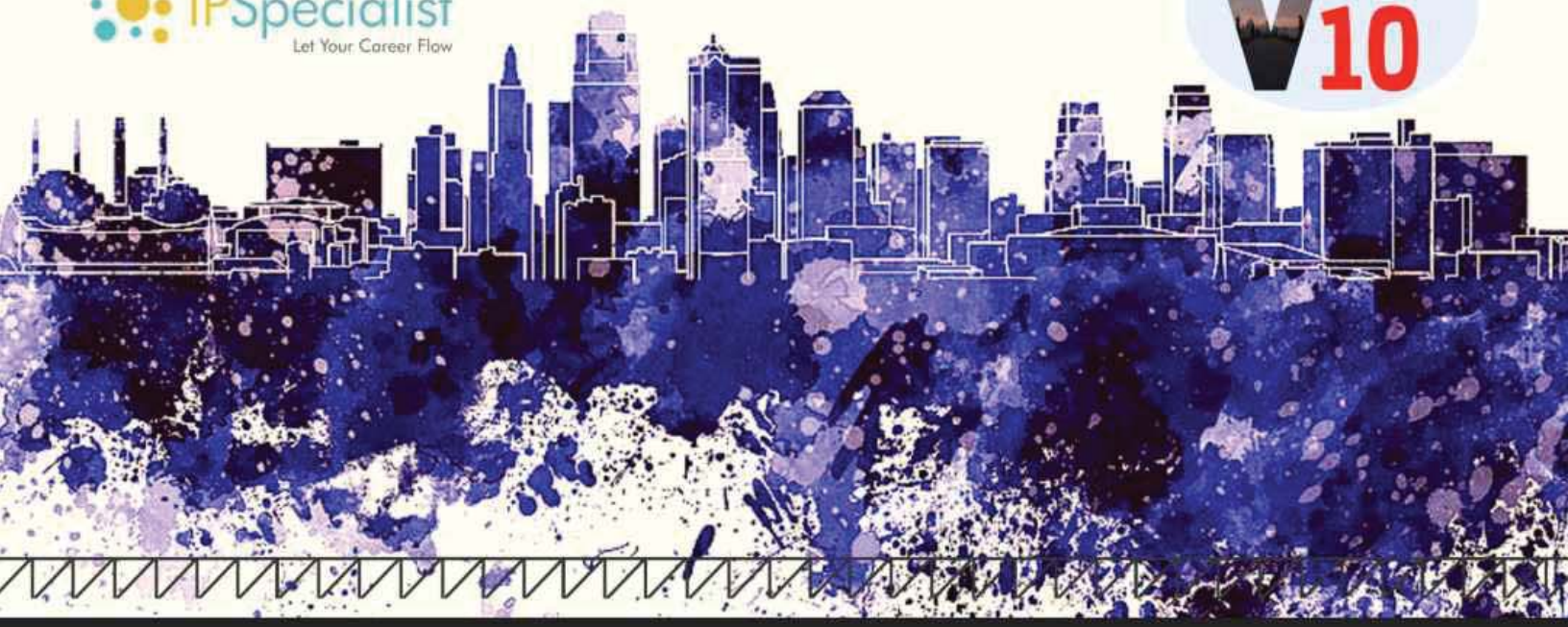
IPSpecialist
Let Your Career Flow

V10



CEH V10 EC-COUNCIL CERTIFIED ETHICAL HACKER

MOST DEMANDING COMPLETE HACKING GUIDE

EXAM: 312-50

C|EH
Certified Ethical Hacker

"To beat a hacker, you need to think like a hacker

MOST ADVANCED HACKING COURSE

www.ipspecialist.net

Chapter 15: SQL Injection

Technology Brief

In this chapter, Structured Query Language (SQL) injection is covered. SQL Injection is a popular and complex method of attack on web services, applications, and Databases. It requires deep knowledge about web application processes and its components such as databases and SQL. SQL Injection is basically insertion of malicious code or script by exploiting vulnerabilities to launch an attack powered by back-end components. This chapter will give information about concepts or SQL injection, types, methodology and defending techniques of SQL injection.

SQL Injection Concepts

SQL Injection

SQL Injection Attacks uses SQL websites or web applications. It relies on the strategic injection of malicious code or script into existing queries. This malicious code is drafted with the intention of revealing or manipulating data that is stored in the tables within the database.
SQL injection is a powerful and dangerous attack. It identifies the flaws and vulnerabilities in a website or application. The fundamental concept of SQL injection is to inject commands to reveal sensitive information from the database. Hence, it can result to a high profile attack.

The scope of SQL Injection

SQL Injection can be a great threat to a website or application. SQL injection impact can be measured by observing the following parameters that an attacker is intended to overcome: -

- Bypassing the Authentication
- Revealing sensitive information
- Compromised Data integrity
- Erasing the database
- Remote Code Execution

How SQL Query works

Injection of SQL query will be executed on the server and replied by the response. For example, following SQL query is requested to the server.

```
SELECT * FROM [Orders]
```

These commands will reveal all information stored in the database "Orders" table. If an organization maintains records of their orders into a database, all information kept in this database table will be extracted by the command.

