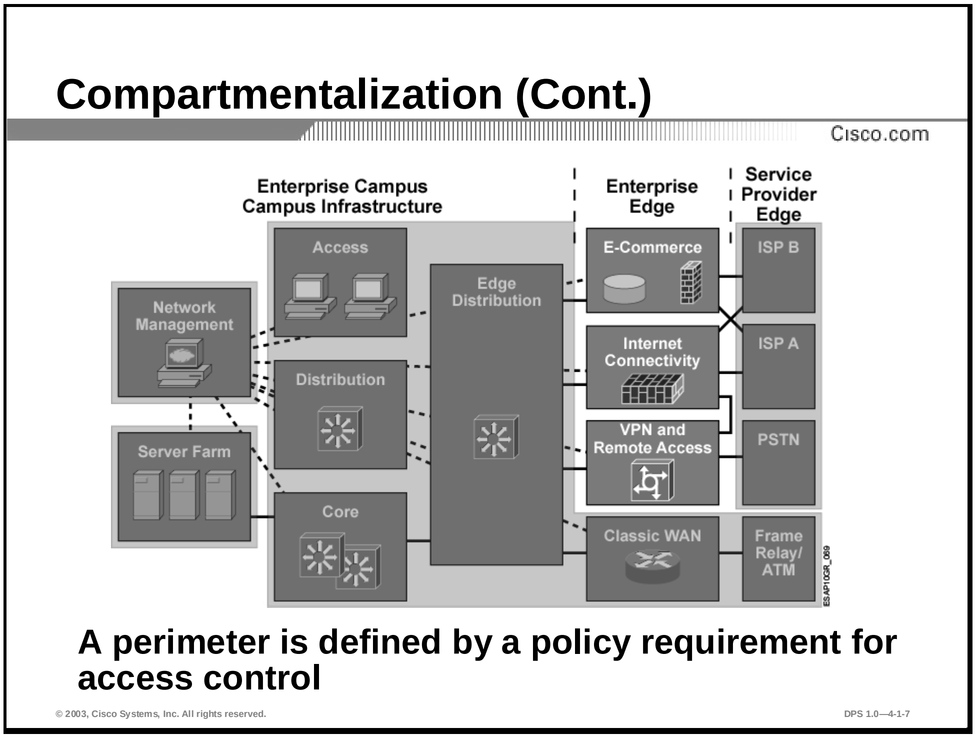
DPS 1.0—4-1-6

A good, basic definition of a perimeter is that a perimeter is a “clearly defined part of the network, which a firewall will perform access control to or from”. A perimeter connects to a firewall network interface. The firewall controls all traffic flow to or from that perimeter, to the other perimeters connected to the same firewall.

Alternative definitions of perimeters include:

- A perimeter is a part of the network that is trusted more-or-less the same. For example, the Internet perimeter, the business partners’ perimeter (or perhaps each business partner might be a separate perimeter), and the internal web server farm perimeter, contain a group of hosts or users, which we consider untrusted, semi-trusted, and very trusted respectively.
- A perimeter is a part of network, which needs to be isolated in a security incident. For example, a DMZ network on a screened subnet firewall can be defined as a small perimeter, and designed only to host a single server or few similar servers.

Based on the definition that “access to a perimeter is controlled by a firewall”, it follows that such firewalls cannot enforce access control within the perimeter.



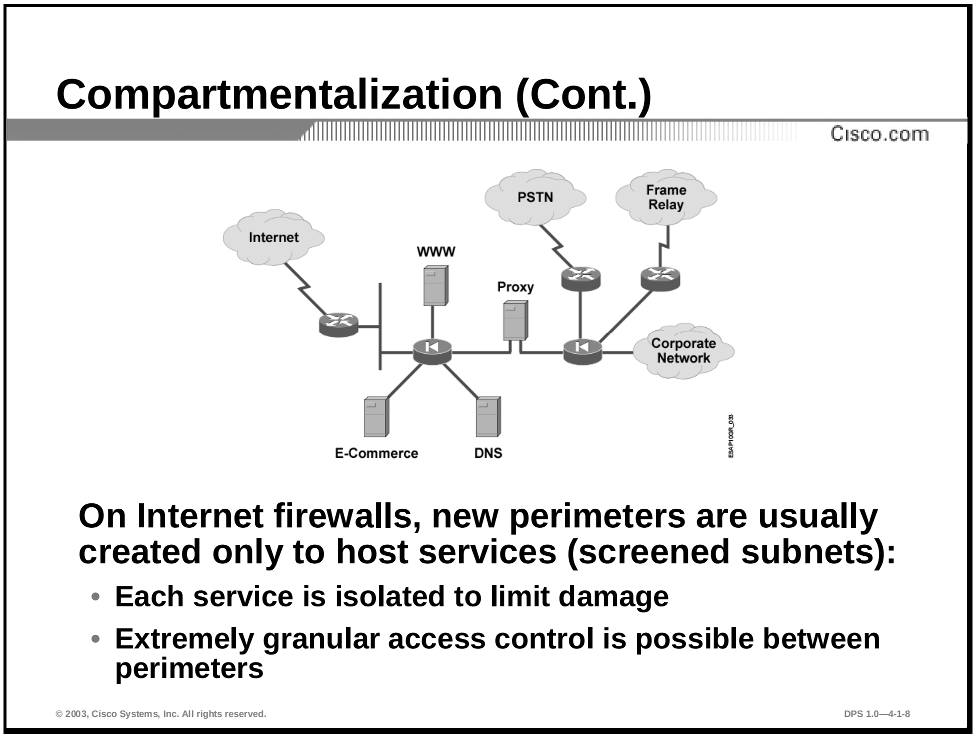
## Compartmentalization (Cont.)

Cisco.com

**A perimeter is defined by a policy requirement for access control**

DPS 1.0—4-1-7

This figure illustrates an example of compartmentalization, where the policy requirements define the perimeters. If the policy specifies that access control needs to be provided between the management LAN, server farms, the switched campus network and WAN, and the Internet, those network segments define the required perimeters.



## Compartmentalization (Cont.)

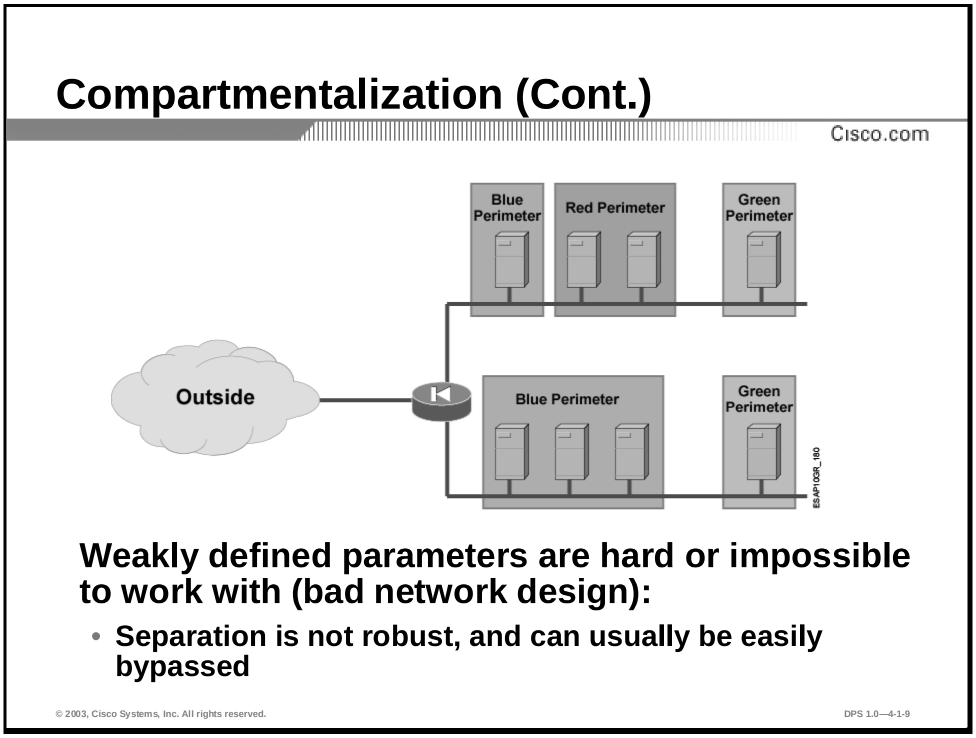
Cisco.com

**On Internet firewalls, new perimeters are usually created only to host services (screened subnets):**

- **Each service is isolated to limit damage**
- **Extremely granular access control is possible between perimeters**

DPS 1.0—4-1-8

This figure illustrates a well-known use of perimeters in a screened-subnet firewall architecture. Any screened subnet attached to a firewall filter is a standalone perimeter, whose purpose is to host exposed services, or connect an external network to the firewall over its own interface. In this case, intrusions can often be isolated to a particular perimeter, especially if the firewall enforces least-privilege access control, making it difficult to enter other perimeters. Additionally, the highly compartmentalized network enables very granular access control inside the network if there is a good separation of edge services and external connection.



## Compartmentalization (Cont.)

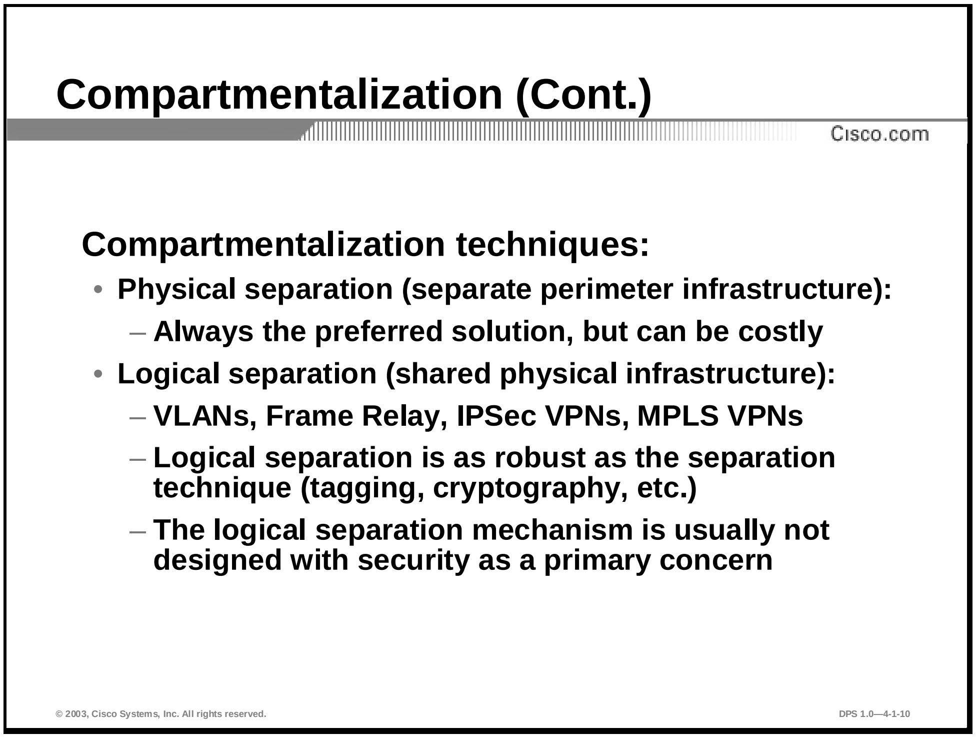
Cisco.com

**Weakly defined parameters are hard or impossible to work with (bad network design):**

- **Separation is not robust, and can usually be easily bypassed**

DPS 1.0—4-1-9

This figure illustrates a weakly defined perimeter, where perimeters are defined as clusters of systems reachable on multiple firewall interfaces. This violates the basic definition of a perimeter and makes policy enforcement difficult, especially because multiple perimeters share the same firewall interface and hopping between perimeters is usually possible. In this case, a redesign of the network is required before implementing network access control.



## Compartmentalization (Cont.)

**Compartmentalization techniques:**

- **Physical separation (separate perimeter infrastructure):**
  - **Always the preferred solution, but can be costly**
- **Logical separation (shared physical infrastructure):**
  - **VLANs, Frame Relay, IPSec VPNs, MPLS VPNs**
  - **Logical separation is as robust as the separation technique (tagging, cryptography, etc.)**
  - **The logical separation mechanism is usually not designed with security as a primary concern**

DPS 1.0—4-1-10

Compartmentalization is usually achieved using pure physical separation of networks. That is, a firewall connects to two physically distinct network infrastructures, and therefore all traffic *must* pass through the firewall, if the firewall is the only interconnection point. Such separation seems logical, and is always the best method of perimeter separation.

However, such separation can become costly, especially when multiple levels of trust need to be distinguished in large access networks.

## Example

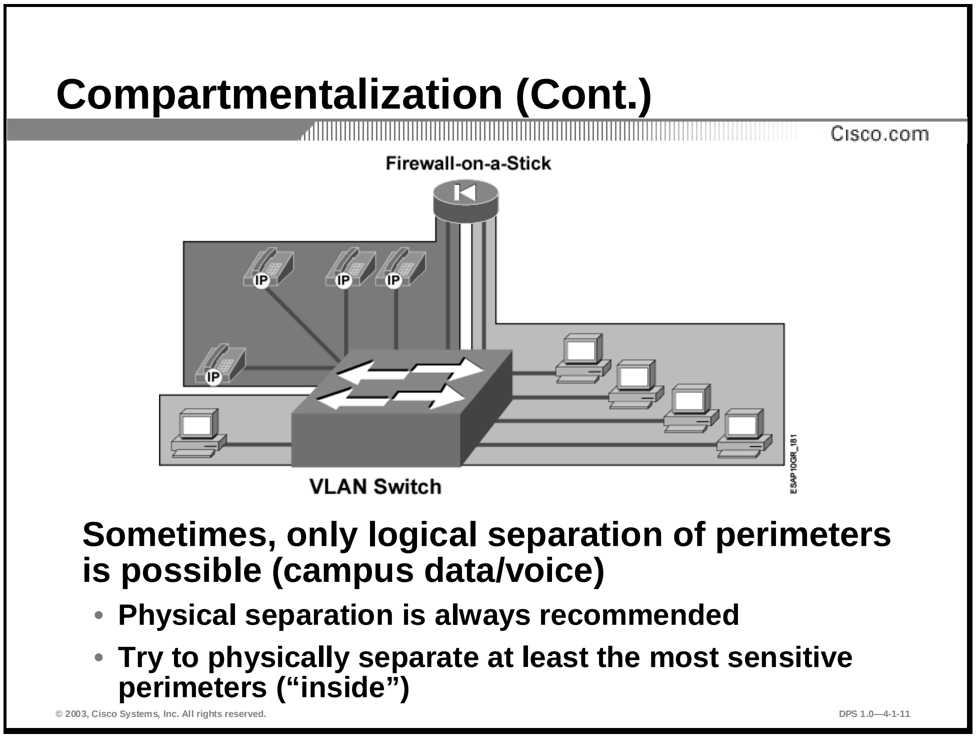
A network administrator needs to assign two classes of users different access rights in a network, which is inside a large campus switched network. Both classes of users connect to the same physical infrastructure, as it is not viable from the cost perspective to maintain two physically separate networks. Therefore, the network administrator needs to logically separate the user groups.

Logical separation provides separate communication channels for different users over the same physical infrastructure. Such separation enables subjects to connect to the same physical infrastructure, and only be able to access their own perimeter, which is a logical subset of the physical infrastructure. Such separation methods include:

- Virtual LAN (VLAN) and private VLAN technology of LAN switches, Frame Relay (FR), or Multiprotocol Label Switching (MPLS)/VPNs where tagging of LAN frames provides separation

- IPSec VPNs, where encryption provides separation

When using logical separation of perimeters, access control is as strong as the perimeter separation technique. The main problem of logical separation is usually that the separation mechanism design is to enable higher performance (VLANs) or to solve addressing problems (MPLS/VPNs); it is not designed as a security mechanism.



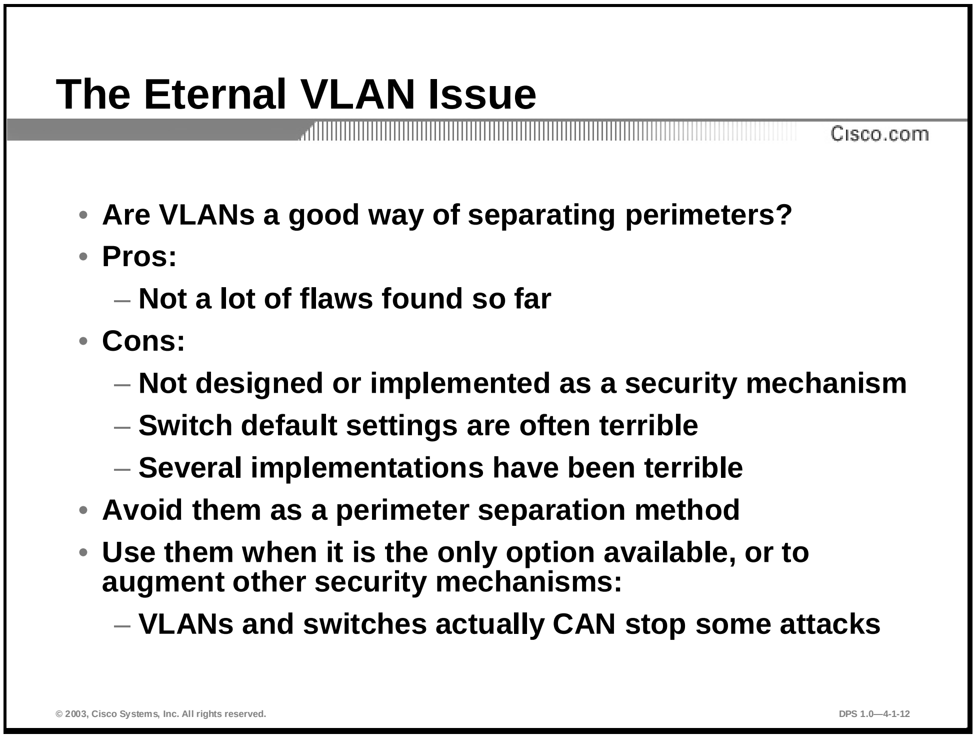
## Compartmentalization (Cont.)

**Sometimes, only logical separation of perimeters is possible (campus data/voice)**

- **Physical separation is always recommended**
- **Try to physically separate at least the most sensitive perimeters (“inside”)**

DPS 1.0—4-1-11

This figure shows a logical separation of the voice and data network using LAN switch VLAN functionality. In high-security designs, always physically separate perimeters—logical separation techniques are otherwise considered the weakest link in the firewall’s security. If total physical separation is not possible, physically separate at least the most sensitive (that is, inside) perimeters.



## The Eternal VLAN Issue

Cisco.com

- Are VLANs a good way of separating perimeters?
- Pros:
  - Not a lot of flaws found so far
- Cons:
  - Not designed or implemented as a security mechanism
  - Switch default settings are often terrible
  - Several implementations have been terrible
- Avoid them as a perimeter separation method
- Use them when it is the only option available, or to augment other security mechanisms:
  - VLANs and switches actually CAN stop some attacks

© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—4-1-12

Security of switch VLANs was, and still is, a constant topic in terms of its robustness for perimeter separation. The arguments for and against using them are as follows.

## Pros

There have not been many flaws found so far.

## Cons

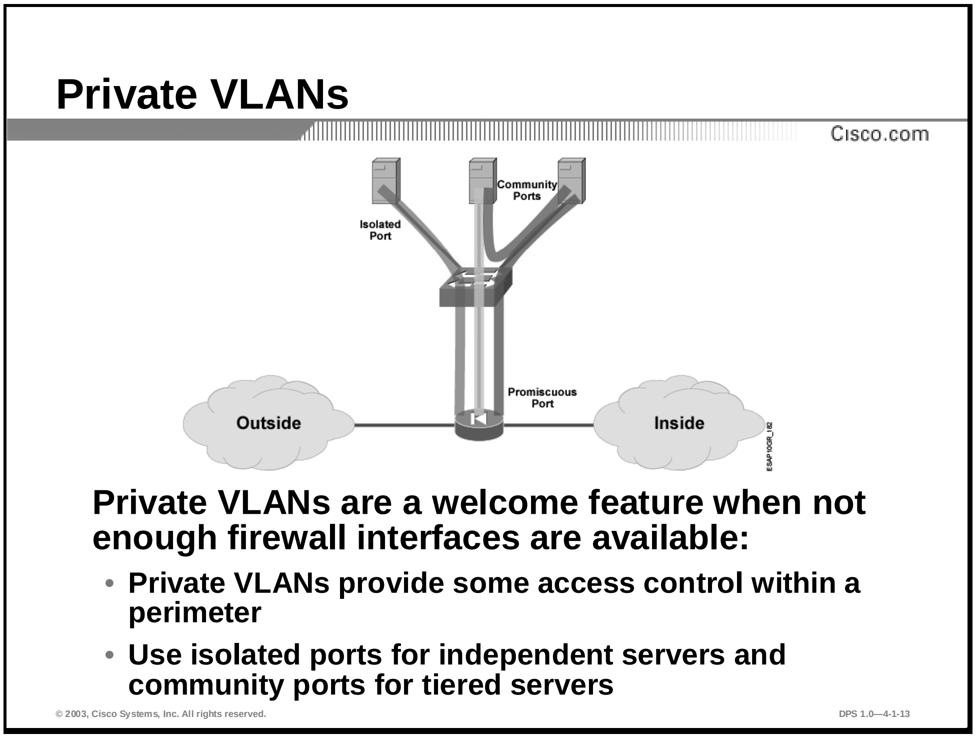
- **VLANs were never designed and implemented as a security feature:** VLANs reduce broadcast domains, and a flaw in the code can enable an attacker to hop between VLANs. A lot of L2 switches fail-open to insecure (all ports in the same VLAN) setting when the configuration is lost due to, for example, a power surge.

- **Some terrible default settings:** These allowed VLAN hopping such as the setting of the trunk port native VLAN to the default VLAN, enabling users of the default VLAN to access any other VLANs.

- **Some broken implementations:** These only limited broadcasts between VLANs, while unicast Layer 2 (L2) packets could hop between VLANs if the attacker knew or guessed the MAC address of the target in another VLAN.

In general, a security designer should not rely on a performance-enhancing method to separate perimeters. If used, the enforcement of access control would rely on two systems: the switch with VLANs, and the firewall filtering between VLANs. If either of those systems failed in terms of security, an attacker could violate access control restrictions. The rationale is that the VLAN switch would probably be more likely to fail, and should be avoided.

Therefore, generally avoid VLANs, especially in high-end solutions, and use them when they are the only option available because of cost restrictions. Additionally, use them to augment other mechanisms—for example, deploy the private VLAN functionality inside a demilitarized zone (DMZ) to provide isolation between hosts that do not need to talk to each other. In this manner, they act as an intra-perimeter access control method, which could not be implemented using other network devices.



## Private VLANs

**Private VLANs are a welcome feature when not enough firewall interfaces are available:**

- **Private VLANs provide some access control within a perimeter**
- **Use isolated ports for independent servers and community ports for tiered servers**

DPS 1.0—4-1-13

When a designer is forced to place multiple systems on the same perimeter, but those systems do not need to communicate, private VLANs are a welcomed feature. This usually occurs when there are not enough firewall interfaces available, and the designer creates a perimeter, in which he places a server of approximately the same trust level.

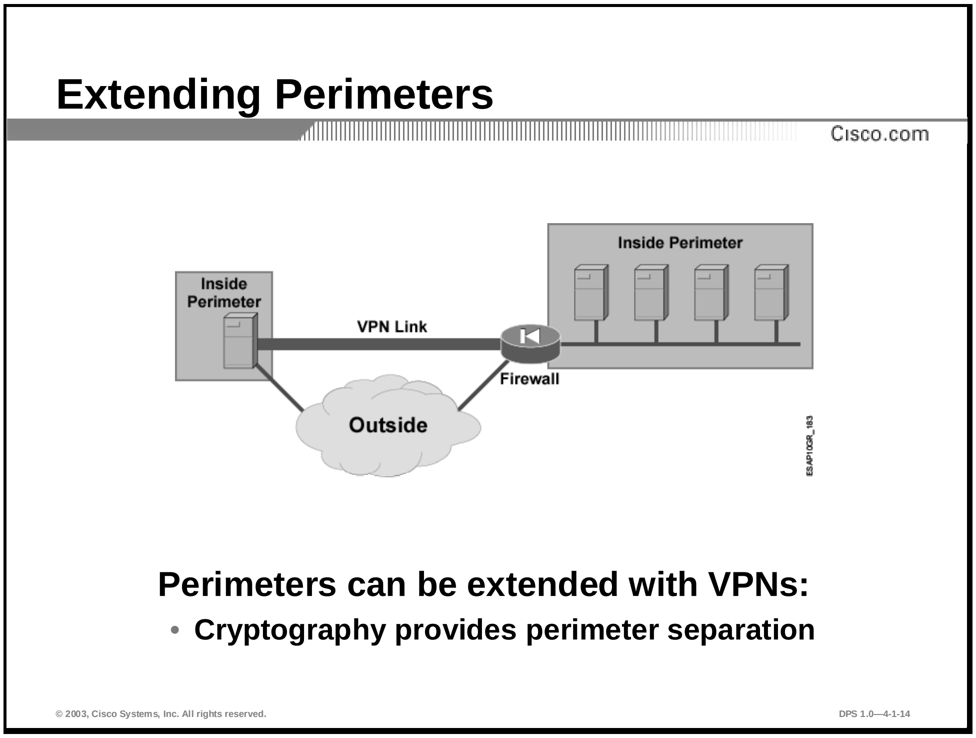
## Example

In an Internet firewall, the devices, for example, e-commerce servers, Domain Name System (DNS) servers, and Virtual Private Network (VPN), have taken all the perimeters, except for one. Therefore, a designer might put the mail relay and the public WWW server of the enterprise on the same perimeter, as it is the only one available.To provide access control within a perimeter, deploy private VLANs to isolate servers on the same perimeter, and only allow the servers to talk to the firewall. Private VLANs usually support three port flavors:

- **Promiscuous ports can talk to any other ports in the same VLAN:** The firewall usually only connects to the promiscuous port.

- **Isolated ports can only talk to the promiscuous port and no other ports:** Use this flavor for independent servers, which do not need to communicate with other servers on the same segment. If an attacker compromises such a server, he/she can only proceed in the direction of the firewall, and cannot attack collocated servers on the same segment.

- **Community ports can talk to the promiscuous port(s) and to other community ports:** Use this flavor for servers of tiered applications, where a group of servers can be isolated from other servers on the same segment.



## Extending Perimeters

**Perimeters can be extended with VPNs:**

- **Cryptography provides perimeter separation**

DPS 1.0—4-1-14

Connecting two disjointed parts of a perimeter with a secure link can also extend perimeters. A good example is a VPN, where a remote branch office can be connected to the firewall using a VPN link, and the firewall treats both the inside network and the branch office as a single perimeter. In this case, cryptography provides the logical separation of perimeter traffic over the untrusted network.

## Practice

Q1) Which of the following is a good solution for separating hosts in the same DMZ on an Internet firewall?

A) multiple perimeters

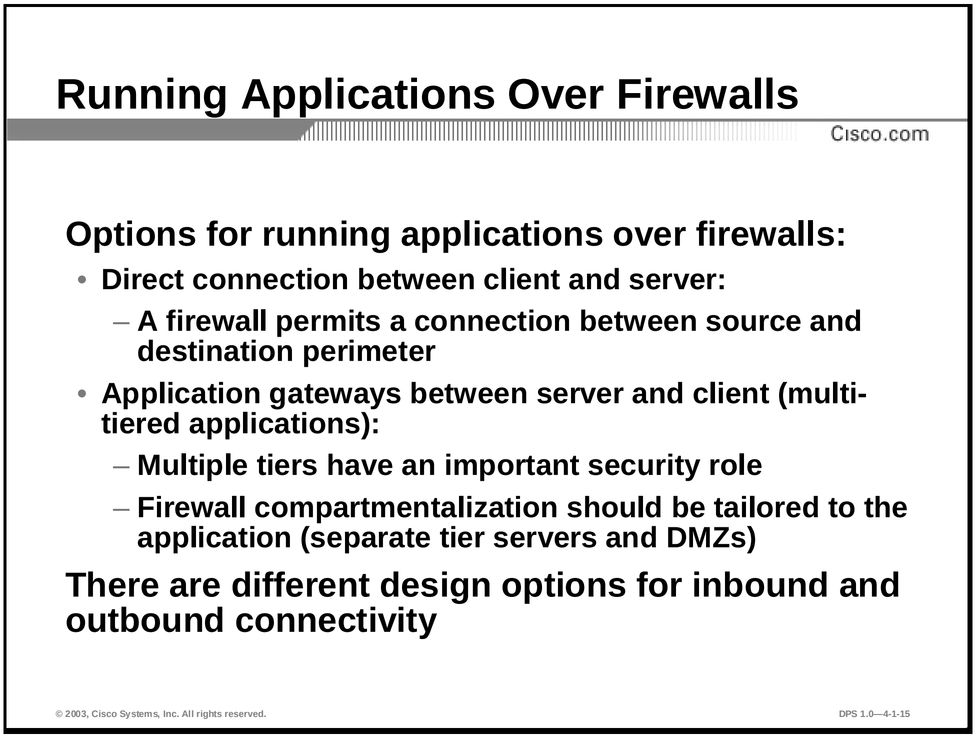
B) private VLANs

C) classic VLANs

D) IPSec tunnels

E) personal firewalls

# Running Applications Over Firewalls



## Running Applications Over Firewalls

**Options for running applications over firewalls:**

- **Direct connection between client and server:**
  - **A firewall permits a connection between source and destination perimeter**
- **Application gateways between server and client (multi-tiered applications):**
  - **Multiple tiers have an important security role**
  - **Firewall compartmentalization should be tailored to the application (separate tier servers and DMZs)**

**There are different design options for inbound and outbound connectivity**

DPS 1.0—4-1-15

## Objective

This section will enable the learner to identify different methods of conveying arbitrary applications over firewalls and choose them in firewall design.

## Introduction

The designer can pass applications over firewalls in a multitude of ways, depending on the trust in clients and servers, and the nature of the application.

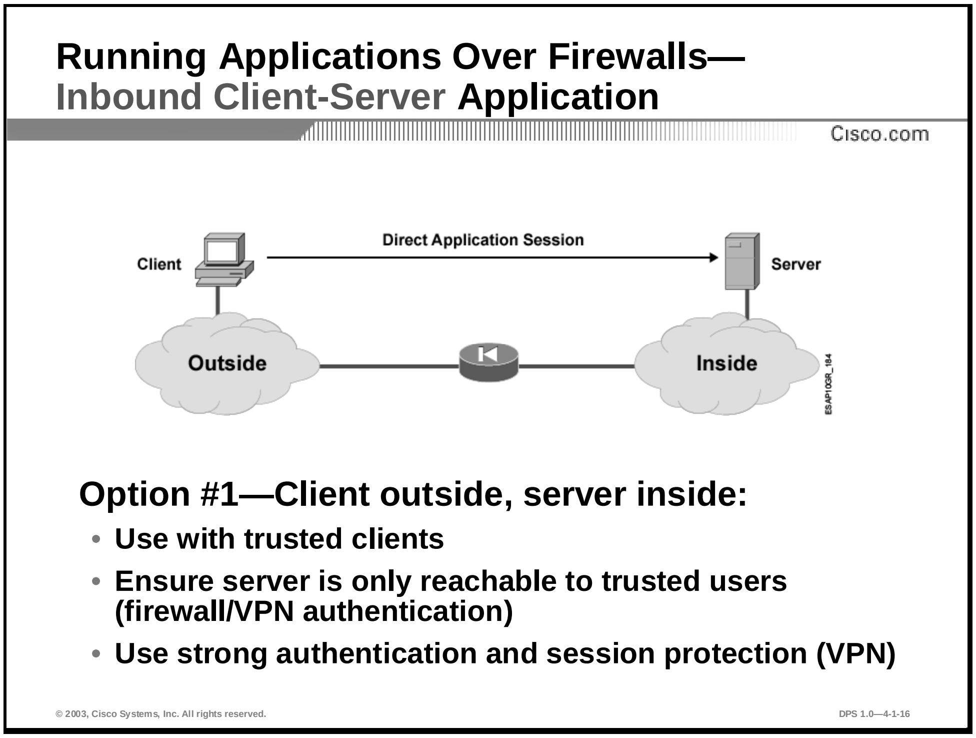
## Application Handling Options

Popular options for passing applications over firewalls include:

- Direct connections between source and destination host over a firewall, where the firewall simply permits the connection between the endpoints and relays the protocol between them.

- Using an application gateway between the client and the server. This application gateway can be built into the application-layer gateway (ALG) firewall, or implemented as a multi-tiered application, where the "server" is split into multiple tiers. The design of a multi-tiered application is usually this way for performance and security reasons. The client can only talk to the exposed first tier, which in turn talks to the next tier, and so forth. Such an application design can significantly increase security, as many mechanisms need to be defeated as a series to compromise the most sensitive data on the innermost tier. A firewall should take this into account and properly separate and protect individual tiers from each other.

Handle connectively separately when considering the choices for running applications over firewalls, inbound (untrusted to trusted) and outbound (trusted to untrusted).



## Running Applications Over Firewalls—Inbound Client-Server Application

Cisco.com

**Option #1—Client outside, server inside:**

- **Use with trusted clients**
- **Ensure server is only reachable to trusted users (firewall/VPN authentication)**
- **Use strong authentication and session protection (VPN)**

© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—4-1-16

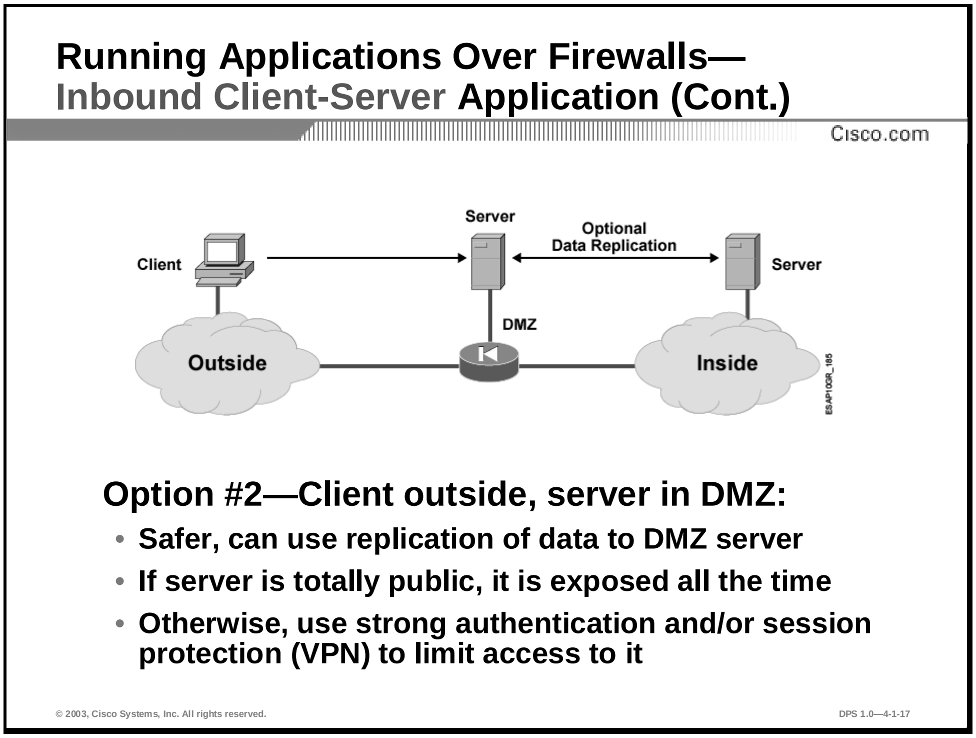
## Direct Inbound Client-Server Connectivity

Direct inbound client-server connectivity is used when directly relaying an application session from an outside client to an inside server. This is usually done when the clients are trusted, therefore no additional protection of the server is needed, and the server is only reachable to trusted clients (that is, not exposed to the untrusted network all the time). Strong firewall authentication that enables outside trusted users to connect to the server, only after they have authenticated to the firewall, achieves this. The server is therefore by default unreachable to all users, and cannot be attacked from the outside.

Care must be taken to provide confidentiality and integrity of the application connection if performing such a communication over an untrusted network. The use of VPNs or application-specific cryptography, for example, SSL, achieves this.

## Example

An organization needs to allow access to mailboxes for roaming users on the Internet. They use Exchange, and need a simple solution to check mail from the Internet anywhere, anytime. On the inside network an Outlook Web Access (OWA) server is set up, which uses Secure HTTP (HTTPS) to access mailboxes through browsers, is simple to pass through firewalls, and provides session protection through Secure Socket Layer (SSL) technology. To access this server directly from the outside, the roaming users first authenticate to the firewall, which then dynamically allows access to the inside server to the authenticated user. Attackers cannot contact the inside server unless they authenticate to the firewall. A one-time password (OTP) system provides robust firewall authentication, which is simple to use and provides a high level of authentication security.



## Running Applications Over Firewalls—Inbound Client-Server Application (Cont.)

Cisco.com

**Option #2—Client outside, server in DMZ:**

- **Safer, can use replication of data to DMZ server**
- **If server is totally public, it is exposed all the time**
- **Otherwise, use strong authentication and/or session protection (VPN) to limit access to it**

DPS 1.0—4-1-17

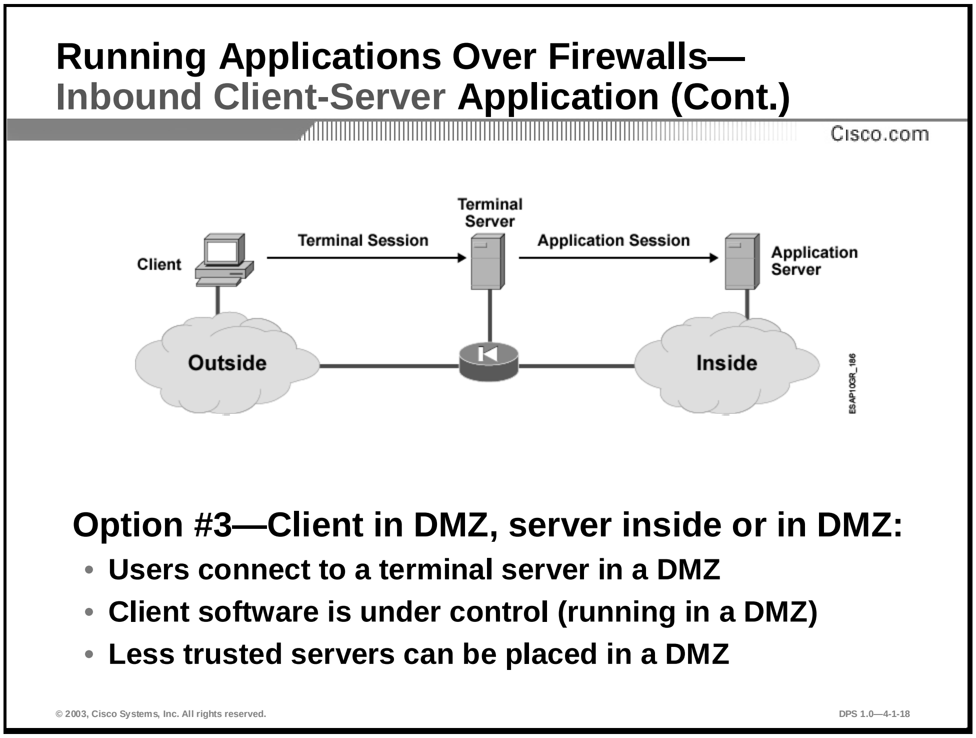
# Inbound Client-Server Connectivity with Server in DMZ

If clients are not trusted, the possibility of server compromise becomes important. The designer needs to ensure the server and its exposed application is appropriately hardened, and to dedicate a DMZ in the firewall to host the exposed server. In the event of server compromise, the attacker is contained within the DMZ and the firewall severely limits his next actions. This design is generally used when hosting public exposed services inside Internet firewalls.

Totally exposed servers, such as the external DNS server, the public mail relay, or the corporate web server, will be exposed to all users all the time. Therefore, use secure server software, and consider the perimeter where the server is located untrusted, as the possibility of compromise is high.

If the server needs to access some inside data, a multi-tiered application design is usually used. Alternatively, if real-time access to inside data is not required, replicate inside data to the exposed server or an additional data server in the same perimeter. This eliminates the need for an inbound connection from the server perimeter to the inside.

If the server contains sensitive data, and the probability of server compromise is high (because, for example, the software used is not very trusted), combine the placement of server in the DMZ with firewall user authentication. This only allows authenticated users access to this server. Use this combination if users are not totally trusted (that is why the server is isolated in the DMZ), but need to view sensitive data. An example would be business partners accessing the exposed server, and firewall authentication preventing arbitrary attackers from connecting to the server at will. An alternative would be to use a VPN terminating at the firewall, authenticating users via VPN methods, and allowing only VPN users to access the more sensitive server.



Running Applications Over Firewalls—
Inbound Client-Server Application (Cont.)

Cisco.com

**Option #3—Client in DMZ, server inside or in DMZ:**

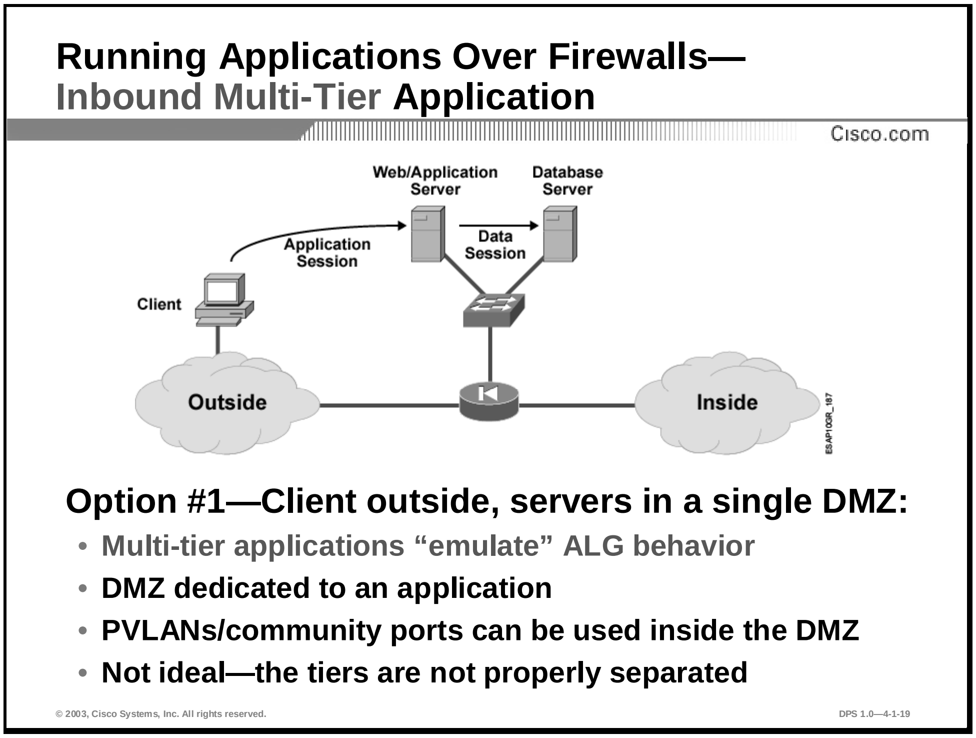
- Users connect to a terminal server in a DMZ
- Client software is under control (running in a DMZ)
- Less trusted servers can be placed in a DMZ

© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—4-1-18

## Inbound Client-Server Connection over a Terminal Server in DMZ

Another option for supporting inbound connections is through a terminal server. Use this option if an organization needs to have full control over client software, allow outside users to connect to a terminal server in a DMZ, and run a preconfigured client on the terminal server, which then connects to the inside network. This is a viable alternative, which simplifies client software deployment, and eliminates potential rogue clients connecting to the sensitive servers. This is conceptually similar to a VPN, except that it uses a terminal session to bring the remote user inside the firewall.

Optionally, if remote users are not very trusted, place the destination server in a DMZ.



## Running Applications Over Firewalls—Inbound Multi-Tier Application

Cisco.com

**Option #1—Client outside, servers in a single DMZ:**

- **Multi-tier applications “emulate” ALG behavior**
- **DMZ dedicated to an application**
- **PVLANs/community ports can be used inside the DMZ**
- **Not ideal—the tiers are not properly separated**

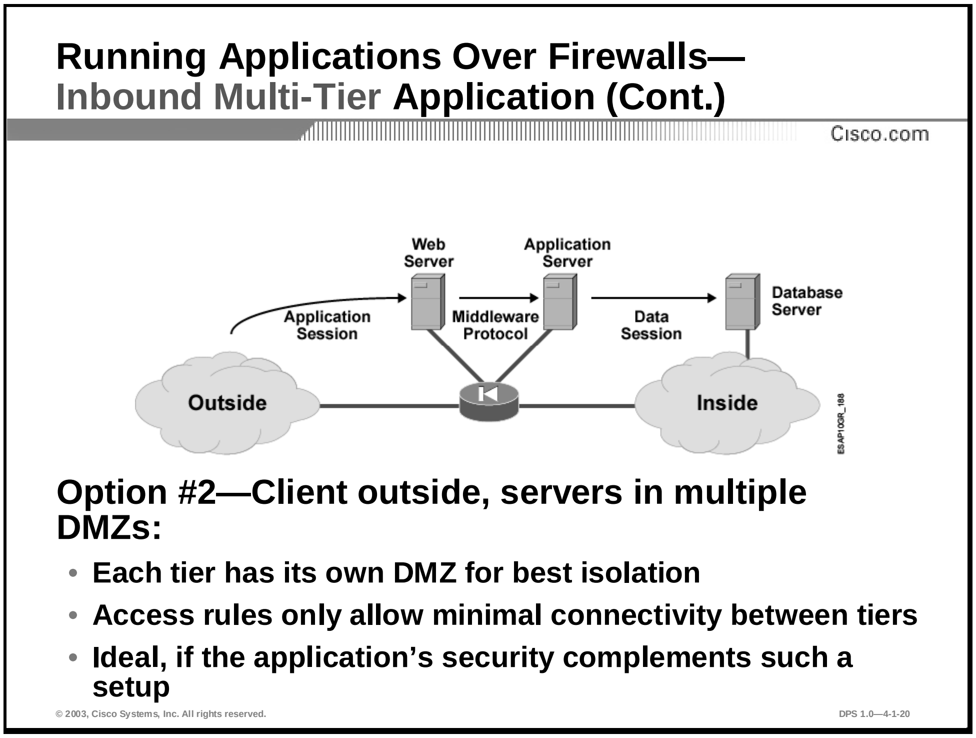
DPS 1.0—4-1-19

## Inbound Multi-Tiered Connection

A multi-tiered application separates its functionality onto multiple systems, which cooperate using proprietary or standard middleware protocols, such as CORBA or DCOM, and/or database protocols. The client connects to an exposed server, which processes the client’s request, translates it into a middleware protocol message or database query, and sends it to the next application tier. If properly designed, this significantly increases the application’s security, as each tier can do its own filtering inside data flow, and the untrusted client can only talk to a single exposed first-tier server. Even if that server is compromised, the attacker needs to compromise other tiers to arrive at the critical data on the inside server, if the application is properly designed.

**Note** This behavior is similar to ALGs, where a server processes a client request and forwards it to another server. Only if multi-tiered applications are purposefully built for a specific application are they called ALGs.

This figure illustrates a DMZ dedicated to a multi-tiered application. All the servers share the same DMZ; therefore the firewall cannot provide access control between the tier servers and they only allow the minimal required connectivity between them (least privilege). Use private VLANs to isolate tiered application servers from other servers on the same segment. However, this setup is not ideal, as it lacks tier separation over the firewall.



**Running Applications Over Firewalls—Inbound Multi-Tier Application (Cont.)**

Cisco.com

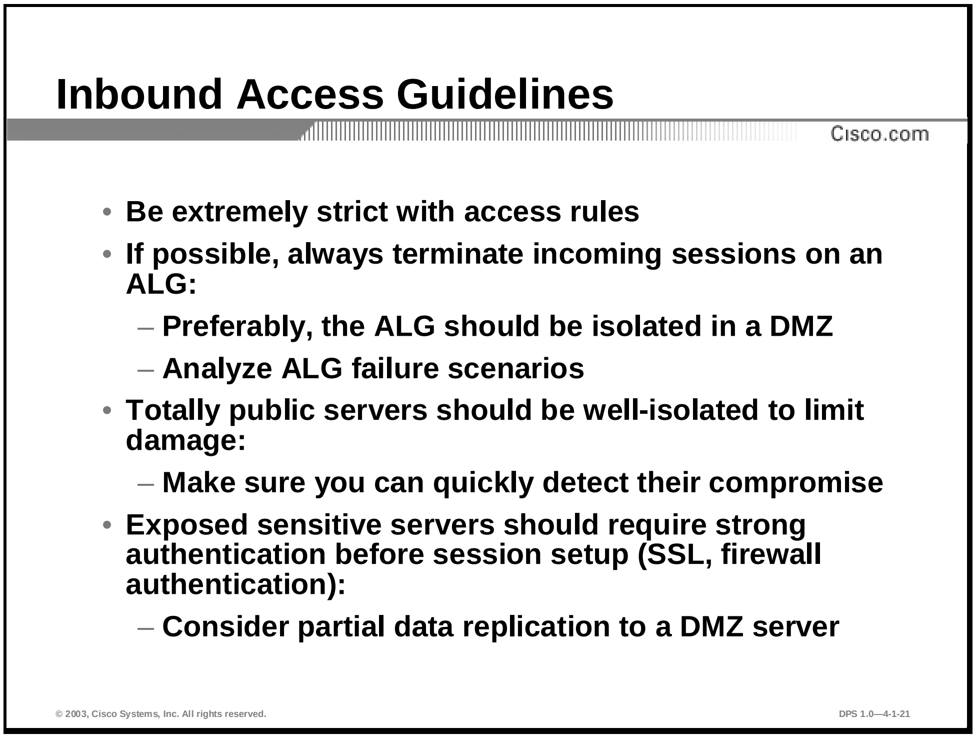
**Option #2—Client outside, servers in multiple DMZs:**

- **Each tier has its own DMZ for best isolation**
- **Access rules only allow minimal connectivity between tiers**
- **Ideal, if the application's security complements such a setup**

DPS 1.0—4-1-20

## Inbound Multi-Tiered Connection with Separate Tier DMZs

Ideally, each tier of a multi-tiered application is hosted inside its own DMZ, providing the most granular access control. This results in proper least-privilege enforcement of access control between the tier DMZ. The designer must therefore develop the application with security in mind, so it does not allow an attacker, who has broken into the first tier server, to immediately send requests to the innermost tier, with the highest possible application privileges.



### Inbound Access Guidelines

- **Be extremely strict with access rules**
- **If possible, always terminate incoming sessions on an ALG:**
  - – **Preferably, the ALG should be isolated in a DMZ**
  - – **Analyze ALG failure scenarios**
- **Totally public servers should be well-isolated to limit damage:**
  - – **Make sure you can quickly detect their compromise**
- **Exposed sensitive servers should require strong authentication before session setup (SSL, firewall authentication):**
  - – **Consider partial data replication to a DMZ server**

DPS 1.0—4-1-21

## Guidelines

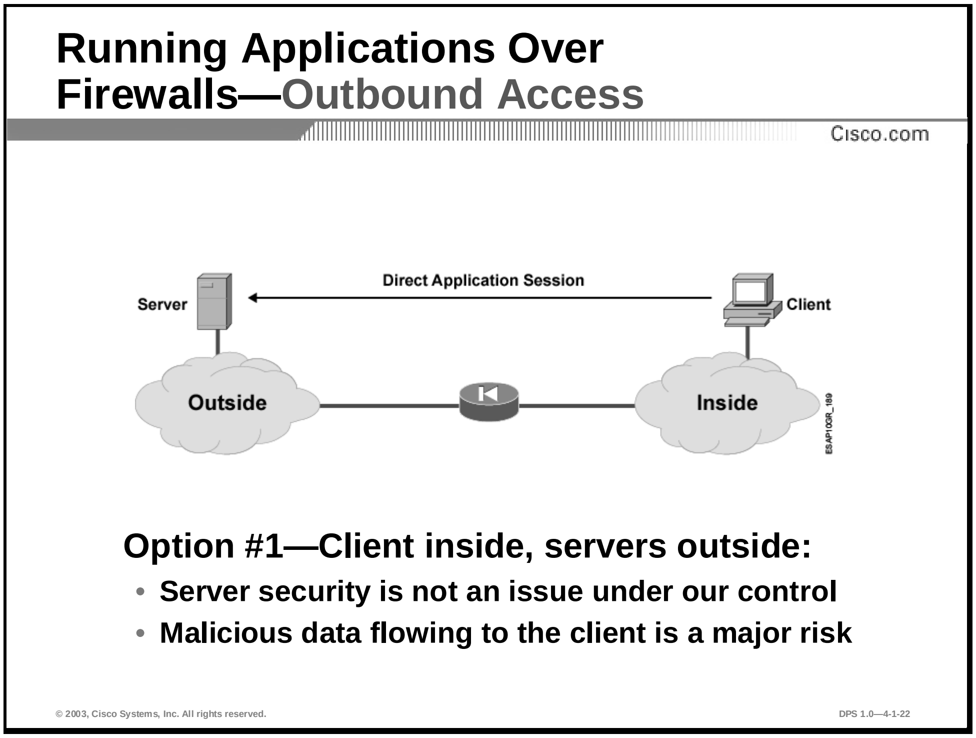
Use least privilege without compromise for inbound access that allows access from less trusted to more trusted perimeters. Deploy extremely strict access rules, allowing minimal necessary connectivity between the perimeters. It is always a good practice to first terminate all incoming sessions from untrusted networks on an ALG (or the first tier of a multi-tiered application) in a firewall DMZ. Always analyze what could happen if an attacker compromises the ALG, that is, what could be the attackers next actions if he gained full access to the ALG host.

## Example

Incoming Simple Mail Transfer Protocol (SMTP) email on the Internet is normally handled using an exposed mail relay (ALG) which filters mail, perhaps performs content scanning, and forwards it to an inside mail server. Some organizations deploy dual-ALG solutions, where the inside mail server is another relay, which in turn talks to the internal mail hub. Such an approach is a good example of defense-in-depth, as someone breaking into the outside mail relay cannot directly attack the inside mail hub.

Totally public servers, which need to be open for access for all untrusted users, should be isolated extremely well to limit damage in the case of a break in. Often called “sacrificial lambs” (as not much damage will be done if they are broken into), these machines are considered untrusted. Ensure good backups of such machines, as well as regular monitoring (Intrusion Detection System [IDS]) to detect an intrusion as quickly as possible.

Exposed sensitive servers should not accept connections from arbitrary clients, as a single application-layer attack could compromise them and violate the confidentiality of data. Authenticate all sessions to such exposed services initially on the firewall, using firewall passthru authentication, or a VPN technology. Also, sensitive data on this server could be replicated from the inside network, eliminating the exposed server’s need to connect to the inside network. If an attacker compromises the server, no connections can be made from that server. Therefore, although the attacker can view local sensitive data on the compromised server he is contained. Such replication can be partial, only transferring data needed by the outside server. This narrows the window of exposure for sensitive data.



## Running Applications Over Firewalls—Outbound Access

Cisco.com

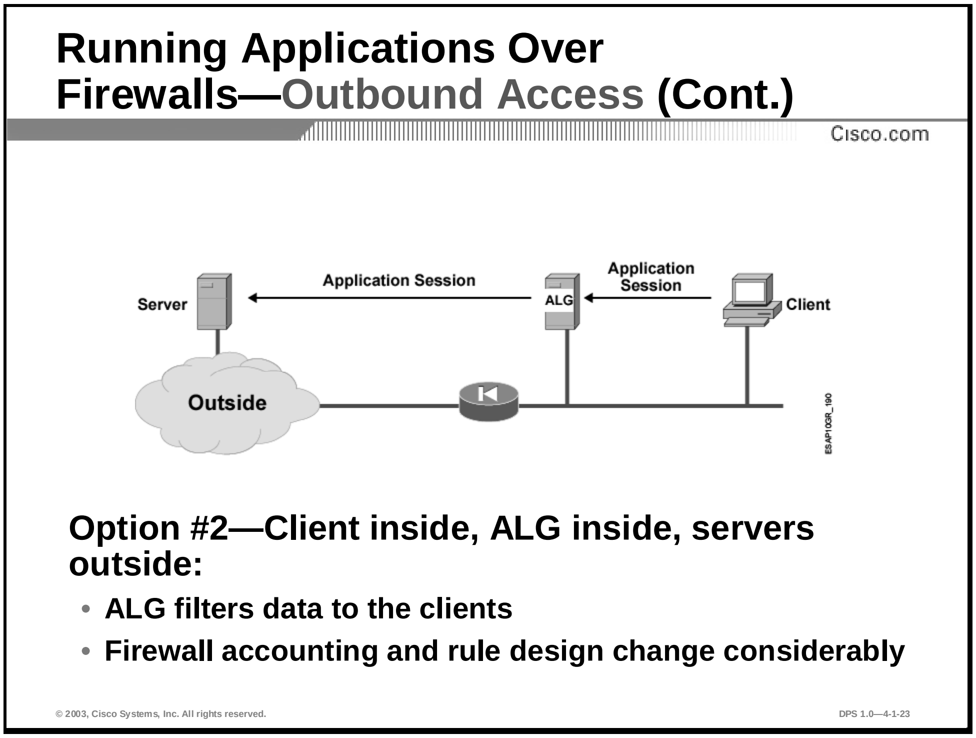
**Option #1—Client inside, servers outside:**

- **Server security is not an issue under our control**
- **Malicious data flowing to the client is a major risk**

DPS 1.0—4-1-22

## Direct Client-Server Outbound Access

The simplest case of outbound access is where the inside client talks directly to an outside server. The outside server is under the control of another party, and might be compromised to send malicious data to the client. Use this method for applications where malicious data is not a major risk, and where performance is of the utmost importance. Examples of this include terminal sessions and multimedia applications such as voice.



## Running Applications Over Firewalls—Outbound Access (Cont.)

Cisco.com

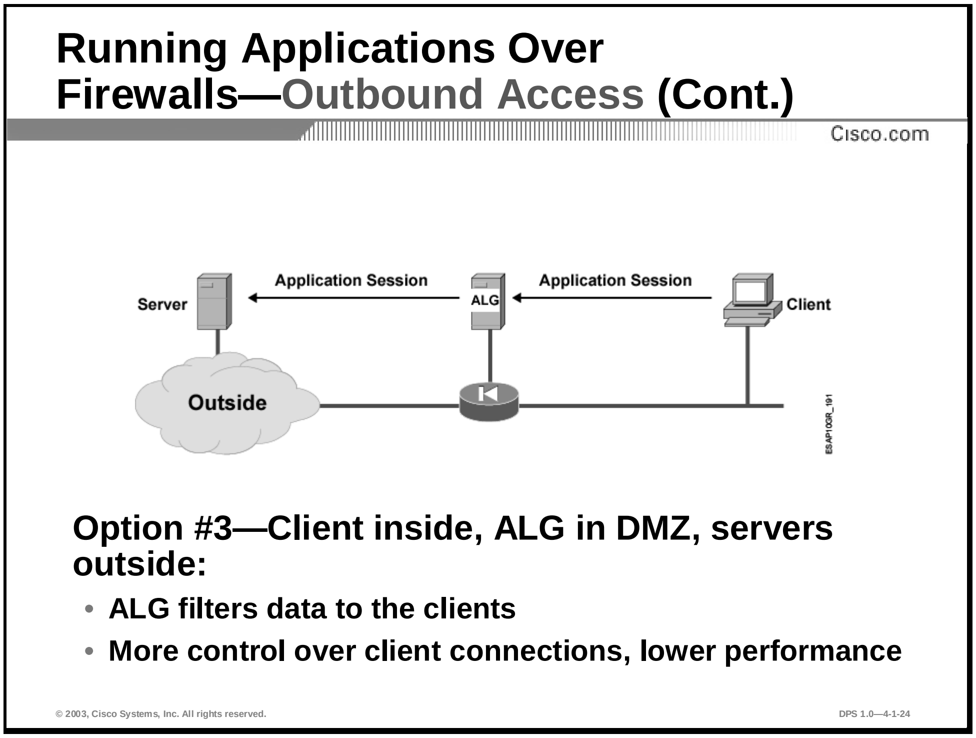
**Option #2—Client inside, ALG inside, servers outside:**

- **ALG filters data to the clients**
- **Firewall accounting and rule design change considerably**

DPS 1.0—4-1-23

## Outbound Access over an ALG on the Inside Network

If the organization requires content, place an ALG behind the firewall for protection, and forward all clients requests to the outside network. Most ALGs use their own IP addresses to initiate sessions, therefore all the inside clients appear to the firewall to be coming from a single IP address of the ALG. This can prevent per-user accounting, and does not allow the firewall filter to perform access control based on the inside IP addresses. The firewall filter (a SPF in the figure) only permits traffic from the ALG, which performs all access control for the application protocol.



## Running Applications Over Firewalls—Outbound Access (Cont.)

Cisco.com

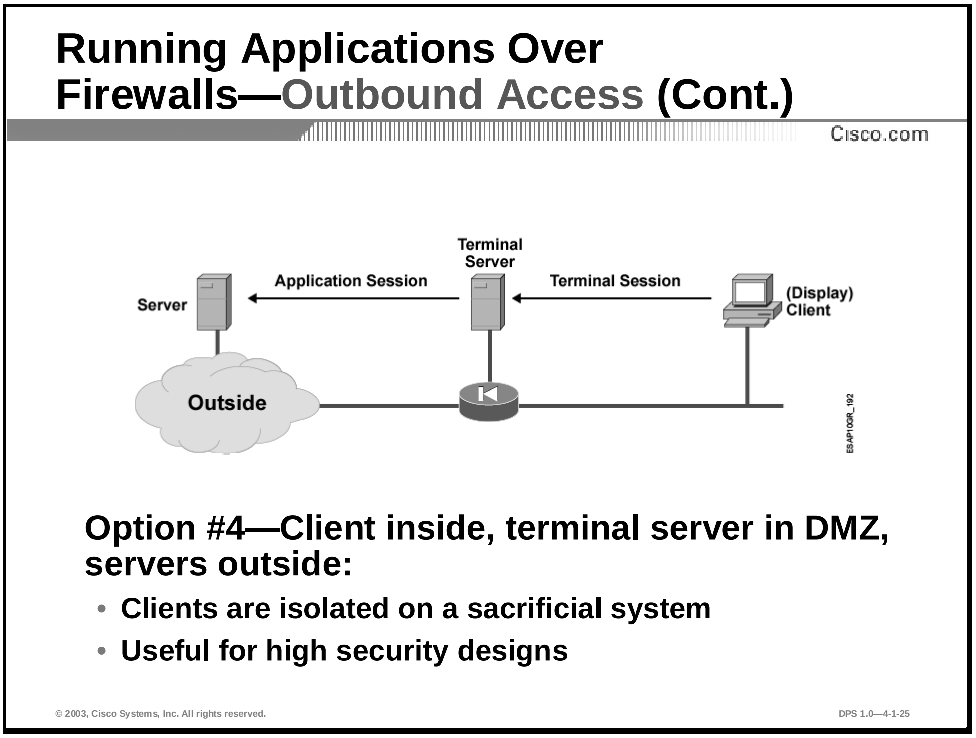
**Option #3—Client inside, ALG in DMZ, servers outside:**

- **ALG filters data to the clients**
- **More control over client connections, lower performance**

DPS 1.0—4-1-24

## Outbound Access over an ALG on a DMZ Network

Another option is to place the ALG for outbound access on a DMZ network. The ALG again performs content filtering for inside clients, and the firewall can now enforce a client IP address-based policy, as connections between clients and the ALG pass over the firewall. Firewall logging is now performed using the real client’s IP address. However, performance is lowered, as all data passes through the firewall twice—first from the outside server to the ALG, and then from the ALG to the inside client.



## Running Applications Over Firewalls—Outbound Access (Cont.)

**Option #4—Client inside, terminal server in DMZ, servers outside:**

- **Clients are isolated on a sacrificial system**
- **Useful for high security designs**

DPS 1.0—4-1-25

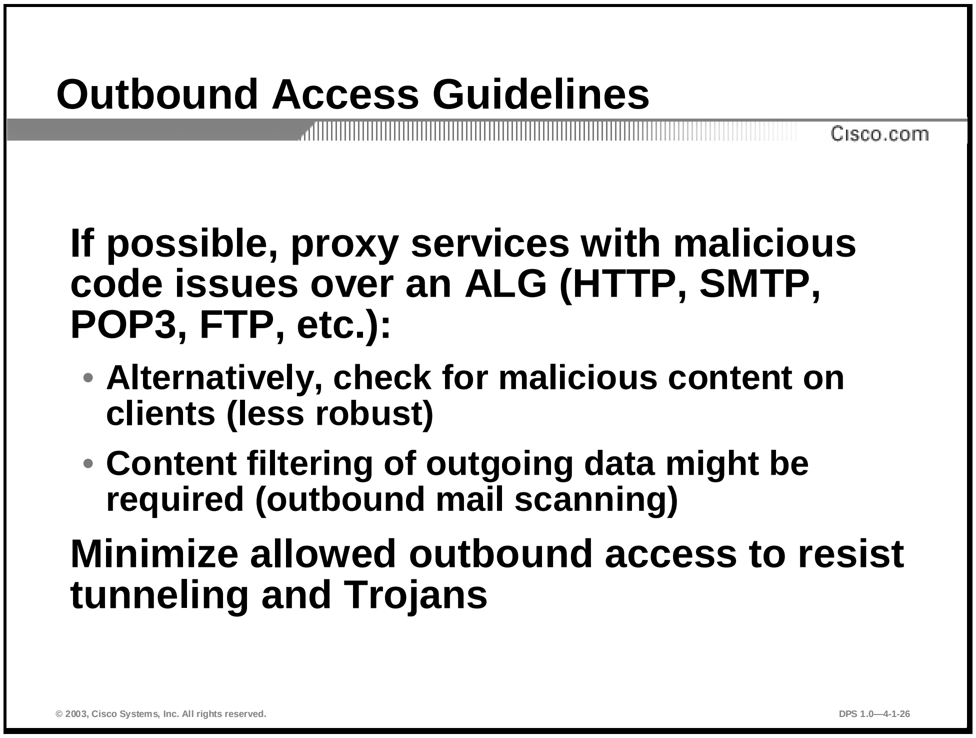
## Outbound Access over a Terminal Server on a DMZ Network

An extreme case of client protection is running the client software not on the end-user workstation, but on a sacrificial host. A terminal server can be set up inside the firewall, and outbound connectivity is achieved by first starting a display session to the terminal server. If the client is compromised, the compromise is limited to the terminal server, and the end-user's workstation is unharmed.

The downside of this approach is that it does not allow the transfer of data from the outside network to the end-station. The only thing an end-user can see is the graphical user interface of the client application over the network.

## Example

Set up a Windows Terminal server inside the firewall if an organization does not trust its web clients. Inside users then connect to the terminal server using a display session, and start the web browser on the terminal server. If the web browser is compromised, for example, via malicious code, the compromise is limited to the client's session on the terminal server, and the end-user's workstation is unharmed.



## Outbound Access Guidelines

If possible, proxy services with malicious code issues over an ALG (HTTP, SMTP, POP3, FTP, etc.):

- Alternatively, check for malicious content on clients (less robust)
- Content filtering of outgoing data might be required (outbound mail scanning)

Minimize allowed outbound access to resist tunneling and Trojans

DPS 1.0—4-1-26

## Guidelines

For outbound, which allows access from more to less trusted perimeters, the major issue driving the deployment of ALGs is malicious code and content. Such applications, which receive data from the outside network, should be proxied over an ALG, which either scans or unconditionally strips suspicious-looking data. Alternatively, deploy content scanning on the clients, however it is harder to manage, and generally less robust compared to specialized gateway software.

Some organizations require content filtering of outgoing data to minimize the risk of sensitive data leaking to less trusted perimeters. A good example is mail gateways, which scan outgoing mail for keywords, indicating sensitive data.

Also, minimize outbound access to only support necessary services. Failure to do so might enable a wider range of “Trojan horses” to connect outside, and provide more opportunities for tunneling over allowed applications.

## Practice

Q1) If an application is separated in many tiers, what does the firewall designer need to do to achieve optimal security?

A) put each tier on a separate perimeter

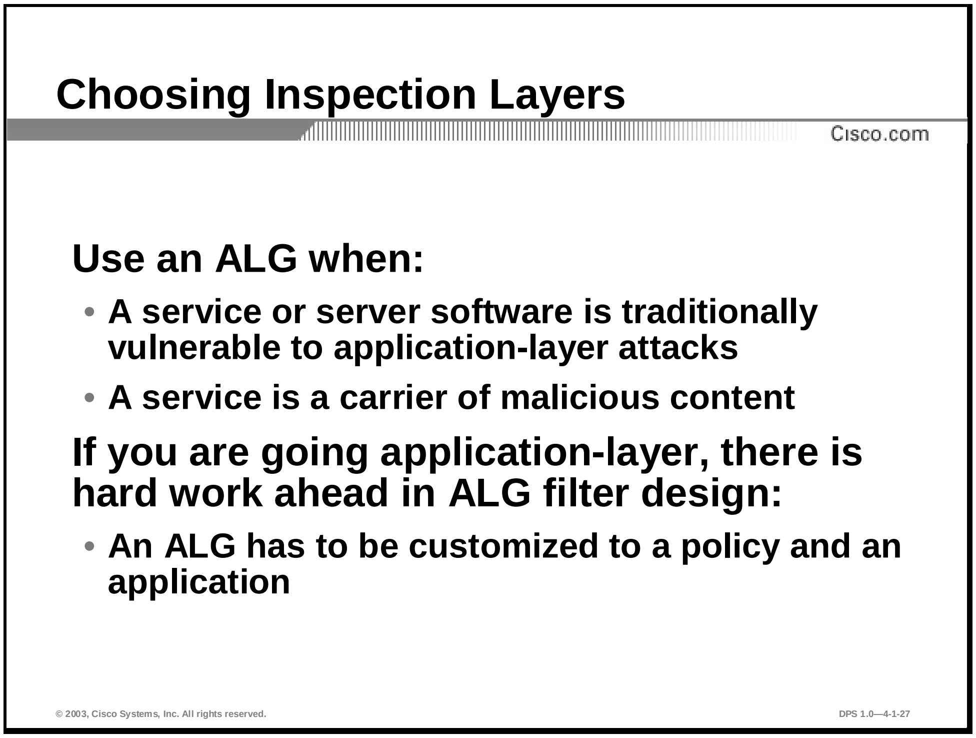
B) put all tiers on the same perimeter, with private VLANs on the switch

C) put all tiers on the same perimeter to achieve the best performance

D) put all but the innermost tier on the least trusted perimeter

E) minimize outbound access to best protect all tiers

# Choosing Inspection Layers



## Choosing Inspection Layers

Use an ALG when:

- A service or server software is traditionally vulnerable to application-layer attacks
- A service is a carrier of malicious content

If you are going application-layer, there is hard work ahead in ALG filter design:

- An ALG has to be customized to a policy and an application

DPS 1.0—4-1-27

## Objective

This section will enable the learner to choose where different layers of inspection should be designed into a firewall system.

## Introduction

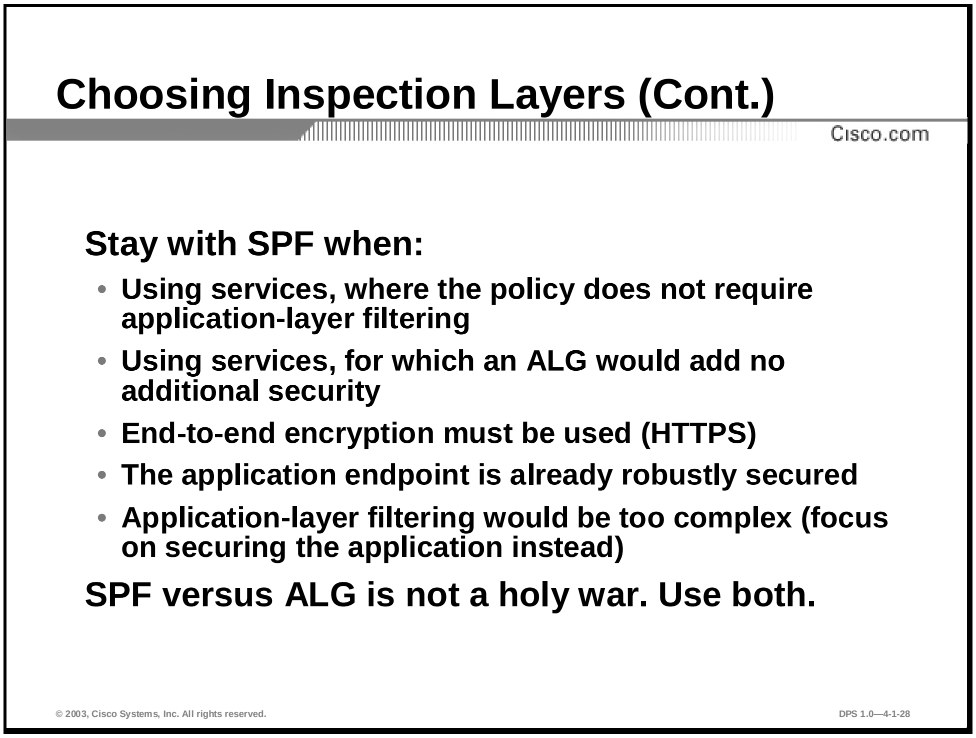
Stateful packet filtering (SPF) or ALGs relay many of the applications over firewall systems. Usually, the technology to use for a particular application is a policy decision.

In general, ALGs are generally used for applications where:

- The server software for incoming connections is, traditionally, vulnerable to application-layer attacks. Place an ALG in front of the application server to filter application data going to the server, so if the ALG is compromised, it does not immediately endanger the server it is protecting.

- Content filtering is desired because a service is a known carrier of malicious content. Of all firewall technologies, only ALGs can perform reliable content filtering.

ALGs are only useful if they provide security services more advanced than those of other technologies, such as SPF. If using an ALG to provide additional application-layer filtering, implement the filtering rules according to the policy. However, this may be hard or even impossible to implement because of policy complexity. In such cases, many organizations abandon this idea and use an ALG only as a proxy without any filtering, which brings no significant value compared to using an SPF.



## Choosing Inspection Layers (Cont.)

**Stay with SPF when:**

- **Using services, where the policy does not require application-layer filtering**
- **Using services, for which an ALG would add no additional security**
- **End-to-end encryption must be used (HTTPS)**
- **The application endpoint is already robustly secured**
- **Application-layer filtering would be too complex (focus on securing the application instead)**

**SPF versus ALG is not a holy war. Use both.**

DPS 1.0—4-1-28

Use stateful filtering in the following application scenarios:

- When the application does not require any application-layer filtering according to the policy, for example, internal voice
- When an ALG cannot provide any additional security to an application, for example, applications for which no proxy exists
- When end-to-end encryption, such as HTTPS, is required
- When the application endpoint is already secured, for example, when every web browser has a locked extremely restrictive content policy, with virus scanning of all content
- When application-layer filtering would be too complex to deploy, for example, for a multi-tier application using Microsoft Simple Object Access Protocol (SOAP), where writing filters to verify correctness of application transactions would be impossible due to lack of documentation and frequent application code changes

**Note** The ALG versus SPF debate is a traditional holy war among security experts. Both technologies have their place, applications, and limitations. In real life situations, these guidelines provide insight where to use each technology to take advantage of its potential.

## Practice

Q1) When would you need to use an application-layer gateway to inspect traffic over a firewall?

A) when complex content filtering is required

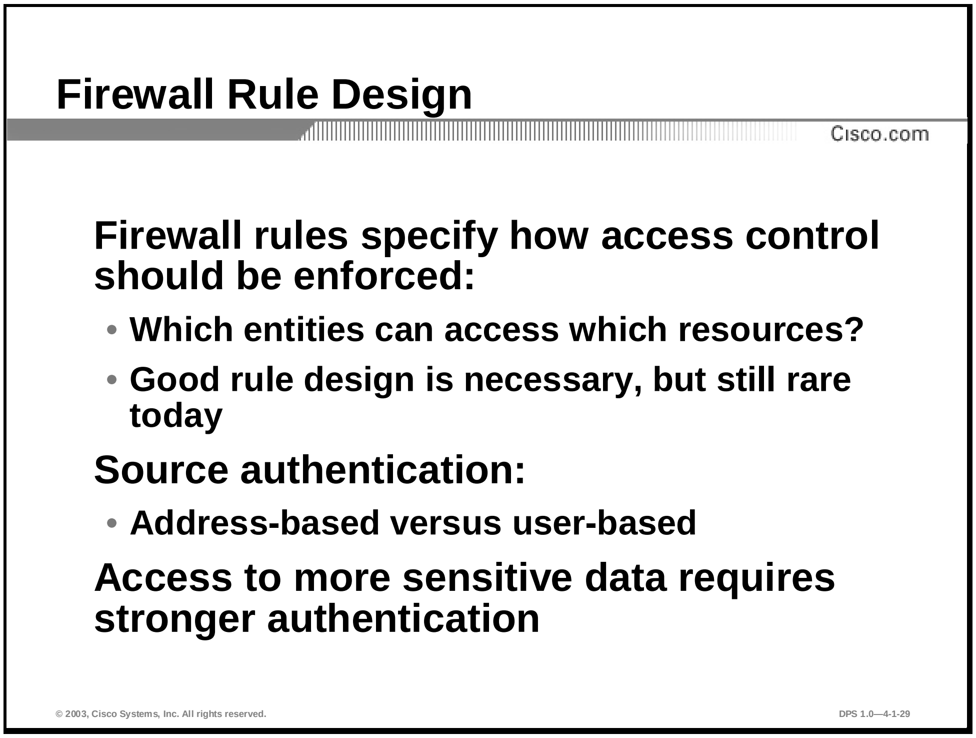
B) to handle multimedia protocols most securely

C) to handle file transfer protocols most securely

D) to provide best protection against flooding attacks

E) when covert channels need to be eliminated

# Firewall Rule Design



## Firewall Rule Design

Firewall rules specify how access control should be enforced:

- Which entities can access which resources?
- Good rule design is necessary, but still rare today

Source authentication:

- Address-based versus user-based

Access to more sensitive data requires stronger authentication

DPS 1.0—4-1-29

## Objective

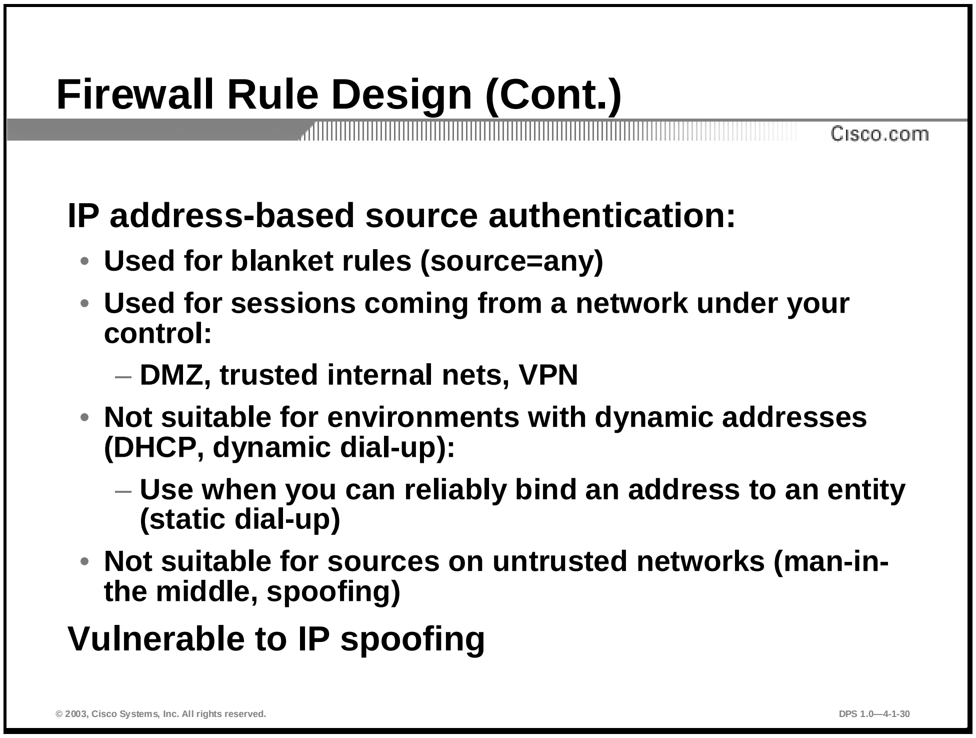
This section will enable the learner to explain the guidelines for source authentication and use the principle of least privilege in rule building and when designing design a firewall system and its rules using it.

## Introduction

Firewall devices use firewall rules to specify how to enforce access control between perimeters. This provides authorization of network connections and enforces which entities can access which resources in what manner. As simple as it sounds, good rule design is rare in modern firewalls, as most operators do not have enough knowledge to define optimal rulesets.

## Source Authentication

To determine which entities can access which resources, entities must be identified and authenticated. Source authentication in firewall rules enables the firewall to reliably identify a subject in a network, and is therefore critically important. There are many methods of authentication, which can be applied in specific scenarios. In general, a good rule of thumb is that access to more sensitive data requires more reliable source authentication, so identity spoofing is unlikely.



## Firewall Rule Design (Cont.)

### IP address-based source authentication:

- Used for blanket rules (source=any)
- Used for sessions coming from a network under your control:
  - DMZ, trusted internal nets, VPN
- Not suitable for environments with dynamic addresses (DHCP, dynamic dial-up):
  - Use when you can reliably bind an address to an entity (static dial-up)
- Not suitable for sources on untrusted networks (man-in-the middle, spoofing)

### Vulnerable to IP spoofing

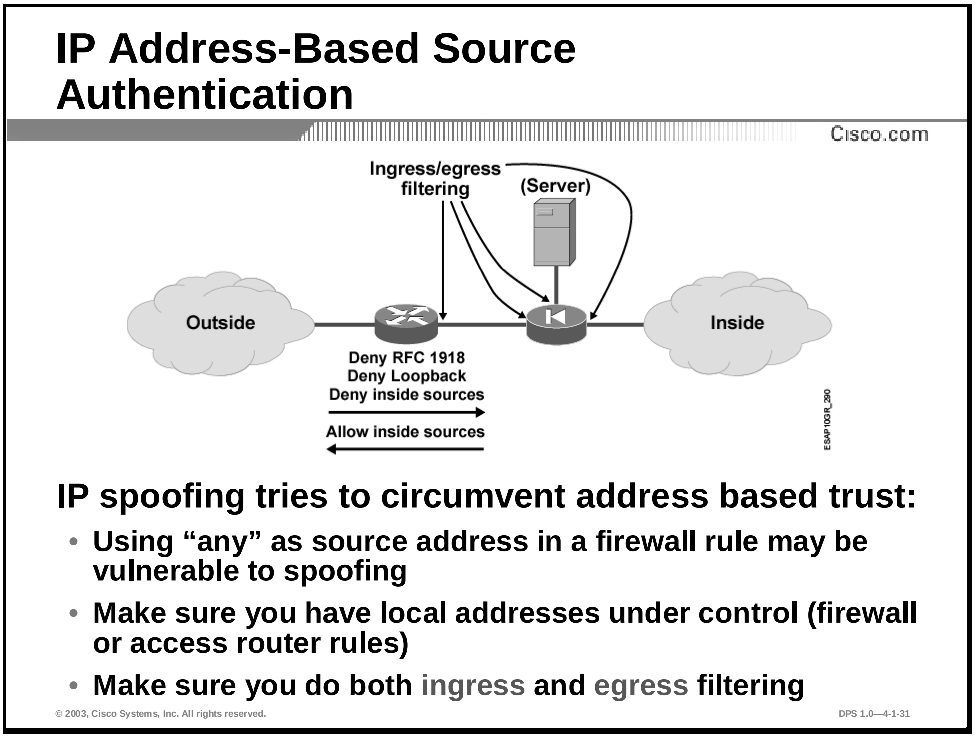
DPS 1.0—4-1-30

The simplest source authentication is based on IP addresses. This is generally used in the following scenarios:

- When blanket rules are used (allow/deny access for everybody, or any IP address)
- When considering sources that are in a network with good control over addressing (inside networks, DMZs, VPNs)

Do not use IP address-based authentication in dynamic addressing environments, such as DHCP or dynamic dial-up environment, or for sources located on or behind untrusted networks. If no protection method, such as a VPN, protects the packets on an untrusted network, forging of the IP addresses or interception of existing connections may occur.

IP spoofing is the most prevalent threat that can defeat IP address-based rules. Therefore, if deploying such rules, perform an analysis of spoofing possibilities, and apply countermeasures.



## IP Address-Based Source Authentication

**IP spoofing tries to circumvent address based trust:**

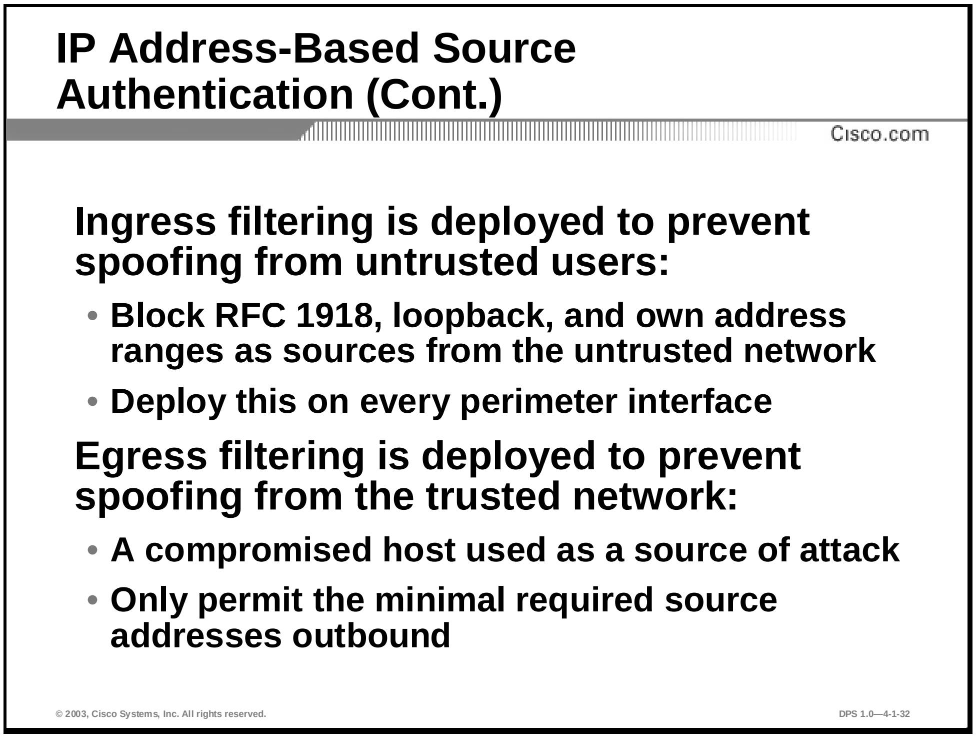
- **Using “any” as source address in a firewall rule may be vulnerable to spoofing**
- **Make sure you have local addresses under control (firewall or access router rules)**
- **Make sure you do both ingress and egress filtering**

DPS 1.0—4-1-31

In the context of a firewall, the firewall device should provide protection against spoofing on its interfaces. The two most common guidelines for deploying anti-spoofing rules are:

- On each perimeter interface, disallow traffic entering the firewall to carry source addresses, which are reachable on another perimeter
- On each perimeter interface, filter out source addresses which should, by definition, not be present on that network, for example, the loopback address—127.0.0.1, RFC 1918 networks on the Internet

Anti-spoofing is usually deployed either using automatic methods, such as Unicast Reverse Path Forwarding, or with manual rule configuration. Sometimes anti-spoofing is not implemented directly on the main access control device, but on another device in the packet path, such as the access router on the Internet connection.



## IP Address-Based Source Authentication (Cont.)

Cisco.com

Ingress filtering is deployed to prevent spoofing from untrusted users:

- Block RFC 1918, loopback, and own address ranges as sources from the untrusted network
- Deploy this on every perimeter interface

Egress filtering is deployed to prevent spoofing from the trusted network:

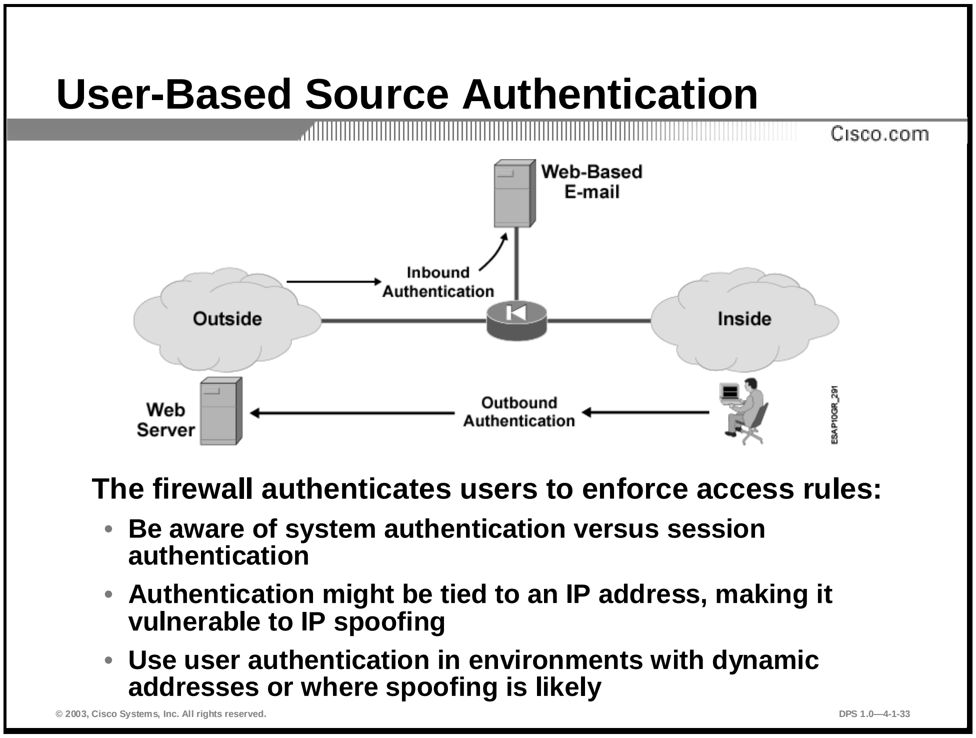
- A compromised host used as a source of attack
- Only permit the minimal required source addresses outbound

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—4-1-32

There are generally two types of anti-spoofing protection: ingress and egress filtering:

- **Ingress filtering:** Filters source addresses when traffic is entering a protected network. Filtering rules should:
  - Block addresses of the protected network to be used as sources of incoming traffic
  - Block other impossible addresses, such as parts of the whole RFC1918 address space, and the loopback network
- **Egress filtering:** Filters source addresses when traffic is exiting a protected network. Filtering rules should only allow traffic sourced in the protected network to exit it, preventing possible compromised inside hosts to use spoofed addresses and attack outside systems.



## User-Based Source Authentication

**The firewall authenticates users to enforce access rules:**

- **Be aware of system authentication versus session authentication**
- **Authentication might be tied to an IP address, making it vulnerable to IP spoofing**
- **Use user authentication in environments with dynamic addresses or where spoofing is likely**

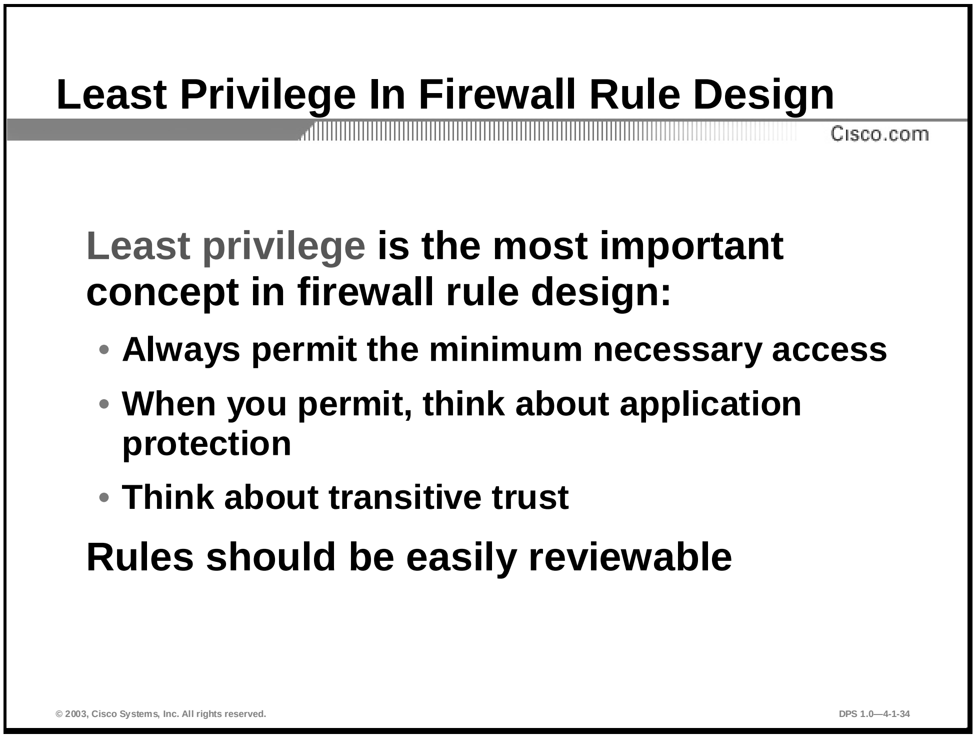
DPS 1.0—4-1-33

As an alternative to IP address-based authentication, the firewall can authenticate users as they attempt to establish sessions over it. Commonly called firewall passthru authentication, it enables the firewall to perform access control based on user identity, which the user authentication confirms.

When authenticating users, a firewall might either “remember” the user as being present on a particular IP address, or require authentication for every application session.

It is important to distinguish between system authentications. That is, associating a user’s identity with a certain IP address, which should only be used if a single user is using an address. This breaks with multi-user systems, where all users on that system are identified as the first users who authenticate to the firewall. Such IP-address-bound user authentication can also be vulnerable to IP spoofing, if an attacker, performing a man-in-the-middle attack, can determine which authenticated users are located at which addresses, and spoofing those addresses to mask as an authenticated user.

User authentication should be used by firewalls in networks where general IP spoofing is very likely, and access rules cannot be based on source addresses only.



## Least Privilege In Firewall Rule Design

**Least privilege is the most important concept in firewall rule design:**

- **Always permit the minimum necessary access**
- **When you permit, think about application protection**
- **Think about transitive trust**

**Rules should be easily reviewable**

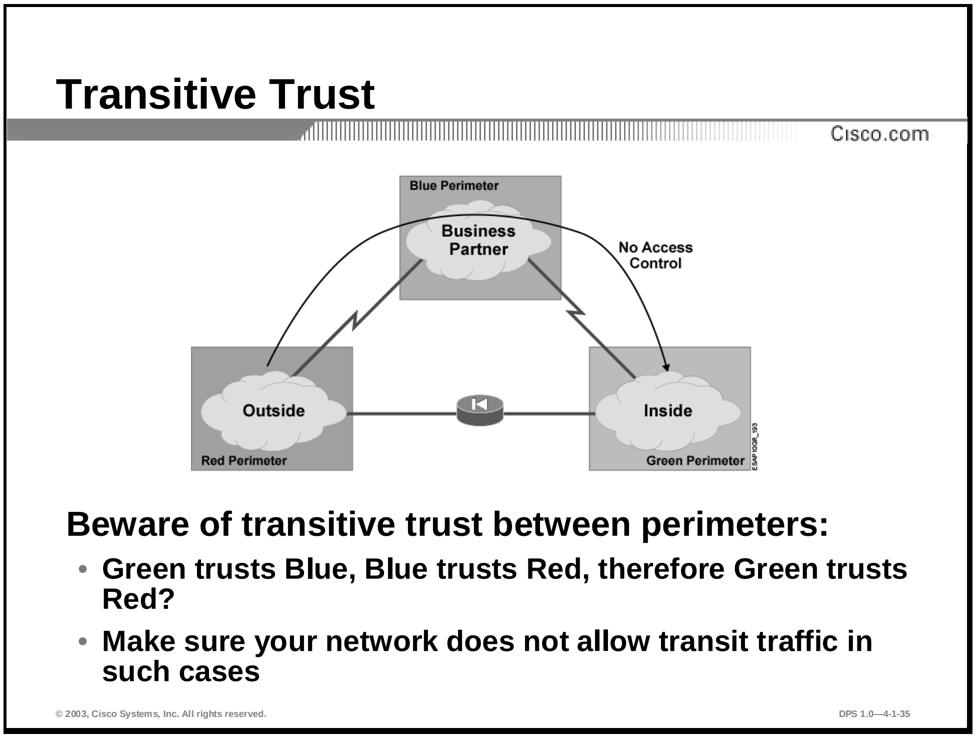
© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—4-1-34

## Least Privilege Concept

The first and foremost principle of good rule design is least privilege. A firewall should only permit the minimum required connectivity between perimeters, and every single rule should exactly implement an access control requirement.

## Example

A bad example of rule design is an administrator who does not know the properties of the Oracle SQL*net protocol. After many unsuccessful attempts to permit the SQL*net protocol between two hosts, he/she opens up all IP connectivity between the hosts to meet a deadline. The rule is forgotten, never optimized, and obviously violates the least privilege principle, enabling an attacker, who compromises one of the hosts, to openly attack the other, which is not protected (all IP is permitted between them).



## Transitive Trust

**Beware of transitive trust between perimeters:**

- **Green trusts Blue, Blue trusts Red, therefore Green trusts Red?**
- **Make sure your network does not allow transit traffic in such cases**

DPS 1.0—4-1-35

Transitive trust is the application of a transitive property to trust relationships. If entity A trusts entity B, and entity B trusts entity C, transitive trust means that entity A trusts entity C. This is usually not desirable, but can often happen if wrong assumptions are made, or if network topologies and access policies are not analyzed in depth.

This figure illustrates two organizations (Green and Blue) connected without a firewall. The Green organization trusts the Blue organization, but does not know that the Blue organization has no access control (firewall) towards the Red organization (for example, the Internet). As the Green organization does not trust the Red organization because there is a firewall between them, this results in an unwanted trust path between the Red and Green organizations, over the Blue organization. In other words, people from the Internet could enter the Green network over the Blue network.

# Practice

Q1) Why is firewall passthru authentication, which binds the user credentials to his/her source IP address, possibly dangerous?

A) because it is vulnerable to IP spoofing attacks

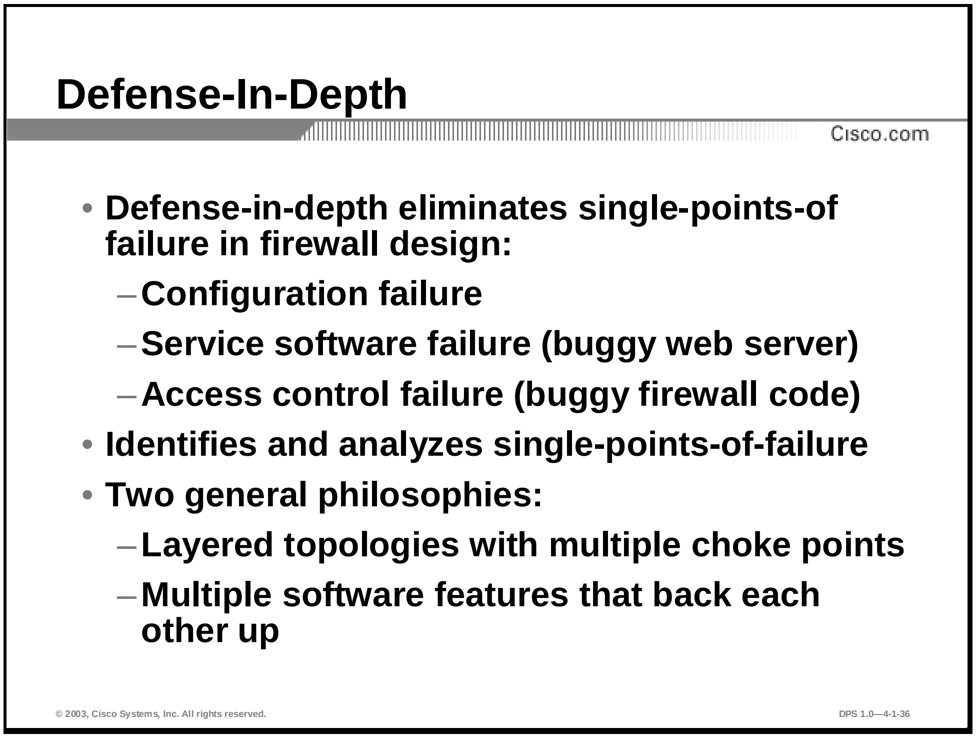
B) because it is vulnerable to application-layer attacks

C) because of transitive trust

D) because the authentication session is not encrypted

E) because of low performance

# Defense-In-Depth



## Defense-In-Depth

Cisco.com

- Defense-in-depth eliminates single-points-of failure in firewall design:
  - Configuration failure
  - Service software failure (buggy web server)
  - Access control failure (buggy firewall code)
- Identifies and analyzes single-points-of-failure
- Two general philosophies:
  - Layered topologies with multiple choke points
  - Multiple software features that back each other up

© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—4-1-36

## Objective

This section will enable the learner to explain the principle of defense in depth and design a firewall system using it.

## Introduction

Many firewalls implement defense-in-depth principles to guard against single-points-of-failure in the following common failure scenarios:

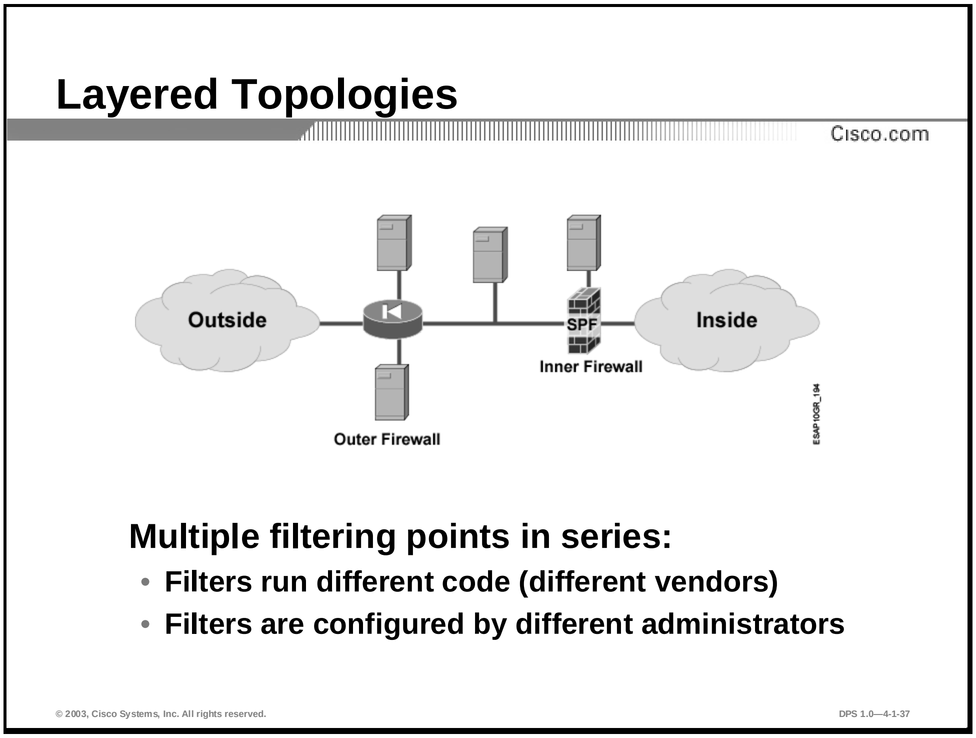
- If an access control device is misconfigured and permits more than it should
- If a certain application service fails (is compromised) because of its bugs or misconfiguration
- If the security devices fail because of bugs in the firewall code

A firewall designer should always identify the single points of failure in a firewall design, using a “what-if” methodology. The designer must correctly identify the consequences of failure for every firewall building block or service, and if the consequences include a major failure of access control, consider a defense-in-depth mechanism.

There are two general approaches to providing defense-in-depth functionality to a network firewall:

- Layered topologies, where multiple choke points are deployed, for example, multiple firewalls in series

- Multiple software features, which back each other up, for example, access lists and policy routing in the same device



## Layered Topologies

Cisco.com

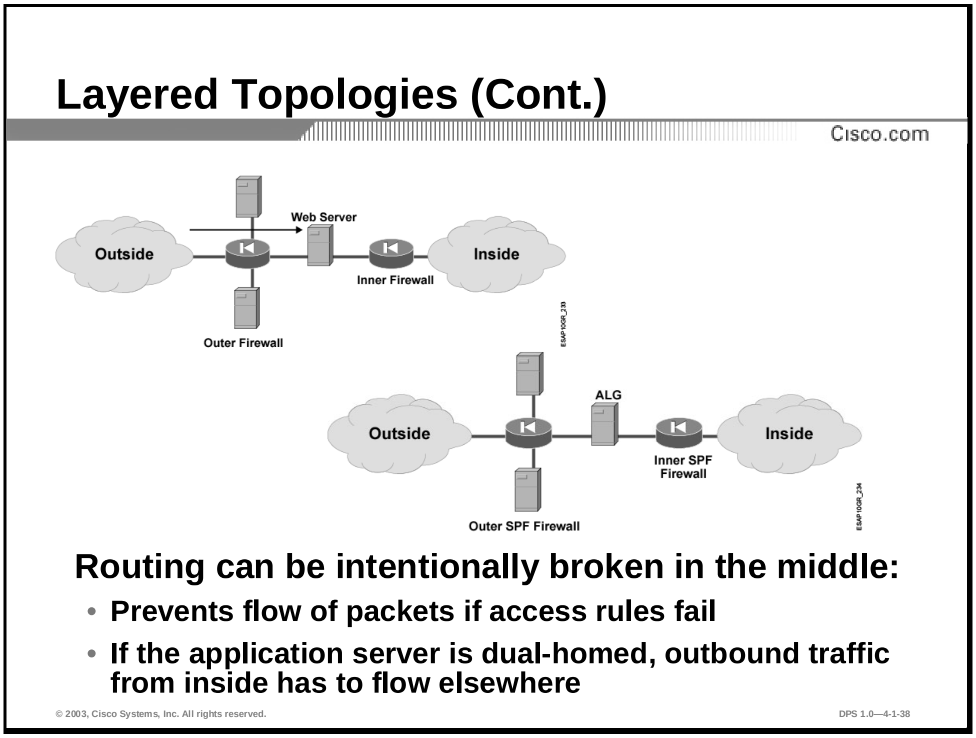
**Multiple filtering points in series:**

- **Filters run different code (different vendors)**
- **Filters are configured by different administrators**

DPS 1.0—4-1-37

# Layered Topologies

A frequently used method for providing defense-in-depth in network firewalls is the deployment of multiple firewalls (choke points) in series. To enter the inside perimeter, traffic needs to pass through two or more filtering systems, which might run different codes from different vendors, to eliminate common failure modes. A separate person, to avoid the same configuration mistake on both systems, should configure each filtering system.



## Layered Topologies (Cont.)

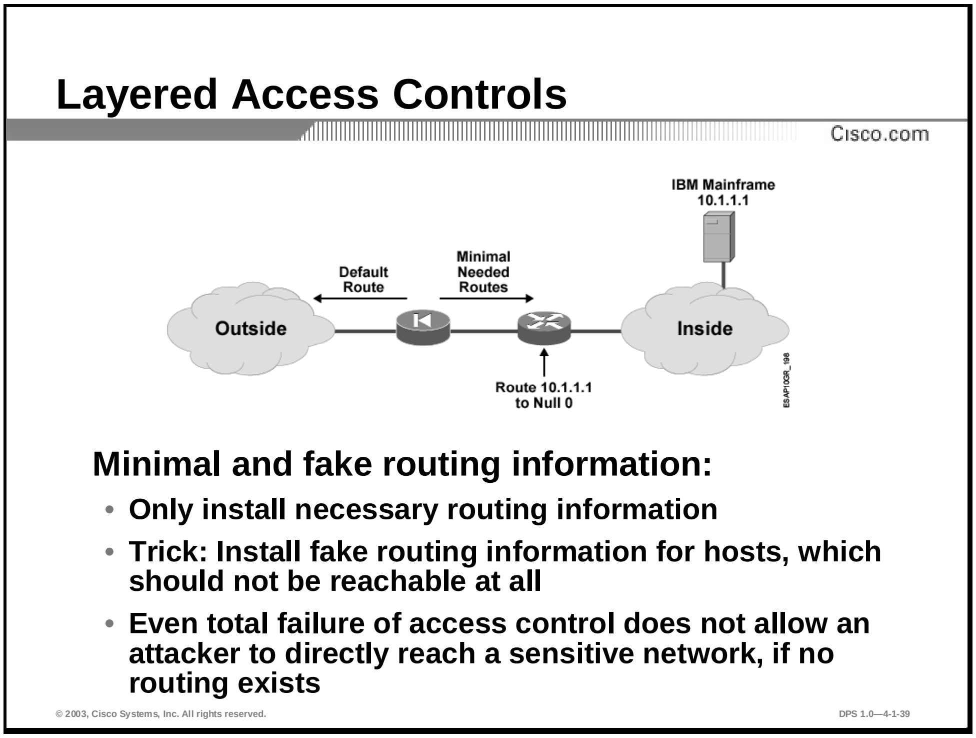
Cisco.com

**Routing can be intentionally broken in the middle:**

- **Prevents flow of packets if access rules fail**
- **If the application server is dual-homed, outbound traffic from inside has to flow elsewhere**

DPS 1.0—4-1-38

A possible evolution of the previous idea is to include another element (such as a host), which intentionally breaks IP routing over a firewall; all traffic has to terminate on a host inside the firewall. To accomplish this install a pure dual-homed host with ALG functionality (for inbound or outbound access), or a dual-homed application server, if only inbound access is desired. Such a setup does not allow traffic to pass between the most and least secure perimeter, even if all access control rules are misconfigured, because packet forwarding is not allowed over the dual-homed system.



## Layered Access Controls

Cisco.com

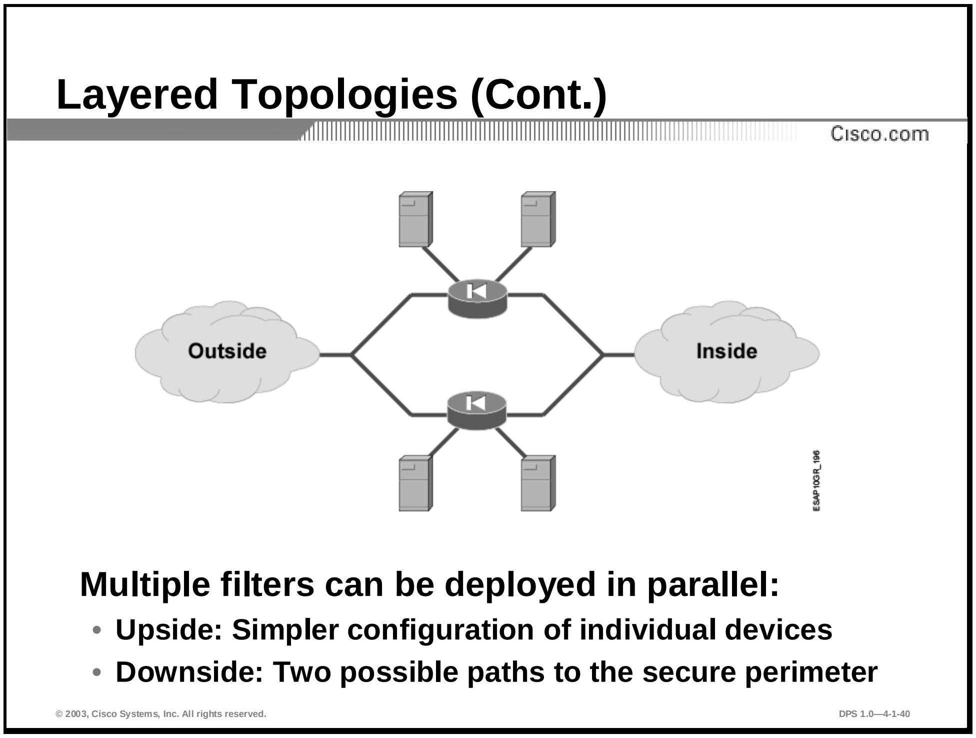
**Minimal and fake routing information:**

- **Only install necessary routing information**
- **Trick: Install fake routing information for hosts, which should not be reachable at all**
- **Even total failure of access control does not allow an attacker to directly reach a sensitive network, if no routing exists**

DPS 1.0—4-1-39

## Layered Access Control

The least privilege principle should also apply to routing inside the firewall. The firewall system should only provide paths to networks, which should be reachable to ensure the minimum required connectivity. This prevents direct connections to sensitive systems, which are not communicating over the firewall and are unreachable, as far as the firewall is concerned. Alternatively, configure devices inside or near the firewall with fake routing information to unnecessary sensitive hosts on networks—this blackholes connections to such destinations.



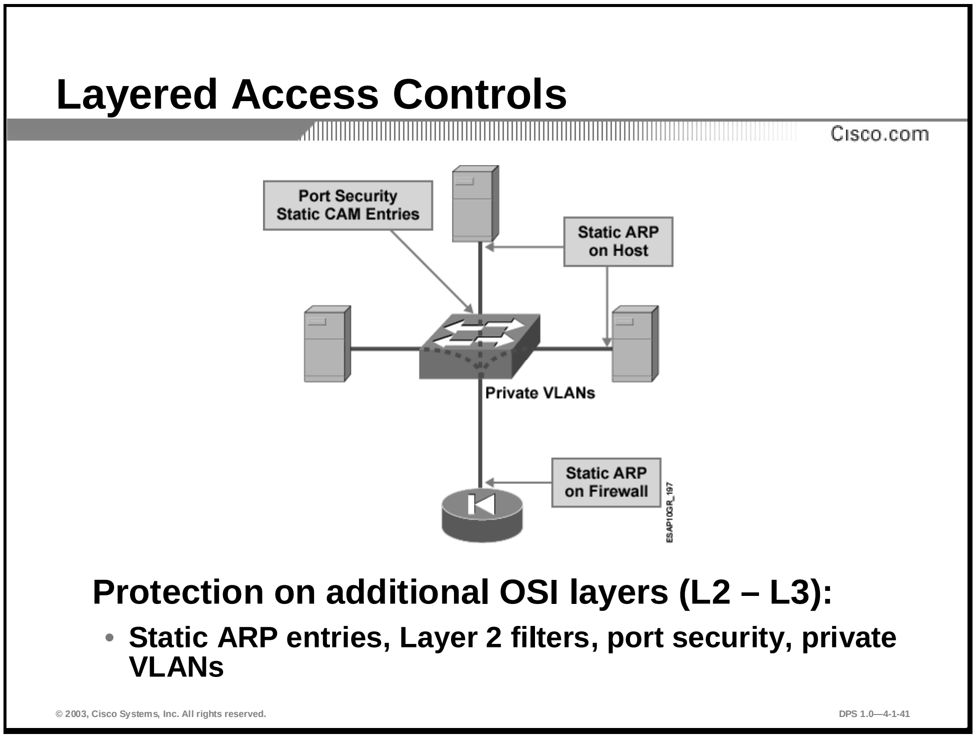
## Layered Topologies (Cont.)

