SecurityTube Python Scripting Expert (SPSE)

SecurityTube Python Scripting Expert

http://www.securitytube.net

Vivek Ramachandran
Course Instructor

Module 3: Network Security

SecurityTube Python Scripting Expert

Part 1:
Client - Server Programming Basics

http://www.securitytube.net

Vivek Ramachandran
Course Instructor

Solutions

- More Students need to try! ☺
- Github / Bitbucket – Security Addicted
- My solutions + SHOWCASE STUDENT WORK ☺
- Module 4! we need more responses! ☺

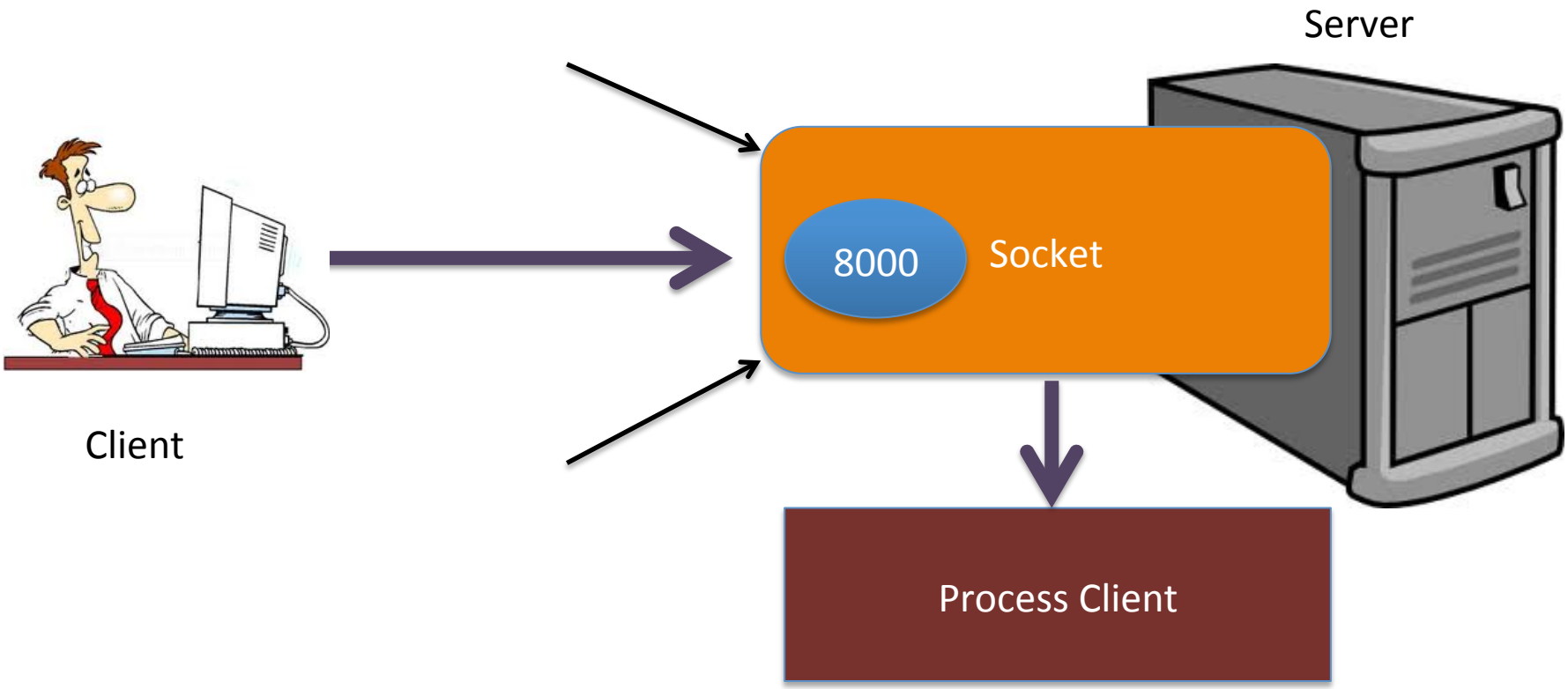
Network Programming

- Socket Programming
  - TCP and UDP Sockets
    - Regular Servers and Clients
  - Raw Sockets
    - Sniffing and Injection

Server and Client Programming

- Server
  - offer a service
- Client
  - use / consume the service
- using the sockets module

Simple TCP Server and Client



Process Client Options

- Process Clients Sequentially and one at a time
- Multi-Threaded Server
- Muti-Process Server
- Non-Blocking Sockets with Select (Multiplexing)

Exercise

- Create a simple Echo Server to handle 1 client
- Create a Multi-Threaded Echo Server
- Create a Multi-Process Echo Server
- Create a Non-Blocking Multiplexed Echo Server using Select()