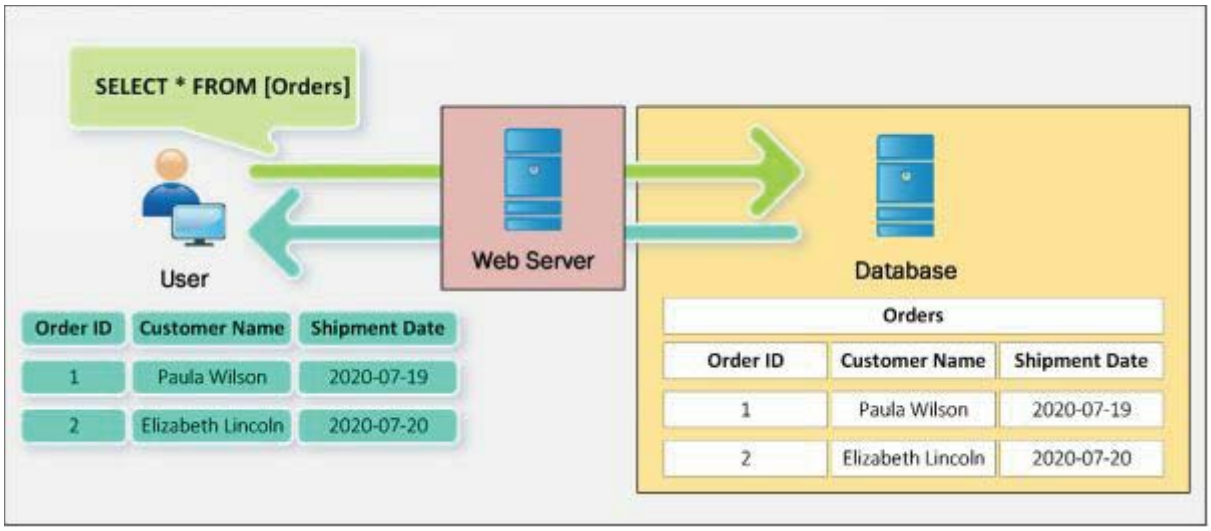


Figure 15-01 SQL Query working

SQL Delete Query

The DELETE statement is used to delete existing records in a table. To understand, consider a table "***Customers***" in a database. The following information is the table "***Customers***" is containing.

CustomerID	CustomerName	City
1	Maria Anders	London
2	Alfreds Futterkiste	Prague
3	Elizabeth Brown	Paris
4	Ana Trujillo	New York
5	Thomas Hardy	Boston

Table 15-01 Database before Delete Query

Execution of "delete" command will erase the record.

```
DELETE FROM Customers
WHERE CustomerName='Alfreds Futterkiste';
```

Now the database table will be like this: -

CustomerID	CustomerName	City
1	Maria Anders	London
3	Elizabeth Brown	Paris
4	Ana Trujillo	New York
5	Thomas Hardy	Boston

Table 15-02 Database After Delete Query

There are lots of SQL query commands that can be used. Above are some of the most common and effective commands that are being used for injection.

SQL Update Query

The UPDATE statement is used to modify the existing records in a table. For example, consider the following command.

```
UPDATE Customers
SET ContactName = 'IPSpecialist, City= 'Frankfurt'
WHERE CustomerID = 1;
```

Now the Database will be: -

CustomerID	CustomerName	City
1	IPSpecialist	Frankfurt
3	Elizabeth Brown	Paris
4	Ana Trujillo	New York
5	Thomas Hardy	Boston

Table 15-03 Database after Update Query

SQL Injection Tools

There are several tools available for SQL injection such as: -

- BSQL Hacker
- Marathon Tool
- SQL Power Injector
- Havij

Types of SQL Injection

SQL Injection can be classified into three major categories.

1. In-band SQLi
2. Inferential SQLi
3. Out-of-band SQLi

In-Band SQL Injection

In-Band SQL injection is a category which includes injection techniques using same communication channel to launch the injection attack and gather information from the response. In-Band Injection techniques include: -

1. Error-based SQL Injection
2. Union based SQL Injection

Error Based SQL Injection

Error-based SQL Injection is one of the in-band SQL Injection technique. It relies on error messages from the database server to reveal information about the structure of the database. Error-based SQL injection is very effective for an attacker to enumerate an entire database. Error messages are useful during the development phase to troubleshoot issues. These messages should be disabled when an application website is live. Error Based SQL Injection can be performed by the following techniques: -

- System Stored Procedure
- End of Line Comment
- Illegal / Logically incorrect Query
- Tautology

Union SQL Injection

Union-based SQL injection is another In-band SQL injection technique that involves the UNION SQL operator to combine the results of two or more SELECT statements into a single result.

```
SELECT <column_name(s)> FROM <table_1>
UNION
SELECT <column_name(s)> FROM <table_2>;
```

Inferential SQL Injection (Blind Injection)

In an Inferential SQL Injection, no data is transferred from a Web

application; the, i.e., attacker is unable to see the result of an attack hence referred as a Blind injection. The attacker just observed the behavior of the server. The two types of inferential SQL Injection Are Blind-Boolean-based SQL injection and Blind-time-based SQL injection.

Boolean Exploitation Technique

Blind SQL injection is a technique of sending a request to the database. the response does not contain any data from database however attacker observe the HTTP response. By evaluating the responses, an attacker can infer whether injection was successful or unsuccessful, as the response will be either true or false however it will not contain any data from the database.

Out-of-band SQL Injection

Out-of-band SQL injection is the injection technique that uses different channels to launch the injection and gather the responses. It requires some features being enabled such as DNS or HTTP requests on database server hence it is not very common.

