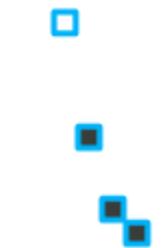




Module 13

Hacking Web Servers

Module Objectives



Module Objectives

- Understanding Web Server Concepts
- Understanding Web Server Attacks
- Understanding Web Server Attack Methodology
- Overview of Web Server Attack Tools
- Understanding Different Web Server Attack Countermeasures
- Understanding Patch Management Concepts
- Overview of Web Server Security Tools
- Overview of Web Server Penetration Testing

Module Flow

1

Web Server Concepts

2

Web Server Attacks

3

Web Server Attack Methodology

4

Web Server Attack Tools

5

Counter-measures

6

Patch Management

7

Web Server Security Tools

8

Web Server Pen Testing

Web Server Operations

- A web server is a computer system that **stores, processes** and **delivers web pages** to clients via HTTP

Components of a Web Server

- Document Root:** Stores critical HTML files related to the web pages of a domain name that will be served in response to the requests
- Server Root:** Stores server's configuration, error, executable and log files
- Virtual Document Tree:** Provides storage on a different machine or disk after the original disk is filled up
- Virtual Hosting:** Technique of hosting multiple domains or websites on the same server
- Web Proxy:** Proxy server that sits in between the web client and web server to prevent IP blocking and maintain anonymity

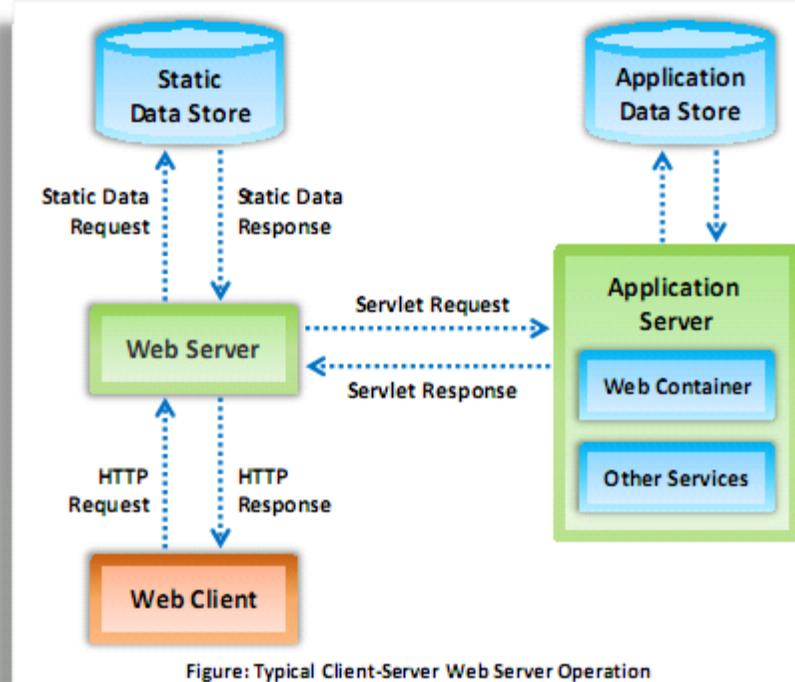
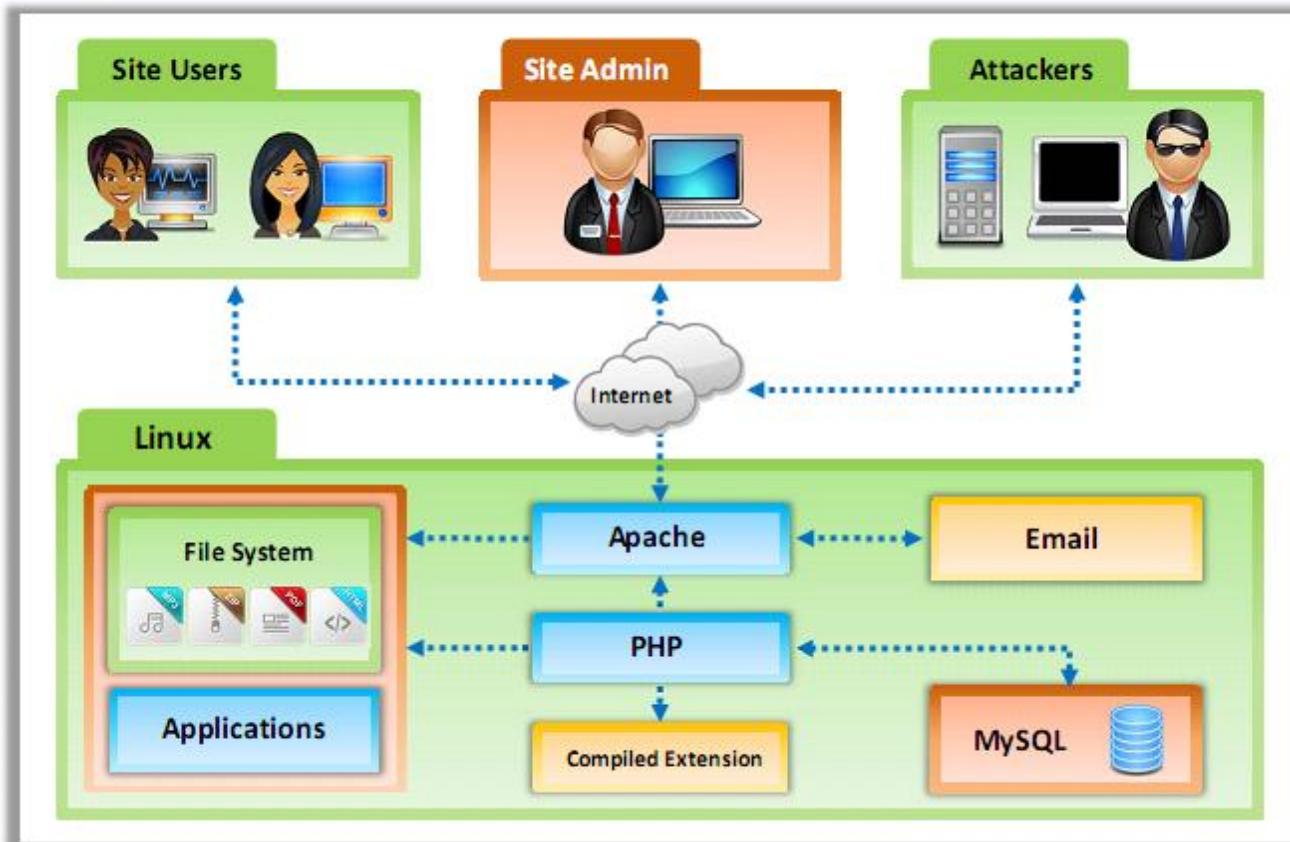
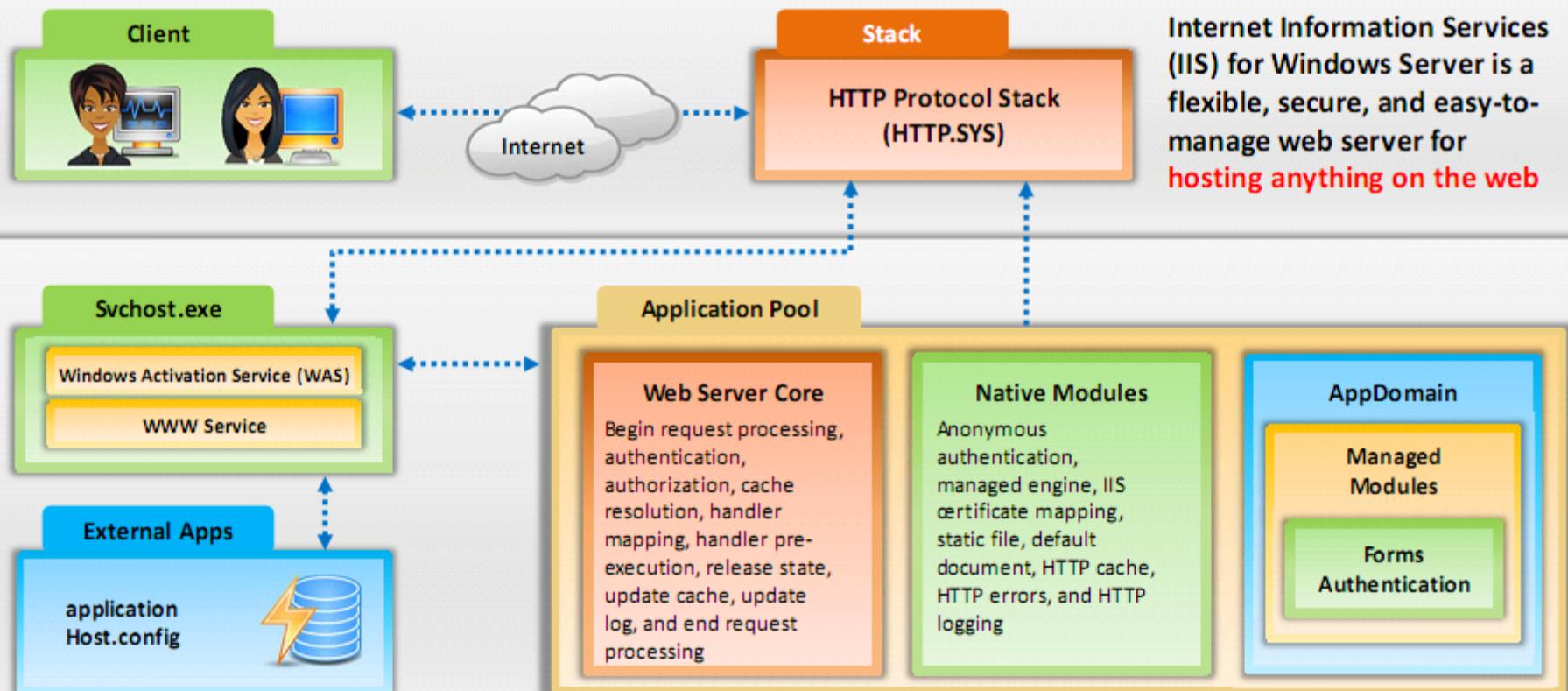