Module 3: Network Security

SecurityTube Python Scripting Expert

End of Part 1:
Client - Server Programming Basics

http://www.securitytube.net

Vivek Ramachandran
Course Instructor

Module 3: Network Security

SecurityTube Python Scripting Expert

Part 2:
SocketServer Framework

http://www.securitytube.net

Vivek Ramachandran
Course Instructor

SocketServer

- Framework in Python to create TCP and UDP servers
- Does all the basic steps for you in the background
- Comes in handy if you want to create a server to lure a client and analyze its behavior

Module Usage

- Has to be subclass of BaseRequestHandler
- Override handle() to process request
- Call handle_request or serve_forever to process clients
- For TCP Servers
  - self.request is the client socket
  - self.client_address is the client details

Creating an ECHO server

- Code and Demo

Exercise

- Is this server multi-threaded?
- Code up the multi-threaded version of the SocketServer

Module 3: Network Security

SecurityTube Python Scripting Expert

End of Part 2:
SocketServer Framework

http://www.securitytube.net

Vivek Ramachandran
Course Instructor

Module 3: Network Security

SecurityTube Python Scripting Expert

Part 3:
Creating a Web Server

http://www.securitytube.net

Vivek Ramachandran
Course Instructor

How does a Web Application Server Work?

- Listen on port 80 / 443
- Wait for client requests (GET, POST, HEAD …)
- Process Request
  - serve files
  - execute CGI scripts

Simple Web Application Server

- SimpleHTTPServer class
- Implement do_GET()

- can be used to serve exploit code to a client
- can be used to penetration test client side code

Exercise

- Is there a module available to run CGI as well?
- Please write a PoC for the above

Module 3: Network Security

SecurityTube Python Scripting Expert

End of Part 3:
Creating a Web Server

http://www.securitytube.net

Vivek Ramachandran
Course Instructor

Module 3: Network Security

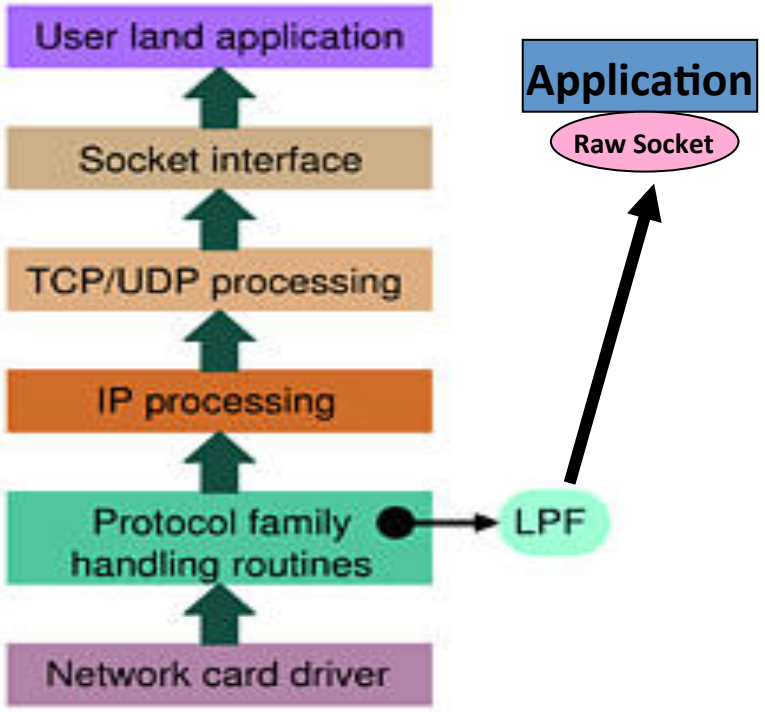
SecurityTube Python Scripting Expert

Part 4:
Packet Sniffing with Raw Sockets

http://www.securitytube.net

Vivek Ramachandran
Course Instructor

Raw Socket Basics



- Raw sockets provide a way to bypass the whole network stack traversal of a packet and deliver it directly to an application
- Multiple ways to create raw sockets. We will concentrate on the PF_PACKET interface

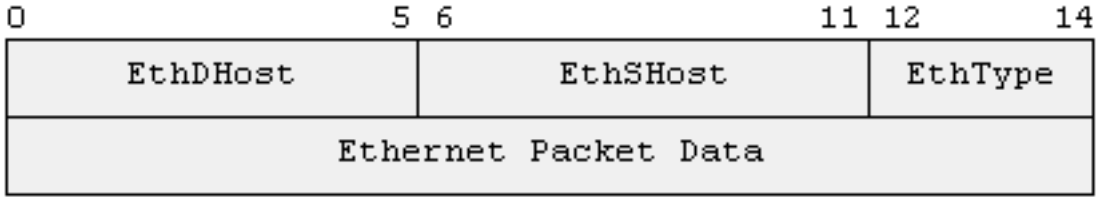
PF_PACKET

- It is a software interface to send/receive packets at layer 2 of the OSI i.e. device driver
- All packets received will be complete with all headers and data
- All packets sent will be transmitted without modification by the kernel to the medium
- Supports filtering using Berkley Packet Filters