SQL Injection Methodology

Information Gathering and SQL Injection Vulnerability Detection

In the information gathering phase, Collect the information about the web application, operating system, database and the structure of the components. Evaluation of extracted information will be helpful to identify the vulnerabilities to exploit. Information can be gathered by using different tools and techniques such as injecting codes into the input fields to observe the response of error messages. Evaluation of input field, hidden fields, get and post requests, cookies, string values and detailed error messages can reveal enough information to initial injection attack.

Launch SQL Injection Attacks

Appropriate SQL injection attack from the category cab be initiate just after gathering the information about the structure of database and vulnerabilities found. By exploiting them, the injection can be successful. SQL injection attacks such as Union SQL injection, Error-based SQL injection, Blind SQL injection and other can be used to extract information from the database such as extracting Database name, tables, columns, rows, and fields. The injection can also have intended for bypassing the authentication.

Advanced SQL Injection

Advanced SQL injection may include an enumeration of databases like MySQL, MSSQL, MS Access, Oracle, DB2, or Postgre SQL, tables and column in order to identify privilege level of users, account information of database administrator and database structure disclosure. it also includes passwords and hashes grabbing, and transferring the database to the remote machine.

Evasion Techniques

Evading IDS

In order to secure database, isolated deployment in a secure network location with an intrusion detection system (IDS) is recommended. IDS keep monitoring the network and host traffic as well as a database application. The attacker has to evade IDS to access the database, for this, it uses different evading techniques. For example, IDS using Signature-based Detection system compare the input strings against the signature to detect intrusion. Now all you have to do is to evade the signature-based detection.

Types of Signature Evasion Techniques

Different techniques as mentioned below are used to evade: -

- Inserting Inline Comment in between keywords
- Character Encoding
- String Concatenation
- Obfuscated Codes
- Manipulating White Spaces
- Hex Encoding
- Sophisticated Matches

Counter-measures

In order to mitigate SQL injection attacks, there are several detection tools available which can be used. These tools perform testing of website and applications and report the data, issues and remediation actions. Some of these advanced tools also offer technical description as the issue.

Lab 15-1: Using IBM Security AppScan Standard

Procedure:

1. Download and Install IBM Security AppScan Standard.
2. Open the Application
3. Select **Create New Scan**

Figure 15-02 IBM Security AppScan Standard

4. Select Scan template, the Regular scan will start a new scan. In our case, we are using a predefined template **demo.testfire.net**