Figure: Typical Client-Server Web Server Operation

Open Source Web Server Architecture

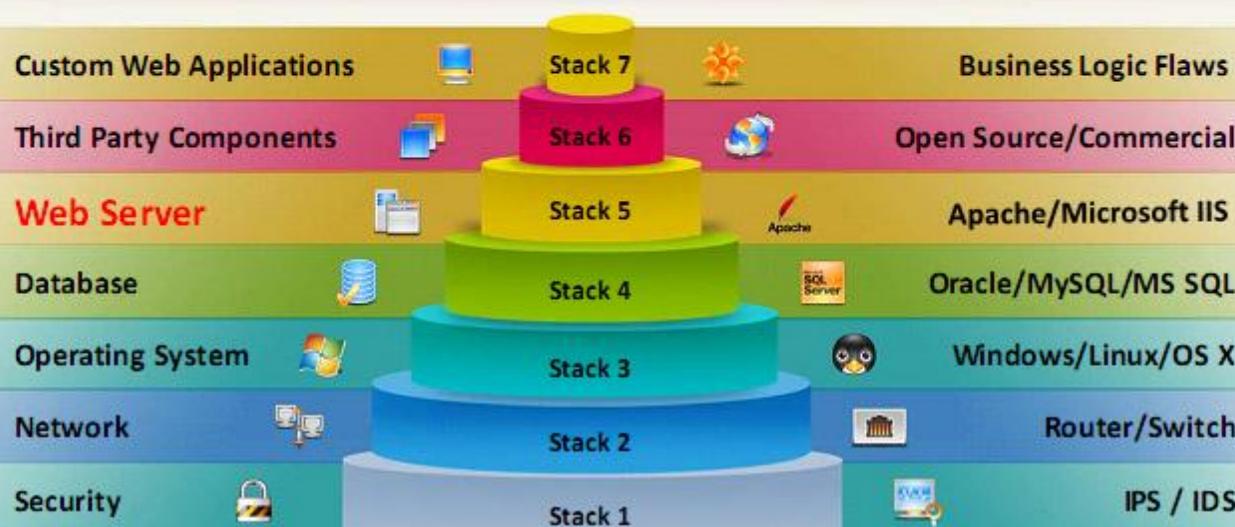


IIS Web Server Architecture



Web Server Security Issue

- Attackers usually target **software vulnerabilities** and configuration errors to compromise web servers
- Network** and **OS level attacks** can be well defended using proper network security measures such as firewalls, IDS, etc. However, web servers are accessible from anywhere on the web, which makes them **more vulnerable** to attacks



Why Web Servers Are Compromised?

- Improper file and directory permissions
- Installing the server with default settings
- Unnecessary services enabled, including content management and remote administration
- Security conflicts with business ease-of-use case
- Lack of proper security policy, procedures, and maintenance
- Improper authentication with external systems
- Default accounts with their default passwords, or no passwords
- Unnecessary default, backup, or sample files
- Misconfigurations in web server, operating systems, and networks
- Bugs in server software, OS, and web applications
- Misconfigured SSL certificates and encryption settings
- Administrative or debugging functions that are enabled or accessible on web servers
- Use of self-signed certificates and default certificates



Impact of Web Server Attacks

1

Compromise of user account

2

Website defacement

3

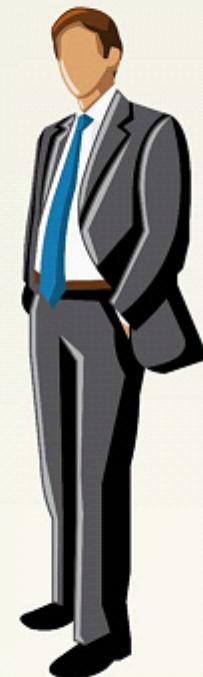
Secondary attacks from the website

4

Root access to other applications or servers

5

Data tampering and data theft



Module Flow

1

Web Server Concepts

2

Web Server Attacks

3

Web Server Attack Methodology

4

Web Server Attack Tools

5

Counter-measures

6

Patch Management

7

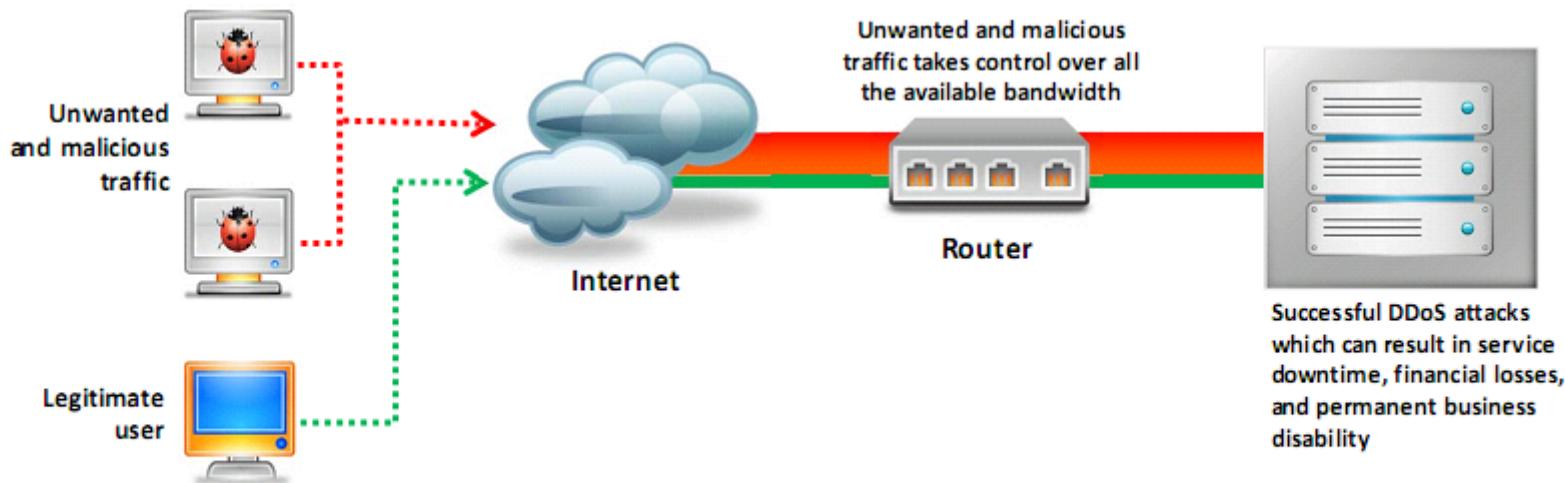
Web Server Security Tools

8

Web Server Pen Testing

DoS/DDoS Attacks

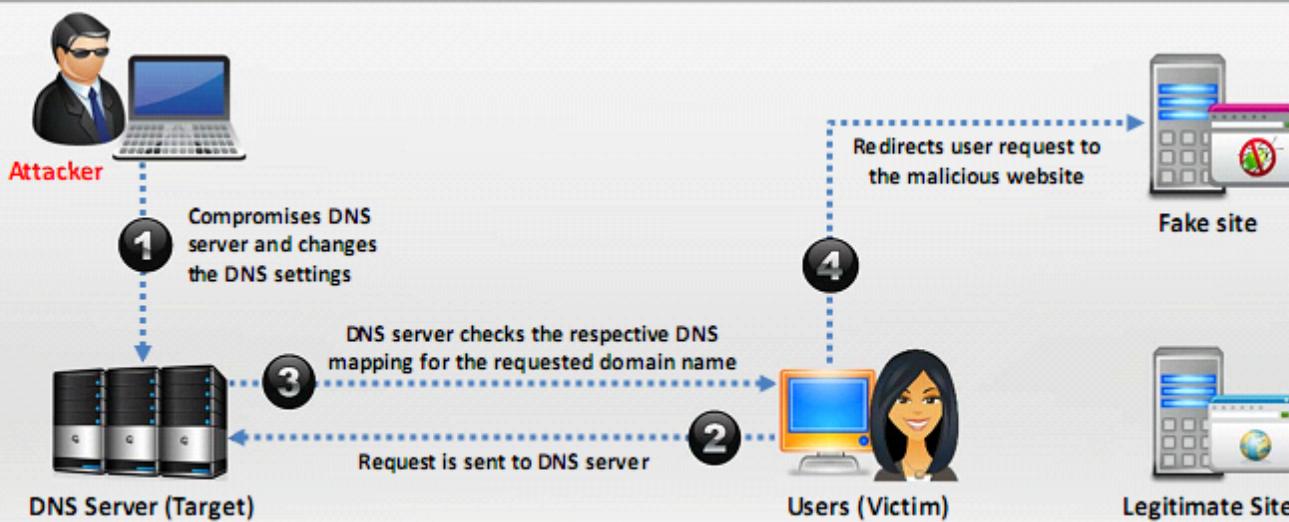
- Attackers may send numerous **fake requests** to the web server which results in the **web server crashing** or becoming unavailable to the legitimate users
- Attackers may target **high profile web servers** such as banks, credit card payment gateways, government owned services, etc. to **steal user credentials**



DNS Server Hijacking

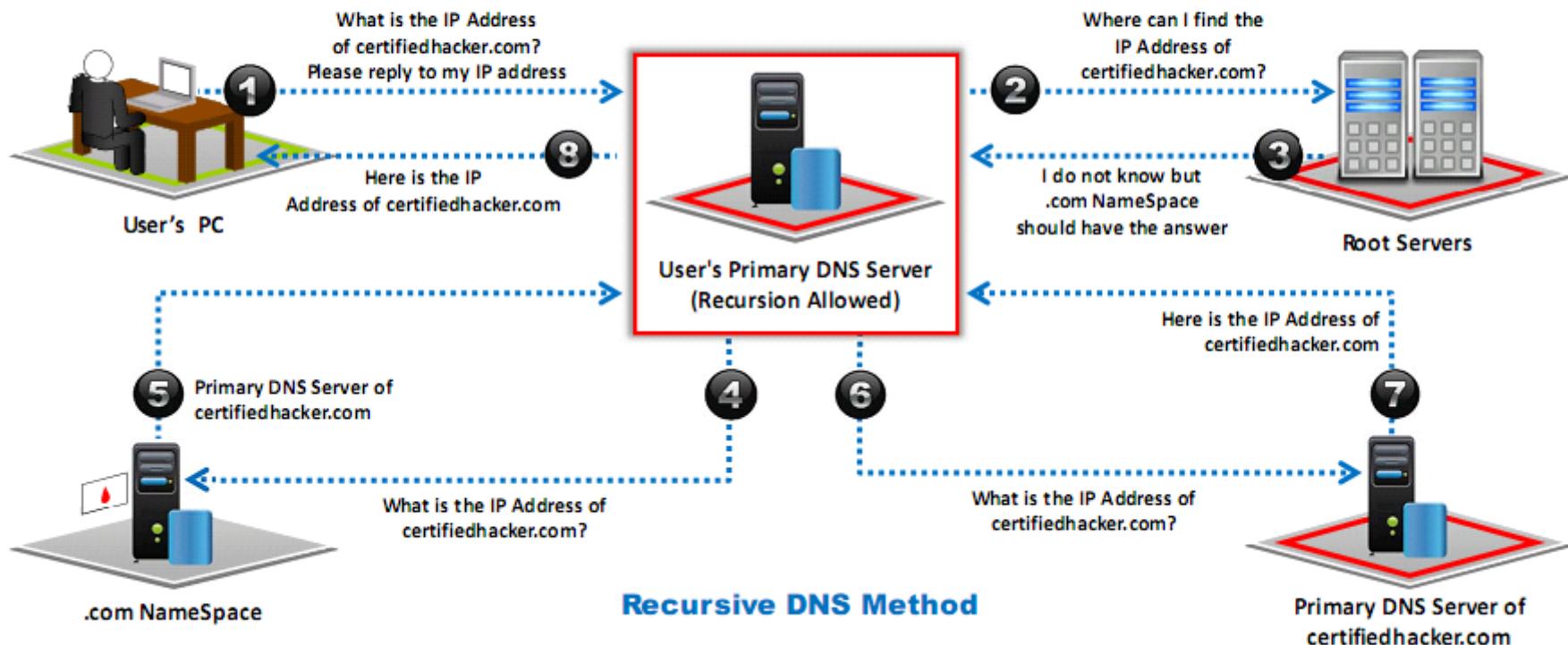


Attacker compromises DNS server and **changes the DNS settings** so that all the requests coming towards the target web server are redirected to his/her own malicious server