**Multiple filters can be deployed in parallel:**

- **Upside: Simpler configuration of individual devices**
- **Downside: Two possible paths to the secure perimeter**

DPS 1.0—4-1-40

This figure illustrates firewall filters, which can also be deployed in parallel. This can be desired, when two functionally different areas of the firewall need to be separated for management simplicity. For example, a bank might want to separate its residential and corporate Internet banking solutions into two systems, which are managed by different teams. This makes each individual firewall part less complex to configure and maintain, thus increasing its security. On the other side, it provides two possible paths to the internal network, and a critical vulnerability in either of the two parts might allow an attacker to enter more secure perimeters.



## Layered Access Controls

Cisco.com

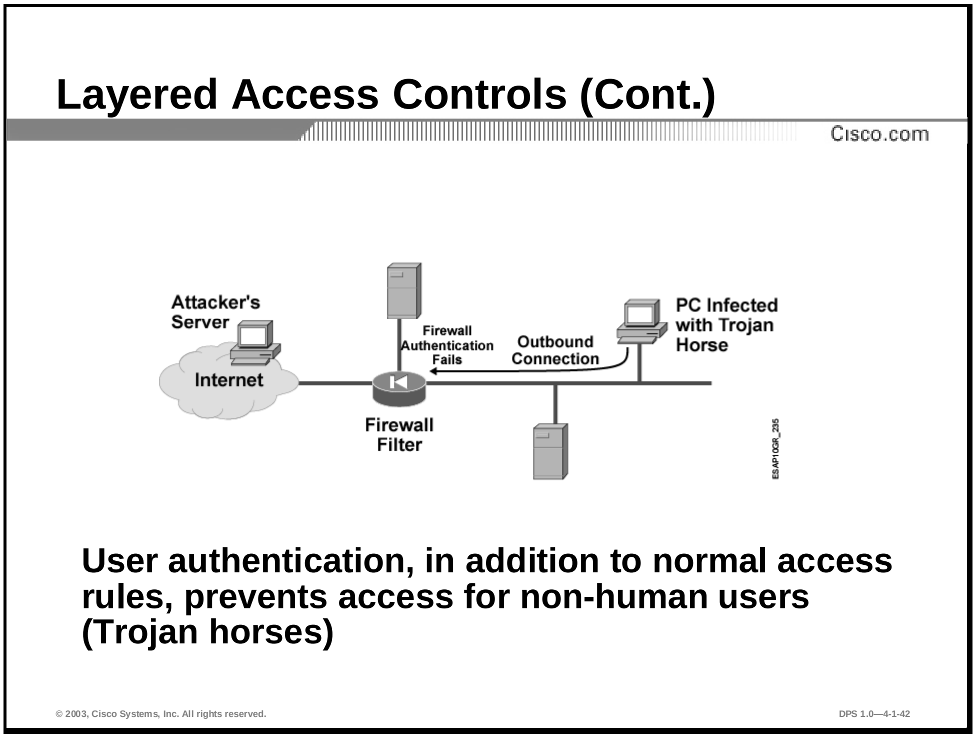
**Protection on additional OSI layers (L2 – L3):**

- **Static ARP entries, Layer 2 filters, port security, private VLANs**

DPS 1.0—4-1-41

This figure illustrates layered software features, which provide additional protection for connectivity within a DMZ network. To further reduce communication options between hosts inside the DMZ, configure the switch with private VLANs. Furthermore, to combat Address Resolution Protocol (ARP) spoofing and content-addressable memory (CAM) table manipulation, configure the:

- Firewall and hosts with static ARP entries for all required neighboring hosts
- Static CAM entries on the switch



## Layered Access Controls (Cont.)

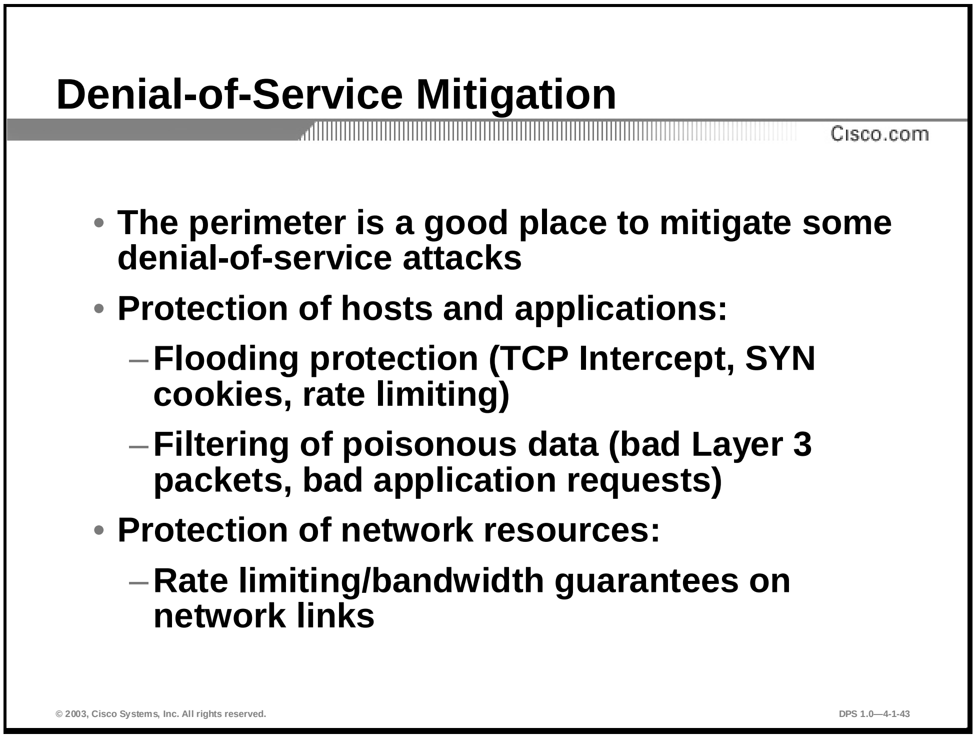
Cisco.com

**User authentication, in addition to normal access rules, prevents access for non-human users (Trojan horses)**

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—4-1-42

This figure illustrates multiple layers of access control provided by multiple firewall authentications. An inside user is allowed access to the outside perimeter, if his connections are sourced from the inside network, and if he authenticates to the firewall for outbound connectivity. Such a stance can lower the risk of malicious software opening unwanted connections to the Internet, as it cannot authenticate as a human user can.



## Denial-of-Service Mitigation

Cisco.com

- The perimeter is a good place to mitigate some denial-of-service attacks
- Protection of hosts and applications:
  - Flooding protection (TCP Intercept, SYN cookies, rate limiting)
  - Filtering of poisonous data (bad Layer 3 packets, bad application requests)
- Protection of network resources:
  - Rate limiting/bandwidth guarantees on network links

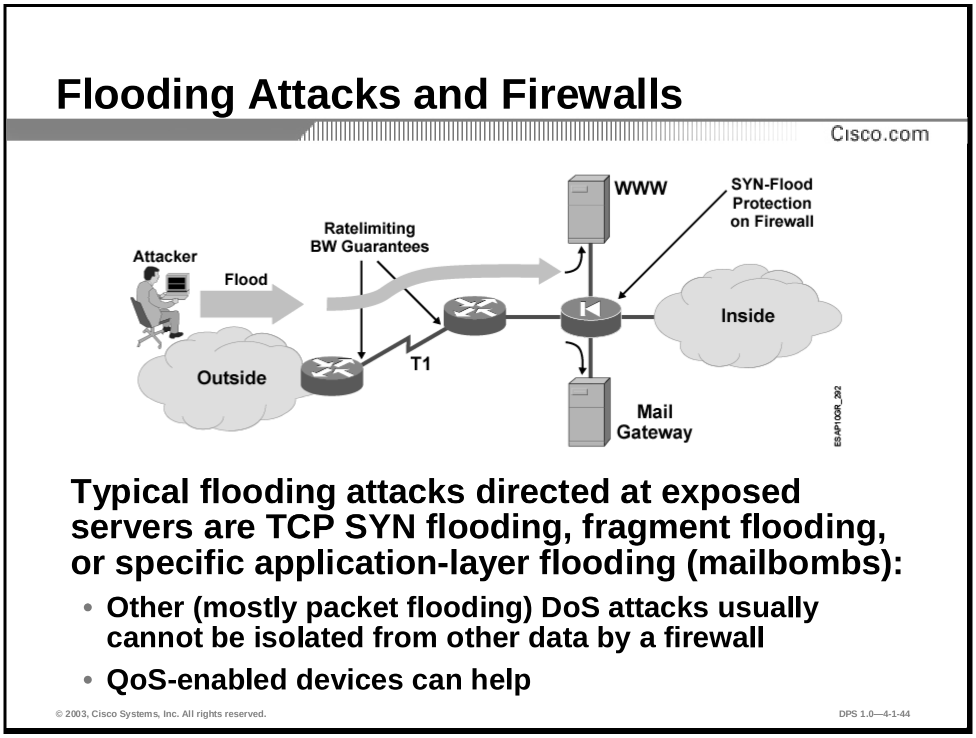
DPS 1.0—4-1-43

## Denial-of-Service Mitigation

The network perimeter is often the place to deploy countermeasures for a wide variety of denial-of-service (DoS) attacks, as they are usually launched from external networks. Such protection involves:

- **Protection of hosts and applications to ensure their availability:** This involves protection against flooding attacks (for example, TCP Intercept or SYN Cookies to guard against SYN flooding) and protection against poisonous data, such as malformed Layer 2 (L2), Layer 3 (L3) or Layer 4 (L4) packets and malformed application requests through, for example, an ALG

- **Protection of network resources:** Such as links through rate limiting or bandwidth guarantees, to ensure availability of bandwidth for mission critical applications during flooding attacks



## Flooding Attacks and Firewalls

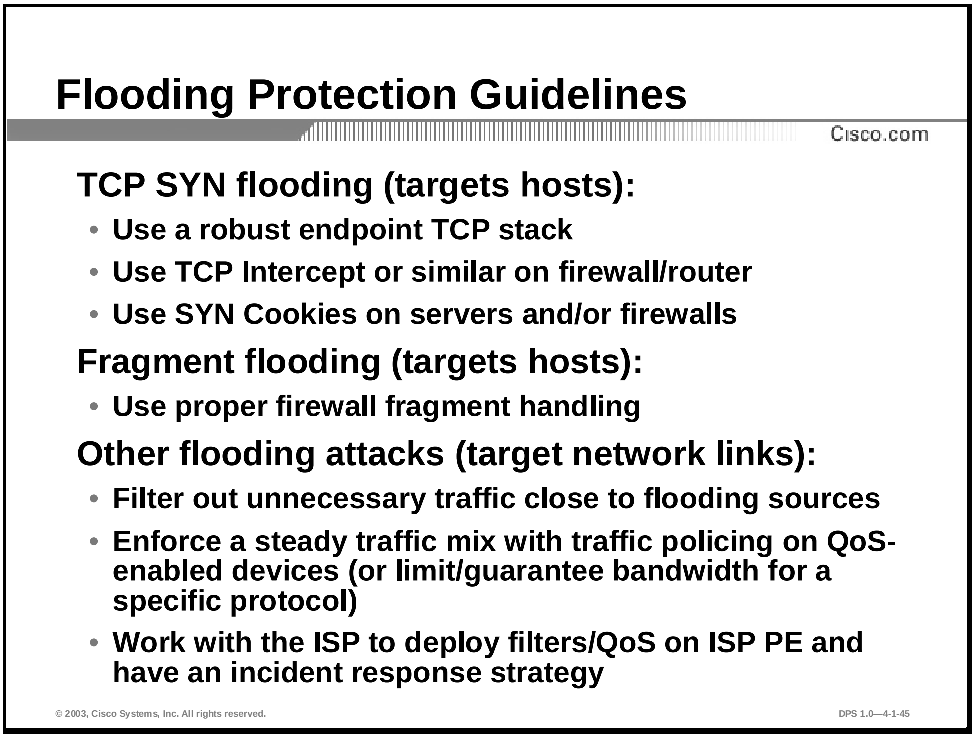
**Typical flooding attacks directed at exposed servers are TCP SYN flooding, fragment flooding, or specific application-layer flooding (mailbombs):**

- **Other (mostly packet flooding) DoS attacks usually cannot be isolated from other data by a firewall**
- **QoS-enabled devices can help**

DPS 1.0—4-1-44

This figure illustrates how quality of service (QoS) devices and firewalls can cooperate to normalize traffic flow to exposed hosts. Firewall technologies can effectively stop some of the well-known flooding attacks, such as SYN flooding. Smart mail gateways can also limit other application-layer attacks, such as mail bombing (sending extremely large mail messages, or an enormous number of items).

In general, flooding DoS attacks are extremely difficult to stop, because the target cannot usually distinguish between legitimate data and data sent by the flooding attacker, as the attacker will try to flood the target with data that closely resembles legitimate packets. QoS-enabled network devices can help limit the impact of a specific flooding attack, so that it impacts a limited number of applications.



## Flooding Protection Guidelines

TCP SYN flooding (targets hosts):

- Use a robust endpoint TCP stack
- Use TCP Intercept or similar on firewall/router
- Use SYN Cookies on servers and/or firewalls

Fragment flooding (targets hosts):

- Use proper firewall fragment handling

Other flooding attacks (target network links):

- Filter out unnecessary traffic close to flooding sources
- Enforce a steady traffic mix with traffic policing on QoS-enabled devices (or limit/guarantee bandwidth for a specific protocol)
- Work with the ISP to deploy filters/QoS on ISP PE and have an incident response strategy

## Guidelines

To defend against TCP SYN Flooding, which targets hosts exposed by the firewall, use:

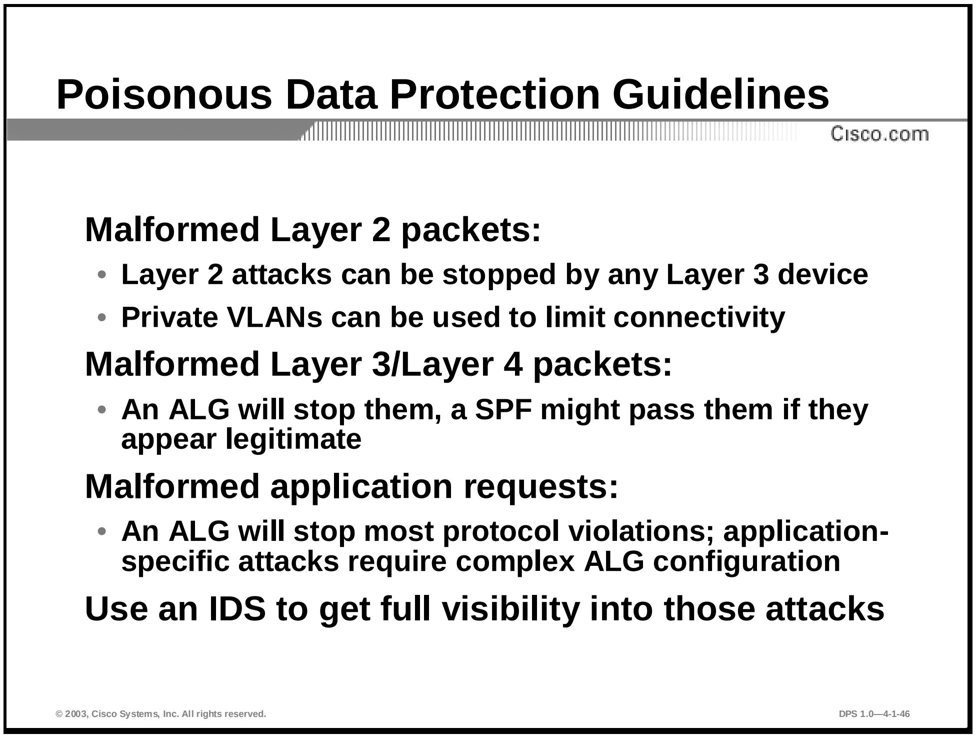
- A robust endpoint TCP stack, if no other mechanisms are available; such a stack will not easily block with a high rate of connections
- TCP Intercept or similar on the firewall/router to “proxy” TCP sessions after they have been established; with TCP Intercept, the firewall system will only allow clients with bidirectional connectivity (i.e. low probability of spoofed source addresses, which are almost always used in TCP SYN flooding attacks) to send packets to the servers
- SYN Cookies on servers and/or firewalls; SYN cookies are a TCP Intercept-like method with better performance characteristics and should be used whenever possible; the PIX Firewall uses SYN cookies for server protection

To defend against fragment flooding, which is directed at hosts exposed by the firewall, use proper firewall fragment handling, where the firewall inspects and reassembles (or, virtually reassembles as in the case of PIX Firewall) IP fragments.

For other flooding attacks, which usually target network links:

- Filter out unnecessary traffic close to flooding sources to minimize the impact on the rest of the network
- Enforce a steady traffic mix with traffic policing on QoS-enabled devices, or limit/guarantee bandwidth for a specific protocol; this will prevent obvious flooding attacks which may use a non-critical protocol (such as ICMP)

- Work with the ISP to deploy filters/QoS on ISP provider edge (PE) and have an incident response strategy, if the flooding cannot be managed at the edge of the network



## Poisonous Data Protection Guidelines

**Malformed Layer 2 packets:**

- **Layer 2 attacks can be stopped by any Layer 3 device**
- **Private VLANs can be used to limit connectivity**

**Malformed Layer 3/Layer 4 packets:**

- **An ALG will stop them, a SPF might pass them if they appear legitimate**

**Malformed application requests:**

- **An ALG will stop most protocol violations; application-specific attacks require complex ALG configuration**

**Use an IDS to get full visibility into those attacks**

DPS 1.0—4-1-46

# Guidelines

- To guard against malformed L2 packets, use a smart L3 device, which can perform some sanity checks. Also, Private VLANs can be used to limit connectivity on L2.

- To guard against malformed L3/L4 packets, use an ALG to stop them. An SPF-based firewall might pass them if they appear legitimate. Fortunately, there are few attacks that work against modern TCP/IP implementations.

- To guard against malformed application requests, use ALG to stop most protocol violations; application-specific attacks will require complex ALG configuration.

**Note** For any DoS attacks, the use of IDS is recommended to get full visibility into the attacks to determine the proper future countermeasures.

# Practice

Q1) What are two common methods of providing defense-in-depth in firewall design?

A) layered topologies

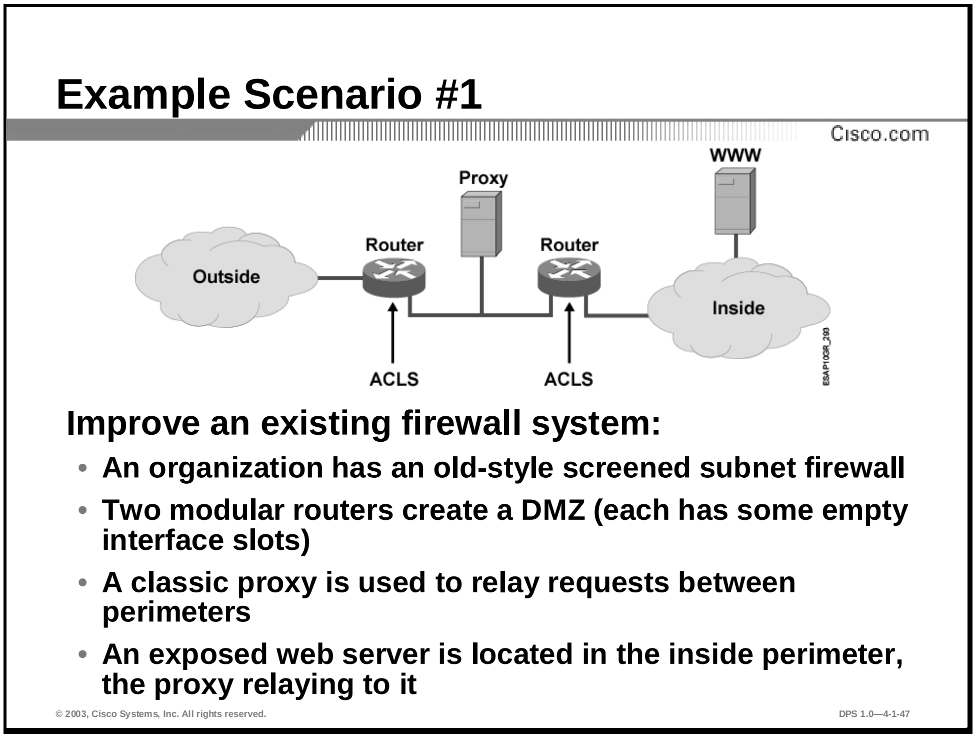
B) private VLANs

C) least privilege rule design

D) layered access control

E) user authentication

# Example Scenarios



### Example Scenario #1

**Improve an existing firewall system:**

- **An organization has an old-style screened subnet firewall**
- **Two modular routers create a DMZ (each has some empty interface slots)**
- **A classic proxy is used to relay requests between perimeters**
- **An exposed web server is located in the inside perimeter, the proxy relaying to it**

DPS 1.0—4-1-47

## Objective

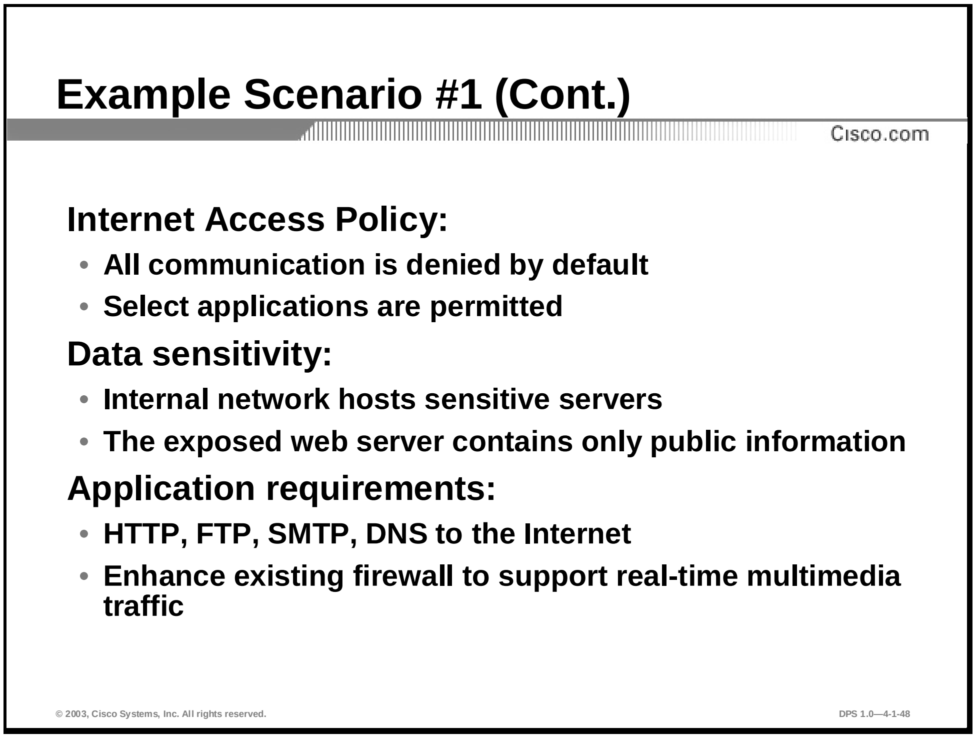
This section will enable the learner to identify common firewall design scenarios.

## Introduction

The following two case studies will illustrate some design decisions, when enhancing an existing firewall, and when building a more complex firewall from scratch.

## Example Scenario #1

An organization has invited you to improve their existing firewall system, which is based on a classic screened subnet architecture, with an ALG passing all traffic between the inside and the outside networks. They use two modular routers, each with some empty slots, to create a single DMZ, and relay inbound web requests to a web server in the inside network.



## Example Scenario #1 (Cont.)

**Internet Access Policy:**

- **All communication is denied by default**
- **Select applications are permitted**

**Data sensitivity:**

- **Internal network hosts sensitive servers**
- **The exposed web server contains only public information**

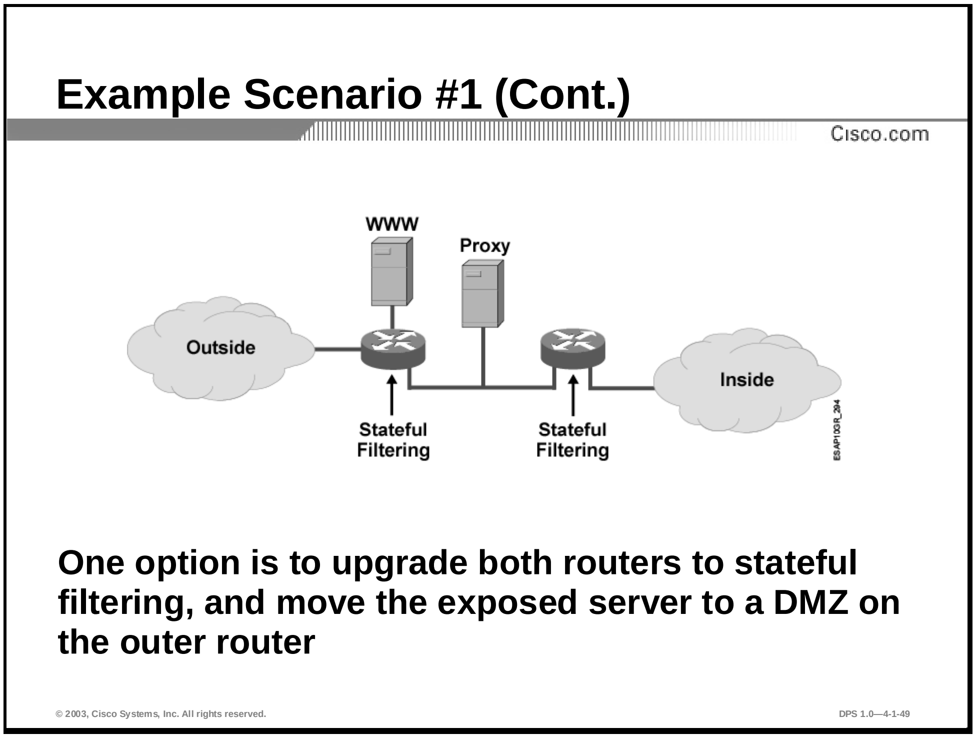
**Application requirements:**

- **HTTP, FTP, SMTP, DNS to the Internet**
- **Enhance existing firewall to support real-time multimedia traffic**

DPS 1.0—4-1-48

The organization access policy is that all communication is denied by default, and only specific applications are permitted. The internal network is a single perimeter, which also hosts sensitive data on its servers. The exposed web server only serves public information, and does not talk to any back-end application.

The policy specifies the following applications should be supported outbound: HTTP, FTP, DNS, and SMTP. The firewall should also support multimedia traffic in the near future.



## Example Scenario #1 (Cont.)

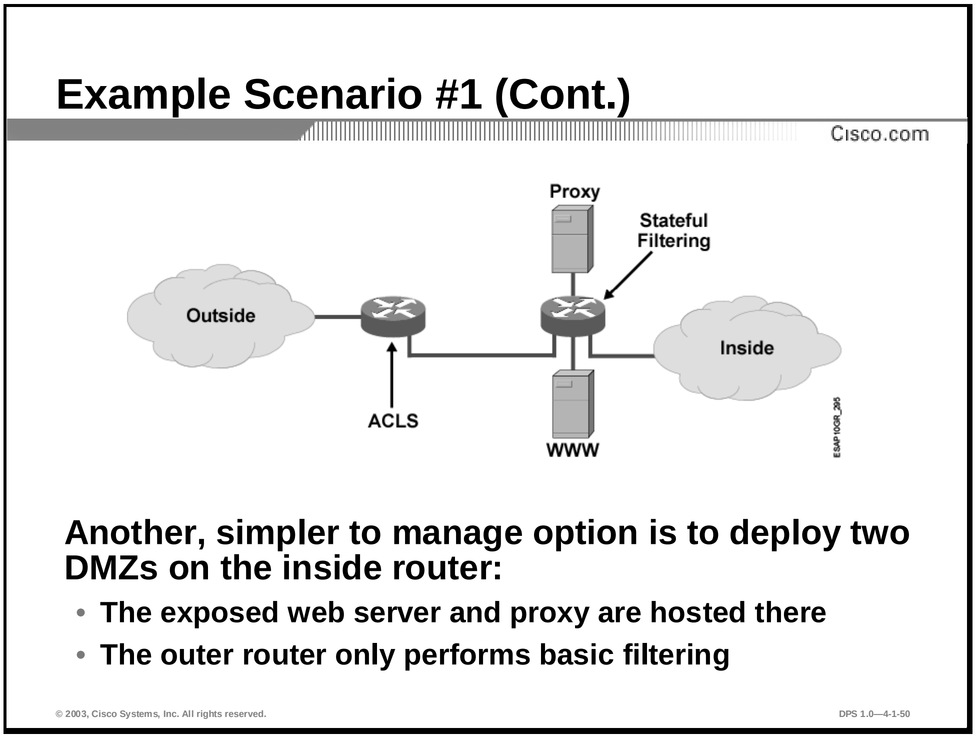
Cisco.com

**One option is to upgrade both routers to stateful filtering, and move the exposed server to a DMZ on the outer router**

DPS 1.0—4-1-49

The first design solution is shown in the picture above. The first design decision must be to enable routing between inside and outside, as the proxy will need to be bypassed to enable multimedia protocols to cross the firewall. Filtering security will be provided by stateful filtering, with select protocols still being proxied over the existing proxy.

The main security issue with the original firewall was the hosting of a public server in the inside network. If broken into, that server would enable the attacker to quickly penetrate other systems on the inside network. The public server was moved into a DMZ, created on the outside router. The public server DMZ is place “more outside” than the proxy, which protects the proxy in the case of public server compromise. Both routers are configured with stateful filtering, and provide defense-in-depth with duplicate rules.



## Example Scenario #1 (Cont.)

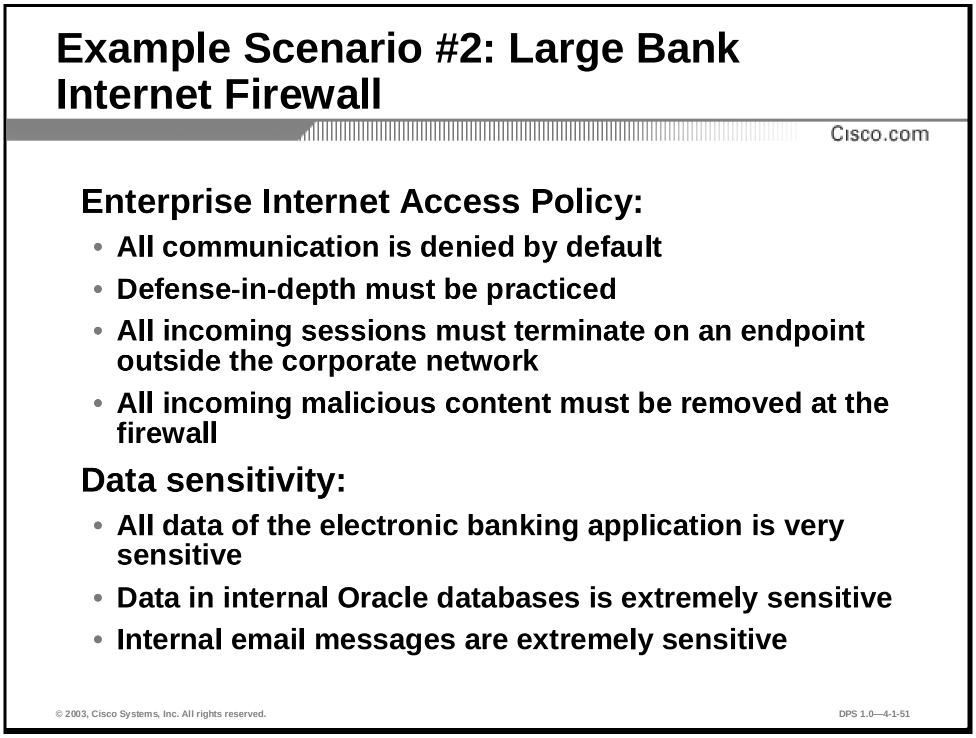
Cisco.com

**Another, simpler to manage option is to deploy two DMZs on the inside router:**

- **The exposed web server and proxy are hosted there**
- **The outer router only performs basic filtering**

DPS 1.0—4-1-50

Another, perhaps simpler and more manageable design is to use the inside router to create DMZ networks, and use the outside router only for most basic security services, such as ingress and egress filtering. The inside router is configured with stateful filtering, and hosts two DMZs for the public server and the proxy server. Multimedia traffic can now be passed directly through the stateful filter, while content inspection for protocols such as HTTP and FTP can still be performed on the proxy. This setup simplifies access control configuration, as the inside router generally performs all access control. If defense-in-depth is desired, the outside router can be configured with matching access rules, designed and configured by another operator.



## Example Scenario #2: Large Bank Internet Firewall

Enterprise Internet Access Policy:

- All communication is denied by default
- Defense-in-depth must be practiced
- All incoming sessions must terminate on an endpoint outside the corporate network
- All incoming malicious content must be removed at the firewall

Data sensitivity:

- All data of the electronic banking application is very sensitive
- Data in internal Oracle databases is extremely sensitive
- Internal email messages are extremely sensitive

DPS 1.0—4-1-51

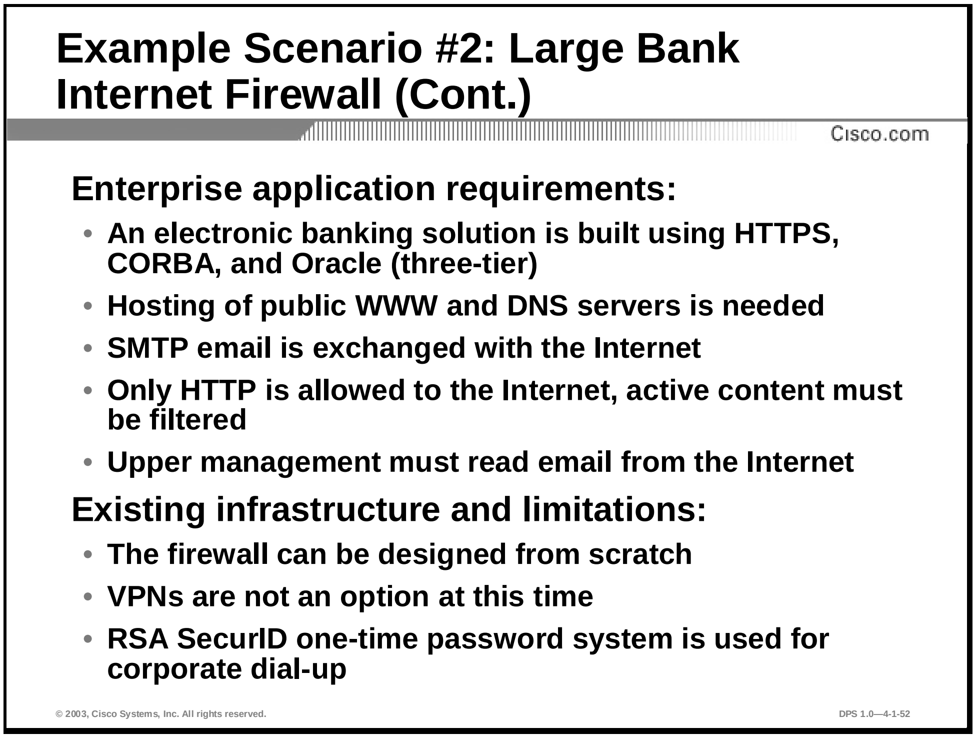
# Example Scenario #2

A more complex example scenario would be that of a bank, which needs to provide many services, and requires the highest levels of security when connecting to untrusted networks.

The general security policy requires that

- All communication over network firewalls must be denied by default
- Defense-in-depth must be practiced whenever possible
- All incoming sessions must terminate outside the inside (campus) network
- All incoming malicious content must be removed at the network boundary

The most sensitive data of the enterprise is kept in the internal Oracle database. There is also an electronic banking application, which accesses internal databases, and its data is also considered very sensitive, as is all internal mail traffic.



## Example Scenario #2: Large Bank Internet Firewall (Cont.)

Cisco.com

### Enterprise application requirements:

- An electronic banking solution is built using HTTPS, CORBA, and Oracle (three-tier)
- Hosting of public WWW and DNS servers is needed
- SMTP email is exchanged with the Internet
- Only HTTP is allowed to the Internet, active content must be filtered
- Upper management must read email from the Internet

### Existing infrastructure and limitations:

- The firewall can be designed from scratch
- VPNs are not an option at this time
- RSA SecurID one-time password system is used for corporate dial-up

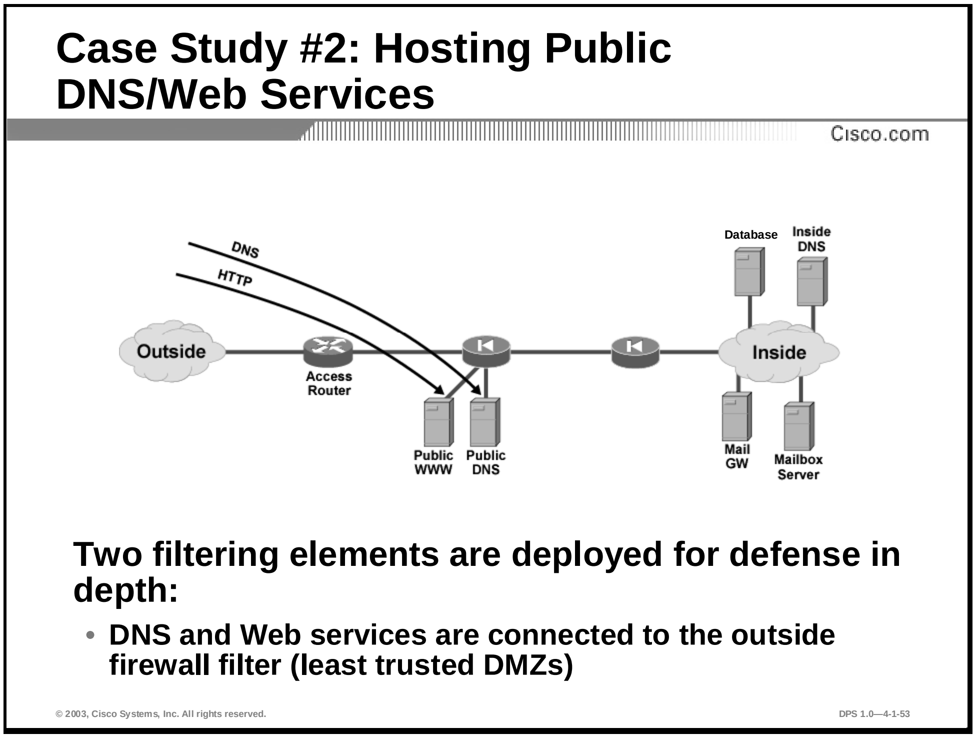
DPS 1.0—4-1-52

The application requirements of the customer are:

- The electronic Internet banking solution uses HTTPS from the client to the web server, CORBA between the first and second application tier, and Oracle SQL*net to access the internal database from the second tier.
- The firewall should enable hosting of public WWW and DNS services
- SMTP email is exchanged with the Internet
- Only HTTP is allowed out to the Internet
- Upper management must be able to access their mail from the Internet

The special requirements and limitations in design are

- The firewall can be designed from scratch
- VPNs cannot be used at this time
- The enterprise already uses the RSA SecurID one-time password system, which should be reused, if possible



## Case Study #2: Hosting Public DNS/Web Services

Cisco.com

**Two filtering elements are deployed for defense in depth:**

- **DNS and Web services are connected to the outside firewall filter (least trusted DMZs)**

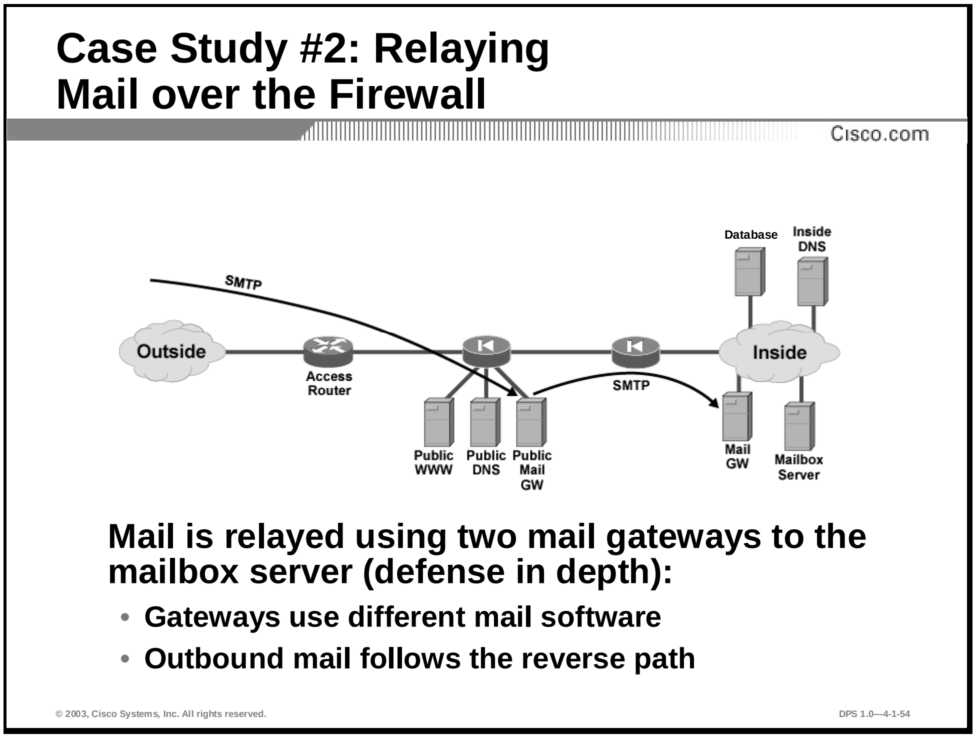
© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—4-1-53

Let’s start building the firewall. To provide initial defense-in-depth, two filtering elements will be deployed to perform access control. This will simplify their individual configurations, and provide back-up in the case that access control fails on one of the filtering elements.

The rules of building DMZs will be:

- Each inbound service should terminate it its own DMZ
- Less trusted services will be placed more “outside” compared to more trusted services
- E-commerce application tiers will be separated

First, let’s create two DMZ for the public DNS and Web servers. Those servers host public information, and will be accessible to all users of the Internet. Their compromise is likely, if application security is not extremely tight, and their compromise should not influence the security of other services. We decide to place them in two least trusted DMZs on the outside filtering element.



# Case Study #2: Relaying Mail over the Firewall

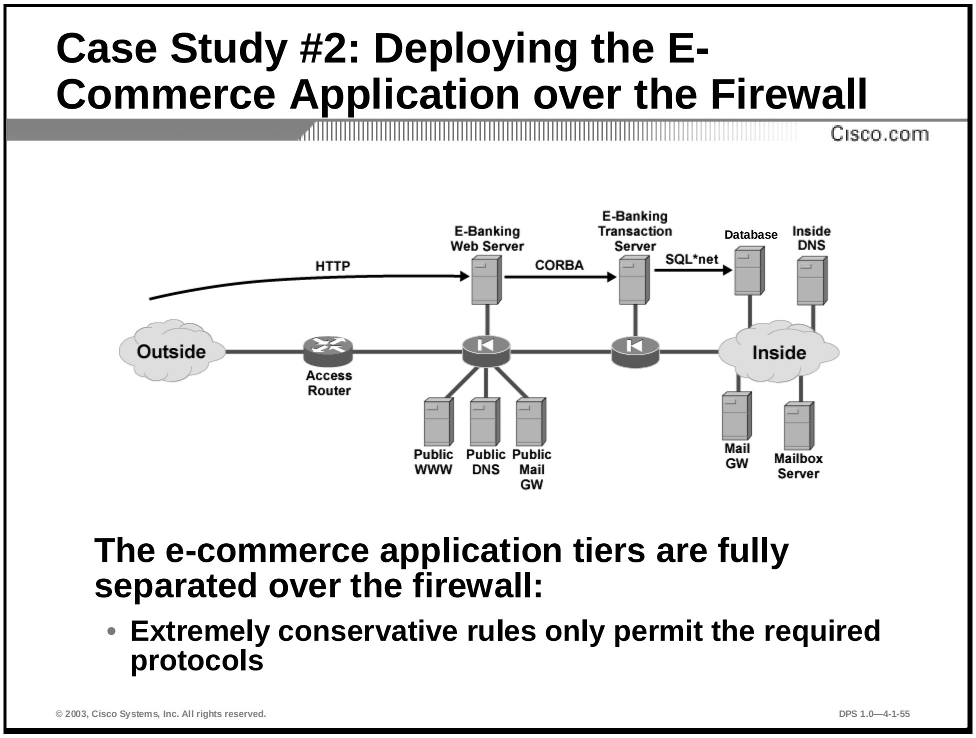
Cisco.com

**Mail is relayed using two mail gateways to the mailbox server (defense in depth):**

- **Gateways use different mail software**
- **Outbound mail follows the reverse path**

© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—4-1-54

To relay SMTP email over the firewall, an application layer gateway is needed to strip malicious content. As the mail relay will be exposed to the Internet, similar precautions apply as with the Web and DNS servers. The mail gateway should use secure mailer software, and there will be two mail gateways relaying mail to the inside, running different mailer software. This provides good defense in depth, where it is needed: we need to support incoming connections, and if the outside mail relay is compromised, another mail relay needs to be compromised before the attacker is on the inside network. The same bug should not be present on both systems, therefore the probability of full compromise is very small. Outbound mail flows over the reverse path over the two mail relays.



## Case Study #2: Deploying the E-Commerce Application over the Firewall

Cisco.com

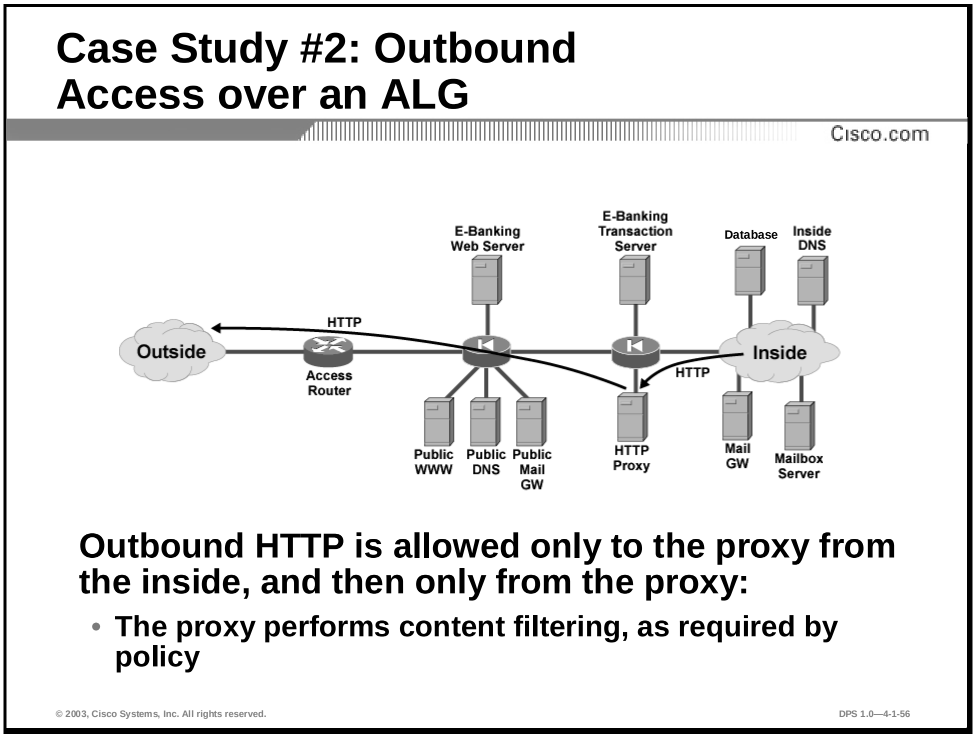
**The e-commerce application tiers are fully separated over the firewall:**

- **Extremely conservative rules only permit the required protocols**

DPS 1.0—4-1-55

The three-tier e-commerce application handles very sensitive data. The external server is open to the Internet, so we host it on the external firewall. Its DMZ is more trusted than all other DMZs on the outside firewall, as that server handles the most sensitive data of all public servers. If an attacker would break into one of the other public servers, the access rules on all DMZs should prevent any further access.

The least privilege concept is extremely important when deploying the multi-tiered application over this firewall. Each tier has its own DMZ, and the access rules on the firewall should permit the minimum necessary connectivity between tiers to minimize exposure. Only a pure application-layer attack should enable the attacker to get to the inside network over the E-commerce tiers.



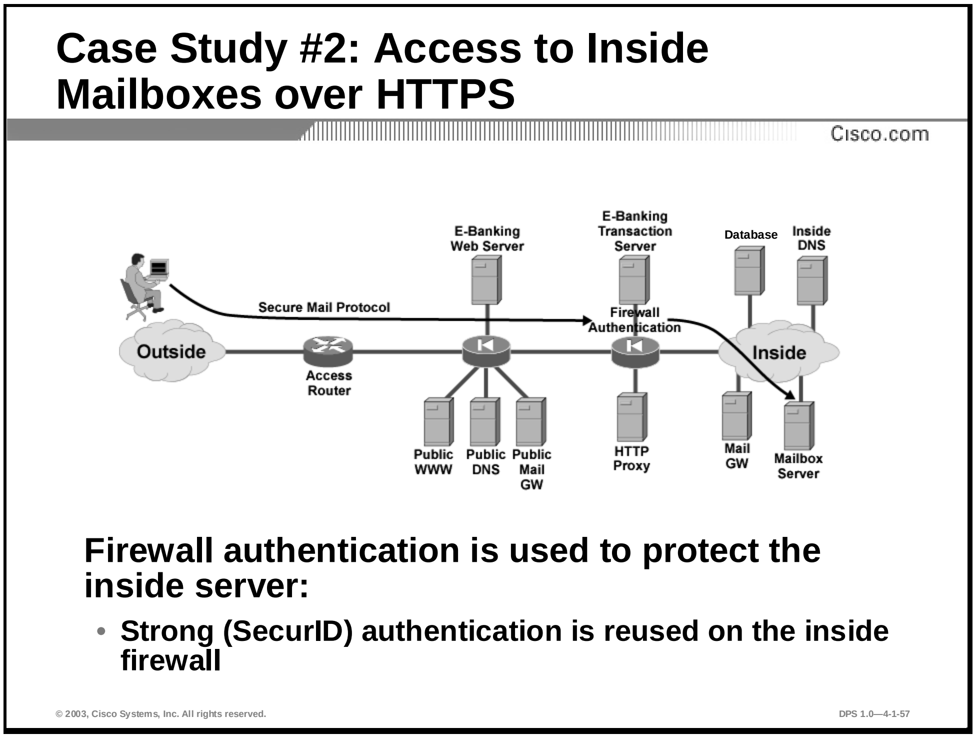
## Case Study #2: Outbound Access over an ALG

**Outbound HTTP is allowed only to the proxy from the inside, and then only from the proxy:**

- **The proxy performs content filtering, as required by policy**

DPS 1.0—4-1-56

Outbound access (only HTTP is required) needs to be filtered to remove all malicious content. An ALG is deployed in a DMZ of the internal firewall, and provides access control and filtering for HTTP content.



## Case Study #2: Access to Inside Mailboxes over HTTPS

Cisco.com

**Firewall authentication is used to protect the inside server:**

- **Strong (SecurID) authentication is reused on the inside firewall**

DPS 1.0—4-1-57

The last requirement is to allow inbound access to mailboxes for the upper management. If a secure (authenticated and encrypted) mail pickup protocol is used (such as HTTPS or Lotus Notes), direct sessions to the inside mailbox server can be permitted, if firewall can properly authenticate the user. In this case, the inside firewall filter will allow mail client connections to the mailbox server, if the client has passed one-time password authentication on the inside firewall.

## Practice

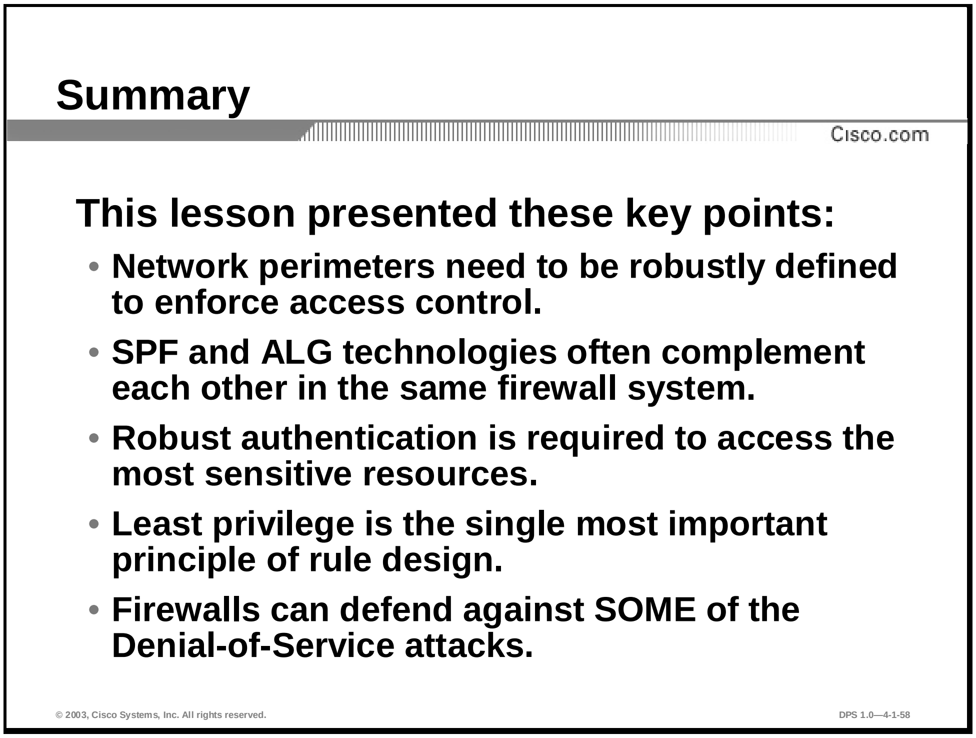
Q1) True or false? This example provides defense-in-depth for SMTP mail relaying to the inside.

A) true

B) false

# Summary

This section summarizes the key points discussed in this lesson.



## Summary

Cisco.com

This lesson presented these key points:

- Network perimeters need to be robustly defined to enforce access control.
- SPF and ALG technologies often complement each other in the same firewall system.
- Robust authentication is required to access the most sensitive resources.
- Least privilege is the single most important principle of rule design.
- Firewalls can defend against SOME of the Denial-of-Service attacks.

DPS 1.0—4-1-58

# Next Steps

After completing this lesson, go to:

- High Performance and High Availability Firewalls lesson

# Quiz: Firewall Design

Complete the quiz to assess what you have learned in this lesson.

## Objectives

This quiz tests your knowledge on how to:

- Design an abstract firewall system, enforcing a defined security policy and using best practices design methods

## Instructions

Answer these questions:

1. What is a network perimeter?
2. When are ALGs deployed to provide outbound connectivity?
3. Why is the issue of transitive trust important in firewall design?
4. What is the main pitfall of IP address-based source authentication?
5. How do firewalls provide defense-in-depth?

## Scoring

You have successfully completed the quiz for this lesson when you earn a score of 80 percent or better.

# High Availability and High Performance Firewalls

## Overview

### Importance

Growing enterprise network sizes and an increasing amount of online workers and electronic commerce users increases the demand of high performance firewalls. The increasing number of transactions made over the Internet demands high availability of services—service outages for hours or days might lead to enormous costs for companies or even bankruptcy.

### Lesson Objective

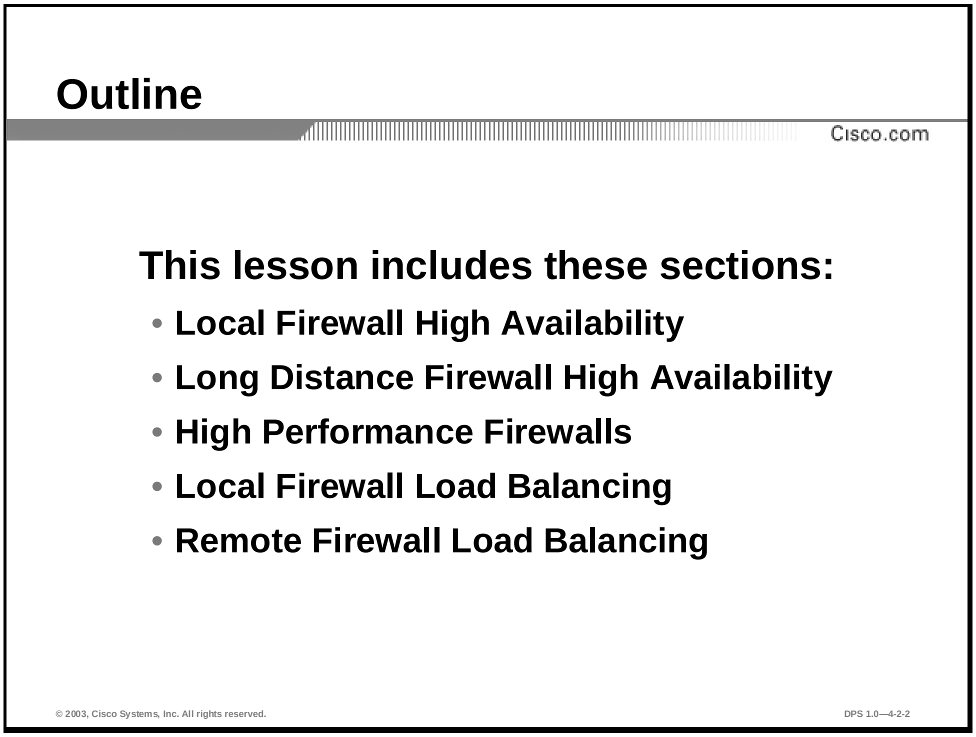
The lesson will enable the learner to design a firewall system supporting high-availability and high levels of performance.

## Learner Skills and Knowledge

To fully benefit from this lesson, you must have these prerequisite skills and knowledge:

- Understand the concept of a firewall
- Solid knowledge about bidirectional NAT
- Basic knowledge about load balancing
- Basic knowledge about the BGP and OSPF routing protocols

## Outline



### Outline

Cisco.com

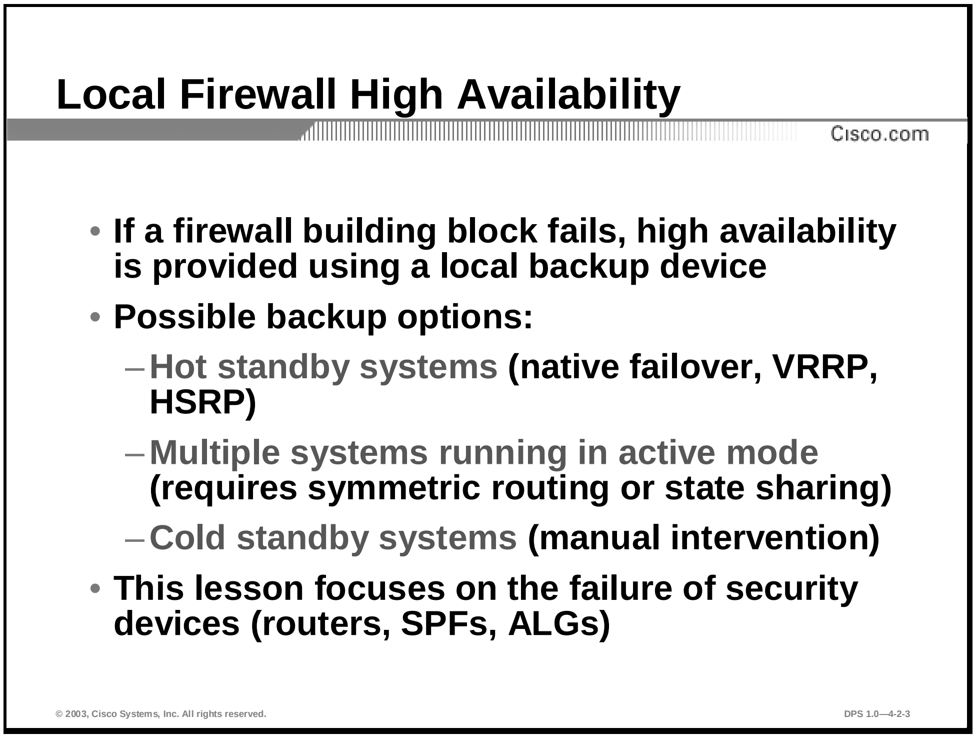
**This lesson includes these sections:**

- **Local Firewall High Availability**
- **Long Distance Firewall High Availability**
- **High Performance Firewalls**
- **Local Firewall Load Balancing**
- **Remote Firewall Load Balancing**

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—4-2-2

# Local Firewall High Availability



## Local Firewall High Availability

- If a firewall building block fails, high availability is provided using a local backup device
- Possible backup options:
  - Hot standby systems (native failover, VRRP, HSRP)
  - Multiple systems running in active mode (requires symmetric routing or state sharing)
  - Cold standby systems (manual intervention)
- This lesson focuses on the failure of security devices (routers, SPFs, ALGs)

© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—4-2-3

## Objective

The section will enable the learner to explain the concept of local firewall high availability and choose it in appropriate design situations.

## Introduction

High availability of network services is very important to enterprises due to the high cost of network outages. The entry point in the network is the most critical issue for service availability, and since firewalls are necessary to protect the whole enterprise network from attackers, this critical entry point is typically a firewall. If one firewall building block fails there are often only two decisions left: either to allow unprotected access to the network or totally cut off the line. Thus, firewall availability is a very important issue for enterprise network architecture. This section presents local firewall high availability techniques.

## Backup Solutions

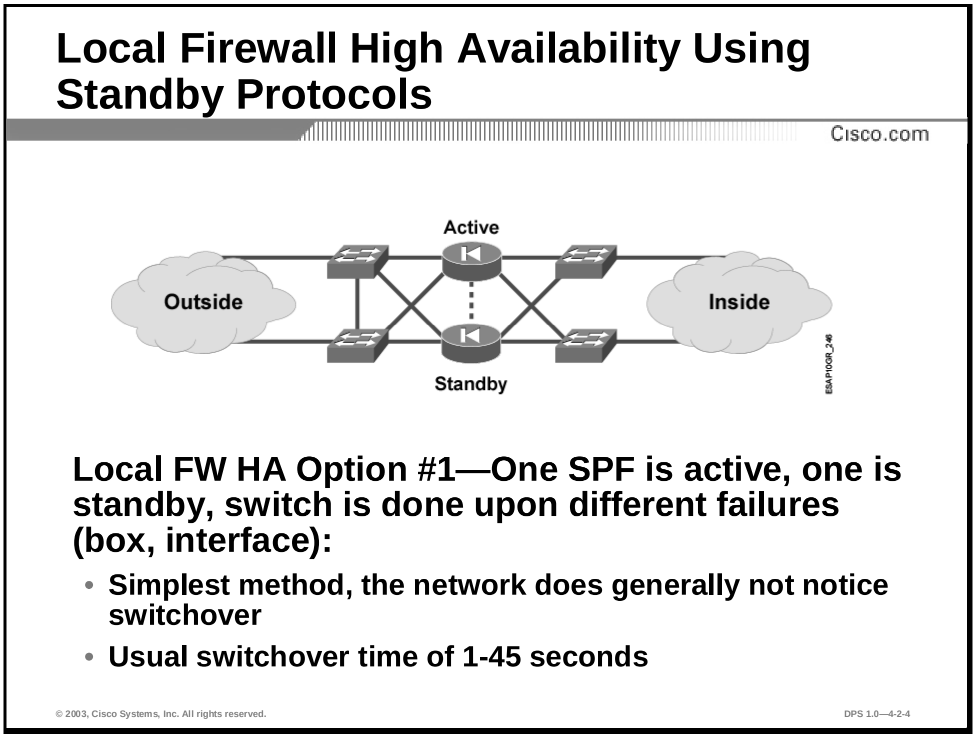
There are three commonly used backup solutions. First, hot standby systems can be employed that enable a native failover in very short times using the Virtual Router Redundancy Protocol (VRRP) or the Hot Standby Routing Protocol (HSRP). Second, multiple systems can be used running in active mode, which requires symmetric routing or state sharing. And third, cold standby systems, which requires manual intervention to solve the problem of a failure.

This lesson assumes failures of security devices such as routers, Stateful Packet Filters (SPF), Application Layer Gateways (ALG), and others.

## HSRP Overview

Using HSRP, a set of routers works in concert to present the illusion of a single virtual router to the hosts on the LAN. This set is known as an HSRP group or a standby group. A single router elected from the group is responsible for forwarding the packets that hosts send to the virtual router. This router is known as the Active router. Another router is elected as the Standby router. In the event that the Active router fails, the Standby assumes the packet forwarding duties of the Active router. Although an arbitrary number of routers may run HSRP, only the Active router forwards the packets sent to the virtual router. To minimize network traffic, only the Active and Standby routers send periodic HSRP messages once the protocol has completed the election process. If the Active router fails, the Standby router takes over as the Active router. If the Standby router fails, or becomes the Active router, then another router is elected as the Standby router. On a particular LAN, multiple hot standby groups may coexist and overlap. Each standby group emulates a single virtual router. The individual routers may participate in multiple groups. In this case, the router maintains separate state and timers for each group. Each standby group has a single, well-known MAC address, as well as an IP address.

**Reference** A comprehensive description of HSRP is given in RFC 2281.



## Local Firewall High Availability Using Standby Protocols

Cisco.com

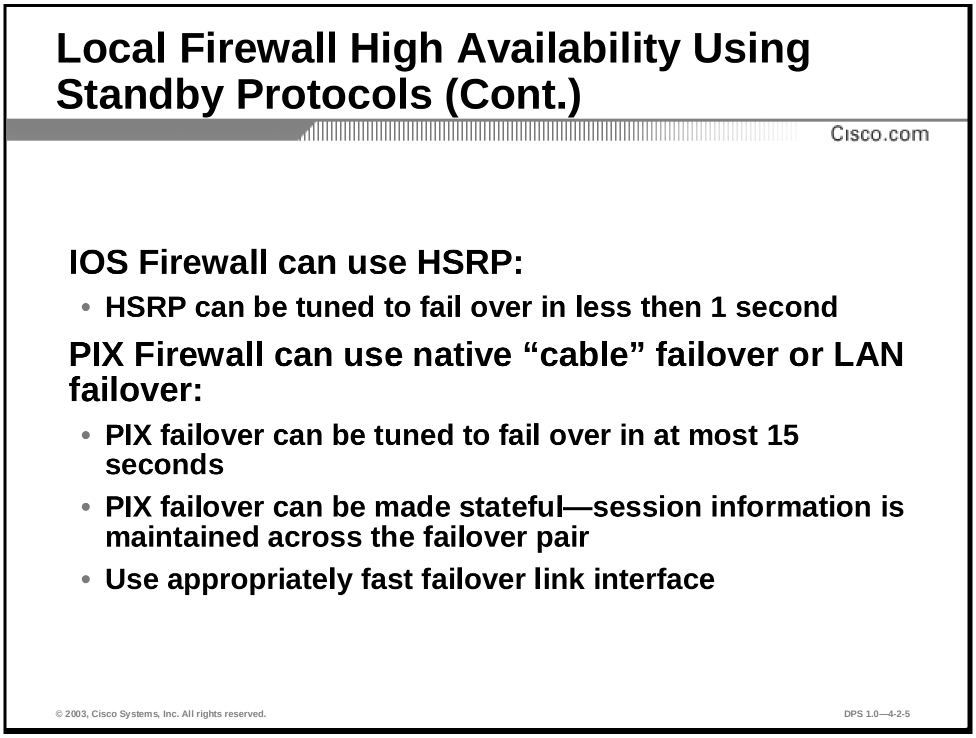
**Local FW HA Option #1—One SPF is active, one is standby, switch is done upon different failures (box, interface):**

- **Simplest method, the network does generally not notice switchover**
- **Usual switchover time of 1-45 seconds**

DPS 1.0—4-2-4

## Availability by Hot Standby

One of the fastest standby solutions is to utilize switches that provide redundant connections between one active SPF and a parallel “hot standby” SPF. This very simple method allows switchover times between 1 and 45 seconds, allowing for unnoticeable switchover of traffic on the network.



### Local Firewall High Availability Using Standby Protocols (Cont.)

**IOS Firewall can use HSRP:**

- **HSRP can be tuned to fail over in less then 1 second**

**PIX Firewall can use native “cable” failover or LAN failover:**

- **PIX failover can be tuned to fail over in at most 15 seconds**
- **PIX failover can be made stateful—session information is maintained across the failover pair**
- **Use appropriately fast failover link interface**

© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—4-2-5

# Failure Detection

To determine the failure of the active components several methods can be implemented. The IOS firewall can use the HSRP, which can be tuned to fail over in less than 1 second.

However, if using a PIX Firewall, configure a simple native “cable” failover. A LAN failover is possible. Both methods achieve switching times below 15 seconds.

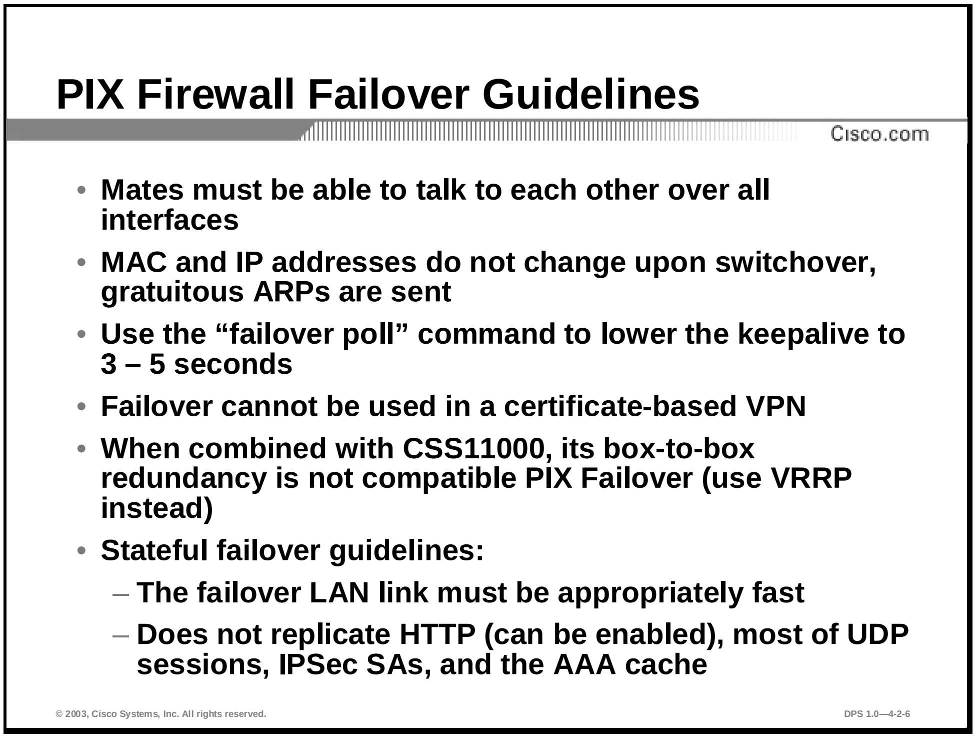
## The Failover Cable

The failover cable is the only additional hardware required to support PIX failover. In PIX 6.2 and later, a failover can be achieved with or without a failover cable. The failover cable is a modified RS−232 serial link cable with a speed setting of 9600 baud. In PIX Software Release 5.2 (5.1.2.201), the speed was changed to 115.2K baud. If a switchover occurs, the units swap the IP address and MAC addresses they are using to replace each other’s presence on the network. This action is invisible to the network.

# Importance of State-Tracking

The most important property of an active standby solution is the capability of the standby device to track all traffic states that the active component possesses. Because of this capability the switchover is so fast.

**Note** When combining a PIX Firewall with the Content Service Switch (CSS) 11000, the box-to-box redundancy is not a compatible PIX failover. Use the Virtual Router Redundancy Protocol (VRRP) instead.



PIX Firewall Failover Guidelines

- Mates must be able to talk to each other over all interfaces
- MAC and IP addresses do not change upon switchover, gratuitous ARPs are sent
- Use the “failover poll” command to lower the keepalive to 3 – 5 seconds
- Failover cannot be used in a certificate-based VPN
- When combined with CSS11000, its box-to-box redundancy is not compatible PIX Failover (use VRRP instead)
- Stateful failover guidelines:
  - The failover LAN link must be appropriately fast
  - Does not replicate HTTP (can be enabled), most of UDP sessions, IPSec SAs, and the AAA cache

DPS 1.0—4-2-6

## Failover Guidelines for the PIX Firewall

Designing a PIX Firewall failover should take the following guidelines into account:

- Mates must be able to talk to each other over all interfaces in order to maximize the reliability of the connectivity.

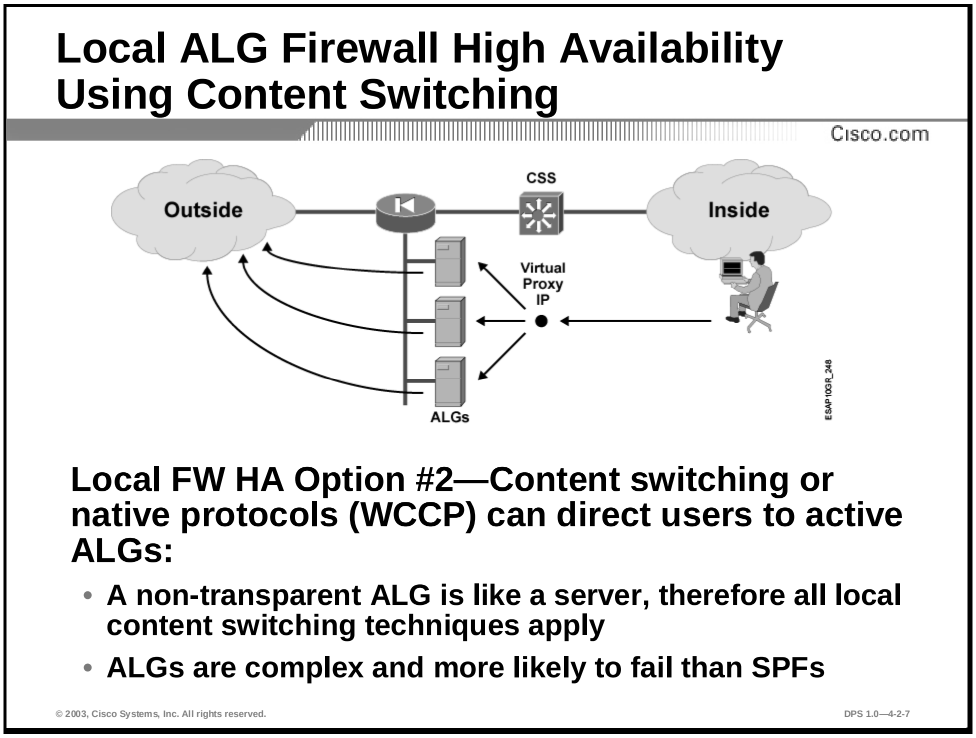
- After a switchover, the MAC and IP addresses do not change. When a device changes state from standby to active, or from active to standby, a “gratuitous ARP” is sent to each network interface to rebroadcast the new IP and MAC addresses.

- The failover poll is used to monitor network activity, failover communications, and the power status. A failure of any of these parameters on the active unit will cause the standby unit to take active control. Use the “failover poll” command to minimize the keepalive to 3-5 seconds. The default failover poll interval is 15 seconds.

- Do not use failover functionality in a certificate-based VPN, as the PIXes cannot share the private RSA key.

- The CSS11000 does not support a compatible box-to-box redundancy, use VRRP instead.

- Stateful failover is more traffic friendly but requires an appropriately fast LAN link in-between. It does not replicate short-time sessions such as HTTP sessions or pseudo-sessions, such as most User Datagram Protocol (UDP) connections. Also, it is not possible to replicate sessions that are protected by cryptographic measures, such as IPSec Security Associations (SAs) and the AAA cache.



## Local ALG Firewall High Availability Using Content Switching

Cisco.com

**Local FW HA Option #2—Content switching or native protocols (WCCP) can direct users to active ALGs:**

- **A non-transparent ALG is like a server, therefore all local content switching techniques apply**
- **ALGs are complex and more likely to fail than SPFs**

DPS 1.0—4-2-7

# Fault Redundancy for ALGs

ALGs require a completely different high availability consideration. Use either content switching or native standby protocols such as the Web Cache Communication Protocol (WCCP) to design a quick failover. Developed by Cisco Systems, the WCCP specifies interactions between one or more routers or Layer 3 (L3) switches, and one or more web-caches. The purpose of the interaction is to establish and maintain the transparent redirection of selected types of traffic flowing through a group of routers. It redirects the selected traffic to a group of web-caches with the aim of optimizing resource usage and lowering response times.

It is important to understand the server-like functionality of a non-transparent ALG. Thus, ALGs can also use all the local content switching that applies to servers.

# Practice

Q1) What is the approximate switchover time when using standby protocols for local firewall high availability?

A) 1-2 seconds

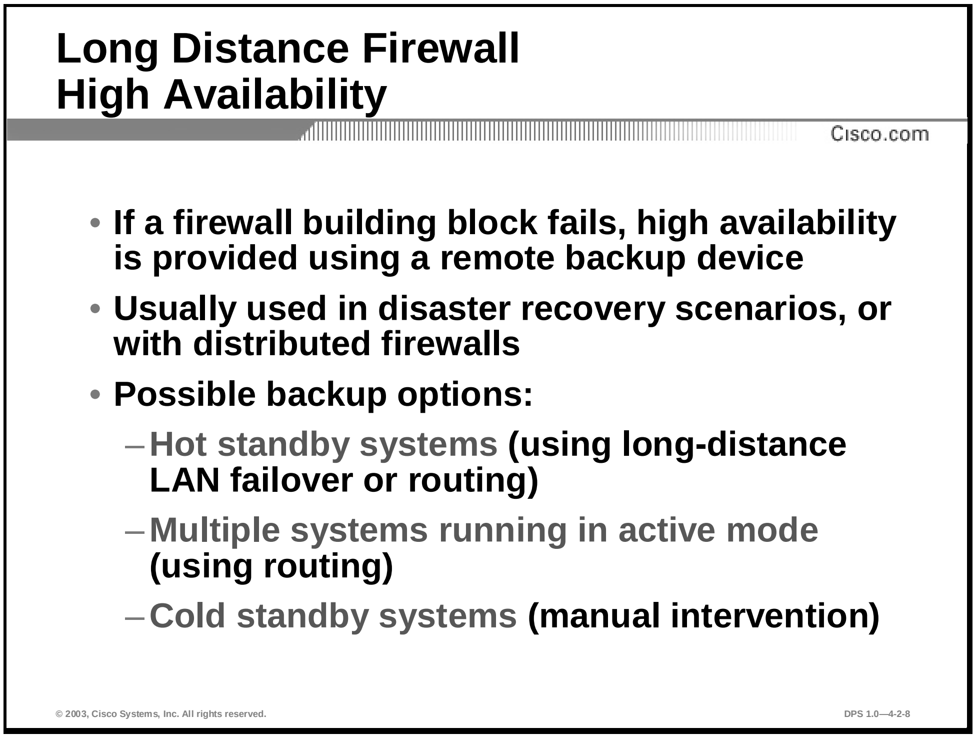
B) 1-45 seconds

C) a couple of minutes

D) up to 10 seconds

E) less than 1 second

# Long Distance Firewall High Availability



## Long Distance Firewall High Availability

Cisco.com

- If a firewall building block fails, high availability is provided using a remote backup device
- Usually used in disaster recovery scenarios, or with distributed firewalls
- Possible backup options:
  - Hot standby systems (using long-distance LAN failover or routing)
  - Multiple systems running in active mode (using routing)
  - Cold standby systems (manual intervention)

DPS 1.0—4-2-8

## Objective

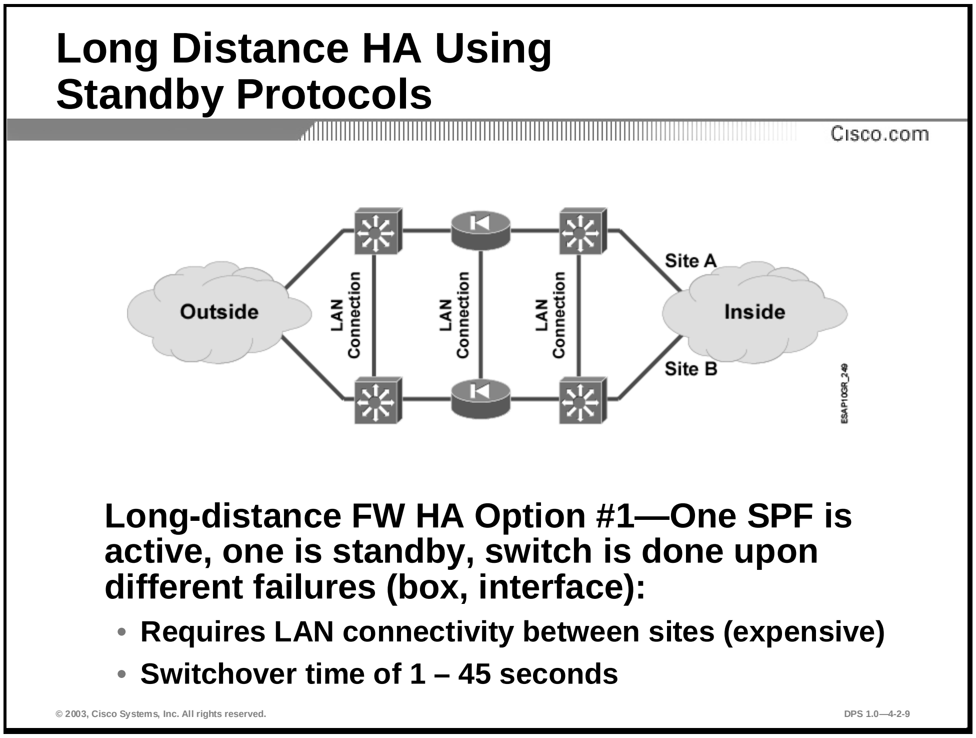
The section will enable the learner to explain the concept of long-distance firewall high availability and choose it in appropriate design situations.

## Introduction

Some high-availability scenarios (such as disaster  recovery centers) require the use of a remote firewall, which is set up in a remote location. Therefore, geographically separate the backup devices from the active device.

Basically, use similar solutions as for local high availability demands, such as hot standby systems, multiple systems running in active mode, or cold standby systems:

- Hot standby systems require a specialized long-distance LAN failover or a routing protocol.
- Employing multiple systems running in active mode requires failover routing.
- Generally, cold standby systems are activated by manual intervention. This method is the slowest of the listed possibilities.



## Long Distance HA Using Standby Protocols

Cisco.com

**Long-distance FW HA Option #1—One SPF is active, one is standby, switch is done upon different failures (box, interface):**

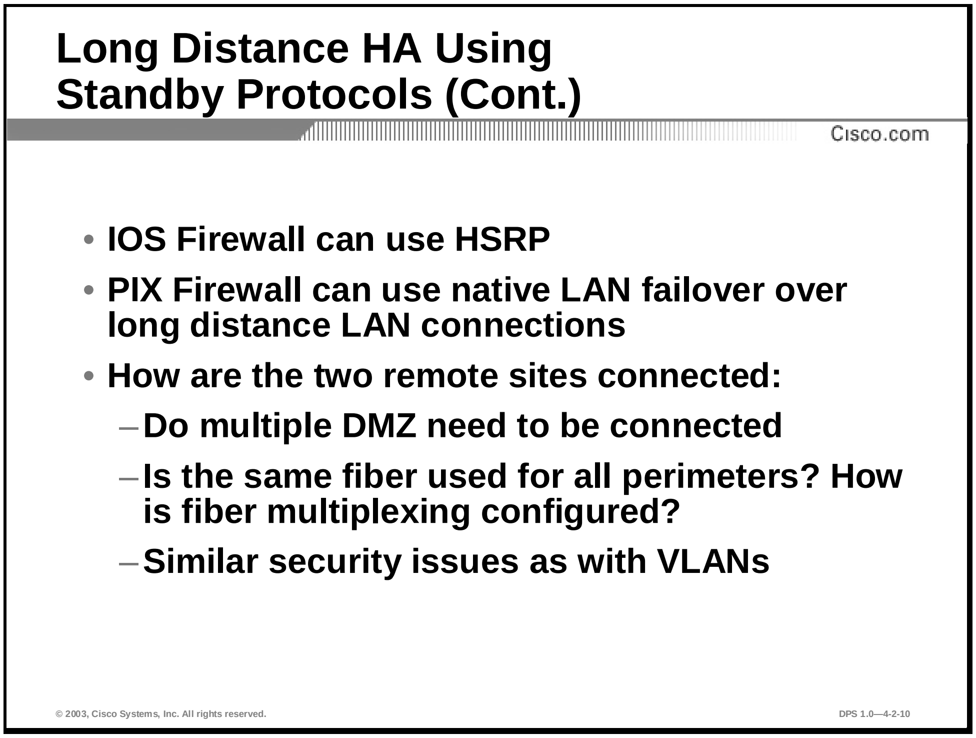
- **Requires LAN connectivity between sites (expensive)**
- **Switchover time of 1 – 45 seconds**

DPS 1.0—4-2-9

## Long Distance Failover Switching

This solution forwards the traffic to the remote standby devices over a LAN connection, when different failure types occur—box or interface malfunctions. It is relatively fast solution providing similar failover switching times as those for local redundancy solutions, typically 1 to 45 seconds.

Using a high-speed LAN technology to connect widely separated sites is both simple and fast although it is relatively expensive.



## Long Distance HA Using Standby Protocols (Cont.)

- IOS Firewall can use HSRP
- PIX Firewall can use native LAN failover over long distance LAN connections
- How are the two remote sites connected:
  - Do multiple DMZ need to be connected
  - Is the same fiber used for all perimeters? How is fiber multiplexing configured?
  - Similar security issues as with VLANs

DPS 1.0—4-2-10

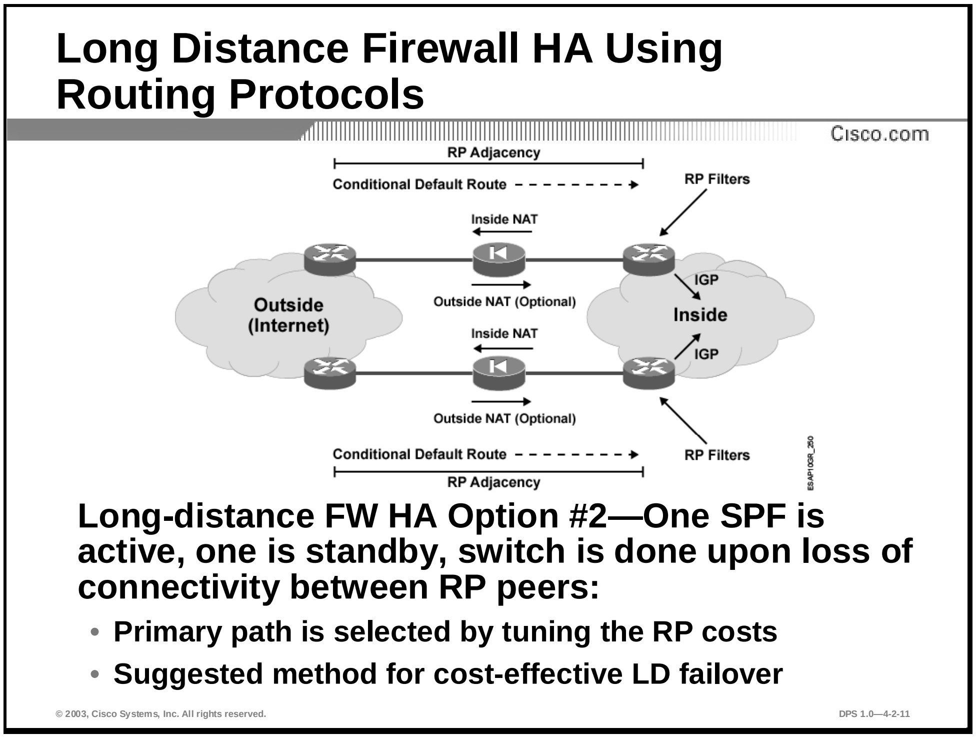
## Standby Protocols

Use HSRP to set up long distance standby connectivity for IOS Firewalls and native LAN links to implement PIX firewall redundant connections. For such connections, several issues must be considered:

- Is it necessary to connect multiple DMZ? In this case, multiple fibers or multiplexing will be used to connect the systems, increasing cost considerably.

- Is the same fiber used for all perimeters? How is the fiber multiplexing configured? Does a simple misconfiguration of the optical switch mix up all your DMZs?

Those issues are similar to those of VLANs. Make sure you are aware of them, and plan accordingly.



## Long Distance Firewall HA Using Routing Protocols

Cisco.com

**Long-distance FW HA Option #2—One SPF is active, one is standby, switch is done upon loss of connectivity between RP peers:**

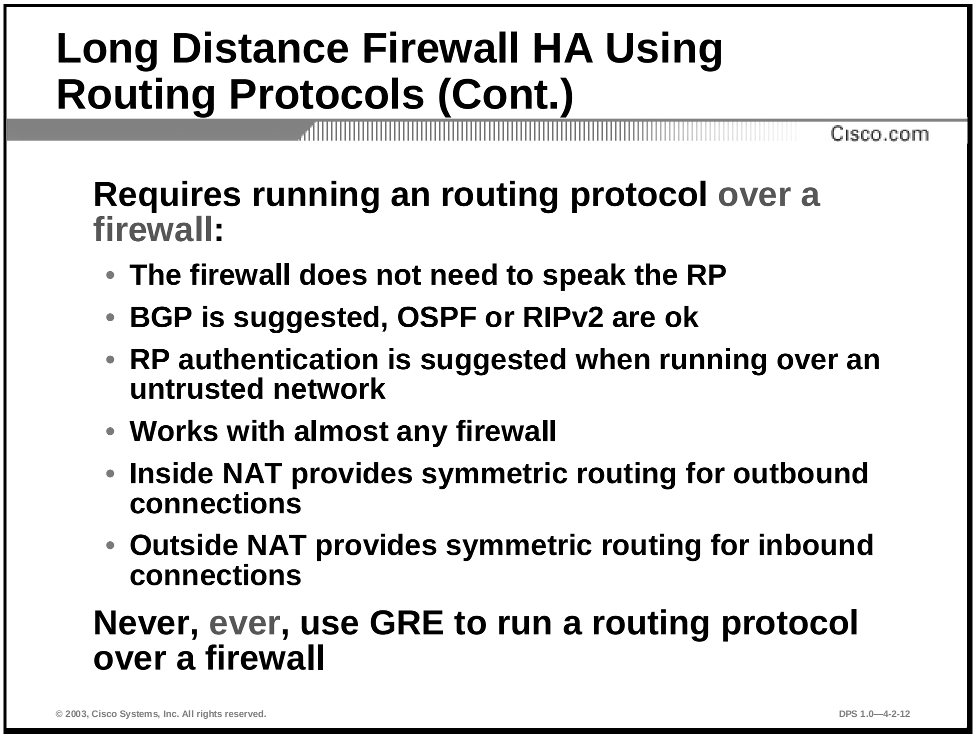
- **Primary path is selected by tuning the RP costs**
- **Suggested method for cost-effective LD failover**

DPS 1.0—4-2-11

## Long Distances Failover Using Routing Protocols

Another high availability solution is implemented with routers. This solution does not need the establishment of a dedicated line between the two sites, and if routing—especially in the outside network—provides high availability routes, it achieves the same total availability degree as redundant switching.

The primary path to be used for the active device is selected by tuning the routing protocol metric. Using Border Gateway Protocol (BGP), the active path can be configured by use of either the local preference BGP attribute, or neighbor weights. Typical switchover time is 10 to 15 seconds, so the routing solution is the most recommended for a low-cost low delay failover.



## Long Distance Firewall HA Using Routing Protocols (Cont.)

Cisco.com

Requires running an routing protocol over a firewall:

- The firewall does not need to speak the RP
- BGP is suggested, OSPF or RIPv2 are ok
- RP authentication is suggested when running over an untrusted network
- Works with almost any firewall
- Inside NAT provides symmetric routing for outbound connections
- Outside NAT provides symmetric routing for inbound connections

Never, ever, use GRE to run a routing protocol over a firewall

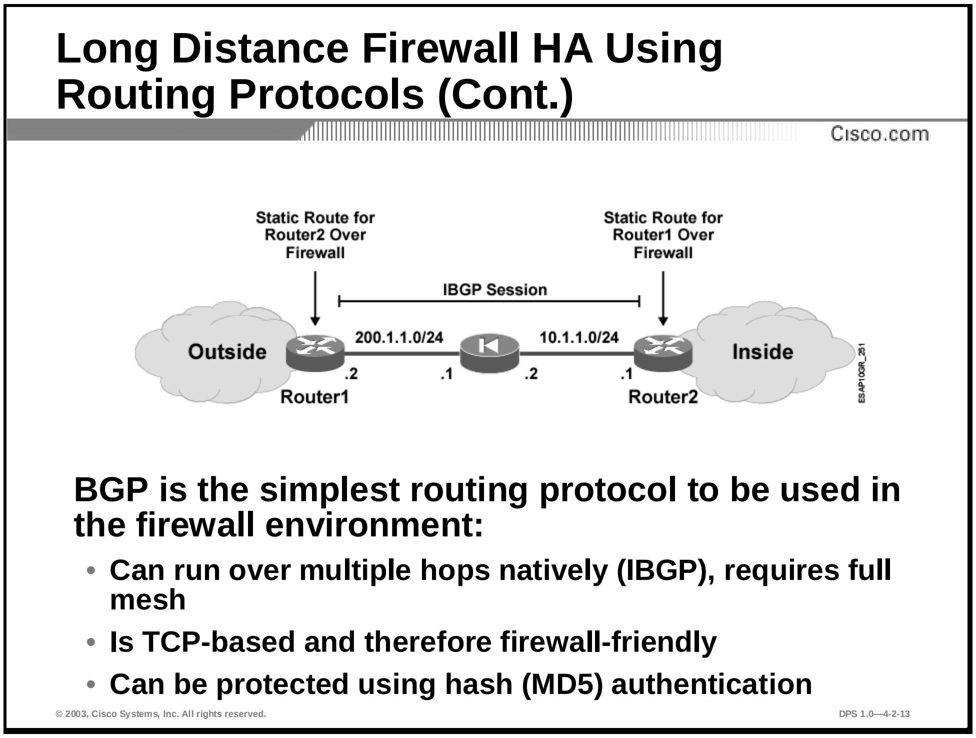
DPS 1.0—4-2-12

## Failover Routing Issues

The long distance high availability (HA) routing solutions require running a routing protocol over a firewall. The firewall does not need to speak to the routing protocol but it must be aware of it. Using BGP provides maximum flexibility because the configuration is independent from technical metrics. Alternatively, use interior gateway routing protocols such as OSPF and RIPv2. This availability relies on correct routing tables. Ensure the routing protocol uses an authentication mechanism when routing updates are sent.

Generally, the routing solution works with any firewall. Bidirectional Network Address Translation (NAT) is necessary, which is inside NAT to provide symmetric routing for return traffic of outbound connections, and outside NAT to provide symmetric routing for return traffic of inbound connections.

**Note** Do not use generic routing encapsulation (GRE) to run a routing protocol over a firewall, as traffic would follow the routing protocol into the tunnel, effectively bypassing all firewall controls.



## Long Distance Firewall HA Using Routing Protocols (Cont.)

Cisco.com

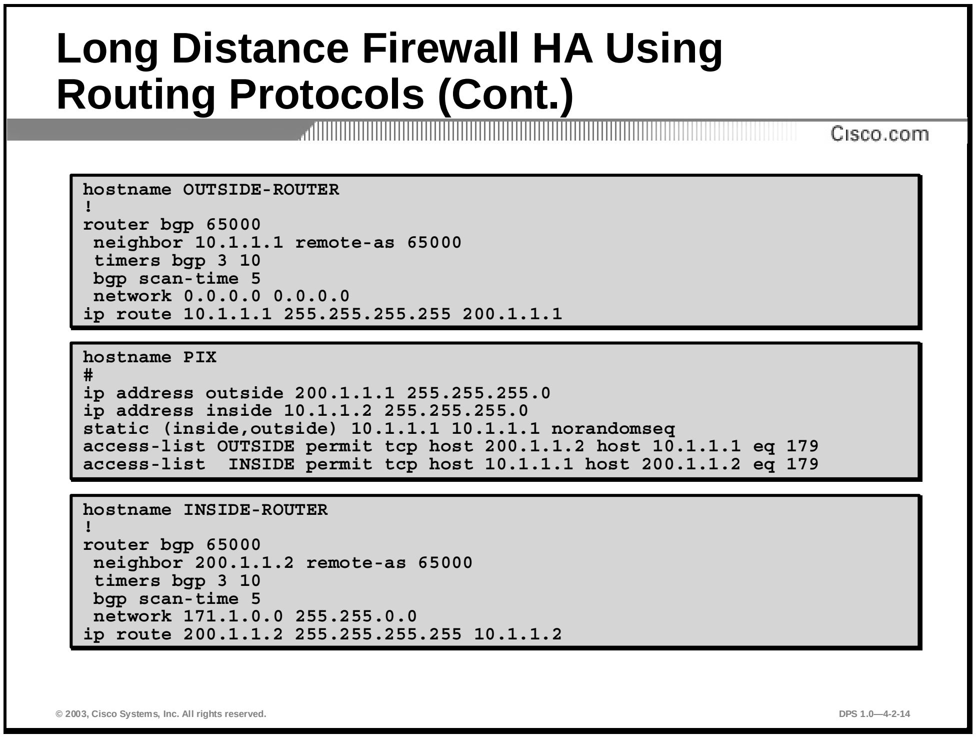
**BGP is the simplest routing protocol to be used in the firewall environment:**

- **Can run over multiple hops natively (IBGP), requires full mesh**
- **Is TCP-based and therefore firewall-friendly**
- **Can be protected using hash (MD5) authentication**

DPS 1.0—4-2-13

## Using BGP for Firewall Failover

Use BGP as the routing protocol due to its simple configuration using policies instead of technical metrics. The connection to the outside network over the firewall can be made using Internal BGP (IBGP), creating static routes in both directions. BGP is transported via TCP and is inherently firewall friendly. Furthermore, BGP provides Message Digest 5 (MD5) authentication, so the BGP updates can be trusted.



## Long Distance Firewall HA Using Routing Protocols (Cont.)

```
hostname OUTSIDE-ROUTER
!
router bgp 65000
 neighbor 10.1.1.1 remote-as 65000
 timers bgp 3 10
 bgp scan-time 5
 network 0.0.0.0 0.0.0.0
ip route 10.1.1.1 255.255.255.255 200.1.1.1
```

```
hostname PIX
#
ip address outside 200.1.1.1 255.255.255.0
ip address inside 10.1.1.2 255.255.255.0
static (inside,outside) 10.1.1.1 10.1.1.1 norandomseq
access-list OUTSIDE permit tcp host 200.1.1.2 host 10.1.1.1 eq 179
access-list  INSIDE permit tcp host 10.1.1.1 host 200.1.1.2 eq 179
```

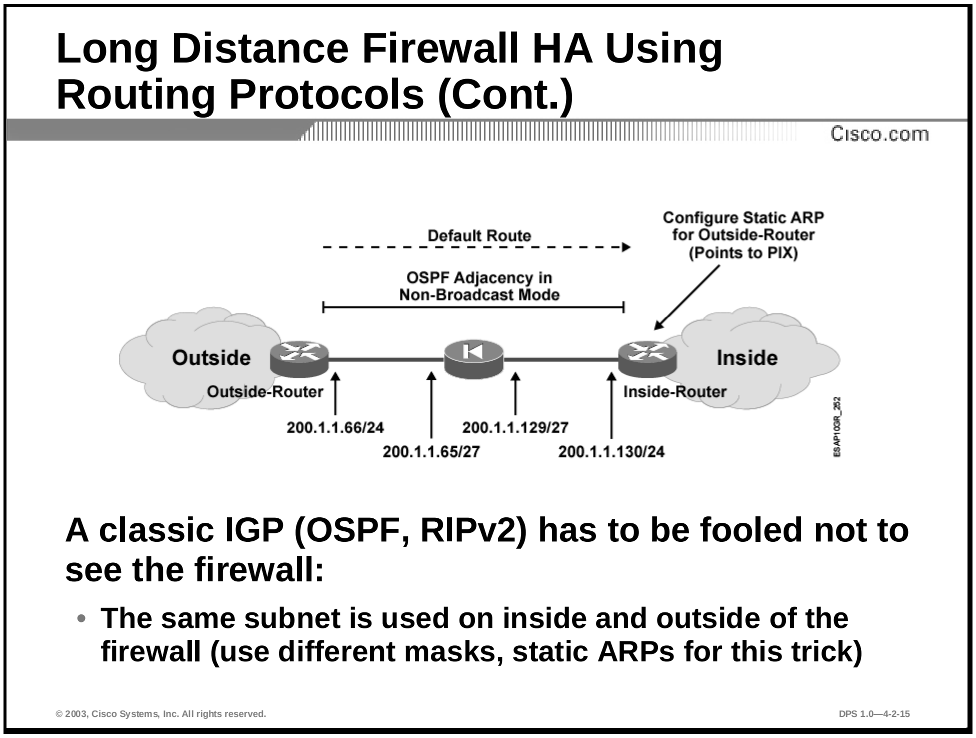
```
hostname INSIDE-ROUTER
!
router bgp 65000
 neighbor 200.1.1.2 remote-as 65000
 timers bgp 3 10
 bgp scan-time 5
 network 171.1.0.0 255.255.0.0
ip route 200.1.1.2 255.255.255.255 10.1.1.2
```

DPS 1.0—4-2-14

## Configuration Example

BGP exchange over firewalls requires a full IBGP mesh between all the BGP speakers. This example only illustrates a configuration to pass a single BGP session over a PIX Firewall. In reality, each site would have two routers on each side of the firewall, and all routers on both sites need to peer using iBGP. This results in six iBGP sessions that need to be configured and permitted over the firewalls.

**Note** Transporting BGP with authentication over the PIX Firewall requires that the firewall does not change anything in the BGP's IP packets, as they are fully authenticated. This requires an identity translation for the inside BGP peers, and the disabling of sequence number randomization in the PIX' static command.



## Long Distance Firewall HA Using Routing Protocols (Cont.)

Cisco.com

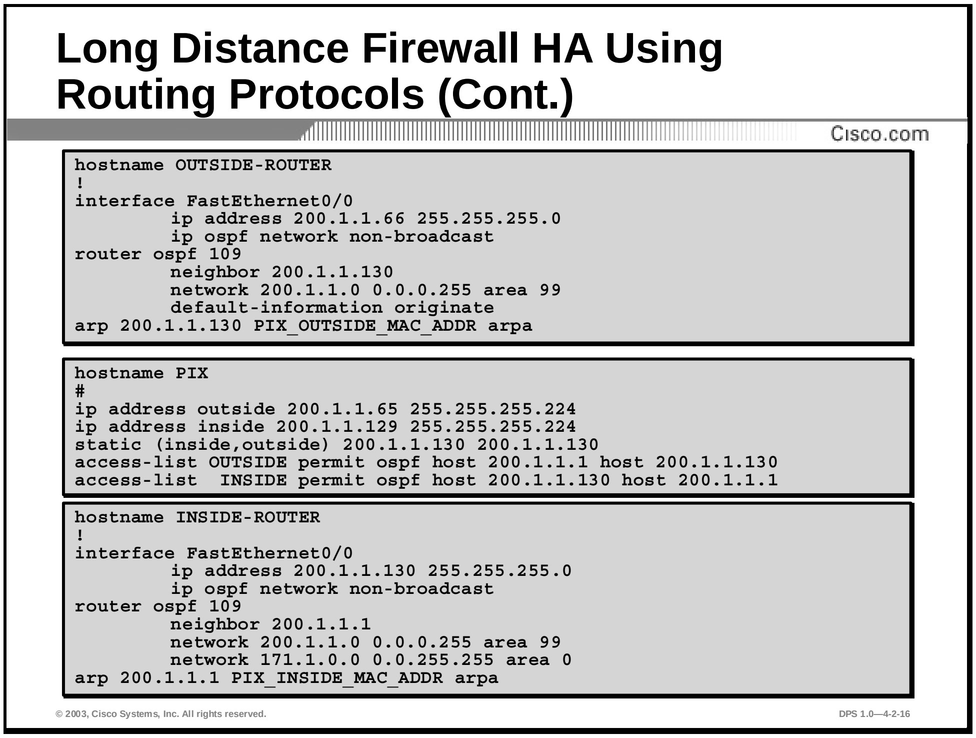
**A classic IGP (OSPF, RIPv2) has to be fooled not to see the firewall:**

- **The same subnet is used on inside and outside of the firewall (use different masks, static ARPs for this trick)**

DPS 1.0—4-2-15

## IGP Solutions

The problem with classic interior gateway routing protocols is that they have to be fooled not to see the firewall; otherwise there is exchanged of reachability information. Achieve this by using the same subnet mask on the router interfaces on the inside and outside of the firewall. Additionally, configure the static ARP entries to allow a router to reach the next hop, which in this example is the PIX.



## Long Distance Firewall HA Using Routing Protocols (Cont.)

Cisco.com

```
hostname OUTSIDE-ROUTER
!
interface FastEthernet0/0
        ip address 200.1.1.66 255.255.255.0
        ip ospf network non-broadcast
router ospf 109
        neighbor 200.1.1.130
        network 200.1.1.0 0.0.0.255 area 99
        default-information originate
arp 200.1.1.130 PIX_OUTSIDE_MAC_ADDR arpa
```

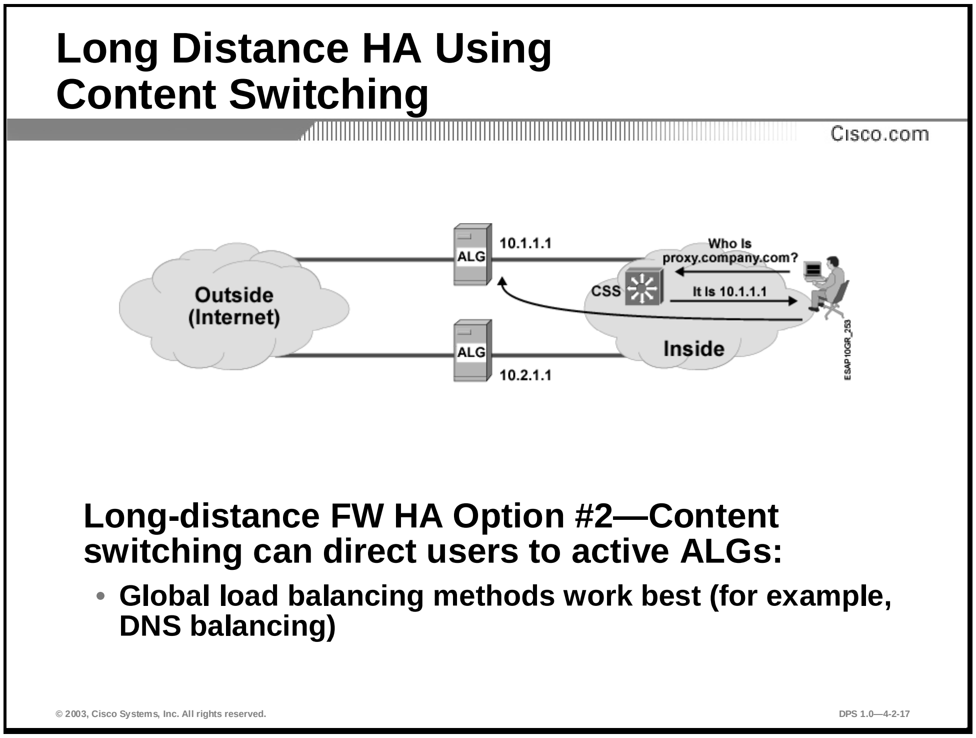
```
hostname PIX
#
ip address outside 200.1.1.65 255.255.255.224
ip address inside 200.1.1.129 255.255.255.224
static (inside,outside) 200.1.1.130 200.1.1.130
access-list OUTSIDE permit ospf host 200.1.1.1 host 200.1.1.130
access-list  INSIDE permit ospf host 200.1.1.130 host 200.1.1.1
```

```
hostname INSIDE-ROUTER
!
interface FastEthernet0/0
        ip address 200.1.1.130 255.255.255.0
        ip ospf network non-broadcast
router ospf 109
        neighbor 200.1.1.1
        network 200.1.1.0 0.0.0.255 area 99
        network 171.1.0.0 0.0.255.255 area 0
arp 200.1.1.1 PIX_INSIDE_MAC_ADDR arpa
```

DPS 1.0—4-2-16

## Configuration Example

This figure illustrates a configuration example for a redundant high availability long distance firewall solution using the routing protocol, Open Shortest Path First (OSPF). Note the static ARP entries needed to reach the next-hop interfaces.



Outside
(Internet)
ALG
10.1.1.1
ALG
10.2.1.1
CSS
Who Is
proxy.company.com?
It Is 10.1.1.1
Inside
ESAP10GR_253

## Long Distance HA Using Content Switching

Cisco.com

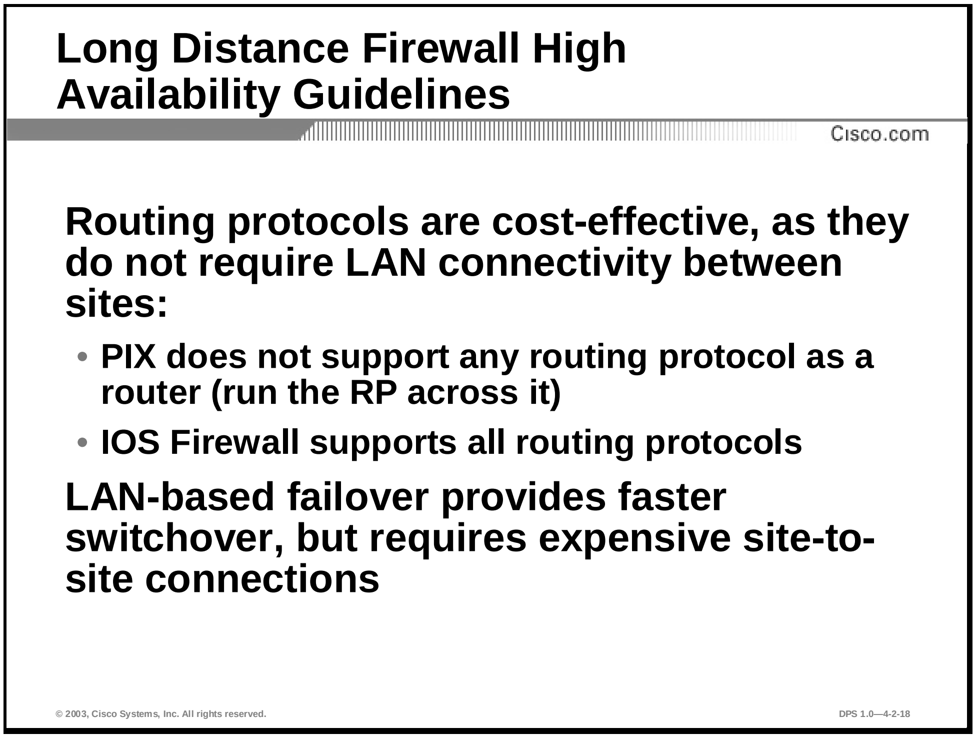
**Long-distance FW HA Option #2—Content switching can direct users to active ALGs:**

- **Global load balancing methods work best (for example, DNS balancing)**

DPS 1.0—4-2-17

## Content Switching for ALGs

When implementing long-distance high availability for ALGs, use the content switching method. Content switching directs user traffic to the current active ALG. Additionally, content switching provides global load balancing, for example DNS balancing.



## Long Distance Firewall High Availability Guidelines

Cisco.com

**Routing protocols are cost-effective, as they do not require LAN connectivity between sites:**

- **PIX does not support any routing protocol as a router (run the RP across it)**
- **IOS Firewall supports all routing protocols**

**LAN-based failover provides faster switchover, but requires expensive site-to-site connections**

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—4-2-18

## Guidelines for Long Distance High Availability

When designing a high availability firewall setup using long distance failover connections, the best solution is to use routing protocols for the failover connections. This method is the most cost-effective solution as there is no expensive LAN connectivity required between the sites.

One issue to consider is that the PIX does not support any routing protocol, so the routing protocol of the network border routers must run over the PIX. A simple solution is to use BGP, which is based on TCP. However, the designer can also use IGPs, which require the same sub-network numbers inside and outside of the PIX. An IOS based firewall supports all routing protocols, so configuration is straightforward.

Switched LAN-based solutions are faster, but due to the dedicated site-to-site connections, this method is much more expensive. Also, the total availability might be higher because the routing solution is more complex from a logical point of view and strongly dependent of consistent routing states.

## Practice

Q1) Which routing protocol is most suitable for long-distance firewall failover using routing protocols?

A) OSPF

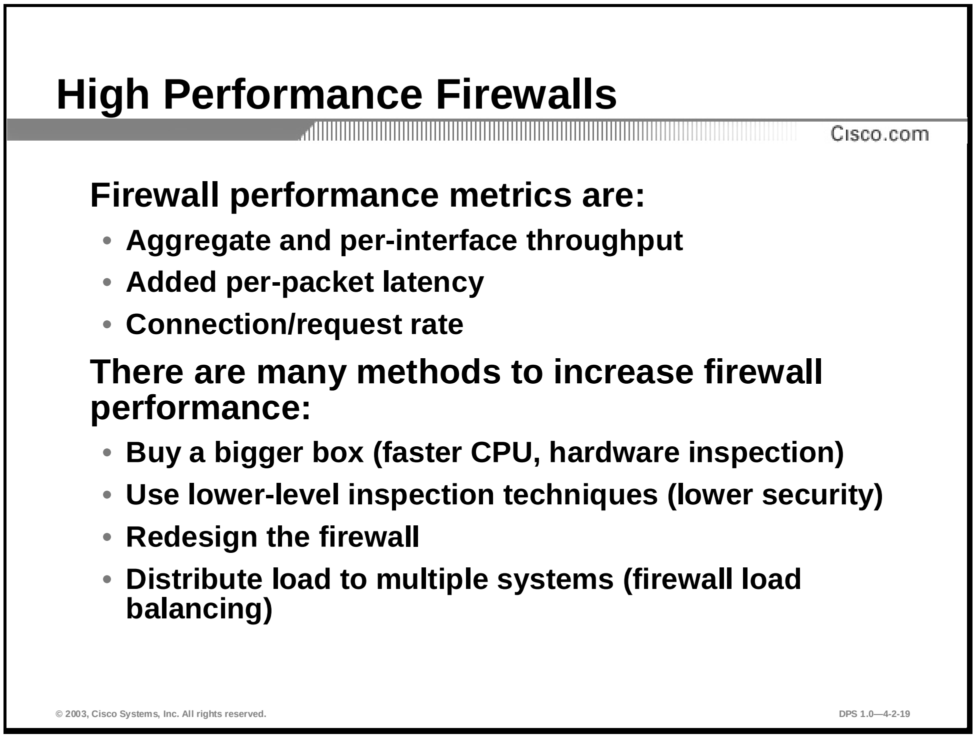
B) RIP

C) RIPv2

D) BGP

E) EIGRP

# High Performance Firewalls



## High Performance Firewalls

Cisco.com

**Firewall performance metrics are:**

- Aggregate and per-interface throughput
- Added per-packet latency
- Connection/request rate

**There are many methods to increase firewall performance:**

- Buy a bigger box (faster CPU, hardware inspection)
- Use lower-level inspection techniques (lower security)
- Redesign the firewall
- Distribute load to multiple systems (firewall load balancing)

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—4-2-19

## Objective

The section will enable the learner to identify solutions for building high-performance single firewalls and choose them in appropriate design situations.