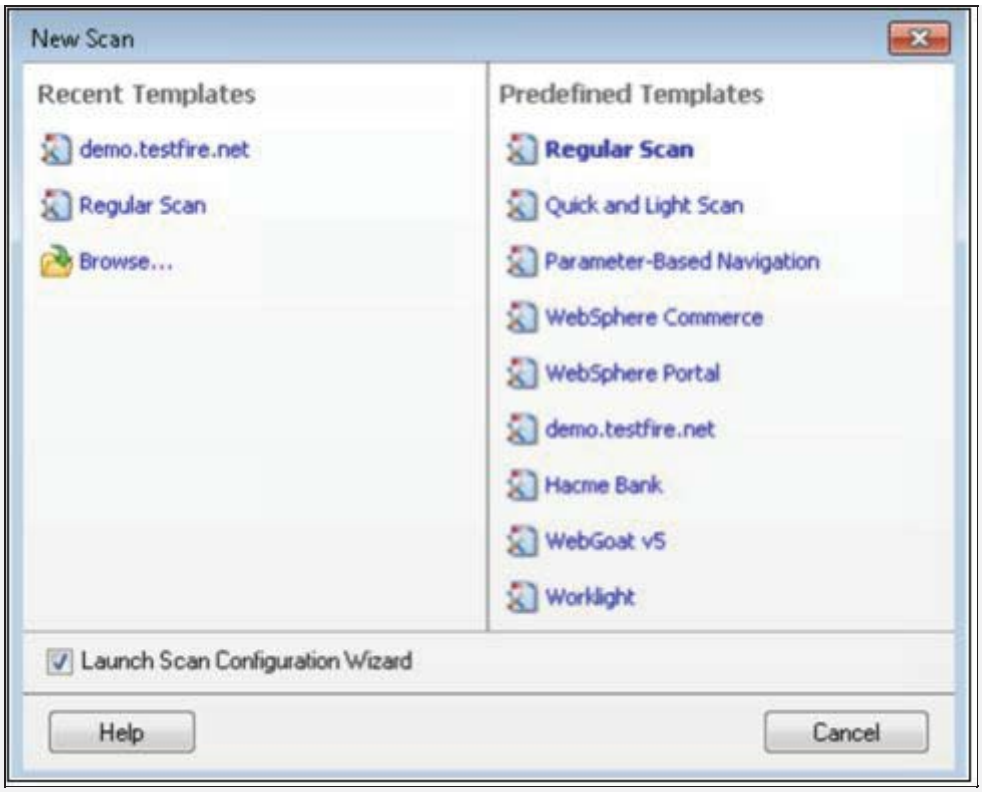


Figure 15-03 New Scan

5. Click **Next**
6. If you want to edit the configuration, Click **Full scan configuration**

Figure 15-04 Configuration Wizard

7. Click **Next**

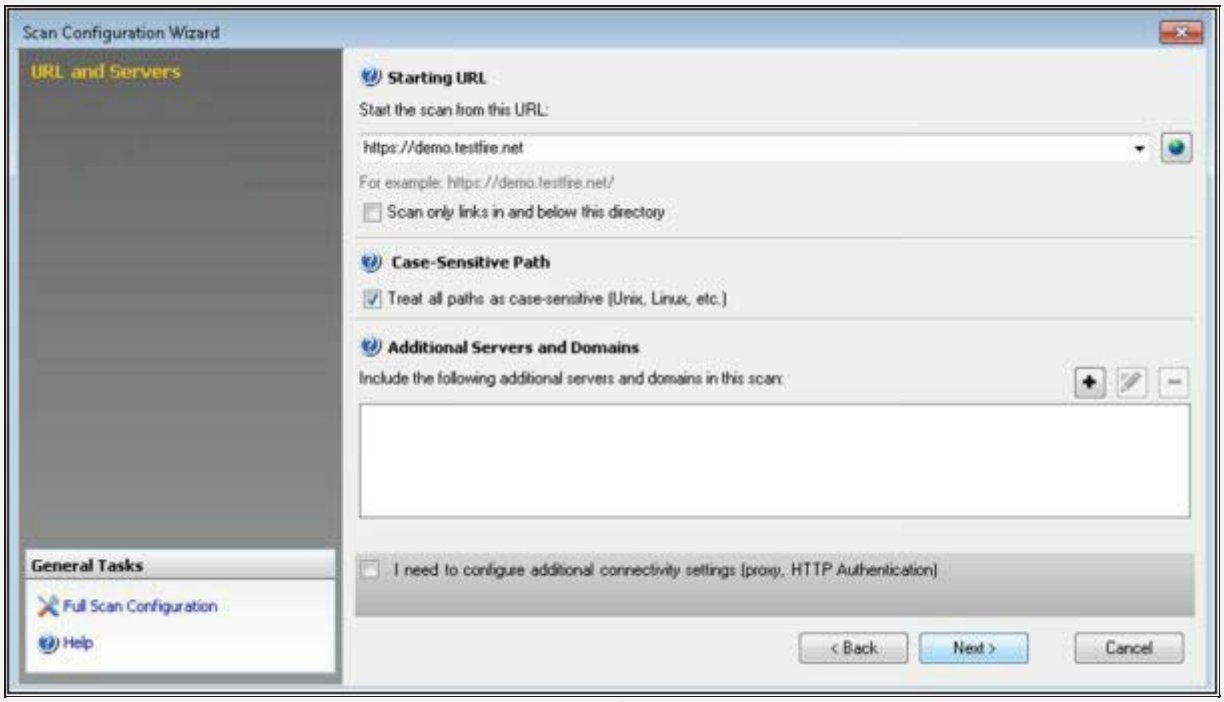


Figure 15-05 Configuration Wizard

8. Select Login Method