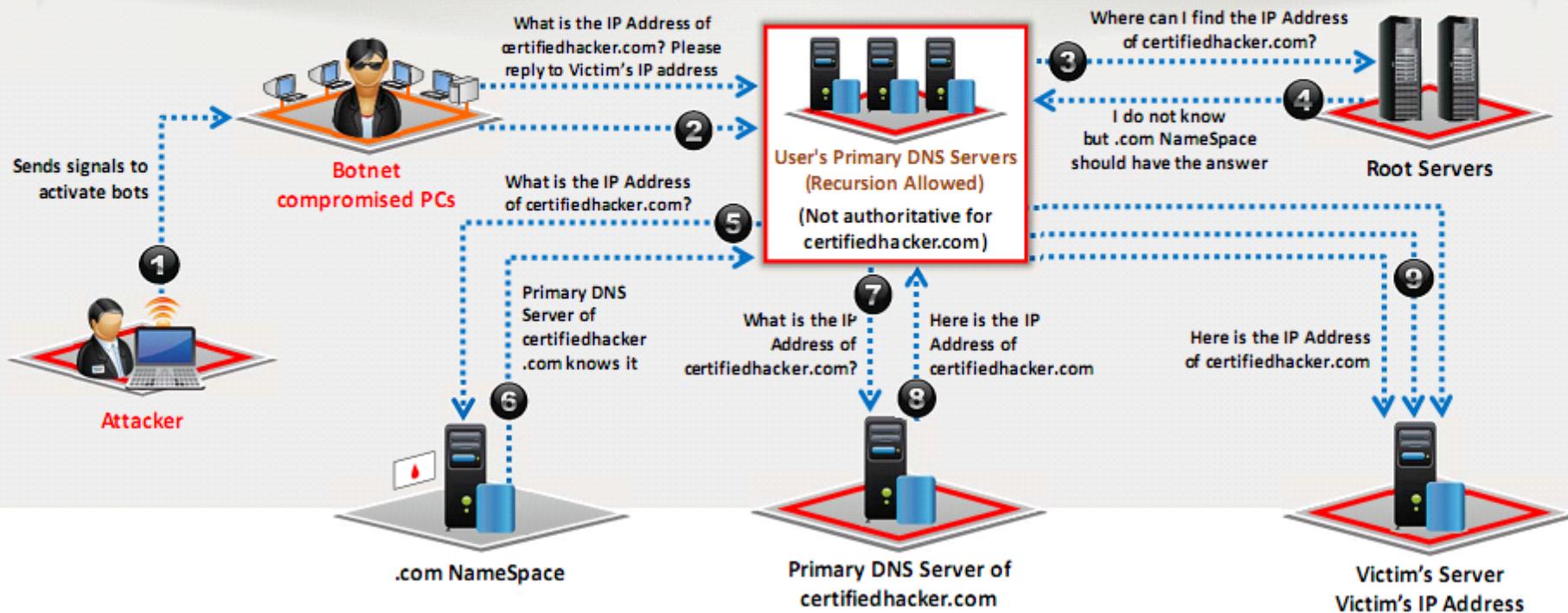
DNS Amplification Attack

- Attacker takes advantage of **DNS recursive method** of DNS redirection to perform DNS amplification attack



DNS Amplification Attack (Cont'd)

Attacker uses compromised PCs with **spoofed IP addresses** to amplify the DDoS attacks on victims' DNS server by exploiting DNS recursive method

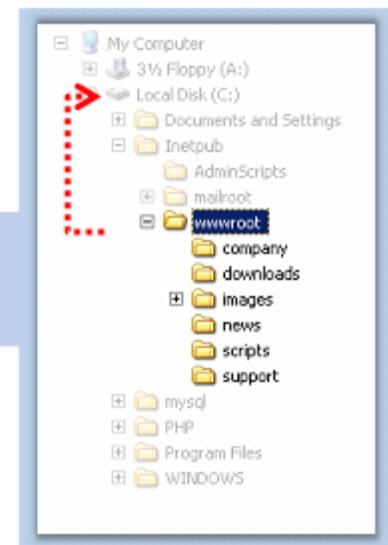


Directory Traversal Attacks

- In directory traversal attacks, attackers use **../ (dot-dot-slash)** sequence to access restricted directories outside of the web server root directory
- Attackers can use **trial and error method** to navigate outside of the root directory and access sensitive information in the system



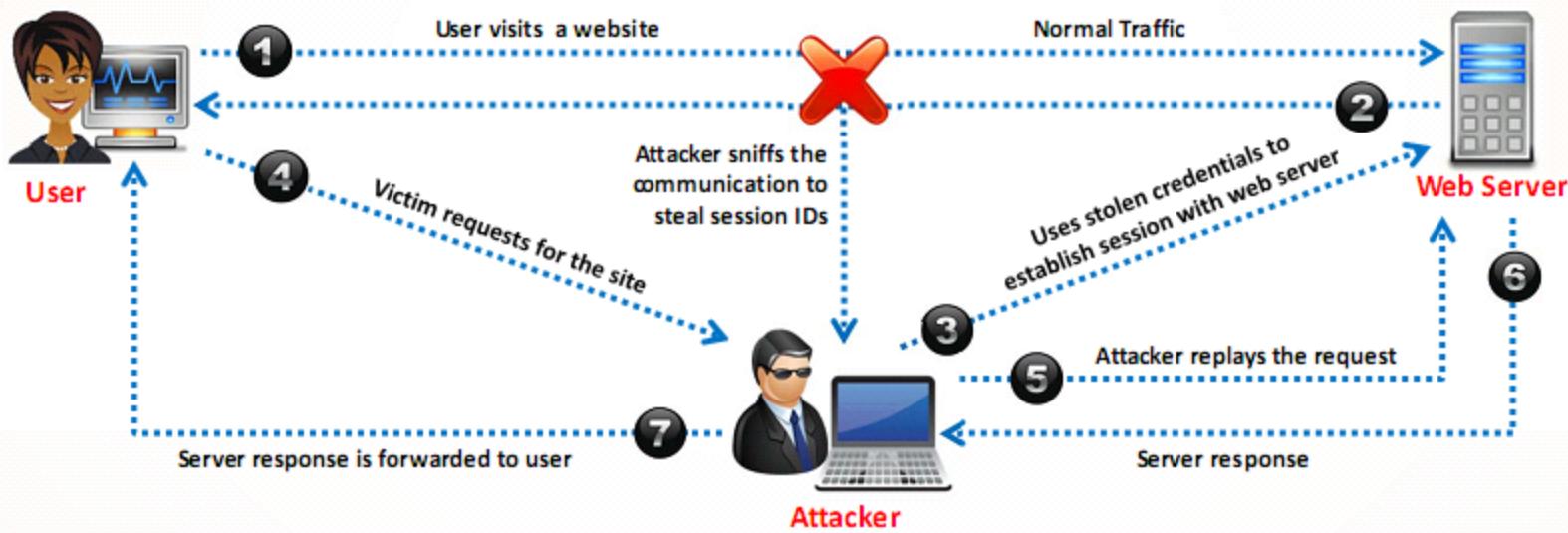
```
Volume in drive C has no label.  
Volume Serial Number is D45E-9FEE  
  
Directory of C:\  
  
06/02/2017 11:31 AM      1,024 .rnd  
09/28/2017 06:43 PM      0 123.text  
05/21/2017 03:10 PM      0 AUTOEXEC.BAT  
09/27/2017 08:54 PM  <DIR>    CATALINA_HOME  
05/21/2017 03:10 PM      0 CONFIG.SYS  
08/11/2017 09:16 AM  <DIR>    Documents and Settings  
09/25/2017 05:25 PM  <DIR>    Downloads  
08/07/2017 03:38 PM  <DIR>    Intel  
09/27/2017 09:36 PM  <DIR>    Program Files  
05/26/2017 02:36 AM  <DIR>    Snort  
09/28/2017 09:50 AM  <DIR>    WINDOWS  
09/25/2017 02:03 PM      569,344 WinDump.exe  
                           7 File(s)   570,368 bytes  
                           13 Dir(s) 13,432,115,200 bytes free
```



Man-in-the-Middle/Sniffing Attack

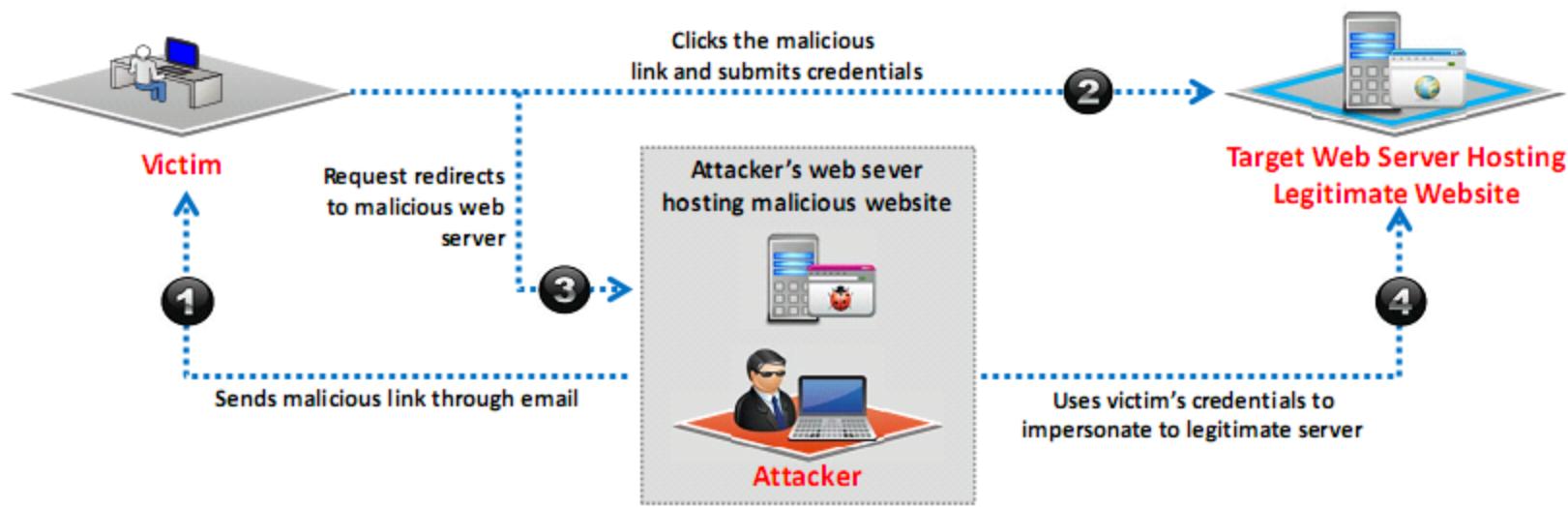
01 Man-in-the-Middle (MITM) attacks allow an attacker to access sensitive information by intercepting and altering communications between an end-user and web servers