## Introduction

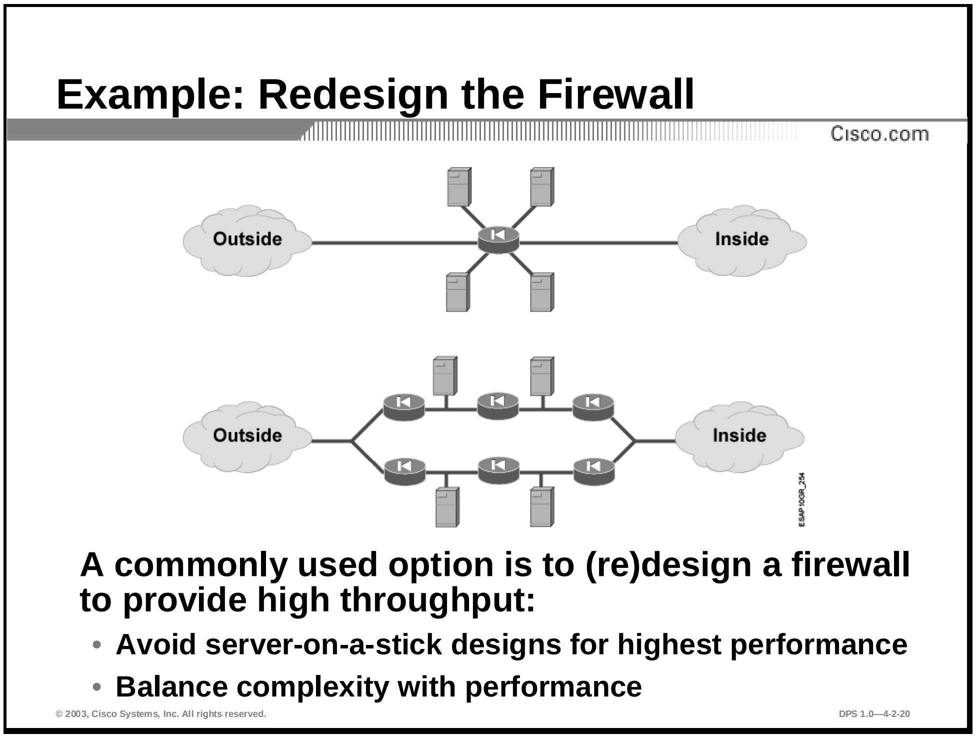
Because firewalls represent a network’s entry and exit point, all user traffic is concentrated there. It is very important that these points do not appear as bottlenecks. This section presents both firewall performance metrics and firewall tuning methods.

## Performance Metrics

A firewall’s performance can be determined by defining metrics such as the aggregate and per-interface throughput, the added per-packet latency, and the connection-request rate. The aggregate and per-interface throughput determines whether the firewall can be considered a bottleneck. The added per-packet delay is critical for real-time traffic, such as Voice over IP (VoIP) and videoconferencing. Firewall resources also limit the connection/request rate, so over-provisioning might be economically justified.

## Firewall Performance Improvements

The simplest, but most expensive performance improvement, is buying a bigger box to provide a faster CPU and a hardware-based inspection engine. A designer can achieve performance improvement by simply using lower layer inspection techniques, which work much faster, but offers a slightly lower degree of security. A redesign of the firewall may also help. The most elegant solution is to distribute the load over multiple firewalls. This can keep the performance and degree of security on a high level, but it may be the most expensive solution.



## Example: Redesign the Firewall

**A commonly used option is to (re)design a firewall to provide high throughput:**

- **Avoid server-on-a-stick designs for highest performance**
- **Balance complexity with performance**

DPS 1.0—4-2-20

## Redesign the Firewall

This example illustrates how total throughput improves by redesigning the firewall architecture. Avoid server-on-a-stick designs, and instead provide parallel paths if very high performance is desired. However, a designer must find a balanced point of efficiency so that the complexity does not dominate over performance.

## Practice

Q1) Which two techniques can improve firewall performance?

A) using generic routers instead of dedicated firewalls

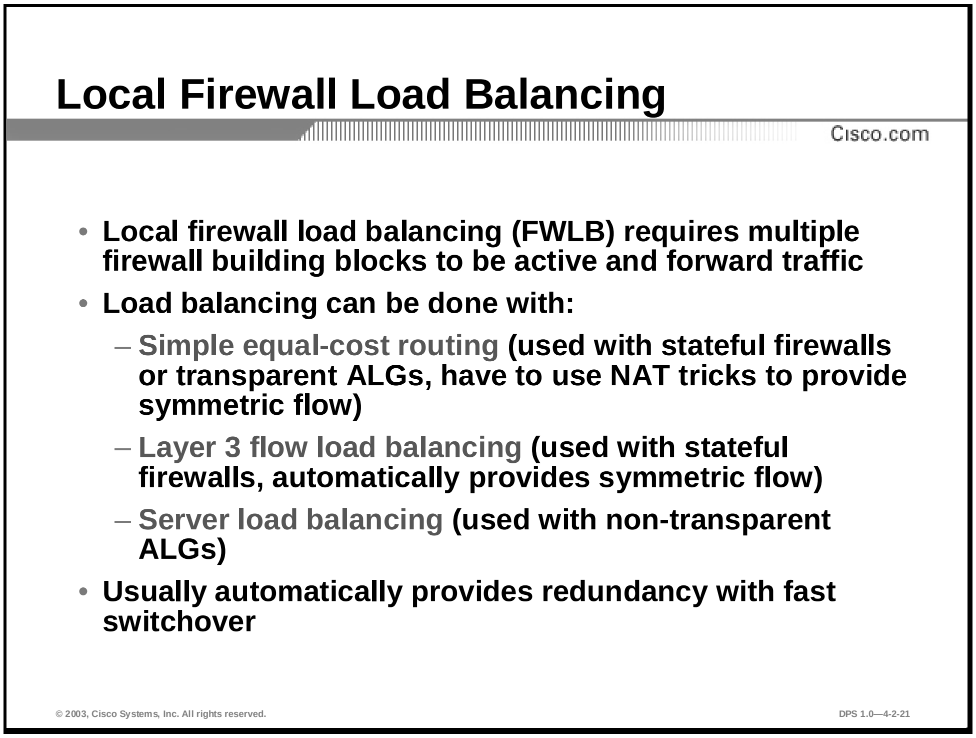
B) redesign of the firewall system

C) inspection at a higher OSI layer

D) faster firewall CPU

E) using routing protocols instead of static routing

# Local Firewall Load Balancing



## Local Firewall Load Balancing

- Local firewall load balancing (FWLB) requires multiple firewall building blocks to be active and forward traffic
- Load balancing can be done with:
  - Simple equal-cost routing (used with stateful firewalls or transparent ALGs, have to use NAT tricks to provide symmetric flow)
  - Layer 3 flow load balancing (used with stateful firewalls, automatically provides symmetric flow)
  - Server load balancing (used with non-transparent ALGs)
- Usually automatically provides redundancy with fast switchover

DPS 1.0—4-2-21

## Objective

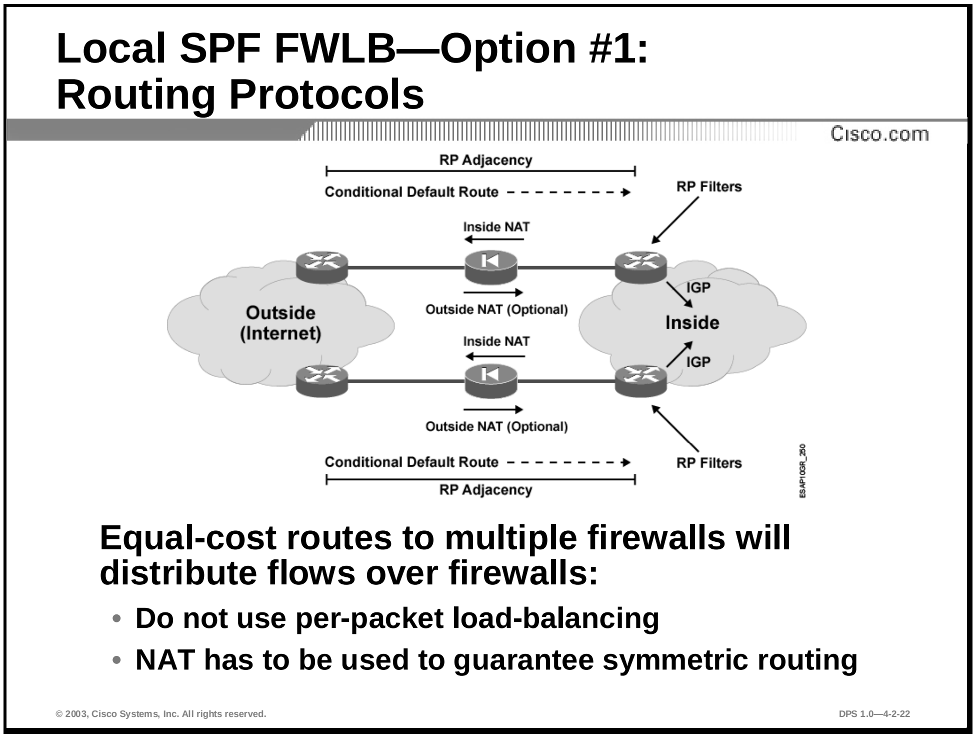
The section will enable the learner to identify local load balancing as a method for building high-performance firewalls and choose it in appropriate design situations.

## Introduction

Local firewall load balancing (FWLB) is a powerful technique, which can provide both high performance and fast switchover upon failures. This section presents important examples used in current high-performance, high-availability enterprise networks.

## Load Balancing Methods

Load balancing requires multiple firewall building blocks to be active at the same time to forward traffic. It uses simple equal-cost routing with stateful firewalls or transparent ALGs. Configure this method using bidirectional NAT tricks to provide a symmetric traffic flow. Use Layer 3 (L3) flow load balancing with stateful firewalls. This method automatically provides a symmetric traffic flow. Finally, server load balancing is a method used with non-transparent ALGs. Typically, all methods allow a fast switchover upon failures.



## Local SPF FWLB—Option #1: Routing Protocols

**Equal-cost routes to multiple firewalls will distribute flows over firewalls:**

- **Do not use per-packet load-balancing**
- **NAT has to be used to guarantee symmetric routing**

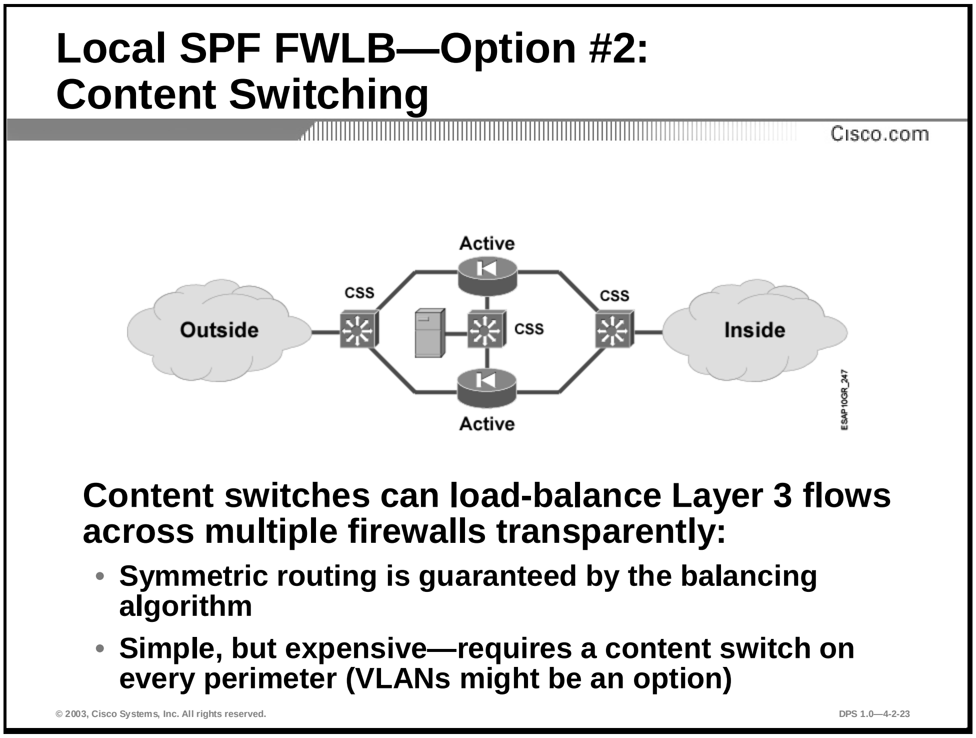
DPS 1.0—4-2-22

## Option #1: Routing Protocols

This FWLB option uses equal-cost routes to multiple firewalls that distribute the flows equally over the firewalls. The inbound load balancing requires bidirectional NAT.

**Note** This now hides the outside client addresses from the servers, which might be unacceptable for logging purposes.

**Caution** Do not use per-packet load balancing! Stateful filters need to see all packets of a session, and drop packets otherwise.



## Local SPF FWLB—Option #2: Content Switching

Cisco.com

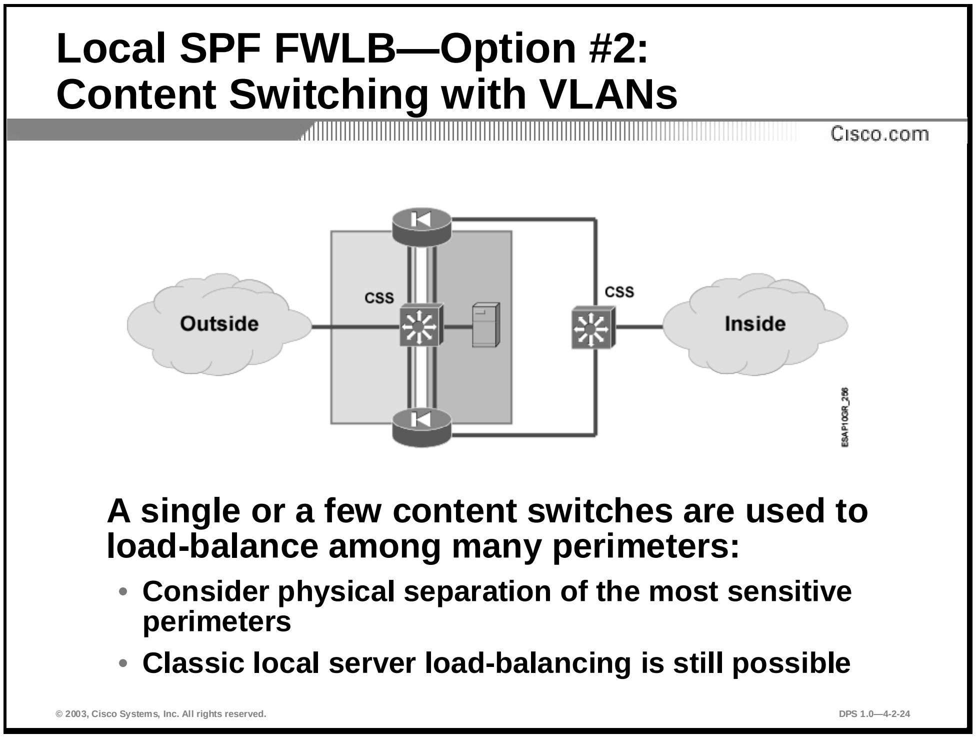
**Content switches can load-balance Layer 3 flows across multiple firewalls transparently:**

- **Symmetric routing is guaranteed by the balancing algorithm**
- **Simple, but expensive—requires a content switch on every perimeter (VLANs might be an option)**

DPS 1.0—4-2-23

## Option #2: Content Switching

Content switches automatically perform transparent load balancing of L3 flows across multiple firewalls. The balancing algorithm guarantees the symmetric routing. This solution is simple but expensive, as it requires a content switch on every perimeter. An additional solution could be using VLANs.



## Local SPF FWLB—Option #2: Content Switching with VLANs

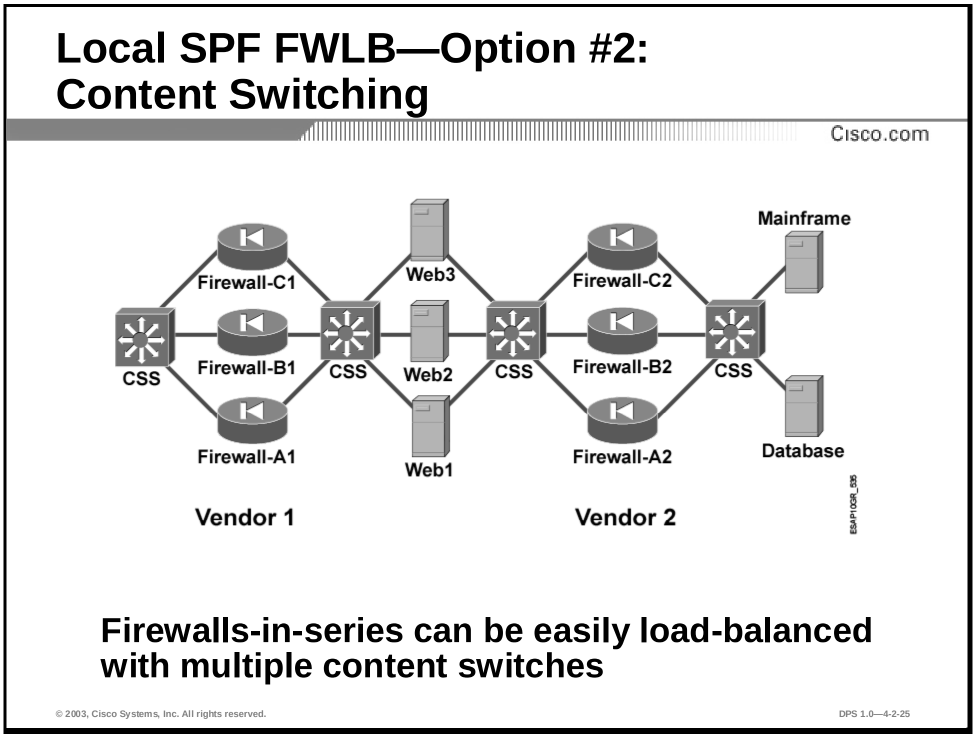
**A single or a few content switches are used to load-balance among many perimeters:**

- **Consider physical separation of the most sensitive perimeters**
- **Classic local server load-balancing is still possible**

DPS 1.0—4-2-24

## Content Switching with VLANs

When using VLANs, only a single or a few content switches are necessary to provide load balancing among many perimeters. In this situation, it is important to consider the physical separation of the most sensitive perimeters. Classic local server load balancing is still possible.



Local SPF FWLB—Option #2: Content Switching

Cisco.com

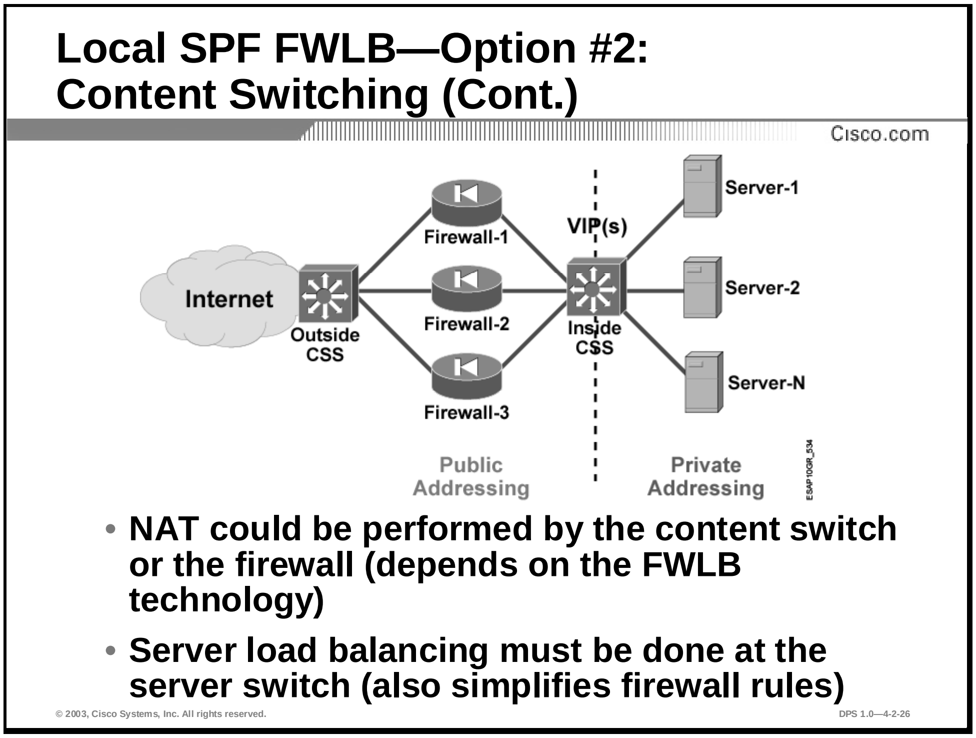
Firewalls-in-series can be easily load-balanced with multiple content switches

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—4-2-25

## Maximum Redundancy Using Content Switching

This figure illustrates a highly redundant configuration introducing diversity by using firewalls from different vendors. This example illustrates the multiple firewalls in a series. Multiple content switches provide the load balancing.



## Local SPF FWLB—Option #2: Content Switching (Cont.)

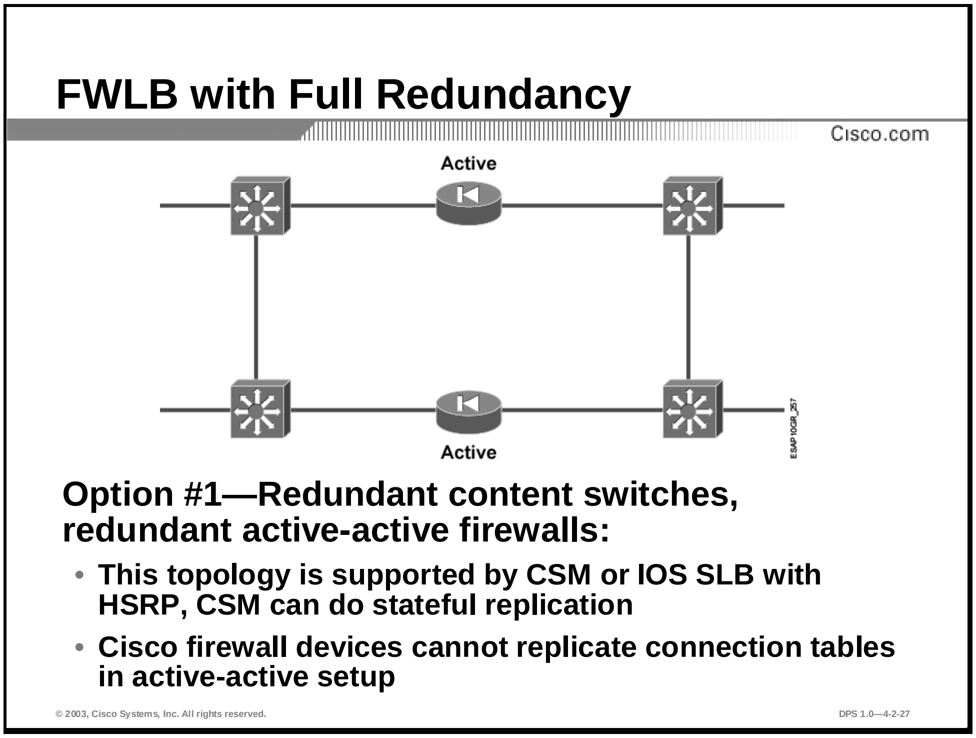
- NAT could be performed by the content switch or the firewall (depends on the FWLB technology)
- Server load balancing must be done at the server switch (also simplifies firewall rules)

DPS 1.0—4-2-26

## FWLB Details

Depending on the load balancing technology, the content switch or the firewall can perform NAT, therefore flexible architectures are possible.

The server load balancing must be done at the server switch, which also simplifies the firewall rules.



## FWLB with Full Redundancy

Cisco.com

**Option #1—Redundant content switches, redundant active-active firewalls:**

- **This topology is supported by CSM or IOS SLB with HSRP, CSM can do stateful replication**
- **Cisco firewall devices cannot replicate connection tables in active-active setup**

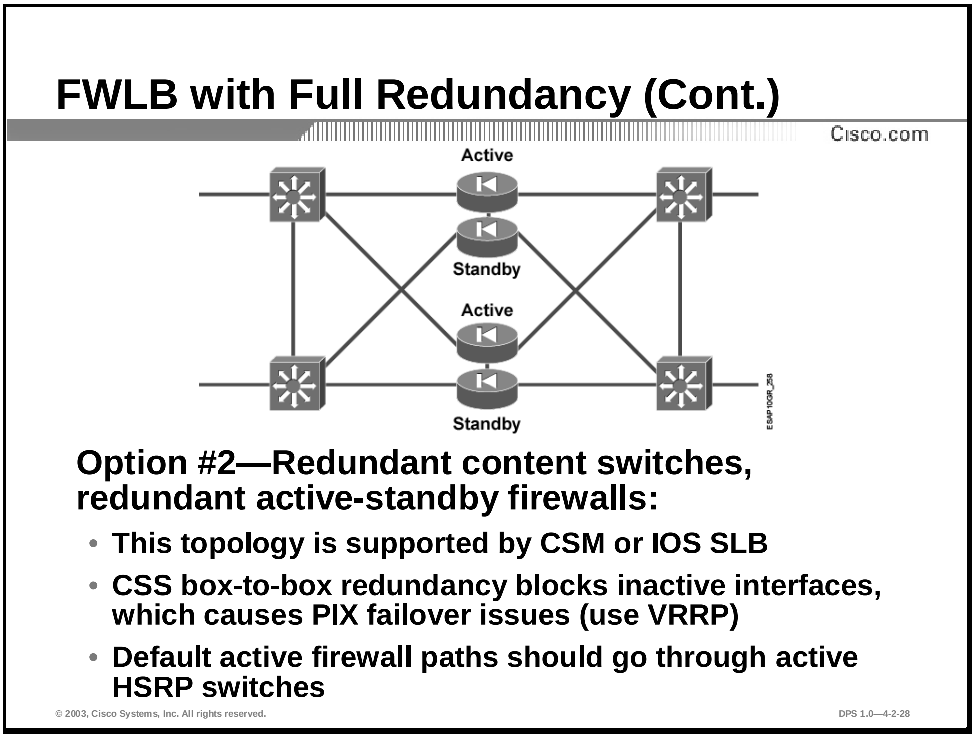
DPS 1.0—4-2-27

## Active-Active Firewalls

One possibility for designing FWLB with full redundancy is to employ both redundant content switches and redundant active firewalls.

The Content Switching Module (CSM) or IOS Server Load Balancing (SLB) with Hot Standby Router Protocol (HSRP) supports this topology. The Cisco CSM is a Catalyst 6500 line card that balances client traffic to farms of servers, firewalls, Secure Socket Layer (SSL) devices, or Virtual Private Network (VPN) termination devices. The CSM is able to track network sessions and server load conditions in real time, and directs each session to the most appropriate server. CSM is also able to perform stateful replication of traffic. SLB is a similar solution integrated in the IOS, and allows defining a virtual server that represents a cluster of real servers, known as a server farm.

| | |
|---|---|
| **Note** | Cisco firewall devices cannot replicate connection tables in active-active setups. |



## FWLB with Full Redundancy (Cont.)

Cisco.com

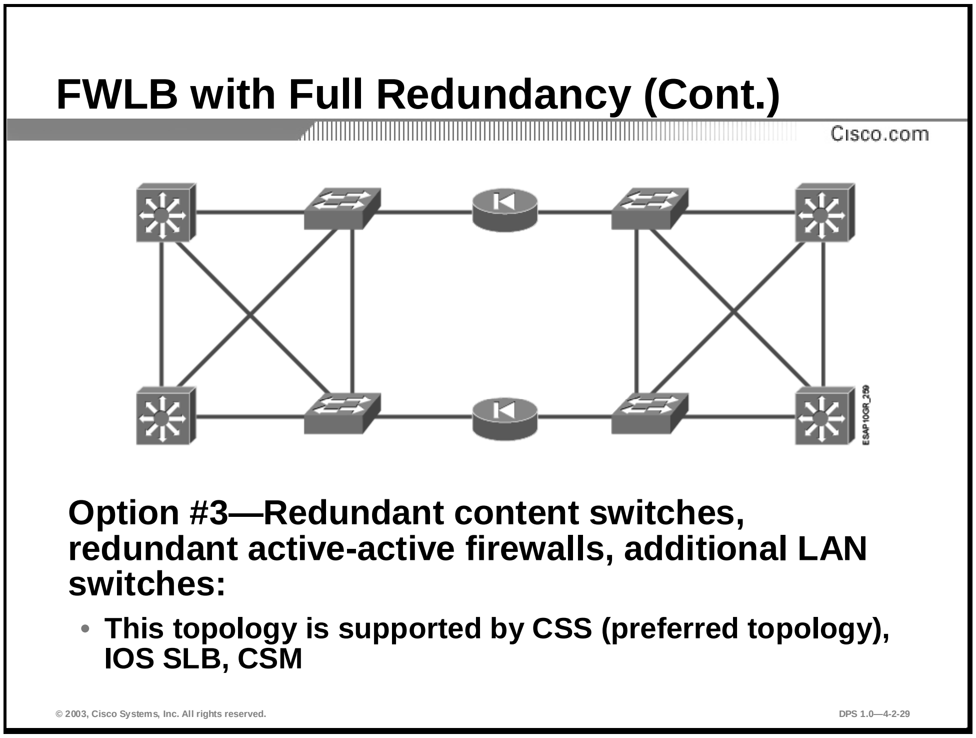
**Option #2—Redundant content switches, redundant active-standby firewalls:**

- **This topology is supported by CSM or IOS SLB**
- **CSS box-to-box redundancy blocks inactive interfaces, which causes PIX failover issues (use VRRP)**
- **Default active firewall paths should go through active HSRP switches**

DPS 1.0—4-2-28

## Active-Standby Firewalls

Another FWLB configuration using full redundancy consists of redundant content switches and redundant active-standby firewalls. Both CSM and IOS SLB support this topology. Using the Content Services Switch (CSS), Virtual Router Redundancy Protocol (VRRP) is necessary as CSS box-to-box redundancy blocks inactive interfaces, which causes PIX failover issues. All default active firewall paths should go through the active HSRP switches. This setup requires 4 PIXes (2 active, 2 standby) to be fully redundant.



## FWLB with Full Redundancy (Cont.)

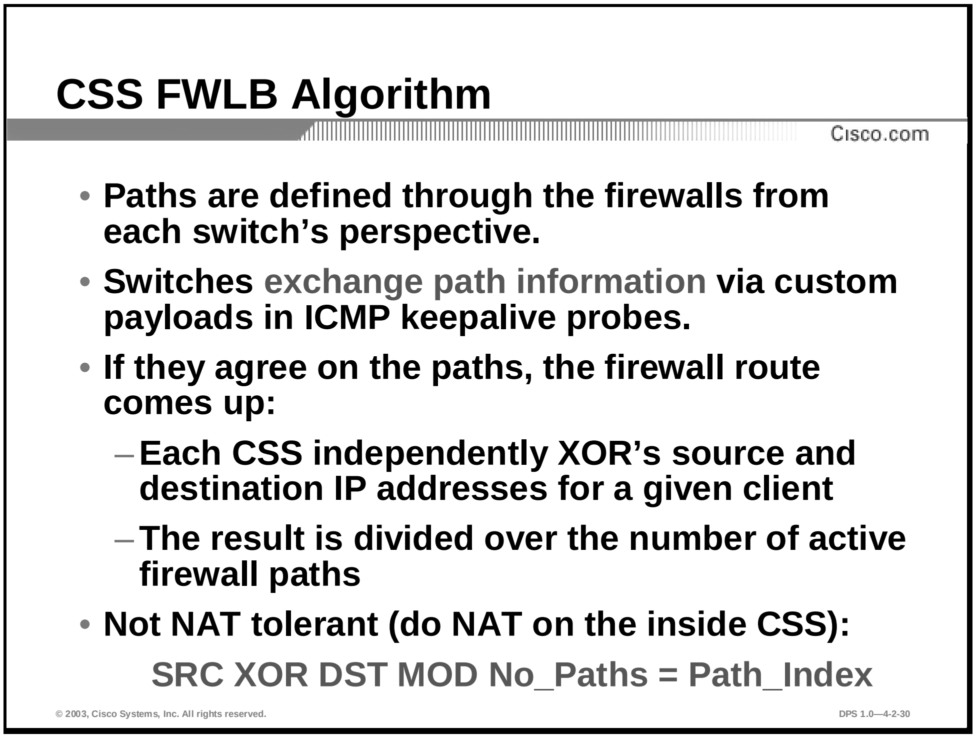
**Option #3—Redundant content switches, redundant active-active firewalls, additional LAN switches:**

- **This topology is supported by CSS (preferred topology), IOS SLB, CSM**

DPS 1.0—4-2-29

# Additional LAN Switches

Using additional LAN switches in a topology of redundant content switches and redundant active-active firewalls provides a solution supported by CSS, IOS SLB, and Content Switching Module (CSM). This is also the preferred topology when using CSS.



## CSS FWLB Algorithm

Cisco.com

- Paths are defined through the firewalls from each switch’s perspective.
- Switches exchange path information via custom payloads in ICMP keepalive probes.
- If they agree on the paths, the firewall route comes up:
  - Each CSS independently XOR’s source and destination IP addresses for a given client
  - The result is divided over the number of active firewall paths
- Not NAT tolerant (do NAT on the inside CSS):

SRC XOR DST MOD No_Paths = Path_Index

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—4-2-30

## How CSS Achieves FWLB

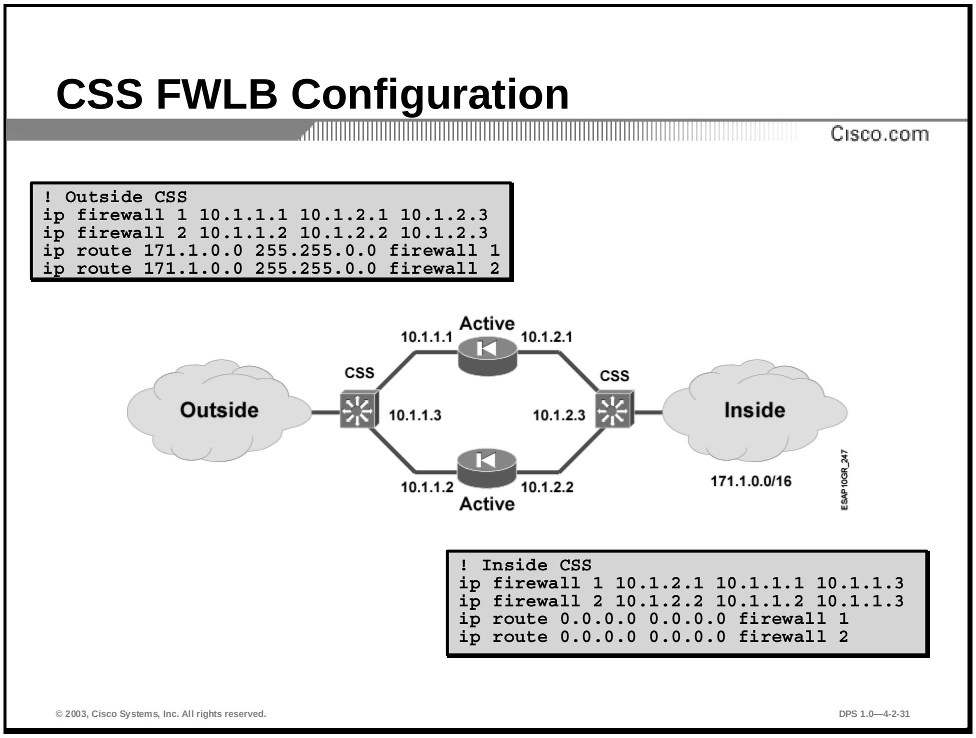
**Step 1** All paths are defined through the firewalls from each switch’s perspective.

**Step 2** The switches exchange path information via custom payloads in Internet Control Message Protocol (ICMP) keepalive probes.

**Step 3** If they agree on the paths, the firewall route opens.

**Step 4** Each CSS independently performs an XOR operation upon the source and destination IP addresses for a given client. The result is divided over the number of active firewall paths.

**Note** This configuration is not NAT tolerant. Configure NAT on the inside CSS instead.



## CSS FWLB Configuration

Cisco.com

```
! Outside CSS
ip firewall 1 10.1.1.1 10.1.2.1 10.1.2.3
ip firewall 2 10.1.1.2 10.1.2.2 10.1.2.3
ip route 171.1.0.0 255.255.0.0 firewall 1
ip route 171.1.0.0 255.255.0.0 firewall 2
```

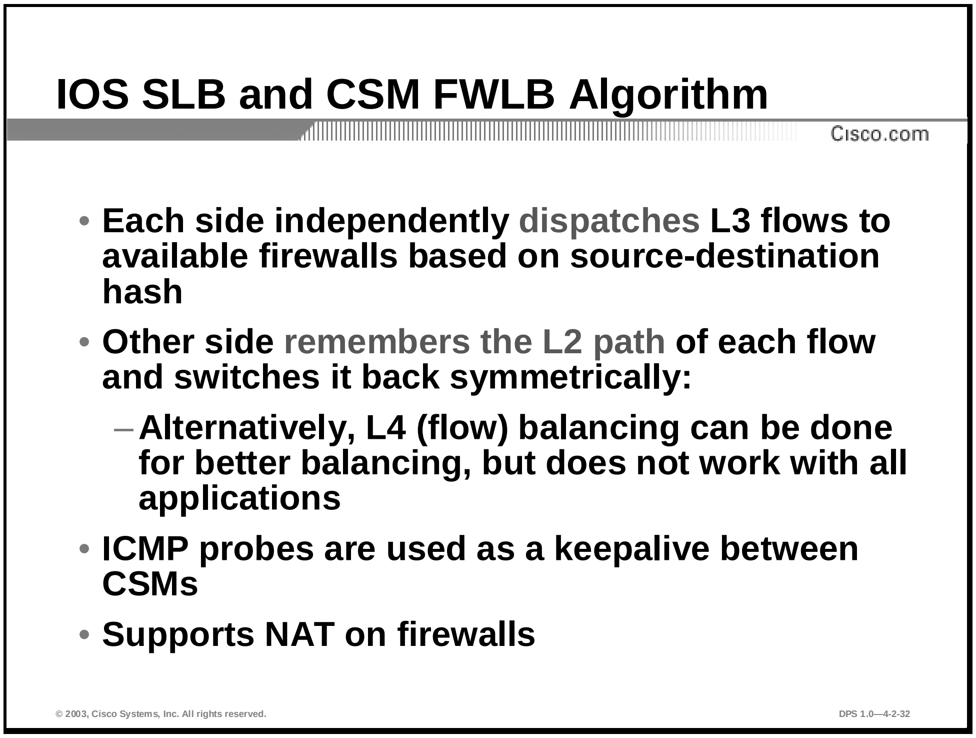
```
! Inside CSS
ip firewall 1 10.1.2.1 10.1.1.1 10.1.1.3
ip firewall 2 10.1.2.2 10.1.1.2 10.1.1.3
ip route 0.0.0.0 0.0.0.0 firewall 1
ip route 0.0.0.0 0.0.0.0 firewall 2
```

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—4-2-31

## Configuration Example

This figure illustrates a CSS FWLB configuration example. Both CSS systems have both firewalls defined, and two paths in the routing table indicating the firewall pair as the next hop. Load balancing is then performed automatically over all available paths using a source-destination hash.



## IOS SLB and CSM FWLB Algorithm

Cisco.com

- Each side independently dispatches L3 flows to available firewalls based on source-destination hash
- Other side remembers the L2 path of each flow and switches it back symmetrically:
  - Alternatively, L4 (flow) balancing can be done for better balancing, but does not work with all applications
- ICMP probes are used as a keepalive between CSMs
- Supports NAT on firewalls

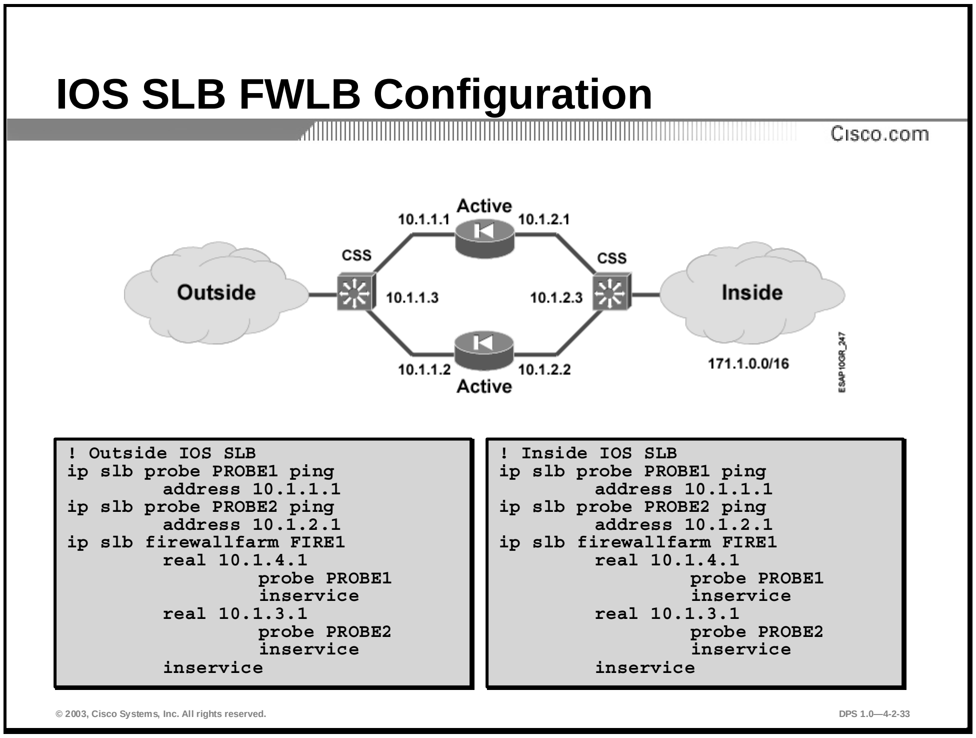
DPS 1.0—4-2-32

# How IOS SLB and CSM FWLB Works

**Step 1** Each side independently dispatches Layer 3 (L3) flows to the available firewalls using a source-destination hash to speed-up performance.

**Step 2** The other side remembers the Layer 2 (L2) path of each flow and switches it back symmetrically.

Alternatively, configure a Layer 4 (L4) flow balancing configured to achieve a better load balancing, but this does not work with all applications. Implement the keepalive procedure using ICMP probes between the CSMs. This solution also supports NAT on firewalls.



## IOS SLB FWLB Configuration

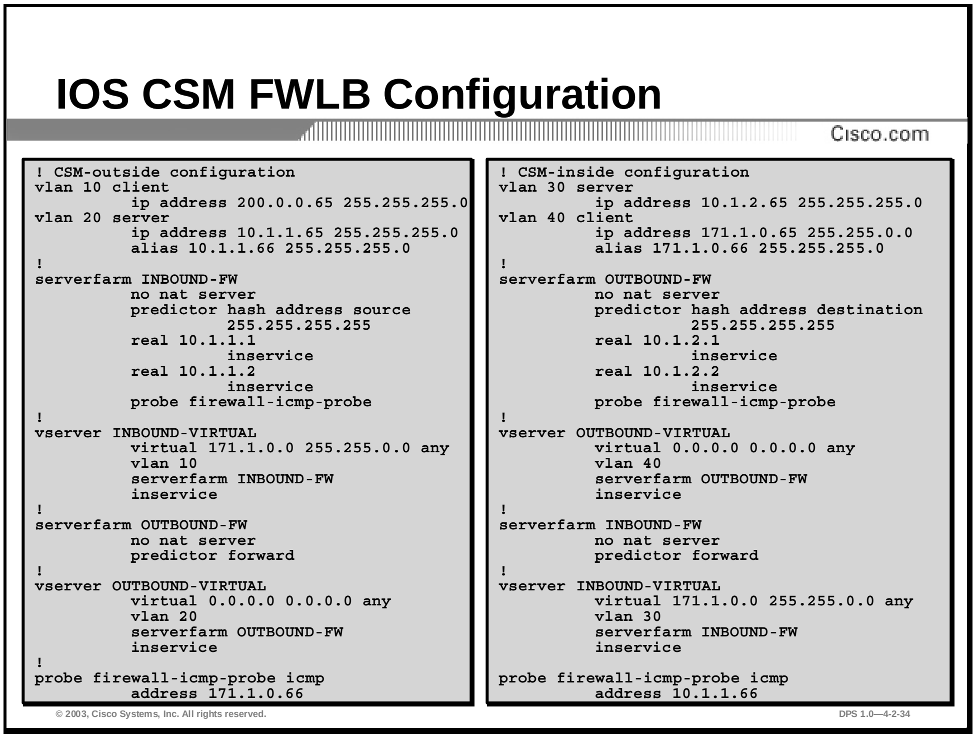
```
! Outside IOS SLB
ip slb probe PROBE1 ping
        address 10.1.1.1
ip slb probe PROBE2 ping
        address 10.1.2.1
ip slb firewallfarm FIRE1
        real 10.1.4.1
                probe PROBE1
                inservice
        real 10.1.3.1
                probe PROBE2
                inservice
        inservice
```

```
! Inside IOS SLB
ip slb probe PROBE1 ping
        address 10.1.1.1
ip slb probe PROBE2 ping
        address 10.1.2.1
ip slb firewallfarm FIRE1
        real 10.1.4.1
                probe PROBE1
                inservice
        real 10.1.3.1
                probe PROBE2
                inservice
        inservice
```

DPS 1.0—4-2-33

## Configuration Example

This figure illustrates an IOS SLB FWLB inside and outside configuration example. Both firewalls are defined as “real” servers, with ICMP (ping) probes verifying their health. The virtual servers, corresponding to route table entries, can then be defined to forward to the traffic to the “real” servers, i.e. firewalls.



## IOS CSM FWLB Configuration

```
! CSM-outside configuration
vlan 10 client
          ip address 200.0.0.65 255.255.255.0
vlan 20 server
          ip address 10.1.1.65 255.255.255.0
          alias 10.1.1.66 255.255.255.0
!
serverfarm INBOUND-FW
          no nat server
          predictor hash address source
                    255.255.255.255
          real 10.1.1.1
                    inservice
          real 10.1.1.2
                    inservice
          probe firewall-icmp-probe
!
vserver INBOUND-VIRTUAL
          virtual 171.1.0.0 255.255.0.0 any
          vlan 10
          serverfarm INBOUND-FW
          inservice
!
serverfarm OUTBOUND-FW
          no nat server
          predictor forward
!
vserver OUTBOUND-VIRTUAL
          virtual 0.0.0.0 0.0.0.0 any
          vlan 20
          serverfarm OUTBOUND-FW
          inservice
!
probe firewall-icmp-probe icmp
          address 171.1.0.66
```

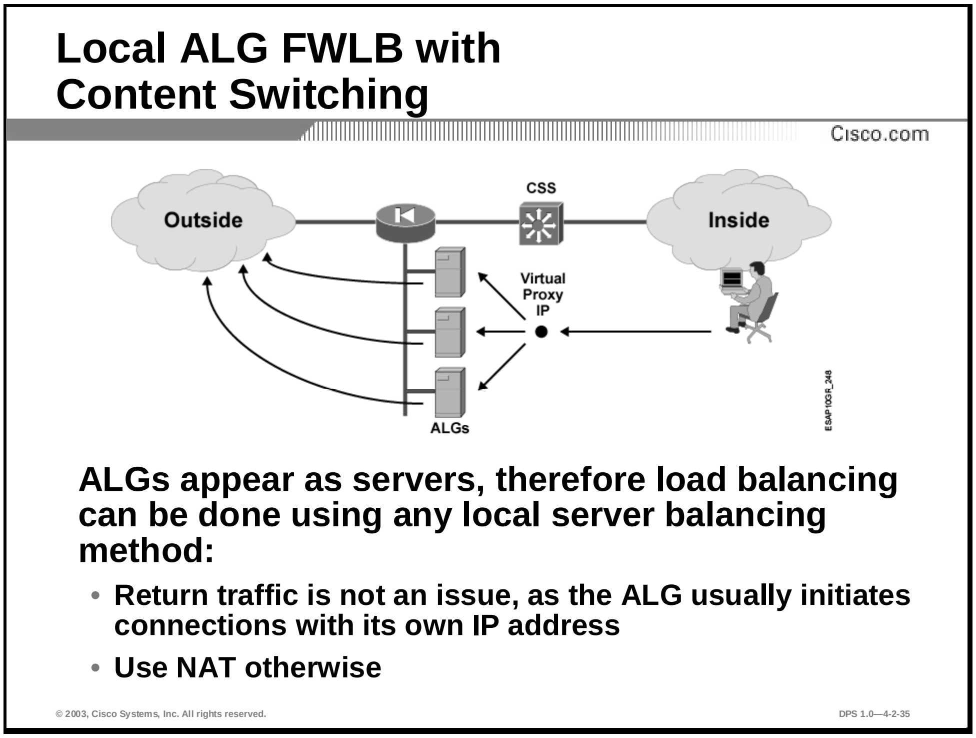
```
! CSM-inside configuration
vlan 30 server
          ip address 10.1.2.65 255.255.255.0
vlan 40 client
          ip address 171.1.0.65 255.255.0.0
          alias 171.1.0.66 255.255.255.0
!
serverfarm OUTBOUND-FW
          no nat server
          predictor hash address destination
                    255.255.255.255
          real 10.1.2.1
                    inservice
          real 10.1.2.2
                    inservice
          probe firewall-icmp-probe
!
vserver OUTBOUND-VIRTUAL
          virtual 0.0.0.0 0.0.0.0 any
          vlan 40
          serverfarm OUTBOUND-FW
          inservice
!
serverfarm INBOUND-FW
          no nat server
          predictor forward
!
vserver INBOUND-VIRTUAL
          virtual 171.1.0.0 255.255.0.0 any
          vlan 30
          serverfarm INBOUND-FW
          inservice

probe firewall-icmp-probe icmp
          address 10.1.1.66
```

DPS 1.0—4-2-34

## Configuration Example

This figure illustrates an IOS CSM FWLB inside and outside configuration example. Both firewalls are defined as “real” servers, with ICMP (ping) probes verifying their health. The virtual servers (0.0.0.0/0 and 171.1.0.0/16), corresponding to route table entries, are then defined to forward to the traffic to the “real” servers, i.e. firewalls.



## Local ALG FWLB with Content Switching

Cisco.com

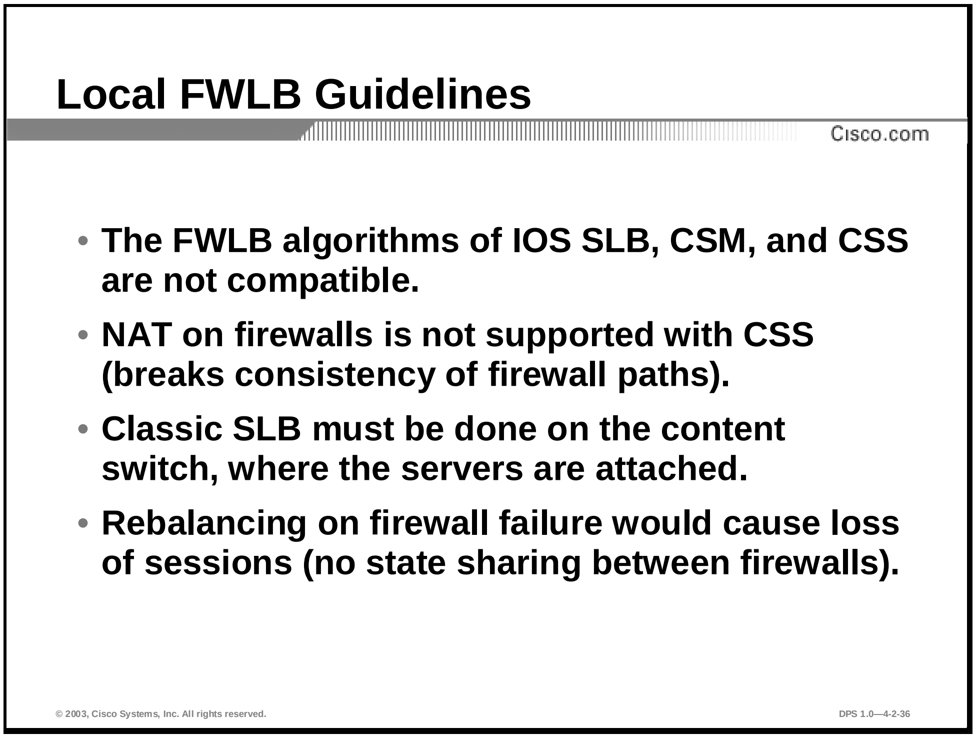
**ALGs appear as servers, therefore load balancing can be done using any local server balancing method:**

- **Return traffic is not an issue, as the ALG usually initiates connections with its own IP address**
- **Use NAT otherwise**

DPS 1.0—4-2-35

## ALG FWLB with Content Switching

ALGs appear as servers, therefore a load balancing solution can be implemented using any local server balancing method. Note that return traffic is not an issue because the ALG usually initiates connections with its own IP address.



## Local FWLB Guidelines

Cisco.com

- The FWLB algorithms of IOS SLB, CSM, and CSS are not compatible.
- NAT on firewalls is not supported with CSS (breaks consistency of firewall paths).
- Classic SLB must be done on the content switch, where the servers are attached.
- Rebalancing on firewall failure would cause loss of sessions (no state sharing between firewalls).

© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—4-2-36

## Guidelines for Designing Local FWLB

It is important to know that the FWLB algorithms of IOS SLB, CSM, and CSS are not compatible to each other. Additionally, CSS does not support NAT on firewalls as it breaks the consistency of firewall paths.

Where the servers attach, configure the classic SLB on the content switch—rebalancing on firewall failure would cause a loss of sessions.

**Note** There is no state sharing between firewalls.

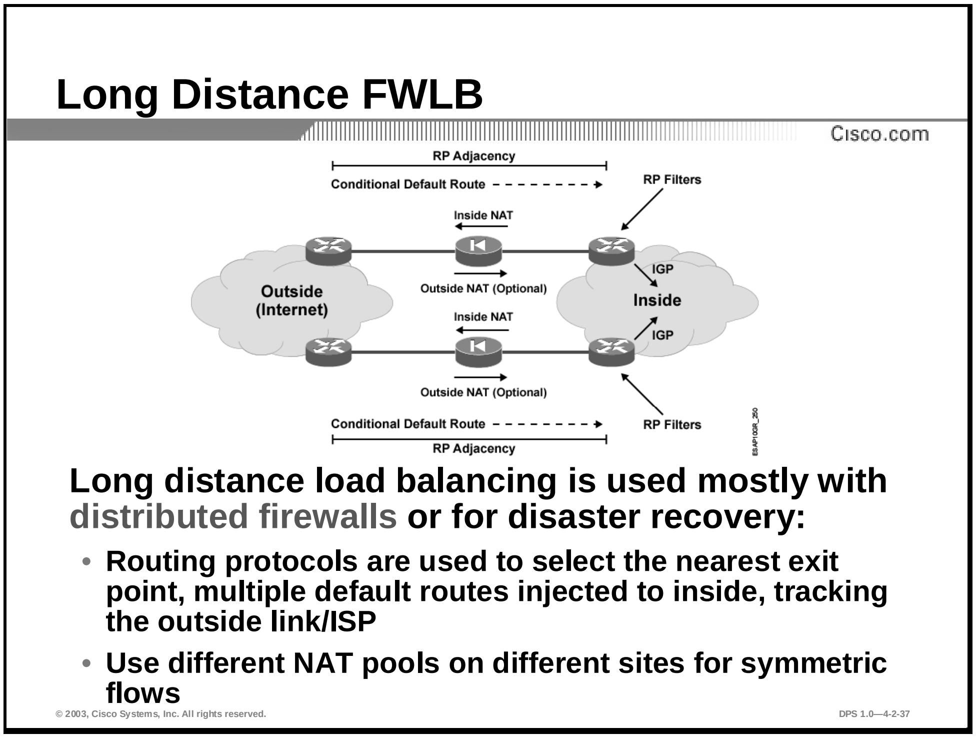
## Practice

Q1) True or false? The firewall load-balancing algorithms of IOS Server Load Balancing and the Content Services Switches are compatible.

A) true

B) false

# Remote Firewall Load Balancing



## Long Distance FWLB

**Long distance load balancing is used mostly with distributed firewalls or for disaster recovery:**

- **Routing protocols are used to select the nearest exit point, multiple default routes injected to inside, tracking the outside link/ISP**
- **Use different NAT pools on different sites for symmetric flows**

DPS 1.0—4-2-37

## Objective

The section will enable the learner to identify long-distance load balancing as a method for building high-performance firewalls and choose it in appropriate design situations.

## Introduction

Long distance load balancing is usually used mostly to achieve distributed firewalling (i.e. multiple Internet connections) or using the connections in an otherwise-standby disaster recovery center for boosting Internet performance.

## Using Distributed Firewalls for Long Distance FWLB

This configuration relies on routing protocols determining the nearest exit point, and for injection of multiple default routes into the inside network. It also performs a tracking of the outside link per Internet service provider (ISP).

Using different NAT pools on different sites can easily configure symmetric flow distribution.

Global server load balancing algorithms balance incoming connections to the servers. This is independent from firewall load balancing.

## Practice

Q1) How is symmetric routing guaranteed, if long distance firewall load-balancing is performed using routing protocols to select the best firewall?

A) using NAT

B) using dedicated load-balancing devices

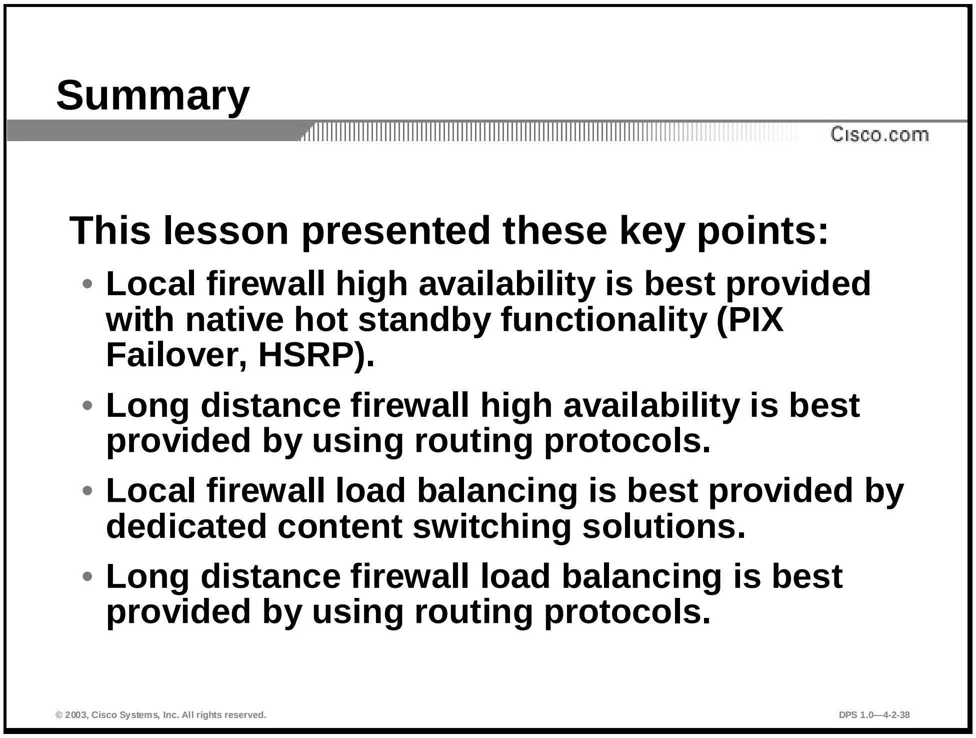
C) using content switching

D) using the routing protocol’s load balancing algorithm

E) using application-layer gateways only

# Summary

This section summarizes the key points discussed in this lesson.



## Summary

This lesson presented these key points:

- Local firewall high availability is best provided with native hot standby functionality (PIX Failover, HSRP).
- Long distance firewall high availability is best provided by using routing protocols.
- Local firewall load balancing is best provided by dedicated content switching solutions.
- Long distance firewall load balancing is best provided by using routing protocols.

DPS 1.0—4-2-38

## Next Steps

After completing this lesson, go to:

- High Availability and High Performance Setups lesson

# Quiz: High Availability and High Performance Firewalls

Complete the quiz to assess what you have learned in this lesson.

## Objectives

This quiz tests your knowledge on how to:

- Design a firewall system supporting high-availability and high levels of performance

## Instructions

Answer these questions:

1. How can remote firewall redundancy be provided?
2. Which routing protocols can run over firewalls?
3. Which significant limitation exists in CSS FWLB code?
4. What are the options to increase firewall performance?

## Scoring

You have successfully completed the quiz for this lesson when you earn a score of 80 percent or better.

# Understanding PIX Firewall NAT

## Overview

The PIX firewall will always translate inside hosts when they communicate with outside hosts. Different strategies can be chosen, depending on the needs and the specific network design

### Importance

NAT is a vital component for perimeter security and is often used in perimeter designs, interfacing a protected network with a global network.

### Lesson Objective

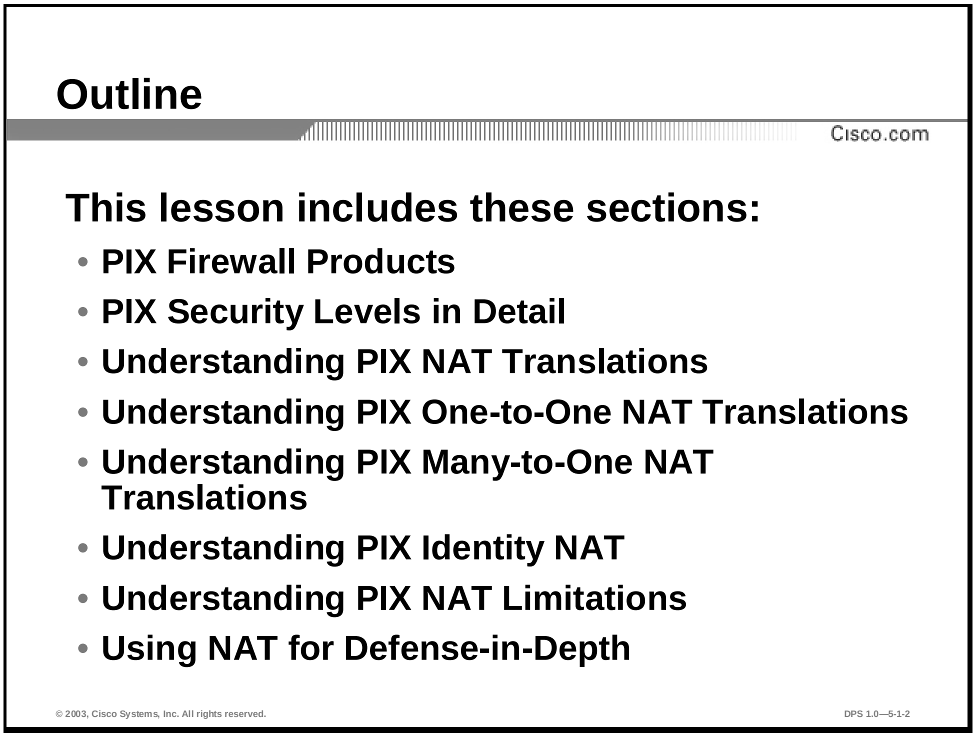
This lesson will enable the learner to identify and configure advanced NAT features, and identify NAT limitations of the Cisco Secure PIX Firewall product, when using it in a firewall system design.

## Learner Skills and Knowledge

To fully benefit from this lesson, you must have these prerequisite skills and knowledge:

- Understand the concepts, features, and limitations of NAT and dual NAT
- Understand the concept of a firewall

## Outline



### Outline

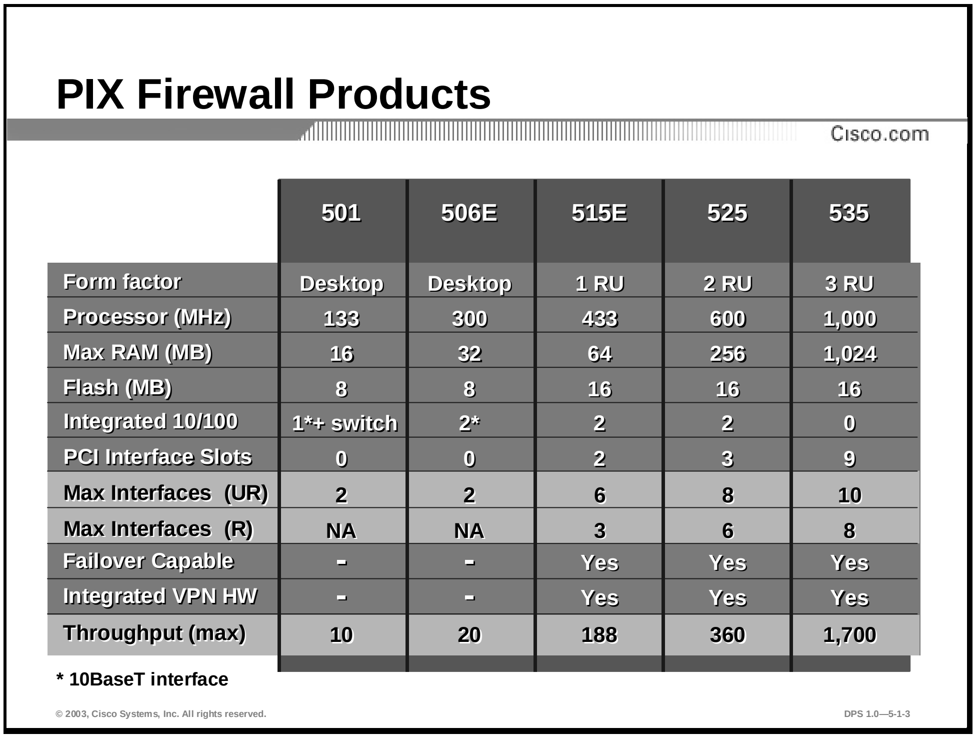
Cisco.com

**This lesson includes these sections:**

- **PIX Firewall Products**
- **PIX Security Levels in Detail**
- **Understanding PIX NAT Translations**
- **Understanding PIX One-to-One NAT Translations**
- **Understanding PIX Many-to-One NAT Translations**
- **Understanding PIX Identity NAT**
- **Understanding PIX NAT Limitations**
- **Using NAT for Defense-in-Depth**

DPS 1.0—5-1-2

# PIX Firewall Products



## PIX Firewall Products

Cisco.com

| | 501 | 506E | 515E | 525 | 535 |
|---|---|---|---|---|---|
| Form factor | Desktop | Desktop | 1 RU | 2 RU | 3 RU |
| Processor (MHz) | 133 | 300 | 433 | 600 | 1,000 |
| Max RAM (MB) | 16 | 32 | 64 | 256 | 1,024 |
| Flash (MB) | 8 | 8 | 16 | 16 | 16 |
| Integrated 10/100 | 1*+ switch | 2* | 2 | 2 | 0 |
| PCI Interface Slots | 0 | 0 | 2 | 3 | 9 |
| Max Interfaces (UR) | 2 | 2 | 6 | 8 | 10 |
| Max Interfaces (R) | NA | NA | 3 | 6 | 8 |
| Failover Capable | - | - | Yes | Yes | Yes |
| Integrated VPN HW | - | - | Yes | Yes | Yes |
| Throughput (max) | 10 | 20 | 188 | 360 | 1,700 |

* 10BaseT interface

DPS 1.0—5-1-3

## Objective

This lesson will enable the learner to identify Cisco Secure PIX Firewall products and select the appropriate model in a firewall design

## Introduction

The Cisco Secure PIX Firewall product line covers all possible uses for a stateful packet filtering firewall in an enterprise environment. Ranging from a low cost entry model for home office use to a high performance appliance designed for backbone implementations, all PIX models provide the same security and the same software features. When selecting a PIX model, the designer focuses on throughput and number of interfaces (perimeters), which are supported.

The specified throughput numbers are optimistic, based on large (1500-byte) packets and show the throughput for packet filtering only. Actual results with IMIX traffic might be substantially lower, depending on other features, which are enabled, such as IPSec tunnels, content filtering or user authentication with AAA-functions.

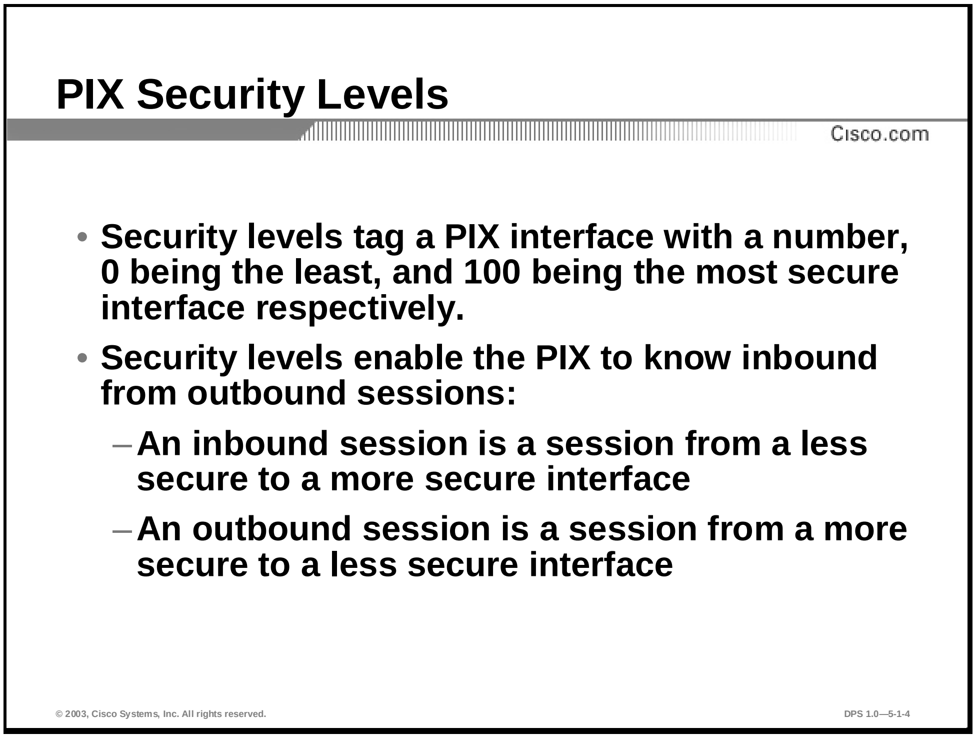
## Practice

Q1) True or false? The PIX Firewall 506 is failover-capable.

A) true

B) false

# PIX Security Levels in Detail



## PIX Security Levels

- Security levels tag a PIX interface with a number, 0 being the least, and 100 being the most secure interface respectively.
- Security levels enable the PIX to know inbound from outbound sessions:
  - An inbound session is a session from a less secure to a more secure interface
  - An outbound session is a session from a more secure to a less secure interface

DPS 1.0—5-1-4

## Objective

This section will enable the learner to explain how the concept of security level influences all aspects of access control in the PIX Firewall

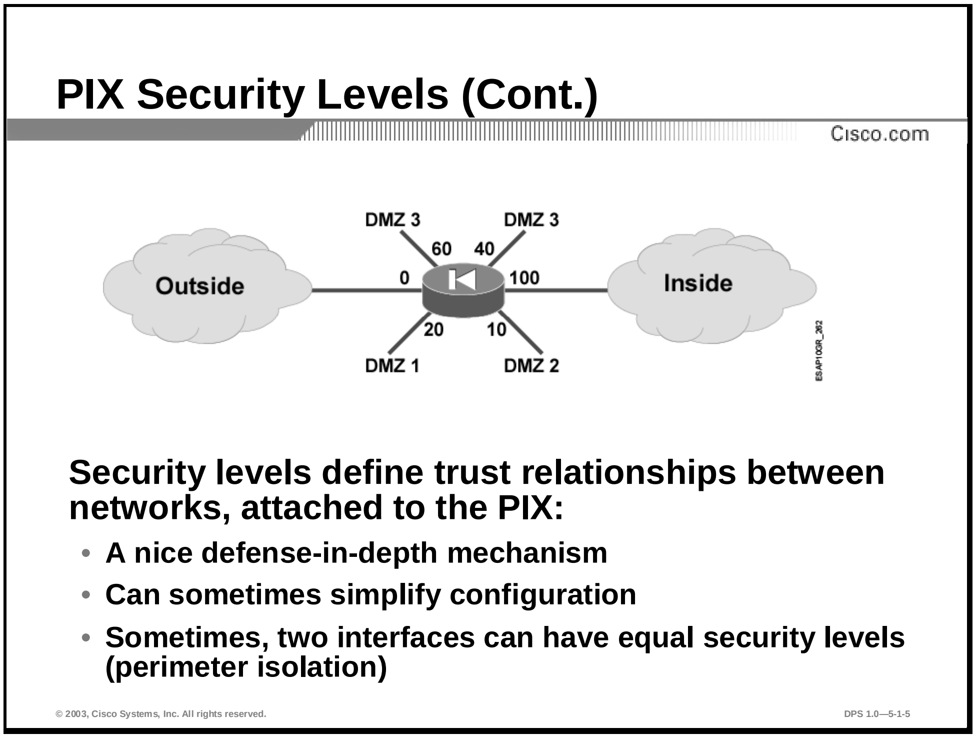
## Introduction

Security levels are the most important parameter for the PIX security algorithm. A low number denotes an “insecure” or “untrusted” network (e.g. the Internet), a high number will describe a “secure” or “trusted” network (e.g. the internal network). Security levels will also influence address translation.

**Inbound sessions** are connections from an interface with a low security level to an interface with a higher security level. These connections will preserve the source address and will translate the destination. Inbound sessions need explicit permissions and special address translations rules (e.g. static or outside address translation).

**Outbound sessions** are connections are originating on an interface with a higher security level than the destination interface. The source address will be translated according to the translations rules (dynamic, static, PAT, identity etc.) on these connections, while the destination address will be preserved. Outbound sessions are permitted by default.

Two interfaces with the same security level cannot communicate at all, because there is no way to determine address translation strategies. This feature can be used to effectively separate two interfaces from each other.



## PIX Security Levels (Cont.)

Cisco.com

**Security levels define trust relationships between networks, attached to the PIX:**

- **A nice defense-in-depth mechanism**
- **Can sometimes simplify configuration**
- **Sometimes, two interfaces can have equal security levels (perimeter isolation)**

DPS 1.0—5-1-5

This example shows a typical implementation of a PIX firewall with multiple DMZs having different security levels. Note, that the inside interface has the highest and the outside interface has the lowest security level. The default policy provides nice defense-in-depth, because even if all access rules are deleted, a sensible policy applies, permitting access from a trusted to an untrusted interface and effectively blocking connections from untrusted sources to trusted destinations.

## Practice

Q1) What do the security levels of an interface pair influence? (Choose best match.)

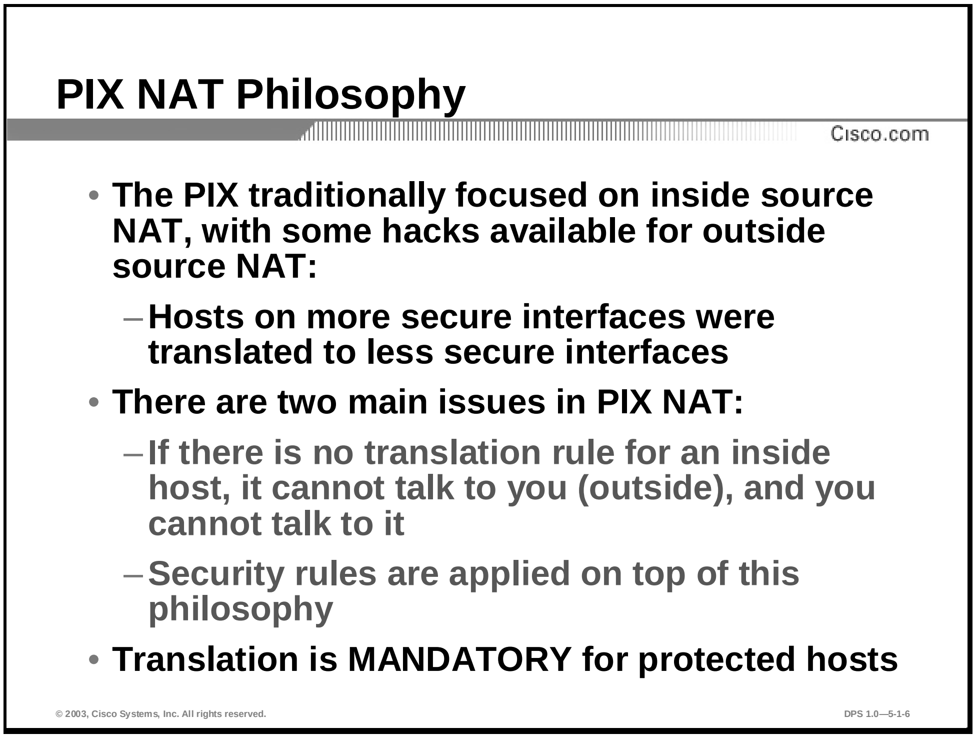
A) NAT only

B) NAT and routing

C) NAT and access rules

D) access rules and routing

# Understanding PIX NAT Translations



## PIX NAT Philosophy

- The PIX traditionally focused on inside source NAT, with some hacks available for outside source NAT:
  - Hosts on more secure interfaces were translated to less secure interfaces
- There are two main issues in PIX NAT:
  - If there is no translation rule for an inside host, it cannot talk to you (outside), and you cannot talk to it
  - Security rules are applied on top of this philosophy
- Translation is MANDATORY for protected hosts

DPS 1.0—5-1-6

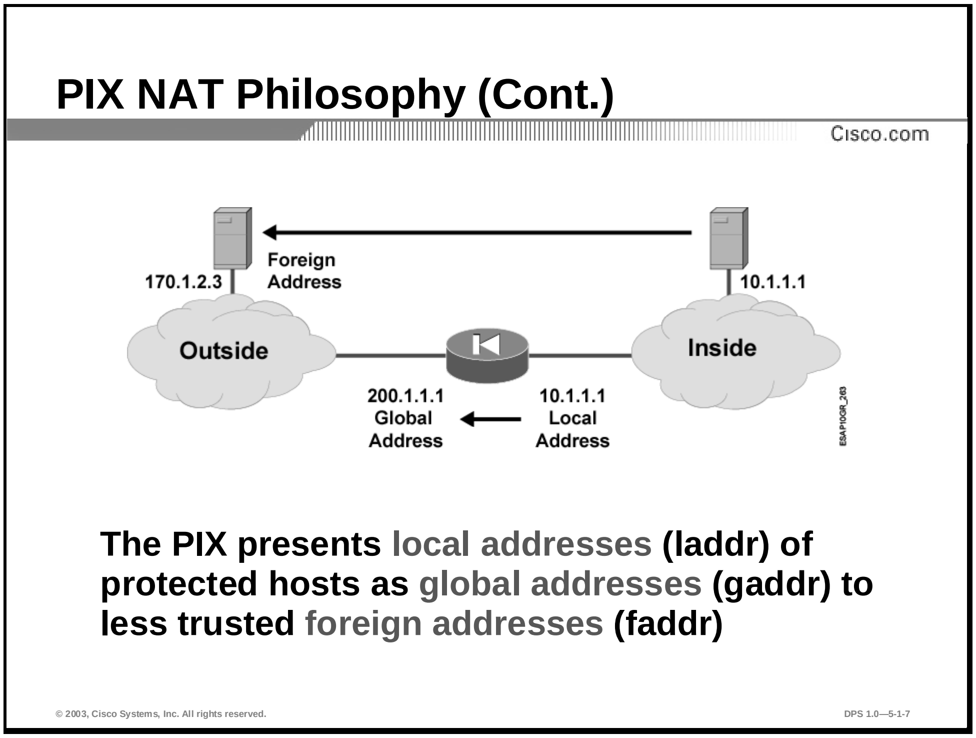
## Objective

This section will enable the learner to explain how PIX Firewall translations influence the flow of traffic through the firewall

## Introduction

NAT will always translate addresses located on interfaces with a higher security level than the other interface of the communication. There is no way to pass packets through the PIX without this feature.

Inside hosts **must** have a valid translation in one or the other way in order to talk to outside hosts. This translation can be dynamic (NAT or PAT), static or identity-NAT (e.g. translating to its own address or with other words keeping the address).



## PIX NAT Philosophy (Cont.)

Cisco.com

**The PIX presents local addresses (laddr) of protected hosts as global addresses (gaddr) to less trusted foreign addresses (faddr)**

DPS 1.0—5-1-7

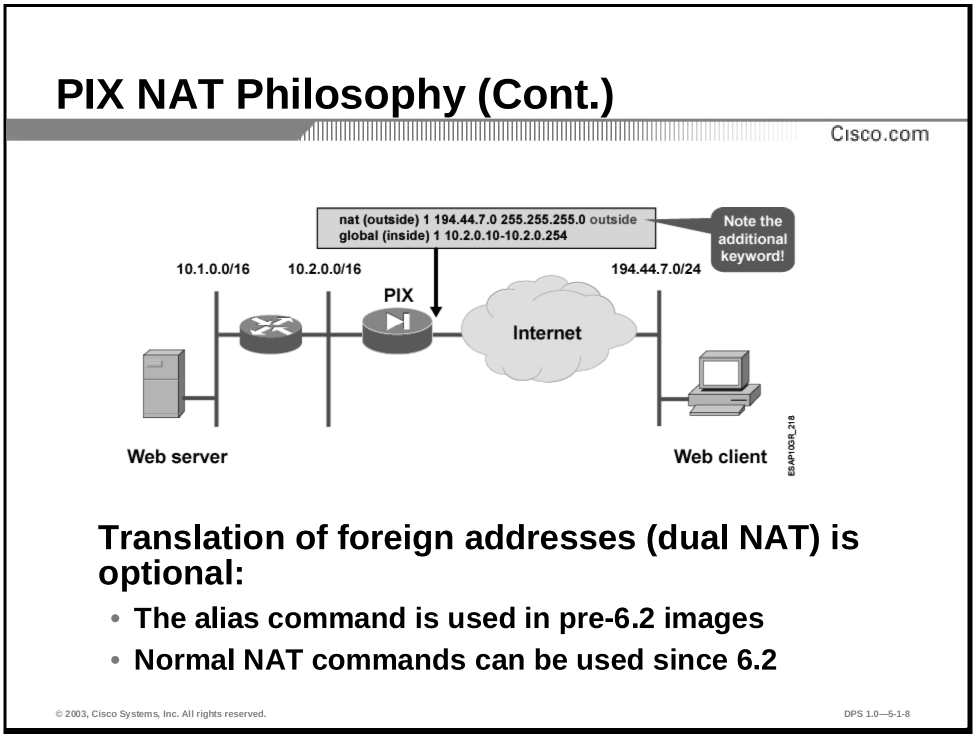
# PIX NAT Terminology

The PIX uses some specific terminology when NAT is concerned:

- A **local address** (laddr) is the “real” address of a host on a more secure interface

- A **global address** (gaddr) is the virtual address counterpart of the local address, when the local host is translated to a less secure interface

- A **foreign address** (faddr) is the address of the outside host (on a less secure interface), with which the local host is communicating



## PIX NAT Philosophy (Cont.)

```
nat (outside) 1 194.44.7.0 255.255.255.0 outside
global (inside) 1 10.2.0.10-10.2.0.254
```

Note the additional keyword!

**Translation of foreign addresses (dual NAT) is optional:**

- **The alias command is used in pre-6.2 images**
- **Normal NAT commands can be used since 6.2**

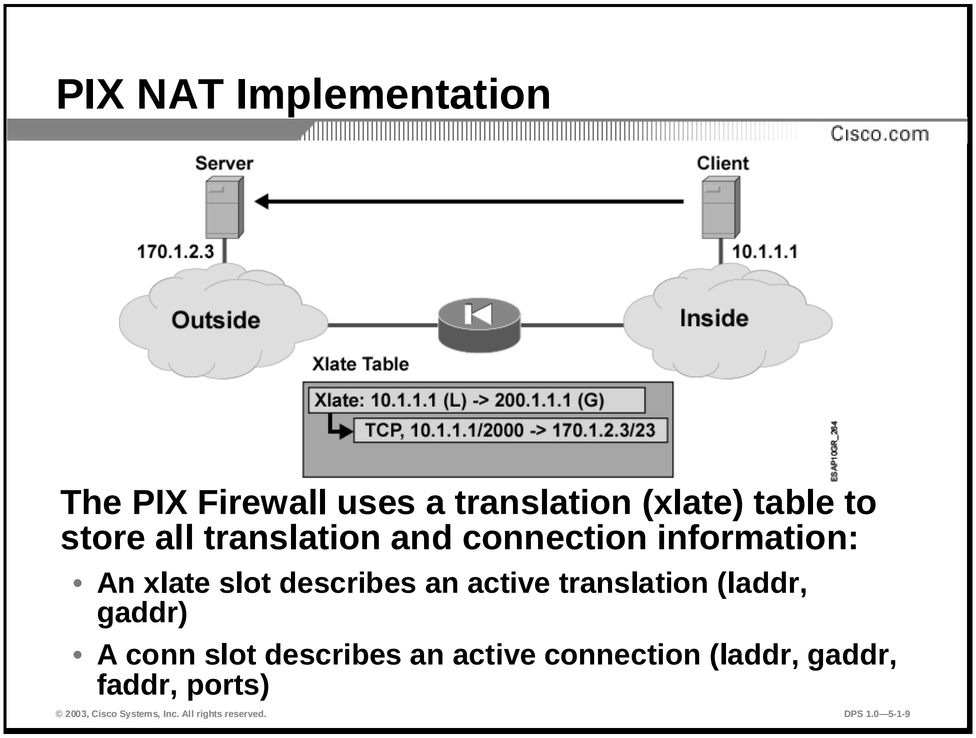
© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—5-1-8

# PIX Outside (Dual) NAT

For some special cases the PIX can additionally translate the address of outside hosts, if the address overlaps with internally used addresses. This strategy is sometimes known as “dual NAT”, “outside NAT” or “overlapping NAT”.

Pre 6.2 versions use the alias command, from version 6.2 and later normal NAT commands with specials parameters can be used.



PIX NAT Implementation

Xlate: 10.1.1.1 (L) -> 200.1.1.1 (G)

TCP, 10.1.1.1/2000 -> 170.1.2.3/23

The PIX Firewall uses a translation (xlate) table to store all translation and connection information:

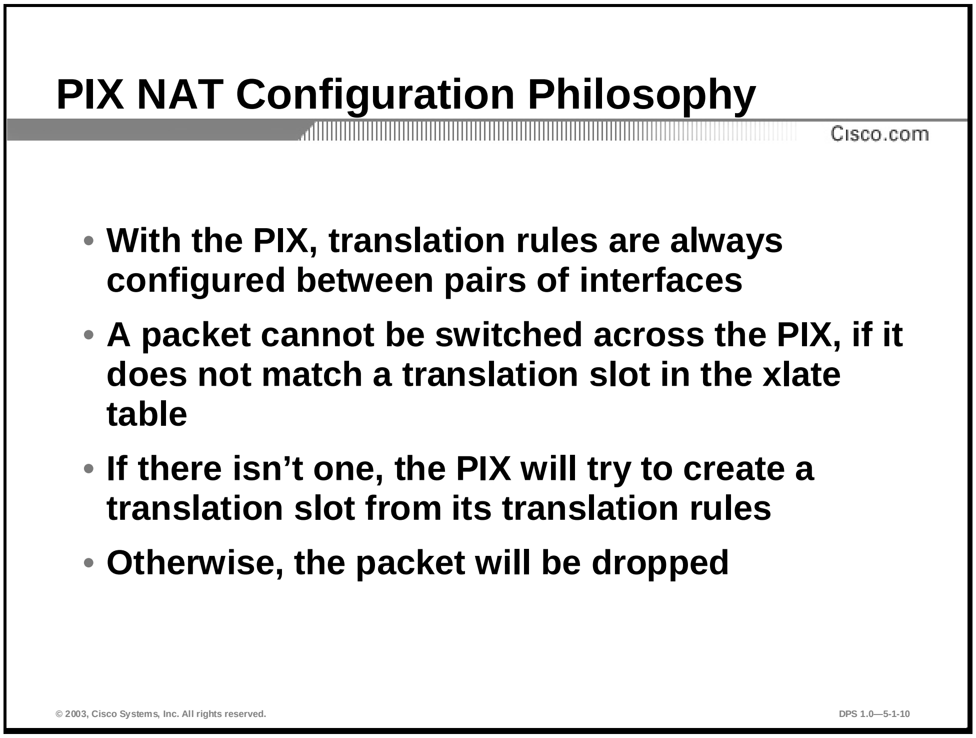
- An xlate slot describes an active translation (laddr, gaddr)
- A conn slot describes an active connection (laddr, gaddr, faddr, ports)

DPS 1.0—5-1-9

## PIX NAT Data Structures

Whenever an inside hosts establishes a connection to an outside host, a translation will be built. The translations are kept in a XLATE table describing all translations from inside local to inside global.

Another table describes all connections. The PIX can keep UDP and TCP connections in this table. The PIX **must** identify a translation first, before an entry into the connection table can be built. If the translation entry is deleted all connections belonging to this translation will also be deleted.



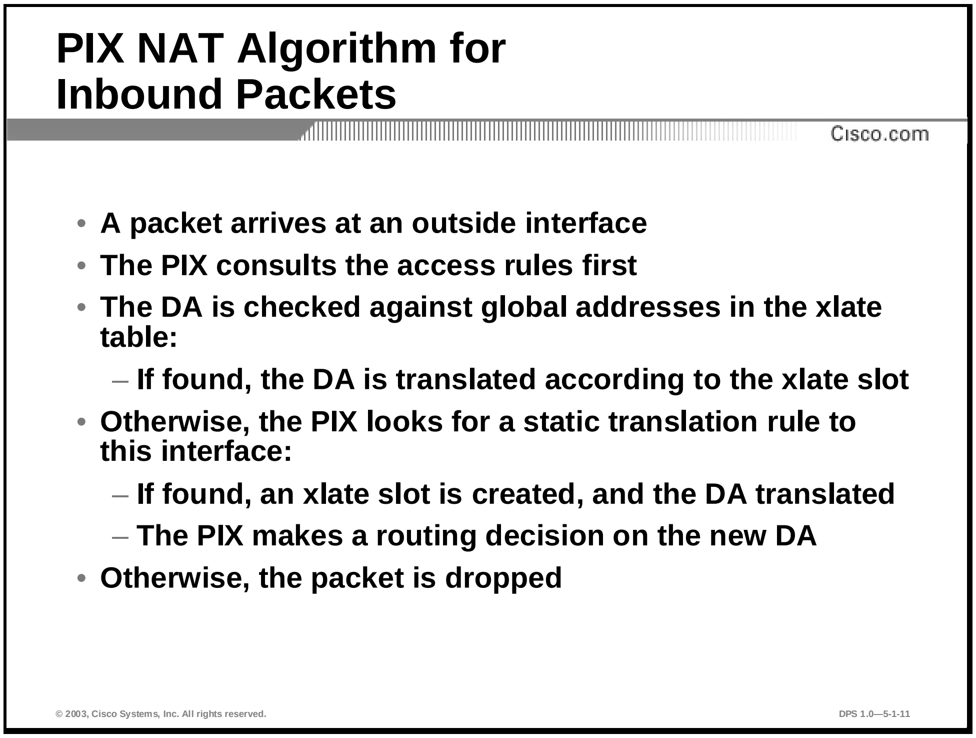
## PIX NAT Configuration Philosophy

- With the PIX, translation rules are always configured between pairs of interfaces
- A packet cannot be switched across the PIX, if it does not match a translation slot in the xlate table
- If there isn’t one, the PIX will try to create a translation slot from its translation rules
- Otherwise, the packet will be dropped

DPS 1.0—5-1-10

# PIX NAT Configuration Philosophy

Translations will be built for every source/destination interface pair. This permits to translate the same internal host to different addresses, depending on the destination. No packet will be forwarded by the PIX, if the packet does not match any of the existing translation entries or if such a translation entry cannot be established according to the translation rules.



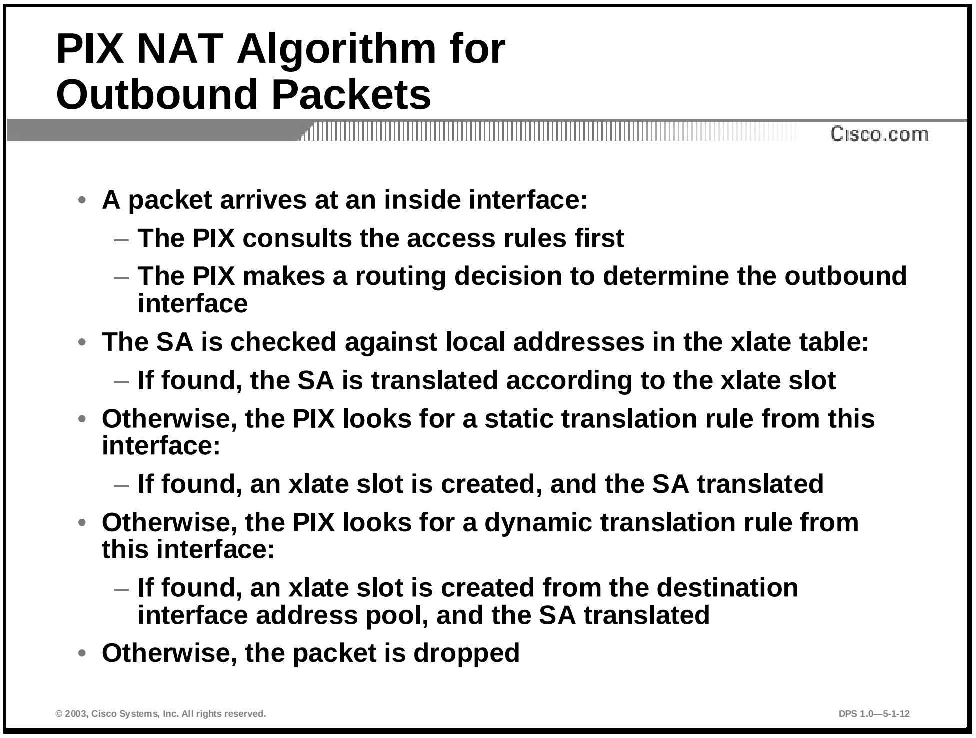
## PIX NAT Algorithm for Inbound Packets

- A packet arrives at an outside interface
- The PIX consults the access rules first
- The DA is checked against global addresses in the xlate table:
  - If found, the DA is translated according to the xlate slot
- Otherwise, the PIX looks for a static translation rule to this interface:
  - If found, an xlate slot is created, and the DA translated
  - The PIX makes a routing decision on the new DA
- Otherwise, the packet is dropped

DPS 1.0—5-1-11

## NATting Inbound Connections

**Inbound** connections are treated according to this description: the PIX will first check access-lists (or conduits) to verify the inbound connection. If the destinations address matches one of the inside global addresses, it will be translated according to this table entry. If no entry is found the PIX will scan through the static- and identity-translations. If a matching entry is found, the PIX will establish a translation in the XLAT table.



## PIX NAT Algorithm for Outbound Packets

- A packet arrives at an inside interface:
  - The PIX consults the access rules first
  - The PIX makes a routing decision to determine the outbound interface
- The SA is checked against local addresses in the xlate table:
  - If found, the SA is translated according to the xlate slot
- Otherwise, the PIX looks for a static translation rule from this interface:
  - If found, an xlate slot is created, and the SA translated
- Otherwise, the PIX looks for a dynamic translation rule from this interface:
  - If found, an xlate slot is created from the destination interface address pool, and the SA translated
- Otherwise, the packet is dropped

© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—5-1-12

# NATting Inbound Connections

**Outbound** connections are treated like this: the destination interface will be evaluated according to the routing table, it **must** have a **lower** security level than the originating interface. The PIX then compares the source address with inside-local entries in the XLATE table. If a match is found, the translation will be used. If no translation exists for this src/dst interface pair a new translation will be created, according to the translation rules.

If no rule matches for this new connection, the packet will be dropped.

# Practice

Q1) Which addresses can be translated on the PIX Firewall?

A) only inside hosts

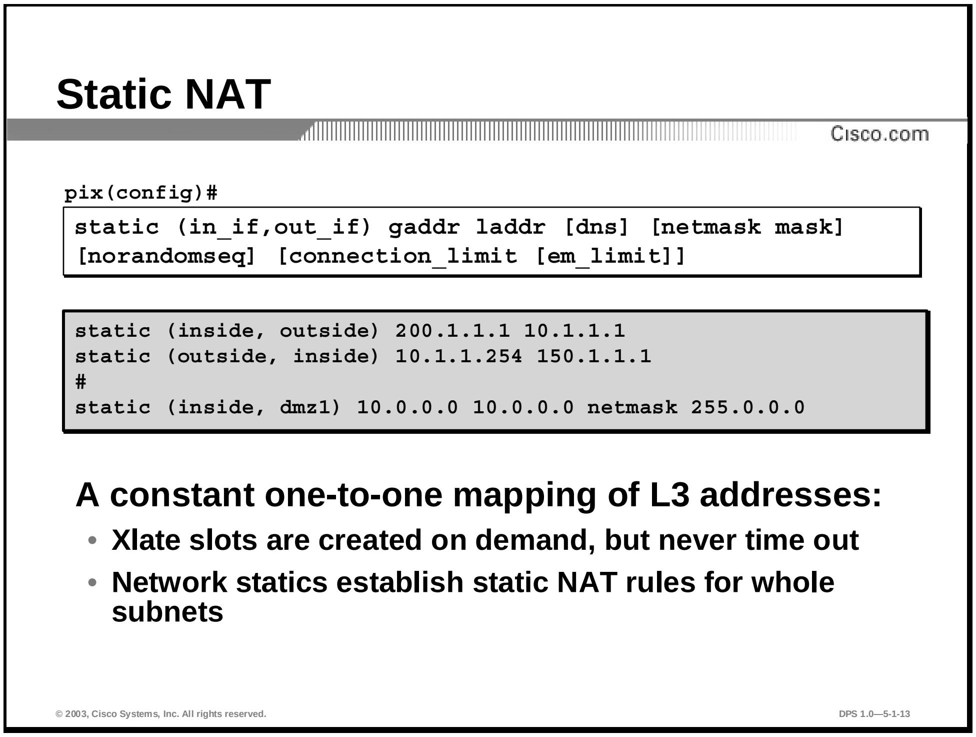
B) inside hosts and outside hosts

C) only outside hosts

D) only protected hosts

E) only foreign addresses

# Understanding PIX One-to-One NAT Translations



## Static NAT

Cisco.com

pix(config)#

```
static (in_if,out_if) gaddr laddr [dns] [netmask mask]
[norandomseq] [connection_limit [em_limit]]
```

```
static (inside, outside) 200.1.1.1 10.1.1.1
static (outside, inside) 10.1.1.254 150.1.1.1
#
static (inside, dmz1) 10.0.0.0 10.0.0.0 netmask 255.0.0.0
```

**A constant one-to-one mapping of L3 addresses:**

- **Xlate slots are created on demand, but never time out**
- **Network statics establish static NAT rules for whole subnets**

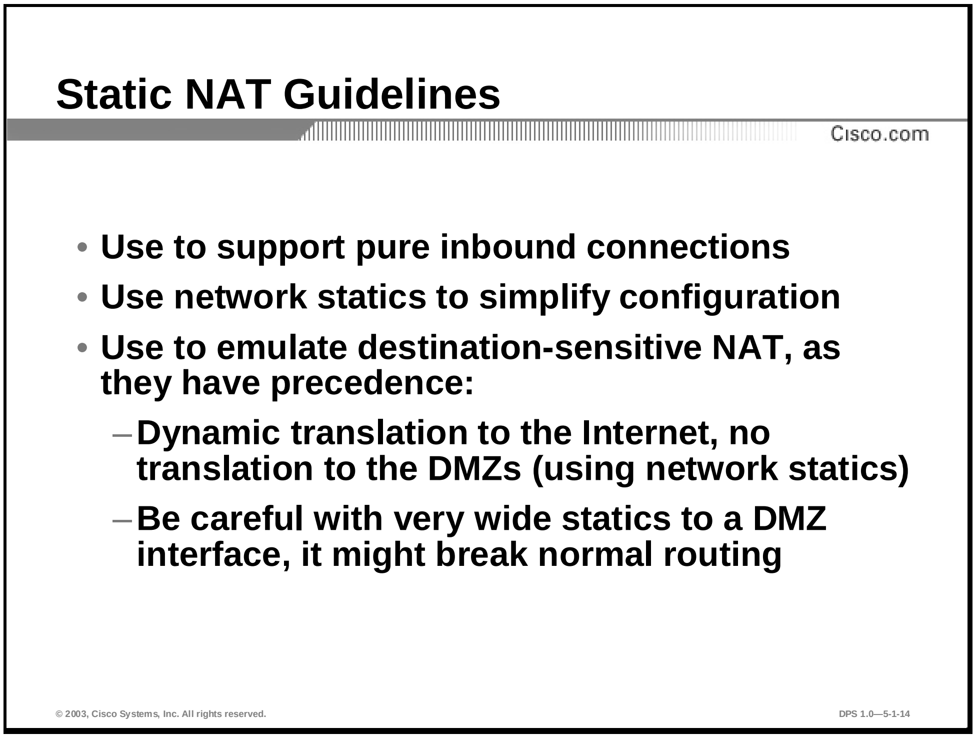
DPS 1.0—5-1-13

## Objective

This section will enable the learner to identify and configure advanced one-to-one NAT features of the Cisco Secure PIX Firewall product and choose them in a firewall design

## Introduction

The static command establishes a rule for translating an internal host always to the same inside global address for a given interface pair.



## Static NAT Guidelines

- Use to support pure inbound connections
- Use network statics to simplify configuration
- Use to emulate destination-sensitive NAT, as they have precedence:
  - Dynamic translation to the Internet, no translation to the DMZs (using network statics)
  - Be careful with very wide statics to a DMZ interface, it might break normal routing

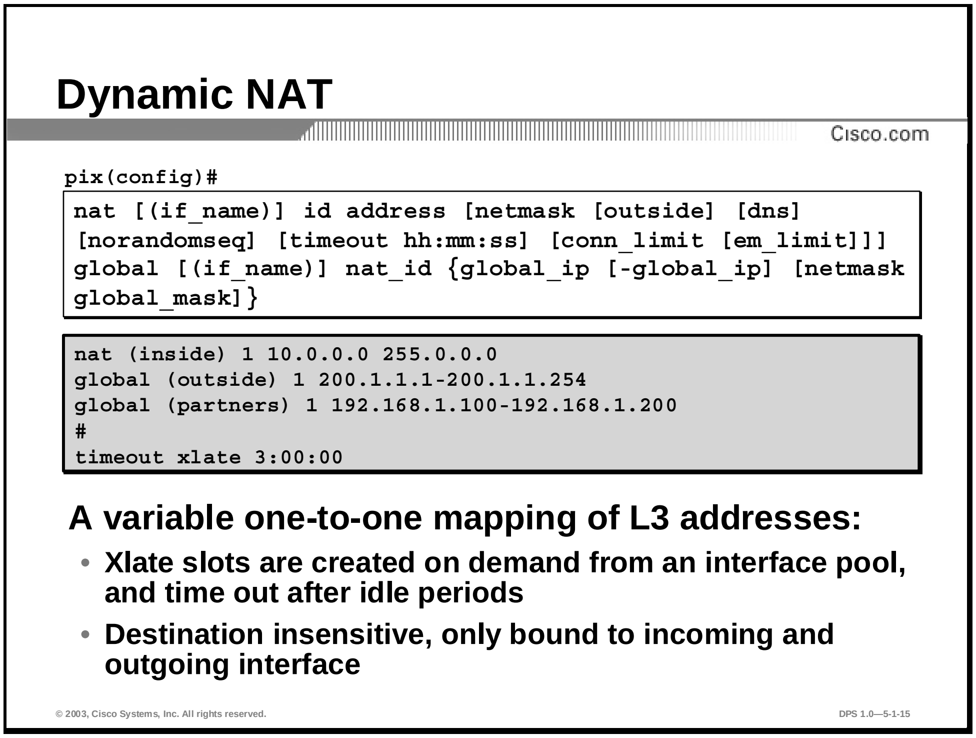
© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—5-1-14

# Static NAT Usage Guidelines

Static translations are used:

- For typical inbound scenarios e.g. a public server, that must accessible from the outside
- If an internal host must always appear with the same global address on a specific network (e.g. if address based security is used on that specific network or application)
- If an exception to the normal translation rules are needed

**Note** Static rules always have precedence over dynamic translations.



## Dynamic NAT

pix(config)#

```
nat [(if_name)] id address [netmask [outside] [dns]
[norandomseq] [timeout hh:mm:ss] [conn_limit [em_limit]]]
global [(if_name)] nat_id {global_ip [-global_ip] [netmask
global_mask]}
```

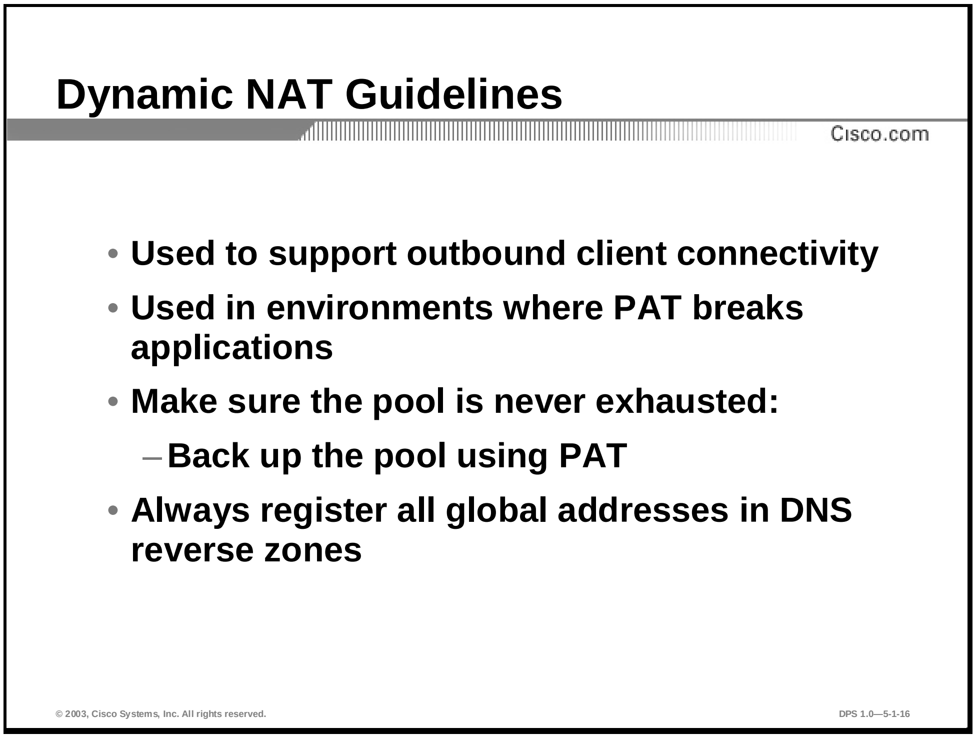
```
nat (inside) 1 10.0.0.0 255.0.0.0
global (outside) 1 200.1.1.1-200.1.1.254
global (partners) 1 192.168.1.100-192.168.1.200
#
timeout xlate 3:00:00
```

**A variable one-to-one mapping of L3 addresses:**

- **Xlate slots are created on demand from an interface pool, and time out after idle periods**
- **Destination insensitive, only bound to incoming and outgoing interface**

DPS 1.0—5-1-15

Dynamic translations are used to translate an internal pool of hosts to a defined pool of global addresses. This is the most common way to translate local hosts.



Dynamic NAT Guidelines

Cisco.com

- Used to support outbound client connectivity
- Used in environments where PAT breaks applications
- Make sure the pool is never exhausted:
  - Back up the pool using PAT
- Always register all global addresses in DNS reverse zones

DPS 1.0—5-1-16

## Dynamic NAT Usage Guidelines

Dynamic one-to-one translations are sometimes necessary when PAT (port address translation) will break the communication (e.g. H.323 must be translated this way) or if the source port must be kept through the translation (some protocols typically expect a specific source port like DNS server-server connections or NTP).

## Practice

Q1) What happens, if a global pool is exhausted for one-to-one dynamic translations?

A) the oldest translation will be deleted to permit the new connection

B) the packet will be dropped

C) the first address of the pool will be used again

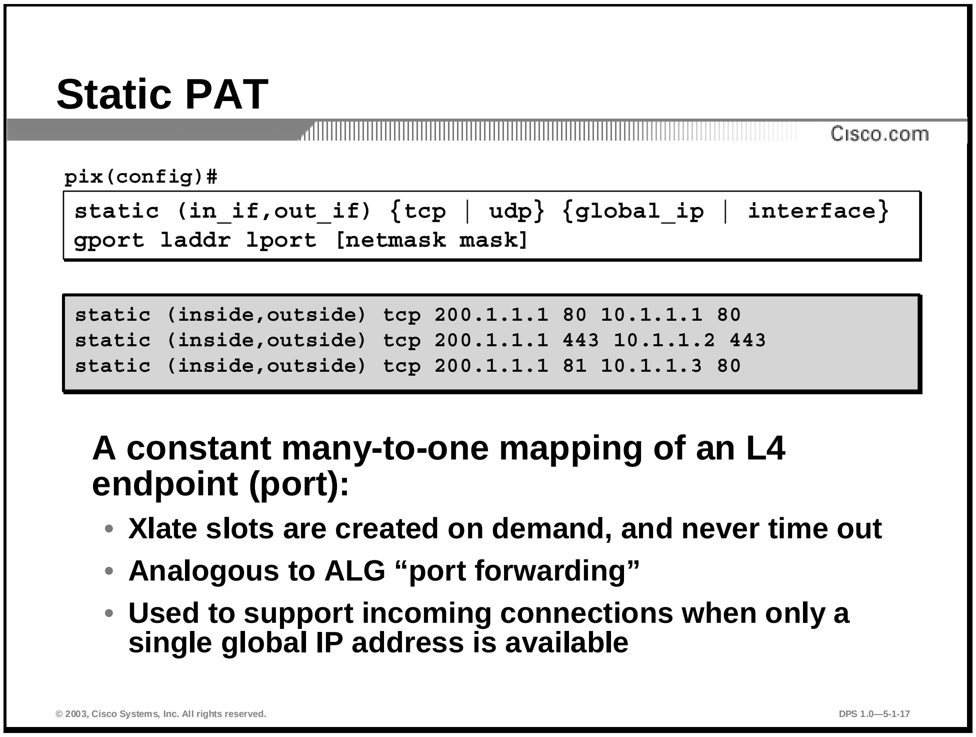
D) the packet will be forwarded with no translation

Q2) A static translation will override a dynamic one-to-one translation. True or false?

A) true

B) false

# Understanding PIX Many-to-One NAT Translations



## Static PAT

Cisco.com

pix(config)#

```
static (in_if,out_if) {tcp | udp} {global_ip | interface}
gport laddr lport [netmask mask]
```

```
static (inside,outside) tcp 200.1.1.1 80 10.1.1.1 80
static (inside,outside) tcp 200.1.1.1 443 10.1.1.2 443
static (inside,outside) tcp 200.1.1.1 81 10.1.1.3 80
```

**A constant many-to-one mapping of an L4 endpoint (port):**

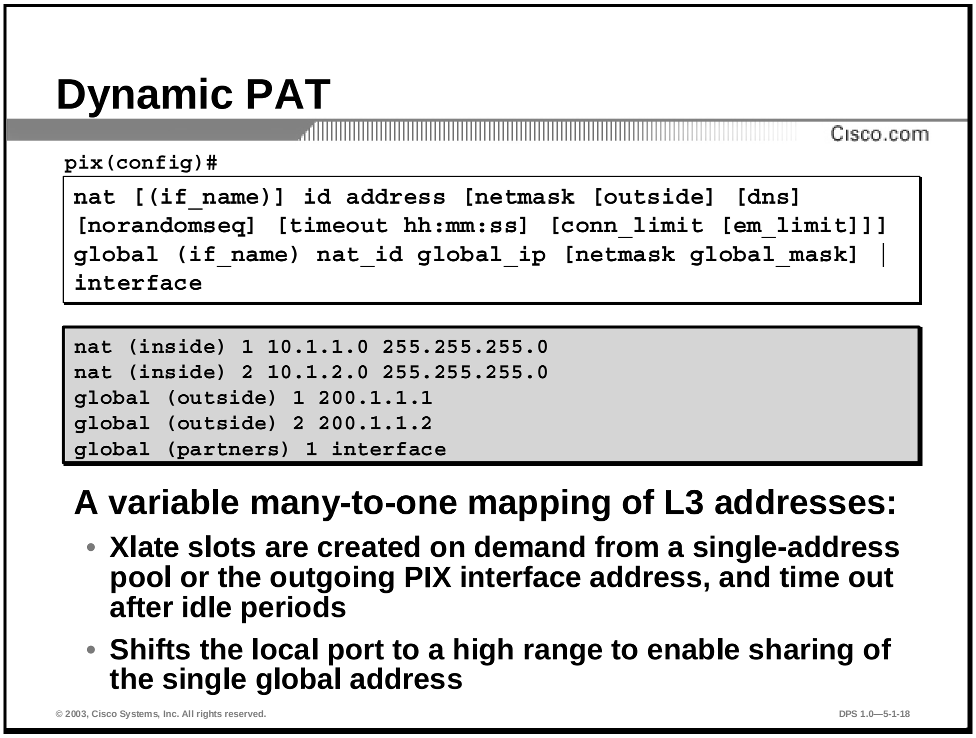
- **Xlate slots are created on demand, and never time out**
- **Analogous to ALG “port forwarding”**
- **Used to support incoming connections when only a single global IP address is available**

## Objective

This section will enable the learner to identify and configure advanced many-to-one NAT features of the Cisco Secure PIX Firewall product and choose them in a firewall design

## Introduction

Many-to-one translations, also known as PAT, allow preserving global addresses by using them for multiple internal hosts. In this case the PIX will extend the translation table by local-inside-port and global-inside-port.