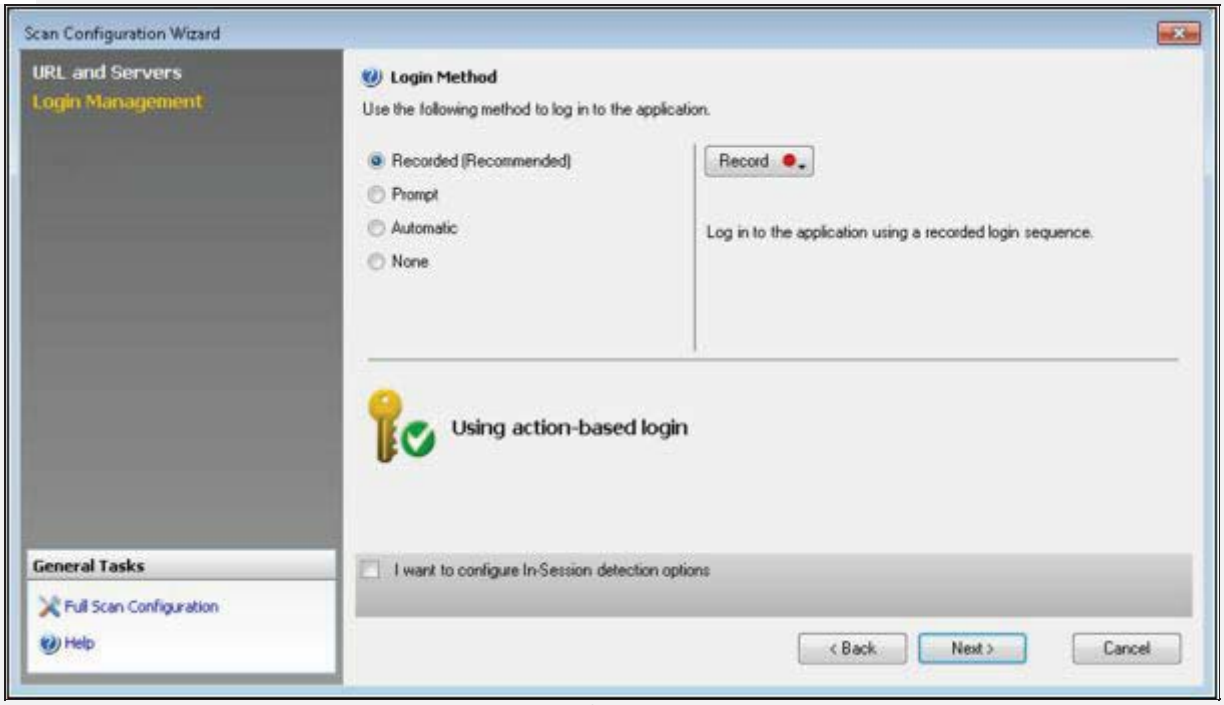


Figure 15-06 Configuring Login Method

9. Select **Test Policy**
0. Click **Next**

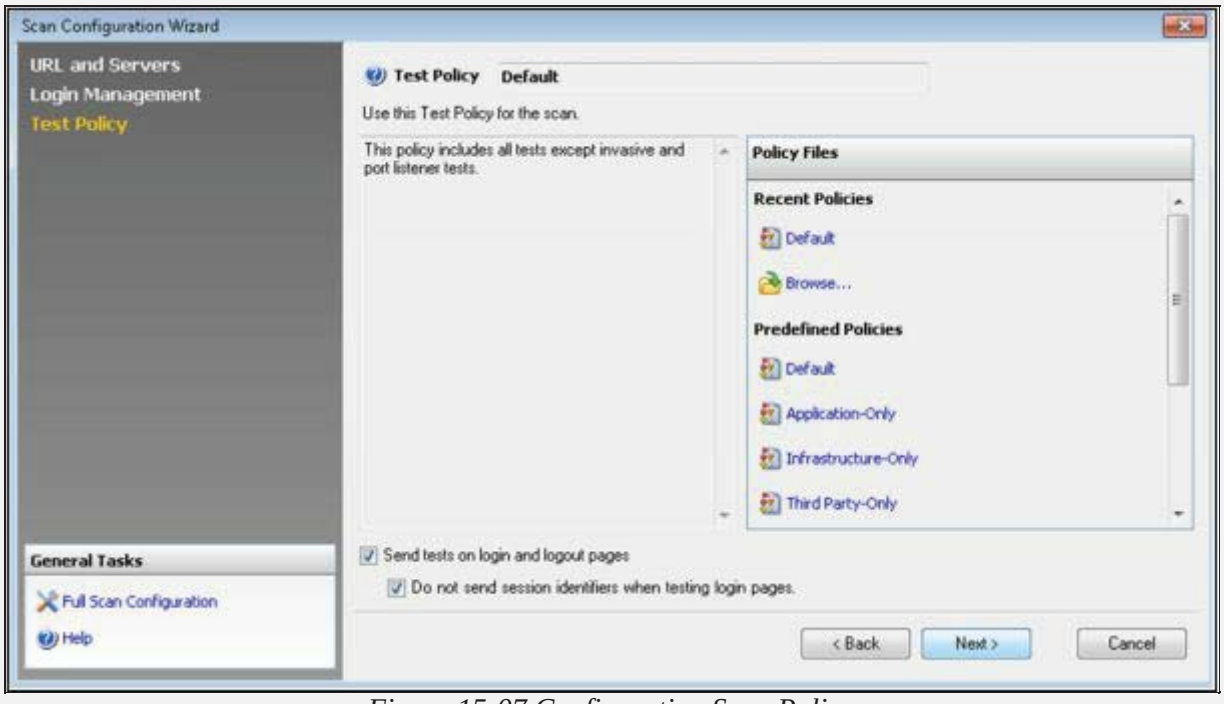


Figure 15-07 Configuration Scan Policy

1. Select how do you want to start the scan.
2. Click **Finish**

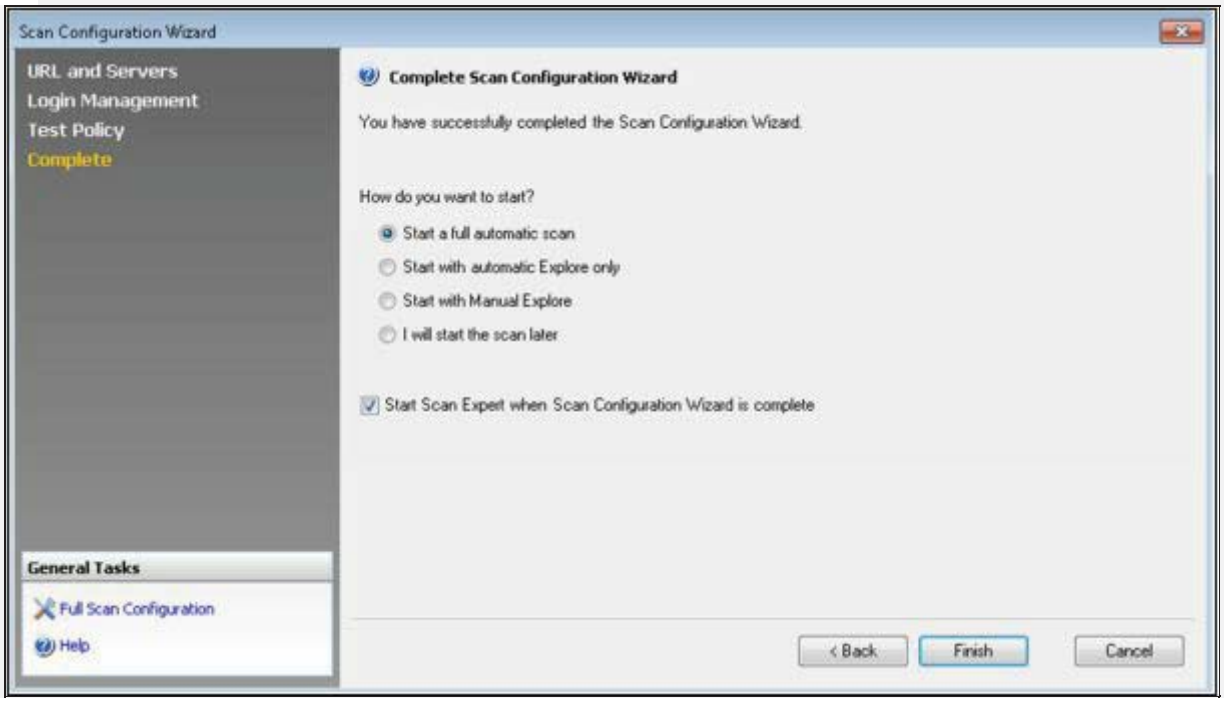