02 Attacker acts as a proxy such that all the communication between the user and web server passes through him



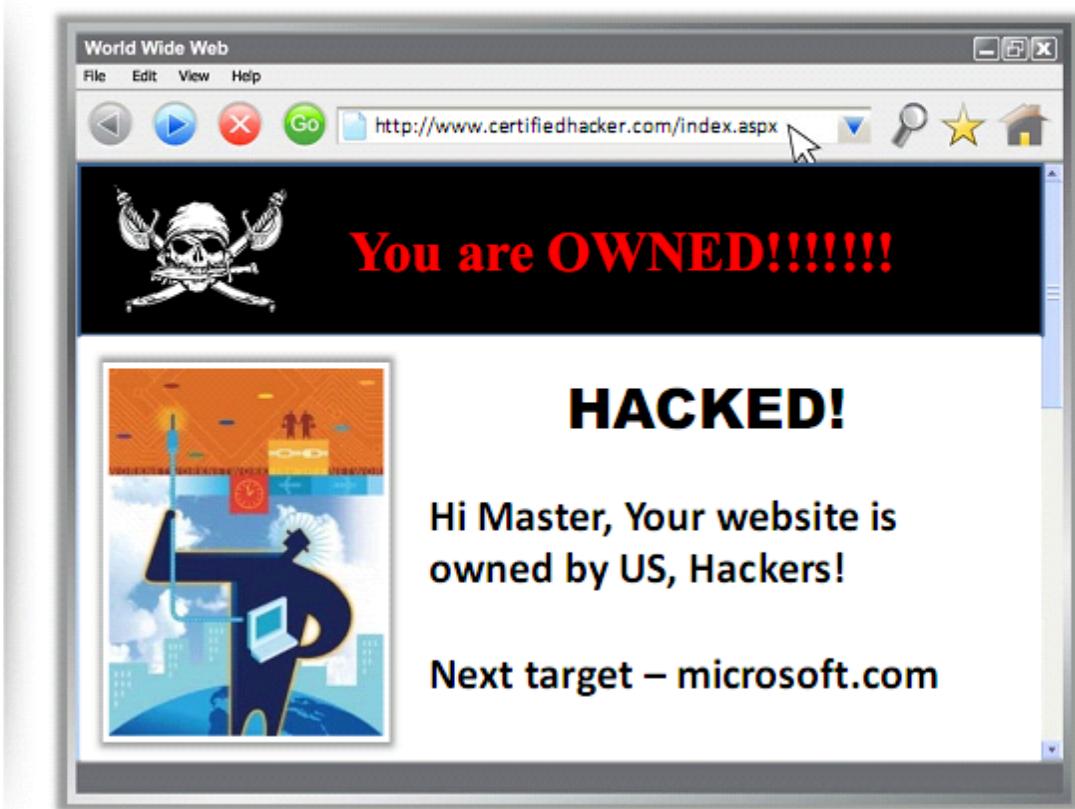
Phishing Attacks

- Attacker tricks user to submit **login details** for website that looks legitimate, but it redirect to the malicious website hosted on attacker web server
- Attacker **steals the credentials** entered and uses it to impersonate the user with the website hosted on the legitimate target server
- Attacker then can perform **unauthorized** or **malicious operations** with the website target server



Website Defacement

- Web defacement occurs when an intruder **maliciously alters the visual appearance of a web page** by inserting or substituting provocative, and frequently, offending data
- **Defaced pages expose visitors to some propaganda** or misleading information until the unauthorized changes are discovered and corrected
- Attackers use a variety of methods such as **MySQL injection** to access a site in order to deface it



Web Server Misconfiguration

- Server misconfiguration refers to **configuration weaknesses in web infrastructure** that can be exploited to launch various attacks on web servers such as directory traversal, server intrusion, and data theft

Web Server Misconfigurations

- Verbose Debug/Error Messages
- Anonymous or Default Users/Passwords
- Sample Configuration and Script Files
- Remote Administration Functions
- Unnecessary Services Enabled
- Misconfigured/Default SSL Certificates

Web Server Misconfiguration Examples

■ This configuration allows anyone to view the **server status** page, which contains detailed information about the current use of the web server, including information about the **current hosts** and requests being processed

httpd.conf file
on an **Apache** server

```
<Location /server-status>  
SetHandler server-status  
</Location>
```

■ This configuration gives **verbose error messages**



php.ini file

```
display_error = On  
log_errors = On  
error_log = syslog  
ignore_repeated_errors = Off
```

HTTP Response Splitting Attack



HTTP response splitting attack involves **adding header response data into the input field** so that the server splits the response into two responses



The attacker can **control the first response to redirect the user to a malicious website** whereas the other responses will be discarded by the web browser

Server Code

```
String author =  
request.getParameter(AUTHOR_PARAM);  
...  
Cookie cookie = new Cookie("author",  
author);  
cookie.setMaxAge(cookieExpiration);  
response.addCookie(cookie);
```



Input = Jason

HTTP/1.1 200 OK
...
Set-Cookie: author=Jason
...

Input = JasonTheHacker\r\nHTTP/1.1 200 OK\r\n

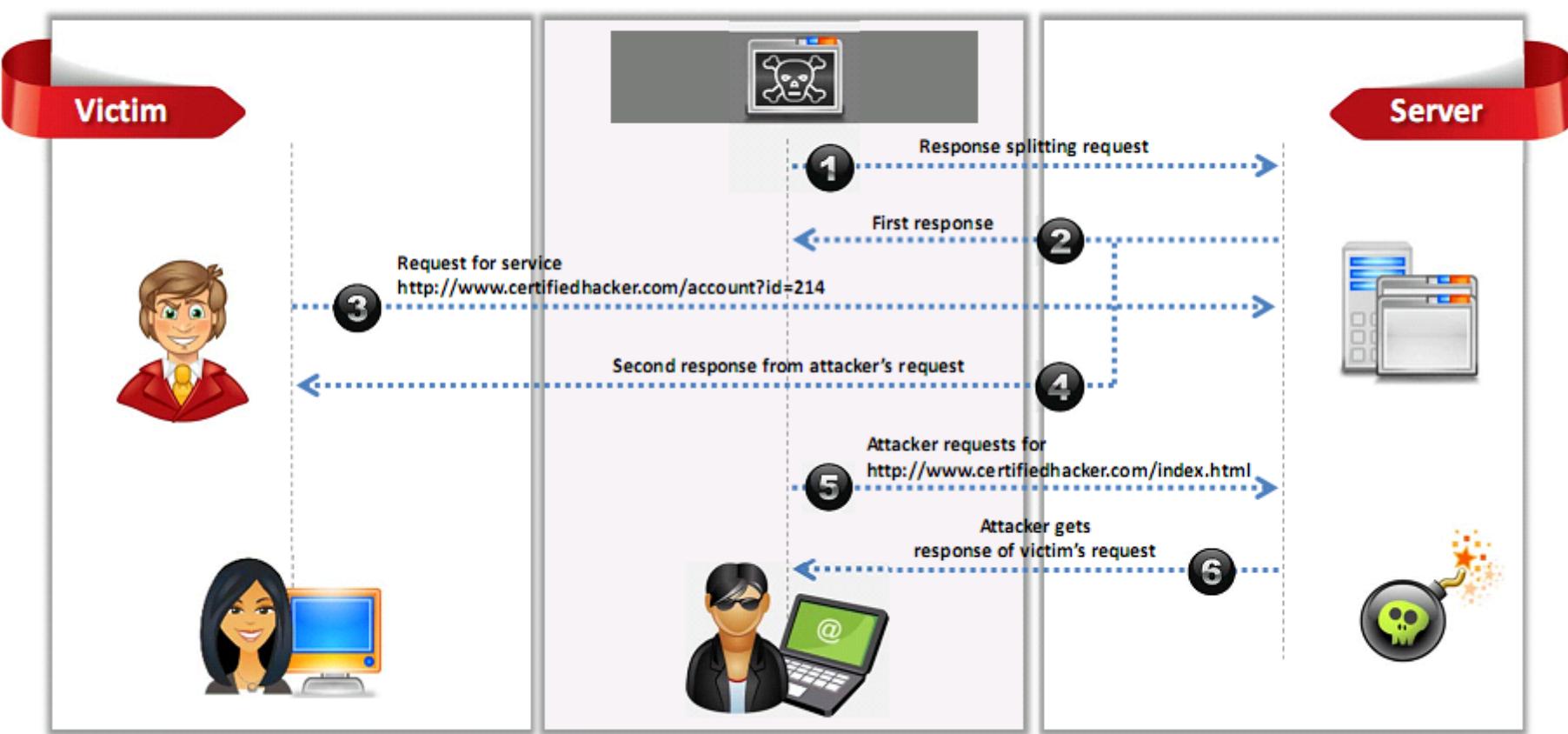
First Response (Controlled by Attacker)

Set-Cookie: author=JasonTheHacker
HTTP/1.1 200 OK
...