## Dynamic PAT

```
pix(config)#
```

```
nat [(if_name)] id address [netmask [outside] [dns]
[norandomseq] [timeout hh:mm:ss] [conn_limit [em_limit]]]
global (if_name) nat_id global_ip [netmask global_mask] |
interface
```

```
nat (inside) 1 10.1.1.0 255.255.255.0
nat (inside) 2 10.1.2.0 255.255.255.0
global (outside) 1 200.1.1.1
global (outside) 2 200.1.1.2
global (partners) 1 interface
```

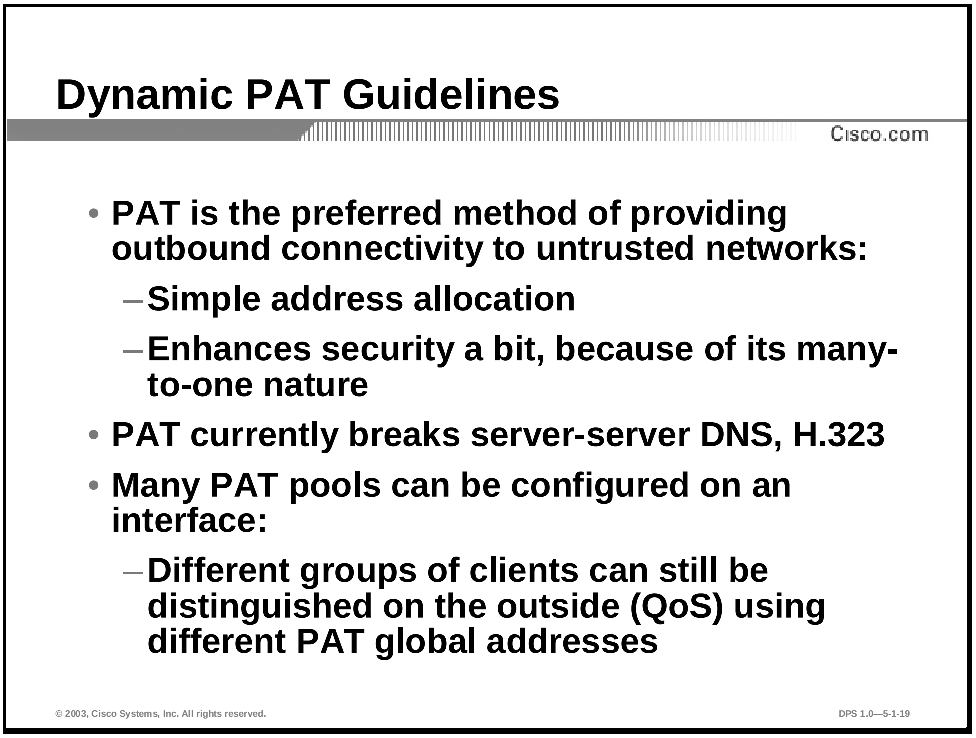
**A variable many-to-one mapping of L3 addresses:**

- **Xlate slots are created on demand from a single-address pool or the outgoing PIX interface address, and time out after idle periods**
- **Shifts the local port to a high range to enable sharing of the single global address**

DPS 1.0—5-1-18

Specifying a single address for a global pool configures PAT.

The special keyword “interface” will the use the PIX’s own interface address for PAT translations.



### Dynamic PAT Guidelines

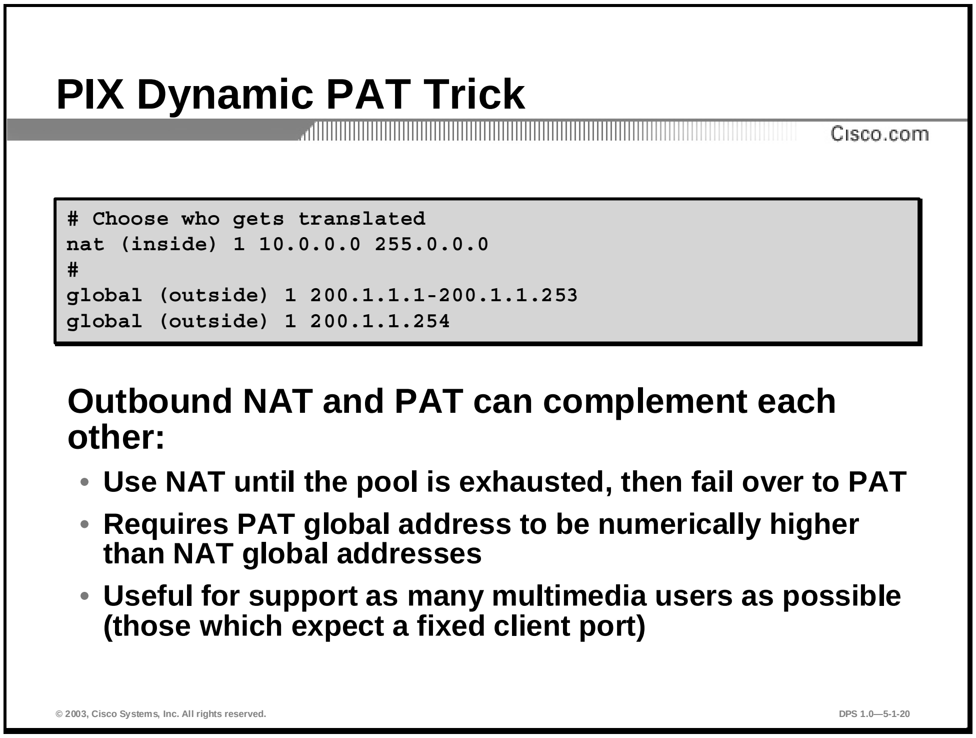
- PAT is the preferred method of providing outbound connectivity to untrusted networks:
  - Simple address allocation
  - Enhances security a bit, because of its many-to-one nature
- PAT currently breaks server-server DNS, H.323
- Many PAT pools can be configured on an interface:
  - Different groups of clients can still be distinguished on the outside (QoS) using different PAT global addresses

DPS 1.0—5-1-19

## Dynamic PAT Usage Guidelines

PAT is usually the preferred method of providing outbound connectivity, because it is simple to configure, and enhances security a bit, because of it’s unidirectional nature. PAT can in some cases break the communication – for example with DNS server-to-server queries, or with applications which expect a fixed client port to be available. In this case use NAT instead.

Many PAT pools can be active on an interface, which enables the operator to distinguish multiple groups of inside users on the outside of the PIX. For example, inside subnets could be translated to different PAT addresses and get different levels of QoS on the Internet link.



## PIX Dynamic PAT Trick

```
# Choose who gets translated
nat (inside) 1 10.0.0.0 255.0.0.0
#
global (outside) 1 200.1.1.1-200.1.1.253
global (outside) 1 200.1.1.254
```

**Outbound NAT and PAT can complement each other:**

- **Use NAT until the pool is exhausted, then fail over to PAT**
- **Requires PAT global address to be numerically higher than NAT global addresses**
- **Useful for support as many multimedia users as possible (those which expect a fixed client port)**

DPS 1.0—5-1-20

A special configuration allows using PAT only, if all one-to-one translations are used up. Simply specify a global statement for the same global pool with a single address.

Translation will be built in a normal one-to-one manner; subsequent connections will use PAT only if no more global addresses are vacant.

# Practice

Q1) The PIX's outside interface can be used for PAT. True or false?

A) true

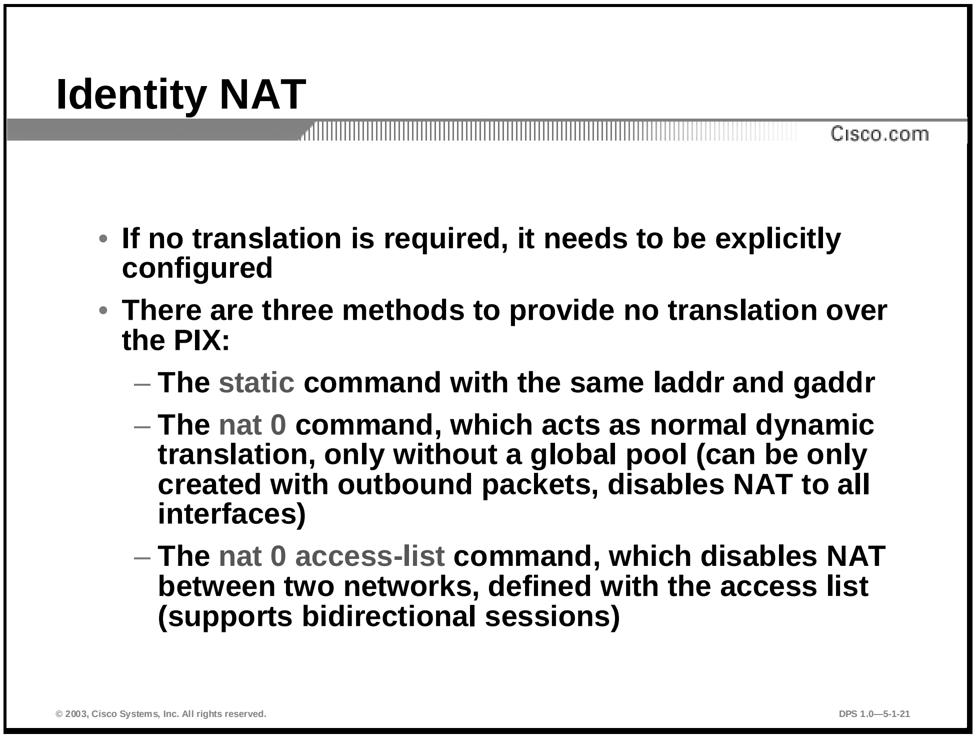
B) false

Q2) NAT and PAT can be configured for the same global pool. True or false?

A) true

B) false

# Understanding PIX Identity NAT



## Identity NAT

- If no translation is required, it needs to be explicitly configured
- There are three methods to provide no translation over the PIX:
  - The static command with the same laddr and gaddr
  - The nat 0 command, which acts as normal dynamic translation, only without a global pool (can be only created with outbound packets, disables NAT to all interfaces)
  - The nat 0 access-list command, which disables NAT between two networks, defined with the access list (supports bidirectional sessions)

© 2003, Cisco Systems, Inc. All rights reserved.

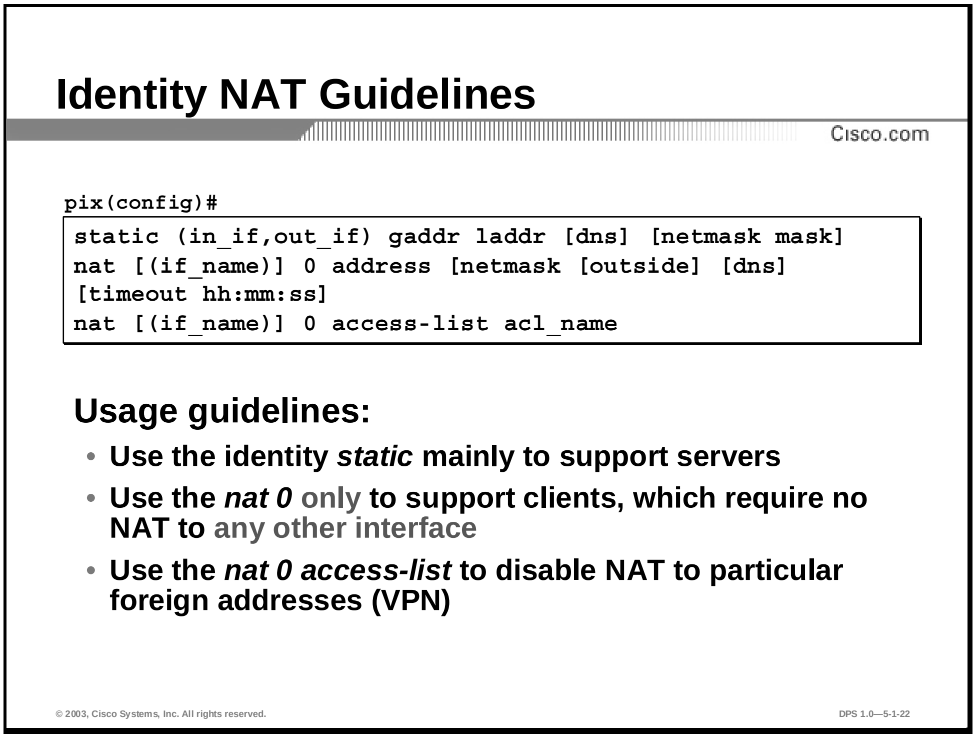
DPS 1.0—5-1-21

## Objective

This section will enable the learner to explain and configure advanced identity NAT features of the Cisco Secure PIX Firewall product and choose them in a firewall design

## Introduction

Sometimes it is necessary to keep local addresses when communicating through the PIX. This can be done with either the static or with NAT 0 commands.



## Identity NAT Guidelines

pix(config)#

```
static (in_if,out_if) gaddr laddr [dns] [netmask mask]
nat [(if_name)] 0 address [netmask [outside] [dns]
[timeout hh:mm:ss]
nat [(if_name)] 0 access-list acl_name
```

**Usage guidelines:**

- **Use the identity *static* mainly to support servers**
- **Use the *nat 0* only to support clients, which require no NAT to any other interface**
- **Use the *nat 0 access-list* to disable NAT to particular foreign addresses (VPN)**

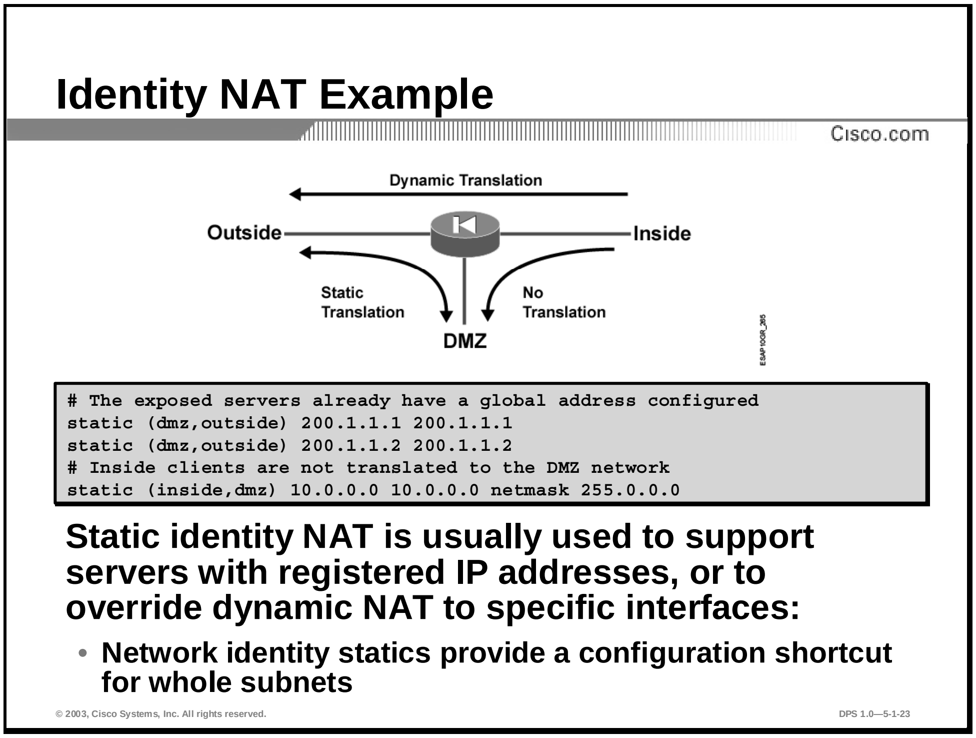
DPS 1.0—5-1-22

# Identity NAT Usage Guidelines

A static command will have precedence over any dynamic translation and is valid for exactly the specified interface combination. Connections using these two interfaces and including the specified internal host will be treated without address translation.

The NAT 0 command will be treated like other NAT commands and assumes a global address that is identical to the local address. It should never be used with local addresses, which are already described with other NAT statements.

The NAT 0 access-list statement turns on identity NAT only for connections, that match a permit statement of the specified access-list.



## Identity NAT Example

Cisco.com

```
# The exposed servers already have a global address configured
static (dmz,outside) 200.1.1.1 200.1.1.1
static (dmz,outside) 200.1.1.2 200.1.1.2
# Inside clients are not translated to the DMZ network
static (inside,dmz) 10.0.0.0 10.0.0.0 netmask 255.0.0.0
```

**Static identity NAT is usually used to support servers with registered IP addresses, or to override dynamic NAT to specific interfaces:**

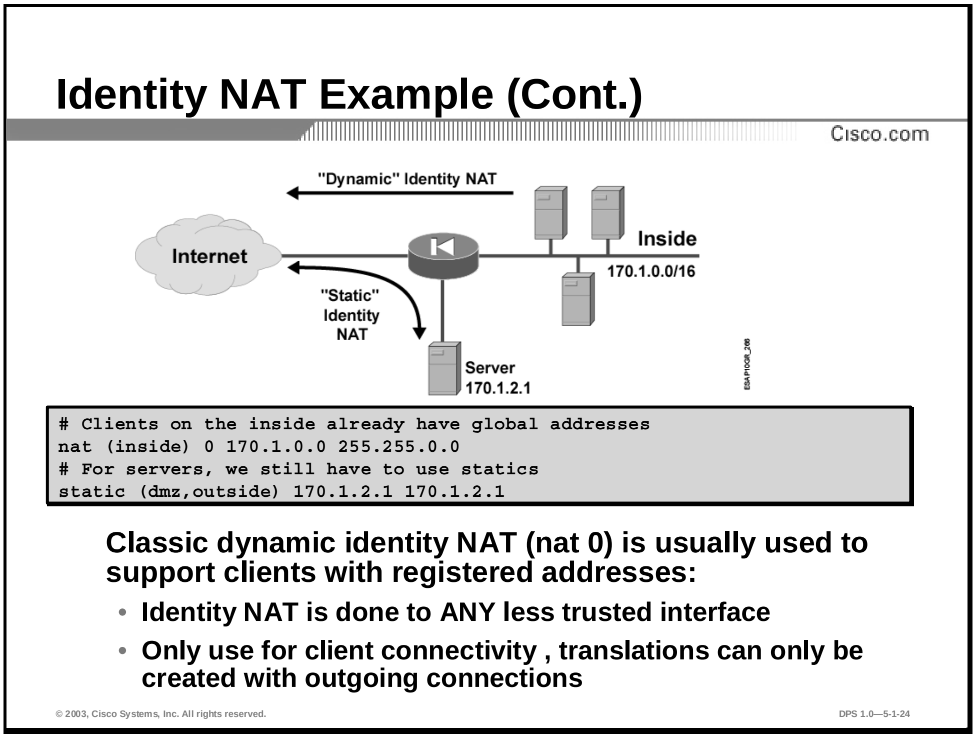
- **Network identity statics provide a configuration shortcut for whole subnets**

DPS 1.0—5-1-23

## Example

A typical example for NAT 0 is shown here, where internal hosts use addresses, which are valid in the global address space (e.g. using INTERNIC registered addresses for internal hosts). No translation of addresses needs to be performed from the “dmz” to the “outside” interface, and static rules are used to accomplish this as connections can initiate from either “dmz” or “outside” interface.

Also, a rule for no translation of inside hosts’ addresses to the DMZ is configured to preserve IP addressing for “inside” to “dmz” connectivity.



## Identity NAT Example (Cont.)

```
# Clients on the inside already have global addresses
nat (inside) 0 170.1.0.0 255.255.0.0
# For servers, we still have to use statics
static (dmz,outside) 170.1.2.1 170.1.2.1
```

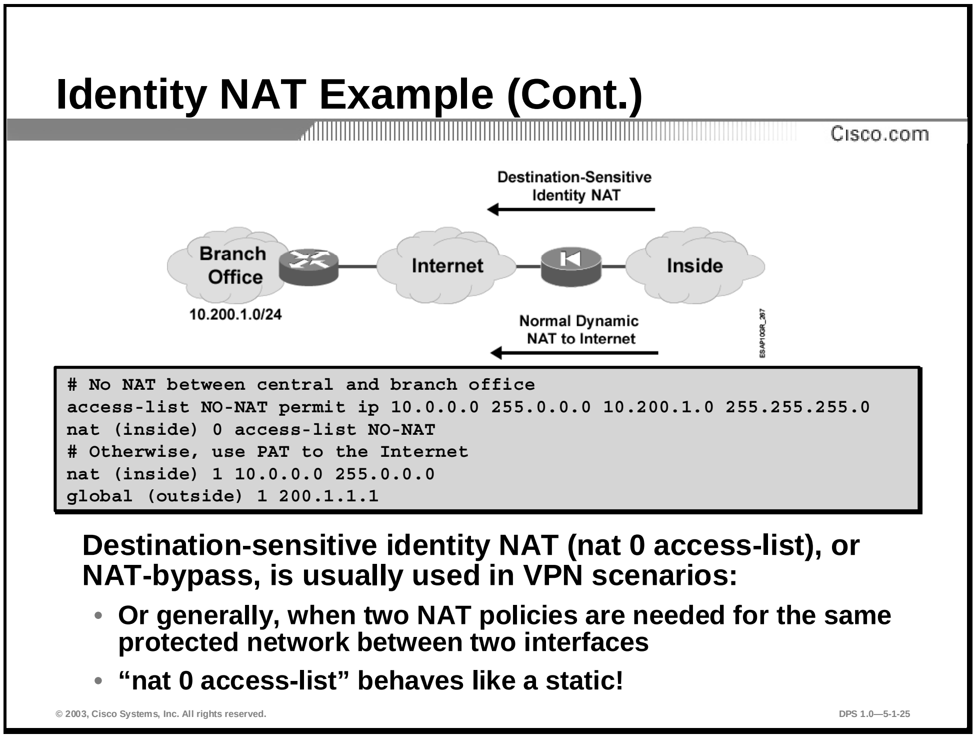
**Classic dynamic identity NAT (nat 0) is usually used to support clients with registered addresses:**

- **Identity NAT is done to ANY less trusted interface**
- **Only use for client connectivity , translations can only be created with outgoing connections**

DPS 1.0—5-1-24

# Example

In this example the public server uses an INTERNIC registered address and a static rule is required to reach it from the outside . Other inside hosts also have registered addresses, but a dynamic rule is configured, as translations can be created on demand with outbound connections only. This limits the exposure of inside hosts when they are not active, as their translations time out, and no inbound connectivity to them would be possible even if access lists on the outside interface were mistakenly configured to permit inbound connectivity.



## Identity NAT Example (Cont.)

Cisco.com

Destination-Sensitive Identity NAT
Branch Office
Internet
Inside
10.200.1.0/24
Normal Dynamic NAT to Internet

```
# No NAT between central and branch office
access-list NO-NAT permit ip 10.0.0.0 255.0.0.0 10.200.1.0 255.255.255.0
nat (inside) 0 access-list NO-NAT
# Otherwise, use PAT to the Internet
nat (inside) 1 10.0.0.0 255.0.0.0
global (outside) 1 200.1.1.1
```

**Destination-sensitive identity NAT (nat 0 access-list), or NAT-bypass, is usually used in VPN scenarios:**

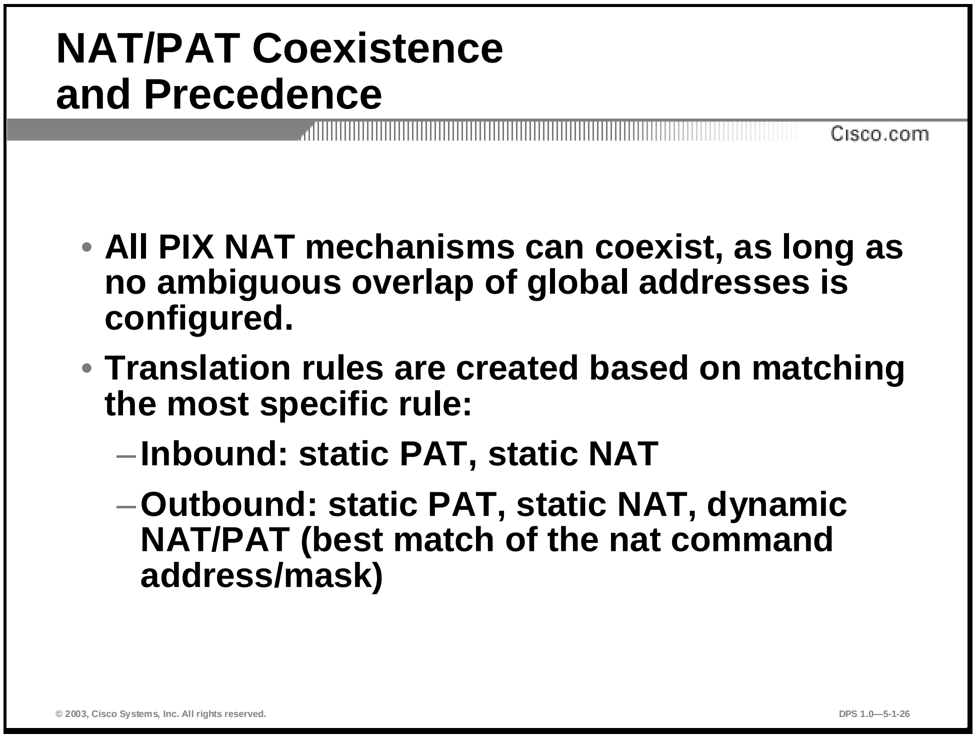
- **Or generally, when two NAT policies are needed for the same protected network between two interfaces**
- **“nat 0 access-list” behaves like a static!**

DPS 1.0—5-1-25

## Example

In this example, the requirement is that NAT should be disabled between two VPN networks (the 10.0.0.0/8 and the 10.200.1.0/24). Therefore, for VPN communications the addresses are identity-translated, when the destination is also a member of the VPN-network. The “nat 0 access-list” command is the only command which allows the designer to use destination-sensitive rules. Packets must match a permit statement of the access-list.

Note that the “nat 0 access-list” command behaves like a static – it can support inbound and outbound connections with no restrictions.



## NAT/PAT Coexistence and Precedence

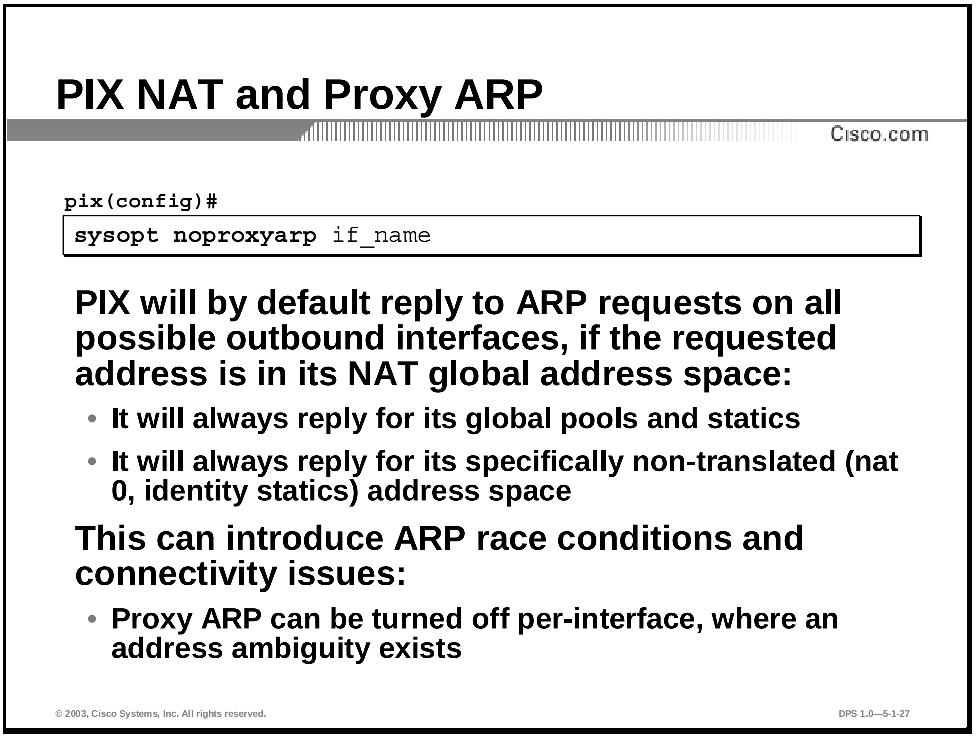
- All PIX NAT mechanisms can coexist, as long as no ambiguous overlap of global addresses is configured.
- Translation rules are created based on matching the most specific rule:
  - Inbound: static PAT, static NAT
  - Outbound: static PAT, static NAT, dynamic NAT/PAT (best match of the nat command address/mask)

DPS 1.0—5-1-26

All PIX NAT mechanisms (dynamic, static, identity NAT) can coexist, as long as interfaces are not configured with overlapping global addresses.

When configuring multiple NAT mechanisms, the most specific rule will apply to traffic. For connections initiated inbound, static PAT is evaluated first, followed by static NAT.

For outbound connections, static PAT has precendece over static NAT, which has precedence over classic dynamic NAT/PAT.



### PIX NAT and Proxy ARP

```
pix(config)#
sysopt noproxyarp if_name
```

**PIX will by default reply to ARP requests on all possible outbound interfaces, if the requested address is in its NAT global address space:**

- **It will always reply for its global pools and statics**
- **It will always reply for its specifically non-translated (nat 0, identity statics) address space**

**This can introduce ARP race conditions and connectivity issues:**

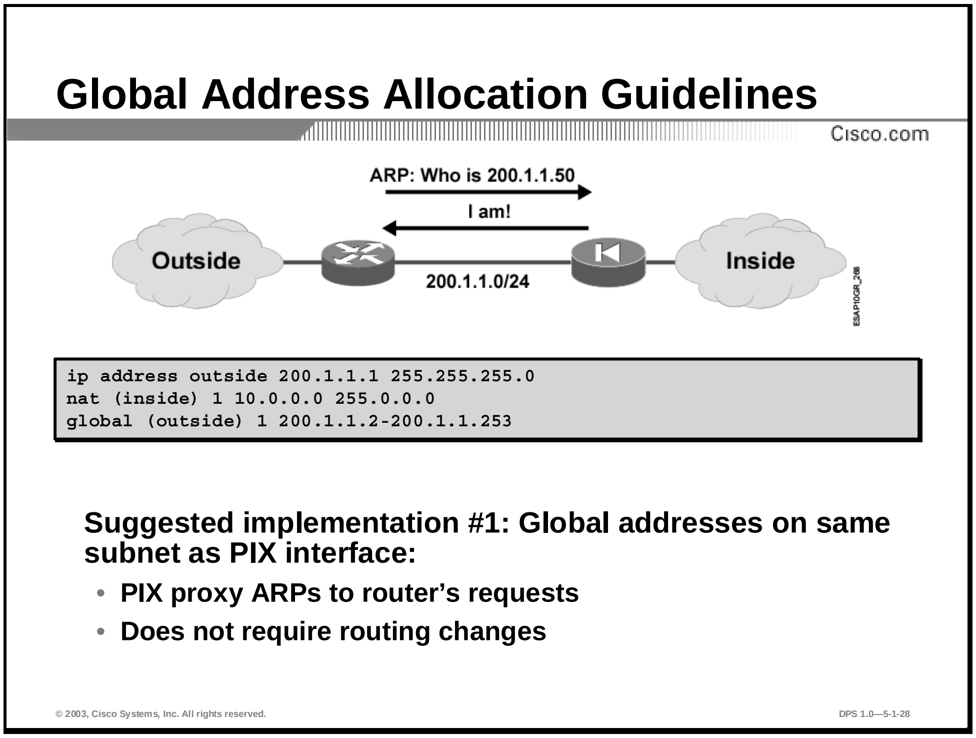
- **Proxy ARP can be turned off per-interface, where an address ambiguity exists**

DPS 1.0—5-1-27

## PIX NAT and Proxy ARP

Proxy ARPing can sometimes lead to problems, turning off proxy ARP on interfaces can help this situation. The PIX by default replies to any ARP requests for addresses, which are configured as global addresses on the PIX (the global pools, or the global addresses in static rules). This includes any identity-translated space.

The problem manifests itself when there is an overlap between the PIX global address space and a host on a network, adjacent to the PIX. In this case, both the PIX and the host will reply to the ARP request, which should be replied by the host only. This will probably disrupt connectivity to and from the host, and can lead to serious network outages.



## Global Address Allocation Guidelines

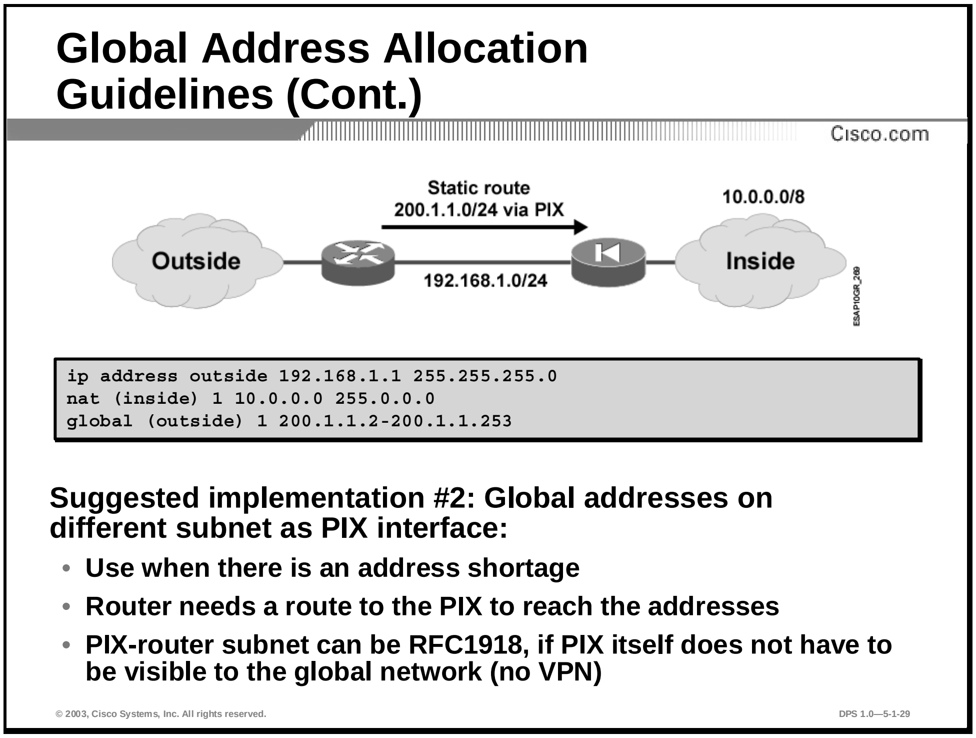
```
ip address outside 200.1.1.1 255.255.255.0
nat (inside) 1 10.0.0.0 255.0.0.0
global (outside) 1 200.1.1.2-200.1.1.253
```

**Suggested implementation #1: Global addresses on same subnet as PIX interface:**

- **PIX proxy ARPs to router's requests**
- **Does not require routing changes**

# Global Address Allocation Guidelines

This is the standard implementation of a global address pool. If it is within the network of the outside interface according to its address and mask, the perimeter router can identify routing etc by the routing table (connected network).



## Global Address Allocation Guidelines (Cont.)

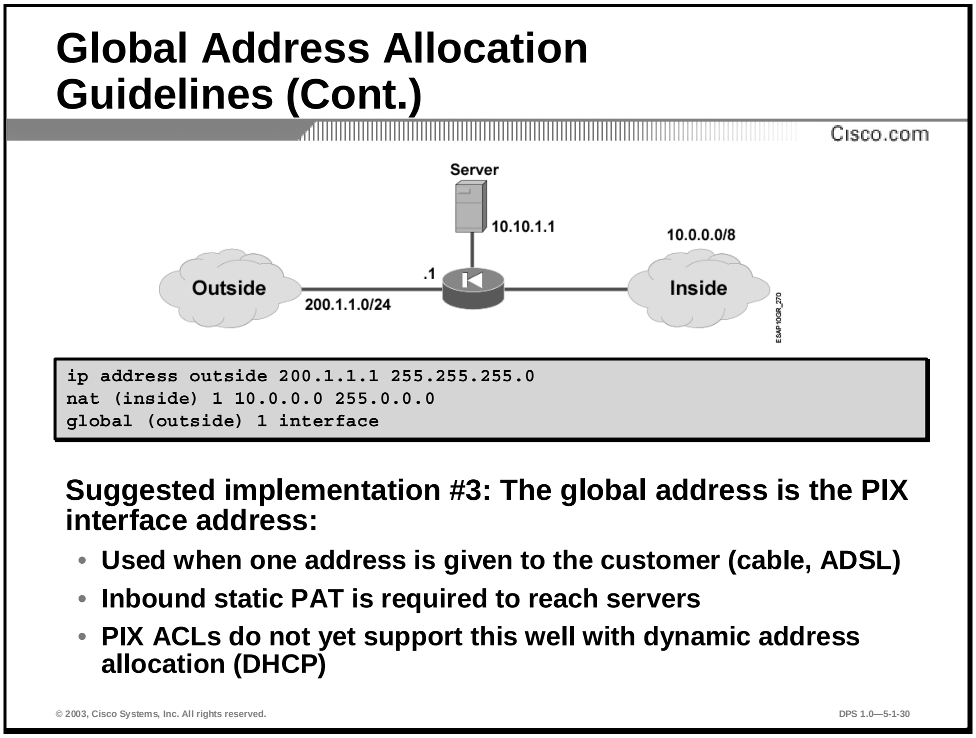
```
ip address outside 192.168.1.1 255.255.255.0
nat (inside) 1 10.0.0.0 255.0.0.0
global (outside) 1 200.1.1.2-200.1.1.253
```

**Suggested implementation #2: Global addresses on different subnet as PIX interface:**

- **Use when there is an address shortage**
- **Router needs a route to the PIX to reach the addresses**
- **PIX-router subnet can be RFC1918, if PIX itself does not have to be visible to the global network (no VPN)**

DPS 1.0—5-1-29

In this figure the router needs a static route to the PIX, because the global pool is unknown to the router.



Server
10.10.1.1
10.0.0.0/8
Outside
.1
Inside
200.1.1.0/24

## Global Address Allocation Guidelines (Cont.)

```
ip address outside 200.1.1.1 255.255.255.0
nat (inside) 1 10.0.0.0 255.0.0.0
global (outside) 1 interface
```

**Suggested implementation #3: The global address is the PIX interface address:**

- **Used when one address is given to the customer (cable, ADSL)**
- **Inbound static PAT is required to reach servers**
- **PIX ACLs do not yet support this well with dynamic address allocation (DHCP)**

DPS 1.0—5-1-30

For dynamically assigned global pools (DHCP) inbound servers must be configured using static PAT. Dynamically assigned addresses cannot be used in access-lists, the keyword “any” must be used instead.

# Practice

Q1) Which of the following PIX translation rules is consulted last, if all are configured on an interface?

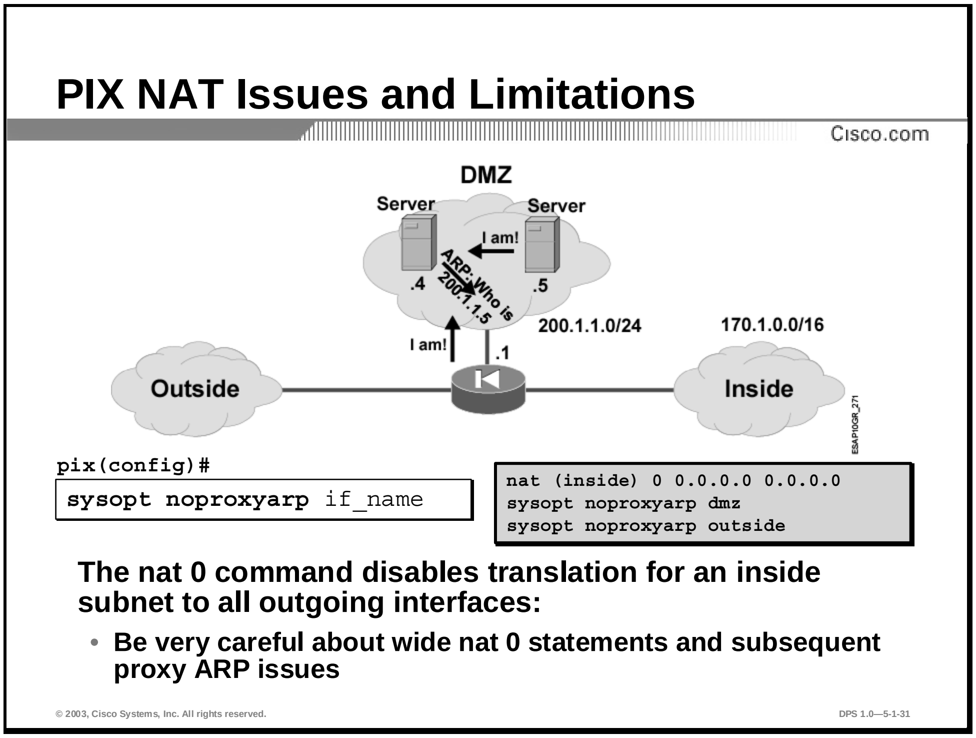
A) static PAT

B) static NAT

C) dynamic NAT

D) identity static NAT

# Understanding PIX NAT Limitations



## PIX NAT Issues and Limitations

Cisco.com

```
pix(config)#
sysopt noproxyarp if_name
```

```
nat (inside) 0 0.0.0.0 0.0.0.0
sysopt noproxyarp dmz
sysopt noproxyarp outside
```

**The nat 0 command disables translation for an inside subnet to all outgoing interfaces:**

- **Be very careful about wide nat 0 statements and subsequent proxy ARP issues**

DPS 1.0—5-1-31

## Objective

This section will enable the learner to identify PIX Firewall NAT limitations and avoid them in a firewall design

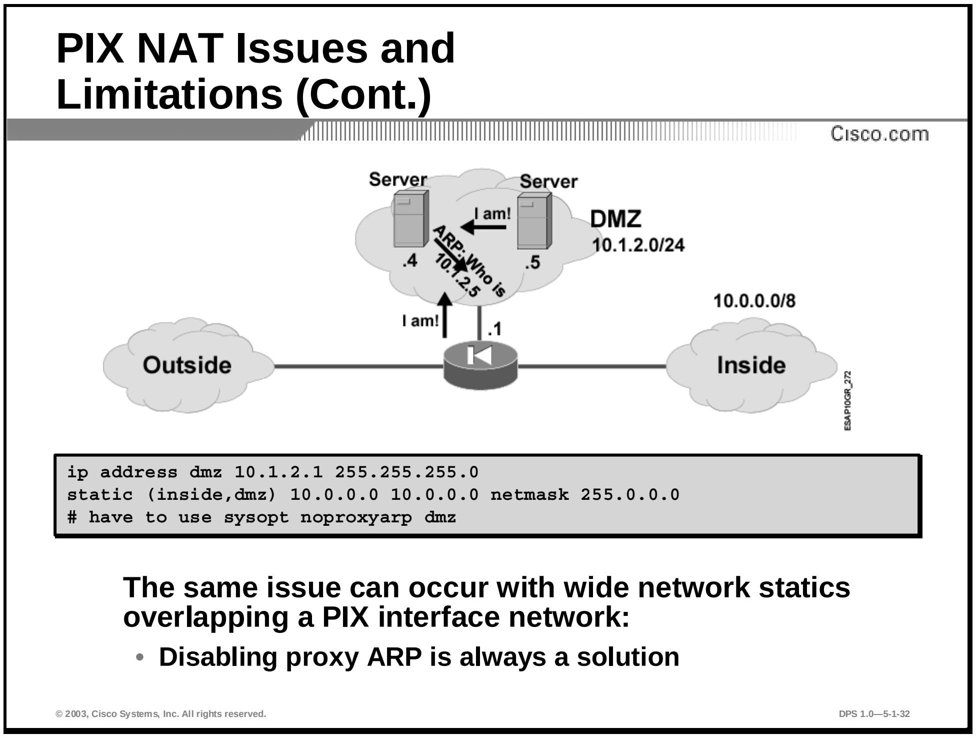
## Introduction

NAT can have limitations. NAT 0 will be valid for all destinations interfaces and can sometimes break routing (e.g. illegal source addresses appearing on the outside network).

Consider this figure, which shows a configuration of “nat 0”, which appears to identity-translate any possible address on the inside interface to any other interface.

Problems arise in such wide configurations, as the PIX Firewall will proxy ARP on outgoing interfaces, if an ARP mapping for an IP  address which the PIX Firewall thinks it could translate is requested. In the above case, the PIX Firewall will answer ALL ARP requests on other interfaces, as it believes it can translate anything to its inside interface.

The “sysopt noproxyarp” command provides a workaround, as it can disable proxy ARP functionality on an interface.



## PIX NAT Issues and Limitations (Cont.)

Cisco.com

Server
Server
I am!
DMZ
10.1.2.0/24
ARP: Who is 10.1.2.5
.4
.5
10.0.0.0/8
I am!
.1
Outside
Inside
ESAP10GR_272

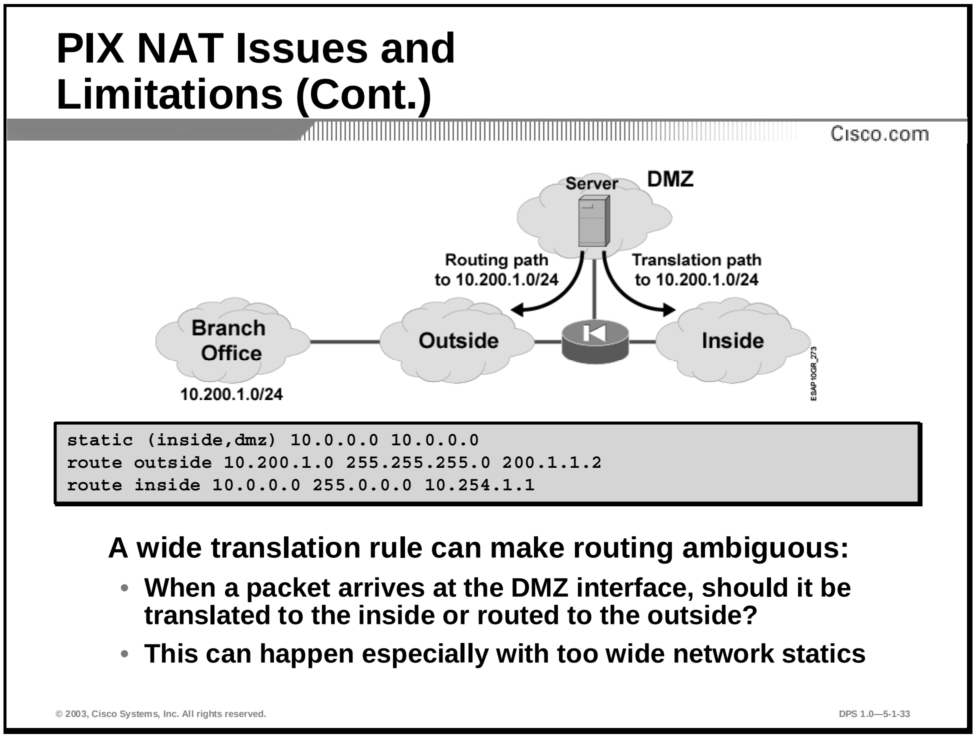
```
ip address dmz 10.1.2.1 255.255.255.0
static (inside,dmz) 10.0.0.0 10.0.0.0 netmask 255.0.0.0
# have to use sysopt noproxyarp dmz
```

**The same issue can occur with wide network statics overlapping a PIX interface network:**

- **Disabling proxy ARP is always a solution**

DPS 1.0—5-1-32

Problems also appear, when a NAT statement covers networks, which are located on more than one interface. In this example, the PIX Firewall will proxy ARP on the “dmz” interface for any address in the 10.0.0.0/8 network, as it believes it can translate it back to the “inside” interface because of the network static rule.



## PIX NAT Issues and Limitations (Cont.)

```
static (inside,dmz) 10.0.0.0 10.0.0.0
route outside 10.200.1.0 255.255.255.0 200.1.1.2
route inside 10.0.0.0 255.0.0.0 10.254.1.1
```

**A wide translation rule can make routing ambiguous:**

- **When a packet arrives at the DMZ interface, should it be translated to the inside or routed to the outside?**
- **This can happen especially with too wide network statics**

DPS 1.0—5-1-33

Routing and NAT information can sometimes interfere, when the NAT statement is “too generous”. In this example, the static translation rule maps the 10.0.0.0/8 network from the inside to the DMZ. Any packets destined to the 10.0.0.0/8 network from the DMZ will be identity-translated to the inside network. There is also a branch office network, which is reachable through the outside interface. This network is now unreachable, as translation takes precedence over routing.

The solution to this problem would be to specify more specific static translation rules between inside and DMZ, specifically tailored to inside networks.

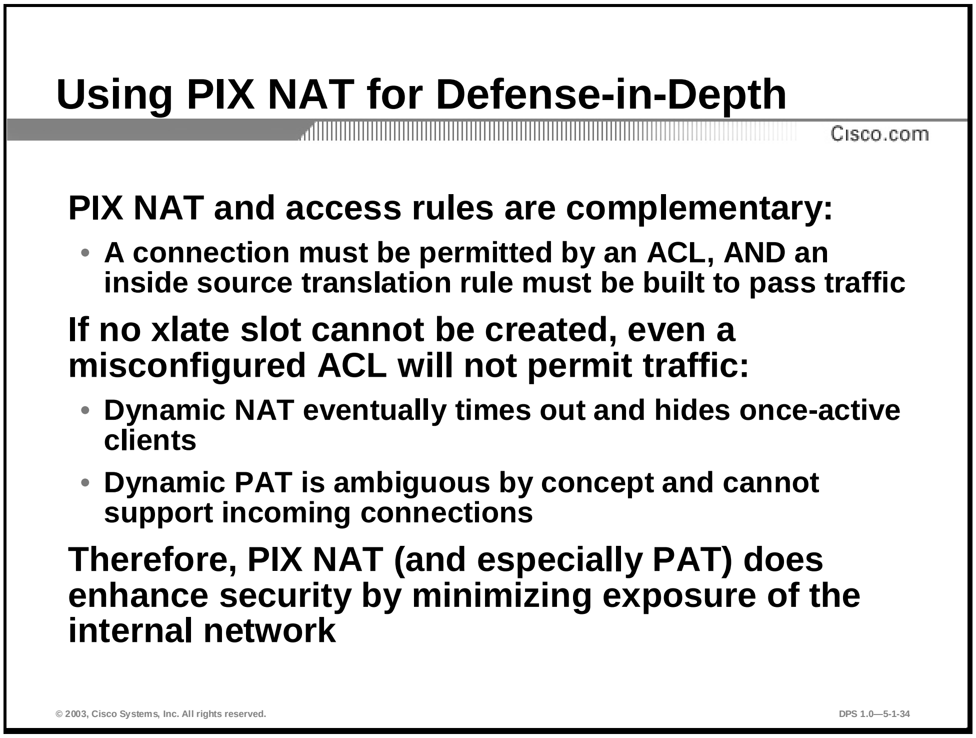
# Practice

Q1) All translation rules can coexist, if configured properly. True or false?

A) true

B) false

# Using NAT for Defense-in-Depth



## Using PIX NAT for Defense-in-Depth

PIX NAT and access rules are complementary:

- A connection must be permitted by an ACL, AND an inside source translation rule must be built to pass traffic

If no xlate slot cannot be created, even a misconfigured ACL will not permit traffic:

- Dynamic NAT eventually times out and hides once-active clients
- Dynamic PAT is ambiguous by concept and cannot support incoming connections

Therefore, PIX NAT (and especially PAT) does enhance security by minimizing exposure of the internal network

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—5-1-34

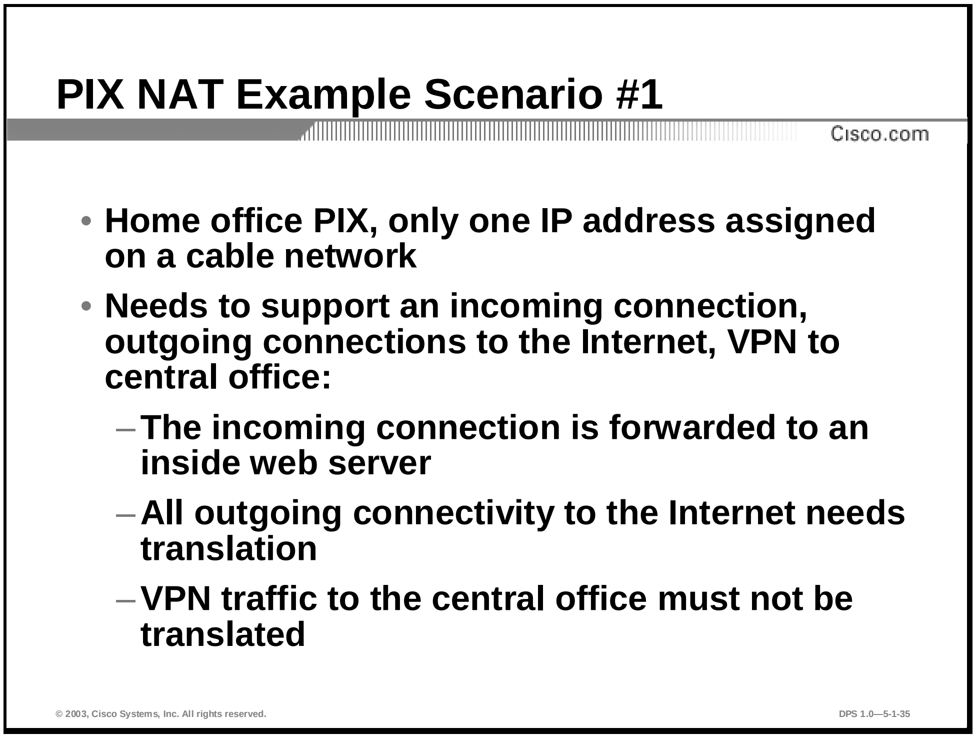
## Objective

This section will enable the learner to identify defense-in-depth features of the PIX Firewall NAT feature product and choose them in a firewall design

## Introduction

For a successful communication a packet must first match the access-rules (e.g. access-lists) and then the translation rules (e.g. a NAT statement). This two-layer philosophy of PIX access control provides nice defense in depth, as even a misconfigured ACL will not permit traffic, if no translation rules can be created.

Also, PIX dynamic NAT provides minimization of exposure for inside hosts, as dynamic translation slots are deleted after an idle timeout. In this way, idle hosts are hidden and unreachable behind the PIX. PAT additionally enhances security, as inbound connections to a PAT slot are ambiguous and are dropped, if no specific static PAT rule exists.



## PIX NAT Example Scenario #1

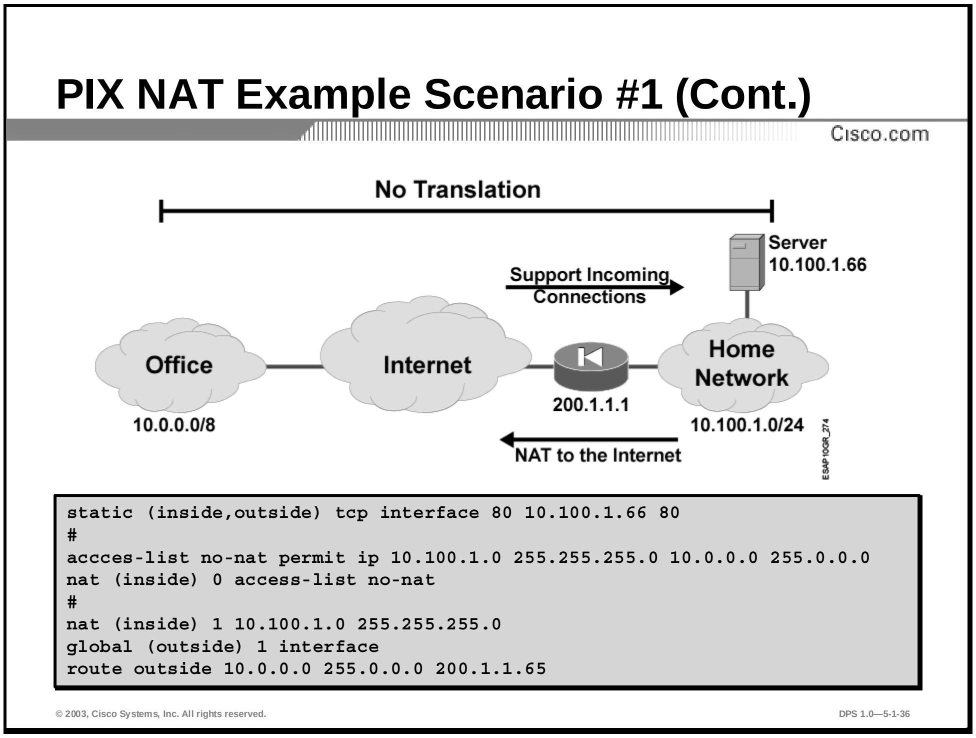
- Home office PIX, only one IP address assigned on a cable network
- Needs to support an incoming connection, outgoing connections to the Internet, VPN to central office:
  - The incoming connection is forwarded to an inside web server
  - All outgoing connectivity to the Internet needs translation
  - VPN traffic to the central office must not be translated

DPS 1.0—5-1-35

## Example Scenario

This is a typical scenario of a PIX protecting a home-office. The cable operator has given a single IP address for the PIX on a cable network. The PIX needs to support an incoming connection to a home web server, outgoing connectivity to the Internet with network address translation, and a VPN connection to central office.

The translation engine can only use a single IP address on the outside interface. The translation rules need to be set up to provide incoming connection forwarding (port forwarding) to the inside web server, translate all outgoing Internet traffic, and not translate VPN traffic to the central location.

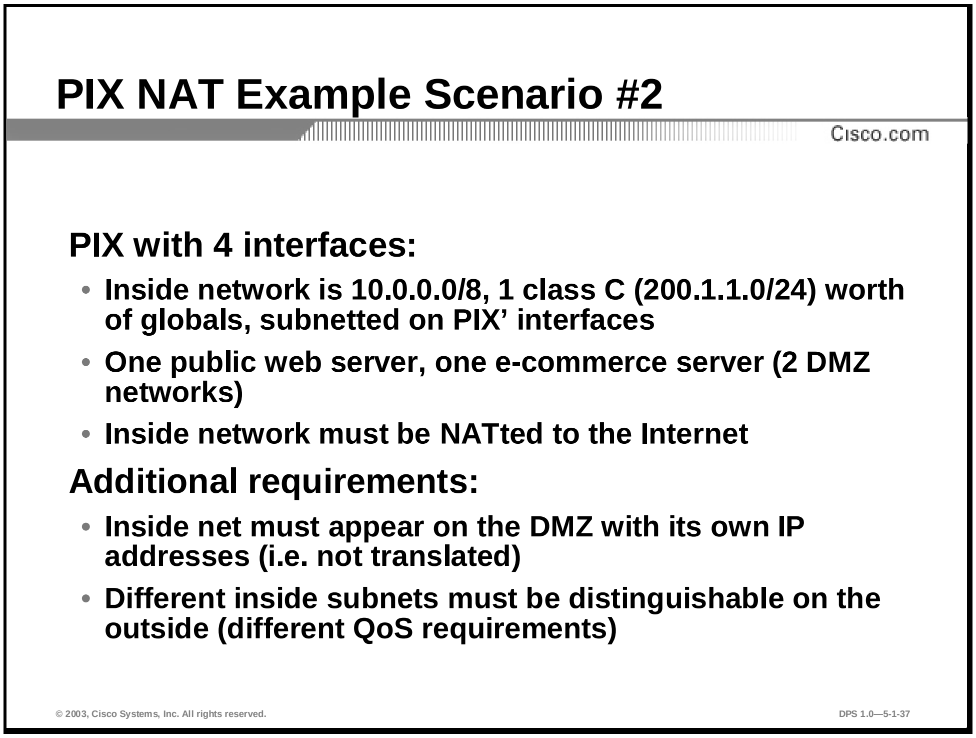


## PIX NAT Example Scenario #1 (Cont.)

```
static (inside,outside) tcp interface 80 10.100.1.66 80
#
accces-list no-nat permit ip 10.100.1.0 255.255.255.0 10.0.0.0 255.0.0.0
nat (inside) 0 access-list no-nat
#
nat (inside) 1 10.100.1.0 255.255.255.0
global (outside) 1 interface
route outside 10.0.0.0 255.0.0.0 200.1.1.65
```

DPS 1.0—5-1-36

This is the basic configuration, showing the most important commands to fulfil the requirements. The port-based static command provides the inbound connection, and is required as only one global IP address is available. The “nat 0 acccess-list” command is provided to ensure there is no translation inside the VPN. Classic PAT to the outside interface provides outbound Internet connectivity.



## PIX NAT Example Scenario #2

**PIX with 4 interfaces:**

- **Inside network is 10.0.0.0/8, 1 class C (200.1.1.0/24) worth of globals, subnetted on PIX' interfaces**
- **One public web server, one e-commerce server (2 DMZ networks)**
- **Inside network must be NATted to the Internet**

**Additional requirements:**

- **Inside net must appear on the DMZ with its own IP addresses (i.e. not translated)**
- **Different inside subnets must be distinguishable on the outside (different QoS requirements)**

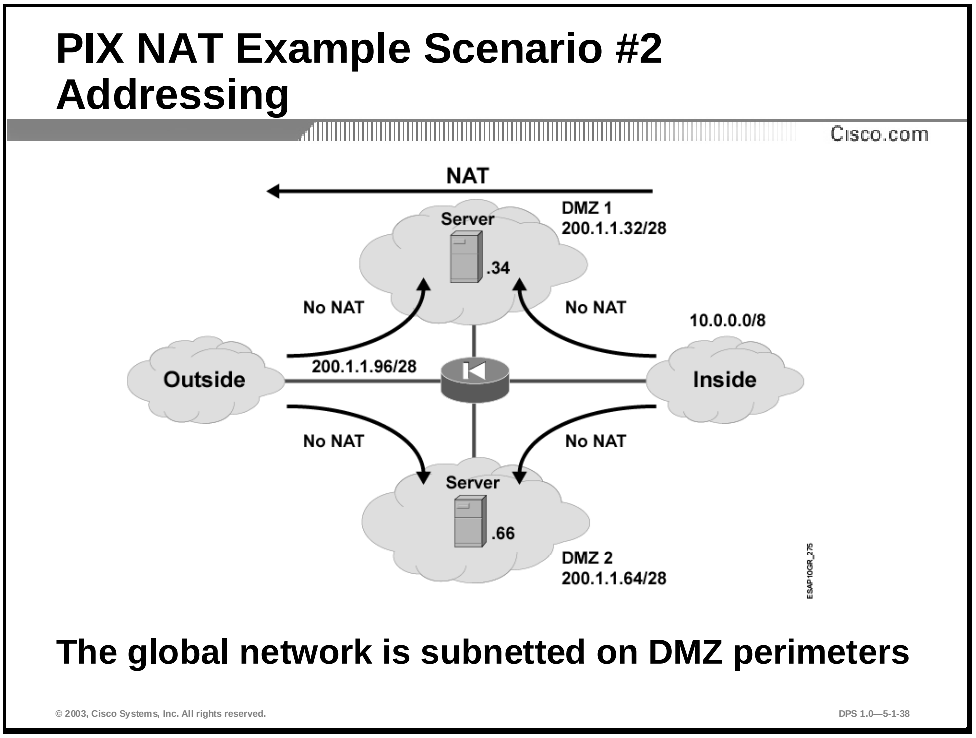
DPS 1.0—5-1-37

## Example Scenario

This example scenario uses a PIX Firewall in an enterprise-firewall setup. A PIX Firewall with four interfaces is used to connect a medium-sized corporation to the Internet.

The PIX Firewall uses the four interfaces to connect the inside (corporate) network to the outside (Internet), and provides two DMZ networks to host public servers. The organization has a 24-bit prefix of global addresses assigned to them (200.1.1.0/24), and divides this prefix into smaller prefixes, which are allocated on PIX Interfaces. Therefore, only the inside network uses private addressing, the outside PIX Firewall interface and the DMZ networks are addressed with public addresses.

The DMZ networks need to host one public server each (web server, E-commerce server). The inside network must be NAT-ted to the Internet. There is a requirement that all inside networks must appear on the DMZ networks with their real addressing – i.e. no translation needs to be performed between the inside and DMZ interfaces. Another requirement is that different groups of inside users should be distinguishable on the outside network, as quality-of-service mechanisms need to be provisioned on the Internet link.

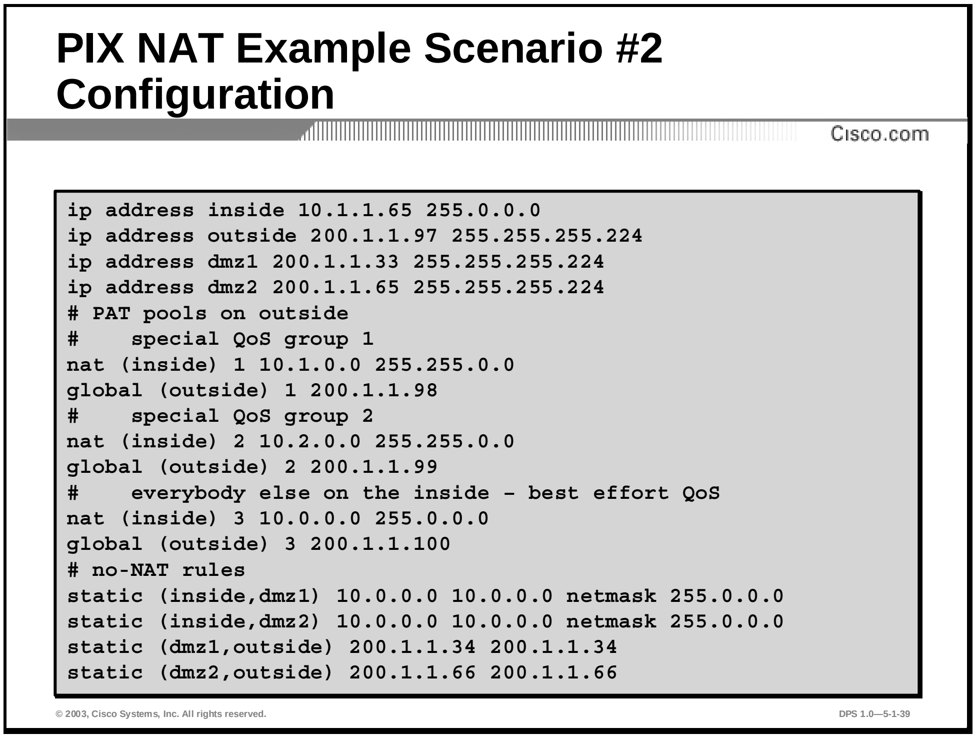


# PIX NAT Example Scenario #2 Addressing

**The global network is subnetted on DMZ perimeters**

DPS 1.0—5-1-38

This figure shows the details of DMZ addressing. A single /24 prefix is split in smaller prefixes, assigned to DMZs to simplify NAT for public servers.



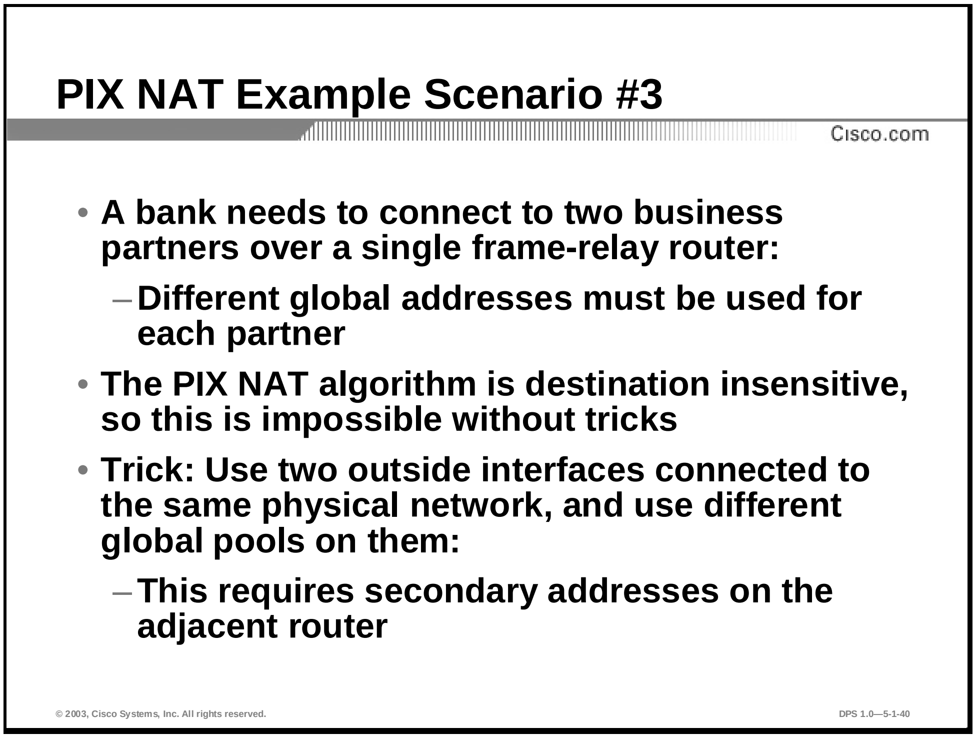
## PIX NAT Example Scenario #2 Configuration

Cisco.com

```
ip address inside 10.1.1.65 255.0.0.0
ip address outside 200.1.1.97 255.255.255.224
ip address dmz1 200.1.1.33 255.255.255.224
ip address dmz2 200.1.1.65 255.255.255.224
# PAT pools on outside
#    special QoS group 1
nat (inside) 1 10.1.0.0 255.255.0.0
global (outside) 1 200.1.1.98
#    special QoS group 2
nat (inside) 2 10.2.0.0 255.255.0.0
global (outside) 2 200.1.1.99
#    everybody else on the inside - best effort QoS
nat (inside) 3 10.0.0.0 255.0.0.0
global (outside) 3 200.1.1.100
# no-NAT rules
static (inside,dmz1) 10.0.0.0 10.0.0.0 netmask 255.0.0.0
static (inside,dmz2) 10.0.0.0 10.0.0.0 netmask 255.0.0.0
static (dmz1,outside) 200.1.1.34 200.1.1.34
static (dmz2,outside) 200.1.1.66 200.1.1.66
```

DPS 1.0—5-1-39

This is the PIX NAT configuration for this example. Network statics are used to perform identity translation between inside and DMZ, satisfying the original goal of no translation between inside and the DMZs. Multiple groups of NAT (multiple “nat” statements) are used to match different inside subnets, and differentiate users on the outside network using different global pools on the outside interface.



## PIX NAT Example Scenario #3

Cisco.com

- A bank needs to connect to two business partners over a single frame-relay router:
  - Different global addresses must be used for each partner
- The PIX NAT algorithm is destination insensitive, so this is impossible without tricks
- Trick: Use two outside interfaces connected to the same physical network, and use different global pools on them:
  - This requires secondary addresses on the adjacent router

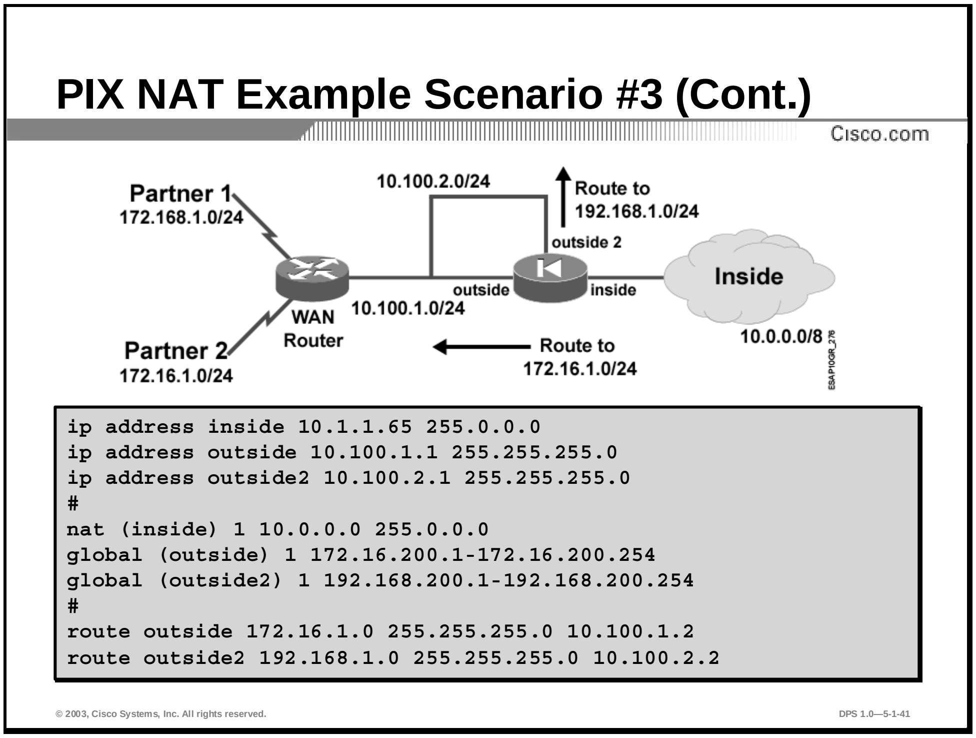
DPS 1.0—5-1-40

# Example Scenario

A bank needs to connect to two business partners over a single frame-relay router, and their requirements for NAT are:

- Different global addresses must be used for each partner

The PIX NAT algorithm is destination insensitive, so this is impossible without tricks. We will use two outside interfaces connected to the same physical network, and use different global pools on them to satisfy the requirements. This requires overlapping subnets on the outside physical network, and secondary addresses on the adjacent router



## PIX NAT Example Scenario #3 (Cont.)

Cisco.com

```
ip address inside 10.1.1.65 255.0.0.0
ip address outside 10.100.1.1 255.255.255.0
ip address outside2 10.100.2.1 255.255.255.0
#
nat (inside) 1 10.0.0.0 255.0.0.0
global (outside) 1 172.16.200.1-172.16.200.254
global (outside2) 1 192.168.200.1-192.168.200.254
#
route outside 172.16.1.0 255.255.255.0 10.100.1.2
route outside2 192.168.1.0 255.255.255.0 10.100.2.2
```

DPS 1.0—5-1-41

Here are the details of the setup and the configuration. Note the same network being translated in two pools, depending on the destination network.

## Practice

Q1) How can translation of inside hosts depend on different destinations, which are reached through the same next hop?

A) static command and disabling proxy ARP

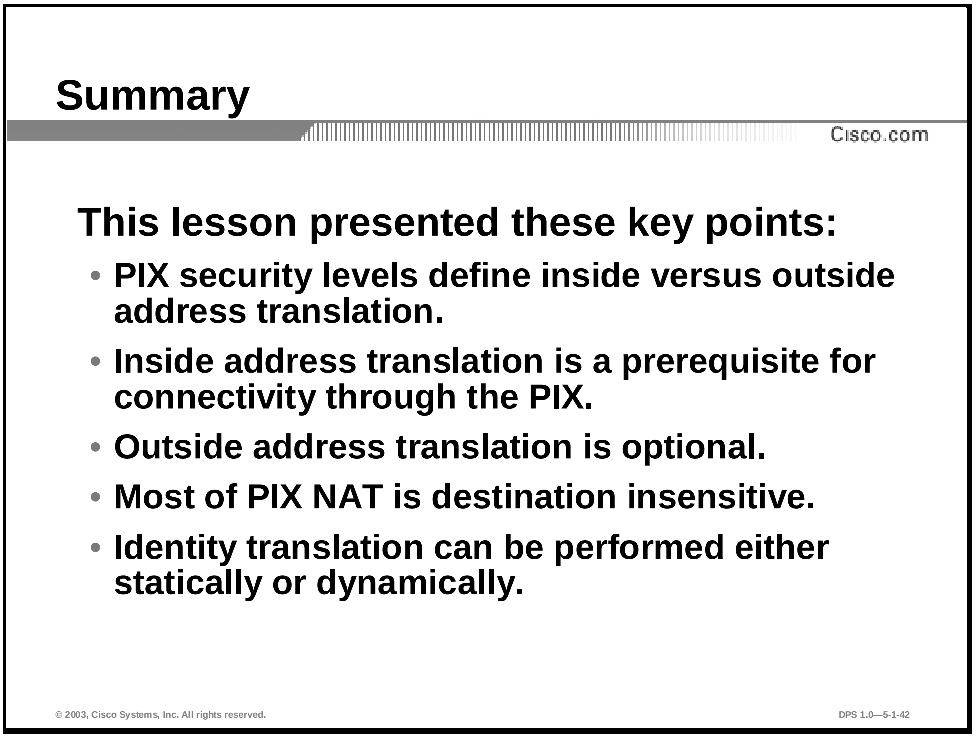
B) connecting two outside interfaces to the same network with different global pools

C) NAT command with the access-list option

D) two global pools on the same outbound interface

# Summary

This section summarizes the key points discussed in this lesson.



## Summary

Cisco.com

**This lesson presented these key points:**

- **PIX security levels define inside versus outside address translation.**
- **Inside address translation is a prerequisite for connectivity through the PIX.**
- **Outside address translation is optional.**
- **Most of PIX NAT is destination insensitive.**
- **Identity translation can be performed either statically or dynamically.**

DPS 1.0—5-1-42

## Next Steps

After completing this lesson, go to:

- Understanding PIX Firewall Security lesson

# Quiz: Understanding PIX Firewall NAT

Complete the quiz to assess what you have learned in this lesson.

## Objectives

This quiz tests your knowledge on how to:

- Identify and configure advanced NAT features
- Identify NAT limitations of the Cisco Secure PIX Firewall product when using it in a firewall system design

## Instructions

Answer these questions:

1. When are dynamic and static translations created?
2. When do dynamic and static translations time out?
3. What are the two possibilities to perform no network address translation on the PIX Firewall?
4. Which PIX NAT function is destination-address-sensitive?
5. For which addresses does the PIX Firewall proxy ARP?

## Scoring

You have successfully completed the quiz for this lesson when you earn a score of 80 percent or better.

# Understanding PIX Firewall Security

## Overview

### Importance

The PIX Firewall provides access control through the Adaptive Security Algorithm (ASA). This lesson presents advanced implementation and configuration guidelines for access control in the PIX Firewall, and is as such extremely important when designing advanced firewalling solutions.

### Lesson Objective

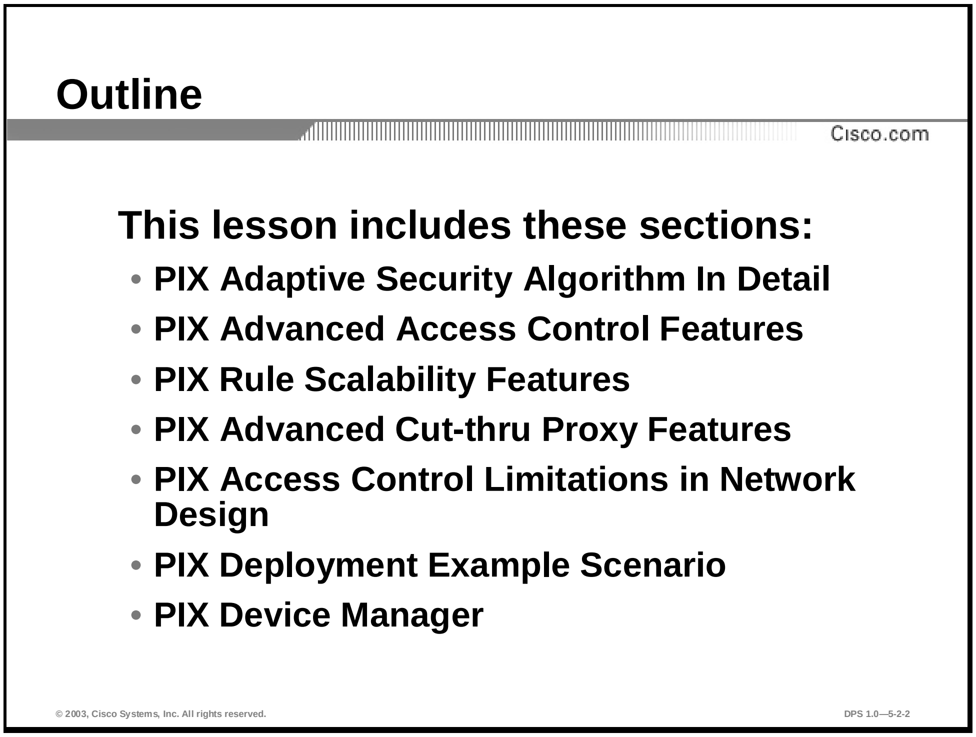
This lesson will enable the learner to identify and configure advanced Adaptive Security Algorithm (ASA) features, and identify ASA limitations of the Cisco Secure PIX Firewall product, when using it in a firewall system design.

## Learner Skills and Knowledge

To fully benefit from this lesson, you must have these prerequisite skills and knowledge:

- A solid understanding of the PIX Firewall NAT functionality

## Outline



### Outline

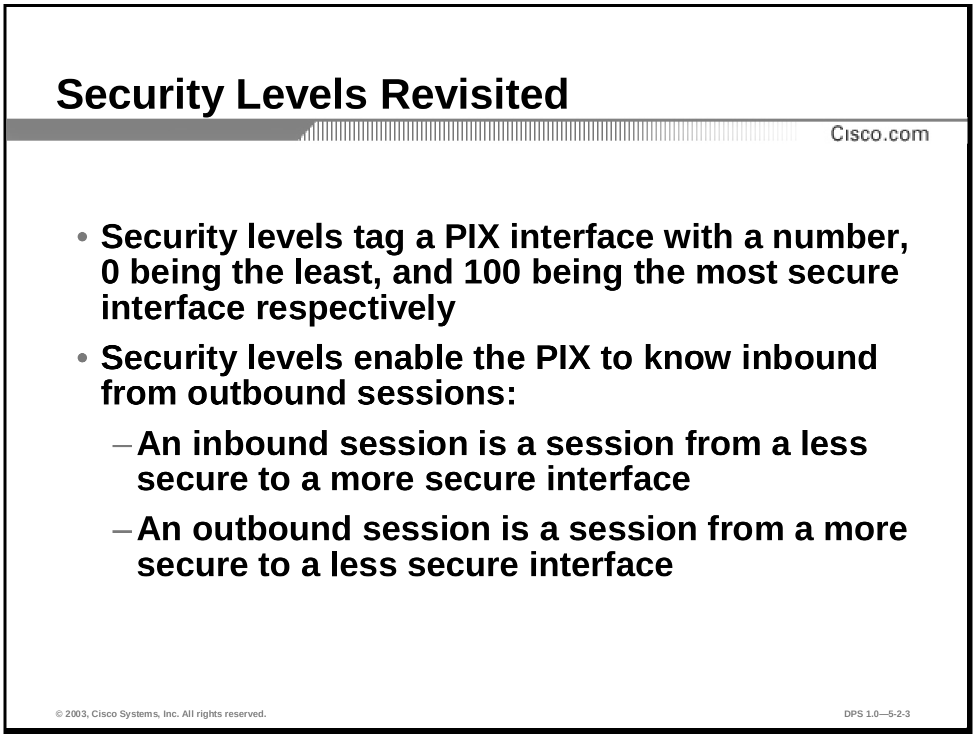
Cisco.com

**This lesson includes these sections:**

- **PIX Adaptive Security Algorithm In Detail**
- **PIX Advanced Access Control Features**
- **PIX Rule Scalability Features**
- **PIX Advanced Cut-thru Proxy Features**
- **PIX Access Control Limitations in Network Design**
- **PIX Deployment Example Scenario**
- **PIX Device Manager**

DPS 1.0—5-2-2

# Overview



## Security Levels Revisited

Cisco.com

- Security levels tag a PIX interface with a number, 0 being the least, and 100 being the most secure interface respectively
- Security levels enable the PIX to know inbound from outbound sessions:
  - An inbound session is a session from a less secure to a more secure interface
  - An outbound session is a session from a more secure to a less secure interface

DPS 1.0—5-2-3

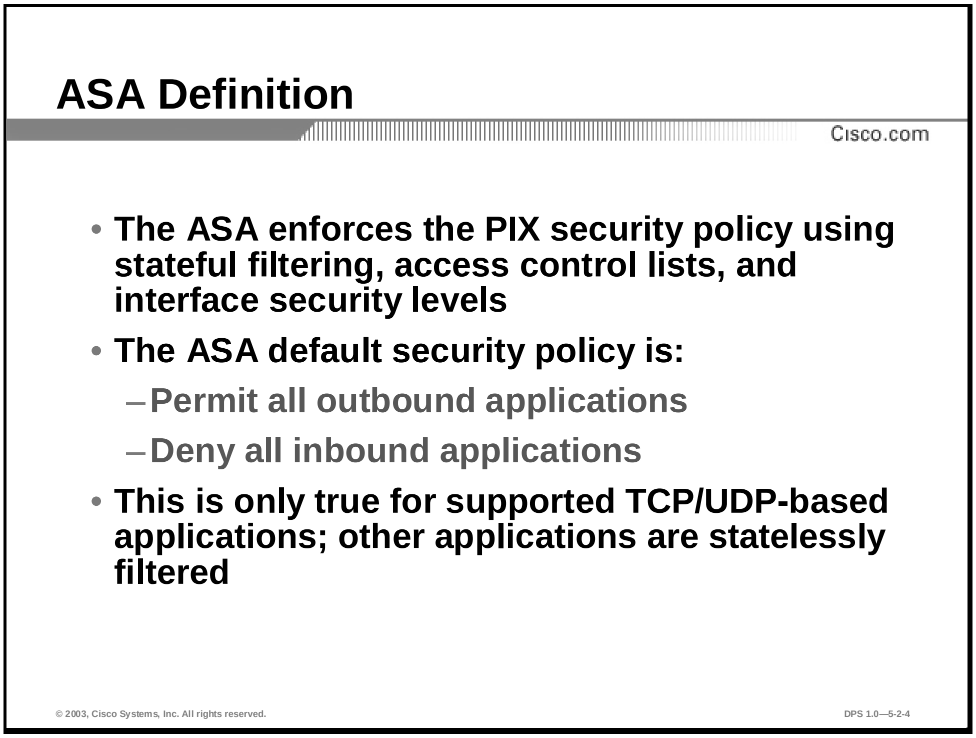
The configuration of every PIX default to an inside interface with a level of 100 and an outside interface with a level of 0. There’s nothing more secure than the internal net, and nothing less secure than the external net.

On PIX operating system 6.0 and later, the default interface names can actually be modified, but generally there is no reason to.

Additional interfaces will be configured with separate security levels somewhere between 1 and 99.

By default, all communications are permitted in an outbound direction, from a more secure level to a less secure level. By default all communications are prohibited in an inbound direction, from a less secure level to a more secure level.

# PIX Adaptive Security Algorithm In Detail



## ASA Definition

Cisco.com

- The ASA enforces the PIX security policy using stateful filtering, access control lists, and interface security levels
- The ASA default security policy is:
  - Permit all outbound applications
  - Deny all inbound applications
- This is only true for supported TCP/UDP-based applications; other applications are statelessly filtered

DPS 1.0—5-2-4

## Objective

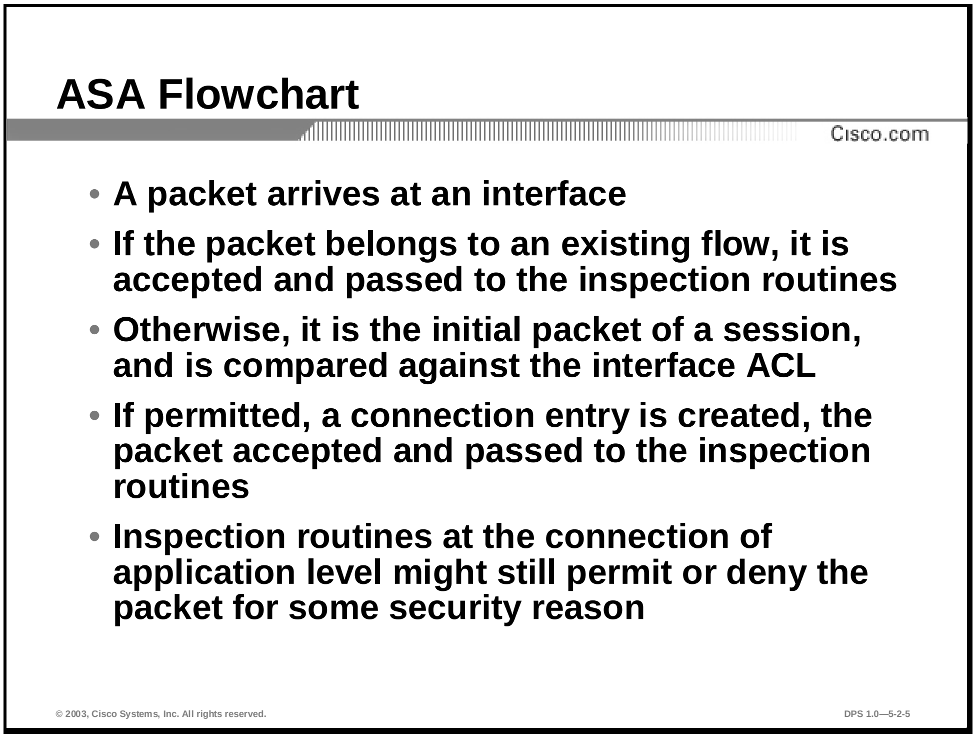
This section will enable the learner to explain how the concept of the Adaptive Security Algorithm level influences all aspects of access control in the PIX Firewall.

## Introduction

The Adaptive Security Algorithm provides stateful filtering on connections. This means that the “state” of a connection is maintained, automatically permitting return packets for established connections whether from TCP or UDP protocols.

During this phase, TCP sequence numbers are further randomized and matched as a mechanism to further prevent session hijacking.

The ASA tracks source and destination addresses, source and destination ports, TCP sequences as well as additional TCP flags for enhanced integrity of the algorithm.



## ASA Flowchart

Cisco.com

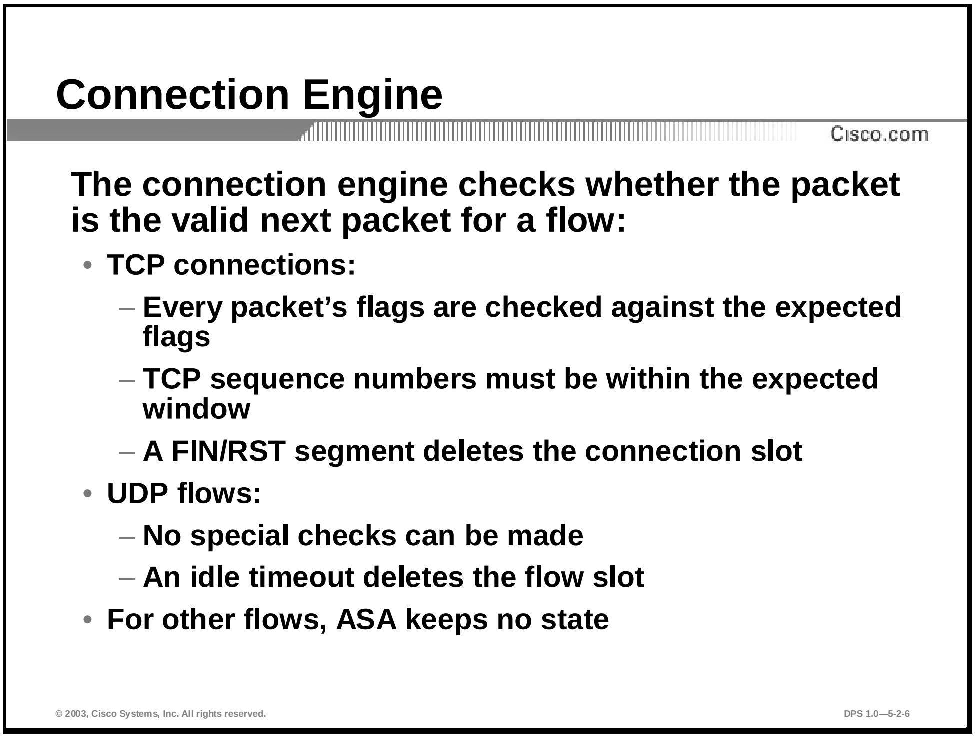
- A packet arrives at an interface
- If the packet belongs to an existing flow, it is accepted and passed to the inspection routines
- Otherwise, it is the initial packet of a session, and is compared against the interface ACL
- If permitted, a connection entry is created, the packet accepted and passed to the inspection routines
- Inspection routines at the connection of application level might still permit or deny the packet for some security reason

DPS 1.0—5-2-5

The ASA handles packet verification as part of the process of deciding to allow information transfer. If a packet is part of a maintained conversation, then information is allowed to flow.

If there is no existing state on an incoming packet/conversation, then a process of comparing ACL and/or conduit information is used to determine whether to permit the packet flow or not.

If the flow is to be permitted, then a new state maintenance entry is created for future packets.



## Connection Engine

Cisco.com

**The connection engine checks whether the packet is the valid next packet for a flow:**

- **TCP connections:**
  - **Every packet’s flags are checked against the expected flags**
  - **TCP sequence numbers must be within the expected window**
  - **A FIN/RST segment deletes the connection slot**
- **UDP flows:**
  - **No special checks can be made**
  - **An idle timeout deletes the flow slot**
- **For other flows, ASA keeps no state**

DPS 1.0—5-2-6

# The PIX Connection Engine

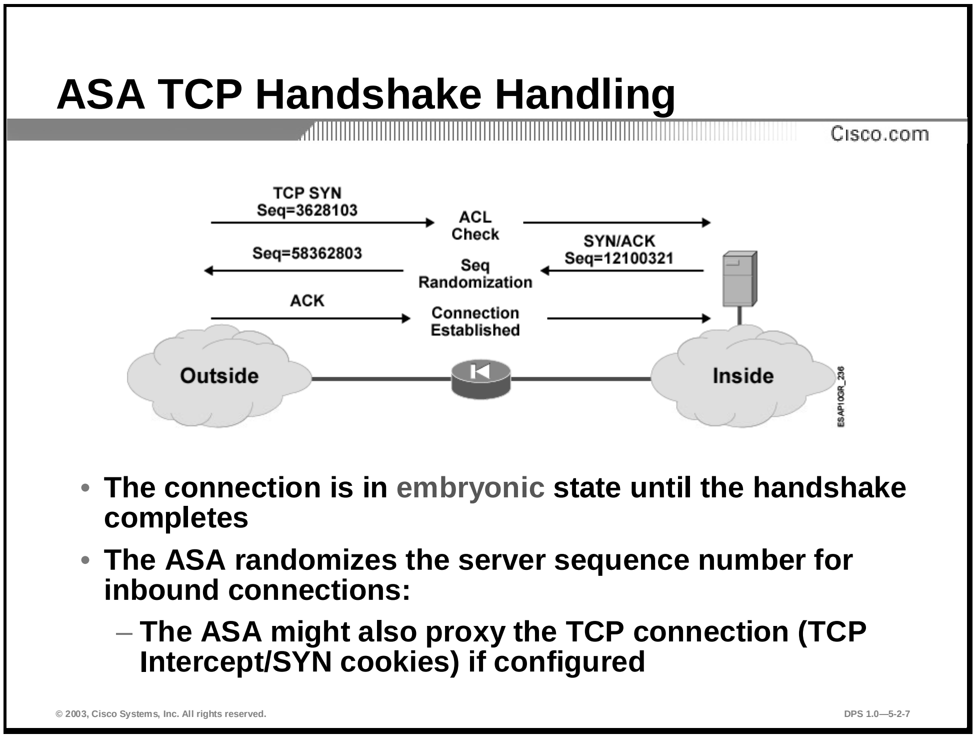
Additional verifications take place in TCP conversations, with other flag options including sequence numbers within the expected window.

In the course of a normal TCP conversation, a FIN or RST flag will delete the connection slot from the ASA’s state table.

With UDP, there is no trackable information beyond source/destination port number and IP addresses. An idle timeout (configurable) on the flow will delete the connection slot from the ASA’s state table.

Idle timers will be explored later, but can apply to TCP sessions as well.

For flows other than TCP/UDP, there may not be any specific ASA state information. Most commonly, this applies to ICMP packets. For example, an outbound ICMP echo does not automatically permit an incoming ICMP echo-reply.



## ASA TCP Handshake Handling

Cisco.com

- The connection is in embryonic state until the handshake completes
- The ASA randomizes the server sequence number for inbound connections:
  - The ASA might also proxy the TCP connection (TCP Intercept/SYN cookies) if configured

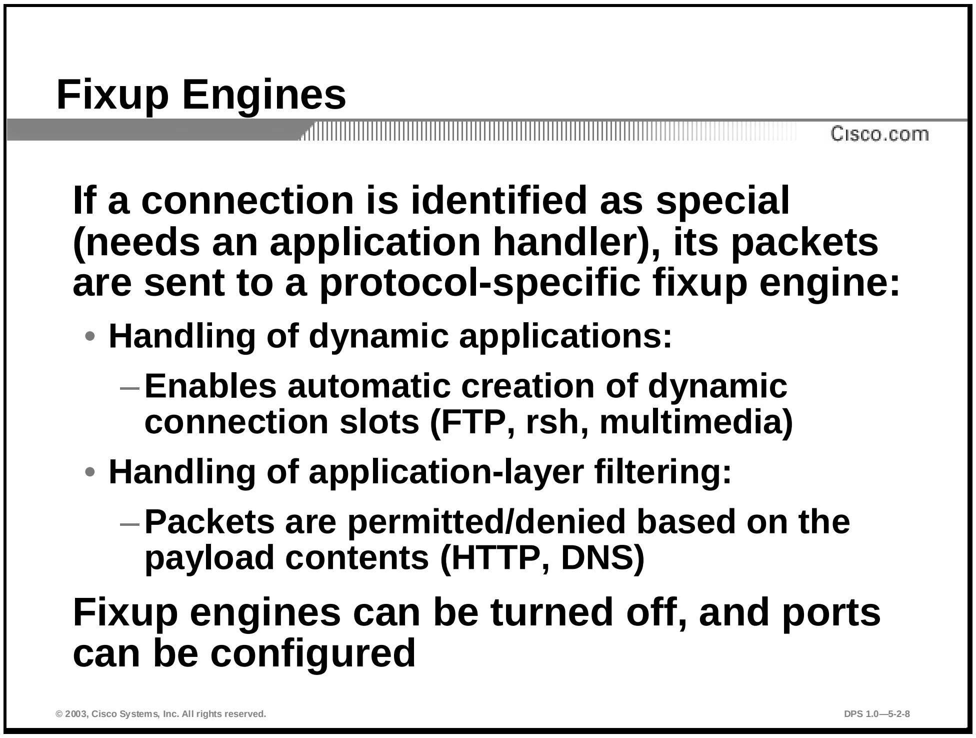
DPS 1.0—5-2-7

Embryonic connections are also known as half-open connections. The PIX maintains entries of embryonic connections as well as counts of them.

Through normal configuration, it is possible to monitor how many simultaneous embryonic TCP sessions exist and begin removing them as well to prevent a form of DoS attack.

If the PIX determines an attack and begins deleting embryonic connections, it will also send an RST flag to the destination station in order to remove connections there as well. This is the behavior of PIX/OS 5.1 and earlier.

The TCP Intercept feature (PIX OS 5.2 and later) provides additional protection. Once the SYN threshold is reached, and until the SYN count falls below the minimum threshold, every incoming SYN is intercepted. For every incoming SYN at that point, the PIX will respond (on behalf of the destination station) with an empty SYN/ACK. If the final ACK is received (valid three-way handshake) then the PIX will initiate with the destination station on behalf of the original sender. The TCP conversation continues as normal.



## Fixup Engines

Cisco.com

**If a connection is identified as special (needs an application handler), its packets are sent to a protocol-specific fixup engine:**

- **Handling of dynamic applications:**
  - **Enables automatic creation of dynamic connection slots (FTP, rsh, multimedia)**
- **Handling of application-layer filtering:**
  - **Packets are permitted/denied based on the payload contents (HTTP, DNS)**

**Fixup engines can be turned off, and ports can be configured**

DPS 1.0—5-2-8

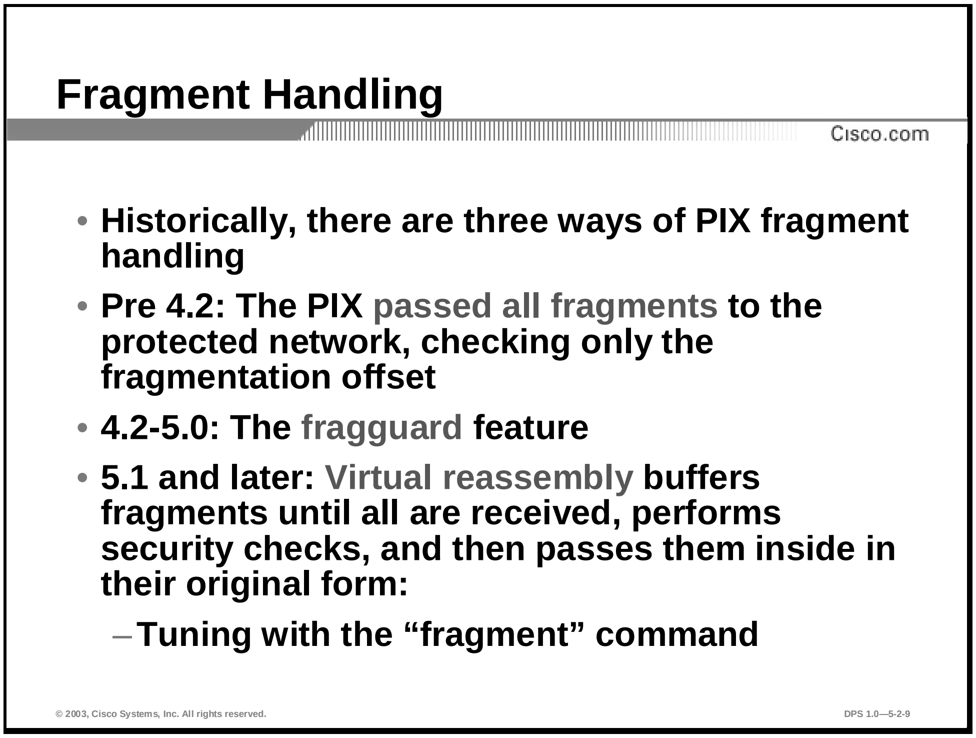
# PIX Fixup Engines

Not all applications perform within RFC specifications for NAT. This means that the source/destination IP address information should only exist within the IP header of a packet, allowing NAT to continue seamlessly on either end.

Many applications, including popular common protocols like HTTP, SMTP and FTP do not follow those rules and will embed IP addresses in PDU's higher than layer 3. The Fixup Protocol operation fixes problems like that. By default, these are enabled at common ports.

The fixup handles also provide security functionality – all dynamic protocols, where the PIX must monitor negotiation of additional sesstions, pass through the fixup handlers, which analyze the application protocol to detect negotiations. All application-layer filtering, such as ActiveX, Java, and URL filtering, is also handled by the HTTP fixup engine.

The fixup engine can be turned off, or the intended ports can be changed as well. The number of applications supported with the fixup engine grows. Consult the PIX version readme files for accurate documentation.



## Fragment Handling

Cisco.com

- Historically, there are three ways of PIX fragment handling
- Pre 4.2: The PIX passed all fragments to the protected network, checking only the fragmentation offset
- 4.2-5.0: The fragguard feature
- 5.1 and later: Virtual reassembly buffers fragments until all are received, performs security checks, and then passes them inside in their original form:
  - Tuning with the “fragment” command

DPS 1.0—5-2-9

## PIX Fragment Handling

In general, the default values should be used. However, if a large percentage of the network traffic through the PIX Firewall is NFS, additional tuning may be necessary to avoid database overflow.

Since the virtual reassembly process has been introduced, the effects of fragment attacks on end stations has been greatly reduced. The PIX maintains a part of memory for buffer space on maintaining fragments for reassembly.

The size and performance of this database area can be maintained with the **fragment** command in PIX OS 5.1 and later.

**fragment size** *database-limit [interface]*

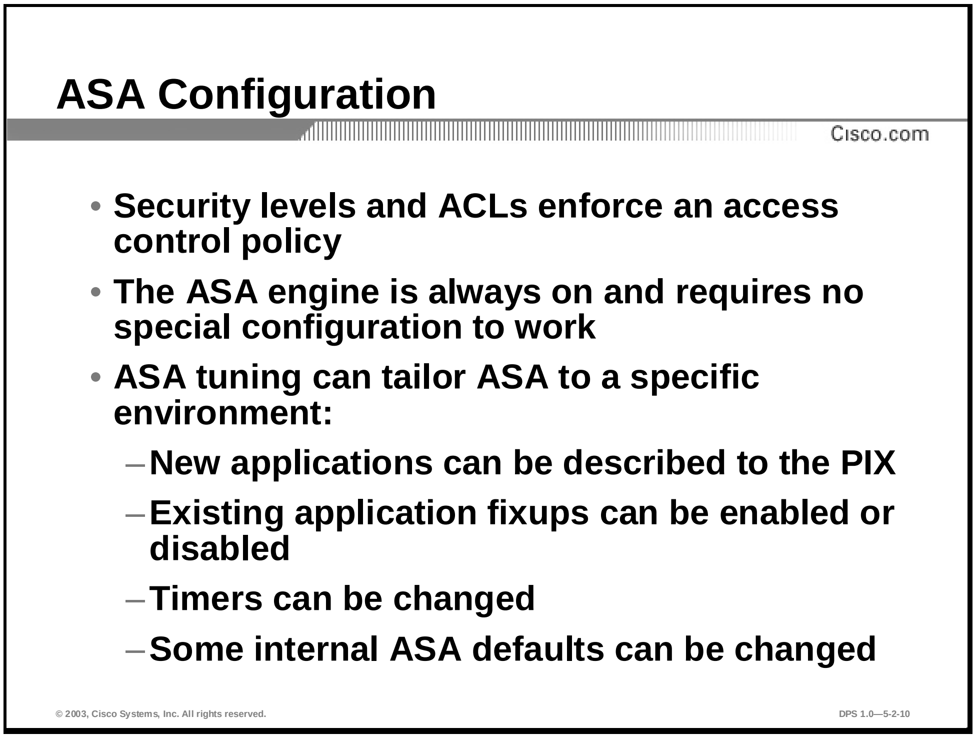
**fragment chain** *chain-limit [interface]*

**fragment timeout** *seconds [interface]*

The database limit defaults to 200 blocks of memory. It can be expanded to 1,000,000 blocks.

The chain limit defaults to a maximum of 24 fragments per chain. The maximum is 8200.

The default timeout of a fragment to completely arrive is 5 seconds. The maximum is 30 seconds. To see current status, use the **show fragment** *[interface]* command.



## ASA Configuration

- Security levels and ACLs enforce an access control policy
- The ASA engine is always on and requires no special configuration to work
- ASA tuning can tailor ASA to a specific environment:
  - New applications can be described to the PIX
  - Existing application fixups can be enabled or disabled
  - Timers can be changed
  - Some internal ASA defaults can be changed

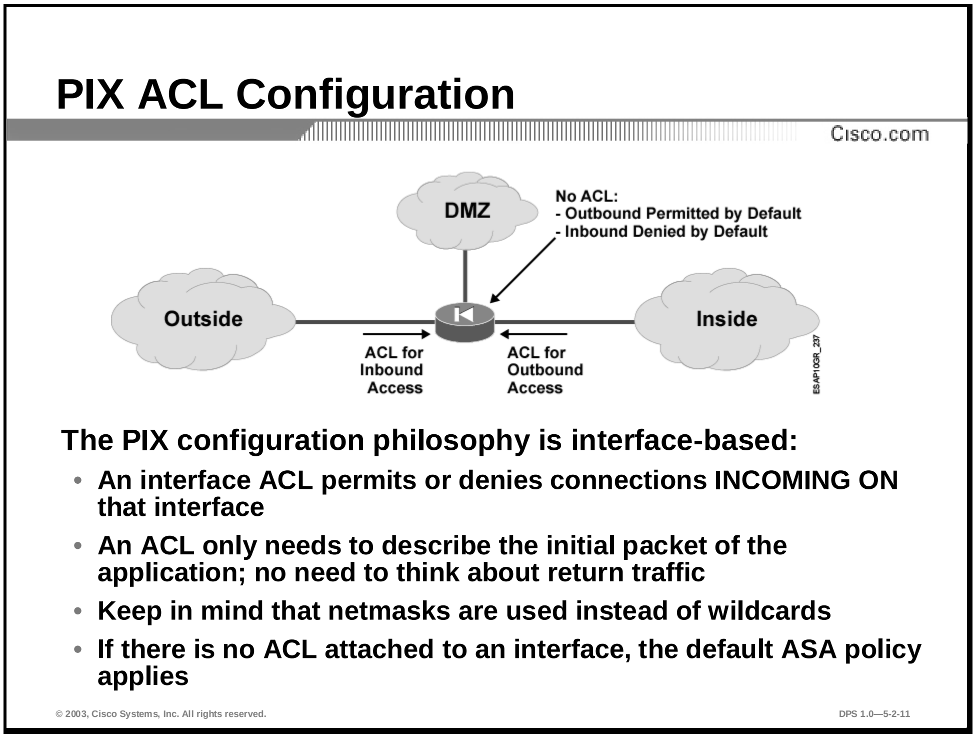
DPS 1.0—5-2-10

# ASA Configuration

Older versions used **conduits** to set permissions between interfaces in an inbound direction. Newer versions of PIX OS can use access-lists to be configured like routers.

The security levels of an interface are used to set permissions for access. Remember the defaults where everything from a higher-security interface to a lower-security interface is permitted. The opposite direction is denied by default.

Almost everything within the PIX configuration is changeable. The firewall can be set to be as secure or as permissive as necessary. If everything is permitted, many people say that the PIX is not doing its job as a firewall. That is partially true. While not doing any job to restrict access, it still maintains all state information and protects against various types of DoS attacks. So a completely permissive firewall IS still a firewall, it is just not living up to its potential!



## PIX ACL Configuration

**The PIX configuration philosophy is interface-based:**

- **An interface ACL permits or denies connections INCOMING ON that interface**
- **An ACL only needs to describe the initial packet of the application; no need to think about return traffic**
- **Keep in mind that netmasks are used instead of wildcards**
- **If there is no ACL attached to an interface, the default ASA policy applies**

© 2003, Cisco Systems, Inc. All rights reserved.  DPS 1.0—5-2-11

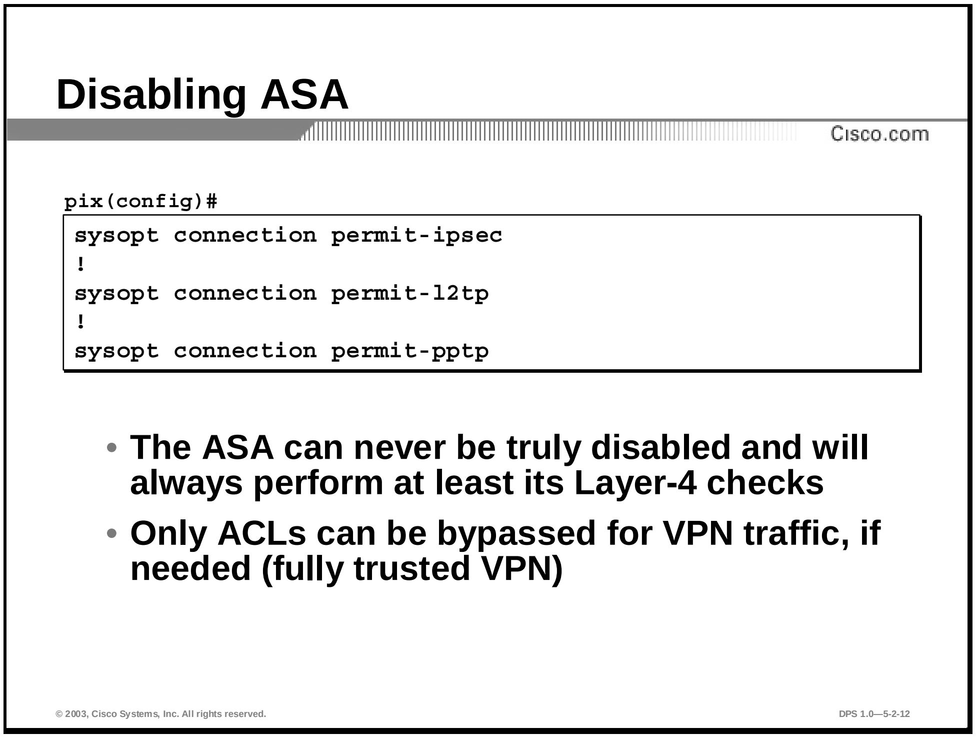
## PIX ACL Configuration Philosophy

Since the ASA access check applies only to the initial packet of a conversation, the access-lists are only evaluated once per connection. If the application changes port information after the initial packet exchange, the ASA will pick up on that, so no additional configuration is necessary on the PIX (unlike a classic router’s ACL).

Normal netmasks are used. If a wildcard netmask is accidentally entered, the PIX is nice enough to replace the appropriate netmask information.

Access-lists, as will be discussed, can work in both directions. Once an access-list is configured, it is activated with an **access-group** command:

**access-group** *acl_name* **in interface** *interface-name*



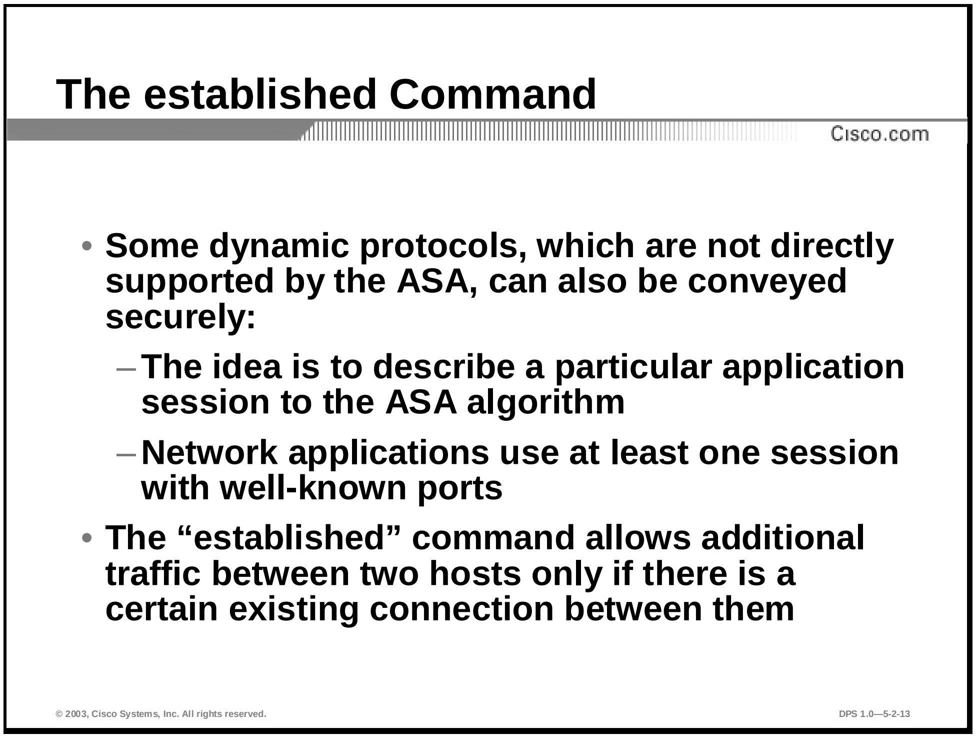
## Disabling ASA

```
pix(config)#
sysopt connection permit-ipsec
!
sysopt connection permit-l2tp
!
sysopt connection permit-pptp
```

- The ASA can never be truly disabled and will always perform at least its Layer-4 checks
- Only ACLs can be bypassed for VPN traffic, if needed (fully trusted VPN)

# Disabling ASA

The sysopt connection command can be used to selectively disable the ASA features. With this command, used for IPSec, PPTP or L2TP packets, the packet will be handed off to a specific area of software code for further handling. This code is outside the typical ASA process.