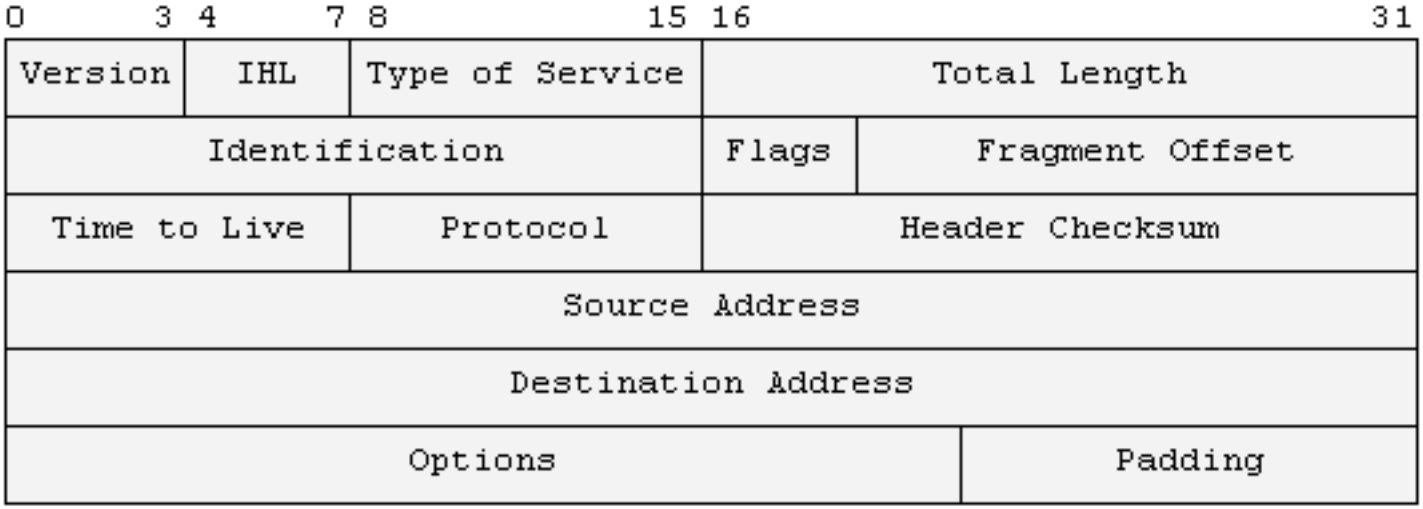
Creating Raw Sockets

- use the socket module
- read packets
- interpret and analyze them
- can send out responses as well

Understanding Packet Headers



Ethernet Header
(14 bytes)



IP Header
(20 bytes)

Extracting Binary Data into Variables

Format	C Type	Python type	Standard size
x	pad byte	no value	
c	char	string of length 1	1
b	signed char	integer	1
B	unsigned char	integer	1
?	_Bool	bool	1
h	short	integer	2
H	unsigned short	integer	2
i	int	integer	4
I	unsigned int	integer	4
l	long	integer	4
L	unsigned long	integer	4
q	long long	integer	8
Q	unsigned long long	integer	8
f	float	float	4
d	double	float	8
s	char[]	string	
p	char[]	string	
P	void *	integer	

- struct.unpack()
- returns tuple format
- First character indicates Byte Ordering
  - Network Byte Order is Big-Endian

Exercise

- Create a Packet Sniffer using Raw Sockets which can parse TCP packets
  - parse individual fields
- Create a sniffer which uses a filter to only print details of an HTTP packet (TCP, Port 80)
  - Also dump the data

Module 3: Network Security

SecurityTube Python Scripting Expert

Part 4:
End of Packet Sniffing with Raw Sockets

http://www.securitytube.net

Vivek Ramachandran
Course Instructor

Module 3: Network Security

SecurityTube Python Scripting Expert

Part 5:
Packet Injection with Raw Sockets

http://www.securitytube.net

Vivek Ramachandran
Course Instructor

Packet Injection

- Ability to inject raw packets into the network
- powerful as we can stimulate responses from the network
- packet construction not scalable with raw sockets

Inject Random Stuff ☺

- If you can inject random data into the network
  - you know you can send anything then ☺

Exercise

- Send an ARP Request Packet using Raw Sockets
- Verify the same with Tcpdump or Wireshark