Second Response

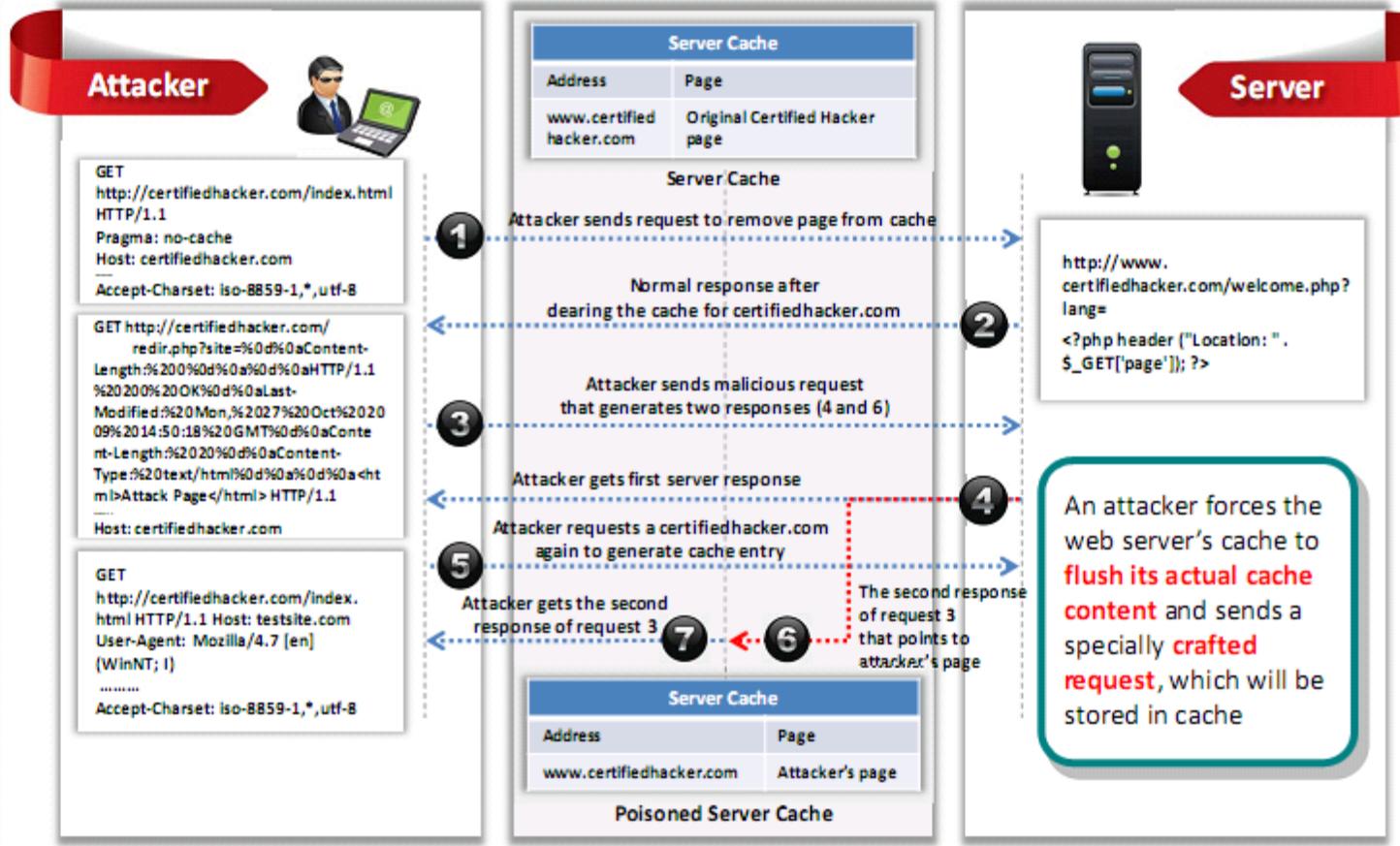
HTTP/1.1 200 OK
...

HTTP Response Splitting Attack (Cont'd)



Web Cache Poisoning Attack

- Web cache poisoning attacks the **reliability of an intermediate web cache source**
- In this attack, the attackers **swap cached content** for a random URL with infected content
- Users of the web cache source can **unknowingly use the poisoned content** instead of the true and secured content when requesting the required URL through the web cache



SSH Brute Force Attack

1

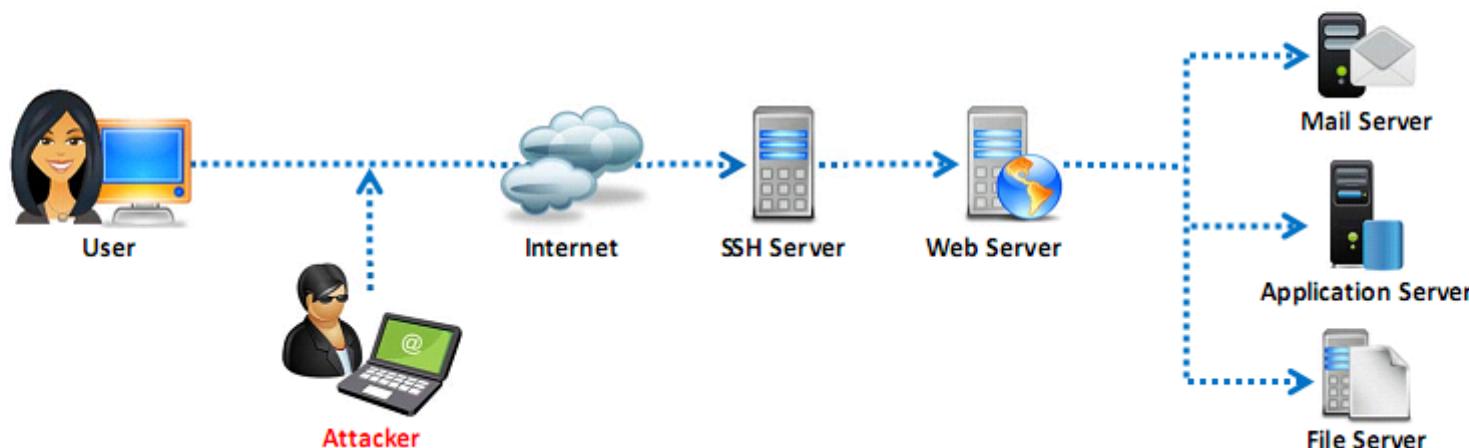
SSH protocols are used to create an **encrypted SSH tunnel** between two hosts in order to transfer unencrypted data over an insecure network

2

Attackers can brute force SSH login credentials to gain **unauthorized access** to a **SSH tunnel**

3

SSH tunnels can be used to **transmit malwares** and other exploits to victims without being detected



Web Server Password Cracking



- An attacker tries to exploit weaknesses to hack **well-chosen passwords**
- The most **common passwords** found are password, root, administrator, admin, demo, test, guest, qwerty, pet names, etc.

Attacker targets mainly for:

- SMTP servers
- Web shares
- SSH Tunnels
- Web form authentication cracking
- FTP servers



- Attackers use different methods such as **social engineering, spoofing, phishing**, using a Trojan Horse or virus, wiretapping, keystroke logging, etc.



- Many hacking attempts start with **cracking passwords** and prove to the web server that they are a valid user



- Passwords can be cracked **manually** by guessing or by performing dictionary, brute force, hybrid attacks using **automated tools** such as Cain & Abel, THC Hydra, etc.

Web Application Attacks

- Vulnerabilities in **web applications** running on a web server provide a broad attack path for web server compromise

Parameter/Form Tampering

Cookie Tampering

Unvalidated Input and File Injection Attacks

Session Hijacking

SQL Injection Attacks

Directory Traversal

Denial-of-Service (DoS) Attack

Cross Site Scripting (XSS) Attacks

Buffer Overflow Attacks

Cross Site Request Forgery (CSRF) Attack

Command Injection Attacks

Source Code Disclosure

Note: For complete coverage of web application attacks refer to Module 14: Hacking Web Applications

Module Flow

1

Web Server Concepts

2

Web Server Attacks

3

Web Server Attack Methodology

4

Web Server Attack Tools

5

Counter-measures

6

Patch Management

7

Web Server Security Tools

8

Web Server Pen Testing

Web Server Attack Methodology

Information Gathering

01

Web Server Footprinting

02

Website Mirroring

03

Vulnerability Scanning

04

Session Hijacking

05

Web Server Passwords Hacking

06

Information Gathering

1

- Information gathering involves collecting information about the **targeted company**

2

- Attackers search the **Internet, newsgroups, bulletin boards**, etc. for information about the company

3

- Attackers use **Whois.net, Whois Lookup**, etc. tools and query the Whois databases to get details such as a domain name, an IP address, or an autonomous system number

The screenshot shows the WHOIS.NET website interface. At the top, it says "Your Domain Starting" followed by a search bar containing "ebay.com". Below the search bar is a link "Whois Lookup — Domain Names Search, Registration". On the right side, there is a section titled "WHOIS LOOKUP" with the message "ebay.com is already registered*". Below this message is a detailed list of domain registration information:

Domain Name: EBAY.COM
Registry Domain ID: 1959284_DOMAIN_COM-VRSN
Registrar WHOIS Server: whois.markmonitor.com
Registrar URL: http://www.markmonitor.com
Updated Date: 2017-01-26T16:49:38Z
Creation Date: 1995-08-04T04:00:00Z
Registry Expiry Date: 2018-08-03T04:00:00Z
Registrar: MarkMonitor Inc.
Registrar IANA ID: 292
Registrar Abuse Contact Email: abusecomplaints@markmonitor.com
Registrar Abuse Contact Phone: +1.2083895740
Domain Status: clientDeleteProhibited https://icann.org/epp#clientDeleteProhibited
Domain Status: clientTransferProhibited https://icann.org/epp#clientTransferProhibited
Domain Status: clientUpdateProhibited https://icann.org/epp#clientUpdateProhibited
Domain Status: serverDeleteProhibited https://icann.org/epp#serverDeleteProhibited
Domain Status: serverTransferProhibited https://icann.org/epp#serverTransferProhibited
Domain Status: serverUpdateProhibited https://icann.org/epp#serverUpdateProhibited
Name Server: A1.VERISIGNWNS.COM
Name Server: A2.VERISIGNWNS.COM
Name Server: A3.VERISIGNWNS.COM
Name Server: NS1.P47.DYNECT.NET
Name Server: NS2.P47.DYNECT.NET
Name Server: NS3.P47.DYNECT.NET
Name Server: NS4.P47.DYNECT.NET
DNSSEC: unsigned
URL of the ICANN Whois Inaccuracy Complaint Form: https://www.icann.org/wicf/

>>> Last update of whois database: 2017-10-10T11:12:09Z <<<

<https://www.whois.net>

Note: For complete coverage of information gathering techniques refer to Module 02: Footprinting and Reconnaissance

Information Gathering from Robots.txt File

- The robots.txt file contains the **list of the web server directories and files** that the web site owner wants to hide from web crawlers
- An attacker can simply request Robots.txt file from the URL and retrieve sensitive information such as **root directory structure, content management system information, etc.**, about the target website



```
robots.txt - Notepad
File Edit Format View Help
User-agent: *
Disallow: /en-us/windows/si/matrix.html
Disallow: /en-us/windows/si/matrix.html
Disallow: /*/security/search-results.aspx?
Disallow: /*/music/*/search/
Disallow: /*/search/
Disallow: /*/music/*/Search/
Disallow: /*/Search/
Disallow: /*/newsearch/
Disallow: *action=catalogsearch&
Allow: /*/store/*/search/
Allow: /*/store/*/layout/
Allow: /*/store/music/groove-music-pass/*
Allow: *action=catalogsearch&catalog_mode=grid&page=2$
Allow: *action=catalogsearch&catalog_mode=grid&page=3$
Allow: *action=catalogsearch&catalog_mode=grid&page=4$
Allow: *action=catalogsearch&catalog_mode=grid&page=5$
Allow: *action=catalogsearch&catalog_mode=grid&page=6$
Allow: *action=catalogsearch&catalog_mode=grid&page=7$
Allow: *action=catalogsearch&catalog_mode=grid&page=8$
Allow: *action=catalogsearch&catalog_mode=list&page=2$
Allow: *action=catalogsearch&catalog_mode=list&page=3$
Allow: *action=catalogsearch&catalog_mode=list&page=4$
Allow: *action=catalogsearch&catalog_mode=list&page=5$
Allow: *action=catalogsearch&catalog_mode=list&page=6$
Allow: *action=catalogsearch&catalog_mode=list&page=7$
Allow: *action=catalogsearch&catalog_mode=list&page=8$
Disallow: *action=accessorysearch&product=*&
Allow: *action=accessorysearch&product=*
Disallow: *action=accessorysearch&
```

Web Server Footprinting/Banner Grabbing

1

- Gather **valuable system-level data** such as account details, operating system, software versions, server names, and database schema details

2

- Telnet** a web server to footprint a web server and gather information such as server name, server type, operating systems, applications running, etc.