## The established Command

- Some dynamic protocols, which are not directly supported by the ASA, can also be conveyed securely:
  - The idea is to describe a particular application session to the ASA algorithm
  - Network applications use at least one session with well-known ports
- The “established” command allows additional traffic between two hosts only if there is a certain existing connection between them

DPS 1.0—5-2-13

# Customizing ASA

The **established** command allows additional connections to be opened through a PIX Firewall, if an already established connection is present in the PIX Firewall connection table.

In the command syntax, the first protocol, destination port, and optional source port specified are for the initial outbound connection. The **permitto** and **permitfrom** options refine the return inbound connection.

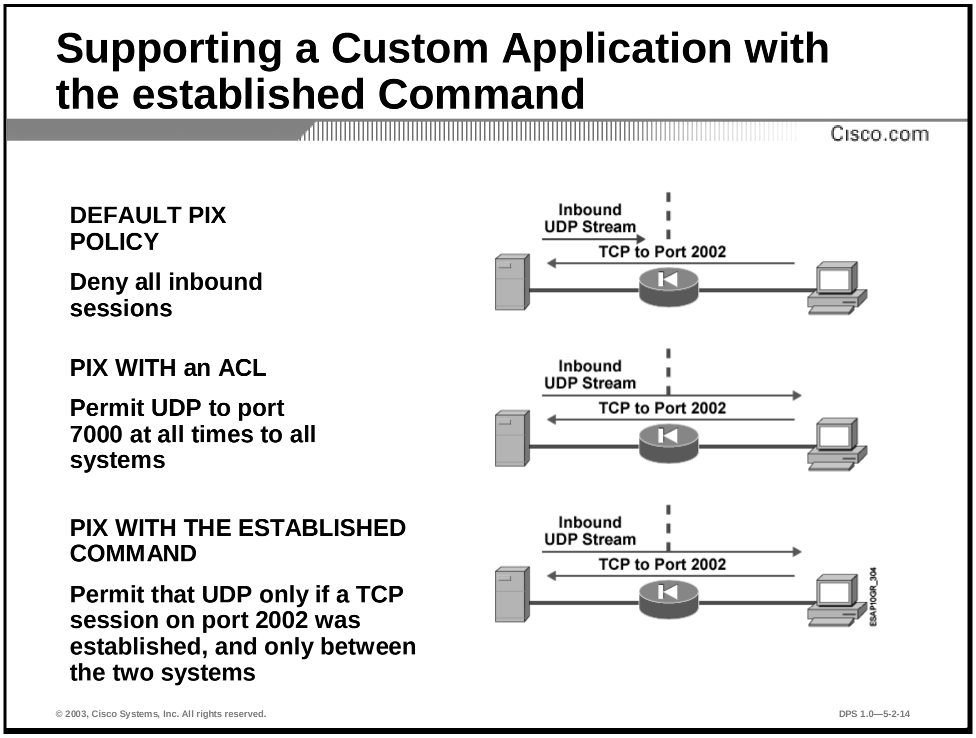
The **permitto** option lets you specify a new protocol or port for the return connection at the PIX Firewall.

The **permitfrom** option lets you specify a new protocol or port at the remote server.

The **no established** command disables the **established** feature.

The **clear established** command removes all **establish** command statements from your configuration.

**Note** The **established** command cannot be used with Port Address Translation (PAT).



## Supporting a Custom Application with the established Command

Cisco.com

**DEFAULT PIX POLICY**

**Deny all inbound sessions**

**PIX WITH an ACL**

**Permit UDP to port 7000 at all times to all systems**

**PIX WITH THE ESTABLISHED COMMAND**

**Permit that UDP only if a TCP session on port 2002 was established, and only between the two systems**

DPS 1.0—5-2-14

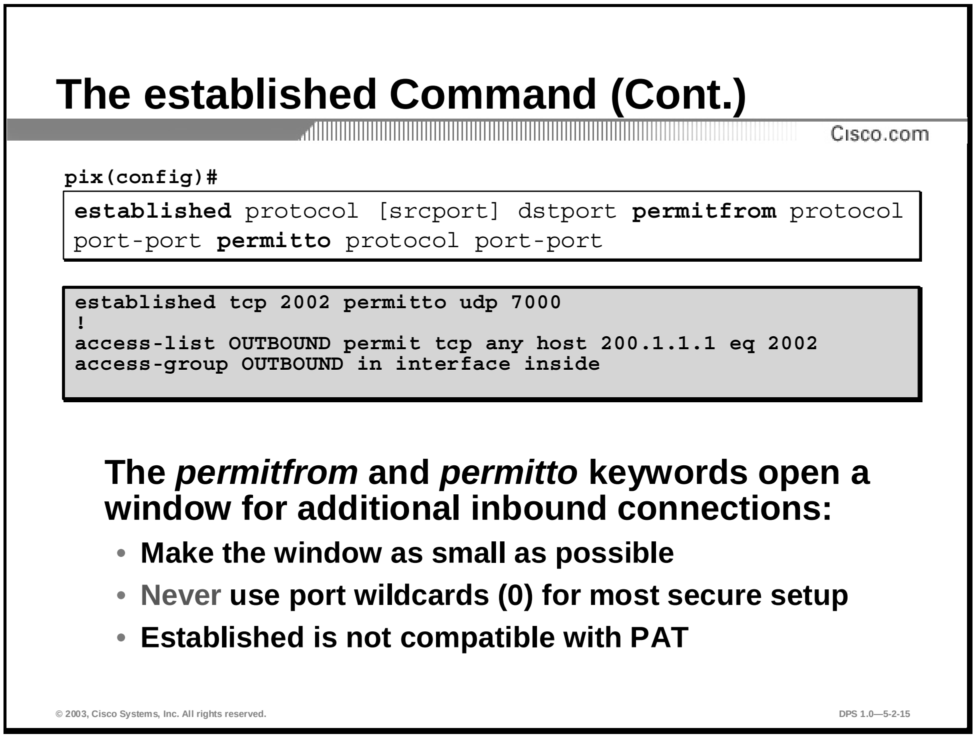
In this figure, an example application running across a firewall opens a single control TCP connection from the client to the server, which opens additional inbound UDP sessions to the client. This is typical behavior of some multimedia streaming protocols.

The protocol in the example uses a random client port and the server port of 2002 for the TCP control session. For inbound UDP sessions, the server uses a random port, and the fixed client port of 7000.

Let’s assume that this application protocol is not known to the PIX Firewall software, therefore by default, the outbound TCP session will be permitted (the default PIX Firewall security-level-based policy), and the inbound UDP stream denied, as the PIX Firewall cannot identify the inbound UDP session as a part of an outgoing application.

One possible solution would be to permit all inbound traffic using an access list, but such an approach would create a permanent opening in the firewall, which is not desirable.

An alternative approach, using the “established” command, can conditionally permit the inbound UDP session, if the outgoing TCP session has already been established.



## The established Command (Cont.)

```
pix(config)#
established protocol [srcport] dstport permitfrom protocol
port-port permitto protocol port-port
```

```
established tcp 2002 permitto udp 7000
!
access-list OUTBOUND permit tcp any host 200.1.1.1 eq 2002
access-group OUTBOUND in interface inside
```

**The *permitfrom* and *permitto* keywords open a window for additional inbound connections:**

- **Make the window as small as possible**
- **Never use port wildcards (0) for most secure setup**
- **Established is not compatible with PAT**

DPS 1.0—5-2-15

The **established** command works as shown in the following format:

**established** A B C **permitto** D E **permitfrom** D F

This command works as though it were written “If there exists a connection between two hosts using protocol A from src port B destined for port C, permit return connections through the PIX Firewall via protocol D (D can be different from A), if the source port(s) correspond to F and the destination port(s) correspond to E.”

For example:

**established tcp 6060 0 permitto tcp 6061 permitfrom tcp 6059**

In this case, if a connection is started by an internal host to an external host using TCP source port 6060 and any destination port, the PIX Firewall permits return traffic between the hosts via TCP destination port 6061 and TCP source port 6059.

For example:

**established udp 0 6060 permitto tcp 6061 permitfrom tcp 1024-65535**

In this case, if a connection is started by an internal host to an external host using UDP destination port 6060 and any source port, the PIX Firewall permits return traffic between the hosts via TCP destination port 6061 and TCP source port 1024-65535.

### Security Problem

The **established** command has been enhanced to optionally specify the destination port used for connection lookups. Only the source port could be specified previously with the destination port being 0 (a wildcard). This addition allows more control over the command and provides support for protocols where the destination port is known, but the source port is not.

The **established** command can potentially open a large security hole in the PIX Firewall if not used with discretion. Whenever you use this command, if possible, also use the **permitto** and **permitfrom** options to indicate ports to which and from which access is permitted. Without these options, external systems to which connections are made could make unrestricted connections to the internal host involved in the connection. The following are examples of potentially serious security violations that could be allowed when using the **established** command.

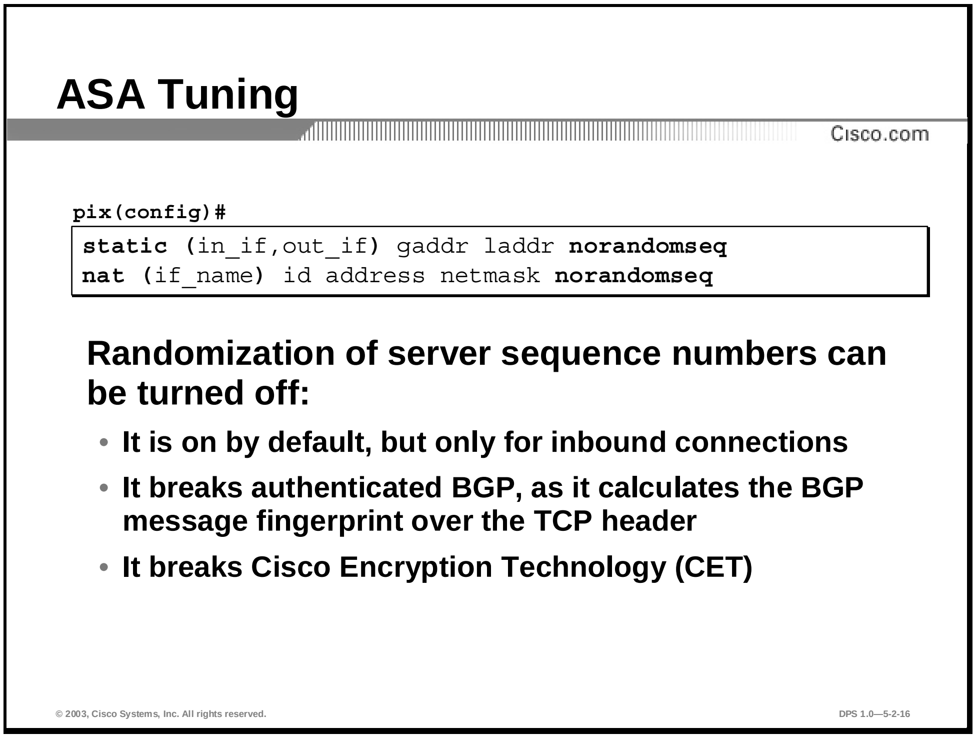
For example:

**established tcp 4000 0**

In this example, if an internal system makes a TCP connection to an external host on port 4000, then the external host could come back in on any port using any protocol:

**established tcp 0 0** (Same as previous releases **established tcp 0** command.)

Use the established command with extreme care and avoid port wildcards or wide ranges to minimize access.



## ASA Tuning

```
pix(config)#
static (in_if,out_if) gaddr laddr norandomseq
nat (if_name) id address netmask norandomseq
```

**Randomization of server sequence numbers can be turned off:**

- **It is on by default, but only for inbound connections**
- **It breaks authenticated BGP, as it calculates the BGP message fingerprint over the TCP header**
- **It breaks Cisco Encryption Technology (CET)**

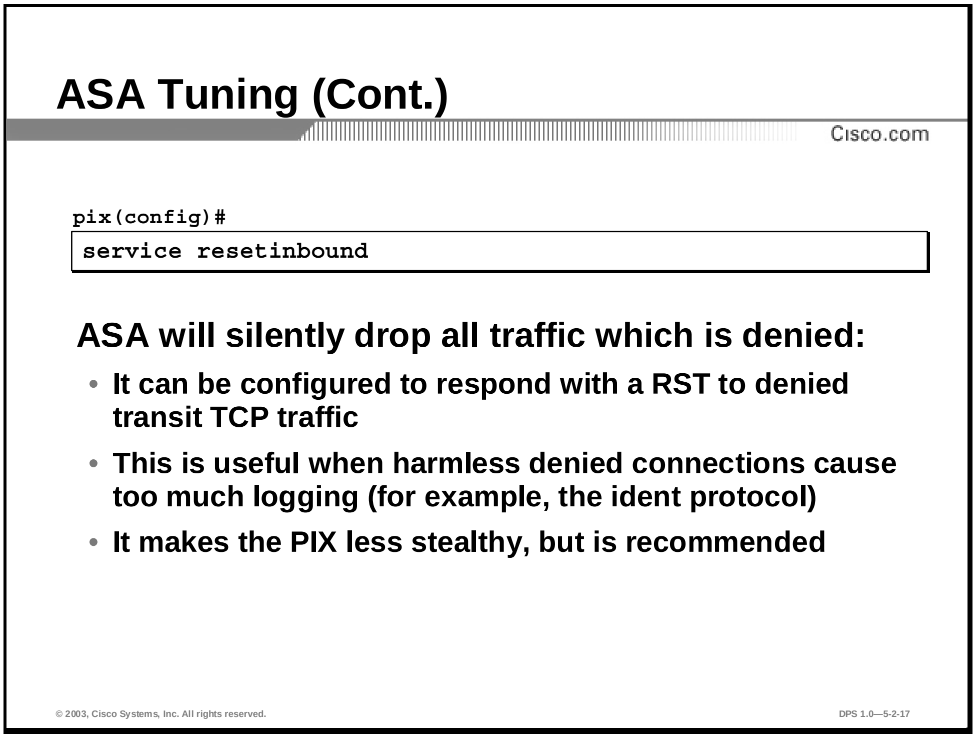
DPS 1.0—5-2-16

# ASA Tuning

For some conversations, it is important to not utilize the randomized TCP sequence numbers that the PIX offers.

CET (no longer a supported technology) is broken due to sequence numbers used as part of the packet authenticity. BGP authenticated sessions use sequence numbers in packet fingerprinting as well.

The feature is turned off in the **nat** or **static** commands for address translation.



## ASA Tuning (Cont.)

```
pix(config)#
service resetinbound
```

**ASA will silently drop all traffic which is denied:**

- **It can be configured to respond with a RST to denied transit TCP traffic**
- **This is useful when harmless denied connections cause too much logging (for example, the ident protocol)**
- **It makes the PIX less stealthy, but is recommended**

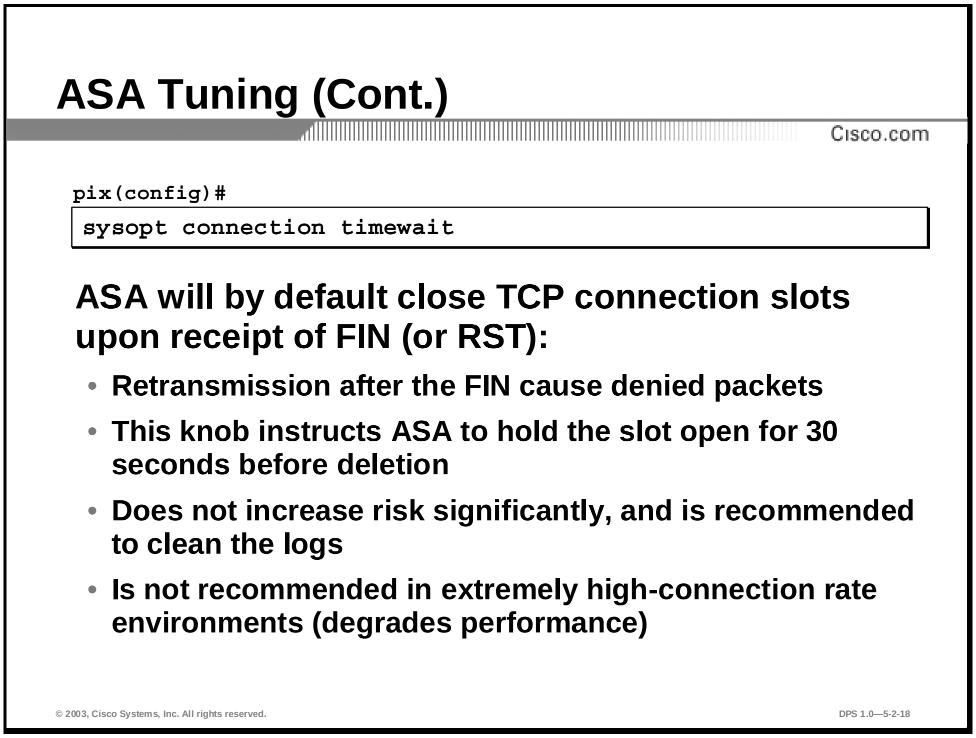
© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—5-2-17

The IDENT protocol causes delays for many common applications running through a PIX firewall. Examples include HTTP, FTP, POP3 and others.

IDENT is a method of verifying a stations name or other information. Typically it is not allowed for an incoming connection, as it is not relevant to business needs. However, a site will wait for the timeout of IDENT (if configured) before proceeding with the original protocol conversation.

The **service resetinbound** command will make the PIX respond with a TCP RST flag to denied incoming TCP traffic. This may speed up hackers' scans of the system, since the RST is sent to other TCP sessions than IDENT. But despite the limitations, it is still recommended practice to use.

For additional information, see: http://www.cisco.com/warp/public/110/2.html



## ASA Tuning (Cont.)

```
pix(config)#
sysopt connection timewait
```

**ASA will by default close TCP connection slots upon receipt of FIN (or RST):**

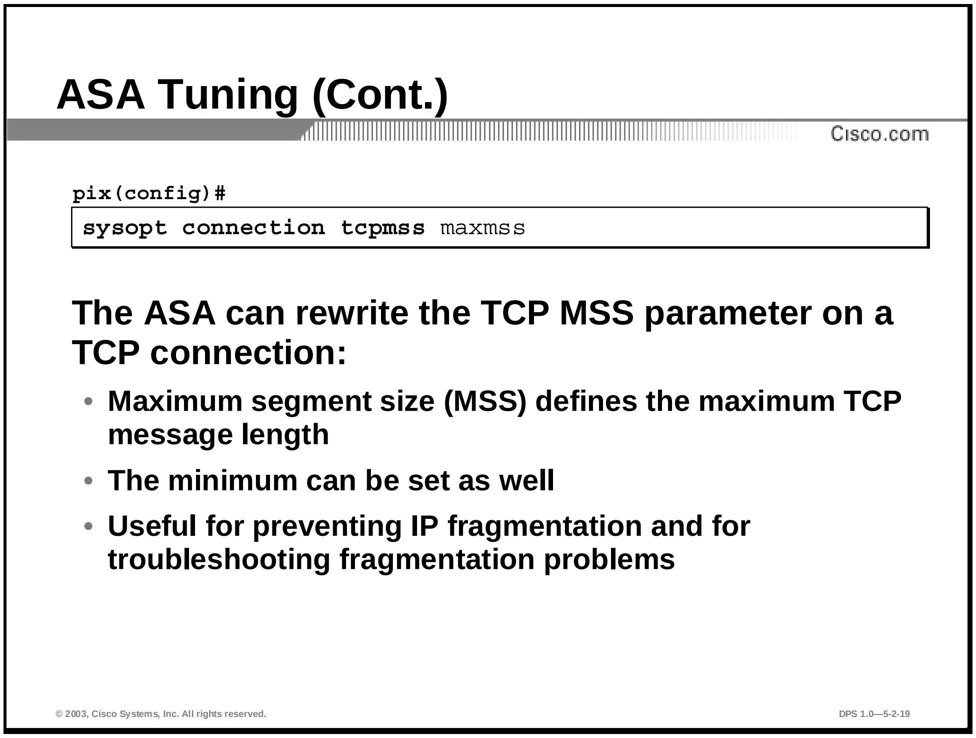
- **Retransmission after the FIN cause denied packets**
- **This knob instructs ASA to hold the slot open for 30 seconds before deletion**
- **Does not increase risk significantly, and is recommended to clean the logs**
- **Is not recommended in extremely high-connection rate environments (degrades performance)**

DPS 1.0—5-2-18

The **sysopt connection timewait** command is necessary for end host applications whose default TCP terminating sequence is a simultaneous close instead of the normal shutdown sequence (see RFC 793). In a simultaneous close, both ends of the transaction initiate the closing sequence, as opposed to the normal sequence where one end closes and the other end acknowledges prior to initiating its own closing sequence.

The default behavior of the PIX Firewall is to track the normal shutdown sequence and release the connection after two FINs and the ACKnowledgment of the last FIN segment. This quick release heuristic enables the PIX Firewall to sustain a high connection rate.

However with a simultaneous close, the quick release forces one side of the connection to linger in the CLOSING state (see RFC 793). Many sockets in the CLOSING state can degrade the performance of an end host. For instance, some WinSock mainframe clients are known to exhibit this behavior and degrade the performance of the mainframe server. Old versions of HP/UX are also susceptible to this behavior. Enabling the **sysopt connection timewait** command enables a quiet time window for the abnormal close down sequence to complete.



## ASA Tuning (Cont.)

Cisco.com

```
pix(config)#
sysopt connection tcpmss maxmss
```

**The ASA can rewrite the TCP MSS parameter on a TCP connection:**

- **Maximum segment size (MSS) defines the maximum TCP message length**
- **The minimum can be set as well**
- **Useful for preventing IP fragmentation and for troubleshooting fragmentation problems**

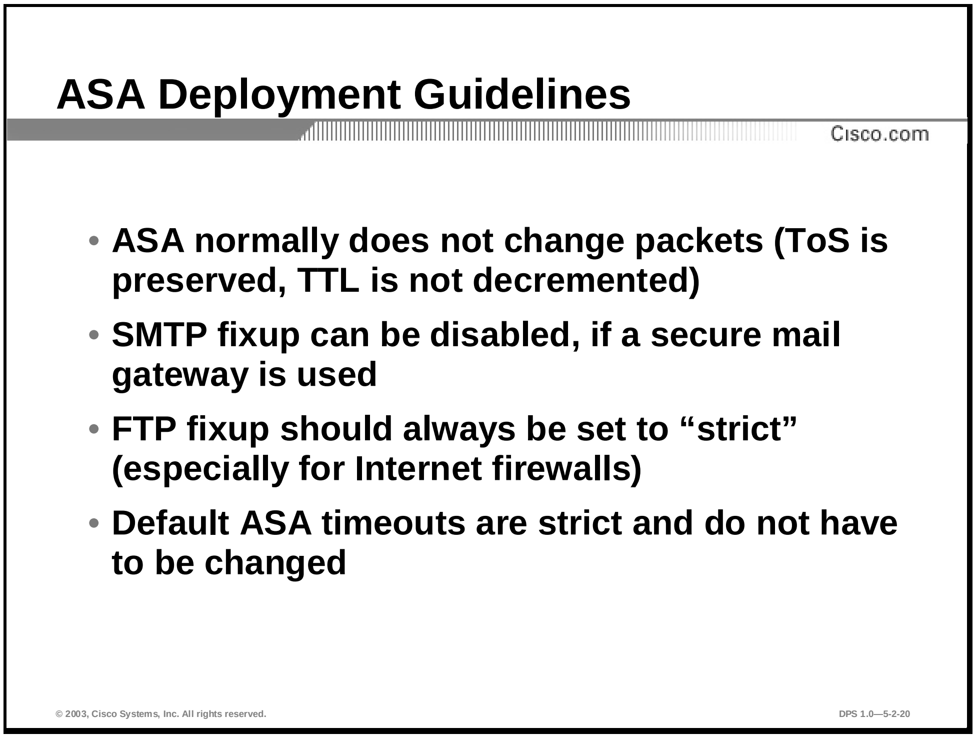
DPS 1.0—5-2-19

This setting is used to control the maximum TCP segment (the PDU of layer 4) size handled by the PIX.

The *bytes* value can be a minimum of 28 and any maximum number. You can disable this feature by setting *bytes* to zero. By default, the PIX Firewall sets 1380 bytes as the **sysopt connection tcpmss** even though this command does not appear in the default configuration. The calculation for setting the TCP maximum segment size to 1380 bytes is as follows.

> 1380 data + 20 TCP + 20 IP + 24 AH + 24 ESP_CIPHER + 12 ESP_AUTH + 20 IP = 1500 bytes

1500 bytes is the MTU for Ethernet connections. We recommend that the default value of 1380 bytes be used for Ethernet and mixed Ethernet and Token Ring environments. If the PIX Firewall has all Token Ring interfaces, you can set *bytes* to 4056. However, if even one link along the path through the network is not a Token Ring, setting *bytes* to such a high value may cause poor throughput.



## ASA Deployment Guidelines

- ASA normally does not change packets (ToS is preserved, TTL is not decremented)
- SMTP fixup can be disabled, if a secure mail gateway is used
- FTP fixup should always be set to “strict” (especially for Internet firewalls)
- Default ASA timeouts are strict and do not have to be changed

DPS 1.0—5-2-20

# Guidelines

The default configuration of the PIX is a good start to network security. Most settings do not need to be changed unless there is a good reason to do so.

In the **fixup protocol ftp 21 strict** command, the strict key prevents web browsers from sending embedded commands in FTP requests. Each FTP command must be acknowledged before a new command is allowed. Connections sending embedded commands are dropped. This setting is highly recommended if the FTP protocol is allowed through the firewall.

By default, the PIX is not a layer 3 connecting device in terms of TTL decrementing. It is one of those “I’m not here” devices. It will also preserve the ToS field and hence the QoS policy indicated through packet marking.

# Practice

Q1) Which command can describe a new application to the PIX Firewall ASA engine?

A) sysopt connection

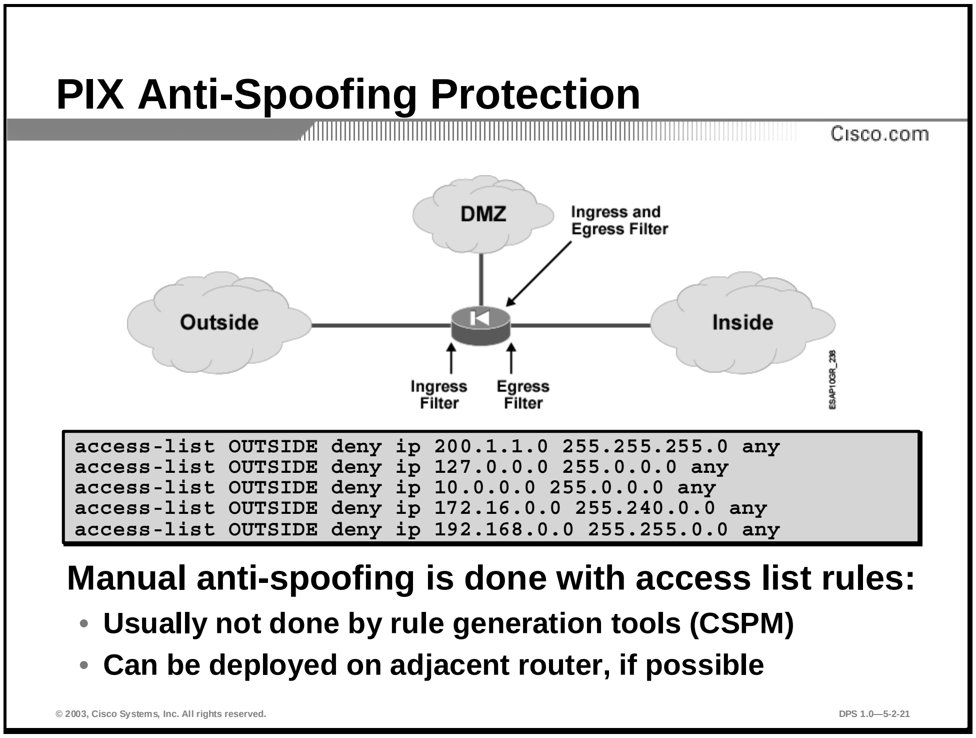
B) established

C) access-list

D) conduit

E) object-group

# PIX Advanced Access Control Features



## PIX Anti-Spoofing Protection

Cisco.com

```
access-list OUTSIDE deny ip 200.1.1.0 255.255.255.0 any
access-list OUTSIDE deny ip 127.0.0.0 255.0.0.0 any
access-list OUTSIDE deny ip 10.0.0.0 255.0.0.0 any
access-list OUTSIDE deny ip 172.16.0.0 255.240.0.0 any
access-list OUTSIDE deny ip 192.168.0.0 255.255.0.0 any
```

**Manual anti-spoofing is done with access list rules:**

- **Usually not done by rule generation tools (CSPM)**
- **Can be deployed on adjacent router, if possible**

DPS 1.0—5-2-21

## Objective

This section will enable the learner to identify and configure advanced access control features of the Cisco Secure PIX Firewall product and choose them in a firewall design.

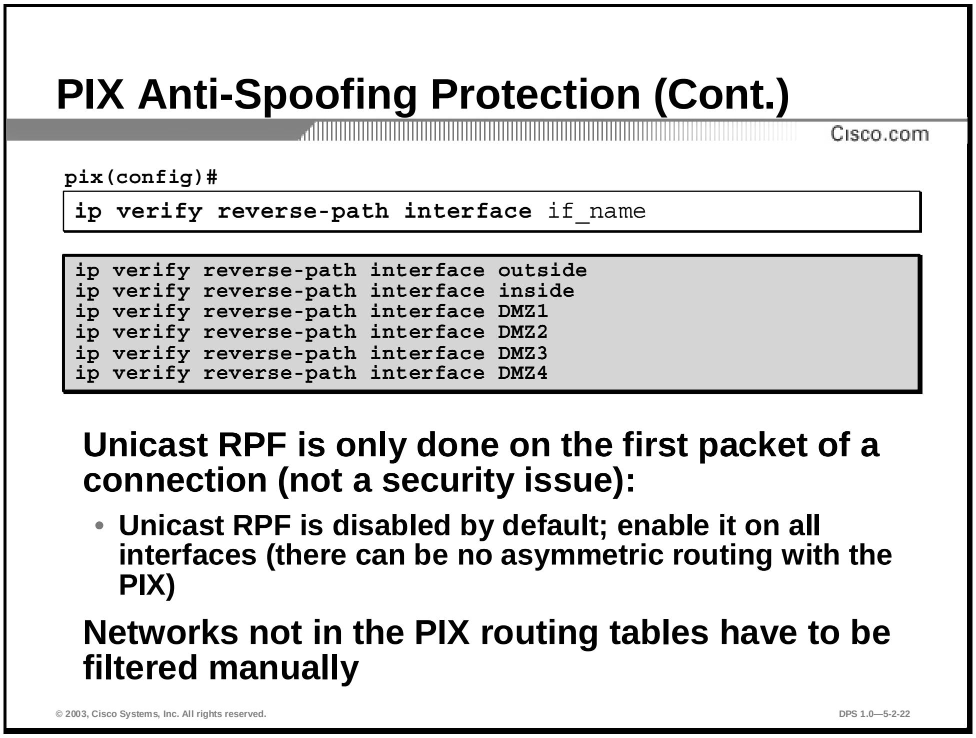
## Introduction

The PIX has several advanced features, which are enabled in firewall systems, which require high levels of security. Some PIX Firewall advanced functionality, though, is only available through proper configuration of basic mechanisms. This section introduces both: the built-in advanced access control mechanisms, and the configuration of existing mechanisms to support advanced requirements.

## Anti-Spoofing using Access Lists

Access-lists are part of the anti-spoofing protection that can be set up. The basis is that no packets with source addresses used on the internal network, or from private IP addresses should come into the network from an outside interface.

While part of an obvious security step, this part is usually accomplished at the external perimeter routers' outside interfaces to prevent any further processing of packets that are clearly malicious.



## PIX Anti-Spoofing Protection (Cont.)

```
pix(config)#
ip verify reverse-path interface if_name
```

```
ip verify reverse-path interface outside
ip verify reverse-path interface inside
ip verify reverse-path interface DMZ1
ip verify reverse-path interface DMZ2
ip verify reverse-path interface DMZ3
ip verify reverse-path interface DMZ4
```

**Unicast RPF is only done on the first packet of a connection (not a security issue):**

- **Unicast RPF is disabled by default; enable it on all interfaces (there can be no asymmetric routing with the PIX)**

**Networks not in the PIX routing tables have to be filtered manually**

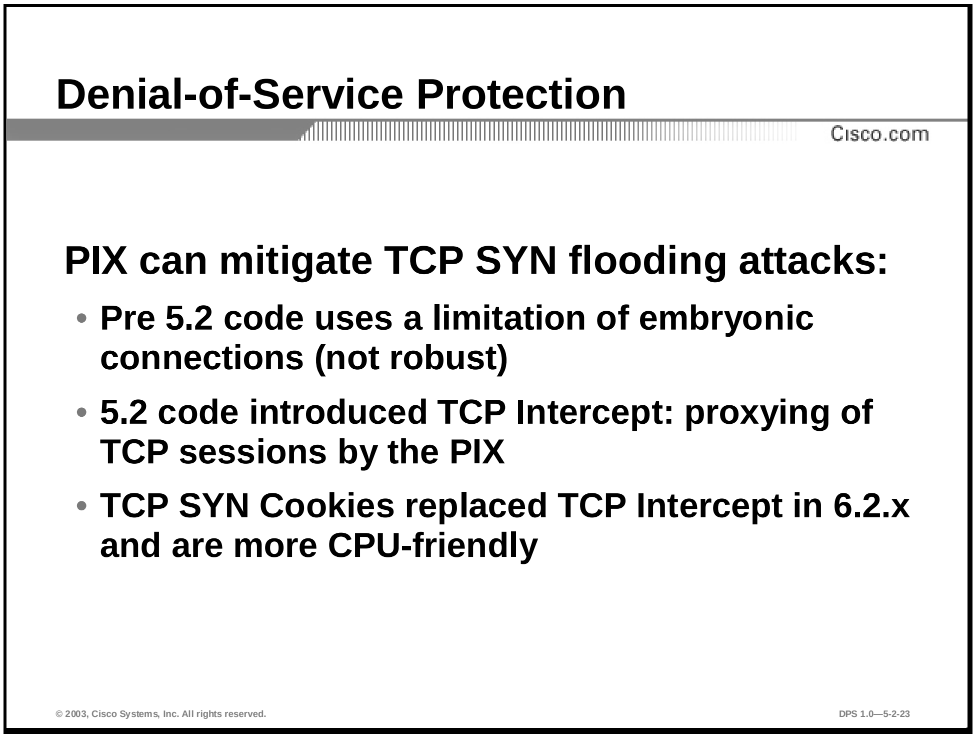
## PIX Unicast RPF

For additional anti-spoofing measures, the PIX can verify the return path of incoming packets to make sure they were received from the correct interface.

The **ip verify reverse-path** command provides both ingress and egress filtering. Ingress filtering checks inbound packets for IP source address integrity, and is limited to addresses for networks in the enforcing entity’s local routing table. If the incoming packet does not have a source address represented by a route, then it is impossible to know whether the packet has arrived on the best possible path back to its origin. This is often the case when routing entities cannot maintain routes for every network.

Egress filtering verifies that packets destined for hosts outside the managed domain have IP source addresses verifiable by routes in the enforcing entity’s local routing table. If an exiting packet does not arrive on the best return path back to the originator, then the packet is dropped and the activity is logged. Egress filtering prevents internal users from launching attacks using IP source addresses outside of the local domain because most attacks use IP spoofing to hide the identity of the attacking host. Egress filtering makes the task of tracing the origin of an attack much easier. When employed, egress filtering enforces what IP source addresses are obtained from a valid pool of network addresses. Addresses are kept local to the enforcing entity and are therefore easily traceable.

**Note** In an Internet environment, this command will depend on the existence of a 0.0.0.0 0.0.0.0 route to work for all addresses. If there is no default route, and a packet comes in from an unmatched IP address, it will be dropped.



## Denial-of-Service Protection

**PIX can mitigate TCP SYN flooding attacks:**

- **Pre 5.2 code uses a limitation of embryonic connections (not robust)**
- **5.2 code introduced TCP Intercept: proxying of TCP sessions by the PIX**
- **TCP SYN Cookies replaced TCP Intercept in 6.2.x and are more CPU-friendly**

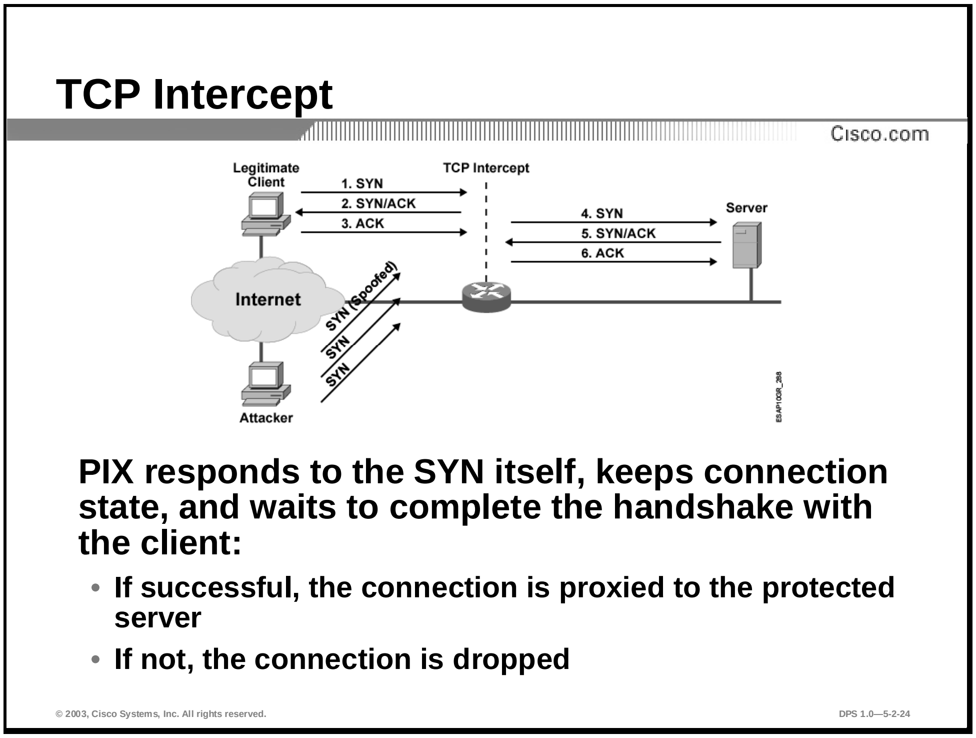
DPS 1.0—5-2-23

## PIX Denial-of-Service Prevention

The protections against various DoS attacks have increased through newer versions of the PIX OS.

In older revisions of the OS, only the total number of existing embryonic connections could be monitored. Any additional were simply dropped.

Beginning version 5.2, TCP Intercept provided for the proxy resets of sessions without any knowledge or inference of the destination station. PIX/OS 6.2 introduced SYN Cookies, which are another proxy verification tool the PIX uses to validate new sessions.



## TCP Intercept

**PIX responds to the SYN itself, keeps connection state, and waits to complete the handshake with the client:**

- **If successful, the connection is proxied to the protected server**
- **If not, the connection is dropped**

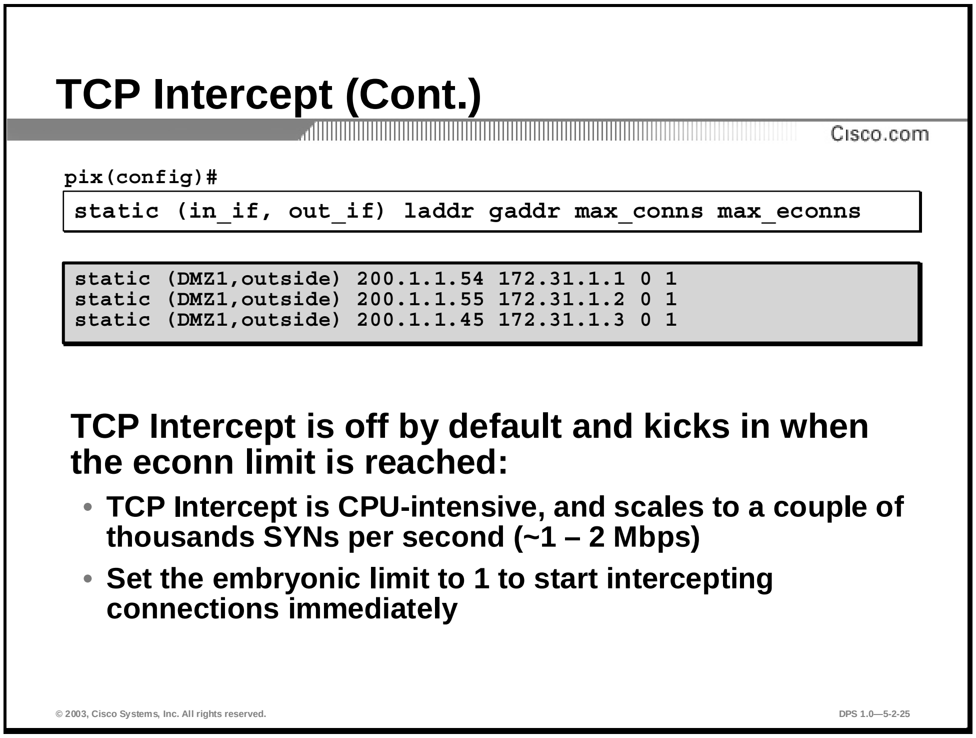
DPS 1.0—5-2-24

## TCP Intercept

Cisco PIX (versions 5.2 through 6.1) has a TCP Intercept capability that is designed to combat SYN Flooding. TCP Intercept checks for incoming TCP connection requests and will proxy-answer on behalf of the destination server to ensure that the request is valid before then connecting to the server. Once TCP Intercept has established a genuine connection with the client and the server, it then merges these two connections into a single source-destination session. It offers a zero window to the client to prevent it from sending data until the server sends a window offer back. In the case of bogus requests, its use of aggressive time-outs on half-open connections and support of threshold levels for both the number of outstanding and incoming rate of TCP connection requests, protect servers while still allowing valid requests through.

Additional information on the TCP Intercept feature, introduced in PIX OS 5.2 can be found at:

http://www.cisco.com/warp/public/cc/pd/fw/sqfw500/prodlit/pix22_ds.htm



## TCP Intercept (Cont.)

```
pix(config)#
static (in_if, out_if) laddr gaddr max_conns max_econns
```

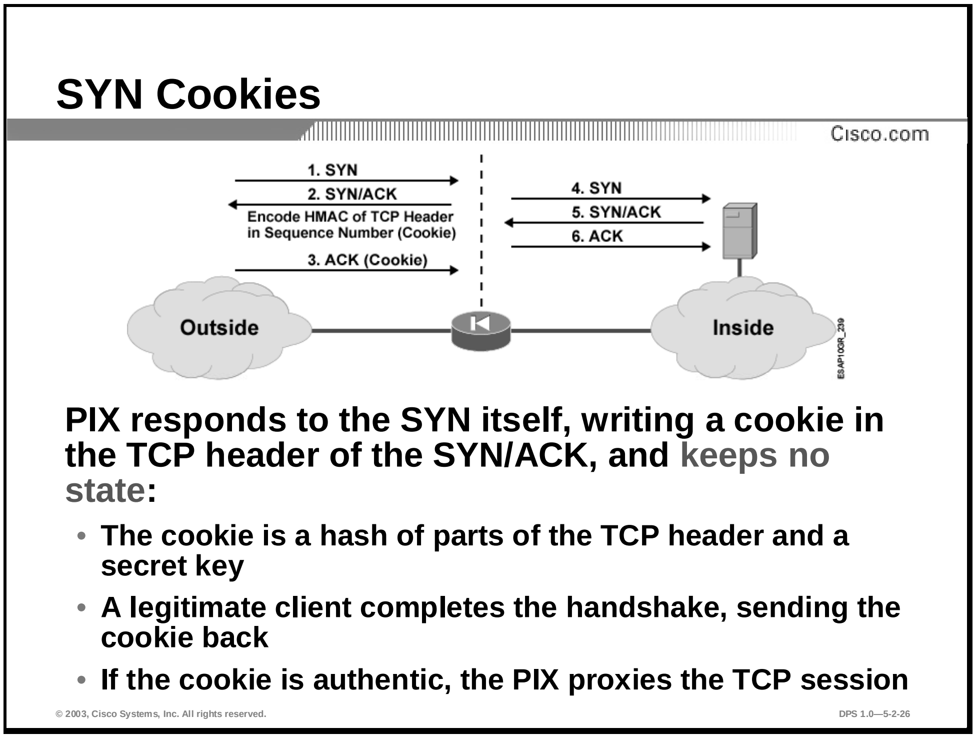
```
static (DMZ1,outside) 200.1.1.54 172.31.1.1 0 1
static (DMZ1,outside) 200.1.1.55 172.31.1.2 0 1
static (DMZ1,outside) 200.1.1.45 172.31.1.3 0 1
```

**TCP Intercept is off by default and kicks in when the econn limit is reached:**

- **TCP Intercept is CPU-intensive, and scales to a couple of thousands SYNs per second (~1 – 2 Mbps)**
- **Set the embryonic limit to 1 to start intercepting connections immediately**

TCP Intercept will kick in when the “econn” (embryonic connection limit) is reached. The embryonic connection limit is the last variable (“1” in example) in the **static** or a **nat** command.

Setting the limit to 1 would cause the PIX to perform a LOT of work as these are CPU intensive. Choose a reasonable number based on typical traffic flow.



SYN Cookies

PIX responds to the SYN itself, writing a cookie in the TCP header of the SYN/ACK, and keeps no state:

- The cookie is a hash of parts of the TCP header and a secret key
- A legitimate client completes the handshake, sending the cookie back
- If the cookie is authentic, the PIX proxies the TCP session

DPS 1.0—5-2-26

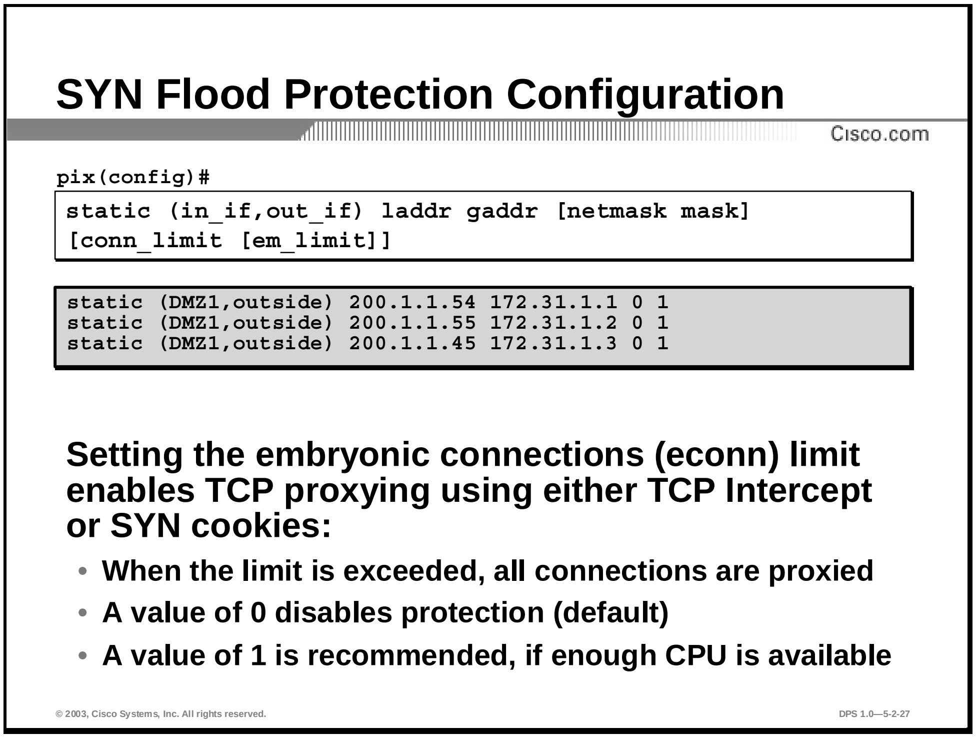
# SYN Cookies

SYN Cookies represent a less-CPU-intensive method of verifying the incoming TCP sessions for validity.

SYN cookies are an implementation of TCP which can respond to a TCP SYN request with a “cookie”. In the original TCP implementation, when a server received a SYN packet, it responded with a SYN-ACK, and entered the “half-open” state to wait for the ACK that will complete the handshake. Too many “half-open” connections can result in full buffers.

In the SYN cookies implementation of TCP, when the server receives a SYN packet, it responds with a SYN-ACK packet where the ACK sequence number is calculated from source address, source port, source sequence, destination address, destination port and a secret seed. Then the server releases all state. If an ACK comes from the client, the server can recalculate it to determine if it’s a response to a former SYN-ACK. If it is, the server can directly enter the TCP_ESTABLISHED state and open the connection. In this way, the server avoids managing a batch of potentially useless half-open connections.

The PIX can respond using SYN cookie instead of a protected server. This feature replaces TCP Intercept, as it is more scalable in terms of performance.



## SYN Flood Protection Configuration

```
pix(config)#
static (in_if,out_if) laddr gaddr [netmask mask]
[conn_limit [em_limit]]
```

```
static (DMZ1,outside) 200.1.1.54 172.31.1.1 0 1
static (DMZ1,outside) 200.1.1.55 172.31.1.2 0 1
static (DMZ1,outside) 200.1.1.45 172.31.1.3 0 1
```

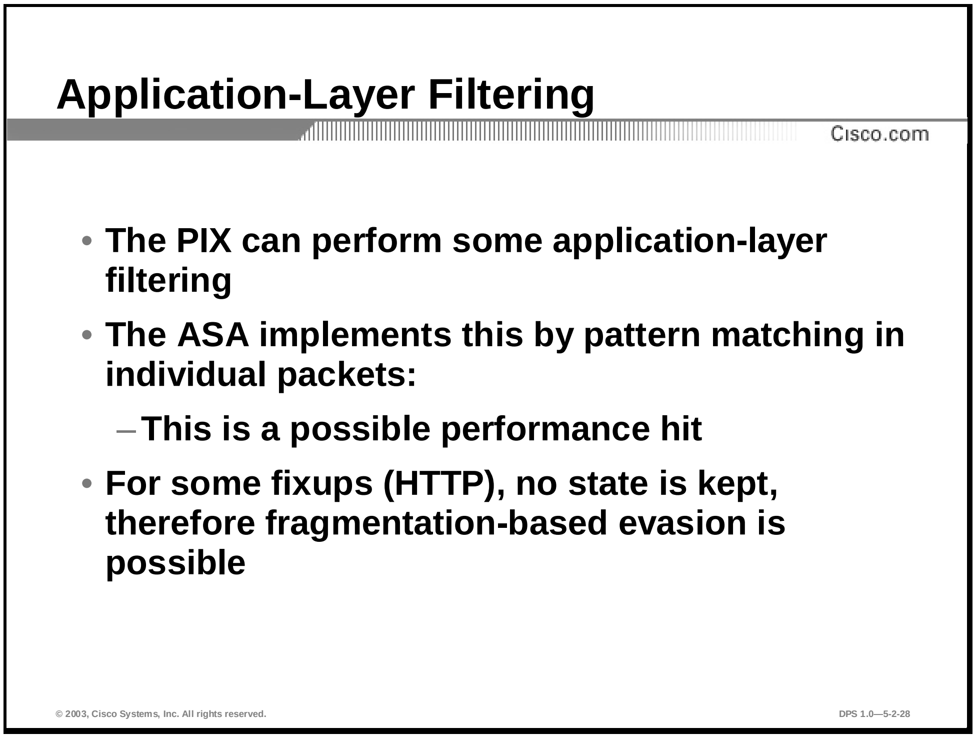
**Setting the embryonic connections (econn) limit enables TCP proxying using either TCP Intercept or SYN cookies:**

- **When the limit is exceeded, all connections are proxied**
- **A value of 0 disables protection (default)**
- **A value of 1 is recommended, if enough CPU is available**

DPS 1.0—5-2-27

Setting connection limits to any translation should only be done after application/business analysis to determine the actual needs. Setting the value too high (0 is default) may make the destination stations more vulnerable. Setting the value too low (1 as in example) may increase the CPU load of the PIX too high.

1 is the recommended limit for embryonic connections in typical networks. The amount of traffic through a network or to destination stations typically occurring may necessitate the changing of these values.



## Application-Layer Filtering

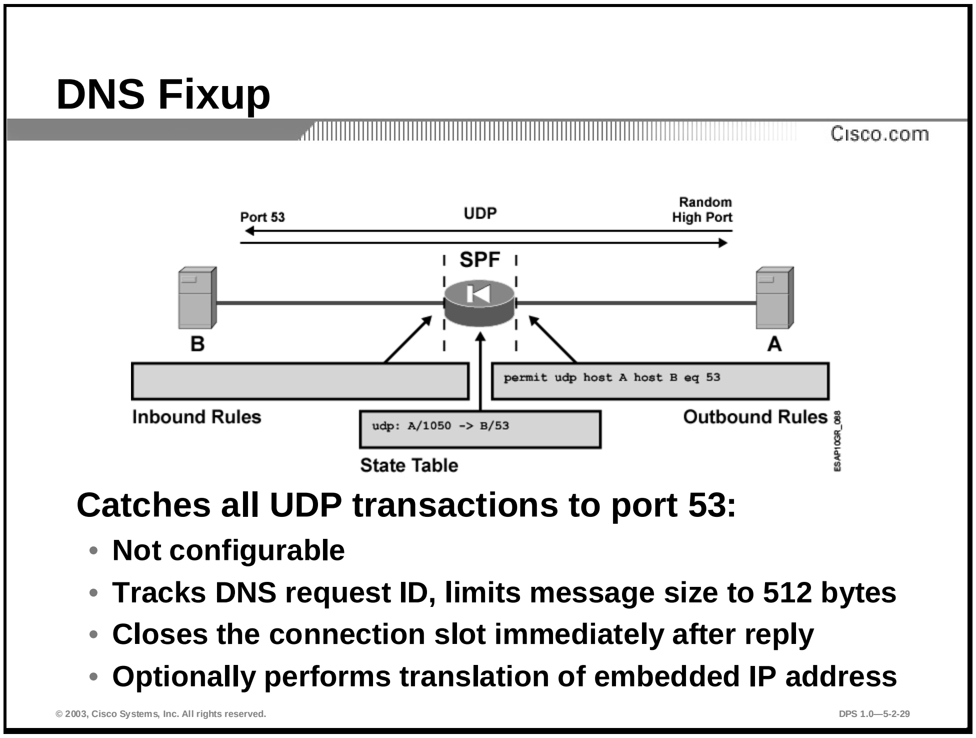
- The PIX can perform some application-layer filtering
- The ASA implements this by pattern matching in individual packets:
  - This is a possible performance hit
- For some fixups (HTTP), no state is kept, therefore fragmentation-based evasion is possible

DPS 1.0—5-2-28

## Application-Layer Filtering

Additional filtering tools can be used for HTTP connections, although additional processing of packets may yield performance hits. The PIX Firewall has some application-layer insight into packets by examining application layer payloads to filter a protocol, or manipulate with application-layer data. All implemented mechanisms involve payload scanning, and can have an impact on performance, if bulk traffic is subject to application inspection (such as URL or ActiveX filtering scenarios).

Some of the content filtering algorithms (ActiveX, Java filtering) cannot interpret the application protocol, if it is fragmented in multiple IP packets, therefore, some evasion attacks are possible.



## DNS Fixup

Cisco.com

**Catches all UDP transactions to port 53:**

- **Not configurable**
- **Tracks DNS request ID, limits message size to 512 bytes**
- **Closes the connection slot immediately after reply**
- **Optionally performs translation of embedded IP address**

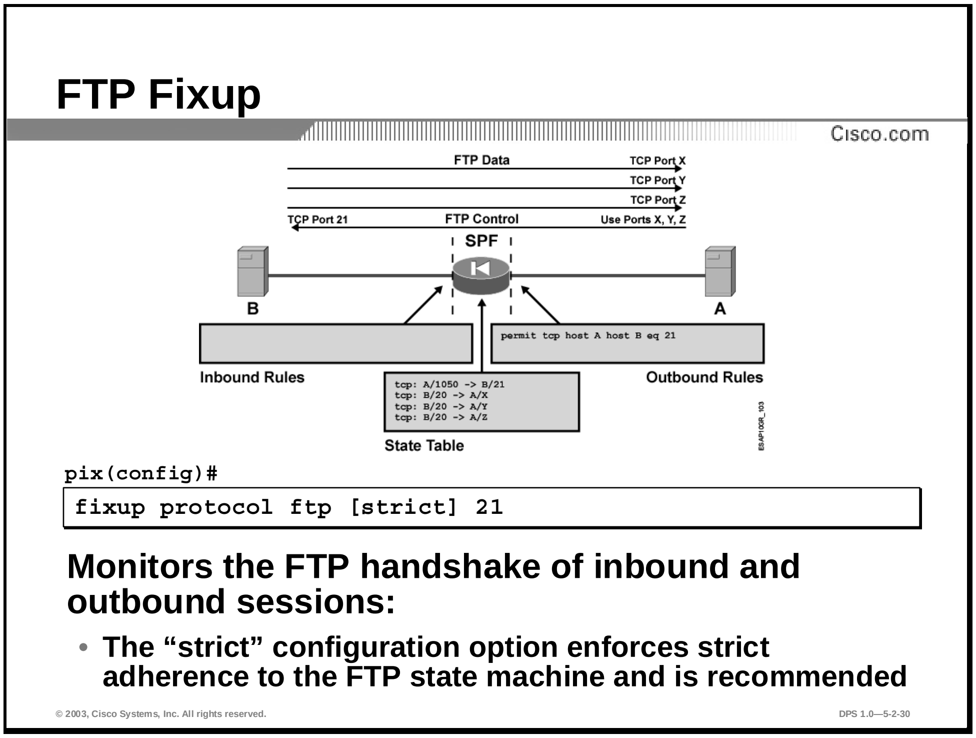
DPS 1.0—5-2-29

# DNS Fixup

DNS is one of the protocols that embeds source/destination addresses within PDU’s higher than layer 3. Translation takes place for UDP DNS traffic only.

The PIX also knows that DNS queries are a one-request, one-answer conversation, so the connection slot is released immediately after an answer is received.

With the **alias** command or **dnat** configuration, the PIX may translate DNS answers of the destination.



## FTP Fixup

Cisco.com

```
permit tcp host A host B eq 21
```

```
tcp: A/1050 -> B/21
tcp: B/20 -> A/X
tcp: B/20 -> A/Y
tcp: B/20 -> A/Z
```

```
pix(config)#
fixup protocol ftp [strict] 21
```

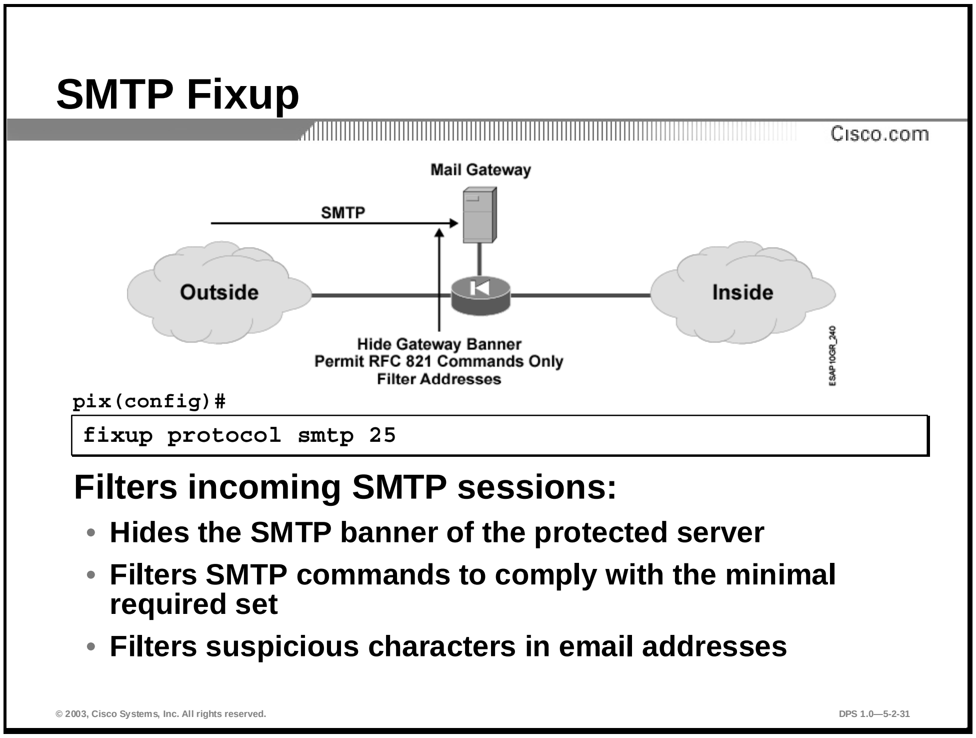
**Monitors the FTP handshake of inbound and outbound sessions:**

- **The “strict” configuration option enforces strict adherence to the FTP state machine and is recommended**

DPS 1.0—5-2-30

# FTP Fixup

The **strict** command implies that no nested commands can be used. Each FTP command must be parsed and validated by the ftp server separately.



## SMTP Fixup

Hide Gateway Banner
Permit RFC 821 Commands Only
Filter Addresses

```
pix(config)#
fixup protocol smtp 25
```

**Filters incoming SMTP sessions:**

- **Hides the SMTP banner of the protected server**
- **Filters SMTP commands to comply with the minimal required set**
- **Filters suspicious characters in email addresses**

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—5-2-31

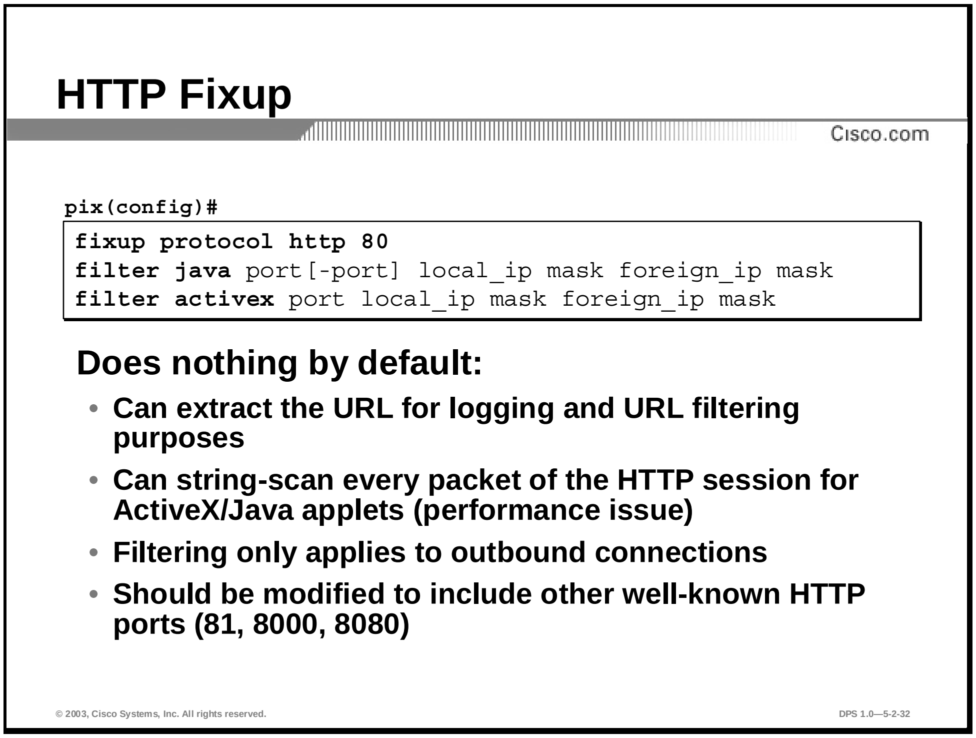
# SMTP Fixup

The SMTP Fixup allows only certain SMTP commands to be utilized, thereby protecting SMTP servers from many types of attacks.

For this reason, some devices using ESMTP and needing it to function, may not run correctly.

When configured, Mailguard allows only the seven SMTP minimum-required commands as described in Section 4.5.1 of RFC 821. These seven minimum-required commands are: HELO, MAIL, RCPT, DATA, RSET, NOOP, and QUIT. Other commands, such as KILL, WIZ, and so forth, are intercepted by the PIX and they are never sent to the mail server on the inside of your network. The PIX responds with an “OK” to even denied commands, so attackers would not know that their attempts are being thwarted.

When the mailserver receives an invalid or unacceptable command, it generates a “500 Command unrecognized” message. In Part 2, when Mailguard is configured, the PIX then *intercepts* the entered command, and passes only valid commands (that is, one of the seven minimum-required SMTP commands) on to the mail server behind the firewall. PIX then returns an “OK” to the user regardless of whether the command entered was passed on or denied. In this way, PIX confuses anyone attempting an attack on the mail system.



## HTTP Fixup

```
pix(config)#
fixup protocol http 80
filter java port[-port] local_ip mask foreign_ip mask
filter activex port local_ip mask foreign_ip mask
```

**Does nothing by default:**

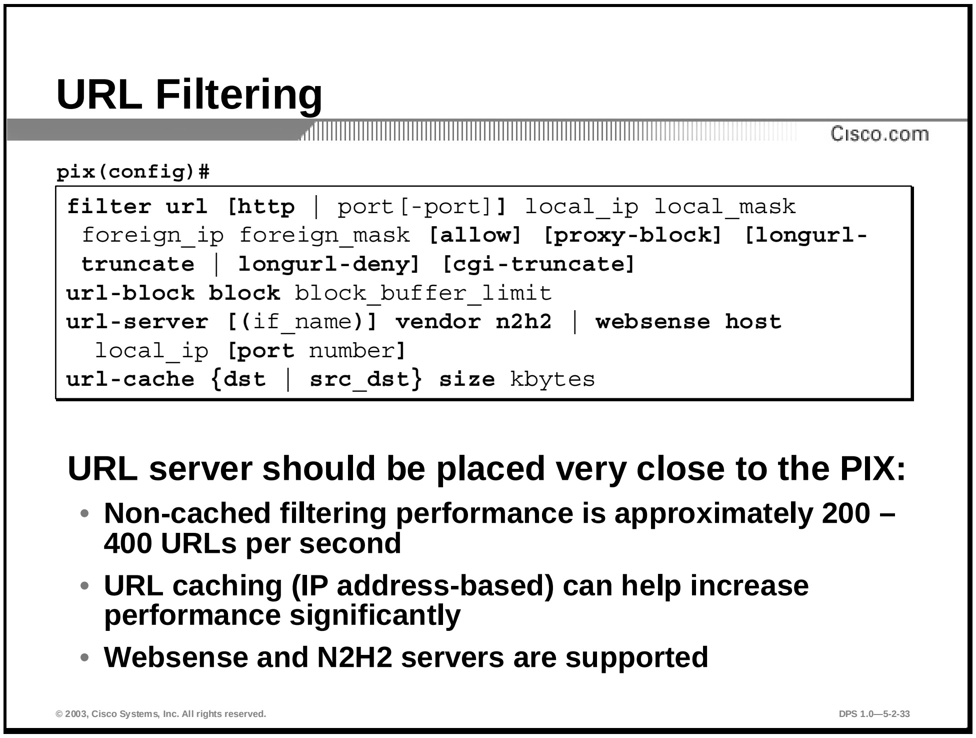
- **Can extract the URL for logging and URL filtering purposes**
- **Can string-scan every packet of the HTTP session for ActiveX/Java applets (performance issue)**
- **Filtering only applies to outbound connections**
- **Should be modified to include other well-known HTTP ports (81, 8000, 8080)**

DPS 1.0—5-2-32

# HTTP Fixup

The basic features of the HTTP Fixup engine are application-layer accounting (URL logging), and application layer filtering (Java, ActiveX, URL filtering). Separate **fixup protocol** commands must be entered for each port desired to be protected, usually, for some other well-known HTTP ports (81, 8000, 8080, etc.).

Note that additional processing of information beyond the TCP stream may induce a performance hit, and might be offloaded to dedicated devices, such as content engines.



## URL Filtering

```
pix(config)#
filter url [http | port[-port]] local_ip local_mask
 foreign_ip foreign_mask [allow] [proxy-block] [longurl-
 truncate | longurl-deny] [cgi-truncate]
url-block block block_buffer_limit
url-server [(if_name)] vendor n2h2 | websense host
  local_ip [port number]
url-cache {dst | src_dst} size kbytes
```

**URL server should be placed very close to the PIX:**

- **Non-cached filtering performance is approximately 200 – 400 URLs per second**
- **URL caching (IP address-based) can help increase performance significantly**
- **Websense and N2H2 servers are supported**

DPS 1.0—5-2-33

# URL Filtering

Cache size should be a few hundred kilobytes. The caching policy should be destination based if the same policy applies to all inside hosts. Otherwise, source-destination pair based caching should be used.

Within the main filtering command:

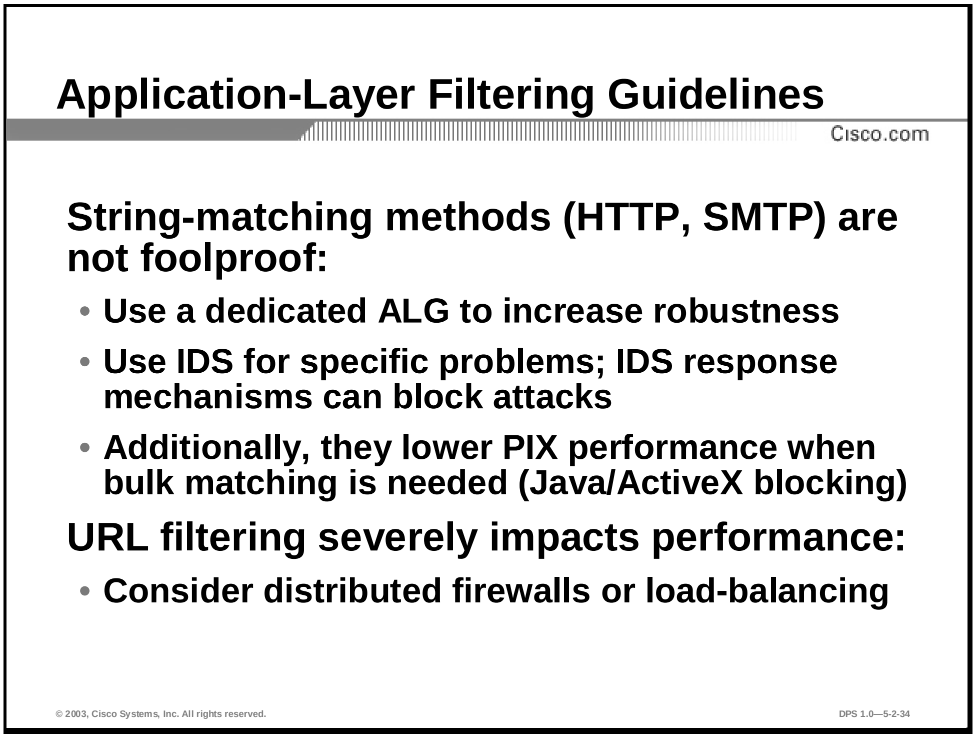
- **http** is a keyword specifying port 80. The port may otherwise be explicitly listed, and must be listed separately if not in sequential order.
- **Local IP** and **mask** information represent the inside network stations
- **Foreign IP** and **mask** information can represent outside web servers (or 0.0.0.0 0.0.0.0 for all)
- The **allow** option is used to determine handling is verification server is offline. By default unverifiable requests will be dropped. With the **allow** parameter, they will be allowed.
- **Proxy-block** can prevent users from connecting to an HTTP Proxy Server.
- The **longurl** options specify actions to a URL beyond the specified size limitation.
- The **cgi-truncate** option will send a CGI script as a URL.

- A **filter url except** command with the same options can be used to create exceptions to the rule of filtered URL’s. Perhaps special users/stations are exempt, or particular applications or ideas such as Anti-virus updates or Automatic OS updates my be exempted.

- The **url-block** commands require a WebSense server to be used for specific URL validation over a particular size limitation. This can log information and possibly reduce attacks from within the network. Default length is 1159 bytes.

- The **url-server** command is used to set up communications with the URL server, either N2H2 or WebSense.

- The **url-cache** command can set aside memory to cache responses from the url-server.



## Application-Layer Filtering Guidelines

Cisco.com

**String-matching methods (HTTP, SMTP) are not foolproof:**

- **Use a dedicated ALG to increase robustness**
- **Use IDS for specific problems; IDS response mechanisms can block attacks**
- **Additionally, they lower PIX performance when bulk matching is needed (Java/ActiveX blocking)**

**URL filtering severely impacts performance:**

- **Consider distributed firewalls or load-balancing**

DPS 1.0—5-2-34

## Guidelines

The algorithms on the PIX for url filtering are process intensive and not foolproof, as they work on a per-packet basis. A dedicated application-layer gateway might be considered to increase the robustness of content filtering inside HTTP sessions.

For additional protections and analysis, it is recommended to have other dedicated resourced and algorithms available for analysis. The CiscoSecure IDS sensors will assist with this type of information.

## Practice

Q1) What is the main limitation of HTTP URL filtering on the PIX Firewall?

A) not all URLs can be filtered

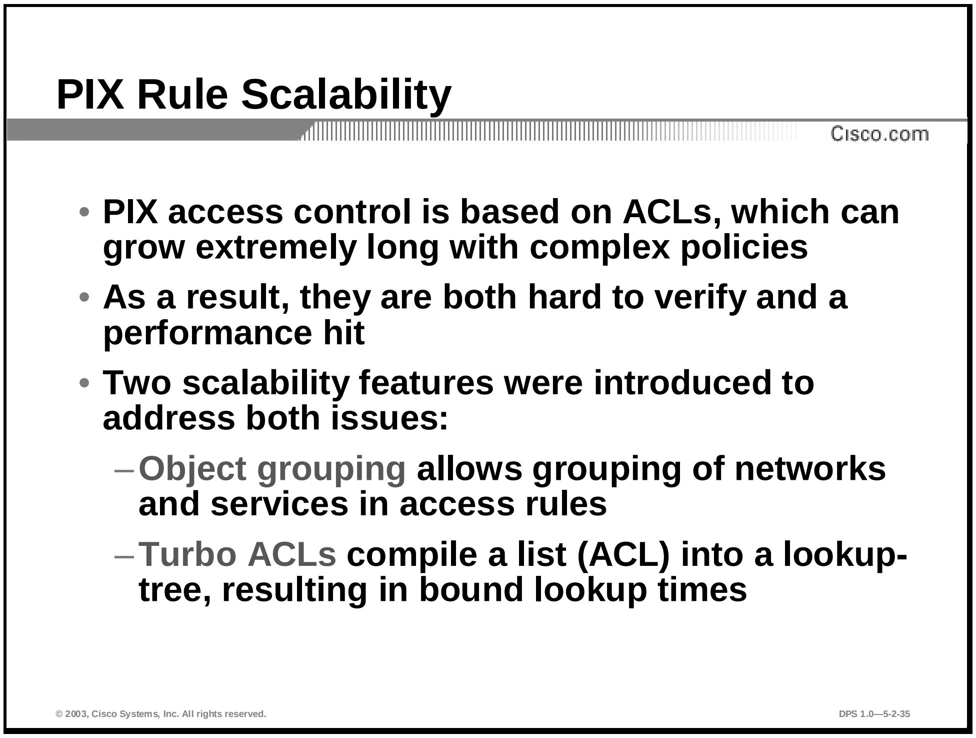
B) it has severe performance impact

C) the URL database on the PIX takes a lot of memory

D) it is not reliable with fragmented traffic

E) it does not filter based on URL categories

# PIX Rule Scalability Features



## PIX Rule Scalability

Cisco.com

- PIX access control is based on ACLs, which can grow extremely long with complex policies
- As a result, they are both hard to verify and a performance hit
- Two scalability features were introduced to address both issues:
  - Object grouping allows grouping of networks and services in access rules
  - Turbo ACLs compile a list (ACL) into a lookup-tree, resulting in bound lookup times

DPS 1.0—5-2-35

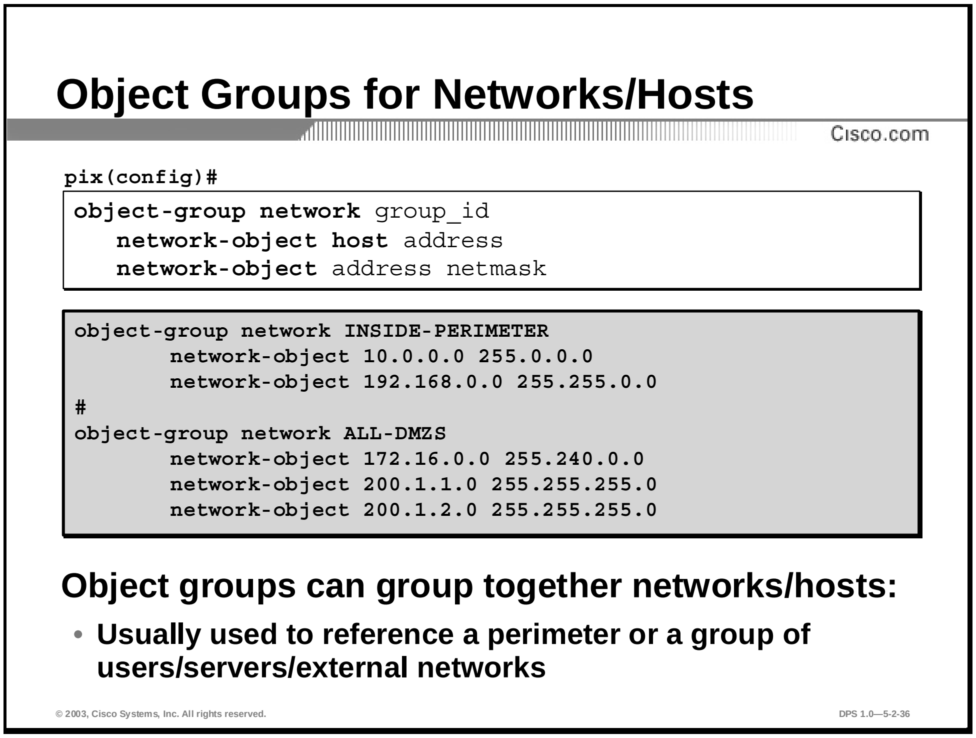
## Objectives

This section will enable the learner to identify and configure advanced access control features of the Cisco Secure PIX Firewall product and choose them in a firewall design.

## Introduction

In order to reduce the command sets necessary for different interfaces, and prevent duplication of access-list entries, grouping commands have been introduced.

In addition, to speed up processing of ACL’s, a feature called “Turbo ACLs” has been implemented in PIX OS 6.2.



## Object Groups for Networks/Hosts

```
pix(config)#
```

```
object-group network group_id
   network-object host address
   network-object address netmask
```

```
object-group network INSIDE-PERIMETER
       network-object 10.0.0.0 255.0.0.0
       network-object 192.168.0.0 255.255.0.0
#
object-group network ALL-DMZS
       network-object 172.16.0.0 255.240.0.0
       network-object 200.1.1.0 255.255.255.0
       network-object 200.1.2.0 255.255.255.0
```

**Object groups can group together networks/hosts:**

- **Usually used to reference a perimeter or a group of users/servers/external networks**

DPS 1.0—5-2-36

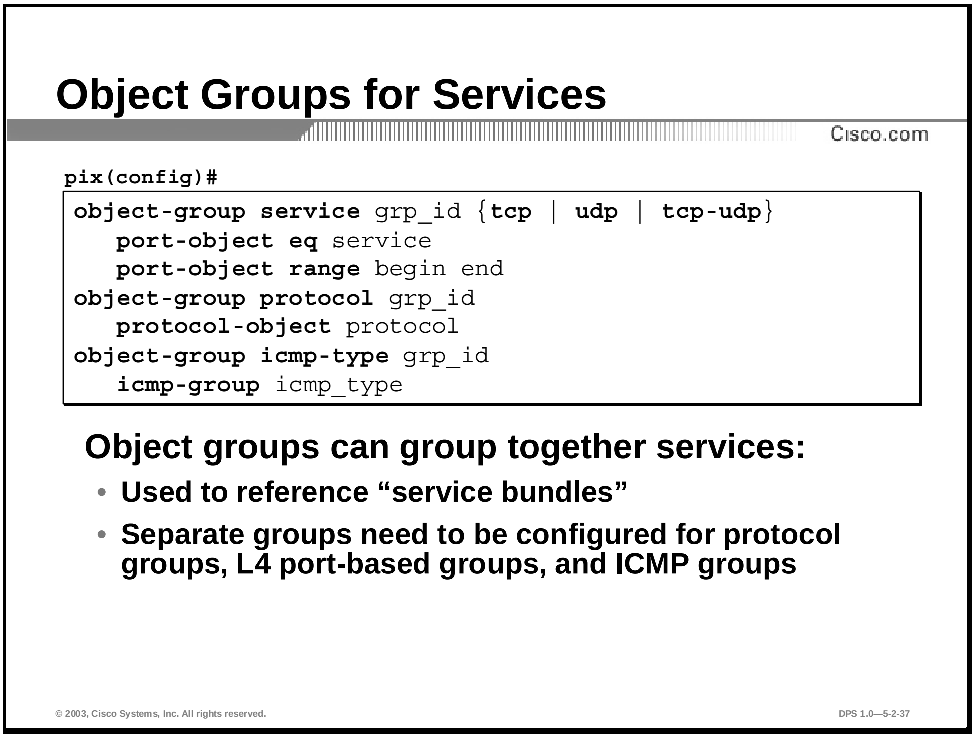
# Object Groups

To group reference points together for access-list use, and object group can be created. This prevents multiple entries of the same permit or deny statements over and over.

Object groups can include protocol (IP protocol field), network-object (IP/mask), service (port info) or ICMP-Type (ICMP type codes).

For additional information, see:

http://www.cisco.com/univercd/cc/td/doc/product/iaabu/pix/pix_62/cmdref/mr.htm#xtocid8



## Object Groups for Services

```
pix(config)#
object-group service grp_id {tcp | udp | tcp-udp}
   port-object eq service
   port-object range begin end
object-group protocol grp_id
   protocol-object protocol
object-group icmp-type grp_id
   icmp-group icmp_type
```

**Object groups can group together services:**

- **Used to reference “service bundles”**
- **Separate groups need to be configured for protocol groups, L4 port-based groups, and ICMP groups**

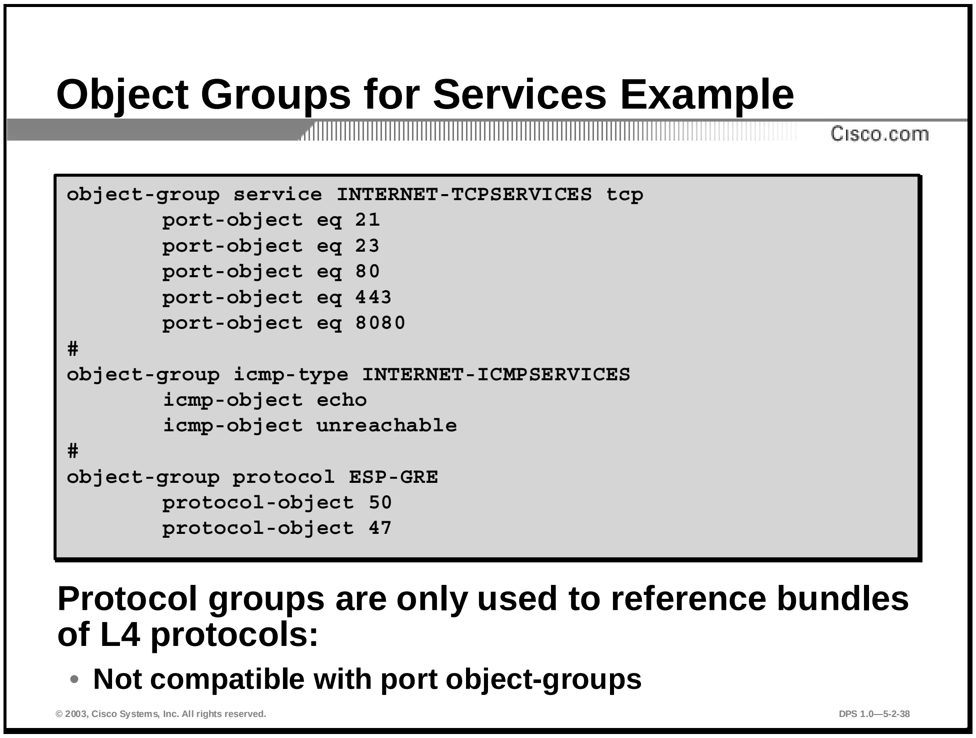
DPS 1.0—5-2-37

Service bundles are used to reduce necessary configuration steps and shorten the configuration, making it more readable and easier to fit in PIX Firewall flash memory.

The following are two examples of the use of an object group once it is defined:

- **conduit permit tcp** *object-group group_name* **any**
- **access-list** *acl_name* **permit tcp any** *object-group group_name*

In these two examples *group_name* is the name of the group.



## Object Groups for Services Example

Cisco.com

```
object-group service INTERNET-TCPSERVICES tcp
       port-object eq 21
       port-object eq 23
       port-object eq 80
       port-object eq 443
       port-object eq 8080
#
object-group icmp-type INTERNET-ICMPSERVICES
       icmp-object echo
       icmp-object unreachable
#
object-group protocol ESP-GRE
       protocol-object 50
       protocol-object 47
```

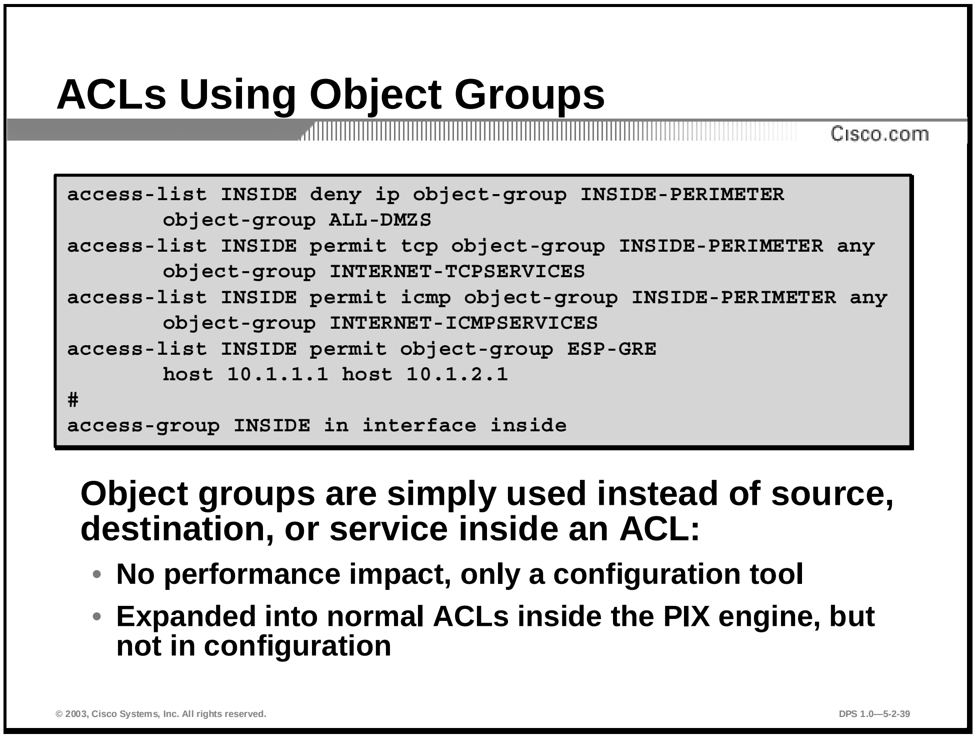
**Protocol groups are only used to reference bundles of L4 protocols:**

- **Not compatible with port object-groups**

DPS 1.0—5-2-38

## Example

Services can be grouped together in protocol groups, similarly to Cisco Secure Policy Manger’s Service Bundle functionality. This will significantly reduce access rules, when multiple networks are allowed to use the same service set. Protocol groups and port groups are supported, grouping together L3 and L4 protocols respectively.



## ACLs Using Object Groups

```
access-list INSIDE deny ip object-group INSIDE-PERIMETER
       object-group ALL-DMZS
access-list INSIDE permit tcp object-group INSIDE-PERIMETER any
       object-group INTERNET-TCPSERVICES
access-list INSIDE permit icmp object-group INSIDE-PERIMETER any
       object-group INTERNET-ICMPSERVICES
access-list INSIDE permit object-group ESP-GRE
       host 10.1.1.1 host 10.1.2.1
#
access-group INSIDE in interface inside
```

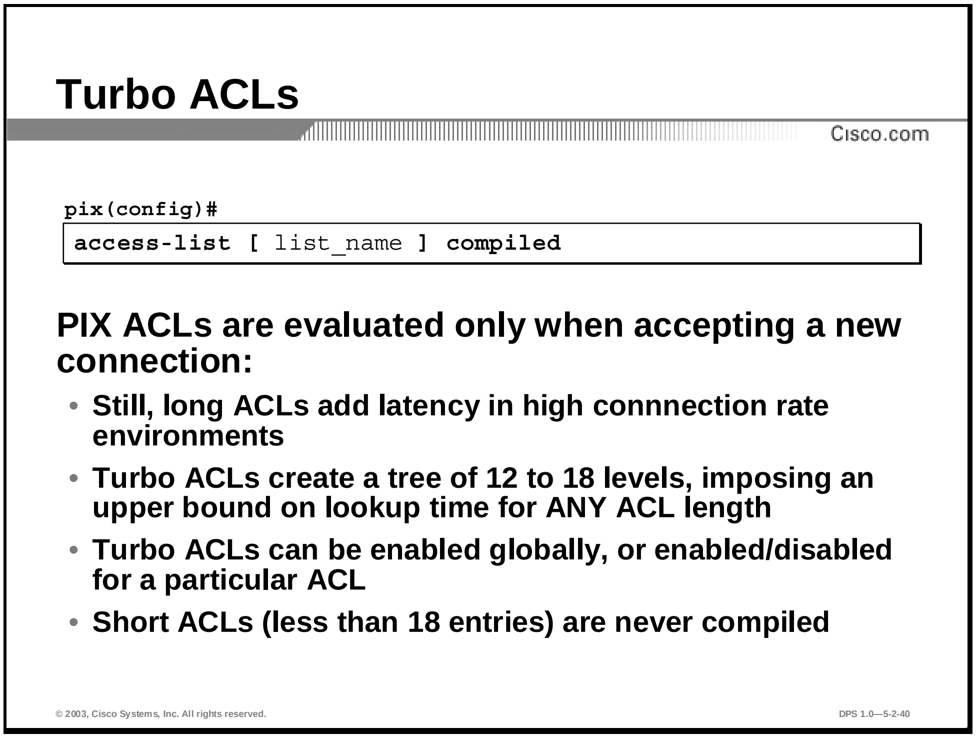
**Object groups are simply used instead of source, destination, or service inside an ACL:**

- **No performance impact, only a configuration tool**
- **Expanded into normal ACLs inside the PIX engine, but not in configuration**

DPS 1.0—5-2-39

## Using Object Groups in ACLs

Object groups are then used inside access lists instead of the source, destination, or service conditions. There is no performance impact, as the object groups are automatically expaneded into normal ACLs inside the PIX runtime engine. However, object groups significantly reduce the configuration size, which is a limited resource on the PIX Firewall.



## Turbo ACLs

```
pix(config)#
access-list [ list_name ] compiled
```

**PIX ACLs are evaluated only when accepting a new connection:**

- **Still, long ACLs add latency in high connnection rate environments**
- **Turbo ACLs create a tree of 12 to 18 levels, imposing an upper bound on lookup time for ANY ACL length**
- **Turbo ACLs can be enabled globally, or enabled/disabled for a particular ACL**
- **Short ACLs (less than 18 entries) are never compiled**

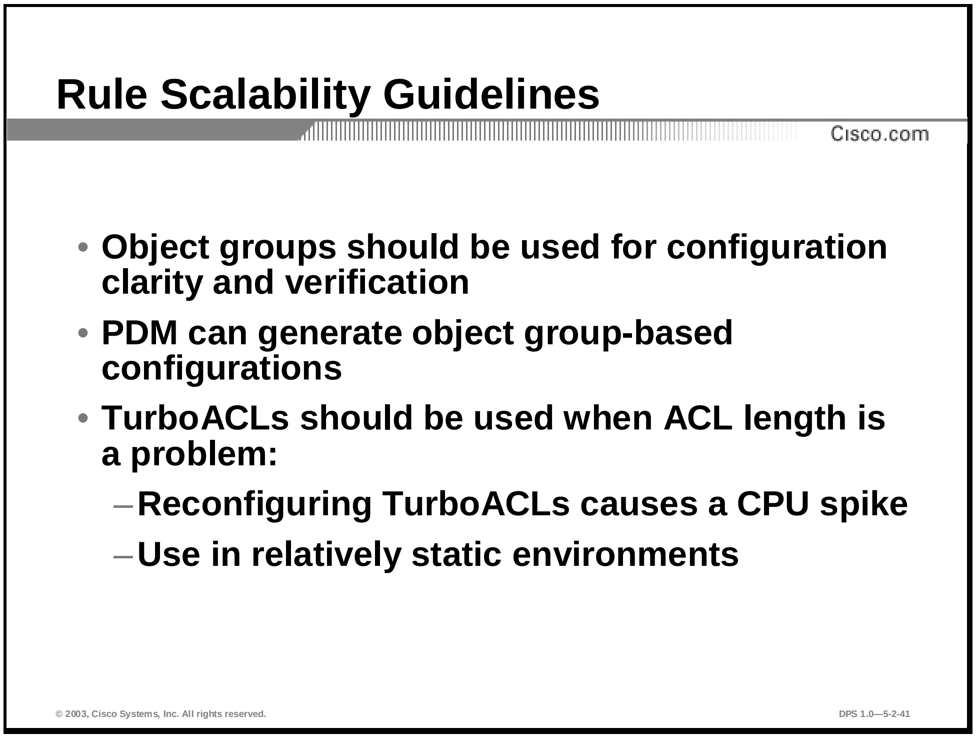
DPS 1.0—5-2-40

# Turbo ACLs

The Turbo ACL feature simple creates data tables for faster searching of elements within the table for ACL processing. Although taking more memory, it can significantly enhance the performance of the firewall.

The minimum memory required for TurboACL is 2.1 MB and approximately 1 MB of memory is required for every 2000 ACL elements.

Low-end PIXen, such as the 501 may be adversely affected by enabling this, as may other PIXen running other intense processes like PIX Device Manager 2.01 or later.



## Rule Scalability Guidelines

Cisco.com

- Object groups should be used for configuration clarity and verification
- PDM can generate object group-based configurations
- TurboACLs should be used when ACL length is a problem:
  - Reconfiguring TurboACLs causes a CPU spike
  - Use in relatively static environments

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—5-2-41

## Guidelines

To maintain logic and simplicity within a complex configuration, object groups are strongly recommended.

Turbo ACLs are also recommended for speed of processing in large access-list environments. Be aware of the performance hit to process the lists initially, when the ACL is deployed or changed.

## Practice

Q1) When is it not advisable to enable the PIX Turbo ACL feature?

A) when all ACLs are less than 18 entries long

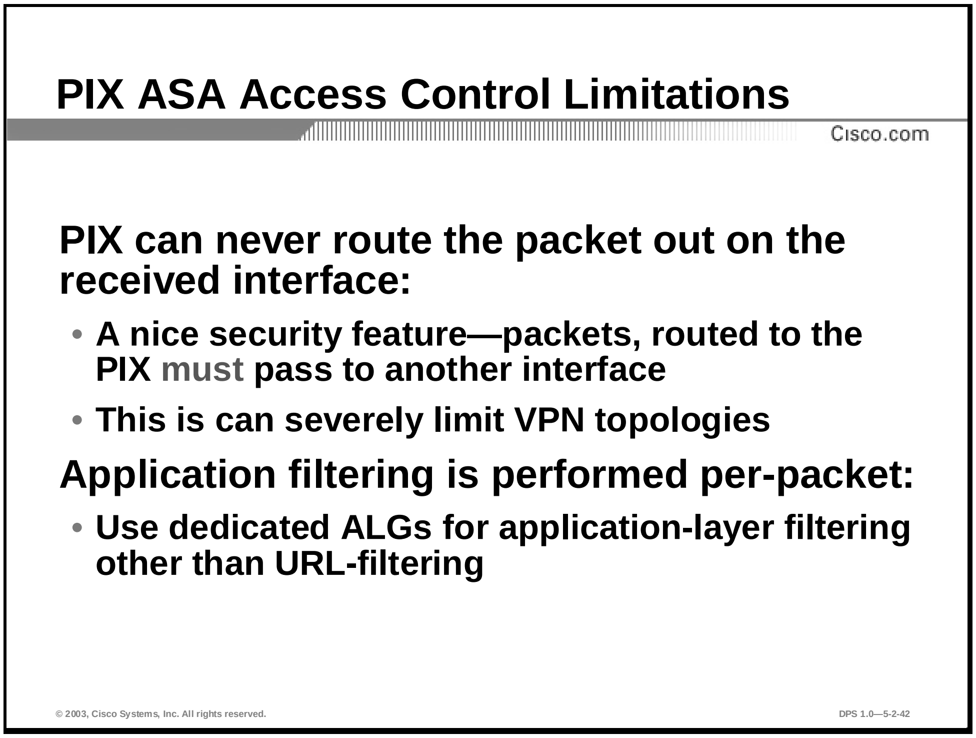
B) there are no caveats and Turbo ACLs are always on

C) in high connection-rate environments

D) when all ACLs are less than 180 entries long

E) if the ACLs do not change much

# PIX Access Control Limitations



## PIX ASA Access Control Limitations

Cisco.com

PIX can never route the packet out on the received interface:

- A nice security feature—packets, routed to the PIX **must** pass to another interface
- This is can severely limit VPN topologies

Application filtering is performed per-packet:

- Use dedicated ALGs for application-layer filtering other than URL-filtering

DPS 1.0—5-2-42

## Objectives

This section will enable the learner to identify limitations of the Cisco Secure PIX Firewall product and avoid them in a firewall design.

## Introduction

The PIX can never route a packet out on the same interface, on which the packet has been received. This can be a nice feature, since an external device (such as a router with policy routing) can force packets from multiple outside networks to the PIX, and this PIX property guarantees, that packets are never routed back. This can nicely isolate multiple external networks, which need to be totally separated, such as an organization’s business partners.

The basic design rule is that a PIX is a firewall, not a router. The PIX will also not allow address translations from same-to-same interfaces.

# Practice

Q1) What happens if the PIX must route the packet out on the same interface, on which the packet was received?

A) the packet is compared only against the inbound ACL on that interface

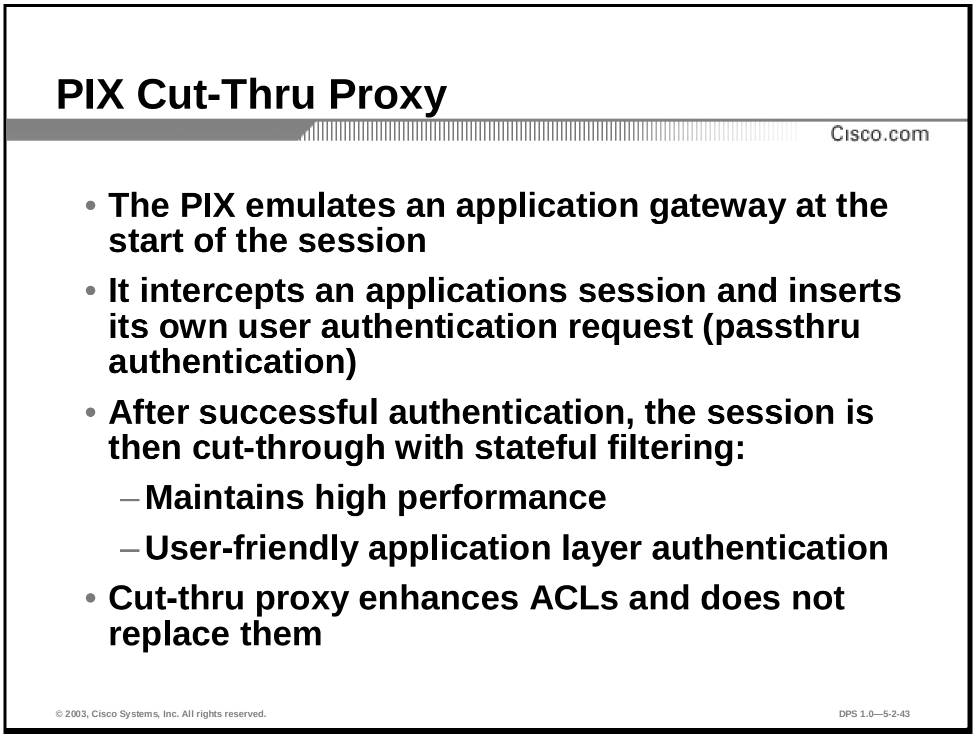
B) the packet is dropped

C) the packet is compared against the inbound and outbound ACL on that interface

D) the packet must necessarily be encrypted

E) the packet bypasses the security rules

# PIX Advanced Cut-Thru Proxy Features



## PIX Cut-Thru Proxy

- The PIX emulates an application gateway at the start of the session
- It intercepts an applications session and inserts its own user authentication request (passthru authentication)
- After successful authentication, the session is then cut-through with stateful filtering:
  - Maintains high performance
  - User-friendly application layer authentication
- Cut-thru proxy enhances ACLs and does not replace them

© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—5-2-43

## Objective

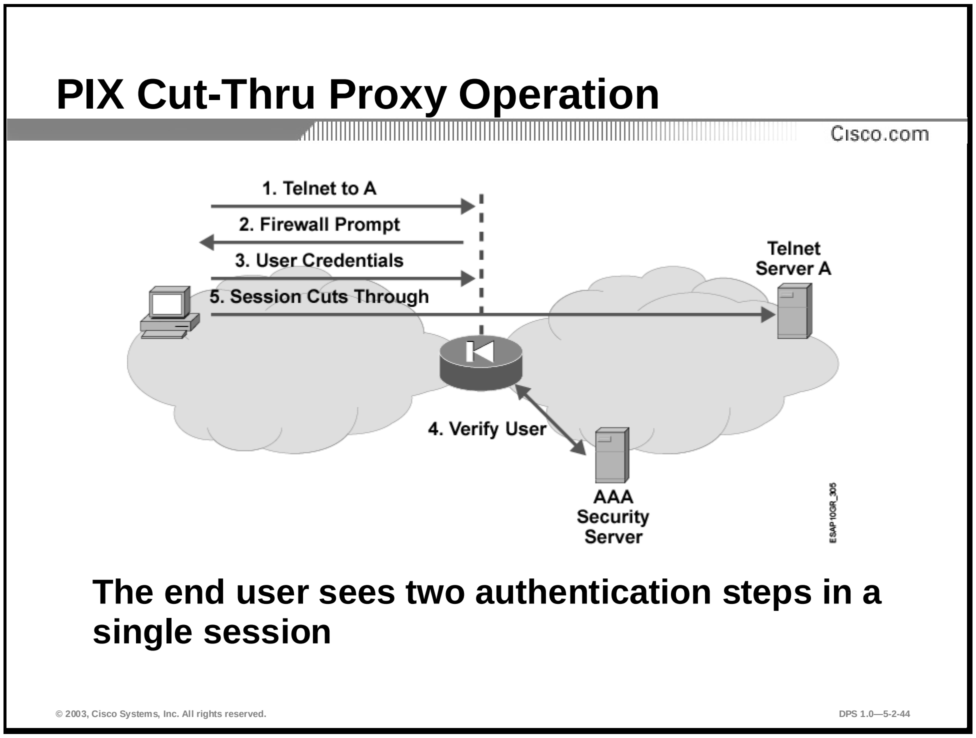
This section will enable the learner to identify and configure advanced features of the PIX Firewall cut-thru proxy feature and choose it in a firewall design.

## Introduction

The PIX Firewall cut-thru proxy is the classic PIX user authentication feature, which can authenticate users when passing over a firewall. The user will receive multiple authentication prompts. The first set of prompts is from the PIX to authenticate and allow to the ASA phase. The second set of prompts will be from the end station application itself.

The cut-thru proxy behaves like an application gateway at the beginning of the session, just enough to authenticate the user on the application layer. Once the user is authenticated, all sessions are cut-thru using stateful packet filtering only, maintaining the high SPF performance.

**Note** PIX cut-thru proxy classic authentication and authorization does not replace ACLs, but enhances them with more specific per-user access rules.



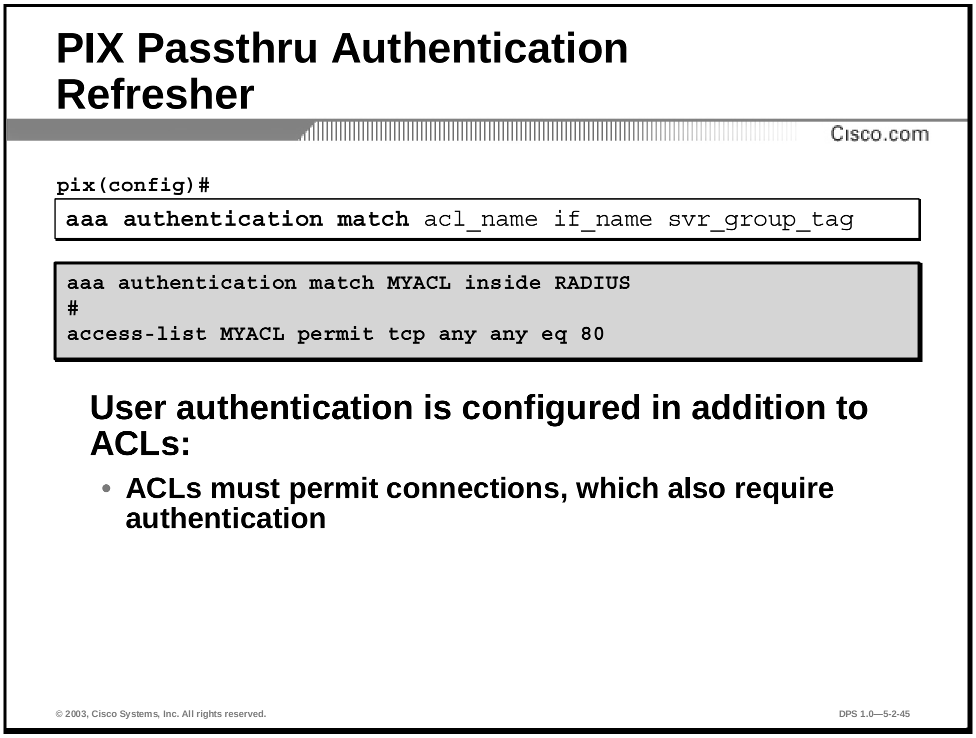
## PIX Cut-Thru Proxy Operation

Cisco.com

**The end user sees two authentication steps in a single session**

DPS 1.0—5-2-44

When a user session is passing through the PIX firewall, the end user will generally see two prompts in the authentication process—first the firewall authentication prompt, and then the destination server’s.



## PIX Passthru Authentication Refresher

```
pix(config)#
aaa authentication match acl_name if_name svr_group_tag
```

```
aaa authentication match MYACL inside RADIUS
#
access-list MYACL permit tcp any any eq 80
```

**User authentication is configured in addition to ACLs:**

- **ACLs must permit connections, which also require authentication**

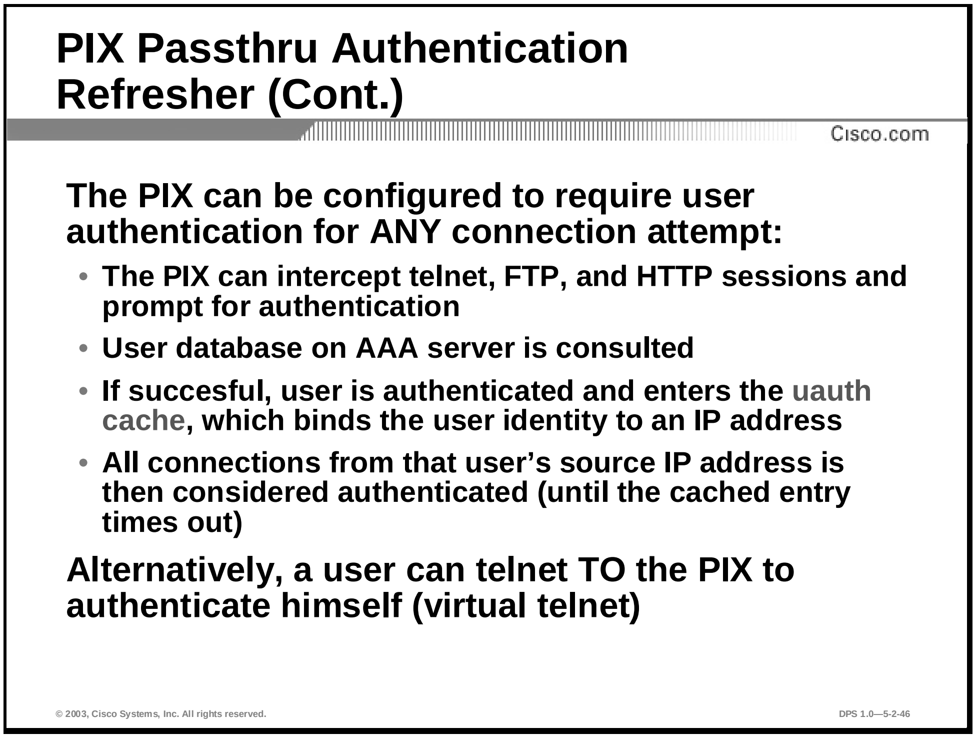
DPS 1.0—5-2-45

# PIX AAA Authentication Proxy Configuration Refresher

If the access-list does not permit the connection, then authenticating on the PIX will accomplish nothing.

The **aaa authentication** command can be used to specify which service is to be handled. Options here are ftp, http, telnet or any. “Any” implies TCP sessions.

Authentication can also be performed on successful match of an ACL entry. This gives complete granularity to the control of the cut-thru authentication performed by the PIX. It can be as selective or permissive as necessary.



## PIX Passthru Authentication Refresher (Cont.)

**The PIX can be configured to require user authentication for ANY connection attempt:**

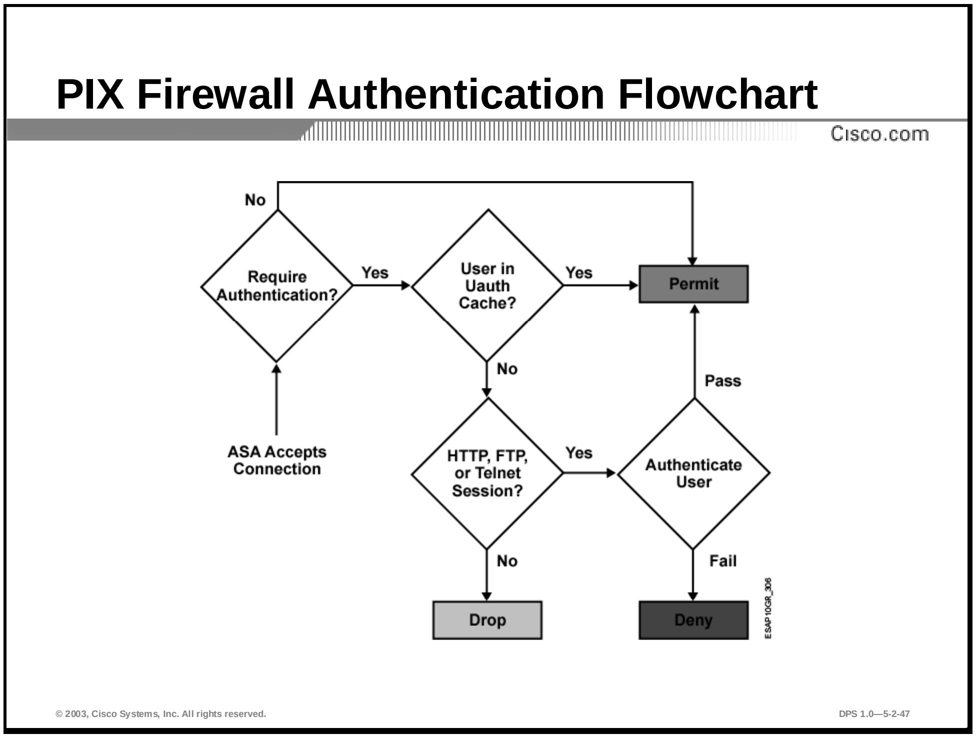
- **The PIX can intercept telnet, FTP, and HTTP sessions and prompt for authentication**
- **User database on AAA server is consulted**
- **If succesful, user is authenticated and enters the uauth cache, which binds the user identity to an IP address**
- **All connections from that user's source IP address is then considered authenticated (until the cached entry times out)**

**Alternatively, a user can telnet TO the PIX to authenticate himself (virtual telnet)**

DPS 1.0—5-2-46

The PIX can be configured to require user authentication for any session across the PIX, as specified in the "aaa authentication" commands. However, to authenticate the user, only telnet, FTP, or HTTP sessions can be intercepted, and the user authenticated. The PIX caches user credentials in the uauth cache, which expires after an idle or absolute timeout, forcing the user to reauthenticate. After the timer expires, existing sessions are not required to reauthenticate.

With HTTP, once authenticated, a user never has to reauthenticate no matter how low the PIX Firewall uauth timeout is set. This is because the browser caches the "Authorization: Basic=Uuhjksdkfhk=" string in every subsequent connection to that particular site. This can *only* be cleared when the user exits *all* instances of Netscape Navigator or Internet Explorer and restarts. Flushing the cache is of no use.



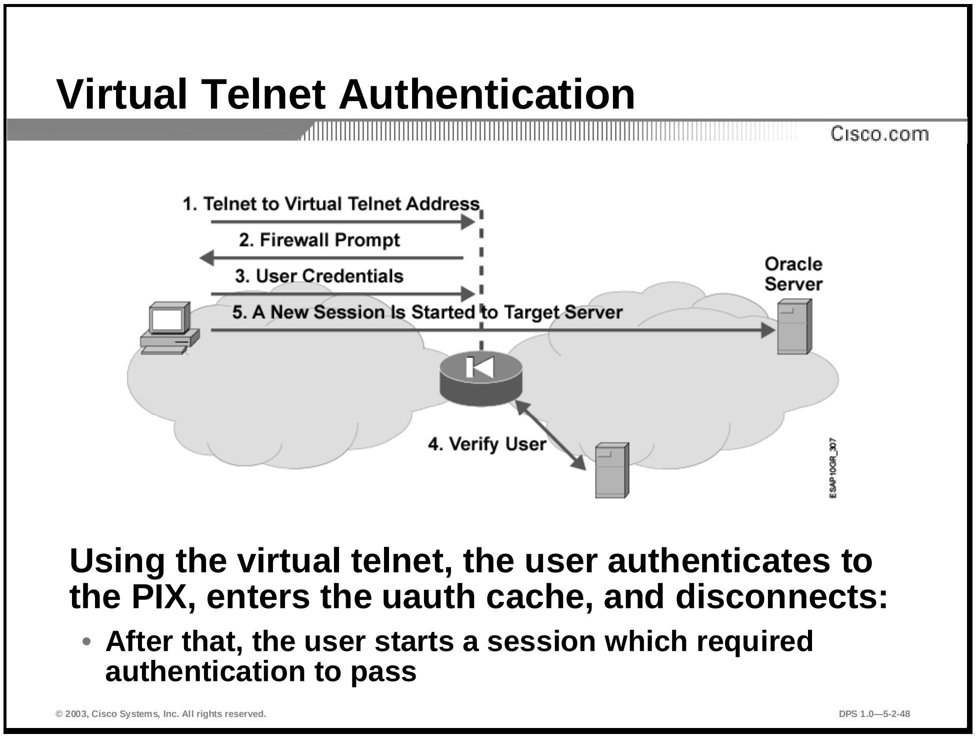
## PIX Firewall Authentication Flowchart

Cisco.com

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—5-2-47

The logical flow of the passthru authentication is fairy simple to follow. If necessary, authentication will occur, if not, the packet goes straight through the ASA.