Figure 15-08 Configuration Wizard

3. It may ask to save the file in the directory.
4. Start the scan

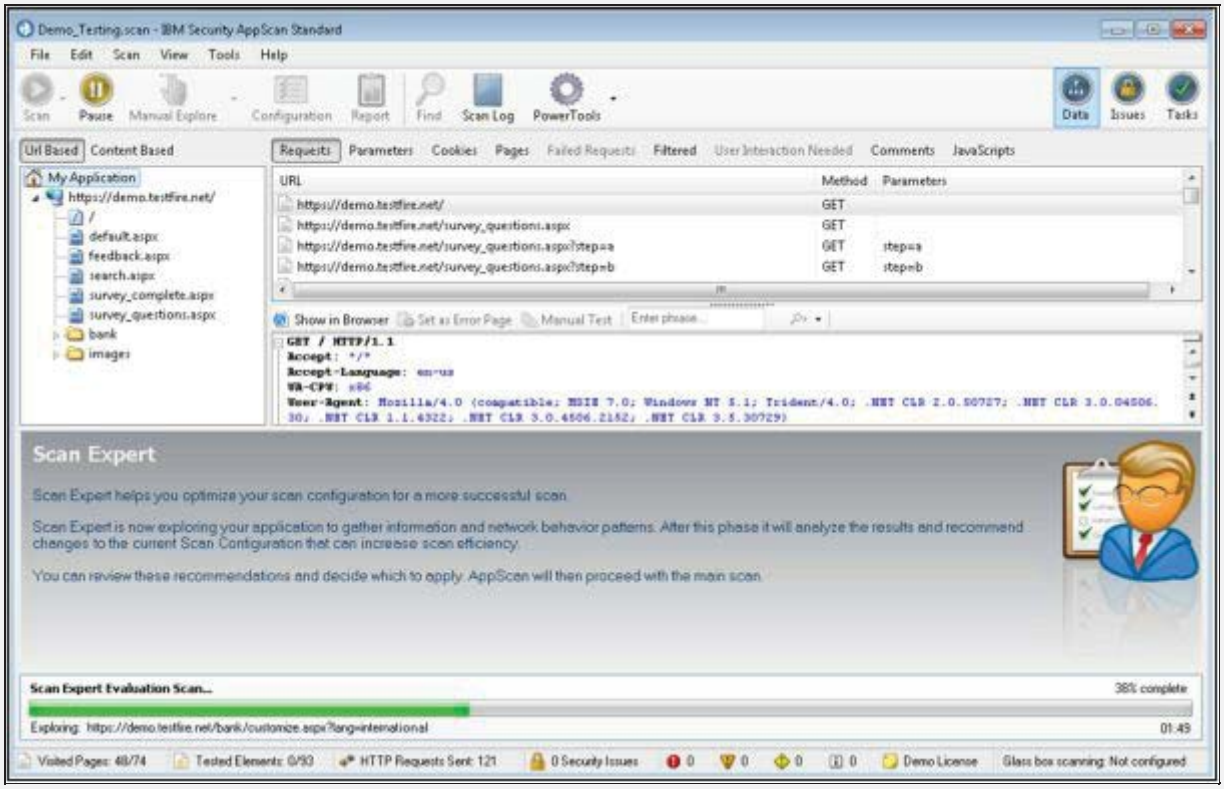


Figure 15-09 Scanning

5. Data Pane showing Data scanned during the process.
6. In our case, we are using a demo testing; it does not find any issue.