3

- Use tool such as **Netcraft**, **httprecon**, and **ID Serve** to perform footprinting

Search Web by Domain

Explore 1,094,729 web sites visited by users of the Netcraft Toolbar

3rd November 2017

Search: search tips

example: site contains .netcraft.com

Results for microsoft.com

Found 292 sites

Site	Site Report	First seen	Netblock	OS
1. go.microsoft.com		november 2001	akamai technologies	linux
2. www.microsoft.com		august 1995	akamai international, bv	linux
3. support.microsoft.com		october 1997	akamai international, bv	linux
4. download.microsoft.com		august 1999	akamai international, bv	linux
5. technet.microsoft.com		august 1999	microsoft corporation	windows server 2012
6. msdn.microsoft.com		september 1998	microsoft corporation	windows server 2012
7. answers.microsoft.com		august 2009	akamai international, bv	linux
8. www.catalog.update.microsoft.com		december 2016	microsoft corporation	windows server 2016
9. windows.microsoft.com		june 1998	akamai international, bv	linux
10. social.technet.microsoft.com		august 2008	microsoft corporation	windows server 2012
11. catalog.update.microsoft.com		october 2007	microsoft corporation	windows server 2008
12. e15.officeredit.microsoft.com		may 2012	microsoft corporation	windows server 2016
13. office.microsoft.com		november 1998	microsoft corp	unknown
14. e.microsoft.com		january 2014	microsoft informatica ltda	85 big-ip
15. azure.microsoft.com		may 2014	microsoft informatica ltda	windows server 2012
16. microsoft.com		may 1996	microsoft corporation	windows server 2012
17. www.update.microsoft.com		may 2007	microsoft corporation	windows server 2012
18. update.microsoft.com		february 2005	microsoft corp	windows server 2012
19. fullproduct.download.microsoft.com		november 2007	akamai technologies	linux
20. apps.microsoft.com		may 2012	akamai international, bv	linux

<https://www.netcraft.com>

Web Server Footprinting Tools

Netcat

This utility **reads and writes data across network connections**, using the TCP/IP protocol

```
# nc -vv www.moviescope.com 80 - press [Enter]
GET / HTTP/1.0 - Press [Enter] twice
```

```
root@kali:~# nc -vv www.moviescope.com 80
DNS fwd/rev mismatch: www.moviescope.com != www.goodshopping.com
www.moviescope.com [10.10.10.16] 80 (http) open

HTTP/1.1 200 OK
Content-Type: text/html
Last-Modified: Tue, 10 Oct 2017 06:07:47 GMT
Accept-Ranges: bytes
ETag: "b53d77188e41d31:0"
Server: Microsoft-IIS/10.0
X-Powered-By: ASP.NET
Date: Wed, 27 Dec 2017 09:48:37 GMT
Connection: close
Content-Length: 703
```

Server identified as Microsoft-IIS/10.0

<http://netcat.sourceforge.net>

Telnet

This technique probes **HTTP servers** to determine the **Server field** in the HTTP response header

```
telnet www.moviescope.com 80 - press [Enter]
GET / HTTP/1.0 - Press [Enter] twice
```

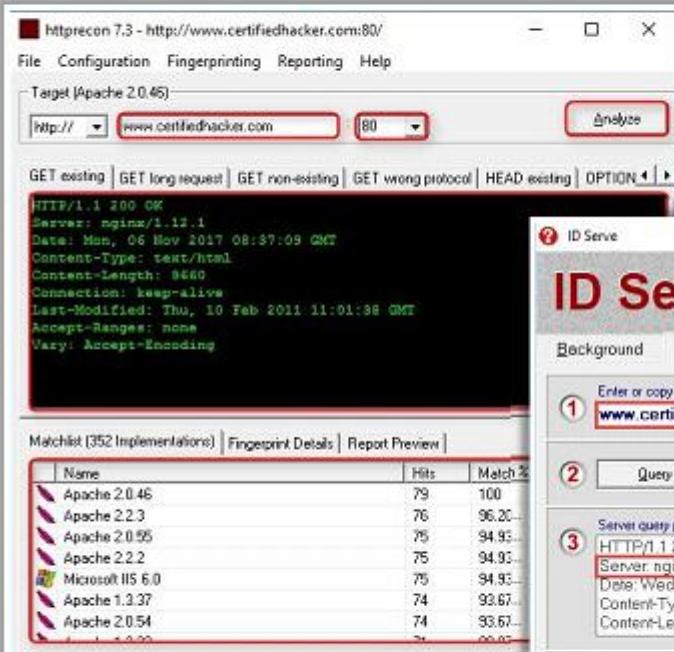
```
root@kali:~# telnet www.moviescope.com 80
Trying 10.10.10.16...
Connected to www.moviescope.com.
Escape character is '^}'.
GET / HTTP/1.0

HTTP/1.1 200 OK
Content-Type: text/html
Last-Modified: Tue, 10 Oct 2017 06:07:47 GMT
Accept-Ranges: bytes
ETag: "b53d77188e41d31:0"
Server: Microsoft-IIS/10.0
X-Powered-By: ASP.NET
Date: Wed, 27 Dec 2017 09:52:24 GMT
Connection: close
Content-Length: 703
```

Server identified as Microsoft-IIS/10.0

Web Server Footprinting Tools (Cont'd)

httprecon



Web Server Footprinting Tools

- ⑤ Recon-ng (<https://bitbucket.org>)
- ⑥ Uniscan (<https://sourceforge.net>)
- ⑦ SpiderFoot (<http://www.spiderfoot.net>)
- ⑧ htprint (<http://www.net-square.com>)
- ⑨ Nmap (<https://nmap.org>)
- ⑩ ScanLine (<https://www.mcafee.com>)
- ⑪ X probe (<https://sourceforge.net>)
- ⑫ POf (<https://github.com>)
- ⑬ Satori (<http://chatteronthewire.org>)

Enumerating Web Server Information Using Nmap

1

Attackers can use advanced **Nmap commands** and **Nmap Scripting Engine (NSE) scripts** to enumerate information about the target website

2

```
nmap -sV -O -p target IP address
```

3

```
nmap -sV --script=http-enum target IP address
```

4

```
nmap target IP address -p 80 --script = http-frontpage-login
```

5

```
nmap --script http-passwd --script-args http-passwd.root =/ target IP address
```

```

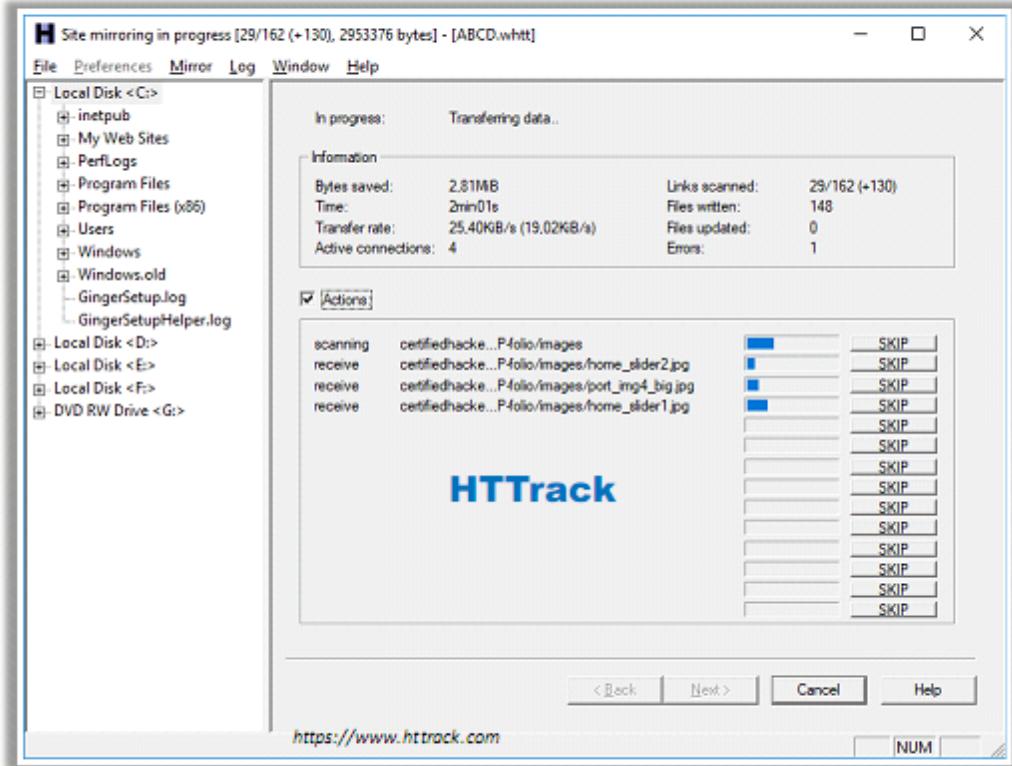
Starting Nmap 7.60 ( https://nmap.org ) at 2017-11-03 18:12
Standard Time
Nmap scan report for certifiedhacker.com (69.89.31.193)
Host is up (0.16s latency).
rDNS record for 69.89.31.193: box393.bluehost.com
Not shown: 927 filtered ports, 58 closed ports
PORT      STATE SERVICE          VERSION
21/tcp    open  ftp   Pure-FTPD
22/tcp    open  ssh   OpenSSH 5.3 (protocol 2.0)
25/tcp    open  smtp  Exim smtpd 4.87
26/tcp    open  smtp  Exim smtpd 4.87
80/tcp    open  http  nginx 1.12.1
| http-enum:
|_ /css/: Potentially interesting directory w/ listing on 'apache'
|_/images/: Potentially interesting directory w/ listing on 'apache'
|_/js/: Potentially interesting directory w/ listing on 'apache'
|_/xml/: Potentially interesting directory w/ listing on 'apache'
110/tcp   open  pop3
| fingerprint-strings:
|_ GenericLines:
|_ >OK POP3 ready <1978300442.1509713084@box393.bluehost.com>
|_ >ERR invalid command
|_ >ERR invalid command
|_ HTTPOptions:
|_ >OK POP3 ready <244472285.1509713095@box393.bluehost.com>
|_ >ERR invalid command
|_ >ERR invalid command
|_ NULL:
|_ >OK POP3 ready <1978300442.1509713084@box393.bluehost.com>
143/tcp   open  imap-proxy   nginx imap proxy
443/tcp   open  ssl/http   Apache httpd
| http-enum:
|_ /robots.txt: Robots file
|_ http-server-header: nginx/1.12.1
465/tcp   open  ssl/smtp   Exim smtpd 4.87
587/tcp   open  ssl/smtp   Exim smtpd 4.87
993/tcp   open  ssl/pop3
| fingerprint-strings:
|_ GenericLines:

```

<https://nmap.org>

Website Mirroring

- Mirror a website to create a complete profile of the site's **directory structure, files structure, external links**, etc.
- Search for comments and other items in the **HTML source code** to make footprinting activities more efficient
- Use tools such as **HTTrack, WebCopier Pro**, etc. to mirror a website



Finding Default Credentials of Web Server

- Many web server administrative interfaces are **publically accessible** and are located in the **web root** directory
- Often these administrative interface credentials are **not properly configured** and remain **set to default**
- Attackers attempt to **identify the running application interface** and performs following techniques to identify the default login credentials:
 - Consult the **administrative interface documentation** and identify the default passwords
 - Use **Metasploit's built-in database** to scan the server
 - Use online resources like **Open Sez Me** (<http://open-sez.me>), **cirt.net** (<https://cirt.net/passwords>), etc.
 - Attempt **password guessing** and **brute-forcing** attacks

The screenshot shows the CIRT.net website with a search bar for default passwords. It displays a list of 523 vendors and 2084 passwords. The table lists various companies such as 2Wire, Inc., 360 Systems, 3COM, 3M, Accelerated Networks, ACCTON, Acer, Adcomtec, Adaptec, ADC Kontrox, AdComplete.com, AddPac Technology, Adobe, ADT, Adtech, Adtran, Advanced Integration, AIRAYA Corp, AirLink, AirLink Plus, Alkonef, Airway, Aladdin, Alcatel, Alien Technology, Allied Telesyn, Alinet, and Alstom. At the bottom, a link to the full URL is provided: <https://cirt.net/passwords>.

2Wire, Inc.	360 Systems	3COM
3M	Accelerated Networks	ACCTON
Acer	Adcomtec	Adaptec
ADC Kontrox	AdComplete.com	AddPac Technology
Adobe	ADT	Adtech
Adtran	Advanced Integration	AIRAYA Corp
AirLink	AirLink Plus	Alkonef
Airway	Aladdin	Alcatel
Alien Technology	Allied Telesyn	Alinet

Finding Default Content of Web Server

- Most of the web application servers contain **default content and functionalities** allowing attackers to leverage attacks

- Check for the following default contents and functionalities in the web servers
 - Administrators **debug and test functionality**
 - **Sample functionality** to demonstrate common tasks
 - Publically accessible **powerful functions**
 - Server **Installation manuals**

- Use tools like **Nikto2** (<https://cirt.net>) and exploit databases like **SecurityFocus** (<http://www.securityfocus.com>) to identify the default content

```
root@kali: ~
File Edit View Search Terminal Help
root@kali:~# nikto -h http://www.certifiedhacker.com -Tuning x
- Nikto v2.1.6
-----
+ Target IP:          162.241.216.11
+ Target Hostname:   www.certifiedhacker.com
+ Target Port:        80
+ Start Time:        2017-12-28 07:53:16 (GMT-5)
-----
+ Server: nginx/1.12.2
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XSS
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type
+ Allowed HTTP Methods: POST, OPTIONS, HEAD, GET
+ OSVDB-3233: /mailman/listinfo: Mailman was found on the server.
+ OSVDB-3092: /cgi-sys/: This might be interesting... possibly a system shell found.
+ OSVDB-3092: /img-sys/: Default image directory should not allow directory listing.
+ OSVDB-3092: /java-sys/: Default Java directory should not allow directory listing.
+ OSVDB-3268: /images/: Directory indexing found.
+ OSVDB-3268: /docs/: Directory indexing found.
+ 9953 requests: 1 error(s) and 10 item(s) reported on remote host
+ End Time:           2017-12-28 08:39:51 (GMT-5) (2795 seconds)
-----
+ 1 host(s) tested
root@kali:~#
```

<https://cirt.net>

Finding Directory Listings of Web Server

- When a web server receives a request for the directory, it responds to the request in the following ways
 - Return **default resource within directory**
 - Return **error**
 - Return **listing of directory content**

- Directory listings sometimes possess the following **vulnerabilities** that allow the attackers to **compromise web server**
 - Improper **access controls**
 - Unintentional **access to web root** of servers

- After discovering the directory on the web server, **make a request for the same directory** and try to **access the directory listings**

- Try to **exploit vulnerable web server software** that gives **access to the directory listings**

Index of /images

Name	Last modified	Size	Description
Parent Directory	09-Jan-2008 07:35	-	
Thumbs.db	04-Nov-2006 07:07	166k	
notes/	04-Nov-2006 06:50	-	
arrow.jpg	04-Nov-2006 07:06	1k	
banner.jpg	04-Nov-2006 07:08	73k	
banner2 01.jpg	04-Nov-2006 07:06	7k	
banner2 02.jpg	04-Nov-2006 07:06	73k	
banner2 03.jpg	04-Nov-2006 07:08	7k	
banner2 04.jpg	04-Nov-2006 07:08	1k	
banner2 05.jpg	04-Nov-2006 07:08	1k	
banner2 06.jpg	04-Nov-2006 07:08	3k	
banner2 07.jpg	04-Nov-2006 07:09	4k	
banner2 08.jpg	04-Nov-2006 07:09	1k	
banner2 09.jpg	04-Nov-2006 07:09	3k	
banner2 10.jpg	04-Nov-2006 07:09	1k	

Vulnerability Scanning

- Implement vulnerability scan to identify weaknesses in a network and determine if the system can be exploited
- Use vulnerability scanners such as WebInspect, Acunetix Web Vulnerability Scanner, etc. to find hosts, services, and vulnerabilities
- Sniff the network traffic to find out active systems, network services, applications, and vulnerabilities present
- Test the web server infrastructure for any misconfigurations, outdated content, and vulnerabilities using vulnerability scanners like Acunetix Web Vulnerability Scanner

The screenshot shows the Acunetix Web Vulnerability Scanner interface. On the left, there's a sidebar with options: Dashboard, Targets, Vulnerabilities (which is selected), Scans, Reports, and Settings. The main area has tabs for Scan Stats & Info, Vulnerabilities, Site Structure (which is highlighted with a red border), and Events. Below these tabs, it shows the URL <http://www.moviescope.com/>. A folder icon indicates subfolders: css, db, images, and js. To the right, there's a table of vulnerabilities:

Se...	Vulnerability	URL
1	Blind SQL Injection	http://www.moviescope.com
1	Blind SQL Injection	http://www.moviescope.com
1	Microsoft IIS tilde directory enumeration	http://www.moviescope.com
1	Unencrypted __VIEWSTATE parameter	http://www.moviescope.com
1	Vulnerable Javascript library	http://www.moviescope.com
1	ASP.NET debugging enabled	http://www.moviescope.com

The URL <https://www.acunetix.com> is visible at the bottom right.

Finding Exploitable Vulnerabilities

- Search for a web server exploitable vulnerabilities based on the web server OS and software application on exploit sites such as **SecurityFocus** (<http://www.securityfocus.com>), **Exploit Database** (<https://www.exploit-db.com>), etc.
- Search for vulnerability based on information gathered in the previous stages by using **More Options** button
- Exploiting these vulnerabilities allows one to execute a command or binary on a target machine to **gain higher privileges** than the existing ones or **bypass security mechanisms**

EXPLOIT DATABASE

Home Exploits Shellcode Papers Google Hacking Database Submit Search

Search the Exploit Database

Search the Database for Exploits, Papers, and Shellcode. You can even search by CVE and OSVDB identifiers.

Web server vulnerabilities I'm not a robot reCAPTCHA Privacy-Terms **Search**

Exploit/Shellcode/Paper Content Author Any Platform

Any Type Port OSVDB

Verified Exploits Only Exploits with Applications Only Include DoS Exploits Include Metasploit Framework Exploits Include Papers

Date	D	A	V	Title	Platform	Author
2017-09-13	-	-	-	Mako Web Server 2.5 - Multiple Vulnerabilities	Windows	hyp3rlinx
2016-08-29	-	-	-	Goron WebServer 2.0 - Multiple Vulnerabilities	Windows	Guillaume K..
2016-08-10	-	-	-	WebNMS Framework Server 5.2/5.2 SP1 - Multiple Vulnerabilities	JSP	Pedro Ribeiro
2015-09-20	-	-	-	ADH-Web Server IP-Cameras - Multiple Vulnerabilities	Hardware	Orwellabs
2014-02-19	-	-	-	Embedthis GoAhead WebServer 3.1.3-0 - Multiple Vulnerabilities	Linux	Makymillian...
2012-10-17	-	-	-	Oracle WebCenter Sites (FatWire Content Server) - Multiple Vulnerabilities	Multiple	SEC Consult
2012-02-02	-	-	-	Sphinx Mobile Web Server 3.1.2.47 - Persistent Cross-Site Scripting Multiple...	Windows	SecPod Rese...
2011-11-18	-	-	-	GoAhead Web Server 2.5 - 'goform/formTest' Multiple Cross-Site Scripting Vulnerabilities	Windows	Prabhu S An...
2011-10-10	-	-	-	GoAhead Web Server 2.18 - 'adduser.asp' Multiple Cross-Site Scripting Vulnerabilities	Windows	Silent Dream
2011-10-10	-	-	-	advrise webM12ADS Web Server 1.0 - Multiple Vulnerabilities	Windows	Luiji Auriemma
2011-03-29	-	-	-	Easy File Sharing Web Server 5.8 - Multiple Vulnerabilities	Windows	AutoSec Tools
2010-04-08	-	-	-	Tiny Java Web Server 1.71 - Multiple Input Validation Vulnerabilities	Multiple	cp77fk4r
2010-04-08	-	-	-	miniature java Web server 1.71 - Multiple Vulnerabilities	Multiple	cp77fk4r
2010-02-24	-	-	-	Web Server Creator Web Portal 0.1 - Multiple Vulnerabilities	PHP	indoushka