## Virtual Telnet Authentication

1. Telnet to Virtual Telnet Address
2. Firewall Prompt
3. User Credentials
5. A New Session Is Started to Target Server
Oracle Server
4. Verify User
ESAP10GR_307

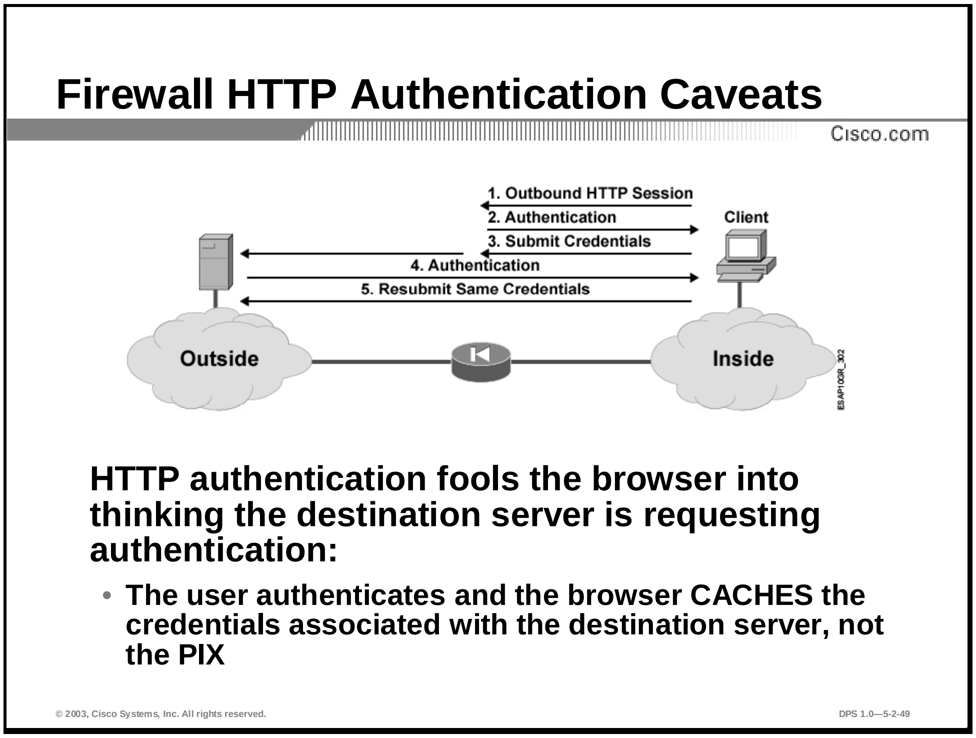
**Using the virtual telnet, the user authenticates to the PIX, enters the uauth cache, and disconnects:**

- **After that, the user starts a session which required authentication to pass**

DPS 1.0—5-2-48

## Virtual Telnet

This can be used to improve the user experience for authenticating sessions other than telnet, FTP, or HTTP. A user logs on to the virtual telnet server, which displays an authentication success message. Then, the user can start another application, which required authentication to pass.



## Firewall HTTP Authentication Caveats

**HTTP authentication fools the browser into thinking the destination server is requesting authentication:**

- **The user authenticates and the browser CACHES the credentials associated with the destination server, not the PIX**

DPS 1.0—5-2-49

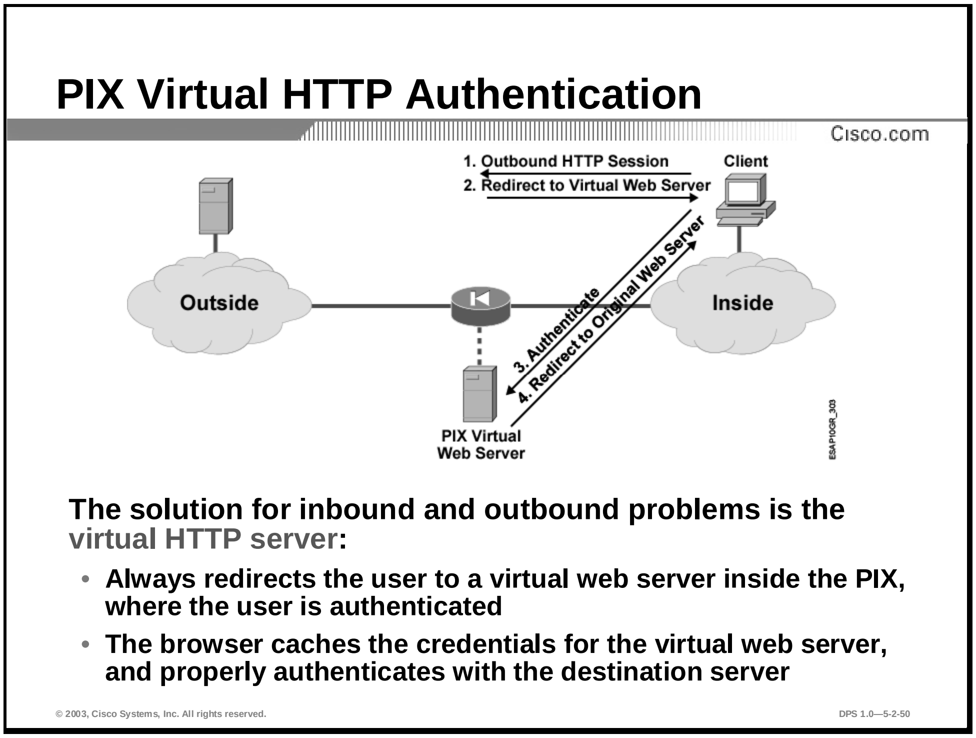
## HTTP Authentication Caveats

There are two significant issues, if the real destination server also requests authentication:

- The browser will submit cached information to the server, who will not accept it (functionality issue)

- The browser will submit cached information to the server, who might record it (security issue)

Remember that HTTP is a clear-text protocol and subject to information interception.



## PIX Virtual HTTP Authentication

**The solution for inbound and outbound problems is the virtual HTTP server:**

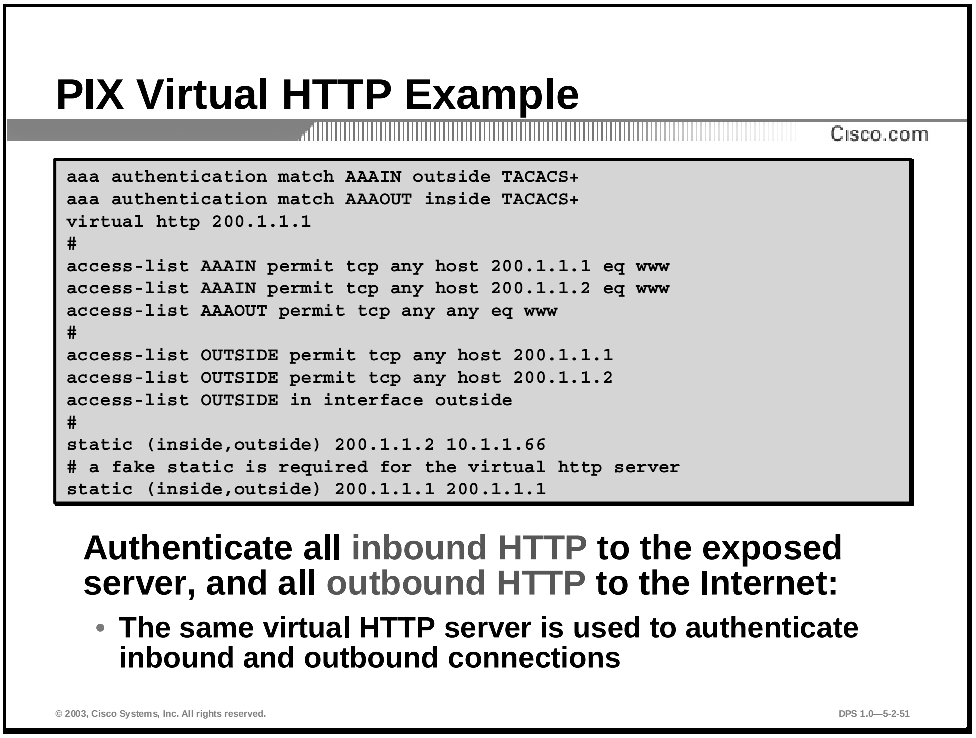
- **Always redirects the user to a virtual web server inside the PIX, where the user is authenticated**
- **The browser caches the credentials for the virtual web server, and properly authenticates with the destination server**

DPS 1.0—5-2-50

## Virtual HTTP

The **virtual http** command lets web browsers work correctly with the PIX Firewall **aaa** command. The **aaa** command assumes that the AAA server database is shared with a web server. PIX Firewall automatically provides the AAA server and web server with the same information. The **virtual http** command works with the **aaa** command to authenticate the user, separate the AAA server information from the web client's URL request, and direct the web client to the web server. Use the **show virtual http** command to list commands in the configuration. Us the **no virtual http** command to disable its use.

The **virtual http** command works by redirecting the web browser's initial connection to the *ip_address*, which resides in the PIX Firewall, authenticating the user, then redirecting the browser back to the URL which the user originally requested. This mechanism comprises the PIX Firewall's virtual server feature.



## PIX Virtual HTTP Example

Cisco.com

```
aaa authentication match AAAIN outside TACACS+
aaa authentication match AAAOUT inside TACACS+
virtual http 200.1.1.1
#
access-list AAAIN permit tcp any host 200.1.1.1 eq www
access-list AAAIN permit tcp any host 200.1.1.2 eq www
access-list AAAOUT permit tcp any any eq www
#
access-list OUTSIDE permit tcp any host 200.1.1.1
access-list OUTSIDE permit tcp any host 200.1.1.2
access-list OUTSIDE in interface outside
#
static (inside,outside) 200.1.1.2 10.1.1.66
# a fake static is required for the virtual http server
static (inside,outside) 200.1.1.1 200.1.1.1
```

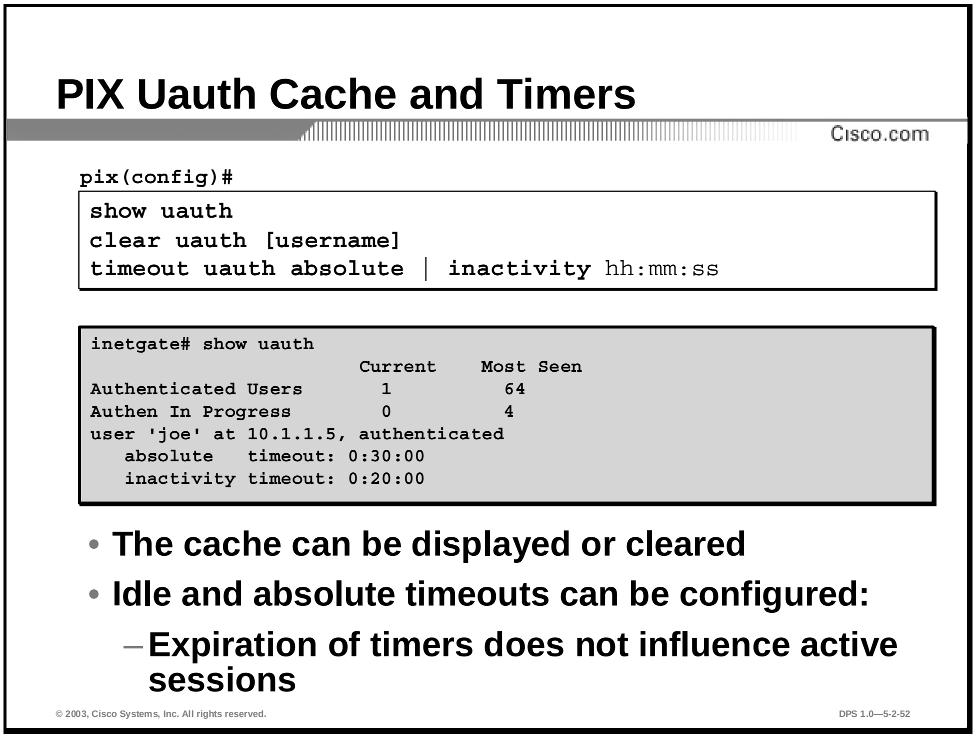
**Authenticate all inbound HTTP to the exposed server, and all outbound HTTP to the Internet:**

- **The same virtual HTTP server is used to authenticate inbound and outbound connections**

DPS 1.0—5-2-51

## Example

This example shows a configuration for the virtual HTTP server. Inbound and outbound sessions are authenticated via the virtual HTTP server. For inbound authentication, an xlate must be set up for the virtual http server as well.



## PIX Uauth Cache and Timers

```
pix(config)#
show uauth
clear uauth [username]
timeout uauth absolute | inactivity hh:mm:ss
```

```
inetgate# show uauth
                        Current    Most Seen
Authenticated Users       1          64
Authen In Progress        0          4
user 'joe' at 10.1.1.5, authenticated
   absolute   timeout: 0:30:00
   inactivity timeout: 0:20:00
```

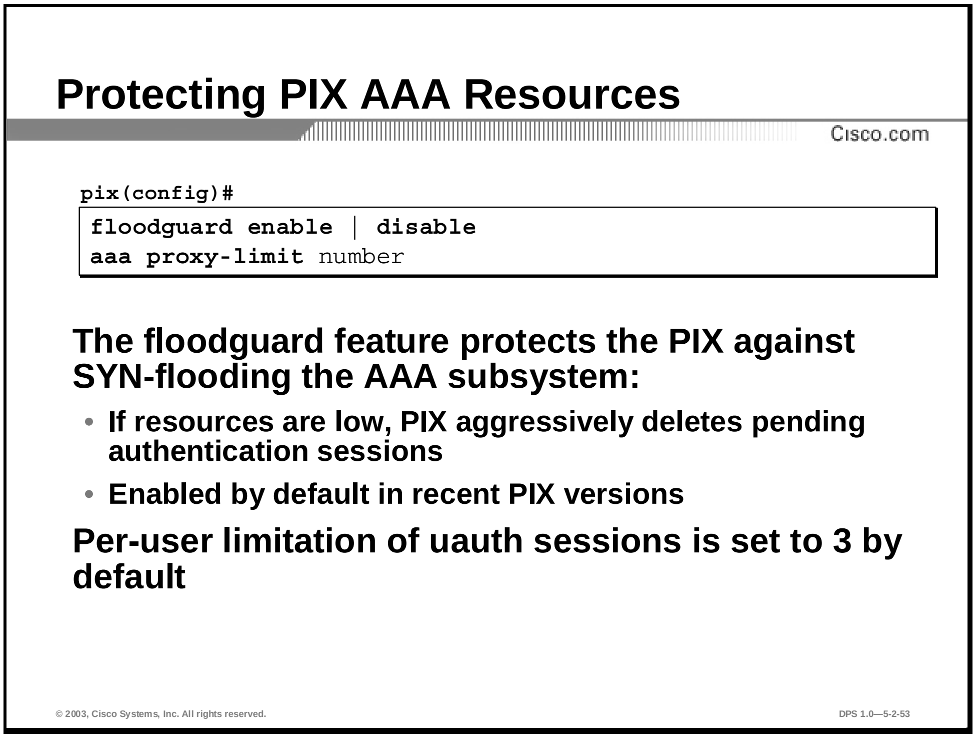
- **The cache can be displayed or cleared**
- **Idle and absolute timeouts can be configured:**
  - **Expiration of timers does not influence active sessions**

DPS 1.0—5-2-52

# PIX Uauth Cache

The PIX Uauth cache consists of mappings between authenticated users and their source IP addresses. When a user enters the uauth cache, all sessions from that IP address are considered to be authenticated. If the path between the user and the PIX is protected by IPSec, such authentication is reliable, as attackers cannot spoof the address of the authenticated user. If cleartext connectivity is used, man-in-the-middle attacks are possible to impersonate authenticated users.

The uauth cache can be cleared, and timeouts can be set to clear the cache quickly if a user is idle foe extended periods of time. A timeout of “0” will allow indefinite authentication. This is not recommended.



## Protecting PIX AAA Resources

```
pix(config)#
floodguard enable | disable
aaa proxy-limit number
```

**The floodguard feature protects the PIX against SYN-flooding the AAA subsystem:**

- **If resources are low, PIX aggressively deletes pending authentication sessions**
- **Enabled by default in recent PIX versions**

**Per-user limitation of uauth sessions is set to 3 by default**

DPS 1.0—5-2-53

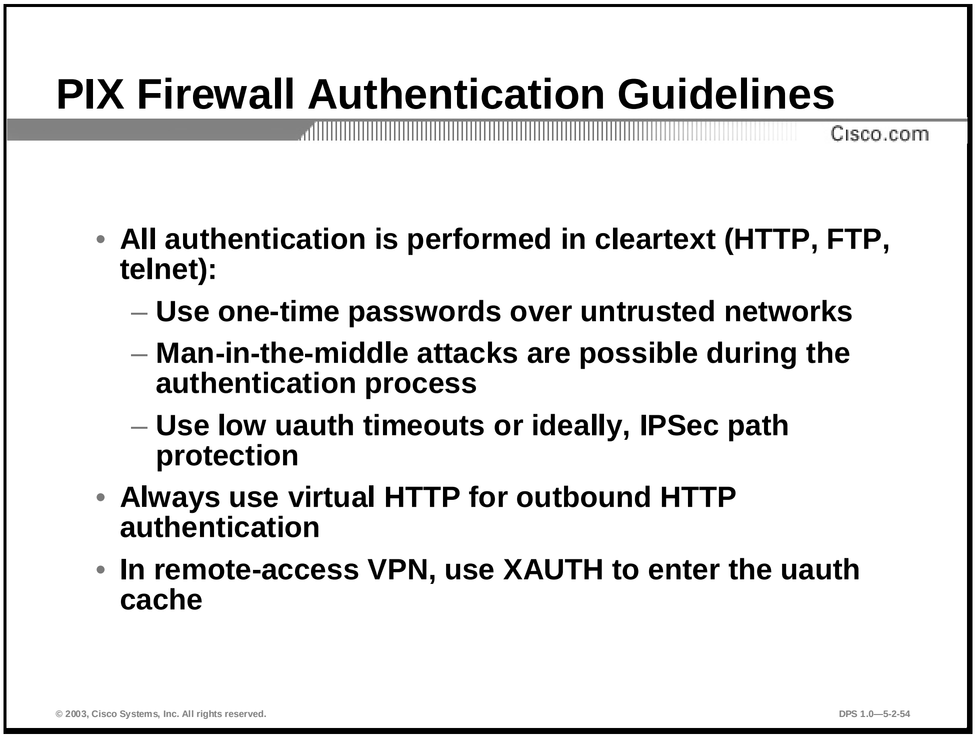
## Floodguard

Floodguard is enabled by default. The **floodguard** command lets you reclaim PIX Firewall resources if the user authentication (uauth) subsystem runs out of resources. If an inbound or outbound uauth connection is being attacked or overused, the PIX Firewall will actively reclaim TCP user resources.

If the PIX Firewall uauth subsystem is depleted, TCP user resources in different states are reclaimed depending on urgency in the following order:

1. Timewait
2. FinWait
3. Embryonic
4. Idle

The **aaa proxy-limit** command enables you to manually configure the uauth session limit by setting the maximum number of concurrent proxy connections allowed per user. By default, this value is set to 3. If a source address is a proxy server, consider excluding this IP address from authentication or increasing the number of allowable outstanding AAA requests.



## PIX Firewall Authentication Guidelines

- All authentication is performed in cleartext (HTTP, FTP, telnet):
  - Use one-time passwords over untrusted networks
  - Man-in-the-middle attacks are possible during the authentication process
  - Use low uauth timeouts or ideally, IPSec path protection
- Always use virtual HTTP for outbound HTTP authentication
- In remote-access VPN, use XAUTH to enter the uauth cache

DPS 1.0—5-2-54

## Guidelines

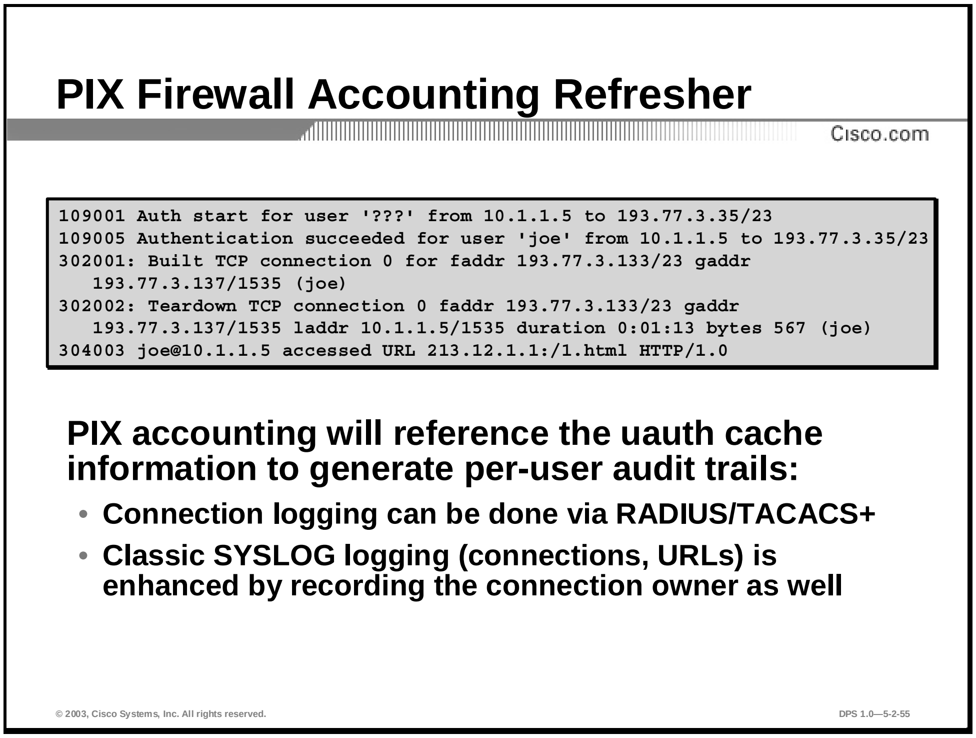
The following firewall passthru user authentication guidelines apply, when the PIX Firewall's classic cut-thru proxy authentication is used

- Use one-time passwords over untrusted networks, as only telnet, FTP, and HTTP (all cleartext) can be used to authenticate to the PIX

- Use IPSec between the user and the PIX, if possible, to mitigate the risk of man-in-the-middle attack at or after authentication

- Always use virtual HTTP for outbound HTTP authentication to prevent user credentials being sent to untrusted networks

- In VPNs, XAUTH can be used to automatically pass the IKE user authentication credentials to the PIX uauth cache



## PIX Firewall Accounting Refresher

```
109001 Auth start for user '???' from 10.1.1.5 to 193.77.3.35/23
109005 Authentication succeeded for user 'joe' from 10.1.1.5 to 193.77.3.35/23
302001: Built TCP connection 0 for faddr 193.77.3.133/23 gaddr
   193.77.3.137/1535 (joe)
302002: Teardown TCP connection 0 faddr 193.77.3.133/23 gaddr
   193.77.3.137/1535 laddr 10.1.1.5/1535 duration 0:01:13 bytes 567 (joe)
304003 joe@10.1.1.5 accessed URL 213.12.1.1:/1.html HTTP/1.0
```

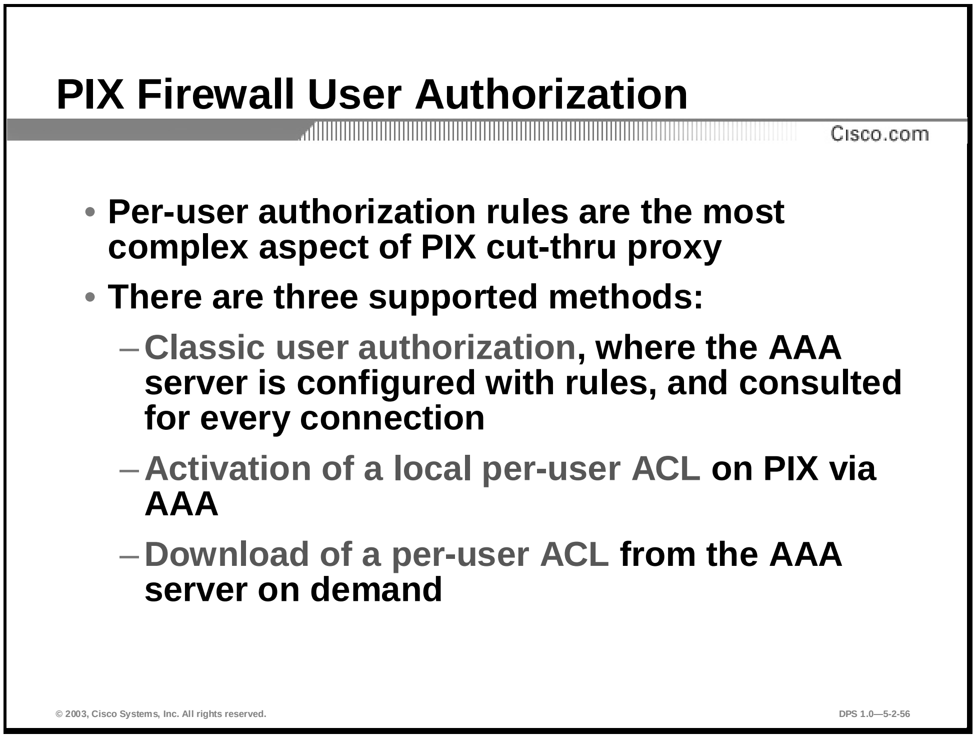
**PIX accounting will reference the uauth cache information to generate per-user audit trails:**

- **Connection logging can be done via RADIUS/TACACS+**
- **Classic SYSLOG logging (connections, URLs) is enhanced by recording the connection owner as well**

DPS 1.0—5-2-55

## PIX AAA Accounting Refresher

The PIX can account for permitted connections over SYSLOG or TACACS+/RADIUS. If AAA authentication is enabled, all accounting records will have the username of the connection owner appended to them.



## PIX Firewall User Authorization

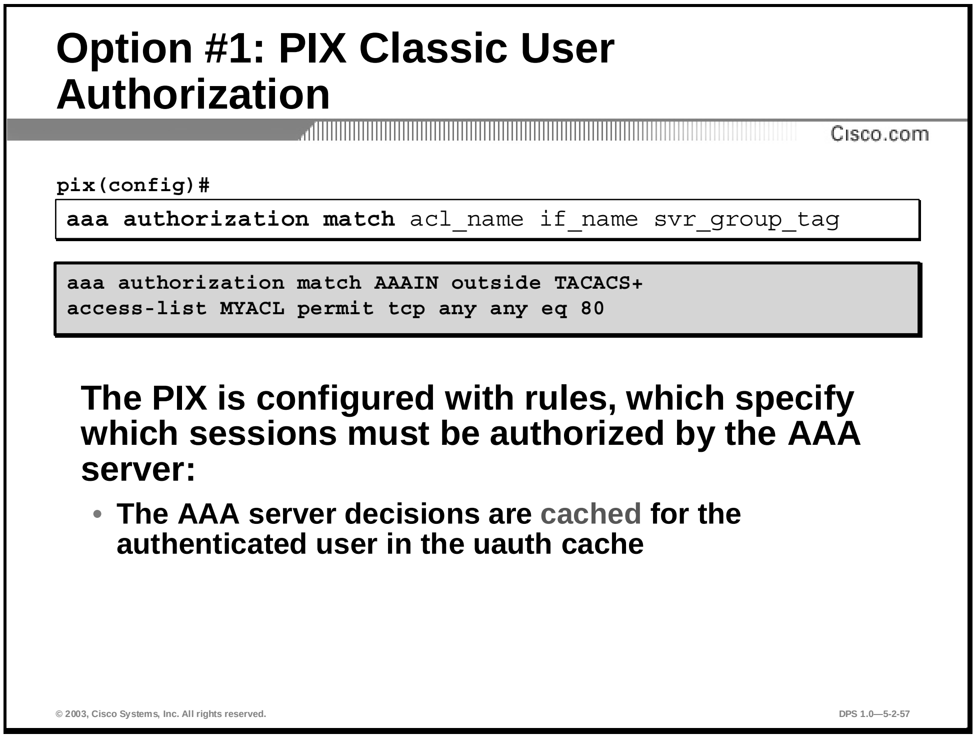
- Per-user authorization rules are the most complex aspect of PIX cut-thru proxy
- There are three supported methods:
  - Classic user authorization, where the AAA server is configured with rules, and consulted for every connection
  - Activation of a local per-user ACL on PIX via AAA
  - Download of a per-user ACL from the AAA server on demand

DPS 1.0—5-2-56

## PIX AAA Authorization

The PIX supports three methods of user authorization (that is, specifying per-user access rules in the context of firewall AAA)

- Classic user authorization, where the access rules are configured on the AAA server and consulted on-demand
- Activation of a local ACLs after a user has authenticated
- Download of a per-user ACL from the AAA server after a user has authenticated



## Option #1: PIX Classic User Authorization

```
pix(config)#
aaa authorization match acl_name if_name svr_group_tag
```

```
aaa authorization match AAAIN outside TACACS+
access-list MYACL permit tcp any any eq 80
```

**The PIX is configured with rules, which specify which sessions must be authorized by the AAA server:**

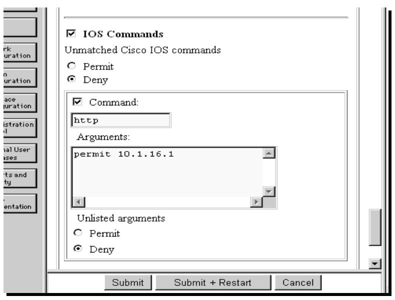
- **The AAA server decisions are cached for the authenticated user in the uauth cache**

# PIX Classic User Authorization

With classic authorization, the PIX is configured with rules, specifying which connections need to be authorized by the AAA server. The AAA server will be consulted for access rights on demand, and its responses will be cached in the uauth cache.

## PIX Classic User Authorization

Cisco.com



☑ IOS Commands

Unmatched Cisco IOS commands

○ Permit
● Deny

☑ Command:

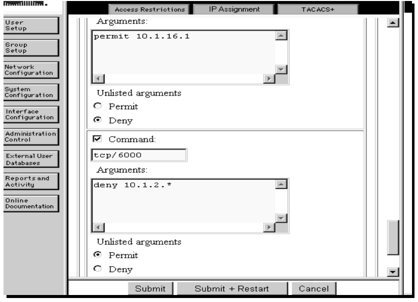
http

Arguments:

permit 10.1.16.1

Unlisted arguments

○ Permit
● Deny



Arguments:

permit 10.1.16.1

Unlisted arguments

○ Permit
● Deny

☑ Command:

tcp/6000

Arguments:

deny 10.1.2.*

Unlisted arguments

● Permit
○ Deny

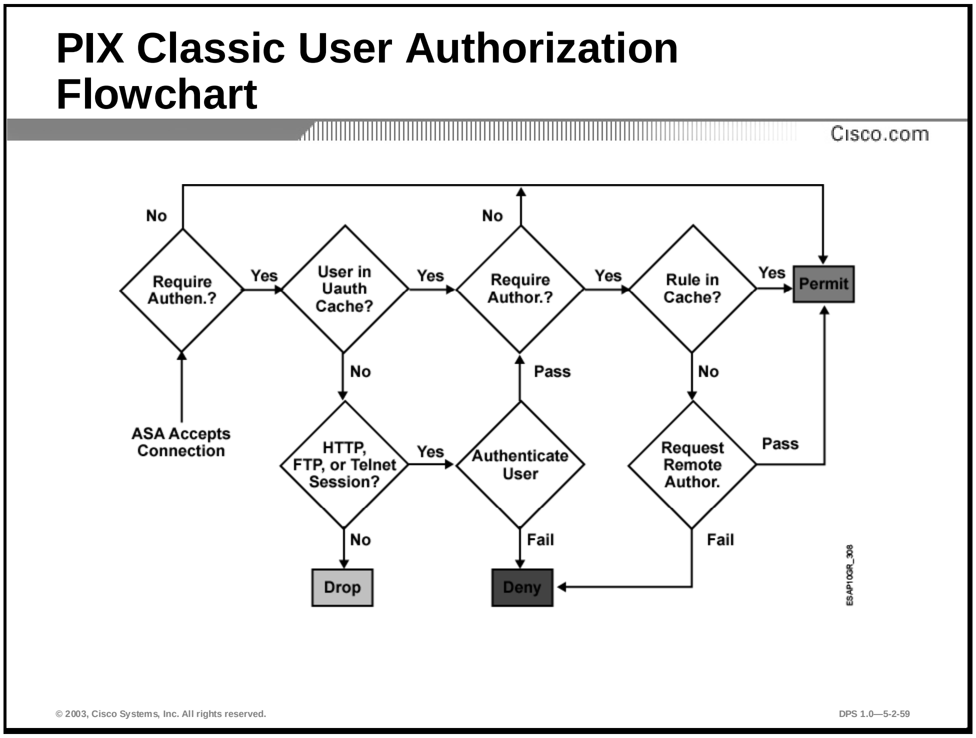
**The AAA server is configured with authorization rules in the "command authorization" section:**

- **The PIX fakes it is an IOS router, and translates user sessions to IOS-style commands**
- **A telnet session to host 10.1.1.1 is sent to the AAA server as an IOS command "telnet 10.1.1.1"**

DPS 1.0—5-2-58

## PIX Classic User Authorization on the AAA Server

The PIX is configured on the ACS as a router, which is authorizing IOS commands. Actually, the PIX translates a session over the PIX into a "fake" IOS command, and demands authorization from the ACS.

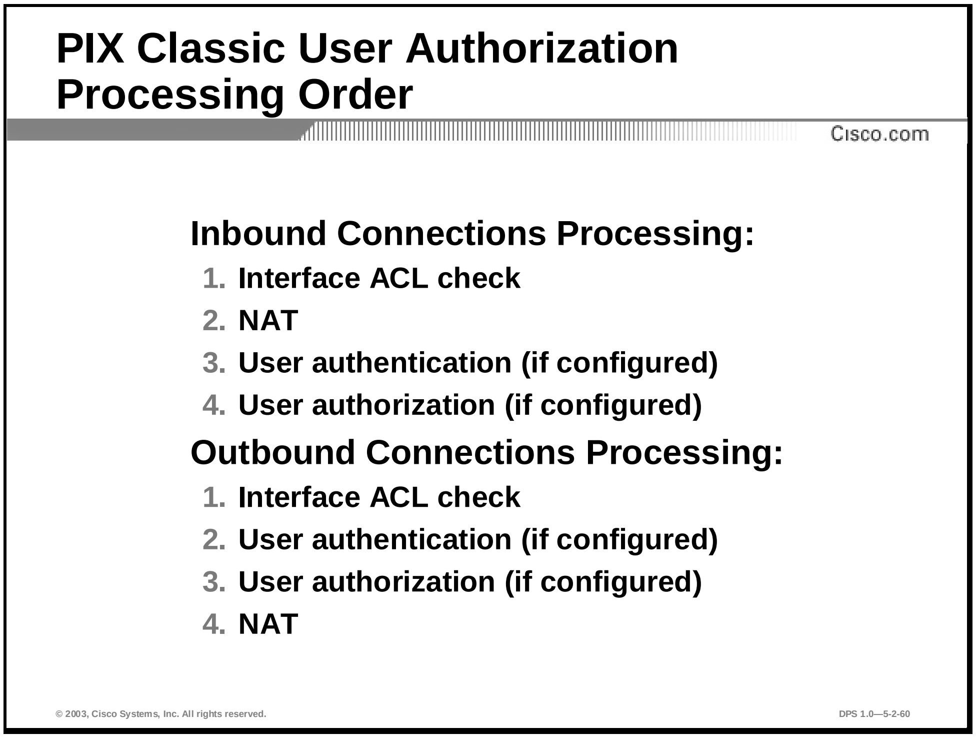


# PIX Classic User Authorization Flowchart

Cisco.com

DPS 1.0—5-2-59

This figure presents the logic behind PIX user authentication, followed by user authorization. As a packet goes through the ASA, and it is determined that a packet needs authorization, the uauth cache is checked. If no entry exists, and the session type matches, the AAA server is consulted, and the session permitted or denied.



## PIX Classic User Authorization Processing Order

**Inbound Connections Processing:**

1. **Interface ACL check**
2. **NAT**
3. **User authentication (if configured)**
4. **User authorization (if configured)**

**Outbound Connections Processing:**

1. **Interface ACL check**
2. **User authentication (if configured)**
3. **User authorization (if configured)**
4. **NAT**

DPS 1.0—5-2-60

# PIX Classic Cut-Thru Authentication and Authorization Order

The order of PIX rules application for inbound connections is

1. Interface ACL check and ASA
2. NAT
3. User authentication
4. User authorization

For outbound connections, the order changes slightly

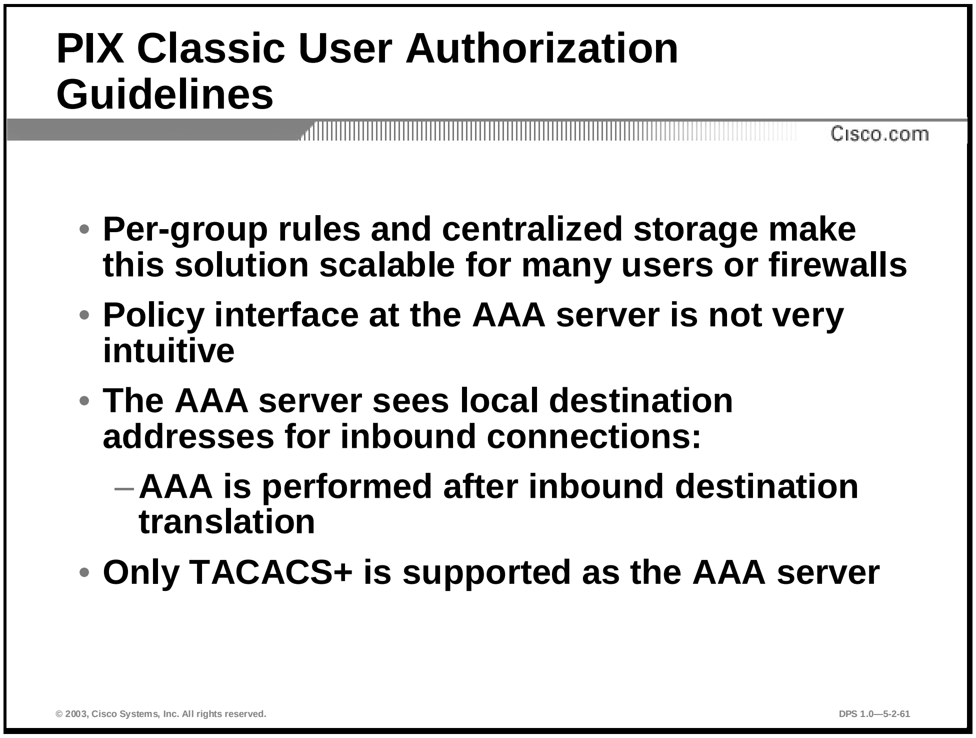
1. Interface ACL check and ASA
2. NAT
3. User authentication
4. User authorization

From this order, several design and configuration guidelines can be extracted:

- Classic AAA does not replace ACLs – instead, it is evaluated AFTER ACLs, which means that connections, permitted through AAA, must also be permitted in interface ACLs
- AAA occurs after NAT for inbound connections, and before NAT for outbound connections. Therefore, all addresses in access rules should refer to LOCAL addresses (i.e. not translated addresses).

For more details, refer to:

- http://www-tac.cisco.com/Support_Library/Hardware/PIX/pix51flow.htm



## PIX Classic User Authorization Guidelines

Cisco.com

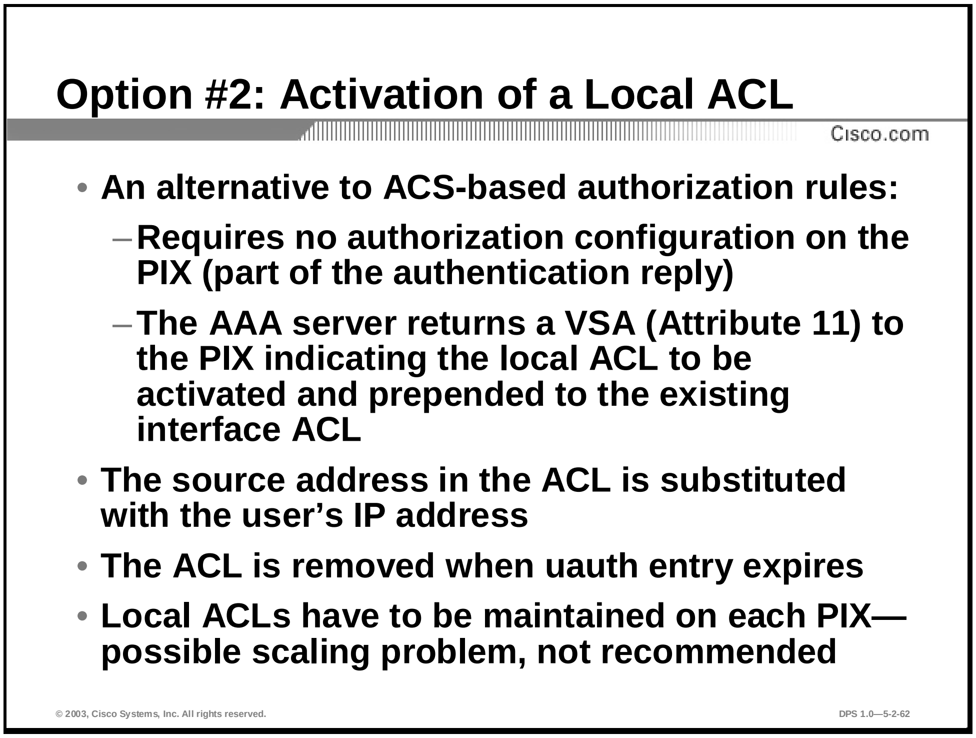
- Per-group rules and centralized storage make this solution scalable for many users or firewalls
- Policy interface at the AAA server is not very intuitive
- The AAA server sees local destination addresses for inbound connections:
  - AAA is performed after inbound destination translation
- Only TACACS+ is supported as the AAA server

DPS 1.0—5-2-61

## Guidelines

While this is a very scalable solution, it may represent some functionality difficulties. First, the policy interface at the AAA server is not very intuitive, as it requires configuring IOS “command authorization” permissions to permit sessions through the PIX. Given the order of operations, NAT is performed before the AAA process is, therefore AAA should always be configured using local addresses.

Only TACACS+ is supported for classic user authorization. If RADIUS is required, the downloadable ACL feature should be used instead.



## Option #2: Activation of a Local ACL

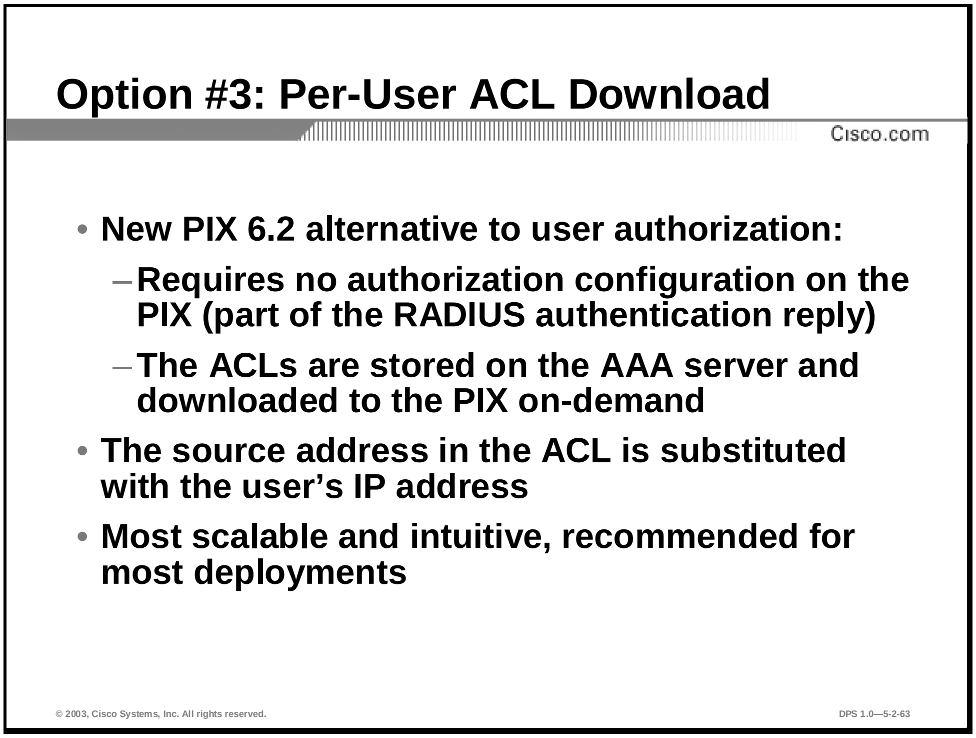
- An alternative to ACS-based authorization rules:
  - Requires no authorization configuration on the PIX (part of the authentication reply)
  - The AAA server returns a VSA (Attribute 11) to the PIX indicating the local ACL to be activated and prepended to the existing interface ACL
- The source address in the ACL is substituted with the user's IP address
- The ACL is removed when uauth entry expires
- Local ACLs have to be maintained on each PIX—possible scaling problem, not recommended

DPS 1.0—5-2-62

# Activation of a Local ACL

The second method of PIX user authorization is the activation of a local per-user ACL in addition to the existing ACL on the interface. It uses RADIUS to indicate which ACL should be activated, as part of RADIUS authentication. The RADIUS attribute #11 will indicate the name of the ACL that is to be used for the authenticated user. The ACL is modified based on the source IP of the authenticated user.

The benefit of such authentication is that it is scalable as long as a single firewall is used. If multiple firewalls perform user authentication, local ACLs need to be maintained on all of them, which is a management nightmare, if ACLs change frequently.



## Option #3: Per-User ACL Download

- New PIX 6.2 alternative to user authorization:
  - Requires no authorization configuration on the PIX (part of the RADIUS authentication reply)
  - The ACLs are stored on the AAA server and downloaded to the PIX on-demand
- The source address in the ACL is substituted with the user's IP address
- Most scalable and intuitive, recommended for most deployments

DPS 1.0—5-2-63

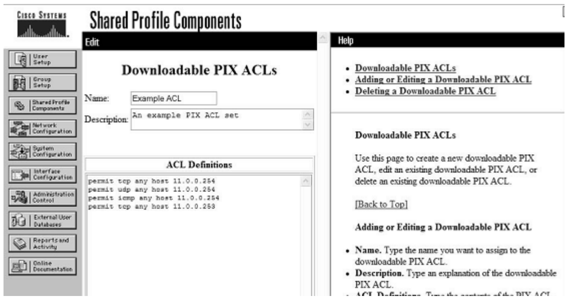
## Per-User ACL Download

PIX OS 6.2 introduced the ability to store full access-lists on the AAA server, and download them to the PIX as needed. The AAA server is configured with an "acl" attribute for a group or individual user. This method conserves storage space necessary for the PIX config. The user access-lists are only active until the uauth timer expires.

This functionality is only supported with RADIUS, and the ACL is returned as a part of the authentication phase. Therefore, only authentication needs to be configured on the PIX, and an ACL attached to the user or group profile on the AAA server. The source address in the ACL is substituted with the user's IP address when the ACL is installed on the PIX.

This is the most scalable and intuitive solution, as centralized user-friendly management is provided through the AAA server.

## Per-User ACL Download



Shared Profile Components

**Downloadable PIX ACLs**

Name: Example ACL

Description: An example PIX ACL set

**ACL Definitions**

```
permit tcp any host 11.0.0.254
permit udp any host 11.0.0.254
permit icmp any host 11.0.0.254
permit tcp any host 11.0.0.253
```

- Downloadable PIX ACLs
- Adding or Editing a Downloadable PIX ACL
- Deleting a Downloadable PIX ACL

**Downloadable PIX ACLs**

Use this page to create a new downloadable PIX ACL, edit an existing downloadable PIX ACL, or delete an existing downloadable PIX ACL.

[Back to Top]

**Adding or Editing a Downloadable PIX ACL**

- **Name.** Type the name you want to assign to the downloadable PIX ACL.
- **Description.** Type an explanation of the downloadable PIX ACL.
- **ACL Definitions.** Type the contents of the PIX ACL

**The ACL is configured on the AAA server, which can be used by many PIXen:**

- **Per-user (timestamped) ACLs are cached on the PIX for performance**
- **No significant ACL length restriction**

DPS 1.0—5-2-64

This figure shows the configuration of the ACL on the AAA server. The “Shared profile components” are used to store the ACL, and attach it to the user or group profile. The PIX will cache the ACL for a while to improve performance, if many users with the same profile authenticate to the PIX in short time intervals.

# Practice

Q1) Where are the authorization rules stored when using classic PIX cut-thru proxy authorization?

A) on the PIX, in a special AAA database

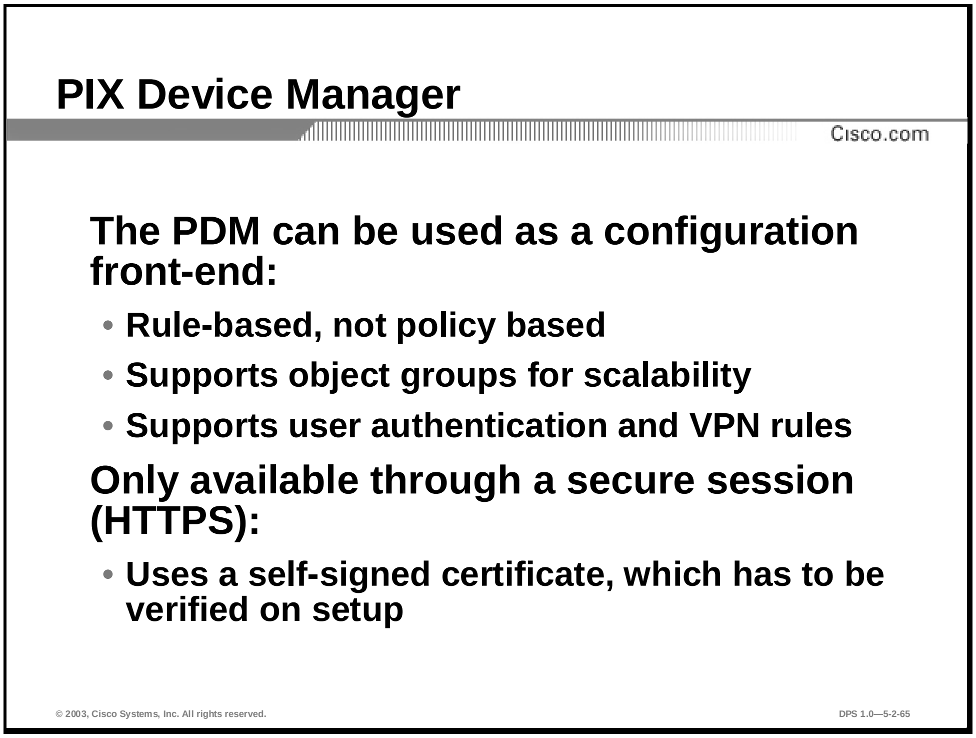
B) the authorization rules are the PIX local ACLs

C) on the AAA server, as access lists

D) on the AAA server, as authorized IOS commands

E) on the PIX, as authorized IOS commands

# PIX Device Manager



## PIX Device Manager

**The PDM can be used as a configuration front-end:**

- **Rule-based, not policy based**
- **Supports object groups for scalability**
- **Supports user authentication and VPN rules**

**Only available through a secure session (HTTPS):**

- **Uses a self-signed certificate, which has to be verified on setup**

DPS 1.0—5-2-65

## Objective

This lesson will enable the learner to identify the PIX device manager as a configuration tool for a single PIX Firewall system.

## Introduction

To configure the PIX Firewall graphically, the simplest method is to use the built-in Java-based PIX Device Manager (PDM). The PDM is accessed through HTTPS (using SSL for protection) access to the PIX, and provides full configuration capability for almost all PIX Features. From version 2.0, scalable deployments are possible, as access rules can be constructed using object groups.

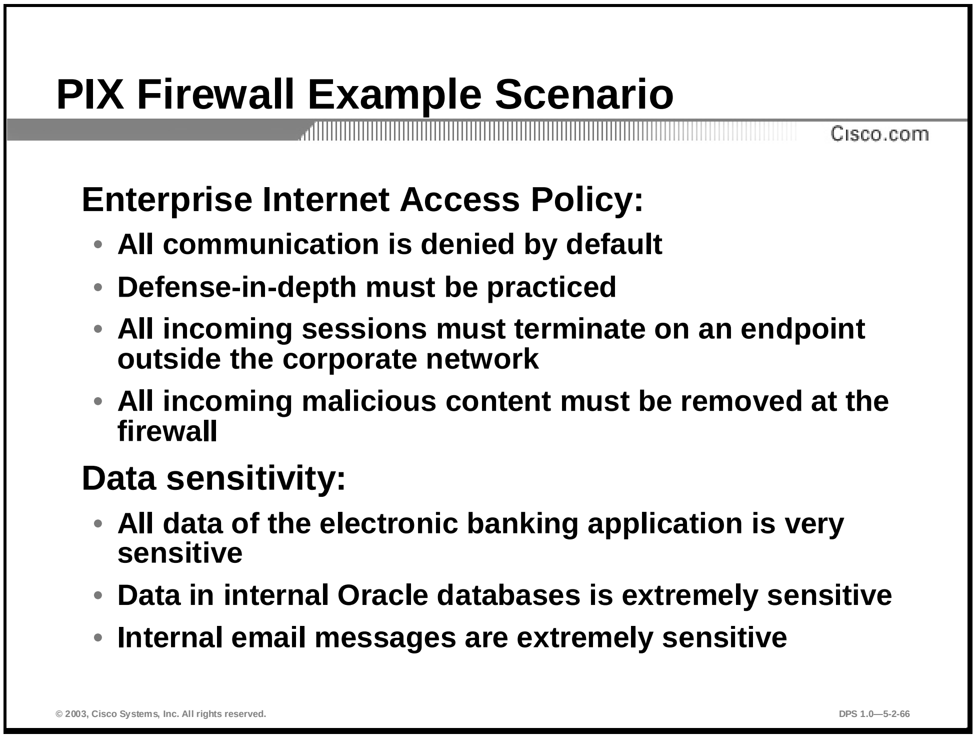
## Practice

Q1) The PIX Device Manager supports object groups. True or false?

A) true

B) false

# PIX Deployment Example Scenario



## PIX Firewall Example Scenario

Cisco.com

Enterprise Internet Access Policy:

- All communication is denied by default
- Defense-in-depth must be practiced
- All incoming sessions must terminate on an endpoint outside the corporate network
- All incoming malicious content must be removed at the firewall

Data sensitivity:

- All data of the electronic banking application is very sensitive
- Data in internal Oracle databases is extremely sensitive
- Internal email messages are extremely sensitive

DPS 1.0—5-2-66

## Objectives

This section will enable the learner to identify common PIX Firewall deployment scenarios to recognize them in firewall design.

## Introduction

This example scenario will present a complex PIX access control configuration, which will be constructed according to best practices of PIX configuration.

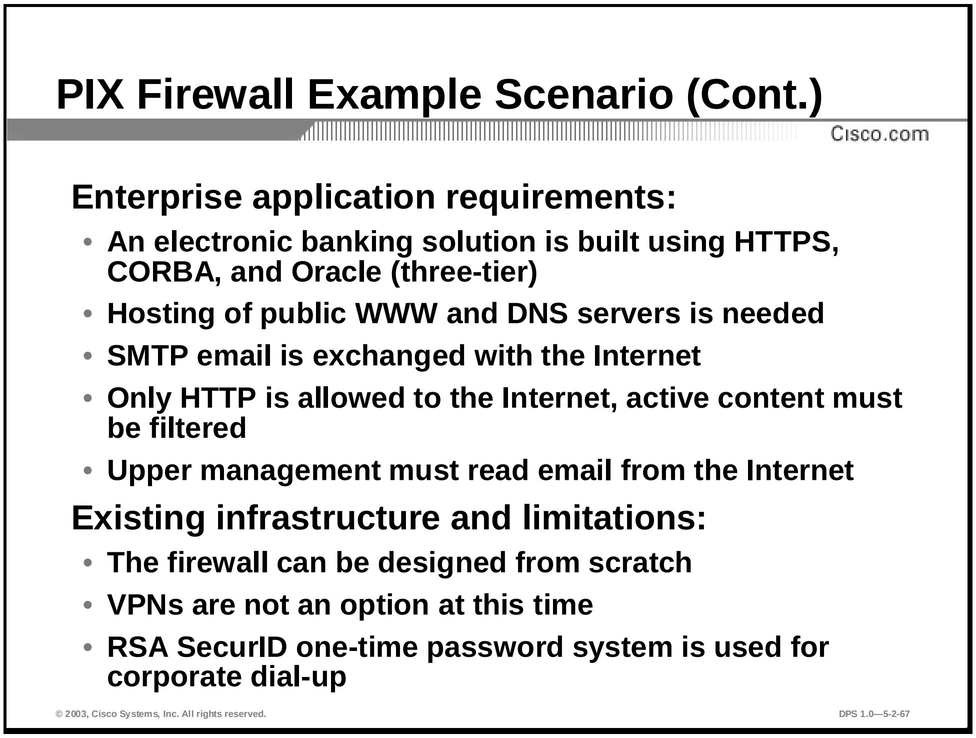
## Policy Description

An enterprise Internet firewall needs to be configured according to the following security policy:

- All communication is denied by default
- Defense-in-depth must be practiced
- All incoming sessions must terminate on an endpoint outside the corporate network
- All incoming malicious content must be removed at the firewall

The sensitivity of data relayed by the firewall is defined as follows:

- All data of the electronic banking application is very sensitive
- Data in internal Oracle databases is extremely sensitive
- Internal email messages are extremely sensitive



## PIX Firewall Example Scenario (Cont.)

Cisco.com

**Enterprise application requirements:**

- **An electronic banking solution is built using HTTPS, CORBA, and Oracle (three-tier)**
- **Hosting of public WWW and DNS servers is needed**
- **SMTP email is exchanged with the Internet**
- **Only HTTP is allowed to the Internet, active content must be filtered**
- **Upper management must read email from the Internet**

**Existing infrastructure and limitations:**

- **The firewall can be designed from scratch**
- **VPNs are not an option at this time**
- **RSA SecurID one-time password system is used for corporate dial-up**

DPS 1.0—5-2-67

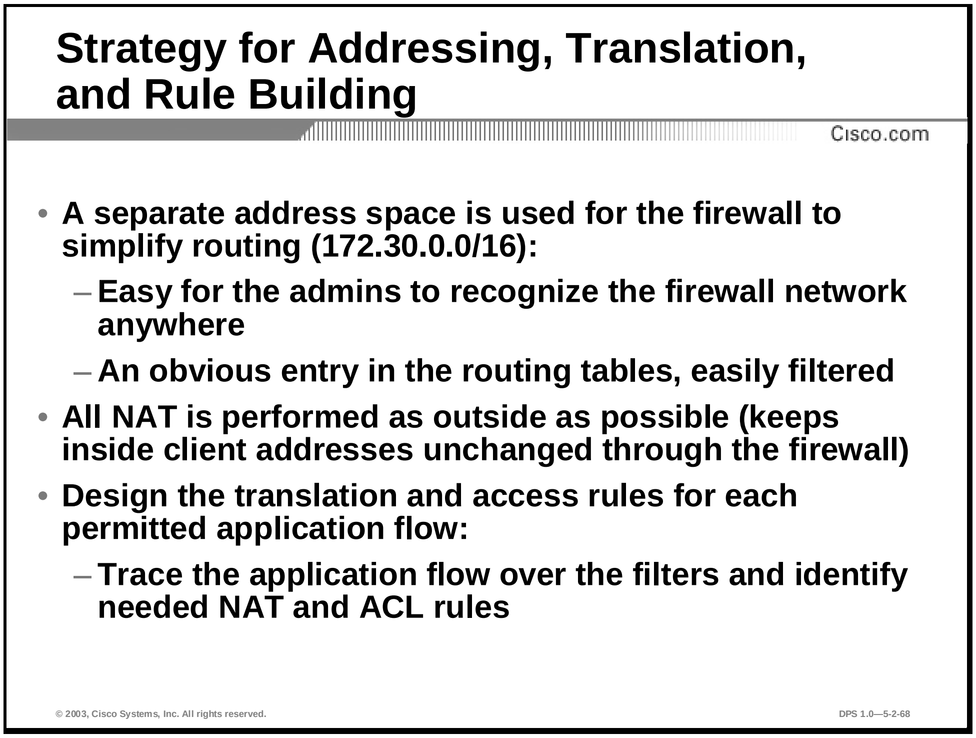
# Application and Protocol Requirements

The application and protocol requirements in the firewall are the following:

- An electronic banking solution is built using HTTPS, CORBA, and Oracle (three-tier)
- Hosting of public WWW and DNS servers is needed
- SMTP email is exchanged with the Internet
- Only HTTP is allowed to the Internet, active content must be filtered
- Upper management must read email from the Internet

The limitations for the design are:

- The firewall can be designed from scratch
- VPNs are not an option at this time
- RSA SecurID one-time password system is used for corporate dial-up



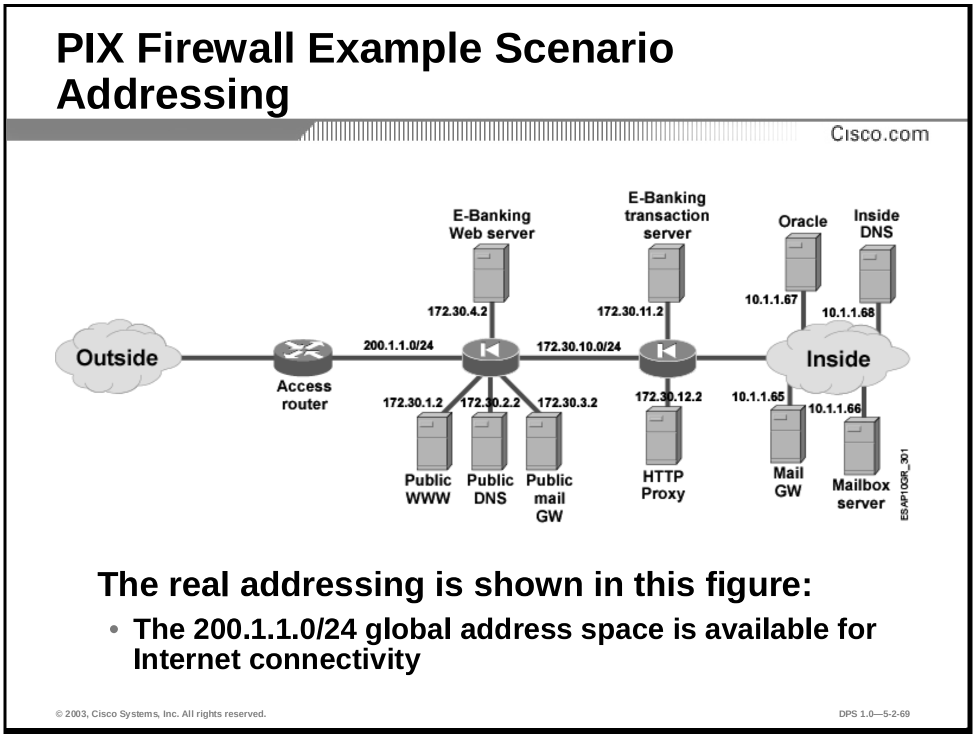
## Strategy for Addressing, Translation, and Rule Building

- A separate address space is used for the firewall to simplify routing (172.30.0.0/16):
  - Easy for the admins to recognize the firewall network anywhere
  - An obvious entry in the routing tables, easily filtered
- All NAT is performed as outside as possible (keeps inside client addresses unchanged through the firewall)
- Design the translation and access rules for each permitted application flow:
  - Trace the application flow over the filters and identify needed NAT and ACL rules

DPS 1.0—5-2-68

## Addressing Strategy

The strategy for addressing was to use a separate network (172.30.0.0/16) for the firewall, which makes it easy to identify traffic terminating inside it or leaving the firewall hosts. NAT is performed as outside as possible, to keep IP addresses of inside clients unchanged as long as possible. This improves visibility into traffic flows and makes rule design more intuitive.



# PIX Firewall Example Scenario Addressing

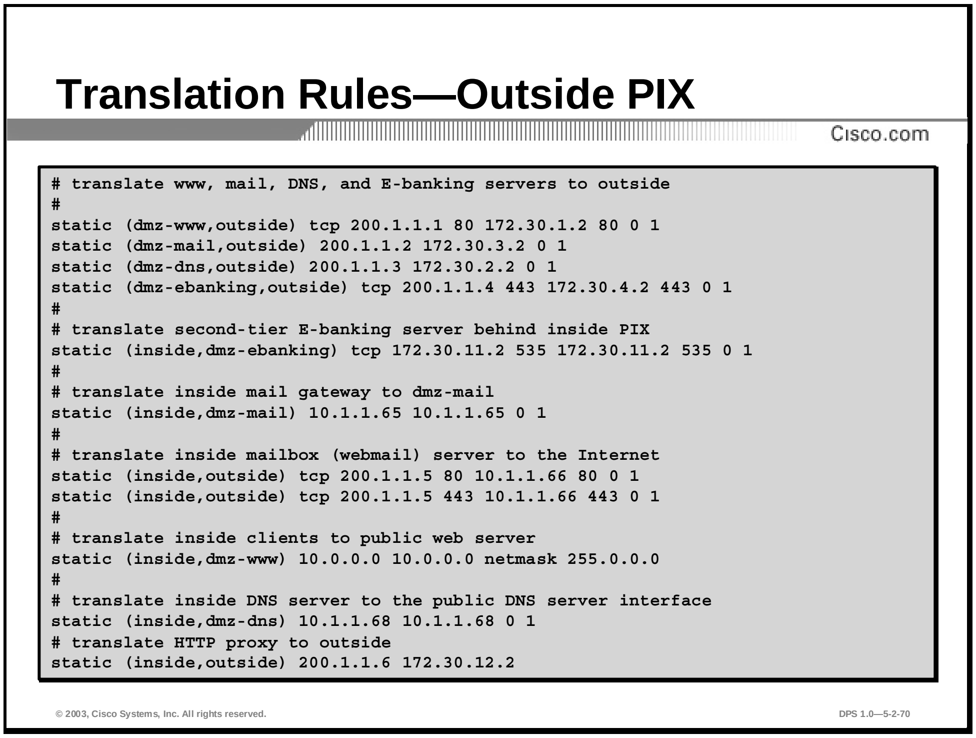
Cisco.com

**The real addressing is shown in this figure:**

- **The 200.1.1.0/24 global address space is available for Internet connectivity**

DPS 1.0—5-2-69

This figure shows the result of the addressing strategy, applied to the firewall subnets and hosts.



## Translation Rules—Outside PIX

Cisco.com

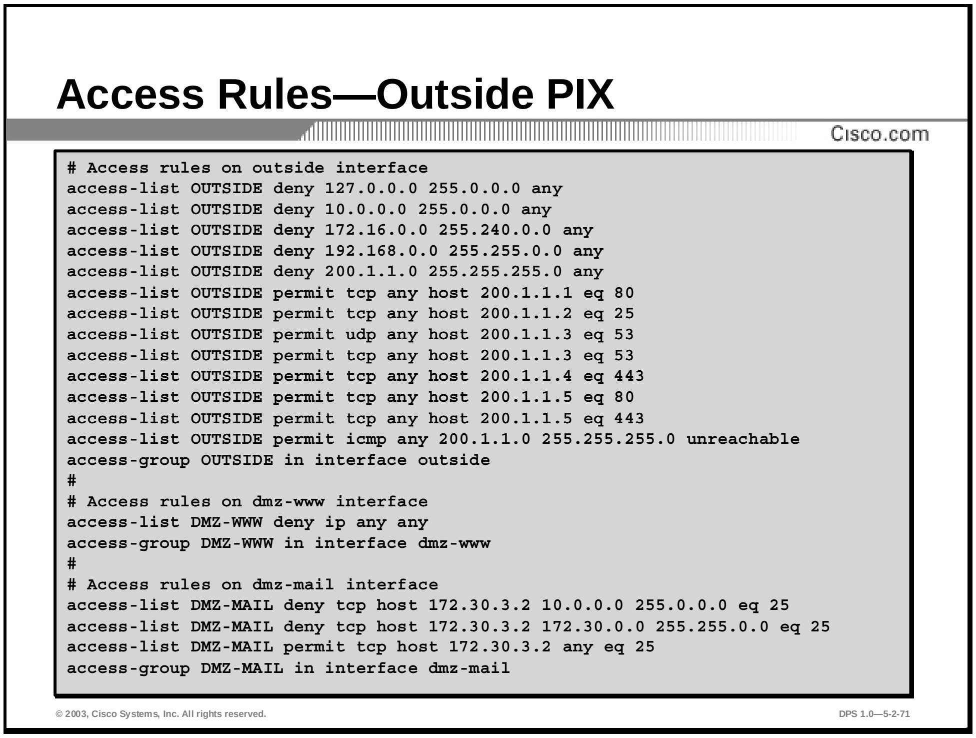
```
# translate www, mail, DNS, and E-banking servers to outside
#
static (dmz-www,outside) tcp 200.1.1.1 80 172.30.1.2 80 0 1
static (dmz-mail,outside) 200.1.1.2 172.30.3.2 0 1
static (dmz-dns,outside) 200.1.1.3 172.30.2.2 0 1
static (dmz-ebanking,outside) tcp 200.1.1.4 443 172.30.4.2 443 0 1
#
# translate second-tier E-banking server behind inside PIX
static (inside,dmz-ebanking) tcp 172.30.11.2 535 172.30.11.2 535 0 1
#
# translate inside mail gateway to dmz-mail
static (inside,dmz-mail) 10.1.1.65 10.1.1.65 0 1
#
# translate inside mailbox (webmail) server to the Internet
static (inside,outside) tcp 200.1.1.5 80 10.1.1.66 80 0 1
static (inside,outside) tcp 200.1.1.5 443 10.1.1.66 443 0 1
#
# translate inside clients to public web server
static (inside,dmz-www) 10.0.0.0 10.0.0.0 netmask 255.0.0.0
#
# translate inside DNS server to the public DNS server interface
static (inside,dmz-dns) 10.1.1.68 10.1.1.68 0 1
# translate HTTP proxy to outside
static (inside,outside) 200.1.1.6 172.30.12.2
```

DPS 1.0—5-2-70

The sample configuration is provided as an example of how to configure the translation rules on the outside PIX. All translation to Internet addresses is performed here. Inside access to DMZ networks is identity-translated to keep addresses unchanged.

In the example, port static rules (mapping only a single global port to a local port of the protected host) are used instead of classic static rules to minimize possible connectivity even further, especially in the event of ACL misconfiguration.

The configuration lines are broken out with commentary to explain specific command sets and their implied actions.

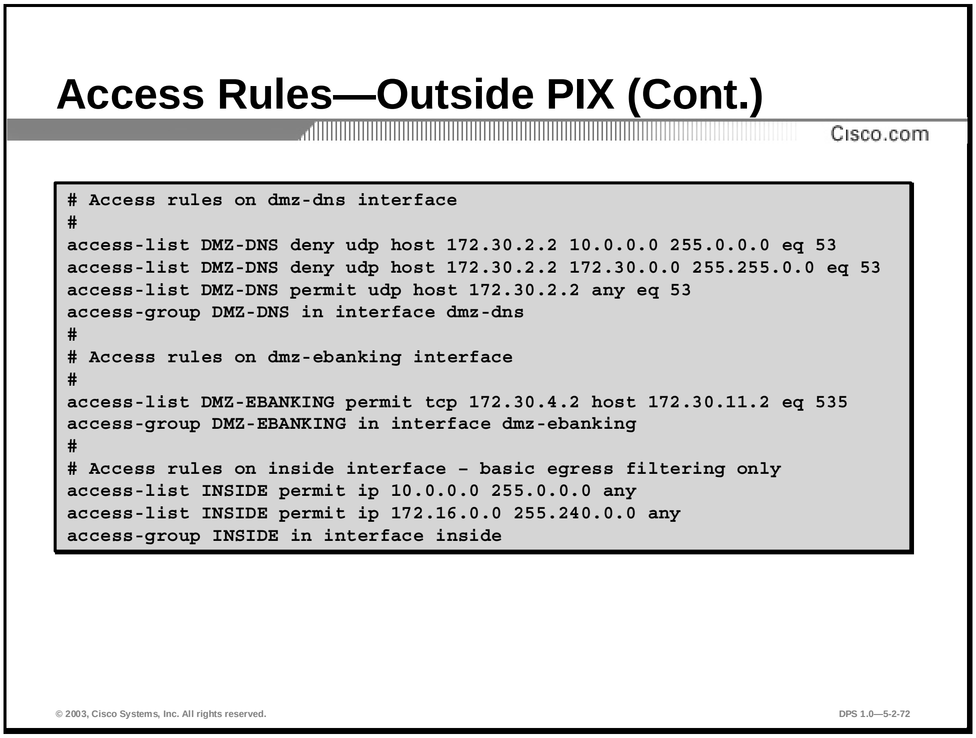


## Access Rules—Outside PIX

```
# Access rules on outside interface
access-list OUTSIDE deny 127.0.0.0 255.0.0.0 any
access-list OUTSIDE deny 10.0.0.0 255.0.0.0 any
access-list OUTSIDE deny 172.16.0.0 255.240.0.0 any
access-list OUTSIDE deny 192.168.0.0 255.255.0.0 any
access-list OUTSIDE deny 200.1.1.0 255.255.255.0 any
access-list OUTSIDE permit tcp any host 200.1.1.1 eq 80
access-list OUTSIDE permit tcp any host 200.1.1.2 eq 25
access-list OUTSIDE permit udp any host 200.1.1.3 eq 53
access-list OUTSIDE permit tcp any host 200.1.1.3 eq 53
access-list OUTSIDE permit tcp any host 200.1.1.4 eq 443
access-list OUTSIDE permit tcp any host 200.1.1.5 eq 80
access-list OUTSIDE permit tcp any host 200.1.1.5 eq 443
access-list OUTSIDE permit icmp any 200.1.1.0 255.255.255.0 unreachable
access-group OUTSIDE in interface outside
#
# Access rules on dmz-www interface
access-list DMZ-WWW deny ip any any
access-group DMZ-WWW in interface dmz-www
#
# Access rules on dmz-mail interface
access-list DMZ-MAIL deny tcp host 172.30.3.2 10.0.0.0 255.0.0.0 eq 25
access-list DMZ-MAIL deny tcp host 172.30.3.2 172.30.0.0 255.255.0.0 eq 25
access-list DMZ-MAIL permit tcp host 172.30.3.2 any eq 25
access-group DMZ-MAIL in interface dmz-mail
```

DPS 1.0—5-2-71

The sample configuration is provided as an example of how to configure the access rules on the outside PIX. Note the least privilege enforcement for inbound connections – the access rules are extremely granular and permit exactly the needed services.



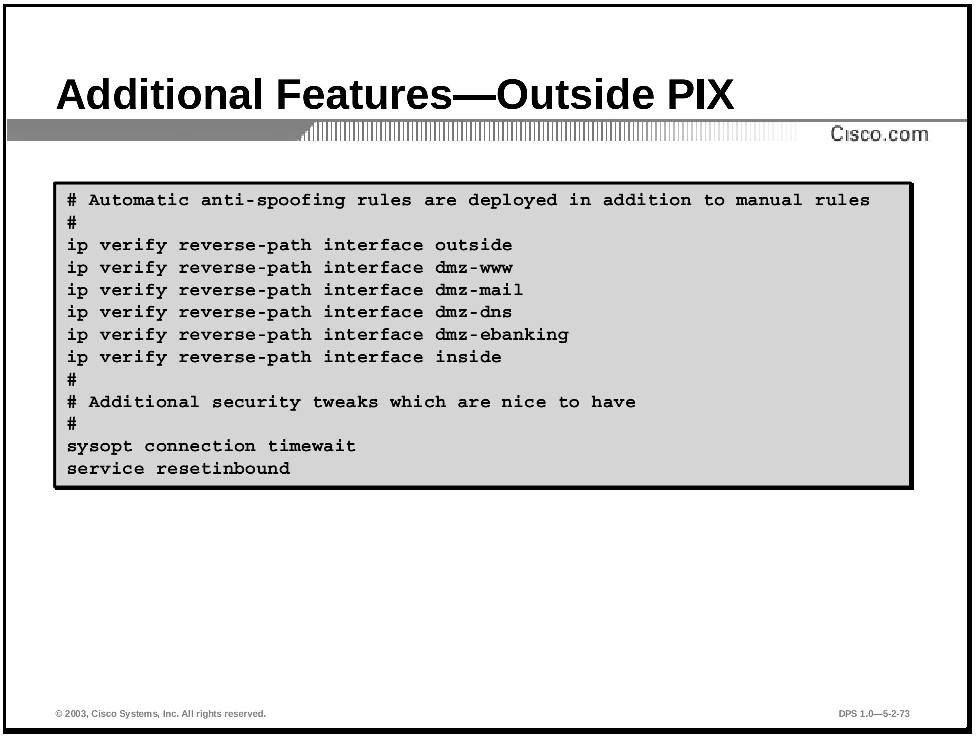
## Access Rules—Outside PIX (Cont.)

Cisco.com

```
# Access rules on dmz-dns interface
#
access-list DMZ-DNS deny udp host 172.30.2.2 10.0.0.0 255.0.0.0 eq 53
access-list DMZ-DNS deny udp host 172.30.2.2 172.30.0.0 255.255.0.0 eq 53
access-list DMZ-DNS permit udp host 172.30.2.2 any eq 53
access-group DMZ-DNS in interface dmz-dns
#
# Access rules on dmz-ebanking interface
#
access-list DMZ-EBANKING permit tcp 172.30.4.2 host 172.30.11.2 eq 535
access-group DMZ-EBANKING in interface dmz-ebanking
#
# Access rules on inside interface – basic egress filtering only
access-list INSIDE permit ip 10.0.0.0 255.0.0.0 any
access-list INSIDE permit ip 172.16.0.0 255.240.0.0 any
access-group INSIDE in interface inside
```

DPS 1.0—5-2-72

The configuration lines are broken out with commentary to explain specific command sets and their implied actions. Here, the connections from the outer DMZs and from the inside perimeter are controlled.



## Additional Features—Outside PIX

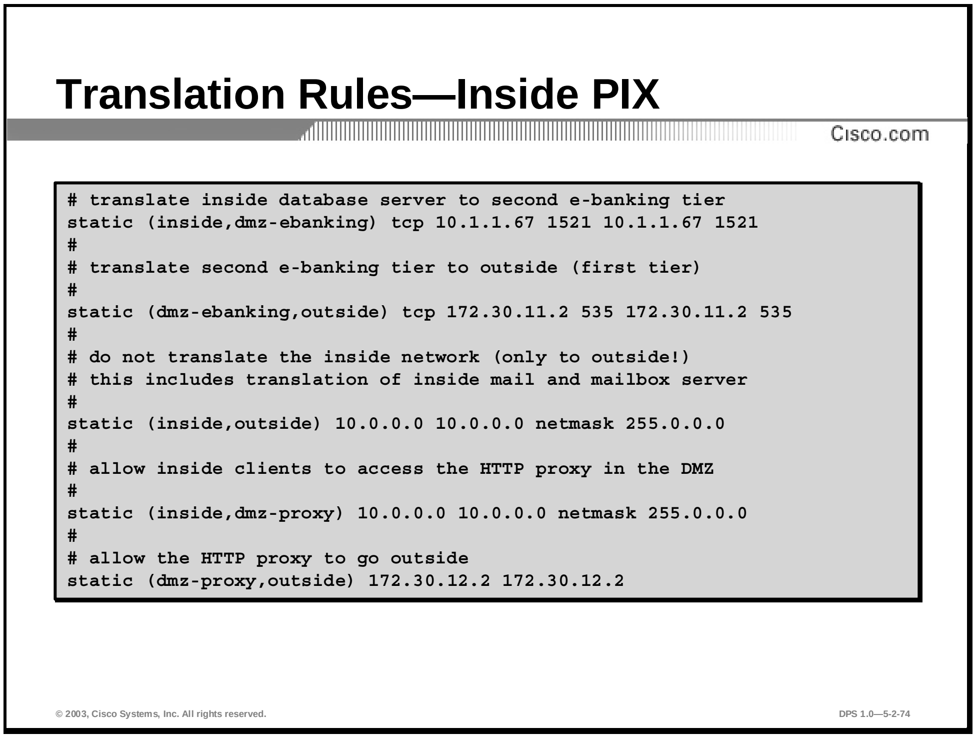
Cisco.com

```
# Automatic anti-spoofing rules are deployed in addition to manual rules
#
ip verify reverse-path interface outside
ip verify reverse-path interface dmz-www
ip verify reverse-path interface dmz-mail
ip verify reverse-path interface dmz-dns
ip verify reverse-path interface dmz-ebanking
ip verify reverse-path interface inside
#
# Additional security tweaks which are nice to have
#
sysopt connection timewait
service resetinbound
```

DPS 1.0—5-2-73

This figure shows the additional features configured on the outside PIX. Anti-spoofing is performed using unicast RPF. The “sysopt connection safeclose” command improves handling of connection close retransmissions, and the “service resetinbound” resets incoming TCP sessions, denied by the outside PIX rules.

The configuration lines are broken out with commentary to explain specific command sets and their implied actions.



## Translation Rules—Inside PIX

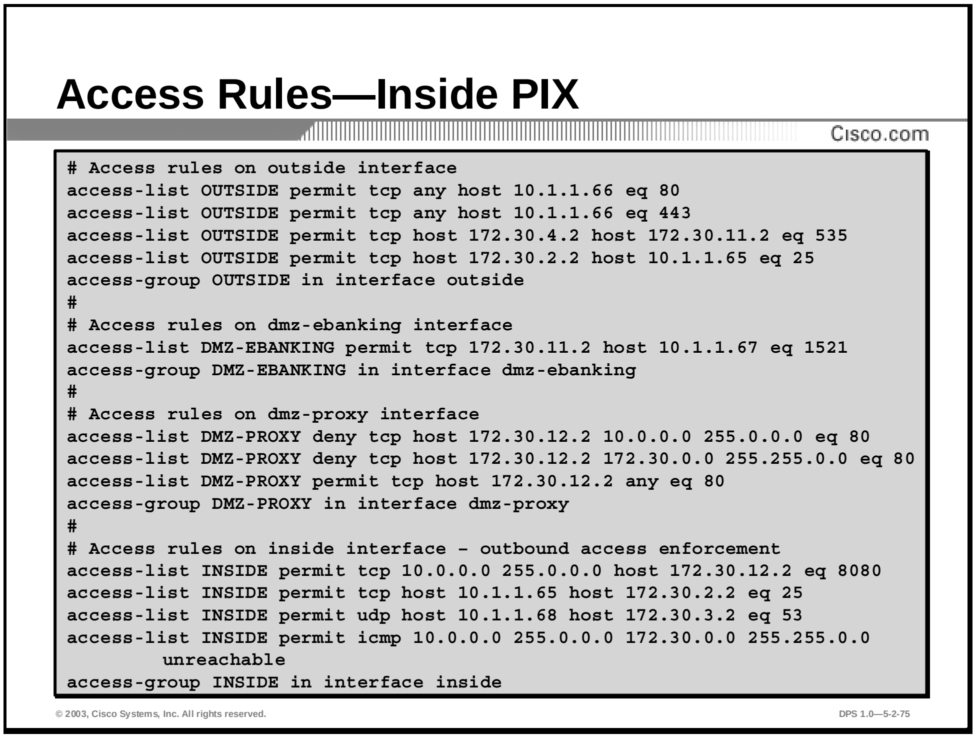
Cisco.com

```
# translate inside database server to second e-banking tier
static (inside,dmz-ebanking) tcp 10.1.1.67 1521 10.1.1.67 1521
#
# translate second e-banking tier to outside (first tier)
#
static (dmz-ebanking,outside) tcp 172.30.11.2 535 172.30.11.2 535
#
# do not translate the inside network (only to outside!)
# this includes translation of inside mail and mailbox server
#
static (inside,outside) 10.0.0.0 10.0.0.0 netmask 255.0.0.0
#
# allow inside clients to access the HTTP proxy in the DMZ
#
static (inside,dmz-proxy) 10.0.0.0 10.0.0.0 netmask 255.0.0.0
#
# allow the HTTP proxy to go outside
static (dmz-proxy,outside) 172.30.12.2 172.30.12.2
```

DPS 1.0—5-2-74

The inside PIX performs translation only to enable connectivity across the PIX – that is, only identity translation rules are configured. All other translation is performed by the outside PIX.

The configuration lines are broken out with commentary to explain specific command sets and their implied actions.

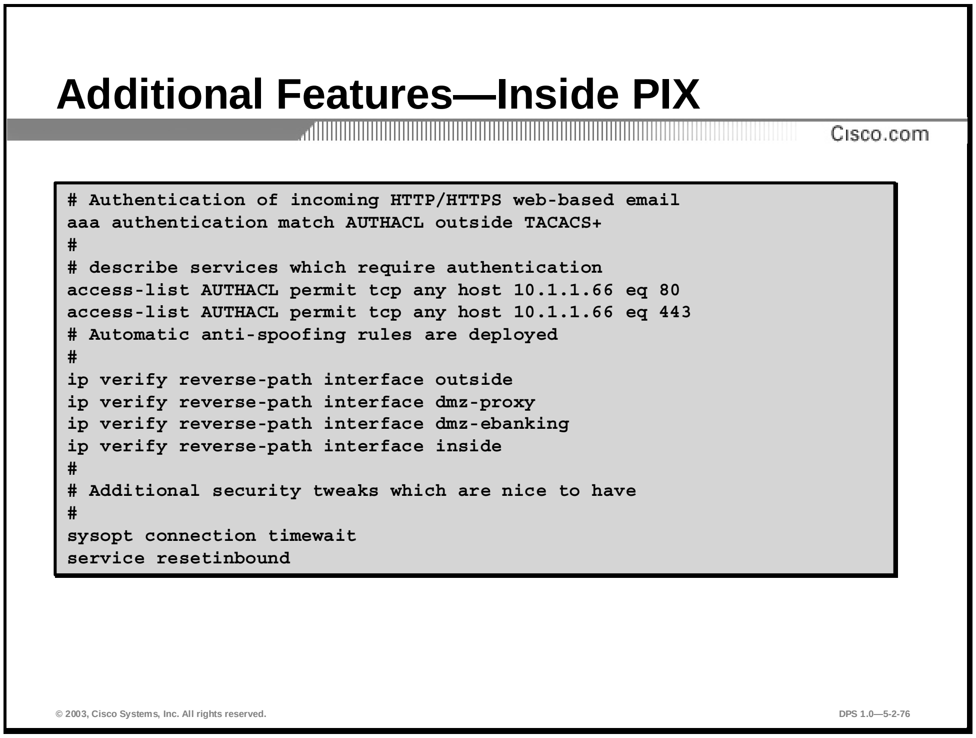


## Access Rules—Inside PIX

```
# Access rules on outside interface
access-list OUTSIDE permit tcp any host 10.1.1.66 eq 80
access-list OUTSIDE permit tcp any host 10.1.1.66 eq 443
access-list OUTSIDE permit tcp host 172.30.4.2 host 172.30.11.2 eq 535
access-list OUTSIDE permit tcp host 172.30.2.2 host 10.1.1.65 eq 25
access-group OUTSIDE in interface outside
#
# Access rules on dmz-ebanking interface
access-list DMZ-EBANKING permit tcp 172.30.11.2 host 10.1.1.67 eq 1521
access-group DMZ-EBANKING in interface dmz-ebanking
#
# Access rules on dmz-proxy interface
access-list DMZ-PROXY deny tcp host 172.30.12.2 10.0.0.0 255.0.0.0 eq 80
access-list DMZ-PROXY deny tcp host 172.30.12.2 172.30.0.0 255.255.0.0 eq 80
access-list DMZ-PROXY permit tcp host 172.30.12.2 any eq 80
access-group DMZ-PROXY in interface dmz-proxy
#
# Access rules on inside interface - outbound access enforcement
access-list INSIDE permit tcp 10.0.0.0 255.0.0.0 host 172.30.12.2 eq 8080
access-list INSIDE permit tcp host 10.1.1.65 host 172.30.2.2 eq 25
access-list INSIDE permit udp host 10.1.1.68 host 172.30.3.2 eq 53
access-list INSIDE permit icmp 10.0.0.0 255.0.0.0 172.30.0.0 255.255.0.0
        unreachable
access-group INSIDE in interface inside
```

The access rules on the inside PIX provide access to the internal network for specific services – inbound mail delivery, the E-commerce application (CORBA/SQL*net over a DMZ), and inbound web-mail connections for roaming users.

The configuration lines are broken out with commentary to explain specific command sets and their implied actions.



## Additional Features—Inside PIX

```
# Authentication of incoming HTTP/HTTPS web-based email
aaa authentication match AUTHACL outside TACACS+
#
# describe services which require authentication
access-list AUTHACL permit tcp any host 10.1.1.66 eq 80
access-list AUTHACL permit tcp any host 10.1.1.66 eq 443
# Automatic anti-spoofing rules are deployed
#
ip verify reverse-path interface outside
ip verify reverse-path interface dmz-proxy
ip verify reverse-path interface dmz-ebanking
ip verify reverse-path interface inside
#
# Additional security tweaks which are nice to have
#
sysopt connection timewait
service resetinbound
```

DPS 1.0—5-2-76

Note the authentication needed to access internal web-based email from outside. The application is designed to accept a user’s web session, and then redirects him to HTTPS to read email.

The configuration lines are broken out with commentary to explain specific command sets and their implied actions.

## Practice

Q1) In this example, why was NAT performed on the outside firewall?

A) to keep the client IP addresses unchanged as long as possible

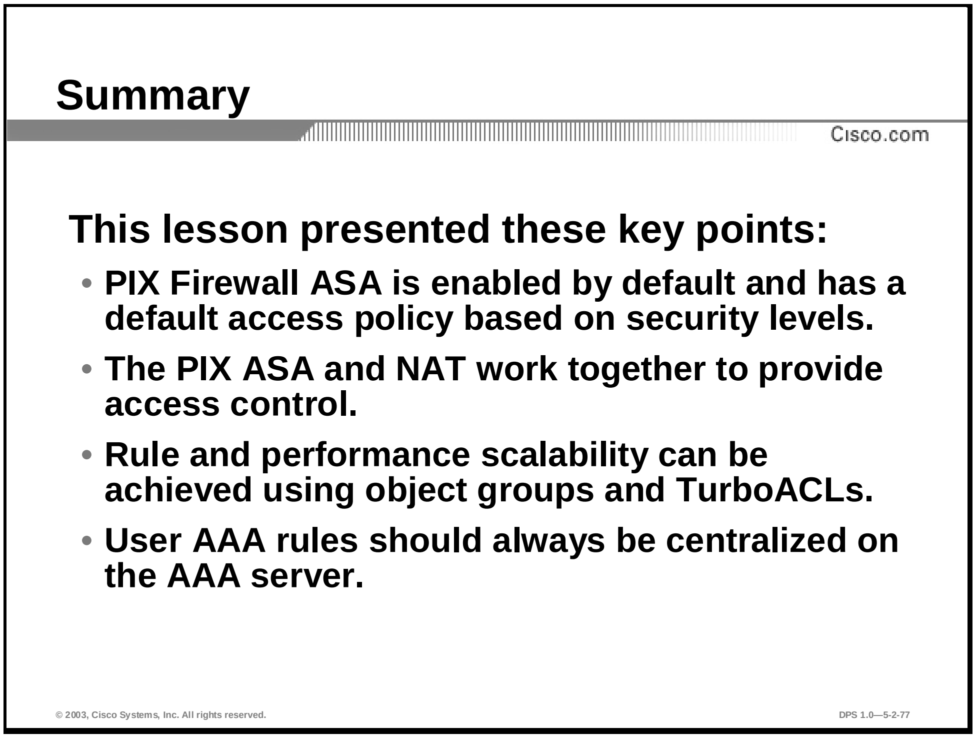
B) for performance reasons

C) for routing reasons

D) to simplify configuration

# Summary

This section summarizes the key points discussed in this lesson.



## Summary

This lesson presented these key points:

- PIX Firewall ASA is enabled by default and has a default access policy based on security levels.
- The PIX ASA and NAT work together to provide access control.
- Rule and performance scalability can be achieved using object groups and TurboACLs.
- User AAA rules should always be centralized on the AAA server.

DPS 1.0—5-2-77

## Next Steps

After completing this lesson, go to:

- Cisco IOS Software Access Control Features lesson

## References

For additional information, refer to these resources:

- An in-depth analysis of packet flow through a PIX can be found at http://www-tac.cisco.com/Support_Library/Hardware/PIX/pix51flow.htm

# Quiz: Understanding PIX Firewall Security

Complete the quiz to assess what you have learned in this lesson.

## Objectives

This quiz tests your knowledge on how to:

- Identify and configure advanced Adaptive Security Algorithm (ASA) features
- Identify ASA limitations of the Cisco Secure PIX Firewall product when using it in a firewall system design

## Instructions

Answer these questions:

1. What is the default ASA security policy?
2. Which TCP packet parameters does the ASA check?
3. Which security issues should be considered with PIX Firewall user authentication?
4. What are the three user authorization options on the PIX Firewall?
5. What denial-of-service protection does the PIX Firewall offer and when should it be enabled?

## Scoring

You have successfully completed the quiz for this lesson when you earn a score of 80 percent or better.

# Cisco IOS Software Access Control Features

## Overview

### Importance

As the Cisco IOS is one of the most widespread network-node operating system in the world it has also great importance on perimeter systems where security demands are critical. In order to create the most efficient design this lesson presents the bolts and nuts of the access control principles implemented in IOS.

### Lesson Objective

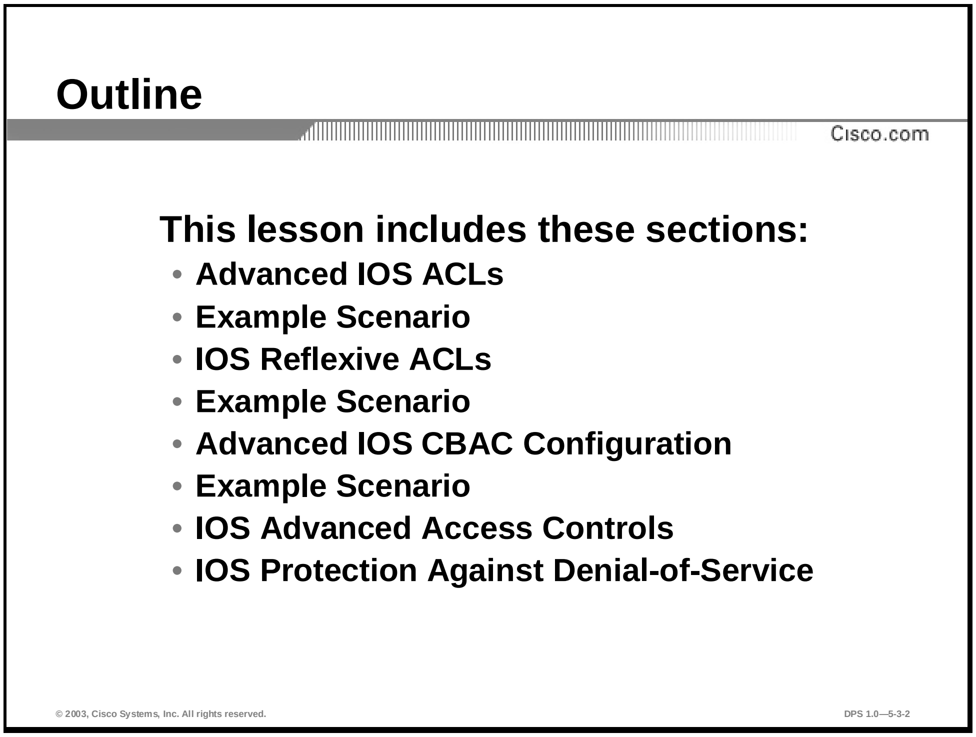
This lesson will enable the learner to identify advanced security features and limitations of the Cisco IOS software, when using it in a firewall system design.

## Learner Skills and Knowledge

To fully benefit from this lesson, you must have these prerequisite skills and knowledge:

- Have a solid understanding of the basic access control features used in Cisco IOS software

## Outline



### Outline

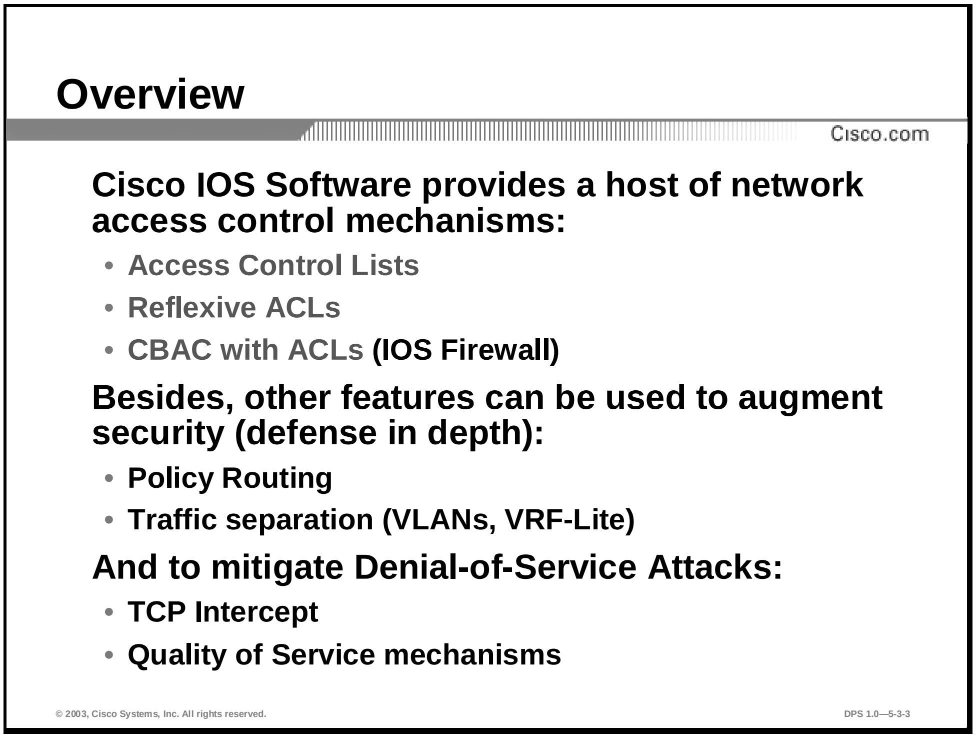
Cisco.com

**This lesson includes these sections:**

- **Advanced IOS ACLs**
- **Example Scenario**
- **IOS Reflexive ACLs**
- **Example Scenario**
- **Advanced IOS CBAC Configuration**
- **Example Scenario**
- **IOS Advanced Access Controls**
- **IOS Protection Against Denial-of-Service**

DPS 1.0—5-3-2

# Overview



## Overview

Cisco.com

**Cisco IOS Software provides a host of network access control mechanisms:**

- **Access Control Lists**
- **Reflexive ACLs**
- **CBAC with ACLs (IOS Firewall)**

**Besides, other features can be used to augment security (defense in depth):**

- **Policy Routing**
- **Traffic separation (VLANs, VRF-Lite)**

**And to mitigate Denial-of-Service Attacks:**

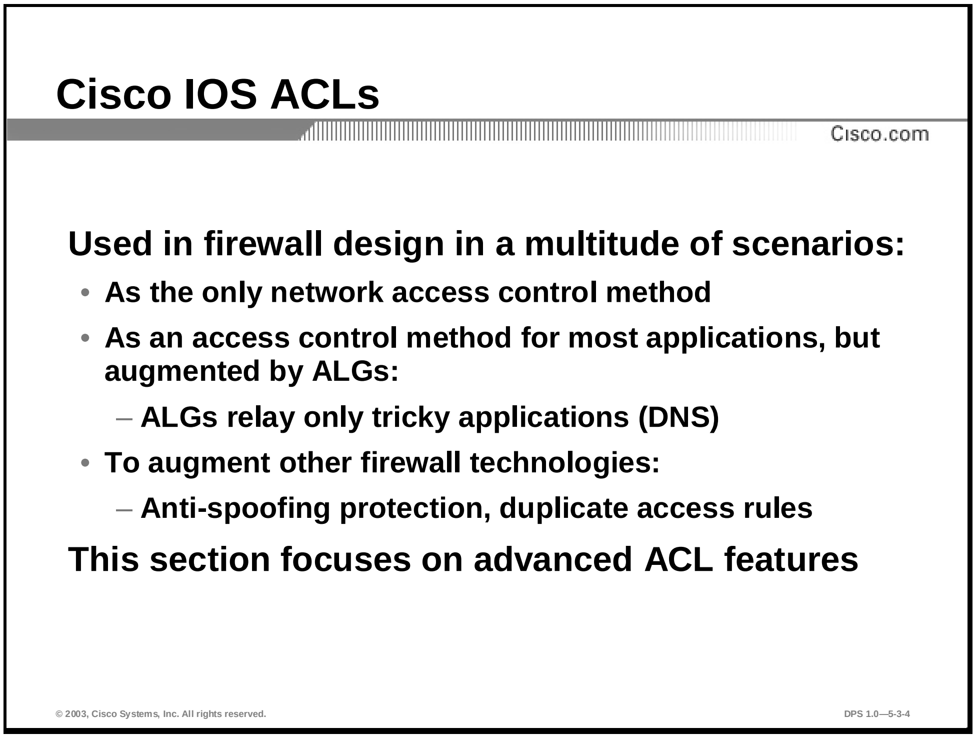
- **TCP Intercept**
- **Quality of Service mechanisms**

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—5-3-3

Access control is fundamental to Internet firewalling and performance improvements. This lesson provides advanced guidelines on mainstream technologies (such as ACLs, reflexive ACLs, and CBAC), and advanced access control features, such as policy routing, or VRF-Lite, available in Cisco IOS software. Methods to mitigate denial-of-service attacks are also investigated, together with configuration guidelines for specific attacks.

# Advanced IOS ACLs



## Cisco IOS ACLs

Cisco.com

**Used in firewall design in a multitude of scenarios:**

- **As the only network access control method**
- **As an access control method for most applications, but augmented by ALGs:**
  - **ALGs relay only tricky applications (DNS)**
- **To augment other firewall technologies:**
  - **Anti-spoofing protection, duplicate access rules**

**This section focuses on advanced ACL features**

DPS 1.0—5-3-4

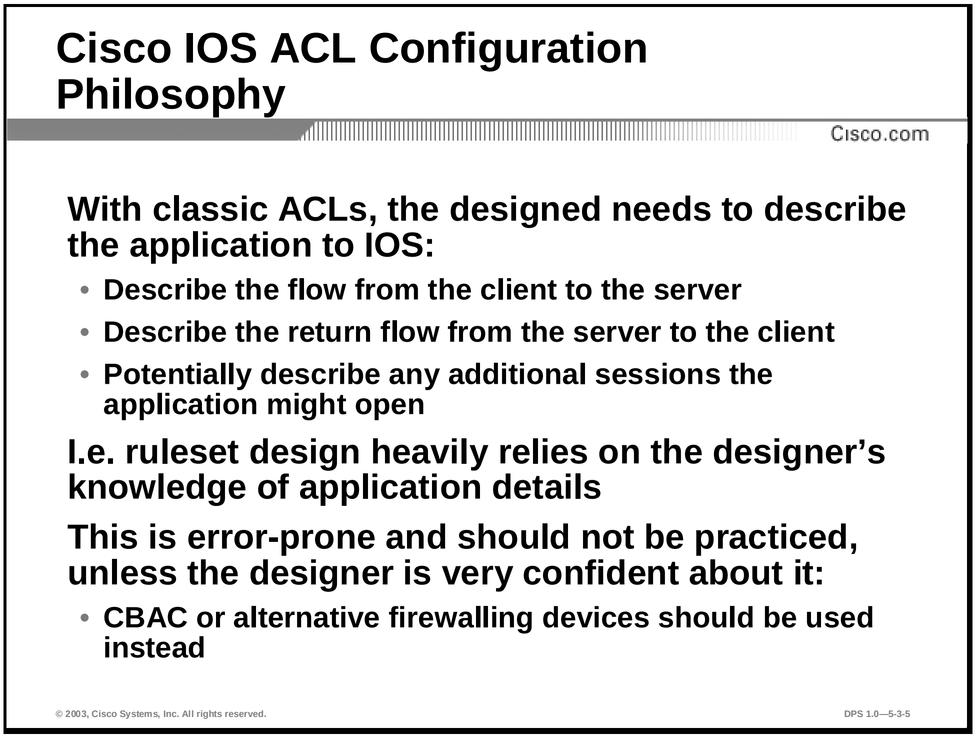
## Objective

This section will enable the learner to identify and configure the advanced aspects of ACLs in Cisco IOS software and choose them in a firewall design.

## Introduction

The Access List (ACL) is the basic component to define traffic patterns for which security rules, policies, and quality of service can be configured. This section explains enhanced ACL issues necessary for an efficient design.

ACLs only match on layer 3 and layer 4 patterns therefore another mechanism is required for more enhanced capabilities.



## Cisco IOS ACL Configuration Philosophy

Cisco.com

With classic ACLs, the designed needs to describe the application to IOS:

- Describe the flow from the client to the server
- Describe the return flow from the server to the client
- Potentially describe any additional sessions the application might open

I.e. ruleset design heavily relies on the designer’s knowledge of application details

This is error-prone and should not be practiced, unless the designer is very confident about it:

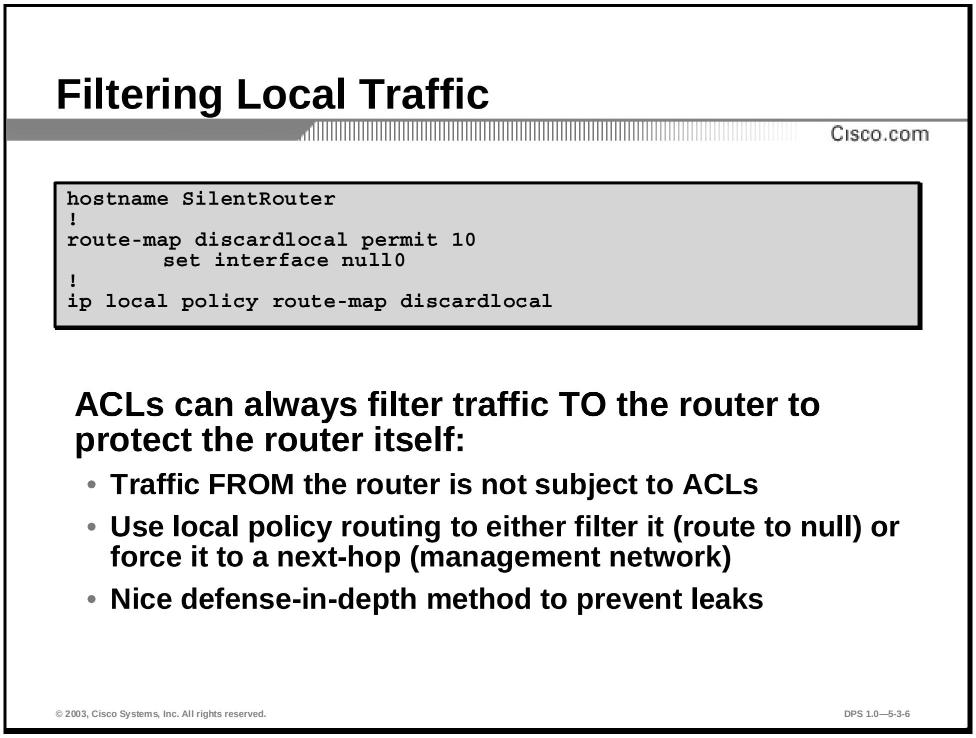
- CBAC or alternative firewalling devices should be used instead

DPS 1.0—5-3-5

# ACL Philosophy

Using classic ACLs the designer must know about protocol header details of various traffic types.

This approach is error-prone and not suitable for a bulletproof security system—unless the designer is very confident with protocol detail. Modern Cisco IOS provides a more application-aware approach, such as Context-Based Access Control (CBAC).



## Filtering Local Traffic

```
hostname SilentRouter
!
route-map discardlocal permit 10
       set interface null0
!
ip local policy route-map discardlocal
```

**ACLs can always filter traffic TO the router to protect the router itself:**

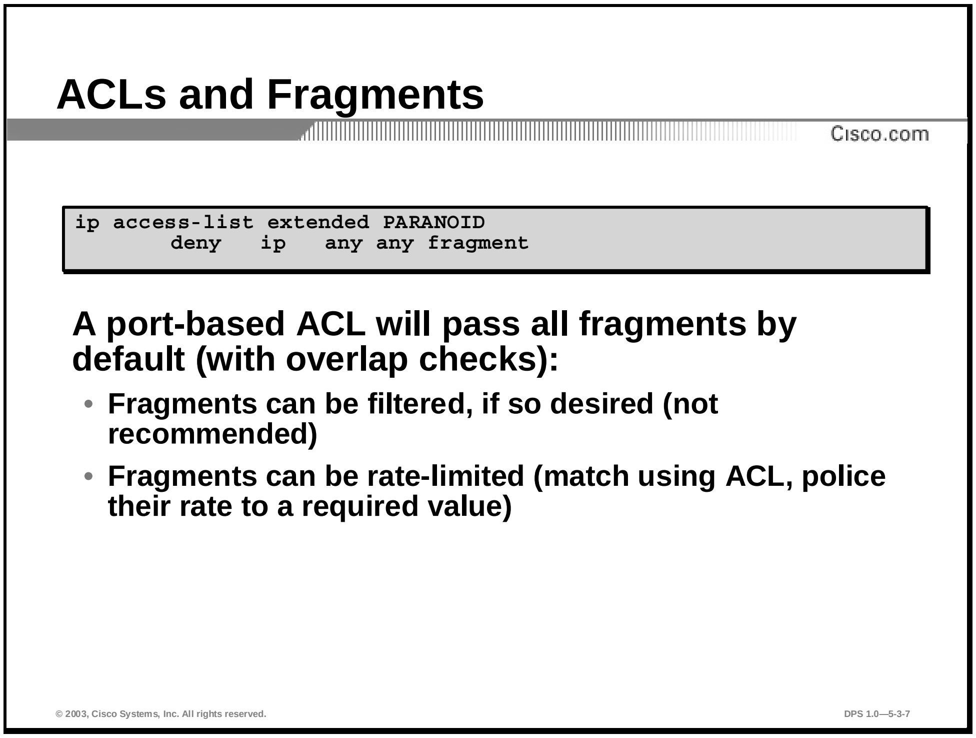
- **Traffic FROM the router is not subject to ACLs**
- **Use local policy routing to either filter it (route to null) or force it to a next-hop (management network)**
- **Nice defense-in-depth method to prevent leaks**

# Local Traffic Issues

ACLs are used to filter traffic to the router in order to protect the router from incoming traffic. However ACLs cannot filter packets that are originated by the router.

One solution is to configure a local routing policy using route-maps to either specify a route to the null-interface to filter the traffic, or specify a route to a dedicated next hop, for example a management network.

This solution provides a nice defense-in-depth method to prevent leaks, because the security issue is not longer a matter of complex packet matching but a simple routing event.



## ACLs and Fragments

```
ip access-list extended PARANOID
       deny   ip   any any fragment
```

**A port-based ACL will pass all fragments by default (with overlap checks):**

- **Fragments can be filtered, if so desired (not recommended)**
- **Fragments can be rate-limited (match using ACL, police their rate to a required value)**

DPS 1.0—5-3-7

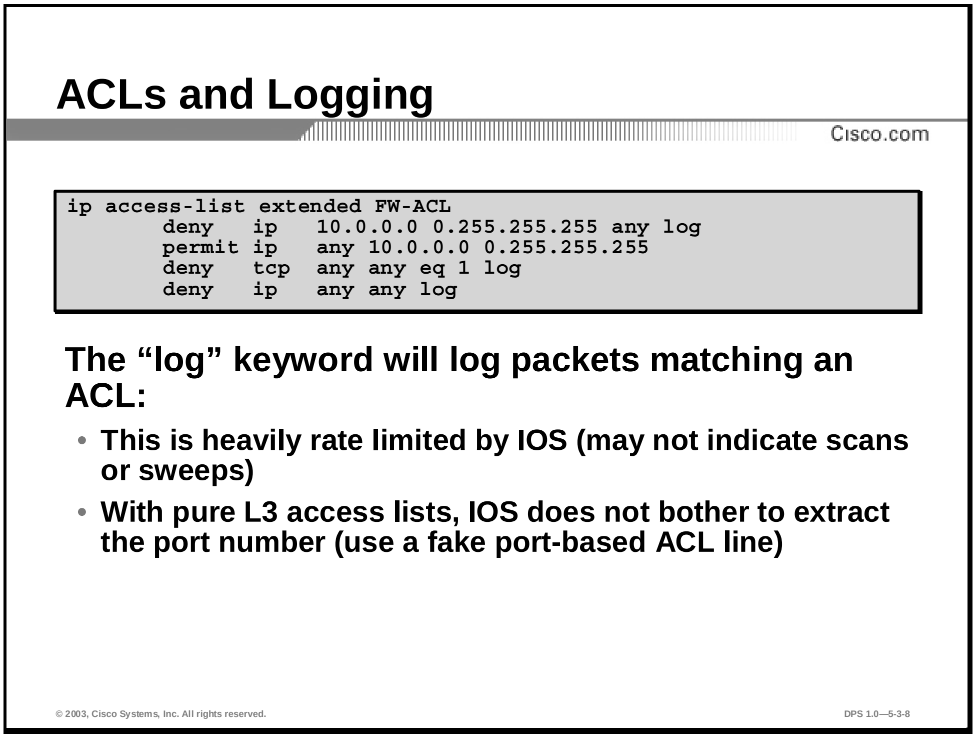
# IP Fragmentation Problems

If IP fragments a packet longer than the actual MTU a port-based access lists will notice the association and pass all fragments by default. Additionally the ACLs are aware of overlapping fragments, which might cause a well-known security problem.

Another important problem with IP fragments is their impact on traffic performance. A receiving IP host starts a reassembly timer for the first fragment received and collects all fragments before reassembling the parts to the original packet. The attacker might fill the reassembly buffers, causing a simple denial-of-service attack directed at the end host.

Because of this reason, an organization might want to filter fragments using ACLs. The configuration example above shows a “paranoid” ACL filtering any fragments.

However, this “paranoid” approach is generally not recommended, as there exists better solutions such as configure a fragment rate-limit, or use more advanced fragment handling methods, such as PIX Firewall’s virtual reassembly.