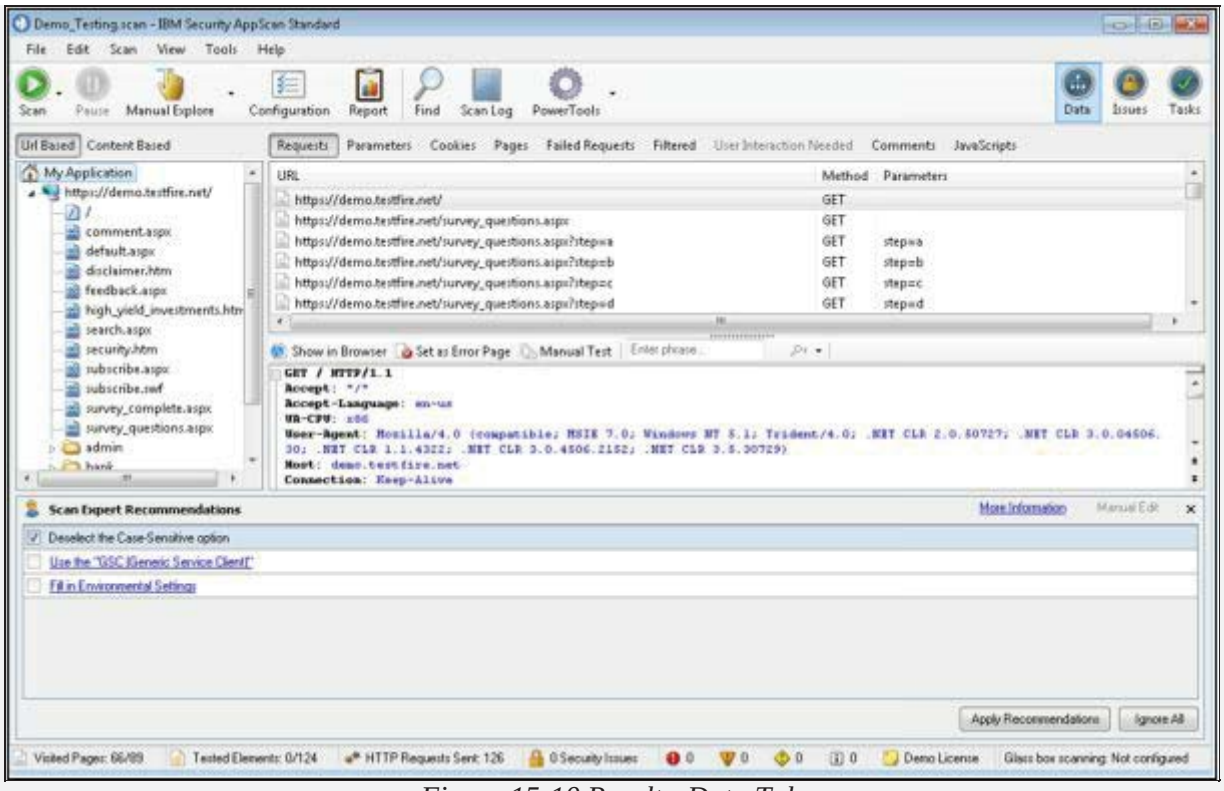


Figure 15-10 Result - Data Tab

7. In case, if it found the issue, Issue section will show the detected issues list.
8. To explore, click the security issue, it will show the details.