Module 3: Network Security

SecurityTube Python Scripting Expert

End of Part 5:
Packet Injection with Raw Sockets

http://www.securitytube.net

Vivek Ramachandran
Course Instructor

Module 3: Network Security

SecurityTube Python Scripting Expert

Part 6:
Packet Sniffing with Scapy

http://www.securitytube.net

Vivek Ramachandran
Course Instructor

3rd Party Libraries

- Raw sockets are painful to use and not too portable across OSs
- Use of 3rd Part libs:
  - pylibpcap
  - pycapy
  - pypcap
  - Impacket
  - Scapy
- We will use Scapy in this course because it's the most powerful and flexible among all other options

Scapy

- Interactive mode or use as library
- Can be used for packet sniffing and forging
- Tons of protocols already implemented
- Allows to build “reactive” tools

http://www.secdev.org/projects/scapy/doc/usage.html

Protocol Layers Available

- ls()
- ls(IP)
- IP().show()
- lsc()
- conf

Sniffing with Scapy

```
sniff(count=0, store=1, offline=None, prn=None, lfilter=None, L2socket=None, timeout=None, opened_socket=None, stop_fi
lter=None, *arg, **karg)
    Sniff packets
    sniff([count=0,] [prn=None,] [store=1,] [offline=None,] [lfilter=None,] + L2ListenSocket args) -> list of packets

      count: number of packets to capture. 0 means infinity
      store: wether to store sniffed packets or discard them
        prn: function to apply to each packet. If something is returned,
             it is displayed. Ex:
             ex: prn = lambda x: x.summary()
    lfilter: python function applied to each packet to determine
             if further action may be done
             ex: lfilter = lambda x: x.haslayer(Padding)
    offline: pcap file to read packets from, instead of sniffing them
    timeout: stop sniffing after a given time (default: None)
    L2socket: use the provided L2socket
    opened_socket: provide an object ready to use .recv() on
    stop_filter: python function applied to each packet to determine
                 if we have to stop the capture after this packet
                 ex: stop_filter = lambda x: x.haslayer(TCP)
```

Exercise

- Create a Packet sniffer with Scapy for HTTP protocol and print out
  - the HTTP Headers
  - Data in GET/POST

- Create a Wi-Fi Sniffer and print out the unique SSIDs of the Wi-Fi networks in your vicinity

Module 3: Network Security

SecurityTube Python Scripting Expert

End of Part 6:
Packet Sniffing with Scapy

http://www.securitytube.net

Vivek Ramachandran
Course Instructor

Module 3: Network Security

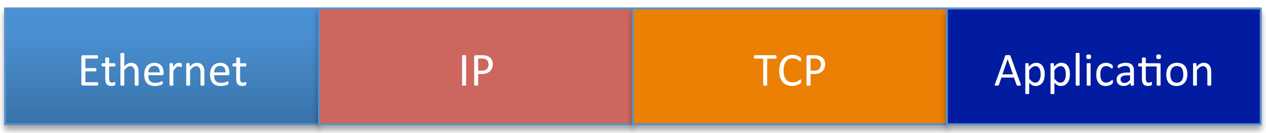
SecurityTube Python Scripting Expert

Part 7:
Packet Injection with Scapy

http://www.securitytube.net

Vivek Ramachandran
Course Instructor

Packet Forging with Scapy




```
Ether()   /   IP()   /   TCP ()   /   Data
```

Send Packets

- sendp - Send packets at Layer 2. Need to give right interface etc.

- send – Send packets at Layer 3. Routing decided based on local table
  - loop on the same packet
  - inter : time interval in seconds

Send and Receive at Layer 2 and 3

- layer 3
  - sr()
    - returns answers and unanswered packets
  - sr1()
    - returns only answer or sent packet
- layer 2
  - srp()
  - srp1()

Injected Packet Routing

- Uses local routes by default
- Can be overridden by Scapy
- Modified routes can be flushed
- Does not affect system routes

Module 3: Network Security

SecurityTube Python Scripting Expert

End of Part 7:
Packet Injection with Scapy

http://www.securitytube.net

Vivek Ramachandran
Course Instructor

Module 3: Network Security

SecurityTube Python Scripting Expert

Part 8:
Programming with Scapy

http://www.securitytube.net

Vivek Ramachandran
Course Instructor

Using Scapy as a Library

- from scapy.all import Ether, IP, TCP, sr1
- use as you please in your program ☺

- Really! Really Powerful!

ARP Scanner

- Create ARP Request packets for the local subnet
- Send and Receive Responses
- Get Results and Publish

* Find out how to get the local subnet automatically

Exercise

- Create a DNS poisoning tool similar to Dnsspoof using scapy
- Create a ARP MITM tool using scapy
- Create a TCP SYN Scanner using Scapy

Exercise

- Explore how to create a Fuzzer with Scapy

- Create a DNS Fuzzer with Scapy and try it against DNSspoof

Module 3: Network Security

SecurityTube Python Scripting Expert

Part 8:
End of Programming with Scapy

http://www.securitytube.net

Vivek Ramachandran
Course Instructor