## ACLs and Logging

```
ip access-list extended FW-ACL
       deny   ip   10.0.0.0 0.255.255.255 any log
       permit ip   any 10.0.0.0 0.255.255.255
       deny   tcp  any any eq 1 log
       deny   ip   any any log
```

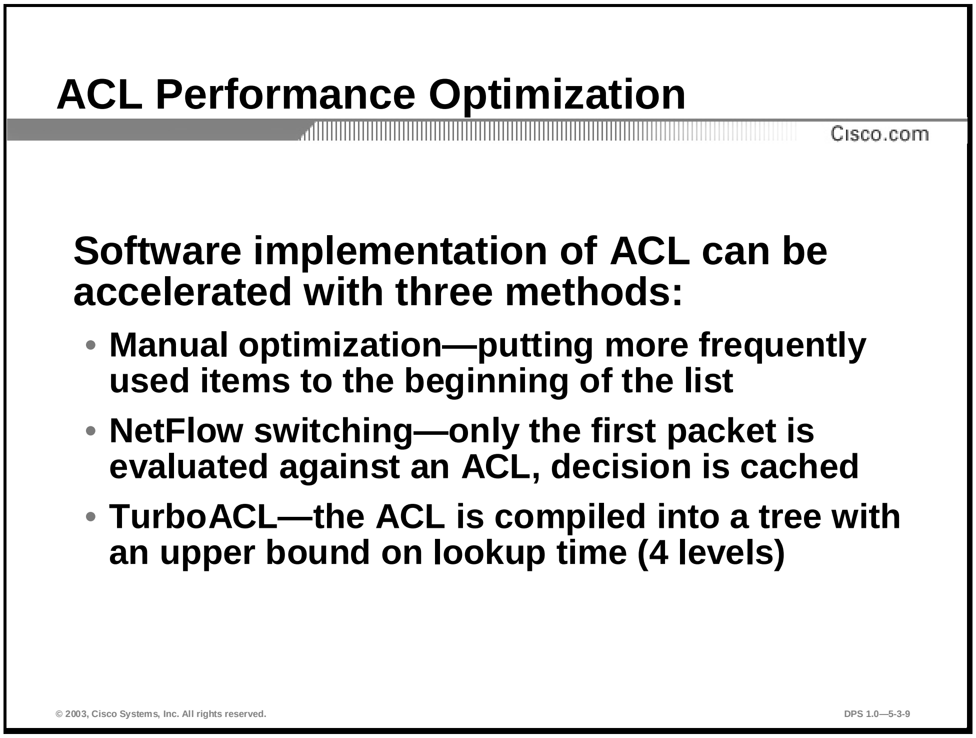
**The “log” keyword will log packets matching an ACL:**

- **This is heavily rate limited by IOS (may not indicate scans or sweeps)**
- **With pure L3 access lists, IOS does not bother to extract the port number (use a fake port-based ACL line)**

DPS 1.0—5-3-8

# Hint

Use the “log” keyword in ACLs specifications in order to enable the device to log all events when packets match the defined ACL patterns. However logging is heavily rate limited so any network scans or sweeps might not be detected. On the other hand, using pure layer-3 ACLs, the IOS does not bother to extract the port numbers, so using a fake layer 4 ACL line at the end is recommended for logging of denied packets’ port numbers.



## ACL Performance Optimization

Cisco.com

**Software implementation of ACL can be accelerated with three methods:**

- **Manual optimization—putting more frequently used items to the beginning of the list**
- **NetFlow switching—only the first packet is evaluated against an ACL, decision is cached**
- **TurboACL—the ACL is compiled into a tree with an upper bound on lookup time (4 levels)**

DPS 1.0—5-3-9

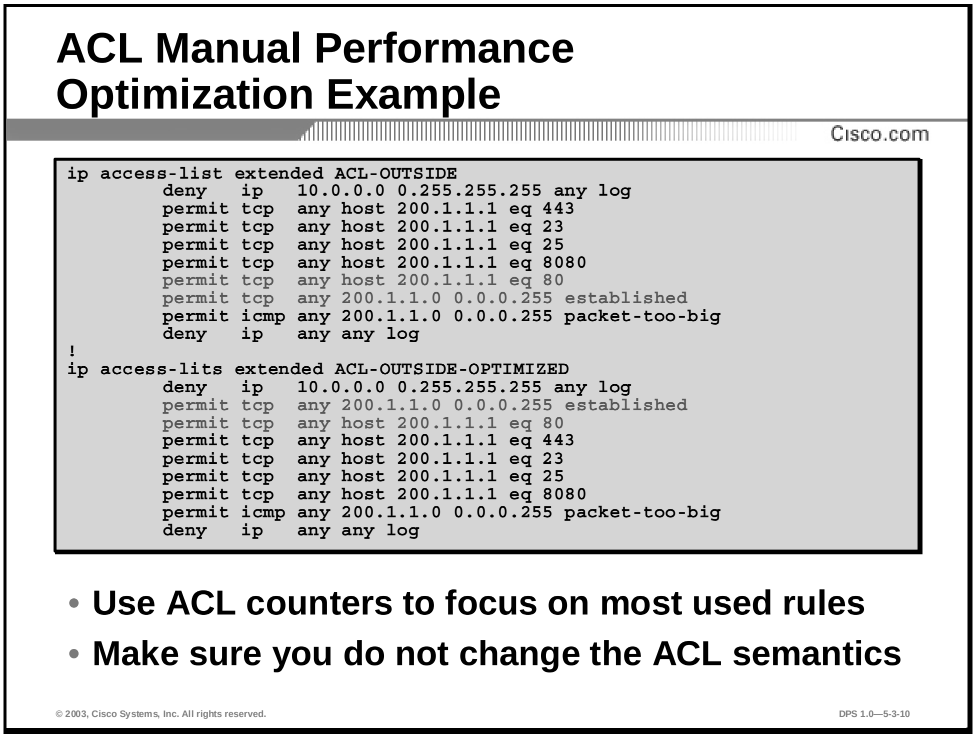
## Tune Your Access Lists

Other than wire-speed hardware based access control, any software based implementation of ACLs create CPU load and throttles the traffic. The impact depends on the complexity of the ACL and the number of entries involved. However, the following guideline enables you to improve the total forwarding speed significantly.

First make manual optimizations: put the more frequently used ACL-items to the beginning of the list—remember, that the ACL-processing stops when a match occurs.

Secondly, use NetFlow switching: Only the first packet of a flow is evaluated against an ACL and the resulting decision is cached.

Thirdly, use TurboACL: This method provides more consistent performance, as it defines an upper bound on lookup times. Here the ACL is compiled into an up to 4-level decision tree.



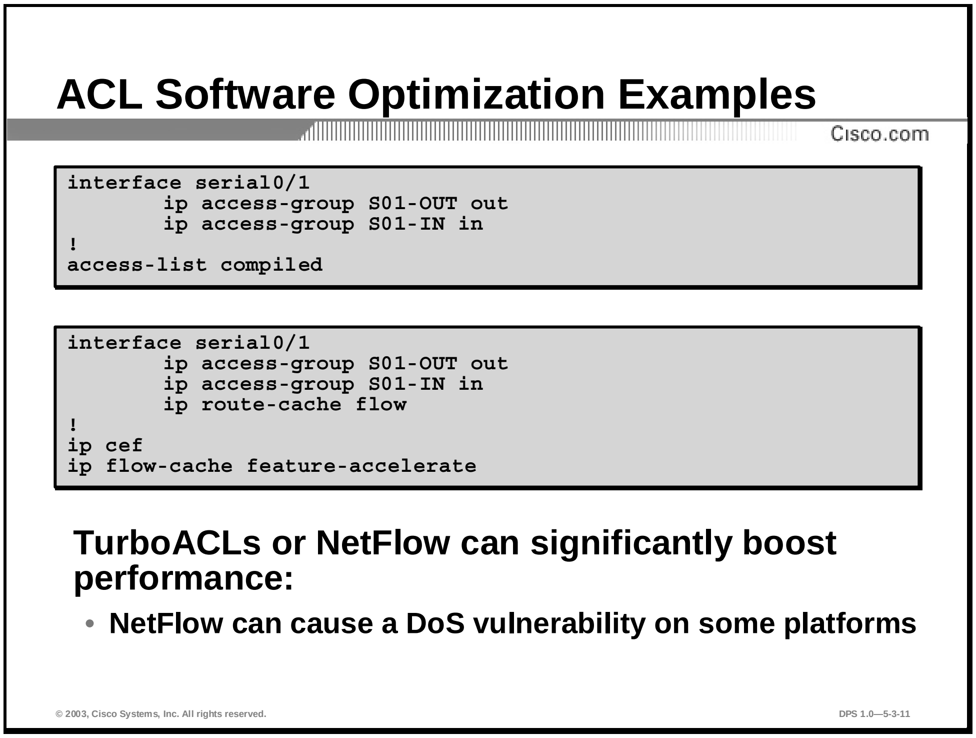
## ACL Manual Performance Optimization Example

```
ip access-list extended ACL-OUTSIDE
        deny   ip   10.0.0.0 0.255.255.255 any log
        permit tcp  any host 200.1.1.1 eq 443
        permit tcp  any host 200.1.1.1 eq 23
        permit tcp  any host 200.1.1.1 eq 25
        permit tcp  any host 200.1.1.1 eq 8080
        permit tcp  any host 200.1.1.1 eq 80
        permit tcp  any 200.1.1.0 0.0.0.255 established
        permit icmp any 200.1.1.0 0.0.0.255 packet-too-big
        deny   ip   any any log
!
ip access-lits extended ACL-OUTSIDE-OPTIMIZED
        deny   ip   10.0.0.0 0.255.255.255 any log
        permit tcp  any 200.1.1.0 0.0.0.255 established
        permit tcp  any host 200.1.1.1 eq 80
        permit tcp  any host 200.1.1.1 eq 443
        permit tcp  any host 200.1.1.1 eq 23
        permit tcp  any host 200.1.1.1 eq 25
        permit tcp  any host 200.1.1.1 eq 8080
        permit icmp any 200.1.1.0 0.0.0.255 packet-too-big
        deny   ip   any any log
```

- **Use ACL counters to focus on most used rules**
- **Make sure you do not change the ACL semantics**

## Example #1

This figure illustrates a configuration example for manual ACL optimization. Lines, which are often matched, are moved to the top of the ACL, but great care must be taken no to change the ACL meaning. In this example, the line, which permits “established” traffic can be pushed up in the ruleset, as it is usually the most frequently matched. However, it can only “overtake” lines, which are completely independent of it (i.e. if both ACLs could never match the same packet). Here, it can be inserted in front of all lines filtering on a specific port number, but behind the anti-spoofing filter, which is a subset of it and should deny dangerous traffic even if would be later evaluated as established TCP traffic.



## ACL Software Optimization Examples

```
interface serial0/1
       ip access-group S01-OUT out
       ip access-group S01-IN in
!
access-list compiled
```

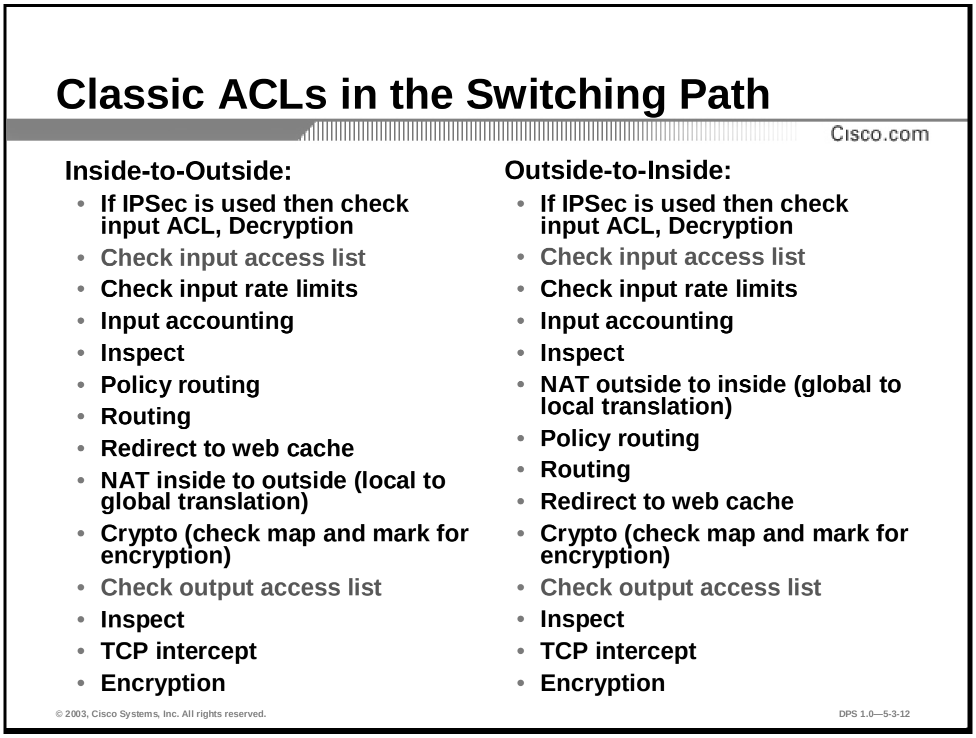
```
interface serial0/1
       ip access-group S01-OUT out
       ip access-group S01-IN in
       ip route-cache flow
!
ip cef
ip flow-cache feature-accelerate
```

**TurboACLs or NetFlow can significantly boost performance:**

- **NetFlow can cause a DoS vulnerability on some platforms**

## Examples #2 and #3

This figure illustrates configuration examples for TurboACLs (top) and ACL using NetFlow (bottom). Netflow acceleration should be used to accelerate ACLs in relatively low connection-rate environments, if software routers are used. Note that NetFlow itself might cause a DoS vulnerability on some platforms, as the first packet of the flow is heavily processed (the Cisco 7500 platform is a prime example, where the packet is copied from the VIP two times to system buffers in main memory), if a flow-flooding attack is attempted at the network device. TurboACLs generally have no caveats and should be used in most cases.



## Classic ACLs in the Switching Path

**Inside-to-Outside:**

- If IPSec is used then check input ACL, Decryption
- Check input access list
- Check input rate limits
- Input accounting
- Inspect
- Policy routing
- Routing
- Redirect to web cache
- NAT inside to outside (local to global translation)
- Crypto (check map and mark for encryption)
- Check output access list
- Inspect
- TCP intercept
- Encryption

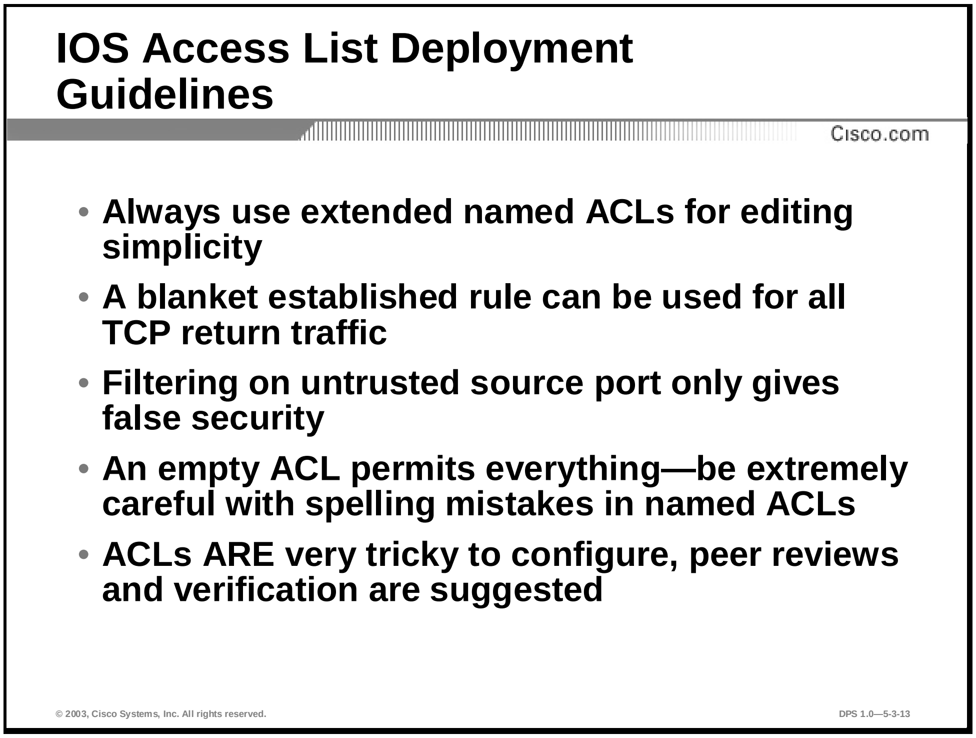
**Outside-to-Inside:**

- If IPSec is used then check input ACL, Decryption
- Check input access list
- Check input rate limits
- Input accounting
- Inspect
- NAT outside to inside (global to local translation)
- Policy routing
- Routing
- Redirect to web cache
- Crypto (check map and mark for encryption)
- Check output access list
- Inspect
- TCP intercept
- Encryption

DPS 1.0—5-3-12

# The Big Picture

The list above shows the chronological order of how an IOS device processes a packet. Note that classical ACLs are processed nearly at the very beginning and before forwarding the packet. After the output ACL passes the packet, a higher-level inspection might follow.



## IOS Access List Deployment Guidelines

- Always use extended named ACLs for editing simplicity
- A blanket established rule can be used for all TCP return traffic
- Filtering on untrusted source port only gives false security
- An empty ACL permits everything—be extremely careful with spelling mistakes in named ACLs
- ACLs ARE very tricky to configure, peer reviews and verification are suggested

DPS 1.0—5-3-13

## Guidelines

When creating IOS ACLs the following rules have been proved to be useful in practice:

- Always use extended named ACLs for editing simplicity—deletion and manipulation of rules is much easier. Furthermore there is less confusion when using reasonable names for each ACL.

- A blanket "established" rule could be used for all TCP return traffic. This rule will permit traffic with either the ACK, FIN, or RST flags set to one (i.e. packets with cannot initiate a TCP session).

- Do not filter on untrusted source ports as they can be easily manipulated and does not mean anything—this provides false security only.

- When using named ACLs be extremely carefully with spelling mistakes because an empty ACL permits anything!

- Do not rely on your own skills: creating ACLs is a complex task and ACLs are indeed tricky. Always ask for peer reviews and verification.

## Practice

Q1) Which of the following TCP packets would be permitted by an “established” rule?

A) A TCP SYN packet

B) A TCP FIN packet

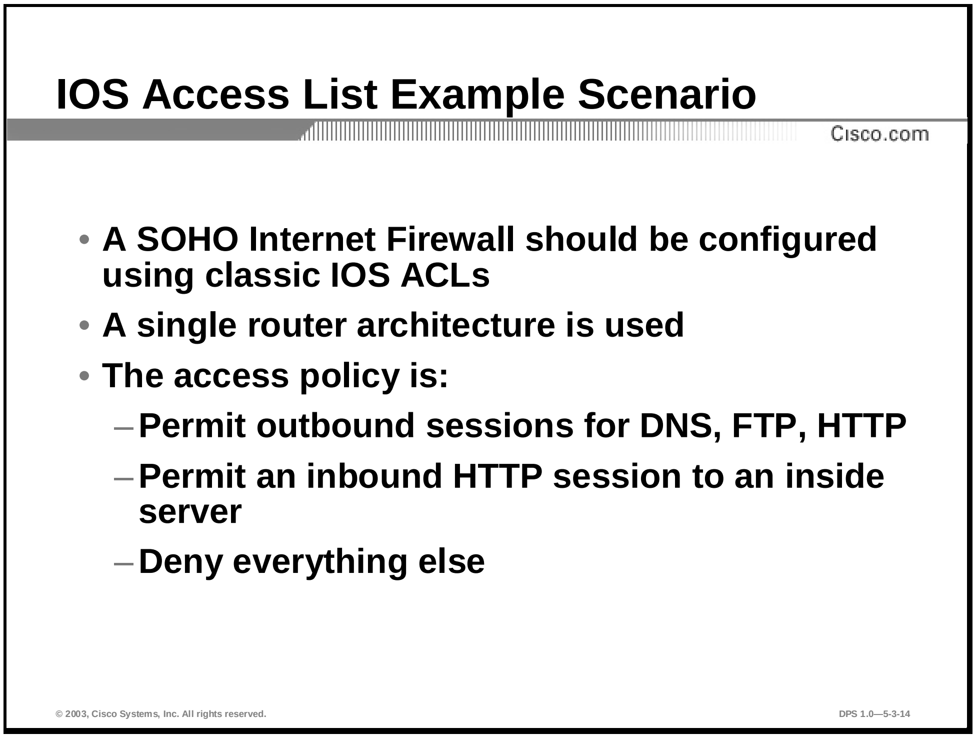
C) A TCP ACK packet

D) A TCP RST packet

E) A TCP PSH packet

F) A TCP URG packet

# Example Scenario



## IOS Access List Example Scenario

Cisco.com

- A SOHO Internet Firewall should be configured using classic IOS ACLs
- A single router architecture is used
- The access policy is:
  - Permit outbound sessions for DNS, FTP, HTTP
  - Permit an inbound HTTP session to an inside server
  - Deny everything else

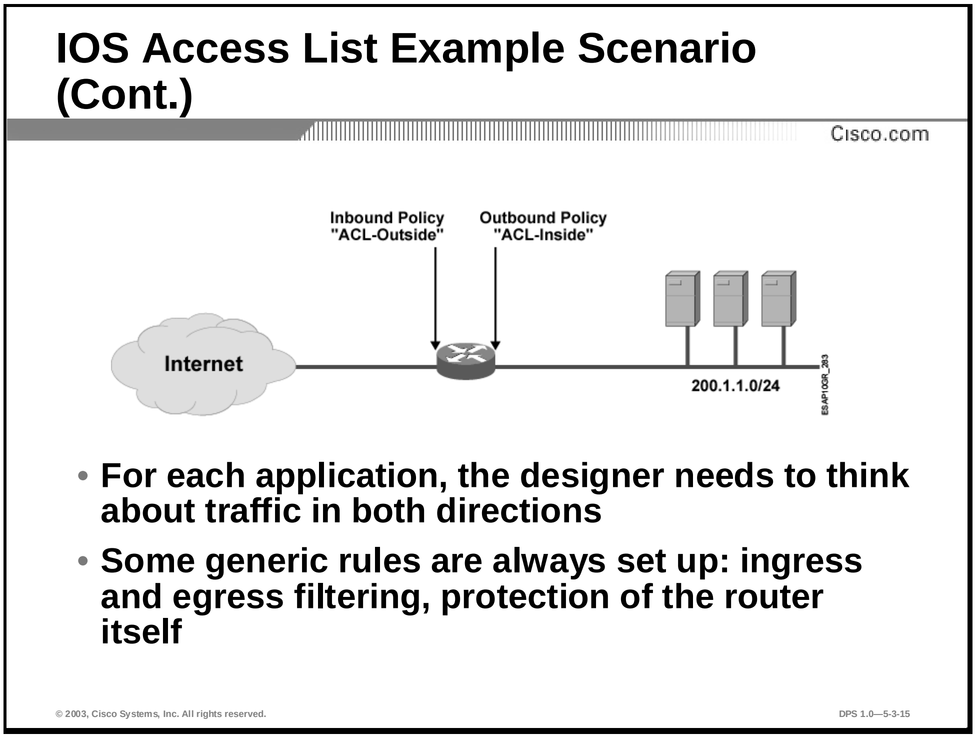
DPS 1.0—5-3-14

## Objective

This section will enable the learner to identify a correct configuration of advanced IOS ACL features in a firewall implementation.

## Introduction

Suppose a SOHO Internet Firewall should be configured using classic IOS ACLs. Only a single router is used. The access policy includes: permit outbound sessions for DNS, FTP, HTTP, permit inbound HTTP sessions to an inside server, finally deny everything else.



## IOS Access List Example Scenario (Cont.)

Cisco.com

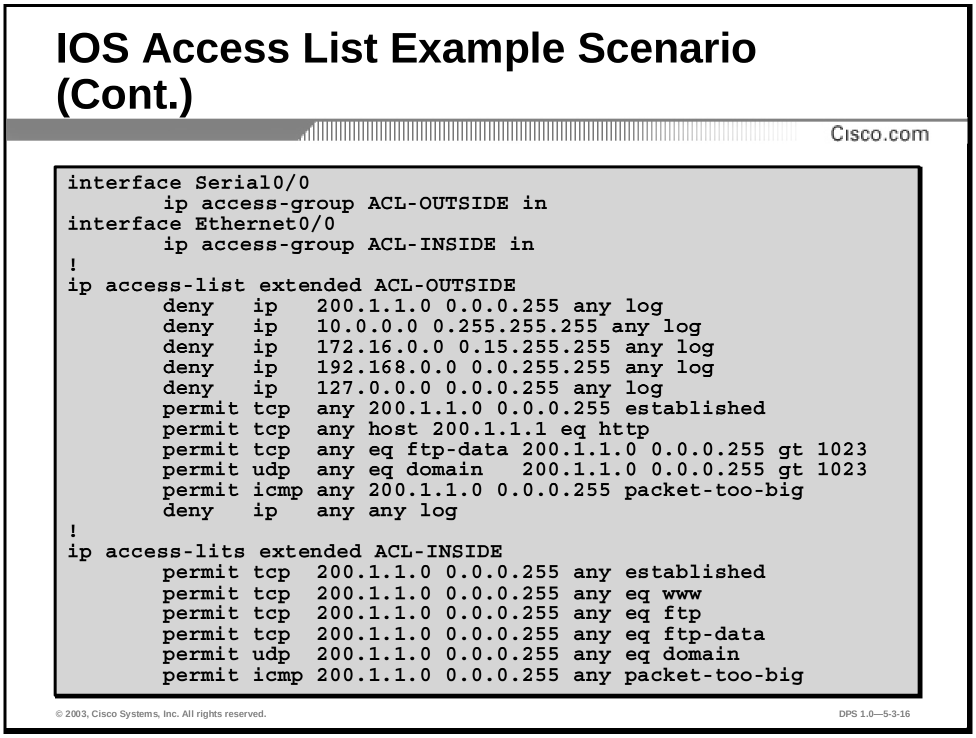
- For each application, the designer needs to think about traffic in both directions
- Some generic rules are always set up: ingress and egress filtering, protection of the router itself

DPS 1.0—5-3-15

## Example Scenario: Basic Filter Design and Placement

When using pure packet filtering, the designer of the ruleset must exactly know, how the supported applications look like on the network, and design rulesets to permit outgoing and incoming traffic. In practice, the most difficult aspect of rule definition is the proper filtering of return (server-to-client) traffic, which often cannot be filtered securely (usually, when incoming sessions to random client ports are needed, such as with FTP).

In this example, the outbound packet traffic will be filtered using the packet filtering ACL (ACL-Inside), applied inbound on the inside interface. All inbound packet traffic will be filtering using the packet filtering ACL (ACL-Outside) on the outside router interface. The outside ACL should also protect the router itself, by denying all, or permitting only select traffic to any router IP address from the untrusted network.



## IOS Access List Example Scenario (Cont.)

```
interface Serial0/0
        ip access-group ACL-OUTSIDE in
interface Ethernet0/0
        ip access-group ACL-INSIDE in
!
ip access-list extended ACL-OUTSIDE
        deny   ip   200.1.1.0 0.0.0.255 any log
        deny   ip   10.0.0.0 0.255.255.255 any log
        deny   ip   172.16.0.0 0.15.255.255 any log
        deny   ip   192.168.0.0 0.0.255.255 any log
        deny   ip   127.0.0.0 0.0.0.255 any log
        permit tcp  any 200.1.1.0 0.0.0.255 established
        permit tcp  any host 200.1.1.1 eq http
        permit tcp  any eq ftp-data 200.1.1.0 0.0.0.255 gt 1023
        permit udp  any eq domain   200.1.1.0 0.0.0.255 gt 1023
        permit icmp any 200.1.1.0 0.0.0.255 packet-too-big
        deny   ip   any any log
!
ip access-lits extended ACL-INSIDE
        permit tcp  200.1.1.0 0.0.0.255 any established
        permit tcp  200.1.1.0 0.0.0.255 any eq www
        permit tcp  200.1.1.0 0.0.0.255 any eq ftp
        permit tcp  200.1.1.0 0.0.0.255 any eq ftp-data
        permit udp  200.1.1.0 0.0.0.255 any eq domain
        permit icmp 200.1.1.0 0.0.0.255 any packet-too-big
```

## Example Scenario: Configuration

This figure illustrates a configuration example for extended inbound and outbound ACLs. Note that the most frequent-used rules are specified at the beginning.

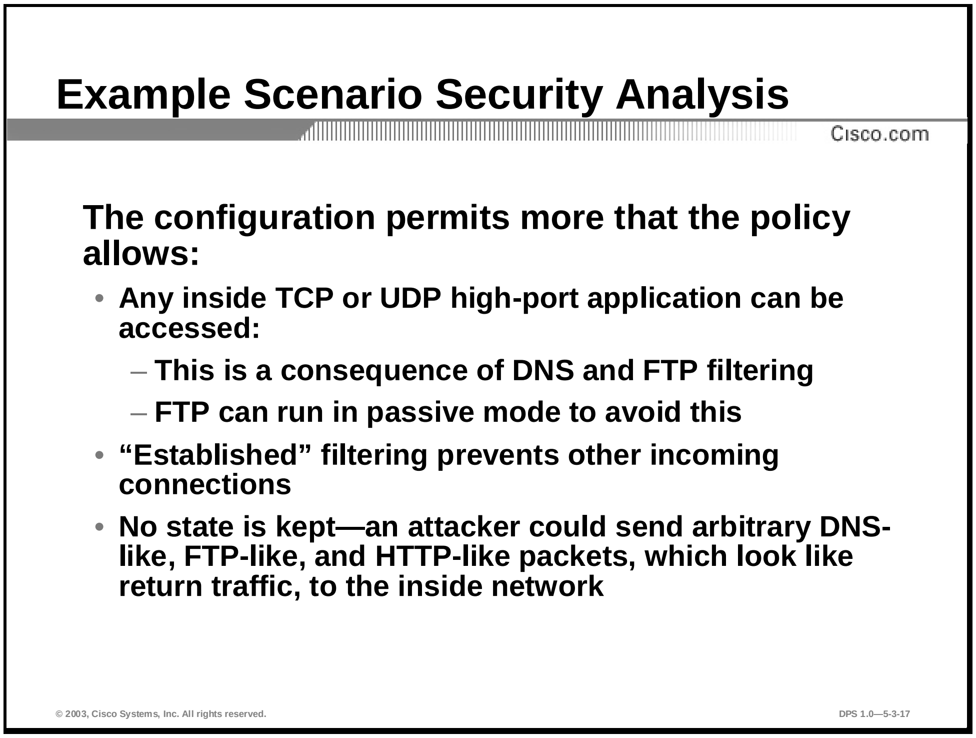
The ruleset ACL-OUTSIDE permitting inbound traffic (from the untrusted to the trusted network), applied inbound to the outside (untrusted) interface, needs to

- deny spoofed addresses using layer-3 filters
- permit established TCP traffic (return traffic of outgoing sessions)
- permit inbound sessions to the internal HTTP server
- permit inbound high-port FTP data sessions, which are a part of outgoing FTP sessions; these cannot be matched by the \lished rule as the data session establishes inbound
- permit inbound high-port UDP packets from outside port 53; these are DNS replies, and use a random client port
- permit ICMP packet-too-big messages, needed for Path MTU Discovery to work

The ruleset ACL-INSIDE permitting outbound traffic (from the trusted to the untrusted network), applied inbound to the inside (trusted) interface needs to

- permit outgoing established TCP traffic (to allow return traffic of the inbound HTTP sessions to the internal web server, and to increase performance for traffic initially matching other TCP entries later in the ACL)

- permit outgoing HTTP packets
- permit outgoing FTP packets
- permit outgoing FTP data packets (to support passive FTP)
- permit outgoing DNS queries
- permit ICMP packet-too-big messages, needed for Path MTU Discovery to work



### Example Scenario Security Analysis

Cisco.com

**The configuration permits more that the policy allows:**

- **Any inside TCP or UDP high-port application can be accessed:**
  - **This is a consequence of DNS and FTP filtering**
  - **FTP can run in passive mode to avoid this**
- **"Established" filtering prevents other incoming connections**
- **No state is kept—an attacker could send arbitrary DNS-like, FTP-like, and HTTP-like packets, which look like return traffic, to the inside network**

© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—5-3-17

## Example Scenario: Security Analysis

This configuration example has several security flaws. First there is no prevention to access any inside TCP or UDP high-port application. Of course DNS and FTP filtering seems to be not a good idea, to mitigate it, you could only allow passive mode FTP, which only uses outgoing connections.

Then simply filtering "established" patterns ultimately prevents all incoming connections. Theses simple ACLs do not maintain states for each established session. Thus non-session related packets might be forged by attackers to make them look like return traffic.

## Practice

Q1) At the minimum, how many packet filtering rules (ACL lines) are required on each interface to support active FTP, and only active FTP, between two networks?

A) one

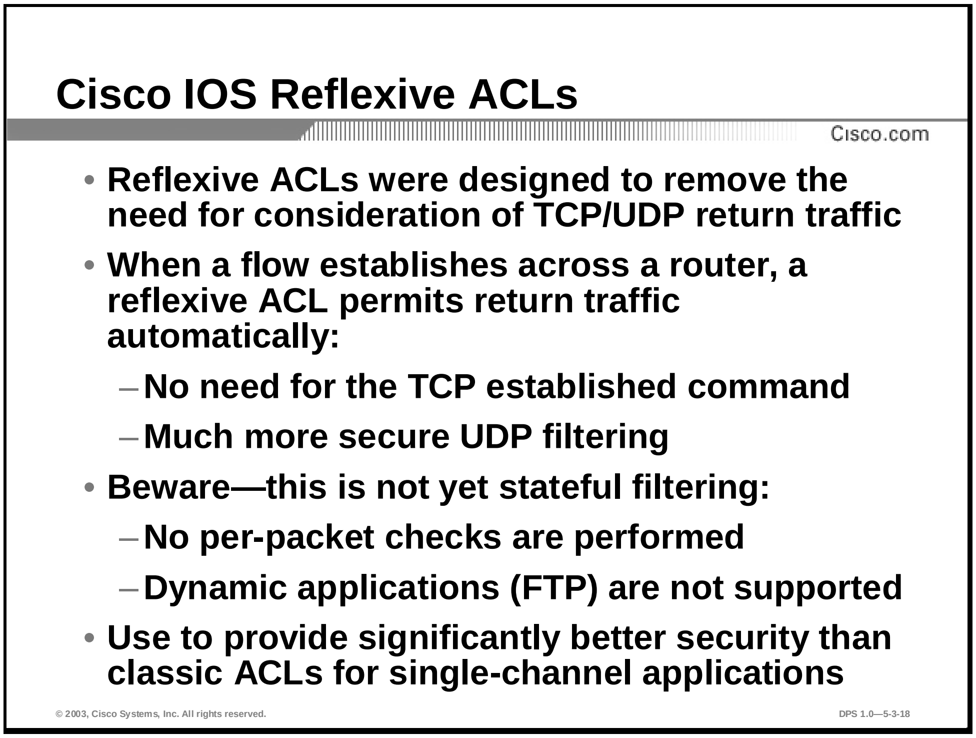
B) two

C) three

D) four

E) five

# IOS Reflexive ACLs



## Cisco IOS Reflexive ACLs

Cisco.com

- Reflexive ACLs were designed to remove the need for consideration of TCP/UDP return traffic
- When a flow establishes across a router, a reflexive ACL permits return traffic automatically:
  - No need for the TCP established command
  - Much more secure UDP filtering
- Beware—this is not yet stateful filtering:
  - No per-packet checks are performed
  - Dynamic applications (FTP) are not supported
- Use to provide significantly better security than classic ACLs for single-channel applications

DPS 1.0—5-3-18

## Objective

This section will enable the learner to configure IOS reflexive ACLs in an advanced firewall design scenario.

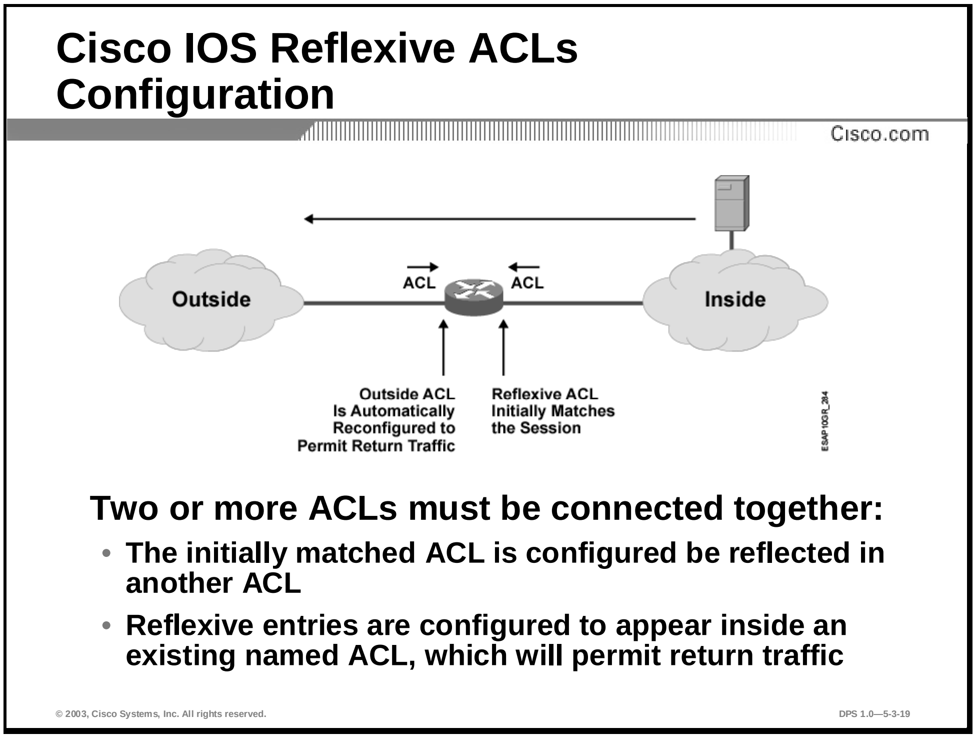
## Introduction

Classic Cisco IOS ACLs address return traffic of connections either via the "established" filtering (for TCP sessions), or by permitting high-port UDP traffic back to the clients. Both approaches are inexact, and lead to possible leaks in the firewall.

## The Return Traffic Problem

By forging packets to make them appear like return traffic is a relatively simple way to capture a session. For this reason, so-called "reflexive" ACLs were designed to prevent such types of attacks. Only when a flow is established—by passing an ACL—the router automatically creates a reflexive ACL especially to permit the return traffic. In this design there is no need for the established command and even UDP filtering is possible.

Note that Reflexive ACLs is not yet stateful filtering, as the individual packets are no examined. Also dynamic applications that move their socket information such as FTP are not supported.



## Cisco IOS Reflexive ACLs Configuration

Cisco.com

Outside
ACL
ACL
Inside
Outside ACL Is Automatically Reconfigured to Permit Return Traffic
Reflexive ACL Initially Matches the Session

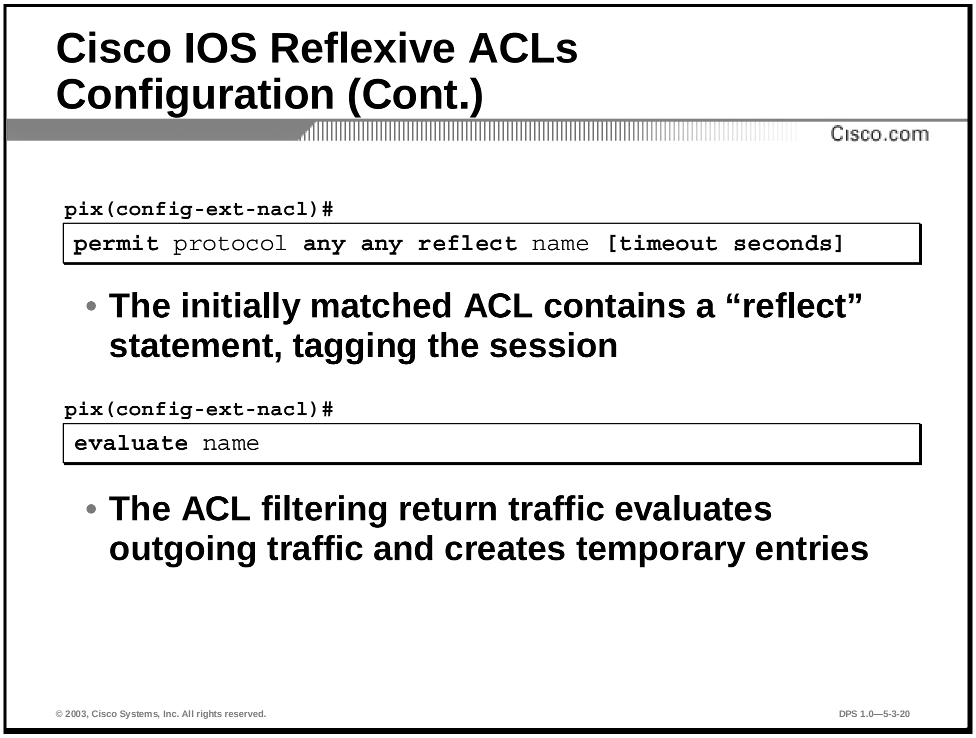
**Two or more ACLs must be connected together:**

- **The initially matched ACL is configured be reflected in another ACL**
- **Reflexive entries are configured to appear inside an existing named ACL, which will permit return traffic**

DPS 1.0—5-3-19

## How It Works

Reflexive ACLs require two or more ACLs that are connected together. For example an inbound ACL is configured to be reflexive so that the matched patterns are reflected in another ACL, for example an outbound ACL. The automatically created “reflected” ACL is reconfigured to permit return traffic of the inbound flow.



## Cisco IOS Reflexive ACLs Configuration (Cont.)

Cisco.com

```
pix(config-ext-nacl)#
permit protocol any any reflect name [timeout seconds]
```

- The initially matched ACL contains a “reflect” statement, tagging the session

```
pix(config-ext-nacl)#
evaluate name
```

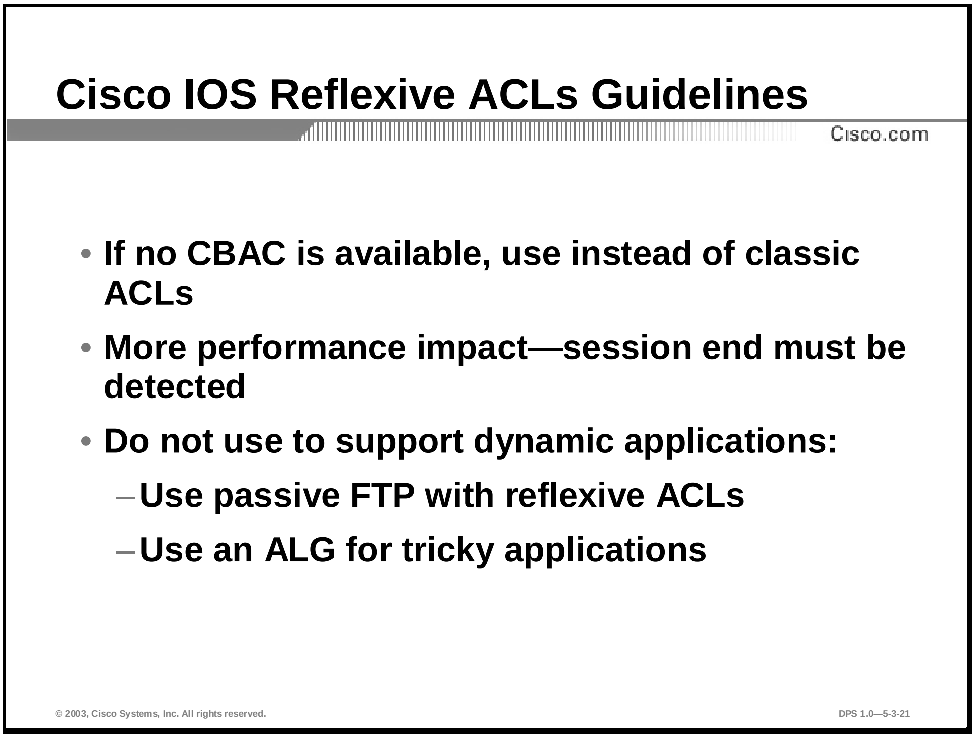
- The ACL filtering return traffic evaluates outgoing traffic and creates temporary entries

DPS 1.0—5-3-20

## Configuration Commands

The commands listed in this figure are used to configure Reflexive ACLs. Note the keyword **reflect** that is added to the standard extended ACL command. The reflected ACL only creates an entry for a specified timeout, which is 5 minutes by default.

The command **evaluate** enables the IOS device to create a dynamic ACL entry when there is a match of a previously configured **permit-**statement. This statement references to the ACL name used with the **permit** statement.



## Cisco IOS Reflexive ACLs Guidelines

Cisco.com

- **If no CBAC is available, use instead of classic ACLs**
- **More performance impact—session end must be detected**
- **Do not use to support dynamic applications:**
  - **Use passive FTP with reflexive ACLs**
  - **Use an ALG for tricky applications**

DPS 1.0—5-3-21

## Guidelines

Context-based ACLs are more powerful and provide more security, so the usage of CBAC should be preferred. Note that Reflexive ACLs cause additional resource efforts because the session end must be detected. Therefore Reflexive ACLs can have some performance impact.

Reflexive ACLs do not support dynamic applications. In case of FTP use the passive mode, because the socket information does not change for each flow. In case of tricky applications there is no way around application aware solutions such as ALGs.

## Practice

Q1) When are access list entries created with reflexive ACLs?

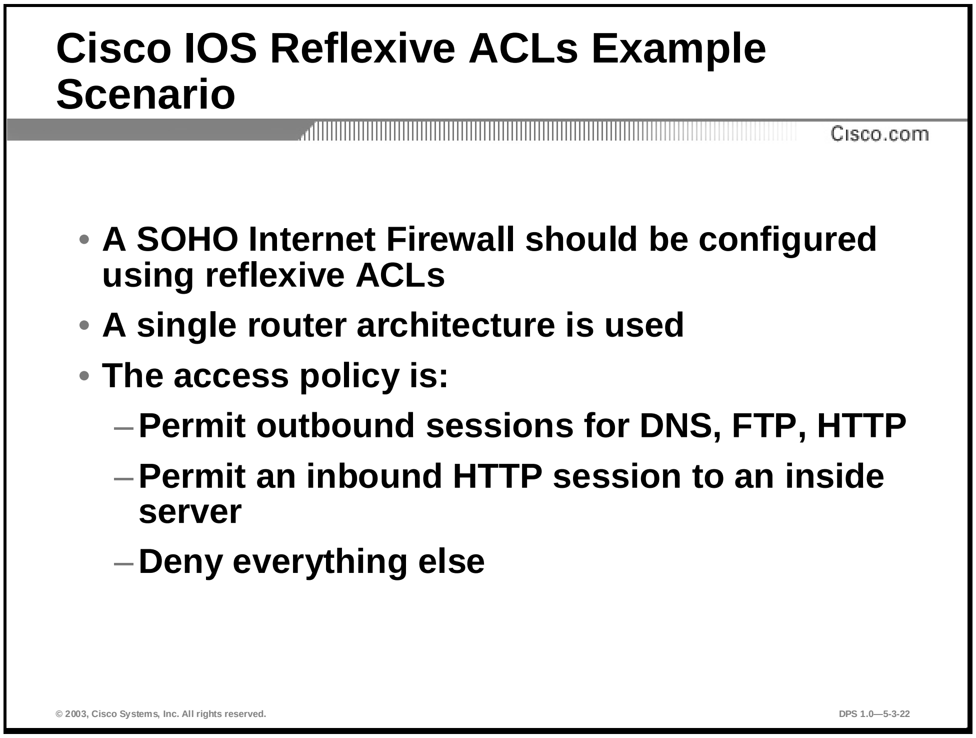
A) when the first packet of a session crosses the router

B) when the three-way handshake has completed

C) when the RST packet is seen

D) when the ACLs are configured

# Example Scenario



## Cisco IOS Reflexive ACLs Example Scenario

- A SOHO Internet Firewall should be configured using reflexive ACLs
- A single router architecture is used
- The access policy is:
  - Permit outbound sessions for DNS, FTP, HTTP
  - Permit an inbound HTTP session to an inside server
  - Deny everything else

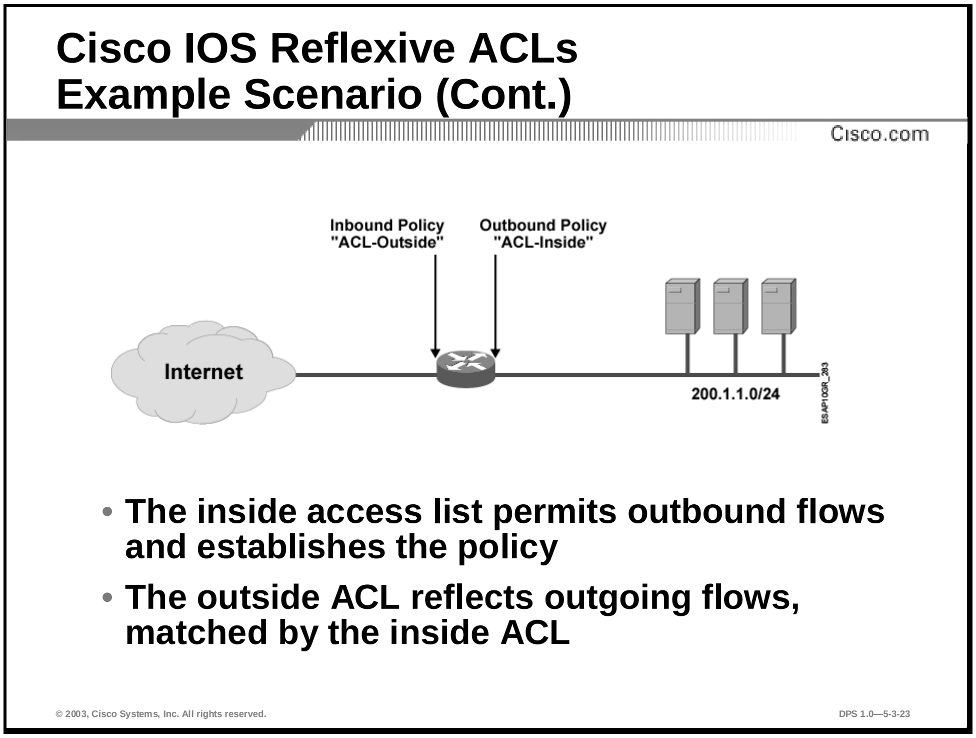
© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—5-3-22

## Objective

This section will enable the learner to identify a correct configuration of IOS ACL reflexive features in a firewall implementation.

## Introduction

Suppose a SOHO Internet Firewall should be configured using classic IOS ACLs. Only a single router is used. The access policy includes: permit outbound sessions for DNS, FTP, HTTP, permit inbound HTTP sessions to an inside server, finally deny everything else.



## Cisco IOS Reflexive ACLs Example Scenario (Cont.)

Cisco.com

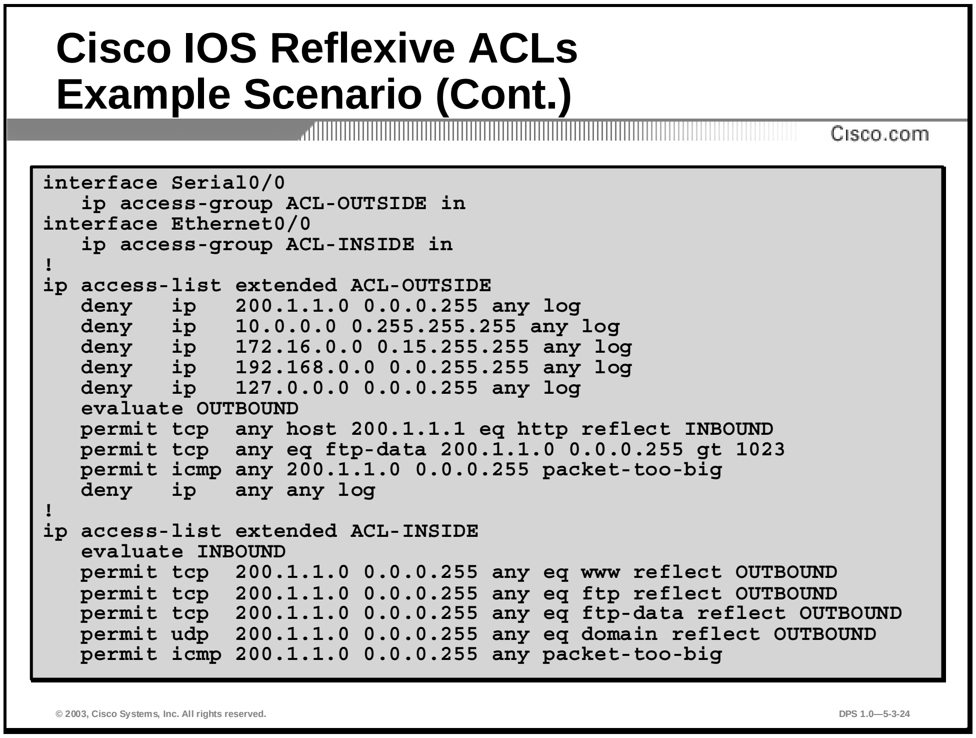
- The inside access list permits outbound flows and establishes the policy
- The outside ACL reflects outgoing flows, matched by the inside ACL

DPS 1.0—5-3-23

## Example Scenario: Basic Filter Design and Placement

Again the direction of traffic flows is fundamental for configuration considerations. Using reflexive ACLs, the inside ACL permits the outbound flow. The outside ACL reflects outgoing flows matched by the inside ACL and allows return traffic.

In this example, the outbound packet traffic will be filtered using the packet filtering ACL (ACL-Inside), applied inbound on the inside interface. All inbound packet traffic will be filtering using the packet filtering ACL (ACL-Outside) on the outside router interface. The outside ACL should also protect the router itself, by denying all, or permitting only select traffic to any router IP address from the untrusted network.



## Cisco IOS Reflexive ACLs Example Scenario (Cont.)

```
interface Serial0/0
   ip access-group ACL-OUTSIDE in
interface Ethernet0/0
   ip access-group ACL-INSIDE in
!
ip access-list extended ACL-OUTSIDE
   deny   ip   200.1.1.0 0.0.0.255 any log
   deny   ip   10.0.0.0 0.255.255.255 any log
   deny   ip   172.16.0.0 0.15.255.255 any log
   deny   ip   192.168.0.0 0.0.255.255 any log
   deny   ip   127.0.0.0 0.0.0.255 any log
   evaluate OUTBOUND
   permit tcp  any host 200.1.1.1 eq http reflect INBOUND
   permit tcp  any eq ftp-data 200.1.1.0 0.0.0.255 gt 1023
   permit icmp any 200.1.1.0 0.0.0.255 packet-too-big
   deny   ip   any any log
!
ip access-list extended ACL-INSIDE
   evaluate INBOUND
   permit tcp  200.1.1.0 0.0.0.255 any eq www reflect OUTBOUND
   permit tcp  200.1.1.0 0.0.0.255 any eq ftp reflect OUTBOUND
   permit tcp  200.1.1.0 0.0.0.255 any eq ftp-data reflect OUTBOUND
   permit udp  200.1.1.0 0.0.0.255 any eq domain reflect OUTBOUND
   permit icmp 200.1.1.0 0.0.0.255 any packet-too-big
```

DPS 1.0—5-3-24

## Example Scenario: Configuration

This figure illustrates a configuration example for the reflexive ACL used in this example scenario. Note that the first statement entered in the ACL-INSIDE is the evaluate statement which reflects the rule specified above and named INBOUND, allowing HTTP return traffic to be forwarded into the enterprise network.

The functionality of the reflexive ACLs will allow us to have a more flow-oriented ruleset, and where single-flow TCP or UDP applications (such as HTTP or DNS) will not have to rely on the “established” rule, or a rule permitting all client ports to address return traffic. Instead, rules to permit return traffic will be created automatically.

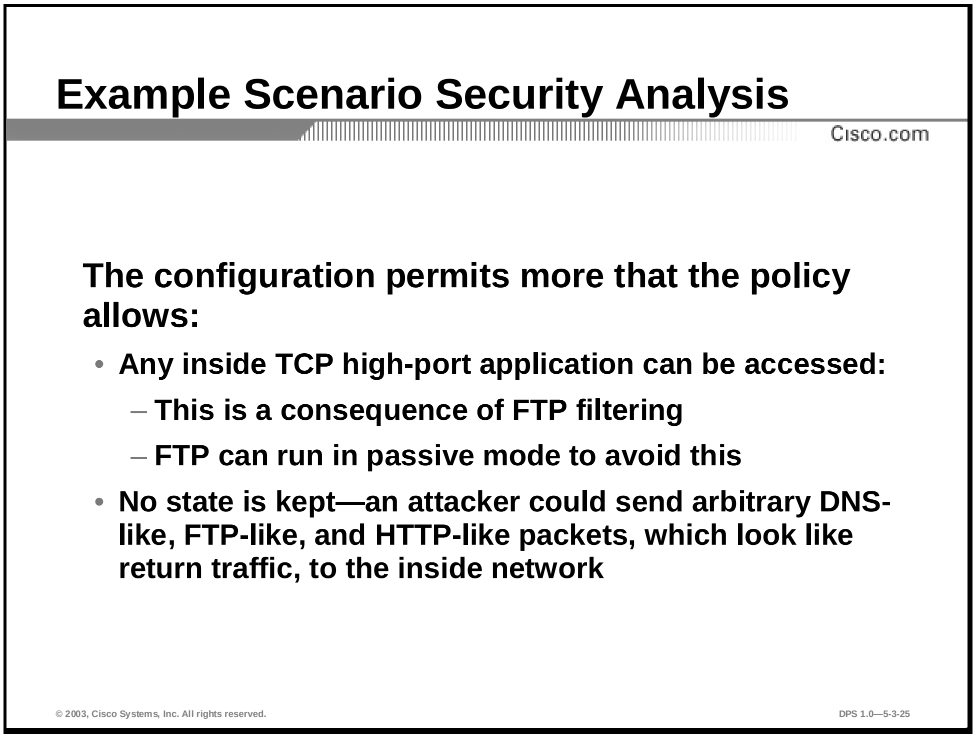
The ruleset ACL-OUTSIDE permitting inbound traffic (from the untrusted to the trusted network), applied inbound to the outside (untrusted) interface, needs to

- deny spoofed addresses using layer-3 filters
- permit inbound sessions to the internal HTTP server
- permit inbound high-port FTP data sessions, which are a part of outgoing FTP sessions; these cannot be matched by the established rule as the data session establishes inbound
- permit ICMP packet-too-big messages, needed for Path MTU Discovery to work

The ruleset ACL-INSIDE permitting outbound traffic (from the trusted to the untrusted network), applied inbound to the inside (trusted) interface needs to

- permit outgoing HTTP packets

- permit outgoing FTP packets
- permit outgoing FTP data packets (to support passive FTP)
- permit outgoing DNS queries
- permit ICMP packet-too-big messages, needed for Path MTU Discovery to work



## Example Scenario Security Analysis

**The configuration permits more that the policy allows:**

- **Any inside TCP high-port application can be accessed:**
  - **This is a consequence of FTP filtering**
  - **FTP can run in passive mode to avoid this**
- **No state is kept—an attacker could send arbitrary DNS-like, FTP-like, and HTTP-like packets, which look like return traffic, to the inside network**

DPS 1.0—5-3-25

## Example Scenario: Limits of Reflexive ACLs

The new configuration is much more strict compared to classic packet filtering, the major improvement being DNS reply handling. Still, the configuration permits more than the policy allows. For example any inside high-port TCP applications can be accessed—as consequence of FTP filtering. Remove the rules permitting FTP backconnections, and use FTP in passive mode to avoid this.

Furthermore no flow-state is maintained. Any forged packet that is assumed to be related to the actual flow may enter the network.

## Practice

Q1) Which of the following packets will  be permitted through a reflexive ACL-enabled interface, which only has a “deny ip any any” line in it and no dynamic entries?

A) TCP ACK packets

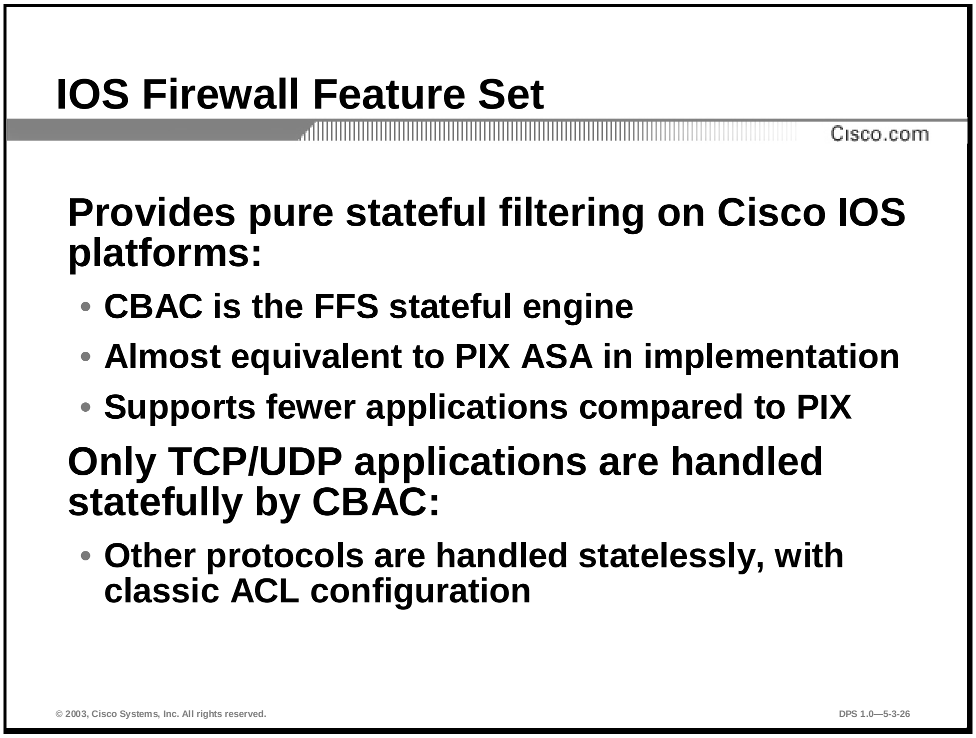
B) TCP RST packet

C) TCP FIN packet

D) ICMP packets

E) None of the listed

# Advanced IOS CBAC Configuration



## IOS Firewall Feature Set

Cisco.com

Provides pure stateful filtering on Cisco IOS platforms:

- CBAC is the FFS stateful engine
- Almost equivalent to PIX ASA in implementation
- Supports fewer applications compared to PIX

Only TCP/UDP applications are handled statefully by CBAC:

- Other protocols are handled statelessly, with classic ACL configuration

© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—5-3-26

## Objective

This section will enable the learner to configure CBAC in an advanced firewall design scenario.

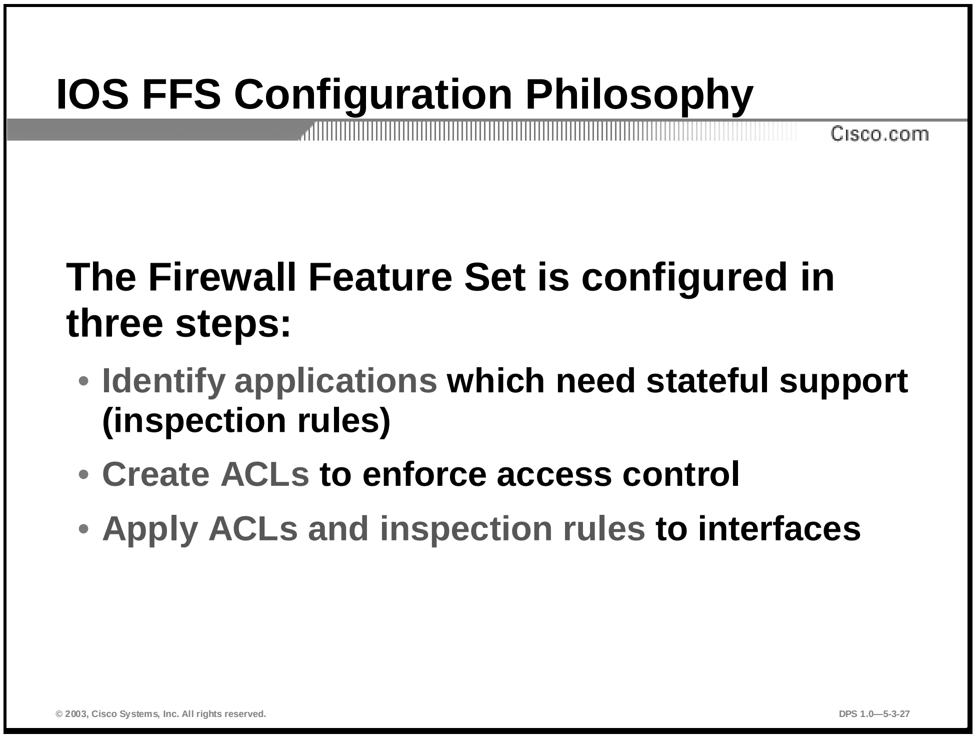
## Introduction

Using Context-based Access Control (CBAC) allows for very clean and elegant configurations.

## The Firewall Feature Set

IOS platforms configured as firewall can provide pure stateful filtering of flows through the Firewall Feature Set (FFS) stateful engine (CBAC), which is almost identically implemented as the PIX Adaptive Security Algorithm (ASA). However, a Cisco IOS firewall supports fewer applications compared to the PIX.

Only TCP and UDP applications are handled statefully by the CBAC; other protocols are handled statelessly using a classic ACL configuration.



## IOS FFS Configuration Philosophy

Cisco.com

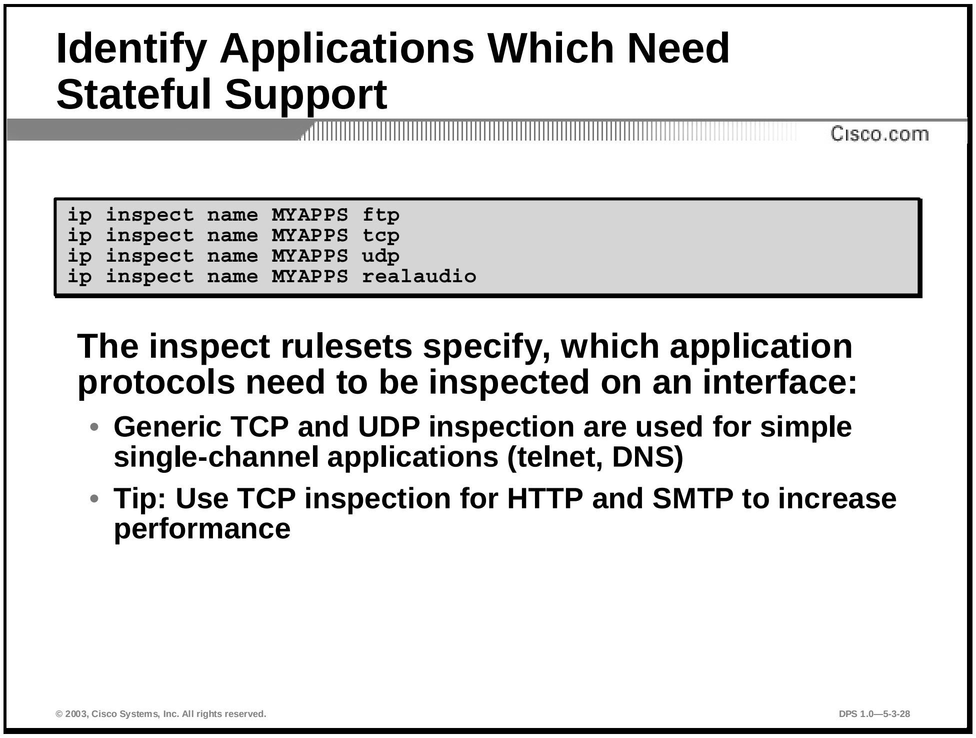
**The Firewall Feature Set is configured in three steps:**

- **Identify applications which need stateful support (inspection rules)**
- **Create ACLs to enforce access control**
- **Apply ACLs and inspection rules to interfaces**

DPS 1.0—5-3-27

# Three Step FFS Configuration

To configure the IOS Firewall Feature Set, three steps must be performed. First identify the applications used, which need a stateful support. In other words, identify the “inspection rules”. Secondly, create ACLs to enforce access control. And third apply these ACLs and inspection rules to the interfaces.



## Identify Applications Which Need Stateful Support

```
ip inspect name MYAPPS ftp
ip inspect name MYAPPS tcp
ip inspect name MYAPPS udp
ip inspect name MYAPPS realaudio
```

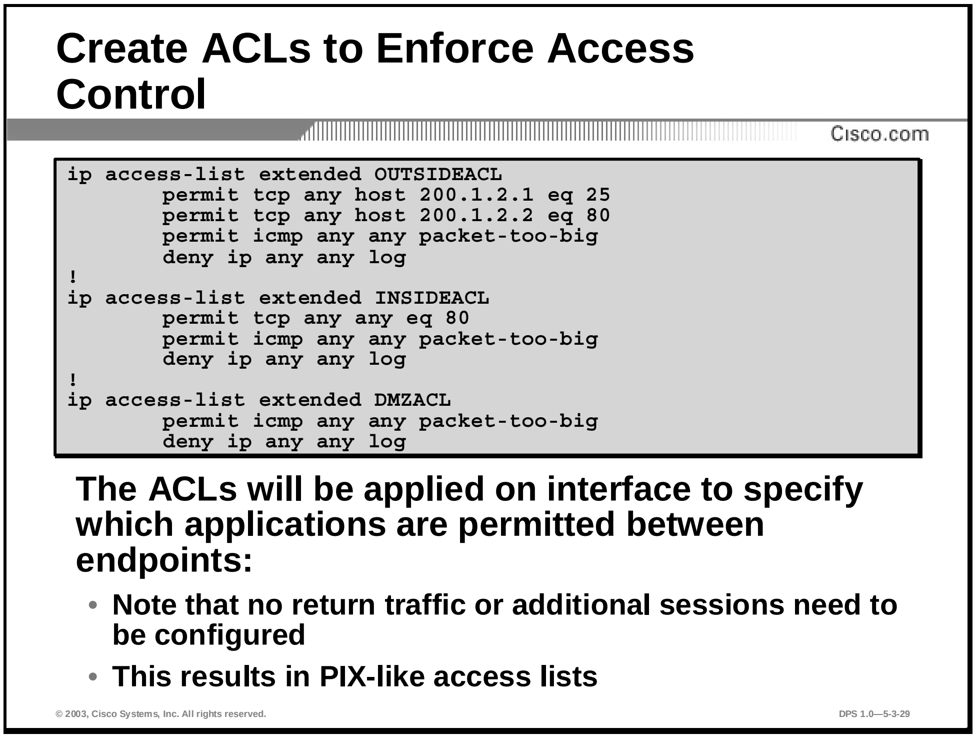
**The inspect rulesets specify, which application protocols need to be inspected on an interface:**

- **Generic TCP and UDP inspection are used for simple single-channel applications (telnet, DNS)**
- **Tip: Use TCP inspection for HTTP and SMTP to increase performance**

DPS 1.0—5-3-28

## FFS—First Step

Identify inspection rulesets that specify which application protocols need to be inspected on an interface. Generic TCP and UDP inspections are used for simple single-channel applications, for example Telnet and DNS.



## Create ACLs to Enforce Access Control

```
ip access-list extended OUTSIDEACL
        permit tcp any host 200.1.2.1 eq 25
        permit tcp any host 200.1.2.2 eq 80
        permit icmp any any packet-too-big
        deny ip any any log
!
ip access-list extended INSIDEACL
        permit tcp any any eq 80
        permit icmp any any packet-too-big
        deny ip any any log
!
ip access-list extended DMZACL
        permit icmp any any packet-too-big
        deny ip any any log
```

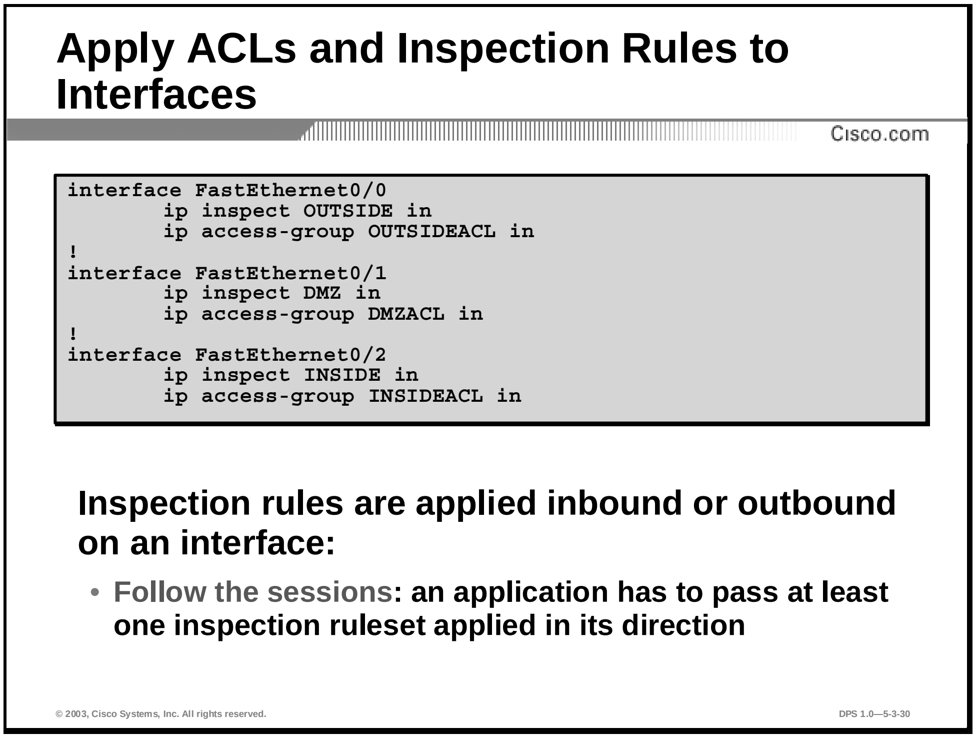
The ACLs will be applied on interface to specify which applications are permitted between endpoints:

- Note that no return traffic or additional sessions need to be configured
- This results in PIX-like access lists

DPS 1.0—5-3-29

## FFS—Second Step

The specified ACLs are applied on interfaces to specify which applications are permitted between endpoints. Note the advantage that no return traffic or additional sessions need to be configured—similar to PIX access lists.



## Apply ACLs and Inspection Rules to Interfaces

```
interface FastEthernet0/0
        ip inspect OUTSIDE in
        ip access-group OUTSIDEACL in
!
interface FastEthernet0/1
        ip inspect DMZ in
        ip access-group DMZACL in
!
interface FastEthernet0/2
        ip inspect INSIDE in
        ip access-group INSIDEACL in
```

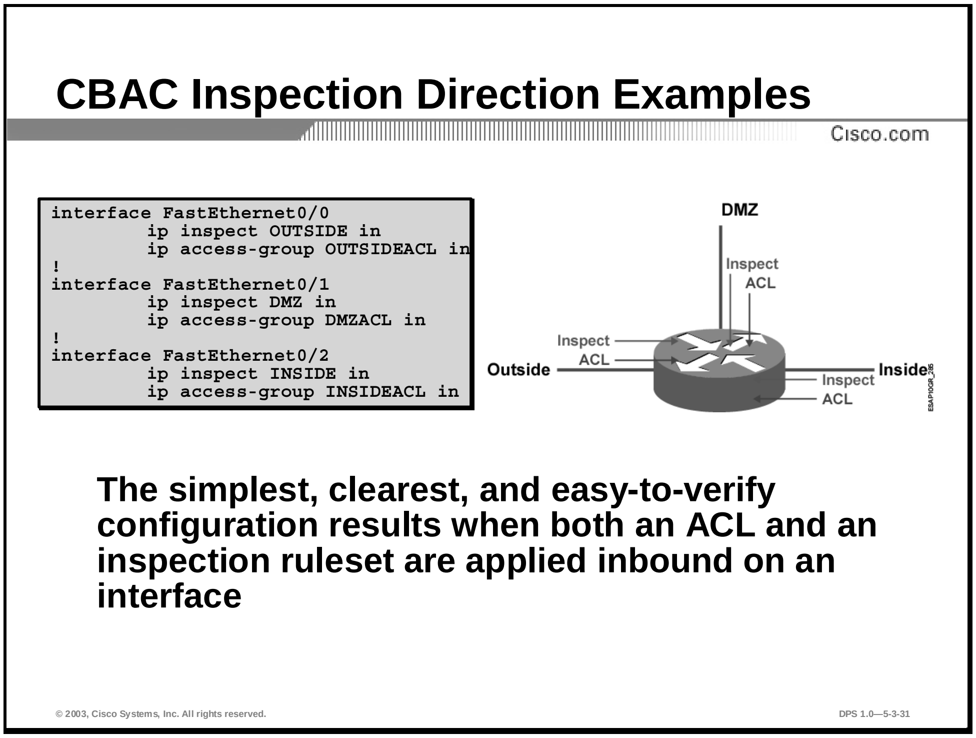
**Inspection rules are applied inbound or outbound on an interface:**

- **Follow the sessions: an application has to pass at least one inspection ruleset applied in its direction**

DPS 1.0—5-3-30

## FFS—Final Step

Finally, the previously defined inspection rules and ACLs are applied inbound or outbound on an interface.



## CBAC Inspection Direction Examples

Cisco.com

```
interface FastEthernet0/0
        ip inspect OUTSIDE in
        ip access-group OUTSIDEACL in
!
interface FastEthernet0/1
        ip inspect DMZ in
        ip access-group DMZACL in
!
interface FastEthernet0/2
        ip inspect INSIDE in
        ip access-group INSIDEACL in
```

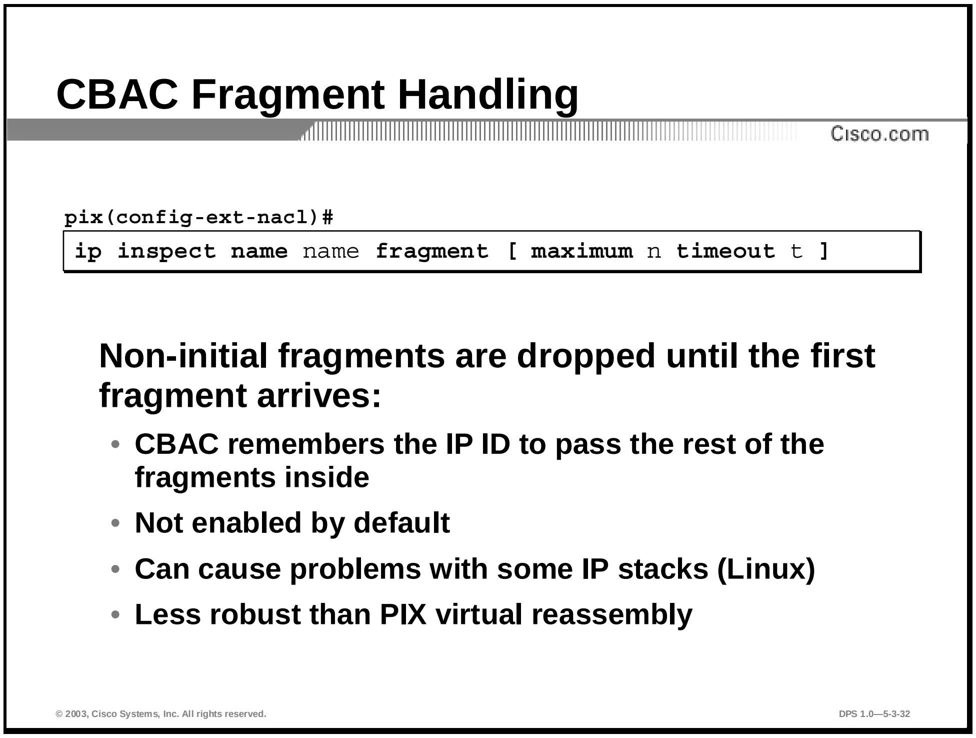
DMZ
Inspect
ACL
Inspect
ACL
Outside
Inside
Inspect
ACL
ESAP10GR_285

**The simplest, clearest, and easy-to-verify configuration results when both an ACL and an inspection ruleset are applied inbound on an interface**

DPS 1.0—5-3-31

# A Clean Solution

The combination of both an ACL and an inspection ruleset for each interface results in a very clear and simple solution. Additionally such CBAC configurations are easy to verify hence the chance to leave backdoors is minimized.



## CBAC Fragment Handling

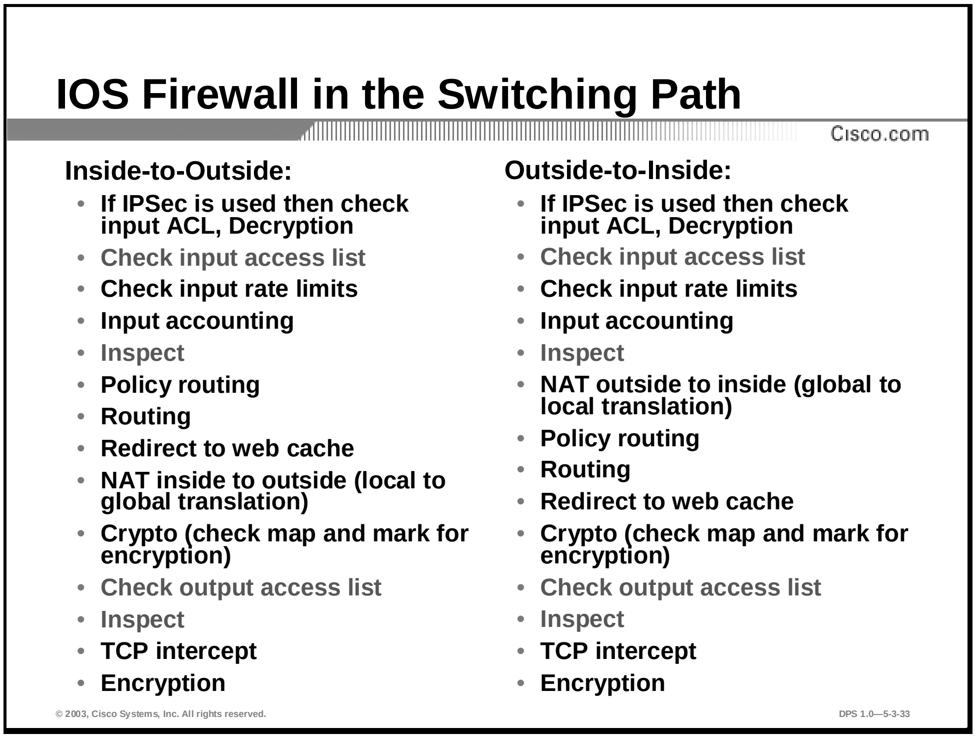
```
pix(config-ext-nacl)#
ip inspect name name fragment [ maximum n timeout t ]
```

**Non-initial fragments are dropped until the first fragment arrives:**

- **CBAC remembers the IP ID to pass the rest of the fragments inside**
- **Not enabled by default**
- **Can cause problems with some IP stacks (Linux)**
- **Less robust than PIX virtual reassembly**

## CBAC and Fragments

Since CBAC is context aware, non-initial fragments are dropped until the first fragment arrives. When the first fragment arrives, CBAC remembers the IP identification number to pass the rest of the fragments inside. However this is not enabled by default because some IP stacks—for example the Linux IP stack—cause problems with this handling. The best method is still the PIX virtual reassembly method.



### IOS Firewall in the Switching Path

**Inside-to-Outside:**

- If IPSec is used then check input ACL, Decryption
- Check input access list
- Check input rate limits
- Input accounting
- Inspect
- Policy routing
- Routing
- Redirect to web cache
- NAT inside to outside (local to global translation)
- Crypto (check map and mark for encryption)
- Check output access list
- Inspect
- TCP intercept
- Encryption

**Outside-to-Inside:**

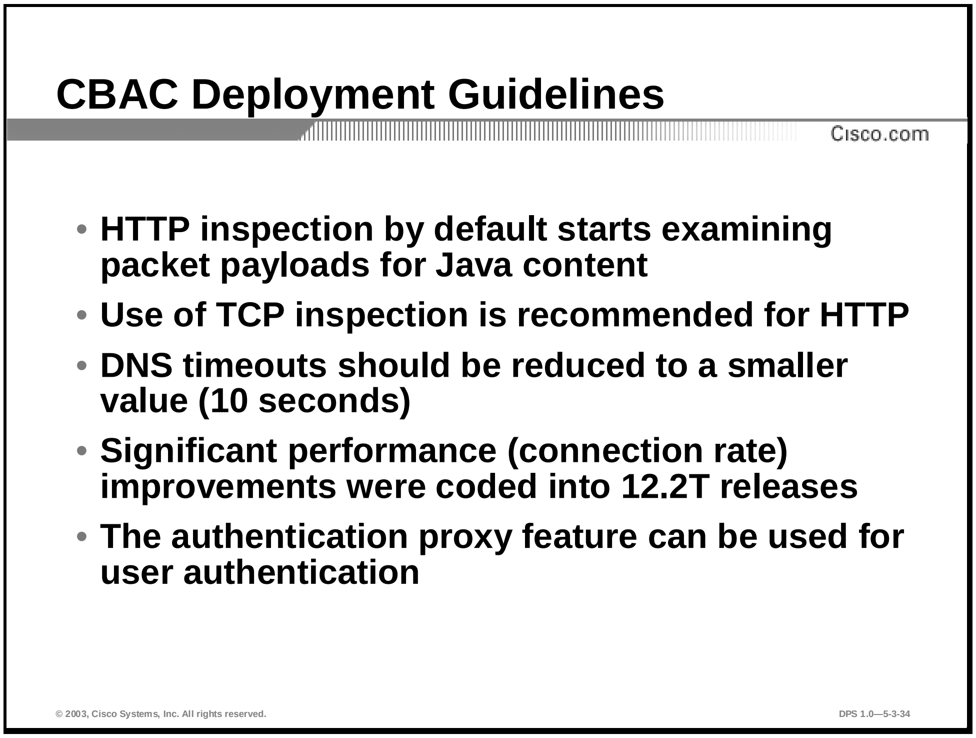
- If IPSec is used then check input ACL, Decryption
- Check input access list
- Check input rate limits
- Input accounting
- Inspect
- NAT outside to inside (global to local translation)
- Policy routing
- Routing
- Redirect to web cache
- Crypto (check map and mark for encryption)
- Check output access list
- Inspect
- TCP intercept
- Encryption

DPS 1.0—5-3-33

## Processing Order

The list above shows the IOS processing pipeline (switching path functions), which is applied to each packet. It is important to note that outside-to-inside flows are subject to ACL checks before NAT, therefore ACLs need to reference **global** addresses. Inspection (CBAC) is applied twice, but this property not have any configuration-related caveats.

In an IPSec environment (if IPSec tunnel mode is used), it is important to note that each packet undergoes ACL filtering twice – before and after IPSec encapsulation. It is therefore important that the interface ACLs permits both IPSec/IKE traffic, and the cleartext traffic after decapsulation.



## CBAC Deployment Guidelines

- HTTP inspection by default starts examining packet payloads for Java content
- Use of TCP inspection is recommended for HTTP
- DNS timeouts should be reduced to a smaller value (10 seconds)
- Significant performance (connection rate) improvements were coded into 12.2T releases
- The authentication proxy feature can be used for user authentication

DPS 1.0—5-3-34

# Guidelines for CBAC

The first inspection effort for HTTP concentrates on packet examination for Java content. Generally, it is recommended to implement TCP inspection for HTTP. In order to minimize the chance for spoofed DNS responses, the DNS timeout should be reduced to a smaller value, for example 10 seconds. When performance is critical use the IOS 12.2T releases. Finally the authentication proxy feature can be used for user authentication.

# Practice

Q1) When are access list entries created with CBAC?

A) when the first packet of a session crosses the router

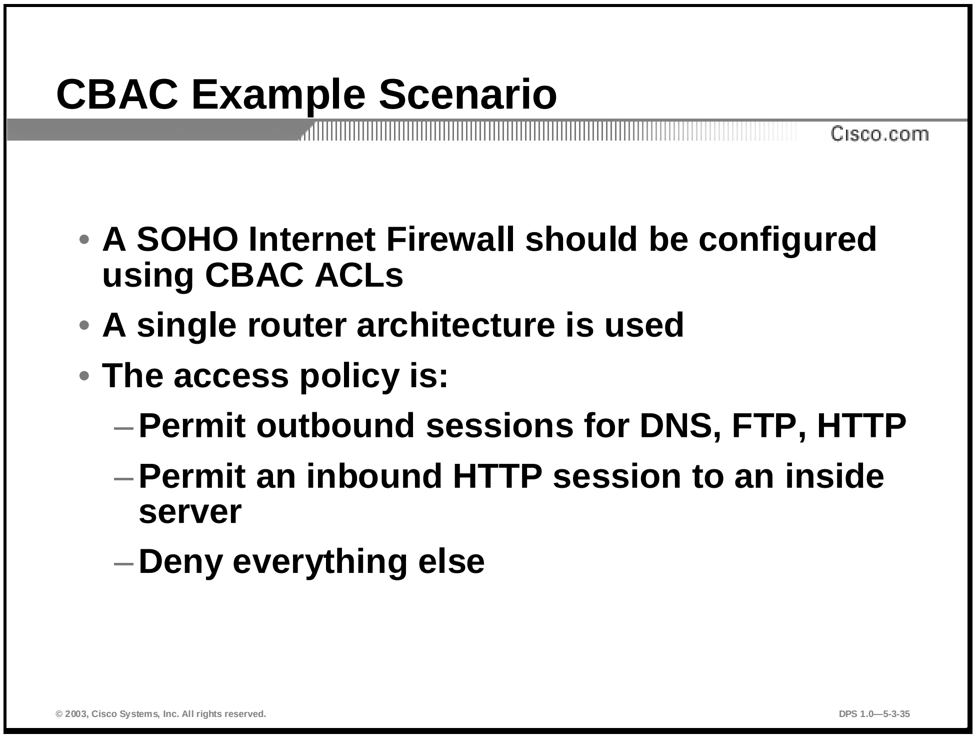
B) when the three-way handshake has completed

C) when the RST packet is seen

D) when the ACLs are configured

E) when inspect rules are created in the configuration

# Example Scenario



## CBAC Example Scenario

- A SOHO Internet Firewall should be configured using CBAC ACLs
- A single router architecture is used
- The access policy is:
  - Permit outbound sessions for DNS, FTP, HTTP
  - Permit an inbound HTTP session to an inside server
  - Deny everything else

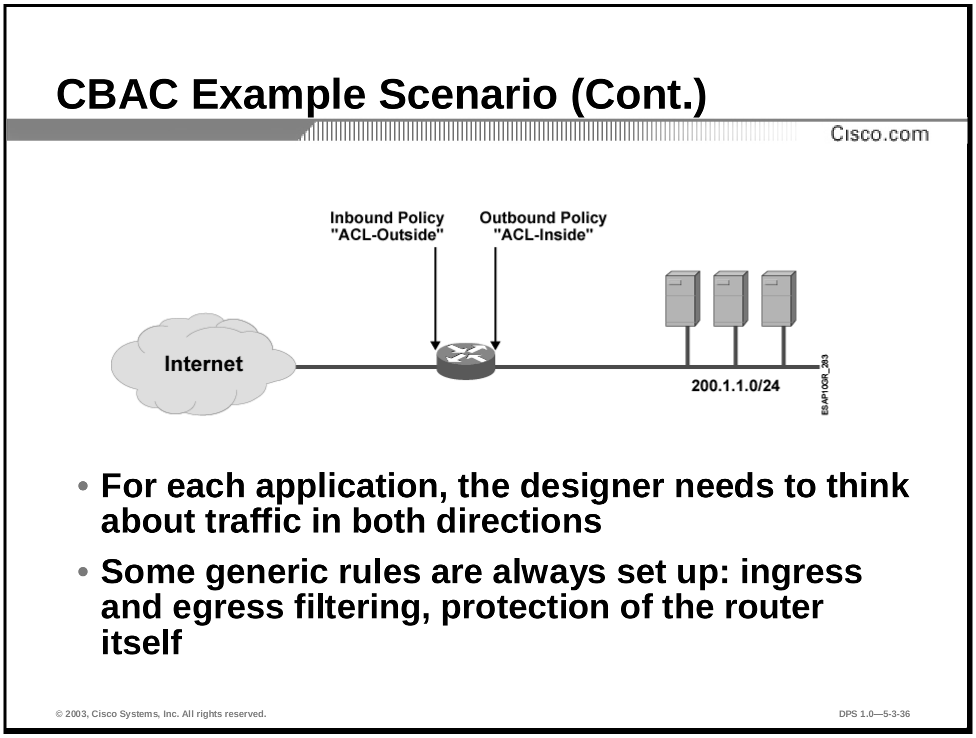
DPS 1.0—5-3-35

## Objective

This section will enable the learner to identify a correct configuration of CBAC features in a firewall implementation.

## Introduction

Suppose a SOHO Internet Firewall should be configured using CBAC and ACLs. Only a single router is used. The access policy includes: permit outbound sessions for DNS, FTP, HTTP, permit inbound HTTP sessions to an inside server, finally deny everything else.



## CBAC Example Scenario (Cont.)

Cisco.com

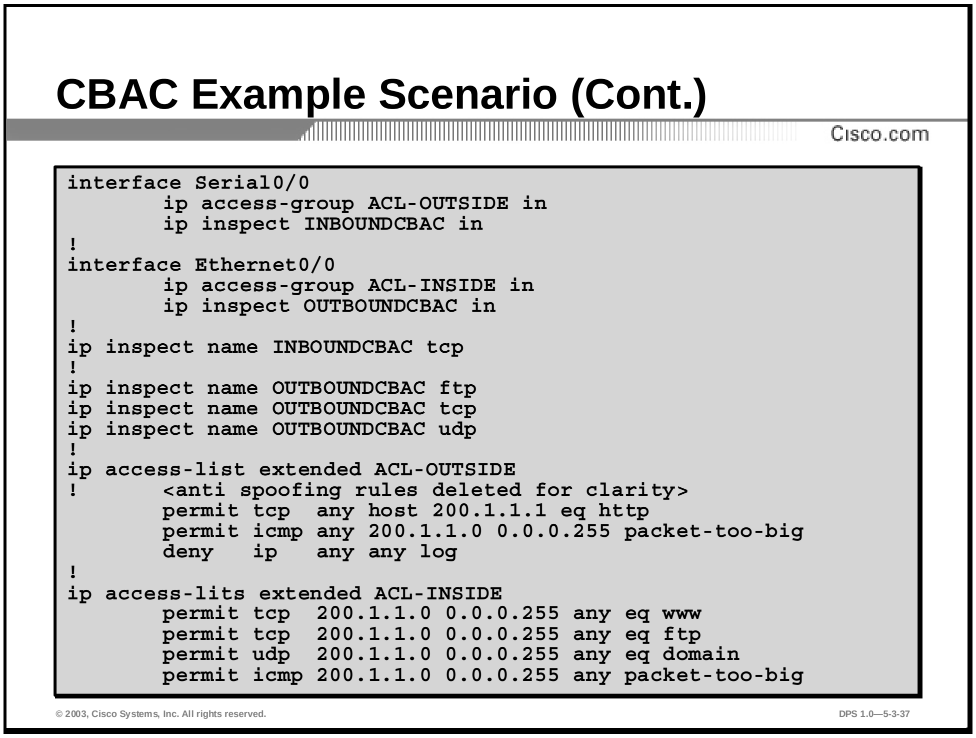
- For each application, the designer needs to think about traffic in both directions
- Some generic rules are always set up: ingress and egress filtering, protection of the router itself

DPS 1.0—5-3-36

# Example Scenario: Know Your Directions

Again the direction of traffic flows is fundamental for configuration considerations. Remember the general rules that are always set up, such as the ingress and egress filtering, and the protection of the router itself.

In this example, the outbound packet traffic will be filtered using the packet filtering ACL (ACL-Inside), applied inbound on the inside interface. All inbound packet traffic will be filtering using the packet filtering ACL (ACL-Outside) on the outside router interface. The outside ACL should also protect the router itself, by denying all, or permiting only select traffic to any router IP address from the untrusted network.



## CBAC Example Scenario (Cont.)

```
interface Serial0/0
        ip access-group ACL-OUTSIDE in
        ip inspect INBOUNDCBAC in
!
interface Ethernet0/0
        ip access-group ACL-INSIDE in
        ip inspect OUTBOUNDCBAC in
!
ip inspect name INBOUNDCBAC tcp
!
ip inspect name OUTBOUNDCBAC ftp
ip inspect name OUTBOUNDCBAC tcp
ip inspect name OUTBOUNDCBAC udp
!
ip access-list extended ACL-OUTSIDE
!       <anti spoofing rules deleted for clarity>
        permit tcp  any host 200.1.1.1 eq http
        permit icmp any 200.1.1.0 0.0.0.255 packet-too-big
        deny   ip   any any log
!
ip access-lits extended ACL-INSIDE
        permit tcp  200.1.1.0 0.0.0.255 any eq www
        permit tcp  200.1.1.0 0.0.0.255 any eq ftp
        permit udp  200.1.1.0 0.0.0.255 any eq domain
        permit icmp 200.1.1.0 0.0.0.255 any packet-too-big
```

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—5-3-37

## Example Scenario: Configuration

This figure illustrates a CBAC configuration example using both ACLs and inspects statements to define context-based access control.

The inspection rules are applied inbound on all interfaces. The ACLs follow the same concept, and decide, which traffic is to be accepted into an interface. As ACLs are now enjoying the support of a stateful engine (CBAC), PIX Firewall-like ACLs can be used, permitting only the first packet of an application, if the CBAC engine supports the application.

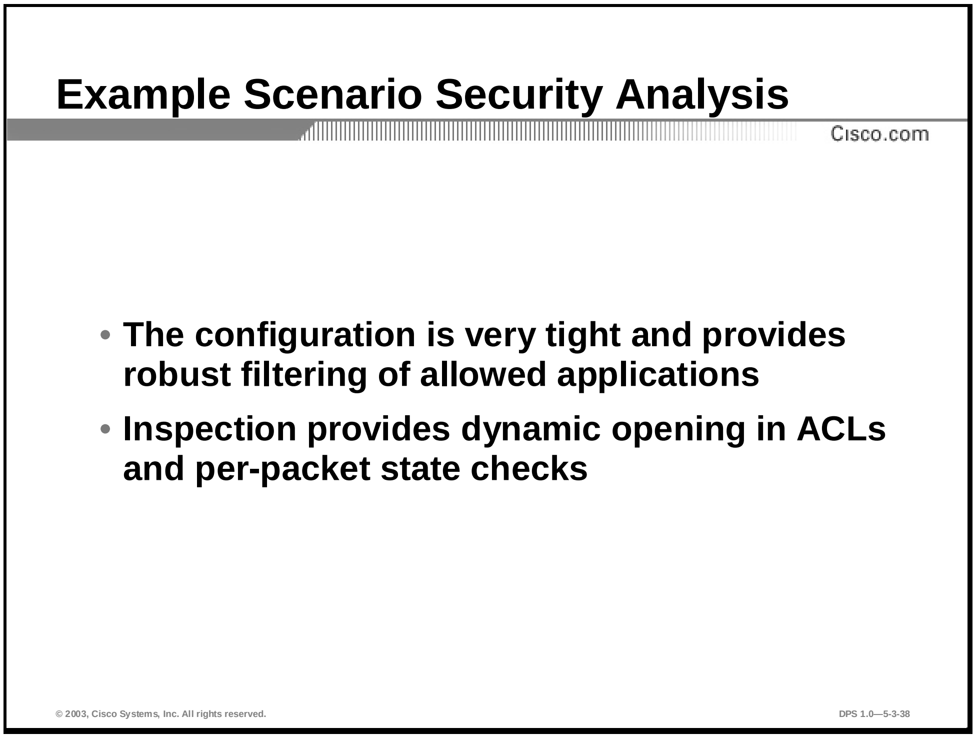
The ruleset ACL-OUTSIDE permitting inbound traffic (from the untrusted to the trusted network), applied inbound to the outside (untrusted) interface, needs to

- deny spoofed addresses using layer-3 filters
- permit inbound sessions to the internal HTTP server
- permit ICMP packet-too-big messages, needed for Path MTU Discovery to work

The ruleset ACL-INSIDE permitting outbound traffic (from the trusted to the untrusted network), applied inbound to the inside (trusted) interface needs to

- permit outgoing HTTP packets
- permit outgoing FTP packets
- permit outgoing DNS queries

- permit ICMP packet-too-big messages, needed for Path MTU Discovery to work



## Example Scenario Security Analysis

Cisco.com

- The configuration is very tight and provides robust filtering of allowed applications
- Inspection provides dynamic opening in ACLs and per-packet state checks

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—5-3-38

## Example Scenario: Analysis

After all the CBAC technique allows for best implementation of the given security rules. Furthermore, the configuration is very tight and provides a robust filtering of allowed applications.

The inspection is a powerful additional element that provides a dynamic opening in ACLs and per-packet state validation.

## Practice

Q1) Which of the following packets might be permitted through a CBAC-enabled interface, when TCP inspection is configured, and no previous session state is stored?

A) A TCP ACK packet

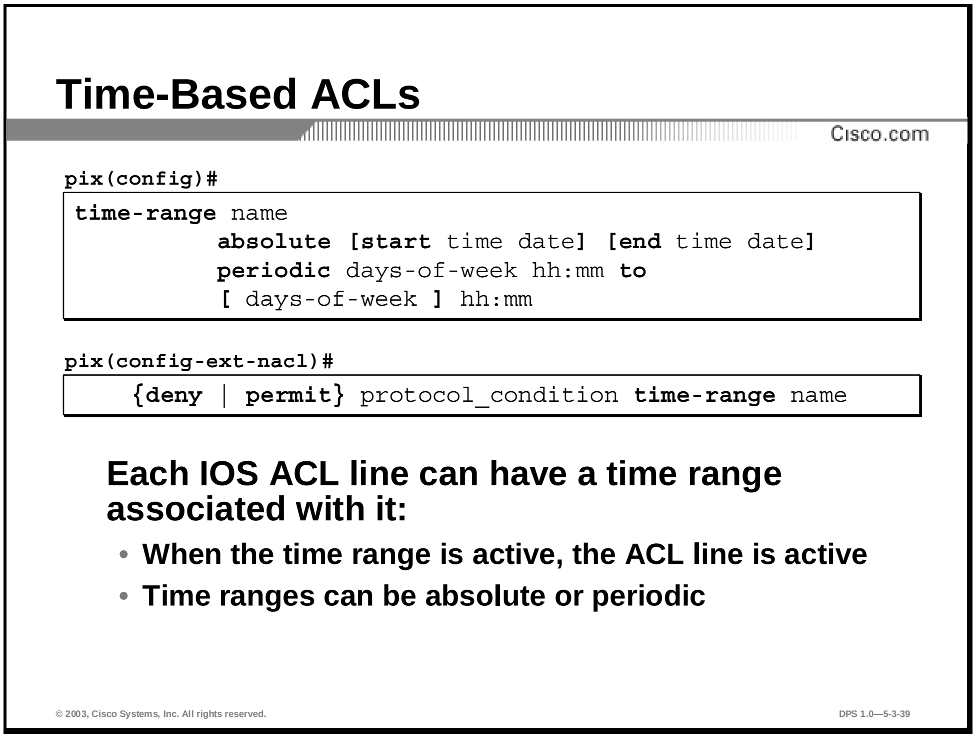
B) A TCP RST packet

C) A TCP FIN packet

D) A TCP SYN packet

E) A TCP NACK packet

# IOS Advanced Access Controls



## Time-Based ACLs

Cisco.com

```
pix(config)#
time-range name
          absolute [start time date] [end time date]
          periodic days-of-week hh:mm to
          [ days-of-week ] hh:mm
```

```
pix(config-ext-nacl)#
     {deny | permit} protocol_condition time-range name
```

**Each IOS ACL line can have a time range associated with it:**

- **When the time range is active, the ACL line is active**
- **Time ranges can be absolute or periodic**

DPS 1.0—5-3-39

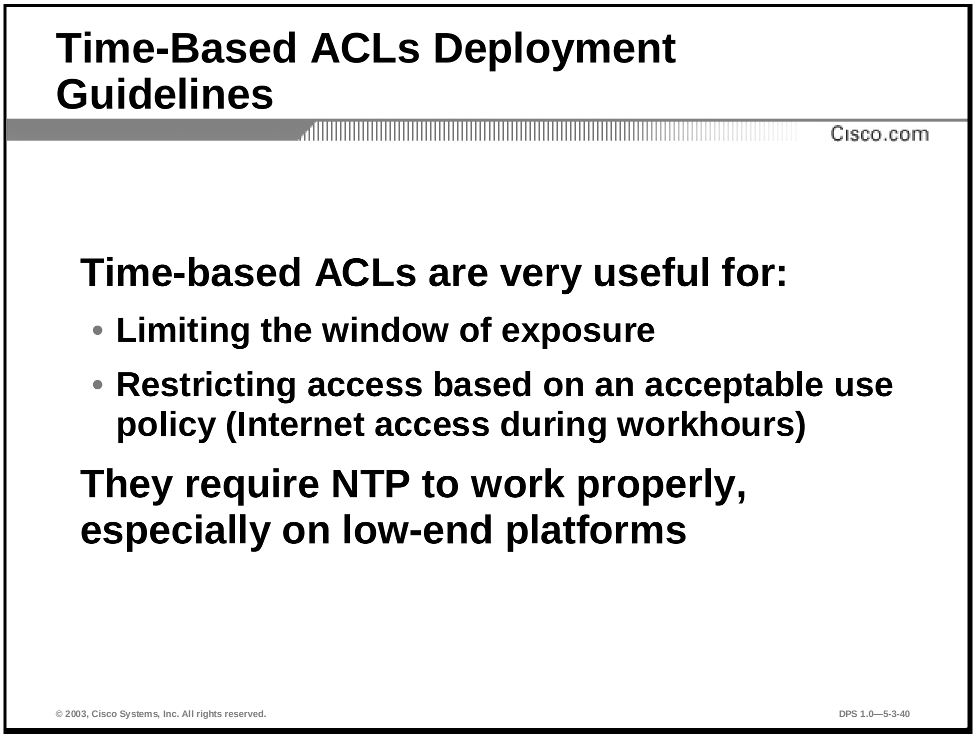
## Objective

This section will enable the learner to explain the features and limitations of other advanced Cisco IOS access control mechanisms and explain how they apply to firewall design.

## Introduction

The IOS provides additional advanced access controls, introducing time relation, unicast reverse path forwarding and policy routing. This section explains these important functions that are necessary to address modern security threats.

Each IOS ACL line can have assigned a time range of validity. Only when the time range is active, the ACL line is processed. Time ranges can be specified for both one-time or periodic events.



## Time-Based ACLs Deployment Guidelines

Cisco.com

Time-based ACLs are very useful for:

- Limiting the window of exposure
- Restricting access based on an acceptable use policy (Internet access during workhours)

They require NTP to work properly, especially on low-end platforms

© 2003, Cisco Systems, Inc. All rights reserved.

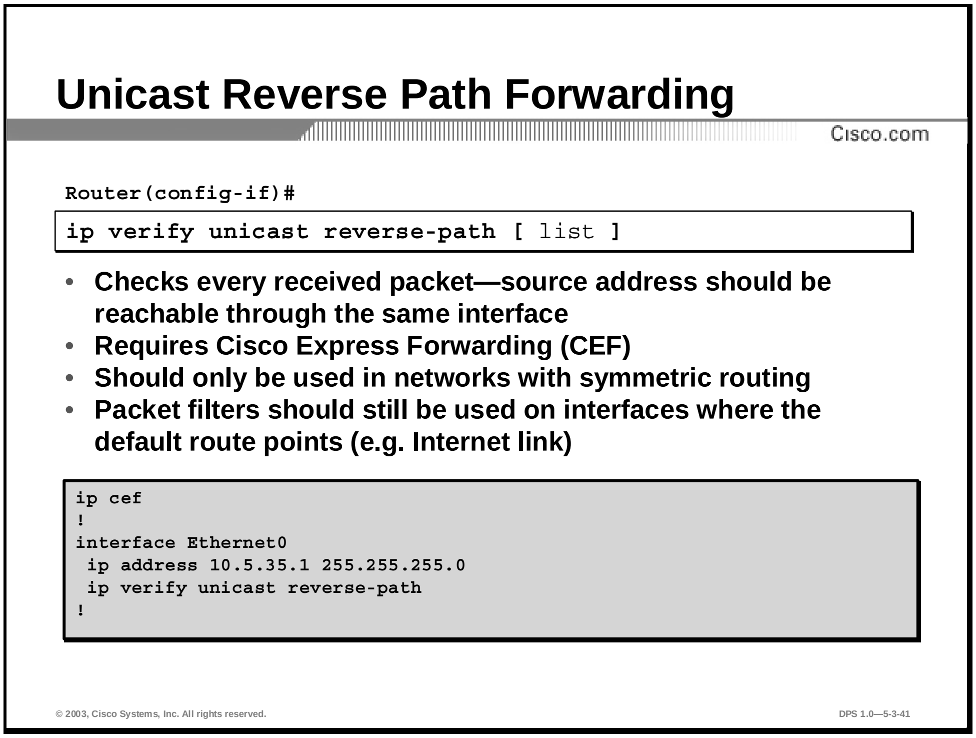
DPS 1.0—5-3-40

## Why Time-Based ACLs?

In many cases it is not necessary to permit certain traffic all day. Hence the time “window of exposure” can be reduced to a minimal duration in order to reduce the probability of an attack.

For example, FTP and Telnet traffic might be allowed during work hours only.

When configuring time-based ACLs the easiest and most scalable way to assure proper time accuracy is to use the network time protocol (NTP), especially on low-end platforms, which do not have a real-time clock.



## Unicast Reverse Path Forwarding

```
Router(config-if)#
```

```
ip verify unicast reverse-path [ list ]
```

- Checks every received packet—source address should be reachable through the same interface
- Requires Cisco Express Forwarding (CEF)
- Should only be used in networks with symmetric routing
- Packet filters should still be used on interfaces where the default route points (e.g. Internet link)

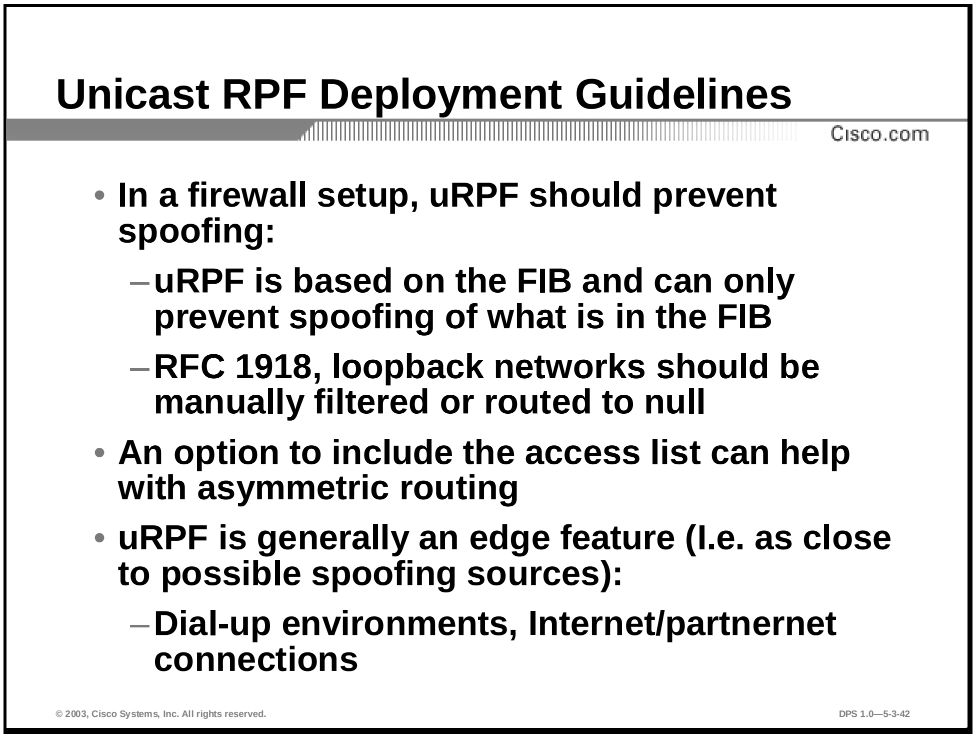
```
ip cef
!
interface Ethernet0
 ip address 10.5.35.1 255.255.255.0
 ip verify unicast reverse-path
!
```

DPS 1.0—5-3-41

## Prevent IP Spoofing

The Reverse Path Forwarding (RPF) method is a well-known simple algorithm used with IP multicast for example but should also be used to prevent IP packets having a forged IP source address. The idea is simple: Only forward a packet whose source IP address can be reached over the interface on which this packet had been received. As this security related RPF variant is used with unicast packets it is called “Unicast RPF” (uRPF) to emphasize the difference to IP multicast applications.

Note that unicast RPF requires Cisco Express Forwarding (CEF) and should only be used in networks with symmetric routing.



## Unicast RPF Deployment Guidelines

Cisco.com

- In a firewall setup, uRPF should prevent spoofing:
  - uRPF is based on the FIB and can only prevent spoofing of what is in the FIB
  - RFC 1918, loopback networks should be manually filtered or routed to null
- An option to include the access list can help with asymmetric routing
- uRPF is generally an edge feature (I.e. as close to possible spoofing sources):
  - Dial-up environments, Internet/partnernet connections

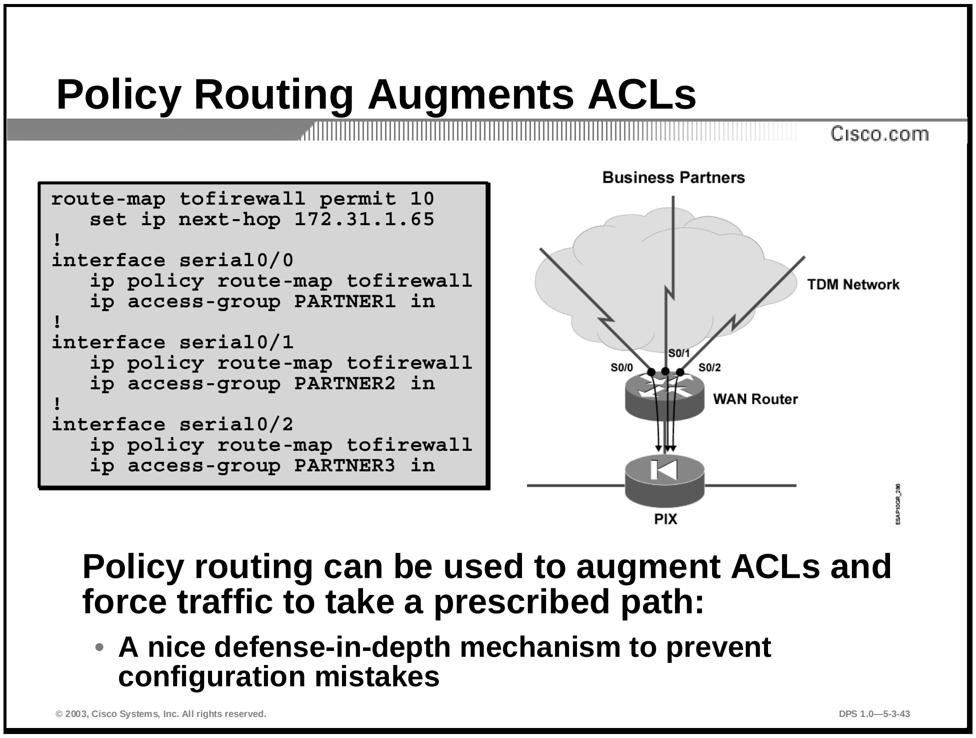
DPS 1.0—5-3-42

# Guidelines for uRPF

Unicast RPF is used in firewall environments to prevent IP address spoofing, which can be an indication of a DoS attack. Note that uRPF relies on the Forward Information Base to determine if the source address is valid or not.