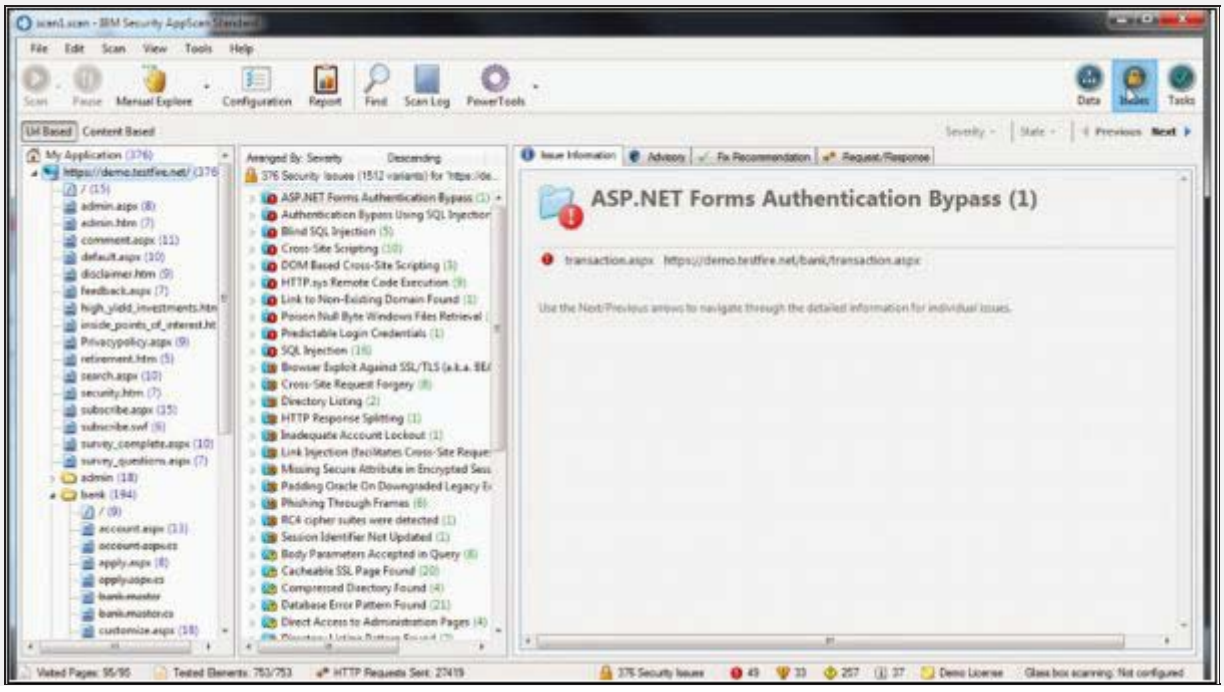


Figure 15-11 Issue Tab

9. In case, when you have detected any issue, Task section will show the recommended remediation actions.

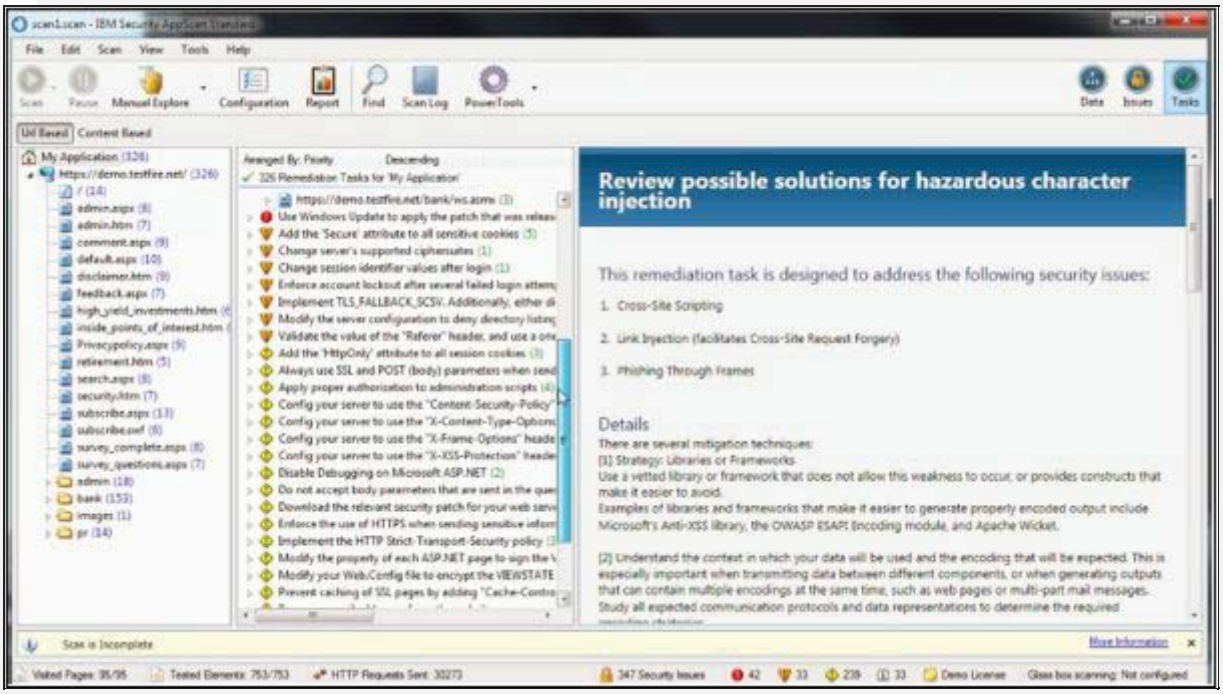
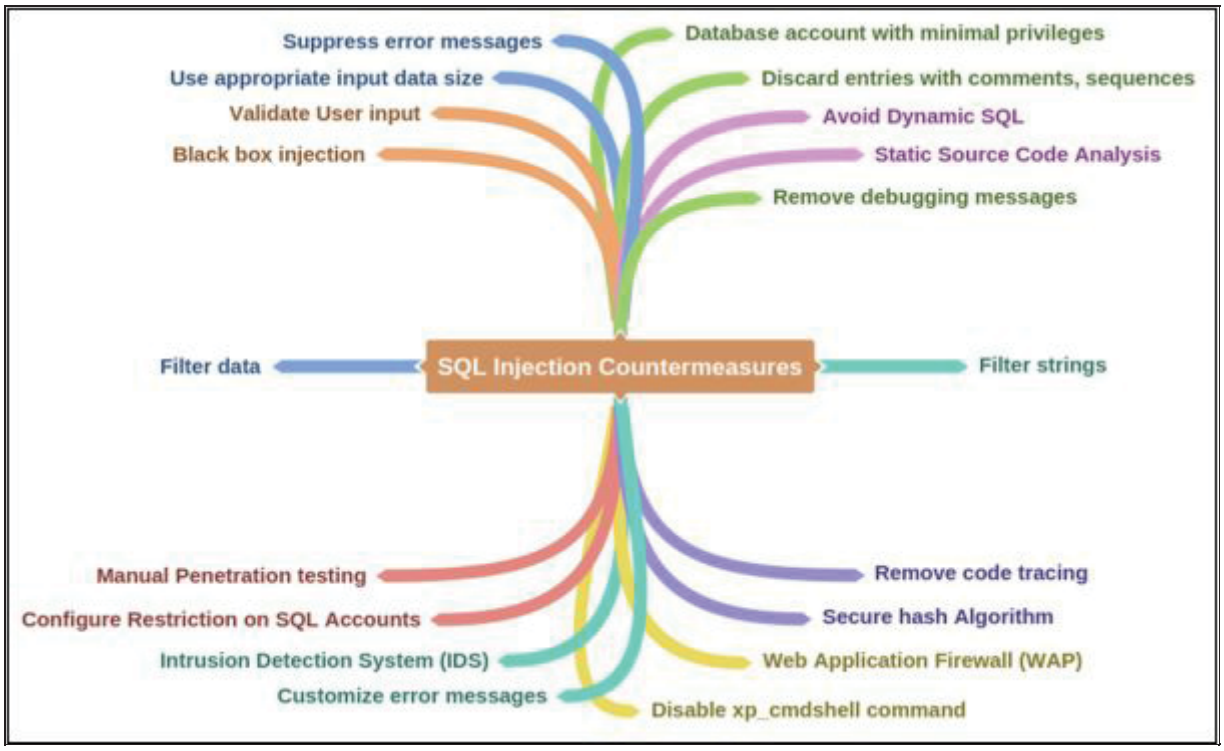


Figure 15-12 Task Tab

Mind Map


Suppress error messages
Use appropriate input data size
Validate User input
Black box injection
Database account with minimal privileges
Discard entries with comments, sequences
Avoid Dynamic SQL
Static Source Code Analysis
Remove debugging messages
Filter data
SQL Injection Countermeasures
Filter strings
Manual Penetration testing
Configure Restriction on SQL Accounts
Intrusion Detection System (IDS)
Customize error messages
Remove code tracing
Secure hash Algorithm
Web Application Firewall (WAP)
Disable xp_cmdshell command