Additionally, RFC 1918 addresses and loopback networks should be manually filtered or routed to the null interface.

Note that uRPF is generally an edge feature as this functionality should be close to possible spoofing sources. Practically, uRPF is configured on dial-up environments and Internet/partnernet connections.



## Policy Routing Augments ACLs

Cisco.com

```
route-map tofirewall permit 10
   set ip next-hop 172.31.1.65
!
interface serial0/0
   ip policy route-map tofirewall
   ip access-group PARTNER1 in
!
interface serial0/1
   ip policy route-map tofirewall
   ip access-group PARTNER2 in
!
interface serial0/2
   ip policy route-map tofirewall
   ip access-group PARTNER3 in
```

Business Partners
TDM Network
S0/1
S0/0
S0/2
WAN Router
PIX
ESAP10GR_288

**Policy routing can be used to augment ACLs and force traffic to take a prescribed path:**

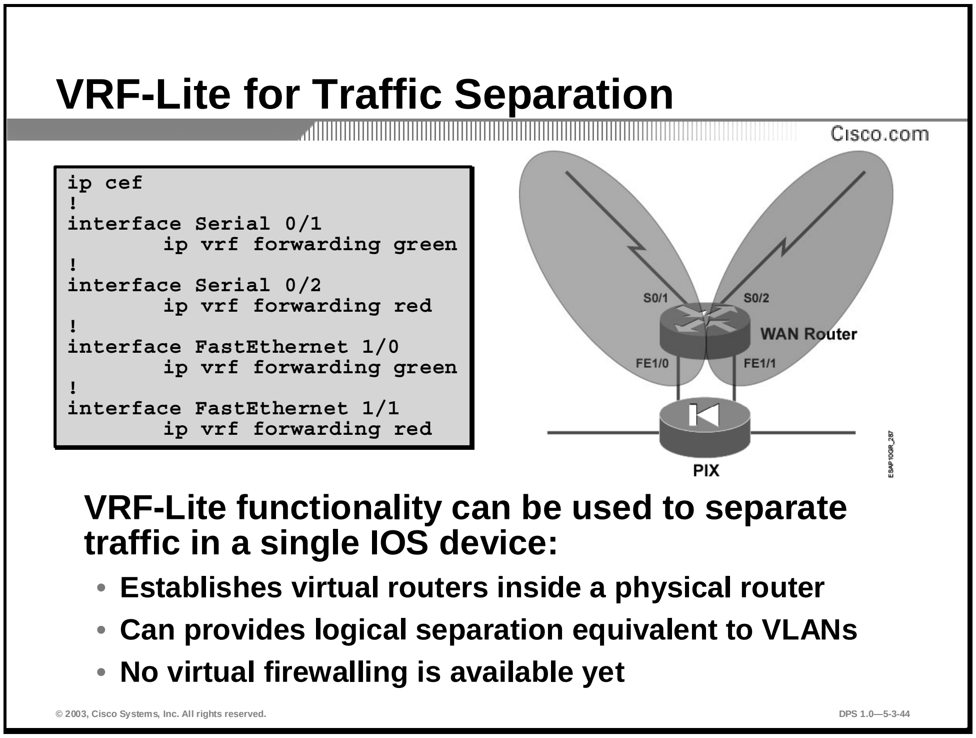
- **A nice defense-in-depth mechanism to prevent configuration mistakes**

DPS 1.0—5-3-43

## Add Policy Routing

Using policy routing commands the classic ACL capabilities can be greatly augmented as traffic can be forced to take a predefined path. This example shows two layers of traffic separation. ACLs and policy routing work in concert to prevent traffic flowing between business partners.

Policy routing is a simple and nice defense-in-depth mechanism to prevent configuration mistakes.



## VRF-Lite for Traffic Separation

Cisco.com

```
ip cef
!
interface Serial 0/1
       ip vrf forwarding green
!
interface Serial 0/2
       ip vrf forwarding red
!
interface FastEthernet 1/0
       ip vrf forwarding green
!
interface FastEthernet 1/1
       ip vrf forwarding red
```

**VRF-Lite functionality can be used to separate traffic in a single IOS device:**

- **Establishes virtual routers inside a physical router**
- **Can provides logical separation equivalent to VLANs**
- **No virtual firewalling is available yet**

DPS 1.0—5-3-44

## Traffic Separation

Proven VPN technology such as the VRF-Lite (VPN Routing and Forwarding Instance) can be used to separate traffic in a single IOS device. VRF-Lite established virtual routers inside a physical router, so the logical separation can be compared to VLAN separation inside a switch. However, there is no support in Cisco IOS for virtual firewalling at the moment.

## Practice

Q1) What can VRF-lite-based separation be compared to security-wise?

A) IPSec traffic separation

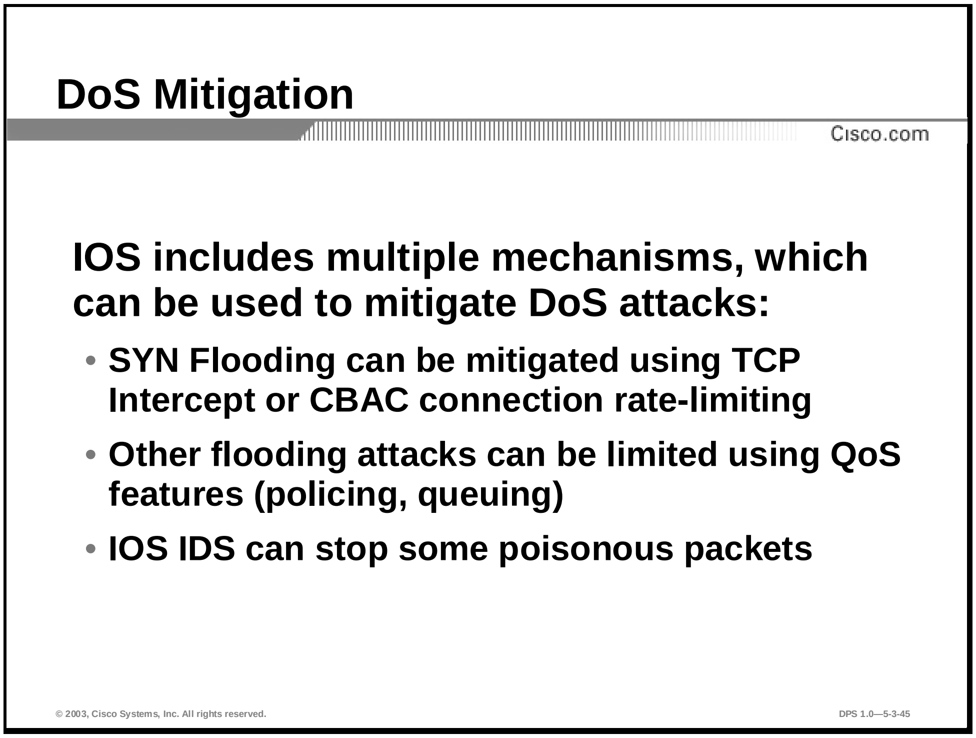
B) VLANs

C) stateful firewalling

D) classic packet filtering

E) switch port security

# IOS Protection Against Denial-of-Service



## DoS Mitigation

Cisco.com

**IOS includes multiple mechanisms, which can be used to mitigate DoS attacks:**

- **SYN Flooding can be mitigated using TCP Intercept or CBAC connection rate-limiting**
- **Other flooding attacks can be limited using QoS features (policing, queuing)**
- **IOS IDS can stop some poisonous packets**

DPS 1.0—5-3-45

## Objective

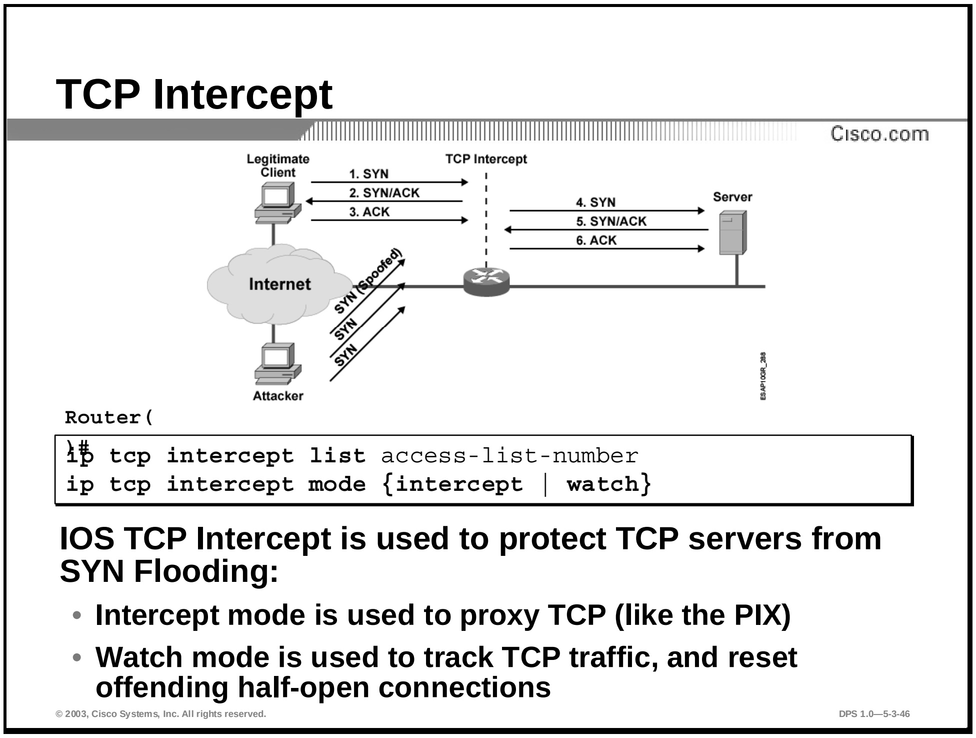
This section will enable the learner to identify and configure TCP Intercept and quality-of-service features, which can be used in the context of perimeter security design.

## Introduction

Denial of Service (Dos) attacks are the most common and most dangerous sort of attacks encountered in the Internet. Because of this, the Cisco IOS provides dedicated mechanisms to prevent DoS and even Distributed DoS attacks. These important add-ons are presented in this section.

Simple SYN-Flooding can be mitigated using TCP Intercept or CBAC connection rate-limiting. Thus the rate of connection establishments is limited and even DDoS is mitigated.

Generally, flooding attacks as it is used with DDoS can be limited using QoS features, including special policies and efficient class-based and fair queuing. The IOS Intrusion Detection System (IDS) can stop some poisonous packets.



## TCP Intercept

```
Router(
)#
ip tcp intercept list access-list-number
ip tcp intercept mode {intercept | watch}
```

**IOS TCP Intercept is used to protect TCP servers from SYN Flooding:**

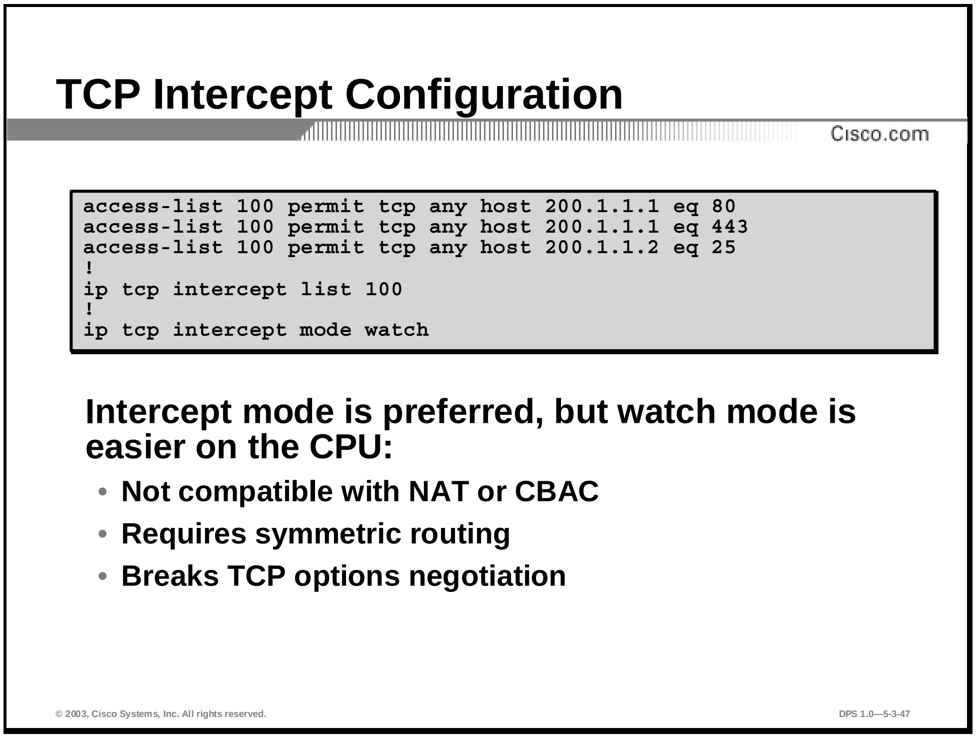
- **Intercept mode is used to proxy TCP (like the PIX)**
- **Watch mode is used to track TCP traffic, and reset offending half-open connections**

DPS 1.0—5-3-46

# TCP Intercept

Cisco IOS has a TCP Intercept capability that is designed to combat SYN Flooding. When used in intercept mode (the default setting) it checks for incoming TCP connection requests and will proxy-answer on behalf of the destination server to ensure that the request is valid before then connecting to the server. Once TCP Intercept has established a genuine connection with the client and the server, it then merges these two connections into a single source-destination session. It offers a zero window to the client to prevent it from sending data until the server sends a window offer back. In the case of bogus requests, its use of aggressive time-outs on half-open connections and support of threshold levels for both the number of outstanding and incoming rate of TCP connection requests, protect servers while still allowing valid requests through. The PIX Firewall implements a similar feature using SYN Cookies.

An alternate “watch” mode is used to track the TCP traffic: if half-opened TCP connections are detected, they are reset if they do not complete in a reasonable amount of time.



## TCP Intercept Configuration

```
access-list 100 permit tcp any host 200.1.1.1 eq 80
access-list 100 permit tcp any host 200.1.1.1 eq 443
access-list 100 permit tcp any host 200.1.1.2 eq 25
!
ip tcp intercept list 100
!
ip tcp intercept mode watch
```

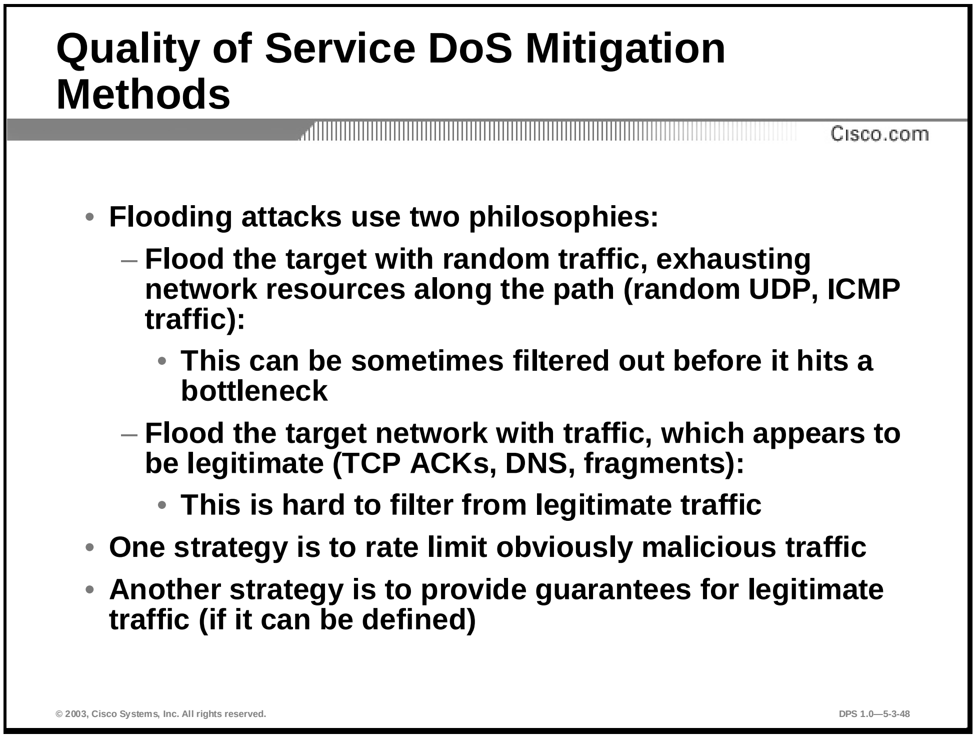
**Intercept mode is preferred, but watch mode is easier on the CPU:**

- **Not compatible with NAT or CBAC**
- **Requires symmetric routing**
- **Breaks TCP options negotiation**

# Example

This figure illustrates a TCP Intercept configuration example. Note that the TCP intercept mode is more powerful but on the other hand the watch mode is less demanding for the CPU.

TCP intercept is not compatible with NAT or CBAC and strictly requires symmetric routing. Since the TCP stream is split into two independent parts, the TCP option negotiation is broken.



## Quality of Service DoS Mitigation Methods

Cisco.com

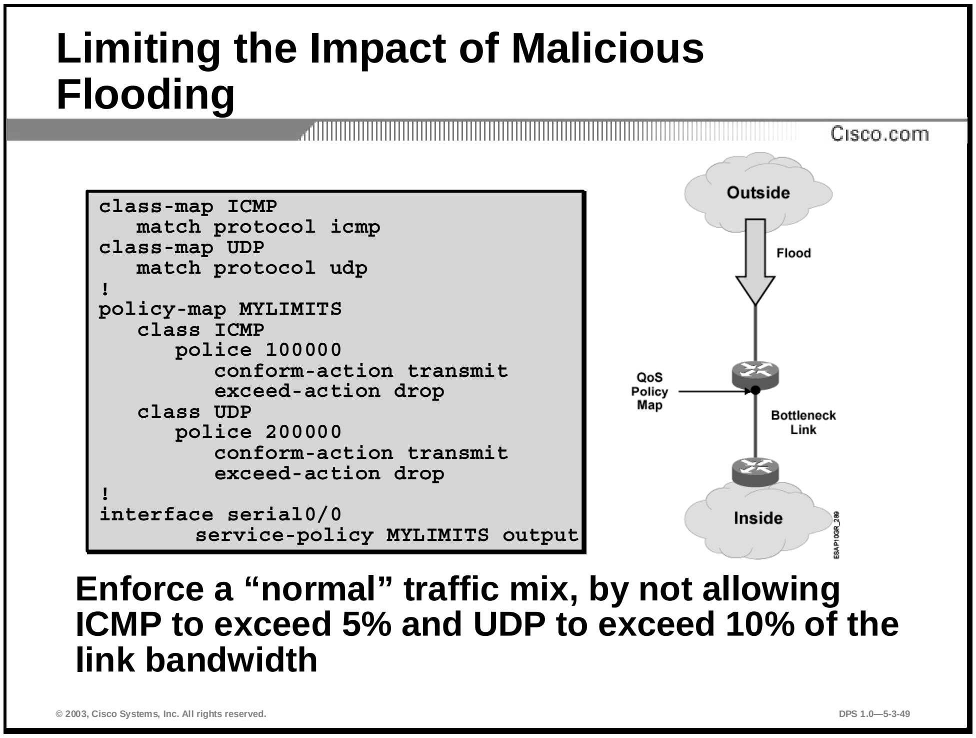
- Flooding attacks use two philosophies:
  - Flood the target with random traffic, exhausting network resources along the path (random UDP, ICMP traffic):
    - This can be sometimes filtered out before it hits a bottleneck
  - Flood the target network with traffic, which appears to be legitimate (TCP ACKs, DNS, fragments):
    - This is hard to filter from legitimate traffic
- One strategy is to rate limit obviously malicious traffic
- Another strategy is to provide guarantees for legitimate traffic (if it can be defined)

DPS 1.0—5-3-48

## QoS Against DoS

Since DoS is typically realized with flooding attacks, simple QoS queuing rules such as priority-, class-based, or even fair-queuing prevents that the network does not fully congest.

Another strategy is to rate limit obviously malicious traffic.



## Limiting the Impact of Malicious Flooding

Cisco.com

```
class-map ICMP
   match protocol icmp
class-map UDP
   match protocol udp
!
policy-map MYLIMITS
   class ICMP
      police 100000
         conform-action transmit
         exceed-action drop
   class UDP
      police 200000
         conform-action transmit
         exceed-action drop
!
interface serial0/0
        service-policy MYLIMITS output
```

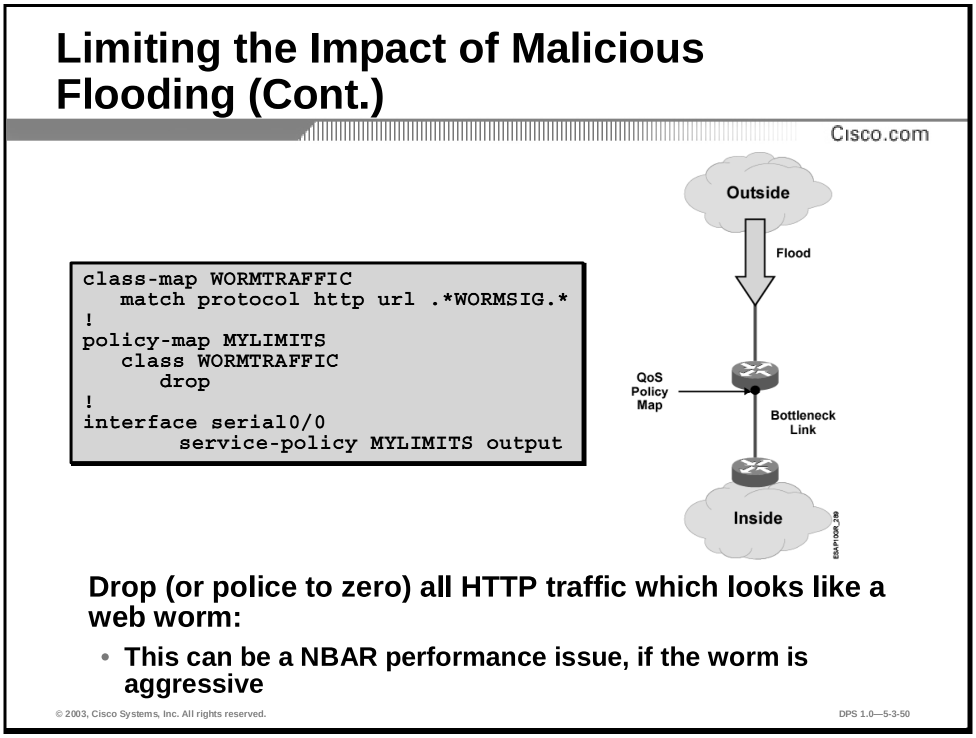
**Enforce a “normal” traffic mix, by not allowing ICMP to exceed 5% and UDP to exceed 10% of the link bandwidth**

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—5-3-49

## Define a Traffic Mix

Using a QoS policy map a “normal” traffic mix can be configured, so that each traffic class is not allowed to possess more than an allocated share of the total queuing resources. This is accomplished through rate limiting using policing.



## Limiting the Impact of Malicious Flooding (Cont.)

Cisco.com

```
class-map WORMTRAFFIC
   match protocol http url .*WORMSIG.*
!
policy-map MYLIMITS
   class WORMTRAFFIC
      drop
!
interface serial0/0
        service-policy MYLIMITS output
```

**Drop (or police to zero) all HTTP traffic which looks like a web worm:**

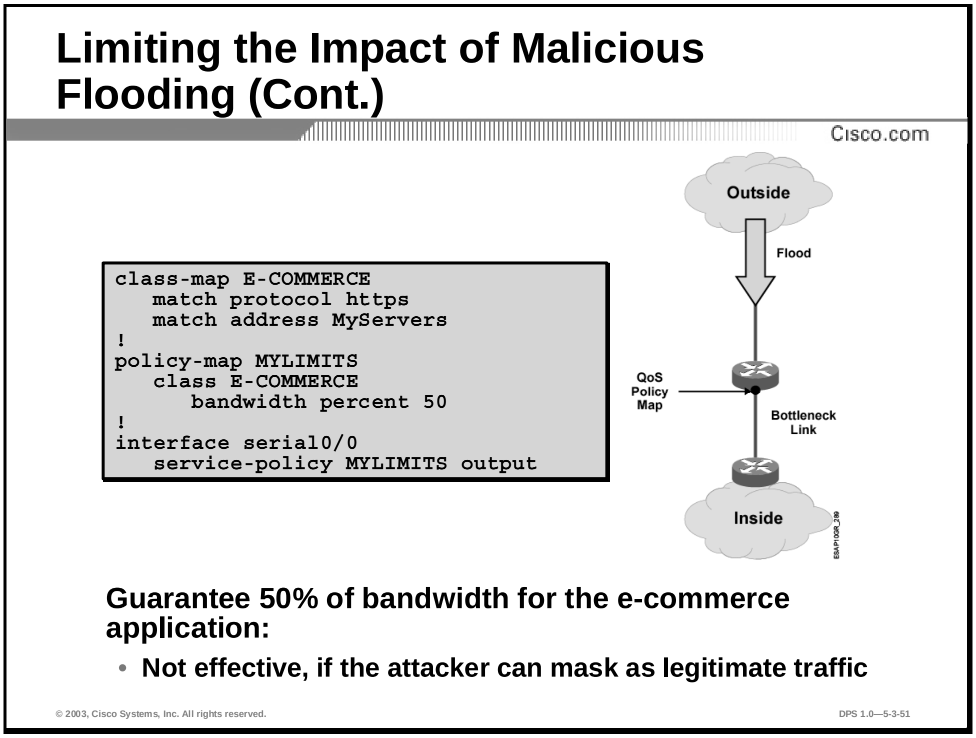
- **This can be a NBAR performance issue, if the worm is aggressive**

DPS 1.0—5-3-50

## Scan for Malicious Traffic

Applying strict policy rules for malicious traffic is another possibility. In this example the router is configured to examine all HTTP traffic that looks like a web worm, that is having a typical worm signature.

This approach requires an up-to-date signature database.



## Limiting the Impact of Malicious Flooding (Cont.)

Cisco.com

```
class-map E-COMMERCE
   match protocol https
   match address MyServers
!
policy-map MYLIMITS
   class E-COMMERCE
      bandwidth percent 50
!
interface serial0/0
   service-policy MYLIMITS output
```

**Guarantee 50% of bandwidth for the e-commerce application:**

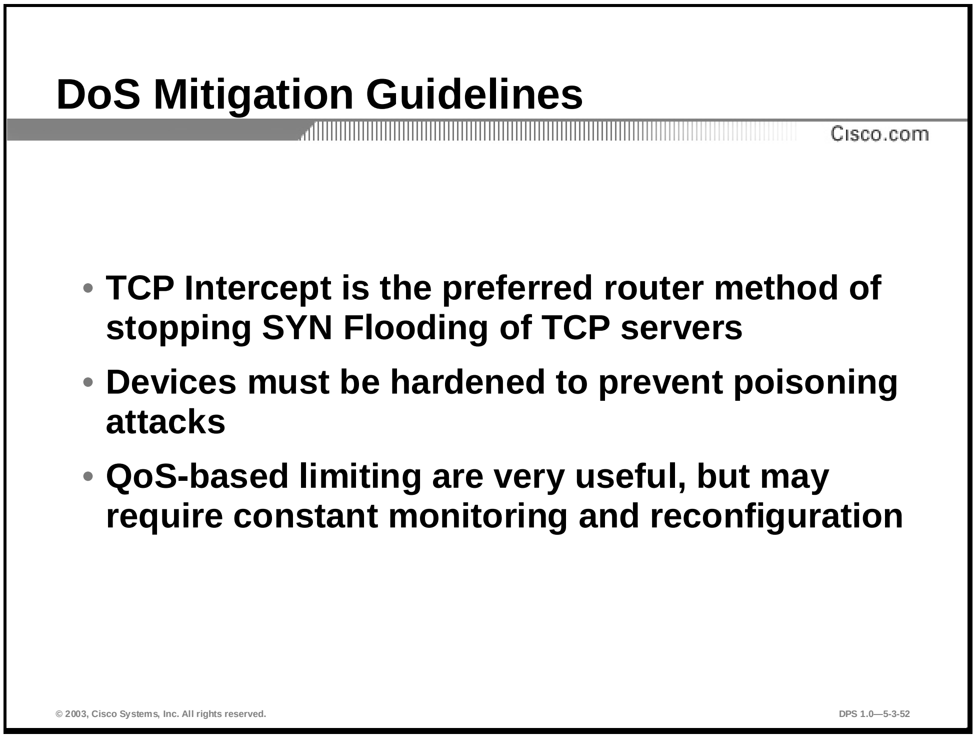
- **Not effective, if the attacker can mask as legitimate traffic**

DPS 1.0—5-3-51

## Using a Bandwidth Guaranteeing Rule

If you want to assure a certain bandwidth to your customers, you might want to configure a policy-map that guarantees 50% of the bandwidth for e-commerce applications.

However, this approach fails if the attacker simulates this sort of traffic.



## DoS Mitigation Guidelines

Cisco.com

- TCP Intercept is the preferred router method of stopping SYN Flooding of TCP servers
- Devices must be hardened to prevent poisoning attacks
- QoS-based limiting are very useful, but may require constant monitoring and reconfiguration

© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—5-3-52

# Guidelines

The most preferred method to stop SYN flooding attacks to TCP servers is to use the IOS TCP Intercept mechanism.

Additionally, all devices must be hardened to prevent poisoning attacks. Hence check for newest security exploits and apply software updates frequently.

Limiting a flooding impact with QoS methods requires a constant monitoring and reconfiguration. This should only be regarded as additional method.

# Practice

Q1) Which TCP Intercept mode has higher performance?

A) Intercept mode

B) Stealth mode

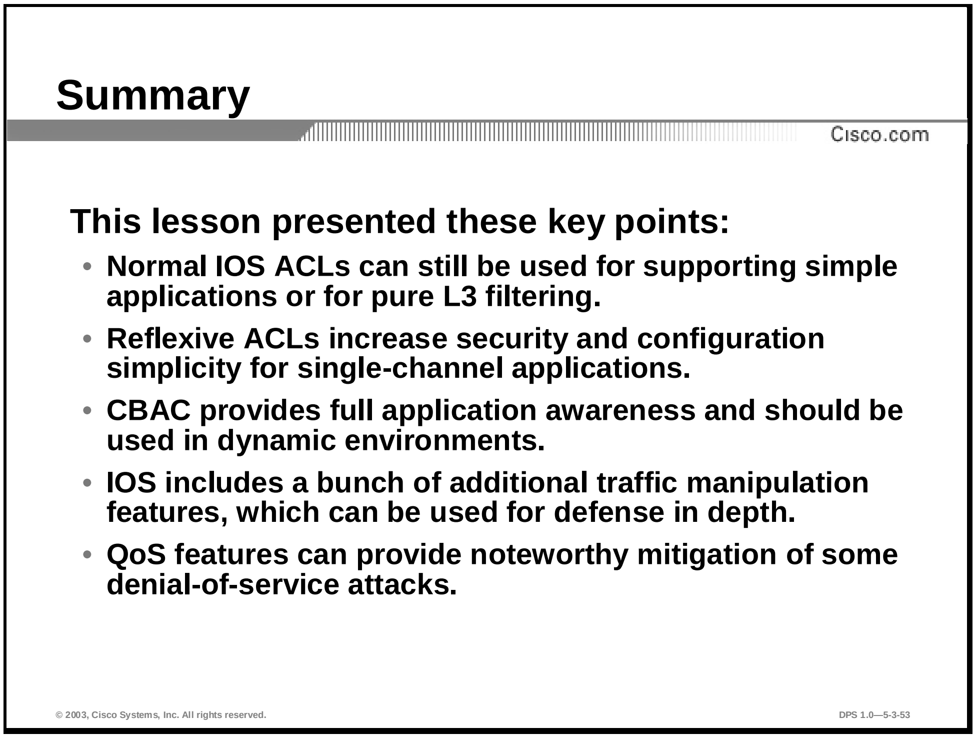
C) Proxy mode

D) Watch mode

E) Main mode

# Summary

This section summarizes the key points discussed in this lesson.



## Summary

Cisco.com

This lesson presented these key points:

- Normal IOS ACLs can still be used for supporting simple applications or for pure L3 filtering.
- Reflexive ACLs increase security and configuration simplicity for single-channel applications.
- CBAC provides full application awareness and should be used in dynamic environments.
- IOS includes a bunch of additional traffic manipulation features, which can be used for defense in depth.
- QoS features can provide noteworthy mitigation of some denial-of-service attacks.

DPS 1.0—5-3-53

## Next Steps

After completing this lesson, go to:

- Content Engines lesson

# Quiz: Cisco IOS Software Access Control Features

Complete the quiz to assess what you have learned in this lesson.

## Objectives

This quiz tests your knowledge on how to:

- Identify advanced security features and limitations of the Cisco IOS software when using it in a firewall system design

## Instructions

Answer these questions:

1. What does the performance impact of classic ACLs depend on?
2. How do reflexive ACLs increase the security of UDP filtering?
3. In which direction must CBAC inspection rules be applied?
4. What are the QoS-based strategies for mitigating flooding attacks?

## Scoring

You have successfully completed the quiz for this lesson when you earn a score of 80 percent or better.

# Content Engines

## Overview

### Importance

Content engines are relatively new to the Internet and promise an improved performance for the application layer. Since many enterprises already integrate content engines in their network it is important to understand how they can be used in their firewall architectures to improve security and what content engines allow for authentication.

### Lesson Objective

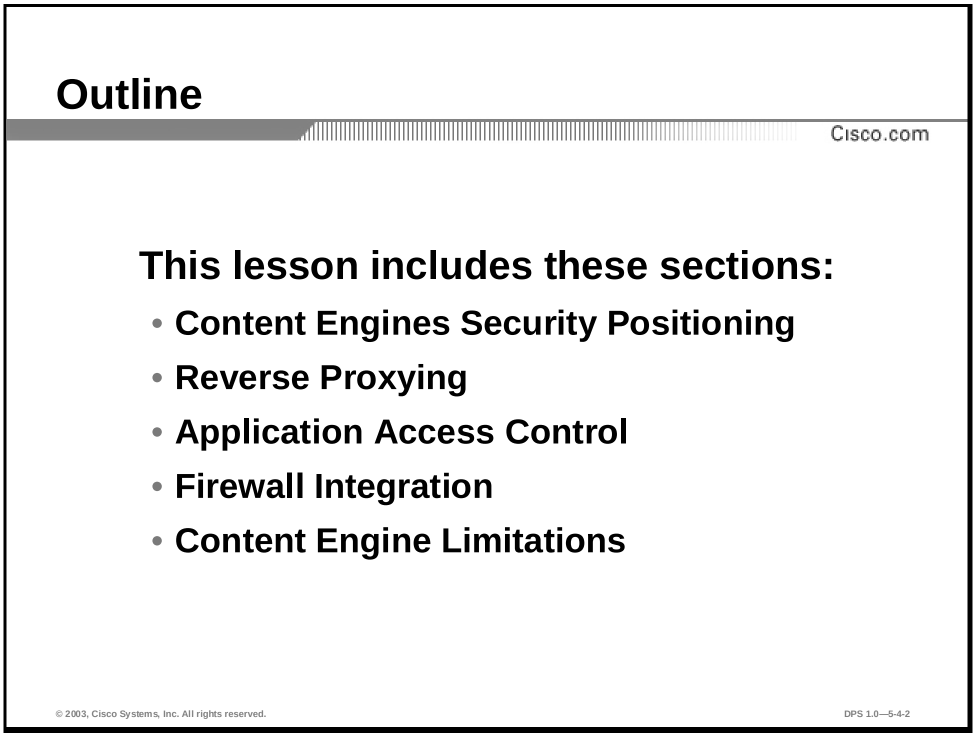
This lesson will enable the learner to identify security features and limitations of Content Engine products, when using them in a firewall system design.

## Learner Skills and Knowledge

To fully benefit from this lesson, you must have these prerequisite skills and knowledge:

- Have solid knowledge about content switching principles
- Have basic knowledge about firewall architectures and NAT

## Outline



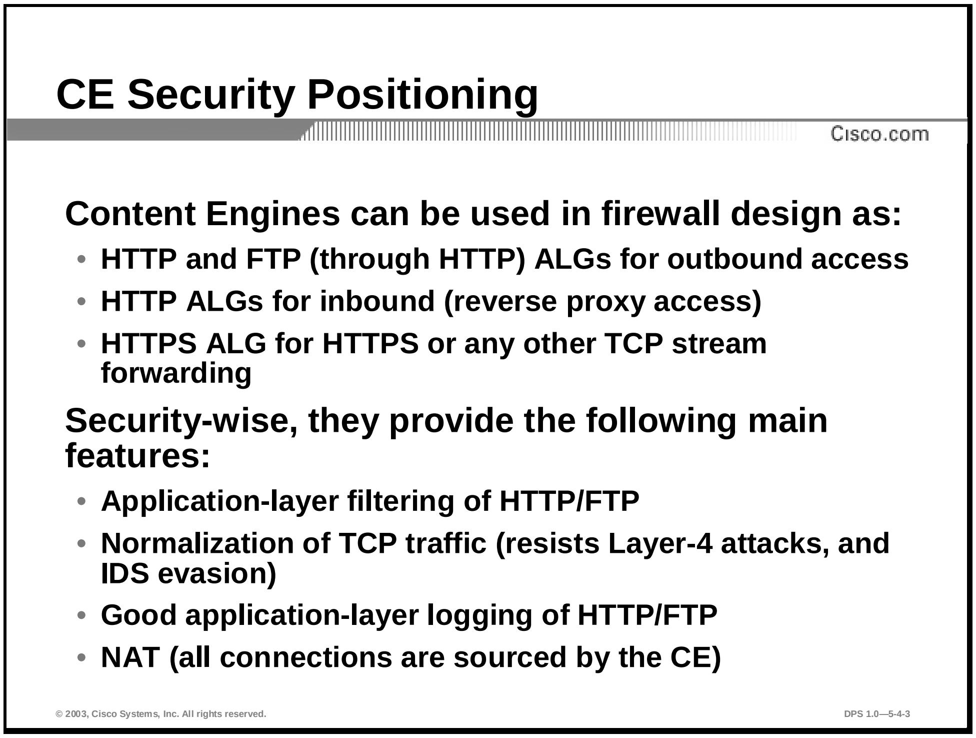
### Outline

**This lesson includes these sections:**

- **Content Engines Security Positioning**
- **Reverse Proxying**
- **Application Access Control**
- **Firewall Integration**
- **Content Engine Limitations**

DPS 1.0—5-4-2

# Content Engines Security Positioning



## CE Security Positioning

**Content Engines can be used in firewall design as:**

- **HTTP and FTP (through HTTP) ALGs for outbound access**
- **HTTP ALGs for inbound (reverse proxy access)**
- **HTTPS ALG for HTTPS or any other TCP stream forwarding**

**Security-wise, they provide the following main features:**

- **Application-layer filtering of HTTP/FTP**
- **Normalization of TCP traffic (resists Layer-4 attacks, and IDS evasion)**
- **Good application-layer logging of HTTP/FTP**
- **NAT (all connections are sourced by the CE)**

DPS 1.0—5-4-3

## Objective

This section will enable the learner to identify security features of Content Engine products.

## Introduction

Content engines have been designed “around” the application layer HTTP protocol and can also deal with many security issues originated in this layer.

## What Content Engines Can Do

For example content engines can be used in firewall design as HTTP and FTP ALGs to control outbound access, and HTTP ALGs for inbound access. Forwarding of other TCP stream (i.e. not only HTTP) can be configured as well, but without any application awareness. For security reasons content engines can perform application layer filtering of HTTP and FTP traffic. A normalization of TCP traffic is possible hereby blocking layer-4 attacks including DoS techniques and IDS evasion.

Additionally content engines allow for detailed logging of HTTP/FTP traffic. Furthermore NAT is performed automatically since all connections are sourced by this device.

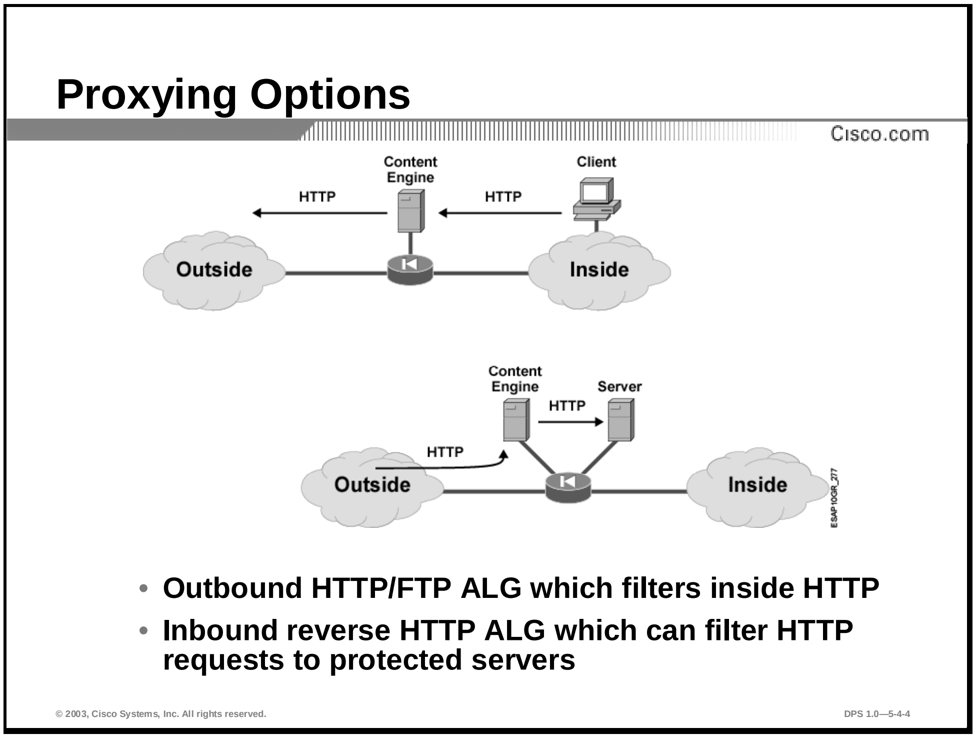
# Practice

Q1) Content engines can act as inbound (reverse) proxies. True or false?

A) true

B) false

# Reverse Proxying



## Proxying Options

Cisco.com

- **Outbound HTTP/FTP ALG which filters inside HTTP**
- **Inbound reverse HTTP ALG which can filter HTTP requests to protected servers**

DPS 1.0—5-4-4

## Objective

This section will enable the learner to explain how reverse proxying can be used to accommodate an organization’s security requirements.

## Introduction

Proxying is a proven security idea based on interrupting and re-creating a traffic flow. Such a technique will sanitize the L3 and L4 protocols, and in the case of ALGs, provide application-layer insight into the stream. Using content engines this proxying can be performed in a powerful way, and filter certain URLs to the protected servers.

This figure illustrates inbound HTTP and outbound HTTP/FTP traffic. The content engine acts as a proxy device and is able to filter HTTP requests, for example to certain protected servers on the inside network.

# Practice

Q1) How does a reverse proxy sanitize the TCP stream between the client and the server?

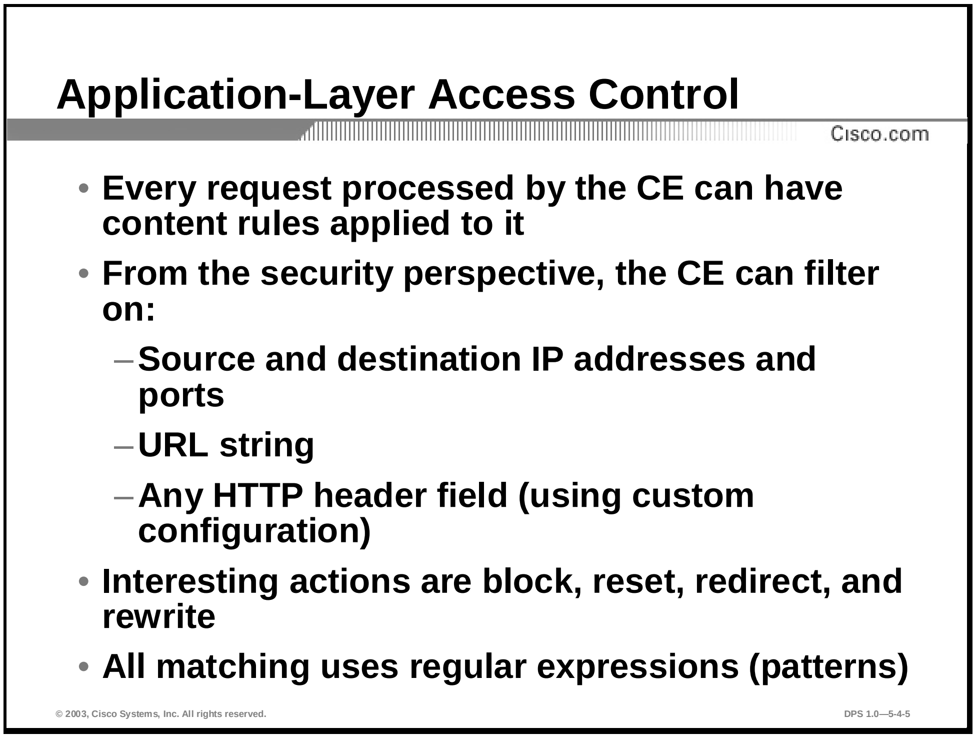
A) by recreating a TCP session and using its own TCP stack to talk to the server

B) by using WCCP instead of HTTP to the server

C) by using UDP instead of TCP to the server

D) by recreating the application layer request in the original TCP session

# Application Access Control



## Application-Layer Access Control

- Every request processed by the CE can have content rules applied to it
- From the security perspective, the CE can filter on:
  - Source and destination IP addresses and ports
  - URL string
  - Any HTTP header field (using custom configuration)
- Interesting actions are block, reset, redirect, and rewrite
- All matching uses regular expressions (patterns)

© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—5-4-5

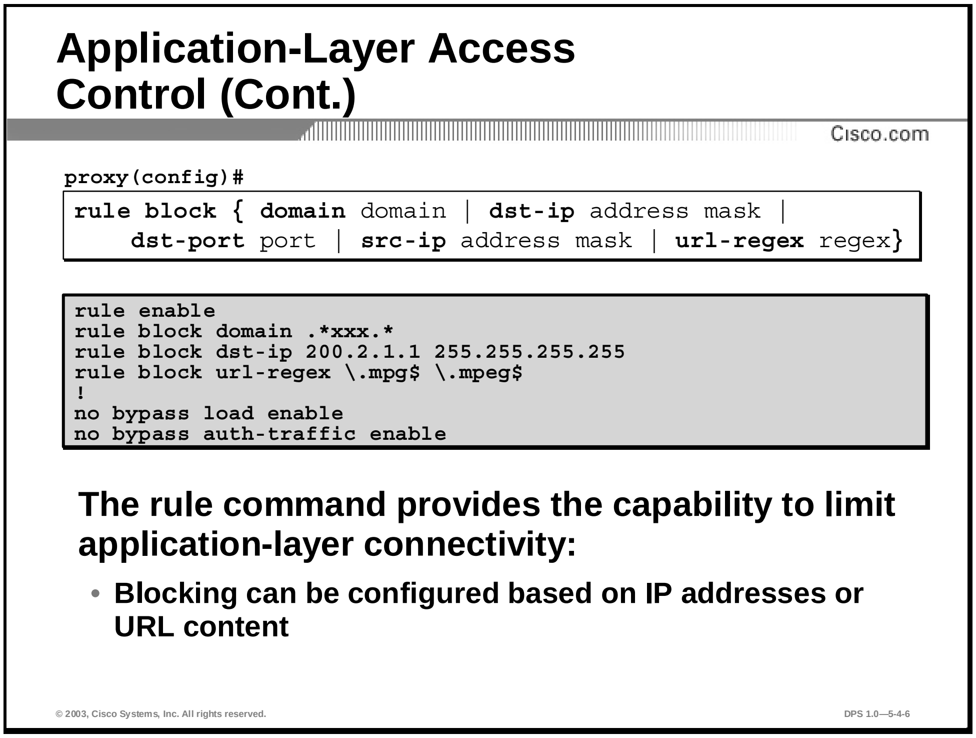
## Objective

This section will enable the learner to explain how Content Engine application access control can be used to accommodate an organization’s security requirements.

## Introduction

Content engines not only assist layer-4 traffic flows by considering layer-7 needs, also filtering rules can be applied on these flows. It is very important to understand the actions and rules specific issues in parallel to content switching.

Note that every single request that is processed by the content engine can require a content rule to be applied on it. Filtering rules an be applied on socket information that is source and destination IP address and source and destination port numbers as well as the layer-7 uniform resource locator (URL) string carried by HTTP. The actions that are triggered from this rule can be one of block, reset, redirect, and rewrite. All rules are defined with matching patterns specified using regular expression.



## Application-Layer Access Control (Cont.)

```
proxy(config)#
```

```
rule block { domain domain | dst-ip address mask |
    dst-port port | src-ip address mask | url-regex regex}
```

```
rule enable
rule block domain .*xxx.*
rule block dst-ip 200.2.1.1 255.255.255.255
rule block url-regex \.mpg$ \.mpeg$
!
no bypass load enable
no bypass auth-traffic enable
```

**The rule command provides the capability to limit application-layer connectivity:**

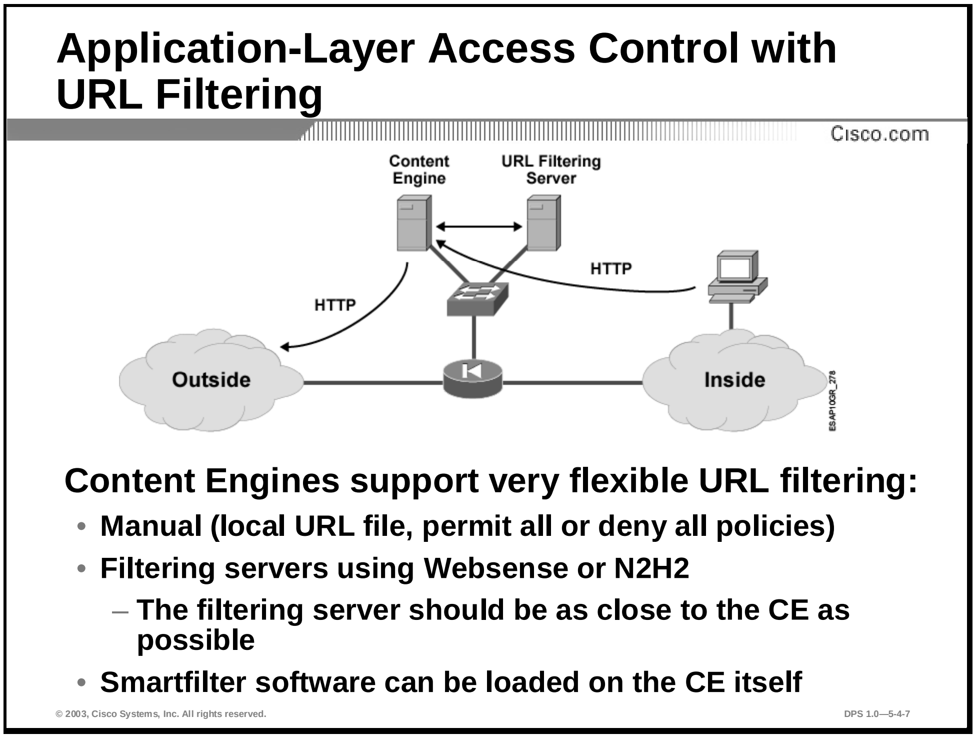
- **Blocking can be configured based on IP addresses or URL content**

DPS 1.0—5-4-6

## Configuration Example

This figure illustrates how to configure typical blocking rules:

- The first rule blocks HTTP requests to an URL containing "xxx" (as it is used for many offending sites)
- The second rule blocks a given IP destination address
- The third rule disallows downloads of MPEG-coded video files



## Application-Layer Access Control with URL Filtering

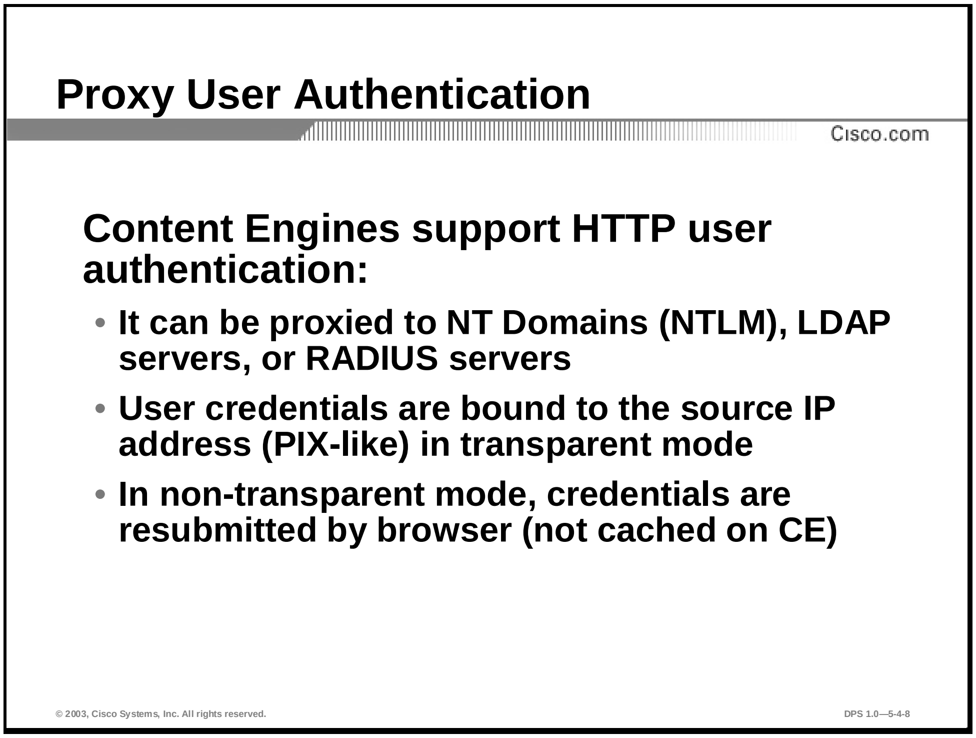
**Content Engines support very flexible URL filtering:**

- **Manual (local URL file, permit all or deny all policies)**
- **Filtering servers using Websense or N2H2**
  - **The filtering server should be as close to the CE as possible**
- **Smartfilter software can be loaded on the CE itself**

DPS 1.0—5-4-7

# URL Filtering

One of the most powerful capabilities with content engines is their flexibility in supporting URL filtering. This can either be configured manually using a local URL file, or by using filtering servers, or by using a Smartfilter software that can be loaded on the content engine itself. Note that a filtering server should be located as close as possible to the content engine for performance reasons.



## Proxy User Authentication

**Content Engines support HTTP user authentication:**

- It can be proxied to NT Domains (NTLM), LDAP servers, or RADIUS servers
- User credentials are bound to the source IP address (PIX-like) in transparent mode
- In non-transparent mode, credentials are resubmitted by browser (not cached on CE)

© 2003, Cisco Systems, Inc. All rights reserved. DPS 1.0—5-4-8

# Content Engines and Authentication

Additionally to filtering capabilities, content engines can also be used for HTTP user authentication. Here the content engine acts as proxy for NT Domains, LDAP servers, or RADIUS servers.

### Transparent Mode

The authentication is verified via user credentials that are bound to the source IP address as it is done by the PIX, hereby simulating a transparent mode.

### Non-transparent Mode

If configured in non-transparent mode, the browser is aware of the proxy, and the user credentials are resubmitted by the browser and do not need to be cached on the content engine.

**Note** Only one authentication server can be used at a time.

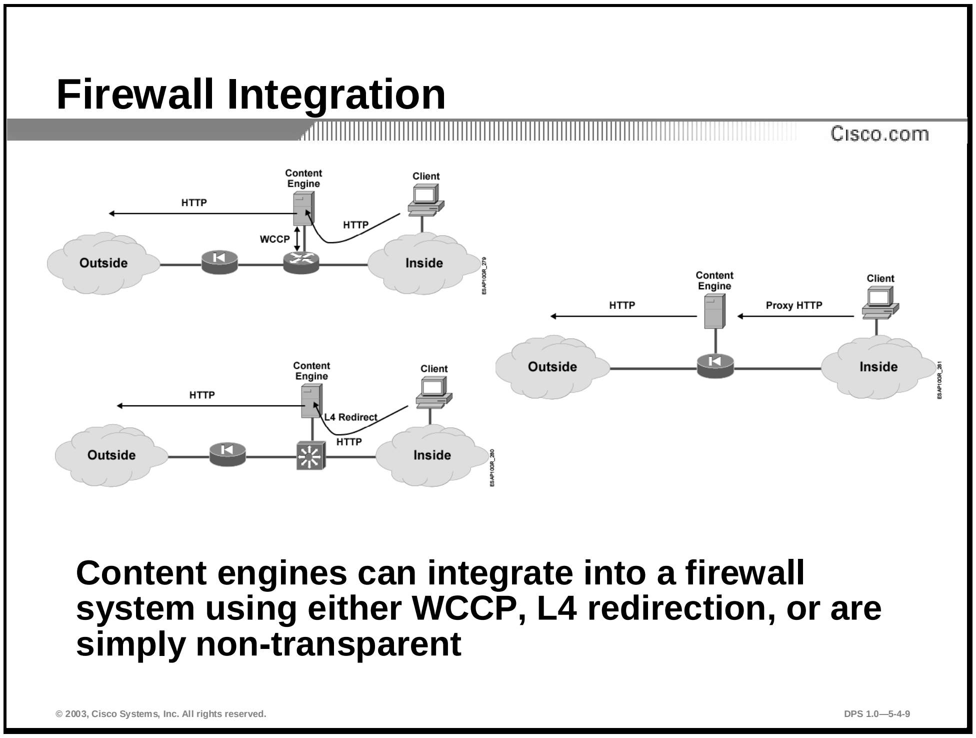
# Practice

Q1) Could you use content engines as reverse proxies to filter out malicious URLs before they hit the protected servers. True or false?

A) true

B) false

# Firewall Integration



## Firewall Integration

Cisco.com

**Content engines can integrate into a firewall system using either WCCP, L4 redirection, or are simply non-transparent**

DPS 1.0—5-4-9

## Objectives

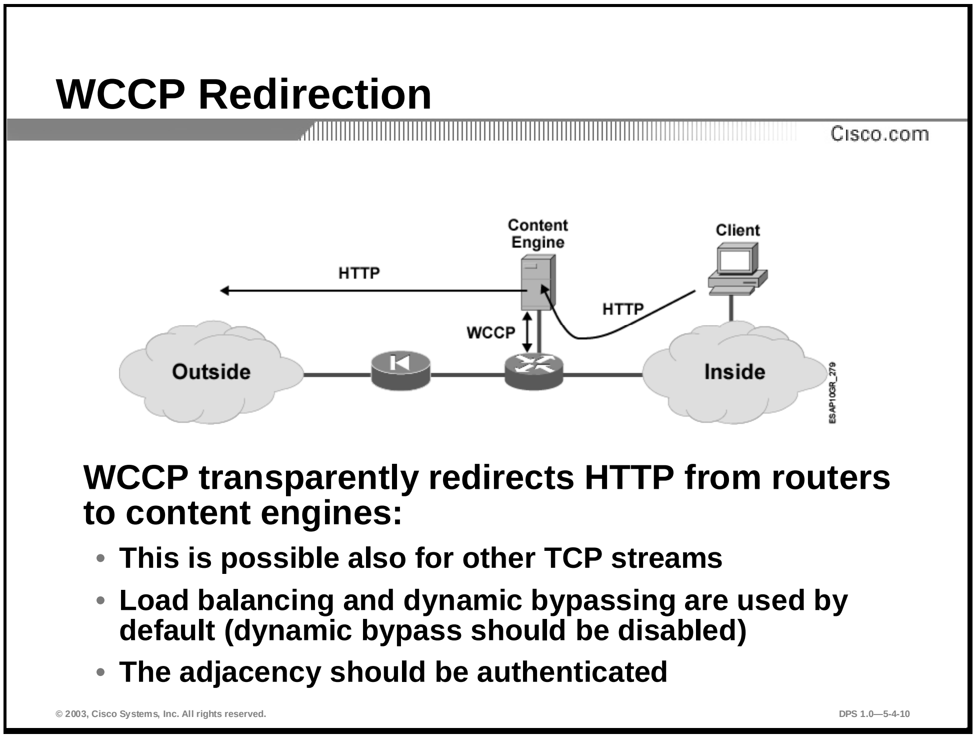
This section will enable the learner to explain how content engines can integrate in a network firewall system.

## Introduction

It is very important to understand how content engines integrate in firewall architectures in order to achieve both maximum performance and maximum security—by keeping the costs low.

Generally, there are three possibilities to integrate content engines within the firewall system:

1. The WCCP protocol can be used as session control method
2. A Layer 4 redirection of inside client traffic can be implemented on the content switch
3. The content engine is simply not transparent, that is, it acts as a proxy system



## WCCP Redirection

Cisco.com

**WCCP transparently redirects HTTP from routers to content engines:**

- **This is possible also for other TCP streams**
- **Load balancing and dynamic bypassing are used by default (dynamic bypass should be disabled)**
- **The adjacency should be authenticated**

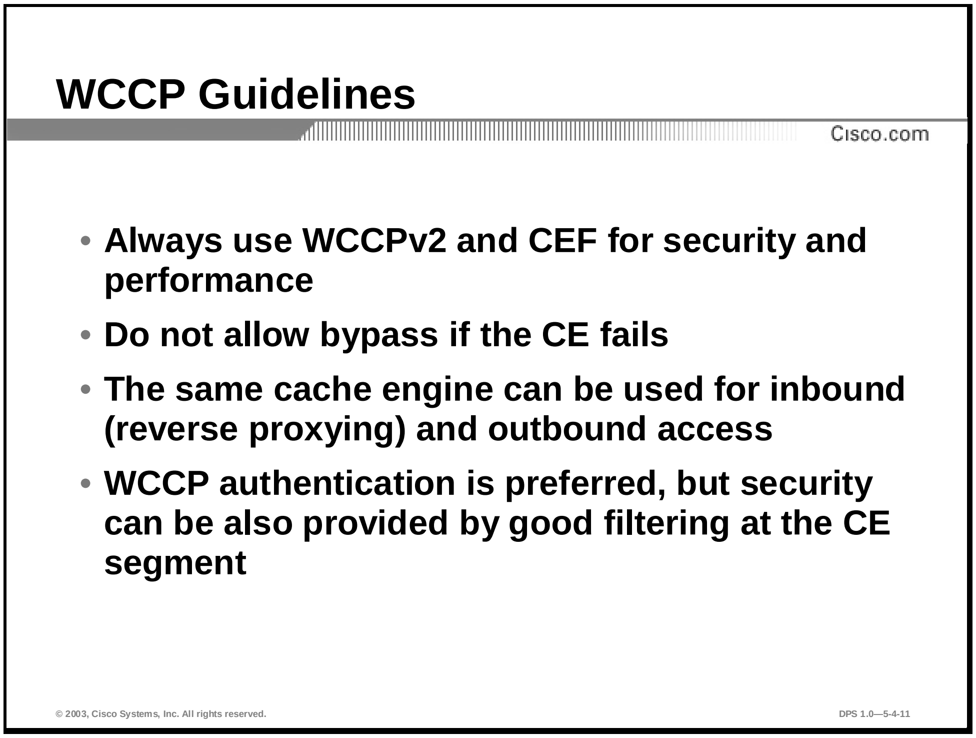
© 2003, Cisco Systems, Inc. All rights reserved.

DPS 1.0—5-4-10

# WCCP and Session Redirection

The Cisco proprietary WCCP protocol can be used to manage HTTP session redirection from routers to content engines. This functionality was originally designed for HTTP only but can also be applied to other TCP streams. WCCP performs load balancing per default. Dynamic bypassing of caching redirections should be disabled, in order not to allow any clients to every bypass the content engine.

The adjacency of the content engine and the router should be authenticated in order to prevent spoofed redirection commands.



## WCCP Guidelines

Cisco.com

- Always use WCCPv2 and CEF for security and performance
- Do not allow bypass if the CE fails
- The same cache engine can be used for inbound (reverse proxying) and outbound access
- WCCP authentication is preferred, but security can be also provided by good filtering at the CE segment

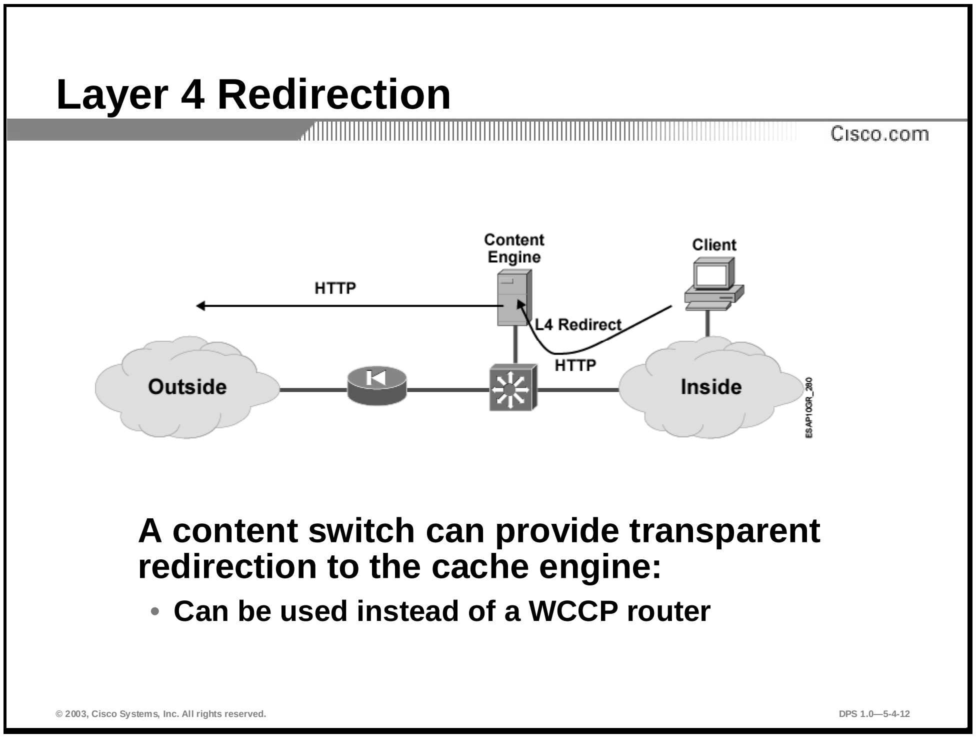
DPS 1.0—5-4-11

## Guidelines for WCCP Adjacencies

In order to assure maximum security you should only enable WCCP version 2. Additionally Cisco Express Forwarding (CEF) should be enabled for maximum performance. Bypassing redirections should not be allowed in case the content engine fails.

For economical reasons it is recommended to use the same caching engine for inbound and outbound access. The inbound access method is called reverse proxying.

To prevent spoofed WCCP messages, it is recommended to enable WCCP authentication. On the other hand, security can be also provided by a good flow filter configuration at the CE segment.



## Layer 4 Redirection

Cisco.com

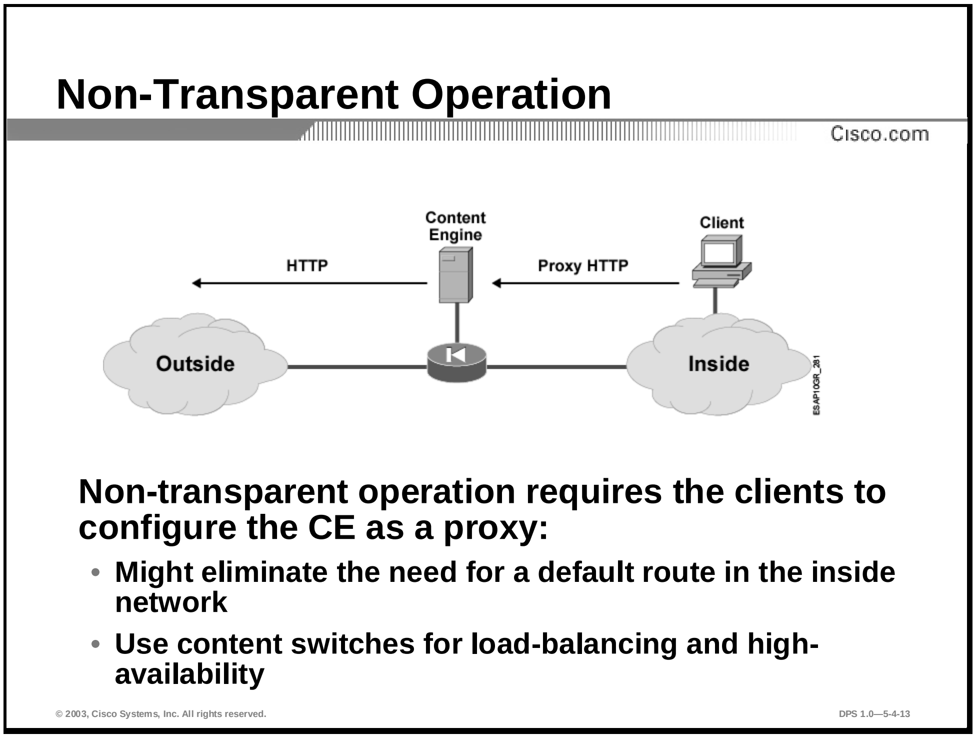
**A content switch can provide transparent redirection to the cache engine:**

- **Can be used instead of a WCCP router**

DPS 1.0—5-4-12

# Layer 4 Redirection

Alternatively to WCCP, a content switch, such as the Cisco CSS 11000, can be configured to provide transparent redirection to the cache engine. That is every inside client HTTP request is automatically redirected to the content engine.



## Non-Transparent Operation

Cisco.com

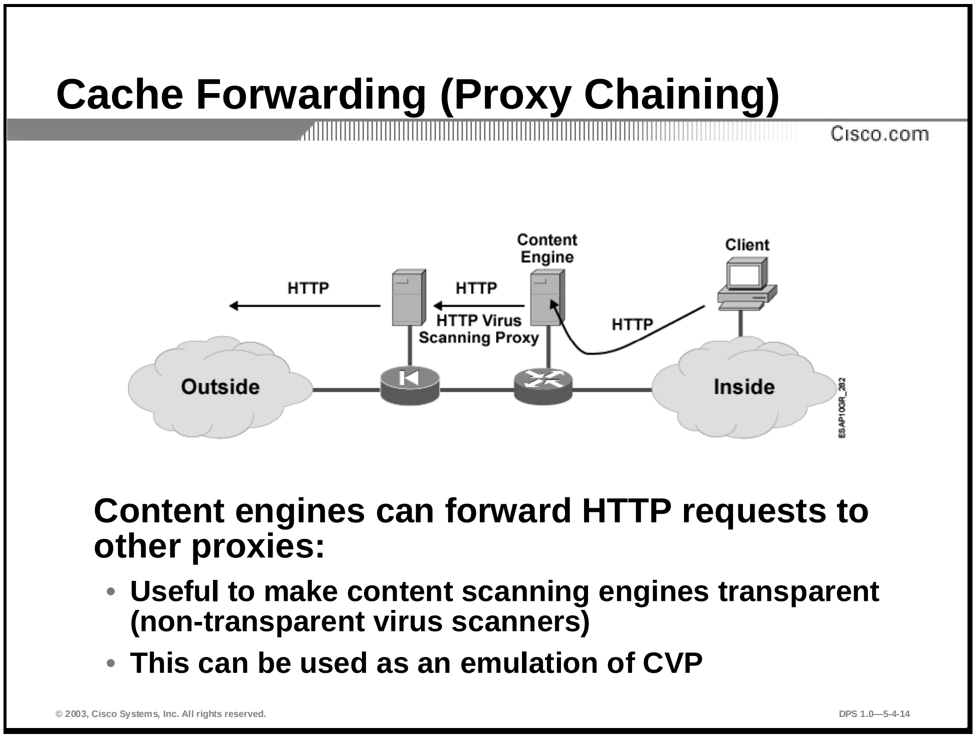
**Non-transparent operation requires the clients to configure the CE as a proxy:**

- **Might eliminate the need for a default route in the inside network**
- **Use content switches for load-balancing and high-availability**

DPS 1.0—5-4-13

## Proxying Requests

If non-transparent operation has been enabled at the content engine, all clients must be configured to use the content engine as HTTP proxy. This method might eliminate to inject a default route in the inside network. Additional content switches can be employed for load-balancing purposes and high availability designs.



## Cache Forwarding (Proxy Chaining)

Cisco.com

**Content engines can forward HTTP requests to other proxies:**

- **Useful to make content scanning engines transparent (non-transparent virus scanners)**
- **This can be used as an emulation of CVP**

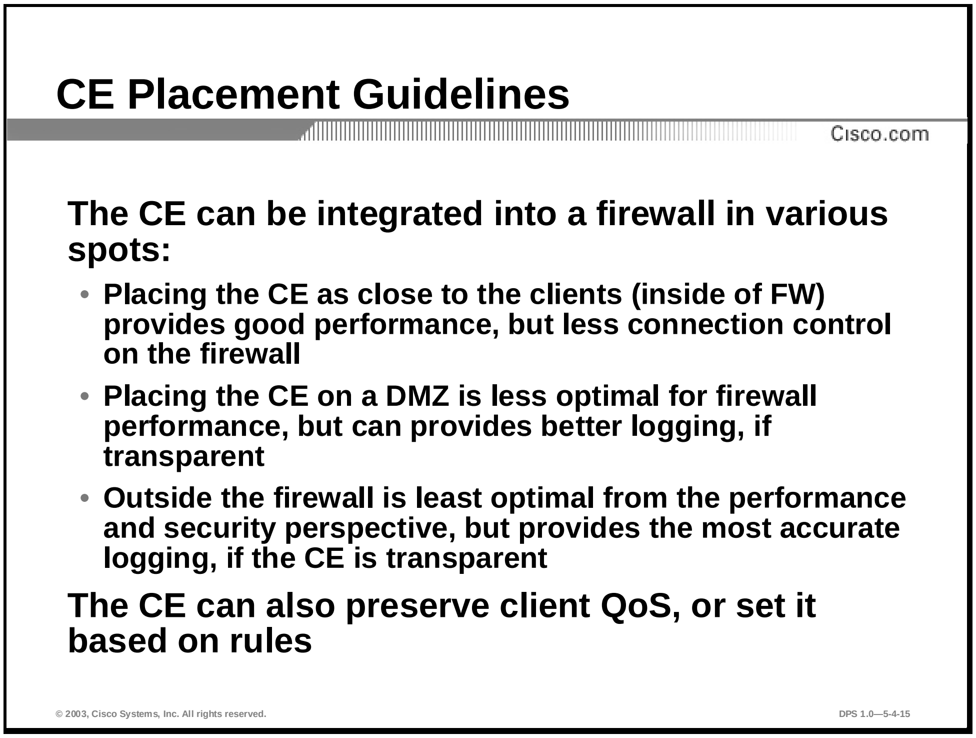
DPS 1.0—5-4-14

# Proxy Chaining

Another method is to employ dedicated HTTP proxy servers and configure the content engine to forward HTTP requests to these proxies. This alternative is useful to make content scanning engines—such as non-transparent virus scanners—transparent to the applications.

This “cache forwarding” or “proxy chaining” method can also be used as emulation for the Content Vectoring Protocol (CVP).

CVP provides an asynchronous interface to server applications that scans file content for virus detection. Using CVP a client-server relationship is established that enables different firewall systems to share a common content validation server. Basically, a single content validation server inspects the flagged incoming files for inspection.



## CE Placement Guidelines

Cisco.com

**The CE can be integrated into a firewall in various spots:**

- **Placing the CE as close to the clients (inside of FW) provides good performance, but less connection control on the firewall**
- **Placing the CE on a DMZ is less optimal for firewall performance, but can provides better logging, if transparent**
- **Outside the firewall is least optimal from the performance and security perspective, but provides the most accurate logging, if the CE is transparent**

**The CE can also preserve client QoS, or set it based on rules**

DPS 1.0—5-4-15

## Guidelines for Physical Design

Several locations can be considered for content engine locations:

- **Place the content engine as close as possible to the inside-clients:** This method provides a good performance but lowers a precise connection control on the firewall.

- **Place the content engine on a DMZ:** This solution is good for a detailed logging functionality but lowers the firewall performance.

- **Place the content engine outside the firewall:** This is the least optimal solution from a performance and security point of view. The only advantage is that the firewall logging is most accurate here (it always reflects the real endpoints of the communication), in case the content engine has been configured to work in transparent mode.

# Practice

Q1) What does proxy chaining refer to when using the Content engine?

A) the ability to forward a HTTP request directly to the destination server

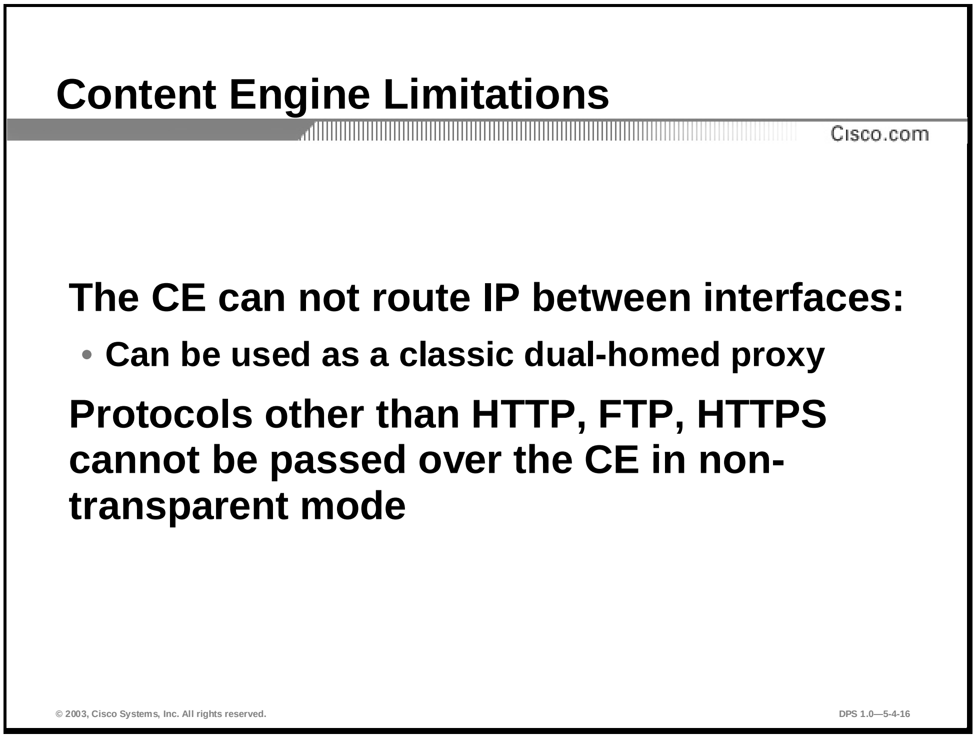
B) the ability to forward a HTTP request to another proxy server

C) the ability to forward a HTTP request to a stateful firewall

D) the ability to forward a HTTP request only to virus scanners

E) the ability to accept non-transparent proxy connections

# Content Engine Limitations



## Content Engine Limitations

**The CE can not route IP between interfaces:**

- **Can be used as a classic dual-homed proxy**

**Protocols other than HTTP, FTP, HTTPS cannot be passed over the CE in non-transparent mode**

DPS 1.0—5-4-16

## Objective

This section will enable the learner to identify the limitations of Content Engine products.

## Introduction

Although this lesson listed many advantages of using content engines integrated in a firewall architecture, the learner must be aware of its limitations. These limitations originate in the design goals of content engines, which are not basically tailored for security purposes.

### A CE is not a Router

Never forget that a content engine has no routing functionality hence IP packets cannot be routed between the interfaces of the CE. But the CE can be used as classical dual-homed proxy device, terminating Layer 7 (L7) streams, not allowing L3 forwarding.

### A CE Cannot Proxy Anything

Additionally, a content engine cannot work in non-transparent mode for protocols other than HTTP, FTP, and HTTPS.

## Practice

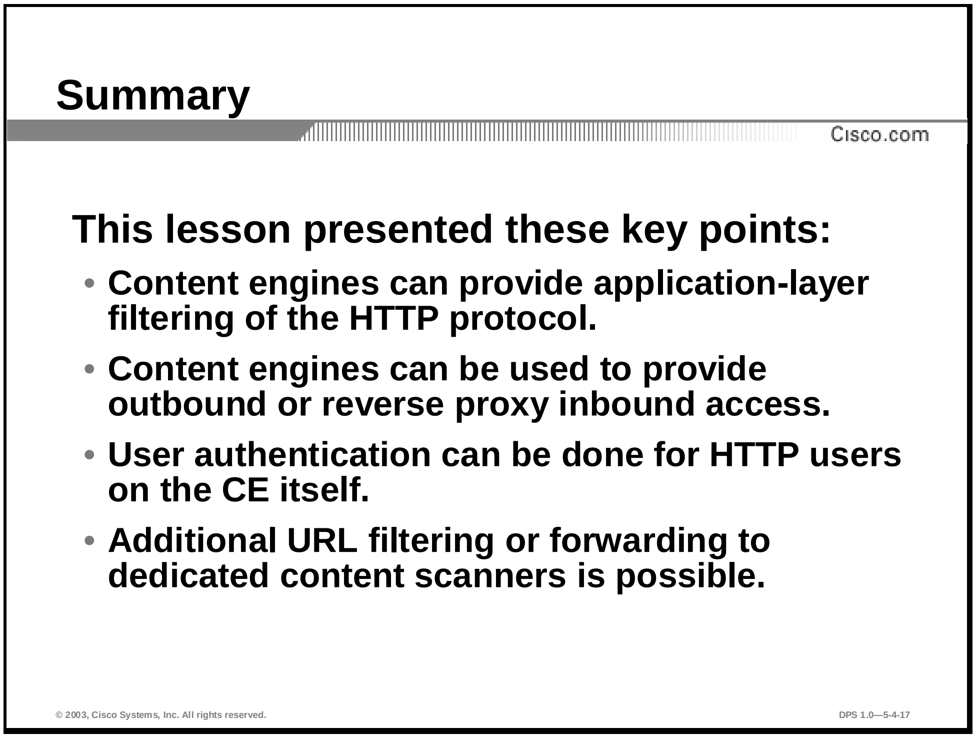
Q1) The CE can act as an IP router for non-HTTP traffic. True or false?

A) true

B) false

# Summary

This section summarizes the key points discussed in this lesson.



## Summary

Cisco.com

This lesson presented these key points:

- Content engines can provide application-layer filtering of the HTTP protocol.
- Content engines can be used to provide outbound or reverse proxy inbound access.
- User authentication can be done for HTTP users on the CE itself.
- Additional URL filtering or forwarding to dedicated content scanners is possible.

DPS 1.0—5-4-17

# Quiz: Content Engines

Complete the quiz to assess what you have learned in this lesson.

## Objectives

This quiz test your knowledge on how to:

- Identify security features and limitations of Content Engine products when using them in a firewall system design

## Instructions

Answer these questions:

1. What are the options for connecting clients to content engines?
2. What are the security features of the WCCP protocol?
3. What are the possible CE placement scenarios inside a firewall?

## Scoring

You have successfully completed the quiz for this lesson when you earn a score of 80 percent or better.

# Firewall Design

## Objective

You were asked to provide input on several firewall designs, ranging from designed-from-scratch systems, to existing systems, which need to be improved. The objective of those exercises is to use the acquired firewall design knowledge and apply it to real-life situations.

# Detailed Instructions

Complete the following tasks to finish this case study exercise.

## Task 1: Enterprise Firewall Redesign

You were called in to help in the redesign of an enterprise (bank) firewall, which is currently set up according to the picture below.

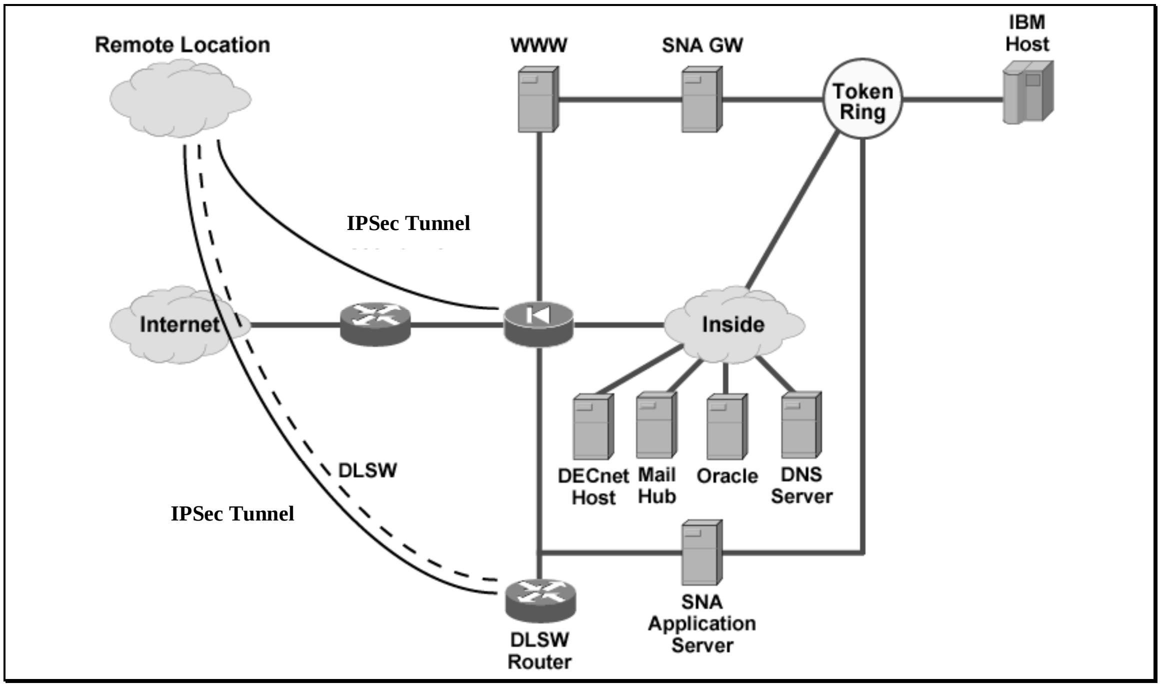


**Figure 1: Firewall/network topology**

A PIX Firewall acts as the central access control element. The following networks are connected to the PIX Firewall:

- The outside (Internet) network.
- The inside (enterprise campus) network.
- A DMZ hosting a WWW server, which talks over IP to a SNA gateway, which is in turn connected to a SNA-only token ring network and the IBM host (mainframe).
- A DMZ hosting a router, which terminates DLSw (SNA in TCP) sessions, which run over an Internet VPN. The IPSec sessions protecting the DLSw sessions also terminate on this router. A SNA application server (speaking **only** SNA protocols) provides a path to the inside, bypassing the firewall.
- The PIX Firewall also terminates VPN sessions from remote locations.

Over the firewall, the following applications are enabled:

- From the Internet, web (HTTP) access is allowed to the public WWW server in the DMZ.

- From the Internet, business partners can access a web-based e-commerce application, which is hosted on the same server as the public WWW server. This e-commerce application talks using IP (IBM MQseries) to the SNA gateway (SNA GW), which is dual-homed to an internal token-ring SNA-only network.

- A SNA application runs between the VPN remote sites and the central site. SNA traffic is bridged into DLSw, which terminates on a router in the DMZ. From there, a SNA application server is connected into the DMZ, dual-homed to the inside. This SNA traffic is fully trusted.

- HTTP, HTTPS, FTP, telnet, and RealAudio are allowed outbound to the Internet.

- SMTP email and DNS are exchanged with the Internet, using direct connections from/to two servers (DNS server, mail hub) in the inside network.

The remote location, reachable over the VPN, is designed like this:

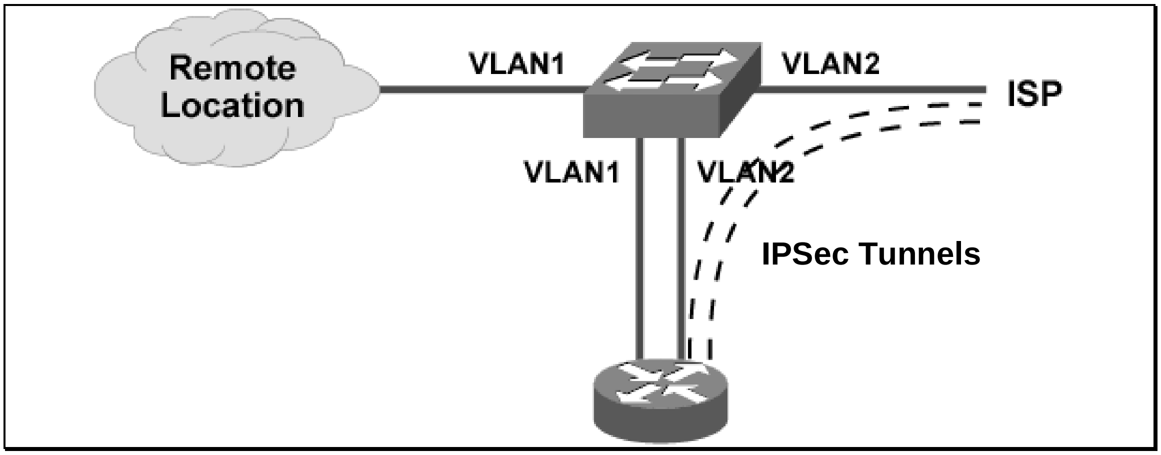


Figure 2: Remote location architecture

## Requirements

The main security risks seen by the bank at the network perimeter are:

- Compromise of important hosts in the inside network, by outside attacks

- Business interruption due to denial-of-service attacks from the Internet

- Lost reputation due to compromised public servers

The bank is worried about the firewall having multiple single-points of failure, and would like to cooperate with you in the firewall redesign. However, the redesign should be done in multiple phases, the highest priority being the integration of four new services in the existing firewall:

- Dial-up and VPN remote access for internal users, which should access the whole inside network, when connected.

- Dial-up and VPN remote access for business partners, which should only access the WWW server’s extranet application.

- A business partner needs to have both IP (direct Oracle access to a server in the inside network) and DECnet connectivity to the bank over a WAN (i.e. non-VPN) link. DECnet connectivity is needed only to a single VAX/VMS host in the company network.

- Integration of virus scanning for email and HTTP for all data from the Internet.

- Per-user accounting of all outbound services to determine Internet usage patterns.

Complete the following tasks:

**Step 1** Identify weak points in firewall design and suggest improvements. Change the existing firewall design by integrating the new services (DECnet connectivity, dial-up/VPN remote access) into it. You can assume that the existing firewall filter can have up to 4 new interfaces. Practice defense-in-depth, where appropriate.

.

## Verification

There will be an extended discussion session, where you will be able to present your ideas and compare them to other groups.

## Task 2: E-Banking Site Design

A financial organization has invited you to design a perimeter solution from scratch. They need a solution, which would host an e-commerce/e-banking site on the Internet, connecting it to its existing backend transaction servers.

## Requirements

The main security risks seen by the organization at the network perimeter are (in the order of importance):

- Compromise of backend transaction servers and their data
- Denial-of-service attacks against the e-commerce system or the supporting architecture (links, network devices, DNS servers)
- Compromise of second-tier servers and the subsequent spoofed transactions
- Compromise of first-tier servers

The technology requirements for the solution are:

- The Internet connection is 2x OC-3 (155 Mbps). The organization realistically expects around 200 Mbps of traffic to be present bidirectionally at peak usage.
- The application is three-tiered: the web server talks to an application server (CORBA protocol), and the application server talks to the backend transaction servers using the IBM CICS protocol.
- No classic VLANs must be used – the customer does not trust them. Private VLANs are acceptable as a defense-in-depth solution.
- A cluster of web servers will be used, as well as a cluster of application servers. Server load balancing should be integrated in the firewall to enable clustering.
- Users will connect to the web servers using HTTPS, using client-side SSL certificates to authenticate them.
- The web server farm will be able to sustain up to 200 Mbps of throughput between users and servers. Around 500 Mbps of aggregate throughput is needed between the web and second-tier application servers, and the same throughput between the second-tier application servers, and the back-end transaction server. The firewall should support such throughput.
- DNS resolution for the e-commerce domain must be provided. Integrate this into the firewall or suggest alternative solutions.

- Denial-of-service prevention should be one of the main focuses of the solution. Design features to provide server (application) and link protection.

There is no need for any outbound sessions through the firewall.

**Step 1** Design a firewall system from scratch, implementing the above requirements.

**Step 2** Design an out-of-band management solution, which would still provide IP (SNMP) management access to any firewall element (except the servers) if any of the other ("in-band") elements, such as a firewall filter, fails. Make sure that your solution does not significantly impact the security of the system.

## Verification

There will be an extended discussion session, where you will be able to present your ideas and compare them to other groups.

# Task 3: Hospital Voice/Multimedia Perimeter Design

You are called in to assist a large hospital with network access control. The network is preparing to integrate Voice over IP services into the switched infrastructure, and is concerned with the security aspects of such integration.

The basic voice/multimedia design of the network is shown in the following figure:

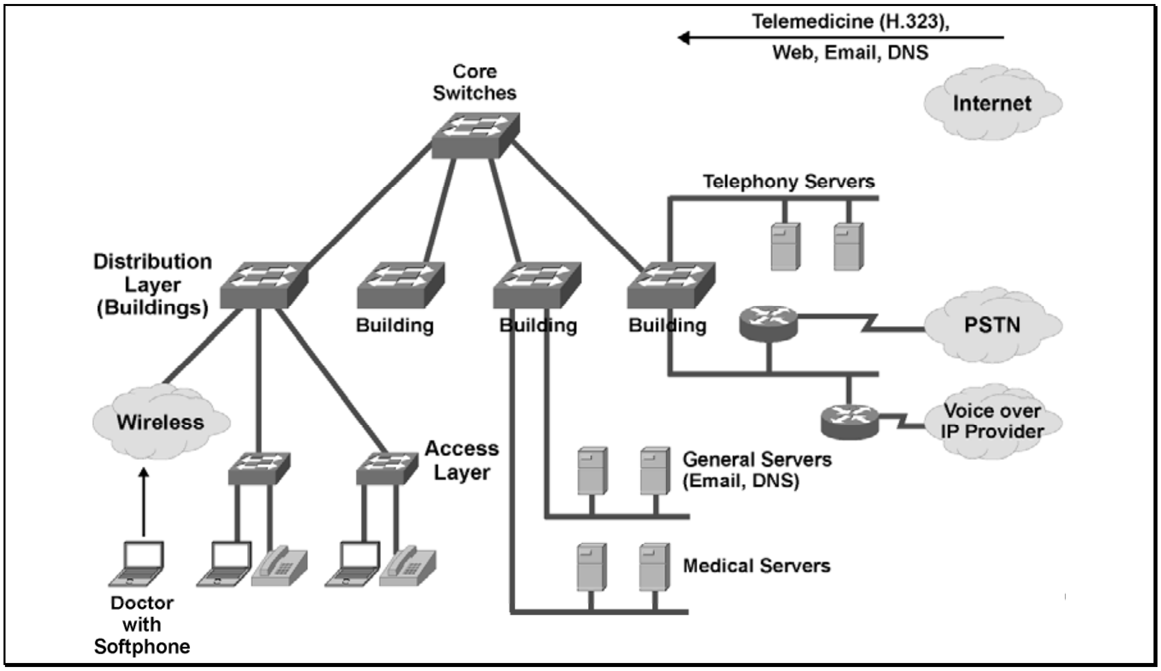


**Figure 3: Network topology**

The distribution layer of the network is where the server farms are connected. There are basically three server farms on the campus network:

- The database servers with confidential medical information
- The general-purpose (email, DNS, etc.) servers
- The telephony servers and gateways

There are three types of users sharing the network:

- Doctors and medical personnel (static/offices and roaming/wireless)
- Administrative personnel
- IT maintenance personnel

# Requirements

The main security risks seen by the hospital are:

- Availability of mission critical data, including inside medical servers, and external (video streaming) servers
- Compromise of medical servers containing confidential information
- Confidentiality and availability of the voice network
- Access to medical information by non-medical personnel (maintenance personnel, administration)

The following applications pass between the hospital network and outside networks

- Inbound HTTP to an exposed server (telemedicine streaming server)
- Inbound HTTP to an exposed server (public web server)
- Inbound and outbound SMTP using an exposed server (mail relay)
- Inbound and outbound DNS using an exposed server (public DNS server)
- Inbound and outbound H.323 to the VoIP provider

Internally, the following services are provided

- Skinny-based VoIP, including Voicemail
- Access to database servers (medical information) using HTTP and direct Oracle SQL*net
- IMAP4-based email with central servers
- DNS

## Voice/Multimedia Architecture Requirements

The voice and multimedia architecture/technology requirements are:

- Internally, the voice network uses Skinny for call control. The call manager software is able to integrate the Skinny domain with H.323 domains.
- The voice network connects to the outside voice world through a PSTN gateway, and a Voice-over-IP provider. The VoIP provider uses strictly H.323 to connect its customers.

- Other hospitals use Telemedicine applications to receive streaming video/audio from the hospital over the Internet. SLAs are set up to guarantee delivery, and a server streams video/audio from cameras, directly connected to the server.

- Simple voicemail must be available to all phone users. The customer is willing to accept your suggestion on how to implement it.

### Security Requirements

The security requirements are:

- There must be separation between the three classes of users, and firewall available to restrict any user class to a specific set of services.

- There must be maximum separation of data and voice traffic throughout the network.

- Some IP phones are located in offices. Those switch ports are considered as trusted as the office data ports.

- Some IP phones are available to the general public (such as patients). Those switch ports are considered **untrusted**.

- Doctors roaming around the campus will use wireless connectivity, and will connect to both the database servers, and use SoftPhones installed on their laptops. Wireless connectivity must be enhanced using VPN technology with strong (3DES) encryption.

- Maintenance personnel will also use the wireless network, but only to access and maintain non-confidential (non-medical) servers.

- A user, who would break in into the voice infrastructure, should not be able to easily compromise the data infrastructure.

### Management Requirements

The customer would like an out-of-band management network connecting all network devices and telephony servers. Provide guidelines on how to design the OOB network and connect it with the campus network.

Complete the following step:

**Step 1** Design the network according to the requirements

## Additional Design Questions

What would change if all users used software instead of hardware IP phones?

## Verification

There will be an extended discussion session, where you will be able to present your ideas and compare them to other groups.

NOTES

NOTES

DPS

# Firewall Design Solutions

## Objective

You were asked to provide input on several firewall designs, ranging from designed-from-scratch systems, to existing systems, which need to be improved. The objective of those exercises is to use the acquired firewall design knowledge and apply it to real-life situations.

# Solutions

## Task 1: Enterprise Firewall Redesign

**Step 1** Identify weak points in firewall design and suggest improvements. Change the existing firewall design by integrating the new services (DECnet connectivity, dial-up/VPN remote access) into it. You can assume that the existing firewall filter can have up to 4 new interfaces. Practice defense-in-depth, where appropriate.

You **SHOULD** have incorporated the following features in your design:

- Prevent direct inbound connections to inside DNS and mail servers (provide application-layer gateways/dedicated outside servers in the firewall DMZs)

- Fully separate tiers of the e-commerce application (i.e. e-commerce web server and backend on different DMZs)

- Separate e-commerce web server from the public web server (they are originally running on the same host).

- Provide a method to filter access from business partners' VPN connections

- Provide a method to filter access from business partners' WAN connection to and limit them to Oracle SQL*net using a firewall which understands the protocol

- Provide a proxy to handle HTTP, and a mail relay for virus scanning

- Address VLAN separation on remote location – use physical separation and/or extremely good router ACLs

You **MIGHT** have also incorporated the following security features in your design. Usually, the more of the below you have implemented, the better:

- Enabled filtering defense in depth using two filtering elements (the access router and a PIX Firewall, two PIX Firewalls, a PIX Firewall and an IOS Firewall)

- Fully separate all services to their own DMZ to provide the best filtering capability.

- Move the DLSw router from the DMZ to the inside (if the VPN is fully trusted, the traffic coming from it (i.e. DLSw) does not need to be filtered).

- Provide a way of filtering SNA traffic after it has crossed SNAGW to limit exposure of internal SNA hosts. A two-interface router acting as a MAC-filtering bridge is one example of such a filter.

- Provide a way of filtering DECnet to the inside network and minimizing exposure of inside DECnet hosts. A two-interface DECnet router with DECnet filters can perform good filtering. The destination VAX/VMS host might be dual-homed, dedicating an interface only to service business partners and NOT acting as a DECnet router.

- Separate the outside (public) DNS and mail servers on the firewall (using separate DMZs/private VLANs).

- Provide a method to totally separate VPN RA users (internal vs. business partners) - so they cannot talk to each other in any way, for example using different VPN RA concentrators on different firewall filter interfaces.

- Practice defense in depth with SMTP email delivery (using two relays to get mail to the inside network – an exposed mail relay on the outside, forwarding to the virus scanning relay, forwarding to the internal mailbox server).

## Task 2: E-Banking Site Design

**Step 1** Design a firewall system from scratch, implementing the above requirements.

You **SHOULD** have incorporated the following features in your design:

- Provide a protected separate (isolated in its own DMZ) DNS server, servicing incoming requests ONLY (alternatively, host DNS at your ISP/ASPs)

- Address performance concerns (firewall filters must never switch more than 200/500 Mbps of traffic)

- Separate the application into tiers, allowing only minimal connectivity between tiers, each tier being in its own DMZ

- Have incorporated SYN flooding protection on the firewall filters or end-host TCP stack

- Provide some ideas about QoS deployment and ISP contract in terms of incident response

- NOT use SSL acceleration - it won't work with client certificates – you can only use TCP-based load balancing

You **MIGHT** have also incorporated the following security features in your design. Usually, the more of the below you have implemented, the better:

- “Break” routing by using dual-homed hosts at each application tier

- Use a physically separate firewall between the 1-2 and 2-3 tier

- Use a dedicated firewall protecting the third tier on the inside

- Practice defense in depth with different filters between app. tiers

- Do not use VLANs to connect the most and the least sensitive perimeters into the same device

**Step 2** Design an out-of-band management solution, which would still provide IP (SNMP) management access to any firewall element (except the servers) if any of the other ("in-band") elements, such as a firewall filter, fails. Make sure that your solution does not significantly impact the security of the system.

You **SHOULD** have incorporated the following features in your design:

- Dedicate a management LAN interface on each device to management traffic only

- Provide a management LAN, which will not allow transit traffic between managed devices using at least two mechanisms (usually private VLANs and filters on management interfaces)

You **MIGHT** have also incorporated the following security features in your design. Usually, the more of the below you have implemented, the better:

- Place a separate filter between the management LAN and the management stations

- Provide more than two mechanisms preventing firewall bypass over the management network. Examples include:

  — Private VLANs (devices are isolated ports, management firewall/router is the promiscuous port)

  — Access control on the management interface of each device

  — Layer-3 ports for each device, and ACLs/policy routing on them

  — Fake ARP entries for all other devices on each device (prevents talking back to a compromised device), etc.

## Task 3: Hospital Voice/Multimedia Perimeter Design

**Step 1** Design the network according to the requirements

You **SHOULD** have incorporated the following features in your design:

- Separate end stations in 5 VLANs: doctors, IT personnel, administration, trusted phones, untrusted phones

- Use Layer-3 access control to keep those VLANs separate throughout the campus (no connectivity is required between end-user VLANs, except the two voice VLANs)

- Use a firewall filter to protect all centralized servers

- Separate all centralized servers into multiple zones – general purpose zone, medical server zone, telephony server zone

- Firewall the call managers from the data network (isolation)

- Firewall all call managers from both phone VLANs

- Use access control at the wireless edge - separate the voice traffic and both data VLANs at the wireless AP/VPN concentrator

- Create an Internet firewall, separate the telemedicine server and the DNS/mail/WWW servers in their own DMZs

- Create a voice firewall to connect to external networks. Establish a H.323 gatekeeper (and possibly a H.323 proxy) to handle all traffic between the internal voice domain and the world.

You **MIGHT** have also incorporated the following security features in your design. Usually, the more of the below you have implemented, the better:

- Provide a separate call manager farm for the untrusted phones

- Provide QoS guidelines for voice over wireless

- Implemented voicemail so that the voicemail server is in the voice network only (and no connectivity is required between the voice/data network)

# Firewall High Availability

## Objective

In its disaster recovery plans, a company needs to provide long-distance firewall failover over a WAN network. The central site has a PIX Firewall installed, and the remote disaster recovery site also has a PIX Firewall system. As there is no LAN connection between sites, LAN-based failover cannot be used.

# Detailed Instructions

Complete the following to finish this laboratory exercise.

## Designing NAT in Active-Active Load-Balancing Using Routing Protocols

Using two PIX Firewalls in active-active setup, using routing protocols to load-balance traffic can a simple and effective method of balancing. However, symmetric traffic flow must be guaranteed at all times, to enable each PIX Firewall to see all packets of a session.

You were called in to assist in firewall design, where such a load-balancing setup is required. The following picture shows the current implementation of the firewall system.

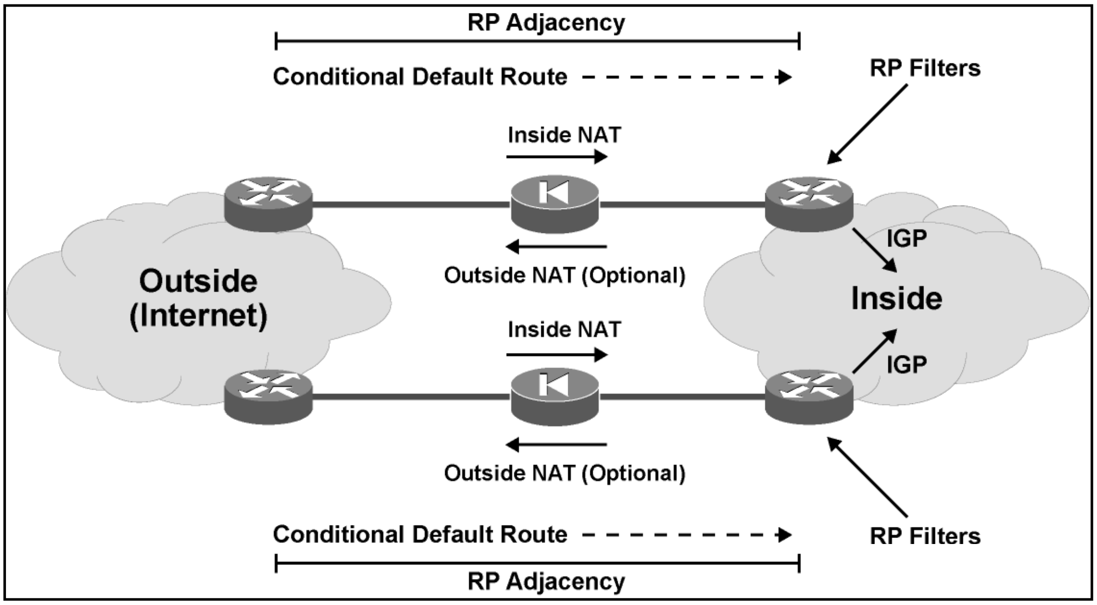


**Figure 1: Load sharing/load balancing with dual NAT**

The organization has attempted to implement such load balancing, but ran into problems when symmetric flow of traffic was required. Obviously, a clever use of NAT will be required to provide symmetric flow.

**Step 1** Provide configuration guidelines on how to configure NAT to provide symmetric flow of traffic over the active-active PIX Firewall pair. Change the basic design of the firewall system, if necessary.

Write your proposed NAT rules in the space below.

____________________________________________________________

____________________________________________________________

____________________________________________________________

____________________________________________________________

____________________________________________________________

**Step 2** Consider the following situations: What if the customer decided to implement your solution for a disaster-recovery center, which is 100 miles from the central site? What would you change in your design? What would be the inter-site connectivity options for the most cost-effective option, and the most robust (quickest failover) option?

## Verification

**Step 3** Discuss your proposed solution with other groups and the instructor.

NOTES

NOTES

DPS

# Firewall High Availability Solutions

## Objective

In its disaster recovery plans, a company needs to provide long-distance firewall failover over a WAN network. The central site has a PIX Firewall installed, and the remote disaster recovery site also has a PIX Firewall system. As there is no LAN connection between sites, LAN-based failover cannot be used.

# Solutions

## Designing NAT in Active-Active Load-Balancing Using Routing Protocols

**Step 1** Provide configuration guidelines on how to configure NAT to provide symmetric flow of traffic over the active-active PIX Firewall pair. Change the basic design of the firewall system, if necessary.

**Answer:** NAT should be configured to provide the translation of client addresses: inside users for outbound connections, and outside users for inbound connections. The two firewalls must use different NAT global pools, which are routed to respective firewalls. BGP should be able to detect failures of firewall elements or external (BGP-enabled) connections. The best firewall is chosen based on preferences inside BGP (local-preference, weights, AS-paths,…). BGP peering needs to be configured between all routers adjacent to the firewall filters.

**Step 2** Consider the following situations: What if the customer decided to implement your solution for a disaster-recovery center, which is 100 miles from the central site? What would you change in your design? What would be the inter-site connectivity options for the most cost-effective option, and the most robust (quickest failover) option?

**Answer:** The best option would be to use BGP routing over the firewall, with timers set low to detect failed connections quickly. NAT can still be used to ensure symmetric flow.

DPS

# Understanding PIX Firewall NAT

## Objective

In this lab exercise, you will analyze various aspects of the PIX Firewall NAT engine to provide connectivity and access control functionality.

# Detailed Instructions

Complete the following to finish this laboratory exercise.

## Design Example Scenario

An organization has a more complex firewall, which connects it to the Internet, over which a intranet VPN is set up, as well to some WAN connections, over which the organization connects to its business partners. This environment has some specific addressing needs:

- The sites reachable over the VPN (in the 10.254.0.0/16 range) should always be visible with their real (internal) IP addresses.
- Some business partners (see picture) have address spaces overlapping with the company address space. Translate your 10.0.1.0/24 subnet to them as 192.168.254.0/24, and their networks into your network as subnets of 172.16.0.0/12.
- All connectivity to the Internet is through a HTTP proxy server (192.168.1.1) in a DMZ. There is no direct Internet connectivity allowed.

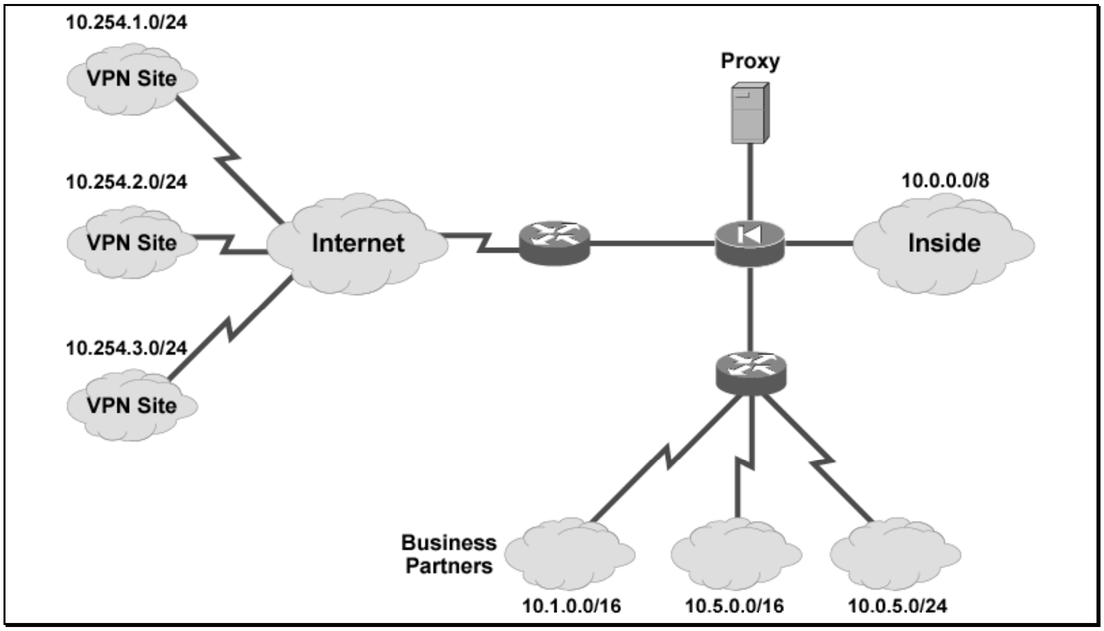


**Figure 1: Example topology**

**Step 1** Describe the required translation rules in the picture and space below.

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

______________________________________________

______________________________________________

______________________________________________

______________________________________________

______________________________________________

### Additional Questions

- If this were a centralized firewall, how would you enable Internet connectivity for all remote (VPN) locations?
- If the company decides to abandon the proxy server and have direct connectivity to the Internet, how would this change your design?

## Verification

Discuss your proposed solution with other groups and the instructor.

# Understanding PIX Firewall NAT Solutions

## Objective

In this lab exercise, you will analyze various aspects of the PIX Firewall NAT engine to provide connectivity and access control functionality.

# Solutions

## Design Example Scenario

**Step 1** Describe the required translation rules in the picture and space below.

```
# inside-to-VPN connectivity
access-list NO-NAT permit ip 10.0.0.0 255.0.0.0 10.254.0.0 255.255.0.0
nat (inside) 0 access-list NO-NAT
# business partners connection
# translate ourselves out as 192.168.254.0/24
static (inside,outside) 192.168.254.0 10.0.1.0 netmask 255.255.255.0
# translate them in as parts of 172.16.0.0/12
static (outside,inside) 172.16.0.0 10.1.0.0 netmask 255.255.0.0 dns
static (outside,inside) 172.17.0.0 10.5.0.0 netmask 255.255.0.0 dns
static (outside,inside) 172.18.1.0 10.0.5.0 netmask 255.255.255.0 dns
# inside access to proxy, proxy access to Internet
static (inside,proxy) 10.0.0.0 10.0.0.0 netmask 255.0.0.0
static (proxy,outside) 200.1.1.1 192.168.1.1
```

## Review Questions

1. If this is a centralized firewall, how would you enable Internet connectivity for all remote (VPN) locations?

   If the firewall is centralized, then the Internet traffic has to run over the VPN (using IPsec or GRE, as no confidentiality is required for Internet traffic), and terminate inside the central firewall. You must not terminate IPsec tunnels on the PIX Firewall outside interface in this case, as traffic cannot exit the PIX Firewall through same interface as it has entered on.

2. If the company decides to abandon the proxy server and have direct connectivity to the Internet, how would this change your design?

   Inside clients then need to be translated out to the Internet (using “nat” and “global” commands).