<https://www.exploit-db.com>

Session Hijacking

- Sniff valid session IDs to gain unauthorized access to the web server and snoop into data

- Use session hijacking techniques such as session fixation, session sidejacking, Cross-site scripting, etc., to capture valid session cookies and IDs

- Use tools such as Burp Suite, Firesheep, JHijack, etc. to automate session hijacking

Burp Suite Community Edition v1.7.30 - Temporary Project

Burp Intruder Repeater Window Help

Target Proxy Spider Scanner Intruder Repeater Sequencer Decoder Comparer Extender Project options User options Alerts

Site map Scope

Filter: Hiding not found items; hiding CSS, image and general binary content; hiding 4xx responses; hiding empty folders

Host	Method	URL	Params	Status	Length	MIME type	Title
http://i.s-microsoft.com	GET	/fonts/icons/Homepag...		200	13585	XML	
http://i.s-microsoft.com	GET	/fonts/Segoe-UINWest...		200	55248	XML	
http://i.s-microsoft.com	GET	/fonts/Segoe-UINWest...		200	56126	XML	
http://i.s-microsoft.com	GET	/home/bimapping.js		200	442		
http://i.s-microsoft.com	GET	#home/bimapping.js?...		200	2807	script	
http://i.s-microsoft.com	GET	/home/script.jsx		200	445		
http://i.s-microsoft.com	GET	/home/script.jsx?ke~...		200	50544	script	
http://i.s-microsoft.com	GET	/home/script.jsx?ke~...		200	3471	script	
http://i.s-microsoft.com	GET	/home/script.jsx?ke~...		200	20959	script	

Request Response

Raw Params Headers Hex

```

GET
/home/bimapping.js?gv=BiMapping&k=en-us/home/Components/config/BiMapping.xml&v=-178947
7394 HTTP/1.1
Host: i.s-microsoft.com
Proxy-Connection: keep-alive
Accept: /*
User-Agent: Mozilla/5.0 (Windows NT 6.3; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/30.0.1599.101 Safari/537.36
Referer: http://www.microsoft.com/en-in/default.aspx
Accept-Encoding: gzip,deflate,sdch
Accept-Language: en-US,en;q=0.8

```

Type a search term 0 matches

<https://pentawiggen.net>

Note: For complete coverage of Session Hijacking concepts and techniques refer to Module 11: Session Hijacking

Web Server Passwords Hacking

- Use password cracking techniques such as **brute force attack, dictionary attack, password guessing** to crack web server passwords
- Use tools such as **Hashcat**, THC-Hydra, Ncrack, etc.



```
$ ./hashcat -m 15600 -a 3 hash.txt ?a?a?a?acat
hashcat (v3.6.0) starting...

openCL Platform #1: NVIDIA Corporation
* Device #1: GeForce GTX 1080, 2028/8107 MB allocatable, 20MCU
* Device #2: GeForce GTX 1080, 2028/8114 MB allocatable, 20MCU
* Device #3: GeForce GTX 1080, 2028/8114 MB allocatable, 20MCU
* Device #4: GeForce GTX 1080, 2028/8114 MB allocatable, 20MCU

Hashes: 1 digests; 1 unique digests, 1 unique salts
Bitmaps: 16 bits, 65536 entries, 0x0000ffff mask, 262144 bytes, 5/13 rotates

Applicable optimizers:
* Zero-Byte
* Single-Hash
* Single-Salt
* Brute-Force
* Slow-Hash-SIMD

Watchdog: Temperature abort trigger set to 90c
Watchdog: Temperature retain trigger set to 75c

$ethereum$p*262144*32383831373131303534383437373837...da3946:hashcat

Session.....: hashcat
Status.....: Cracked
Hash.Type...: Ethereum Wallet, PBKDF2-HMAC-SHA256
Hash.Target.: $ethereum$p*262144*32383831373131303534383437373837...da3946
Time.Started.: Wed Jun 7 14:47:52 2017 (11 mins, 41 secs)
Time.Estimated.: Wed Jun 7 14:59:33 2017 (0 secs)
Guess.Mask...: ?a?a?a?acat [7]
Guess.Queue...: 1/1 (100.00%)
Speed.Dev.#1.: 5192 H/s (42.69ms)
Speed.Dev.#2.: 5202 H/s (42.73ms)
Speed.Dev.#3.: 5212 H/s (42.32ms)
Speed.Dev.#4.: 5213 H/s (42.30ms)
Speed.Dev.#*: 20819 H/s
Recovered....: 1/1 (100.00%) digests, 1/1 (100.00%) salts
Progress.....: 14146688/81450625 (17.37%)
Rejected....: 0/14146688 (0.00%)
Restore.Point.: 0/857375 (0.00%)
Candidates.#1.: haricat => hp/ycat
Candidates.#2.: h/rycat => h)-jcat
Candidates.#3.: myQ8cat => nhFEcat
Candidates.#4.: nBjlcat => n uVcat
HMon.Dev.#1.: Temp: 74c Fan: 69% Util:100% Core:1974MHz Mem:4513MHz Bus:1
HMon.Dev.#2.: Temp: 74c Fan: 80% Util:100% core:1974MHz Mem:4513MHz Bus:1
HMon.Dev.#3.: Temp: 75c Fan: 64% Util:100% core:1974MHz Mem:4513MHz Bus:1
HMon.Dev.#4.: Temp: 75c Fan: 79% Util:100% core:1974MHz Mem:4513MHz Bus:1

Started: wed Jun 7 14:47:43 2017
Stopped: wed Jun 7 14:59:34 2017
```

<https://hashcat.net>

Using Application Server as a Proxy

- Web servers with **forwarding** and **reverse HTTP proxy functions** enabled, are employed by the attackers to perform the following attacks:
 - Attacking third party systems on internet
 - Connecting to arbitrary hosts on the organization's internal network
 - Connecting back to other services running on the proxy host itself
- Attackers use **GET** and **CONNECT** requests **to use vulnerable web servers as proxies** to connect and obtain information from target systems through these proxy web servers



Module Flow

1

Web Server Concepts

2

Web Server Attacks

3

Web Server Attack Methodology

4

Web Server Attack Tools

5

Counter-measures

6

Patch Management

7

Web Server Security Tools

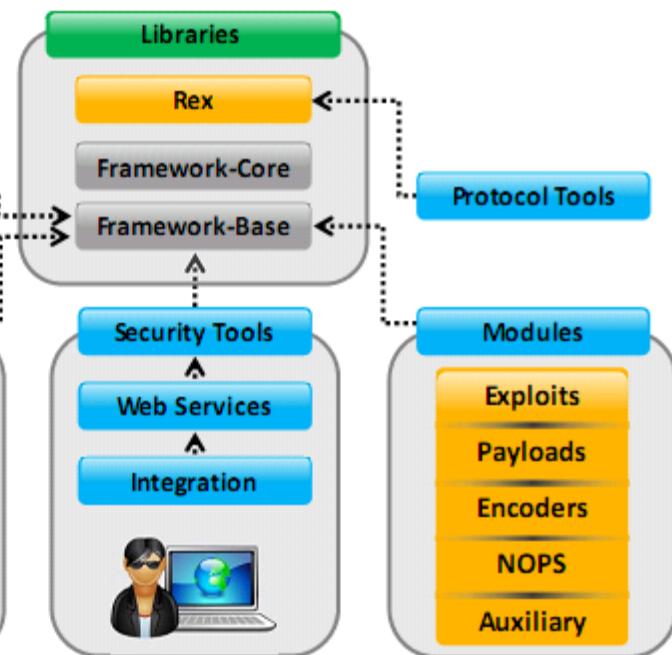
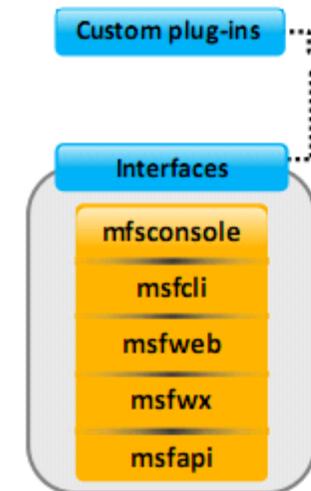
8

Web Server Pen Testing

Metasploit

- The Metasploit Framework is a exploit development platform which supports fully automated **exploitation of web servers**, by abusing known vulnerabilities and leveraging weak passwords via Telnet, SSH, HTTP, and SNM

Metasploit Architecture



IP Address	Hostname	Operating System	VM	Purpose	Size	Virus	Act.	Notes	Updated	Status
192.168.168.300	Windows-PC	Microsoft Windows (TM) Professional SP2		server	19		4		9 minutes ago	Scanned
192.168.168.302	BBB Unknown	BBB Unknown		device	1				8 minutes ago	Scanned
192.168.168.303	BBB Unknown	BBB Unknown		device	1				8 minutes ago	Scanned
192.168.168.311	www.Johns-PC	Microsoft Windows (TM) Professional 7601 Service Pack 3		client	18		4		8 minutes ago	Scanned
192.168.168.310	BBB Unknown	BBB Unknown		device	1				8 minutes ago	Scanned
192.168.168.311	Johns-PC	Microsoft Windows (TM) Professional 7601 Service Pack 3		client	19		4		8 minutes ago	Scanned
192.168.168.313	ADMIN-PC	Microsoft Windows (TM) Professional 7601 Service Pack 3		client	9		4		8 minutes ago	Scanned
192.168.168.315	www.Johns-PC	Microsoft Windows (TM) Professional 7601 Service Pack 3		device	6		3		8 minutes ago	Scanned
192.168.168.320	BBB Unknown	Microsoft Windows (TM) Professional 7601 Service Pack 3		client	9		4		8 minutes ago	Scanned
192.168.168.333	BBB Unknown	Microsoft Windows (TM) Professional 7601 Service Pack 3		client	11		4		8 minutes ago	Scanned
192.168.168.333	ADMIN-PC	Microsoft Windows (TM) Professional 7601 Service Pack 3		client	6		4		8 minutes ago	Scanned
192.168.168.34	6014	Microsoft Windows (TM) SP2		client	8		2		7 minutes ago	Scanned

<https://www.metasploit.com>

Metasploit Exploit Module

- It is the basic module in Metasploit used to **encapsulate an exploit** with the help of which users target many platforms with a single exploit
- This module comes with **simplified meta-information fields**
- Using a Mixins feature, users can also **modify exploit behavior dynamically**, brute force attacks, and attempt passive exploits



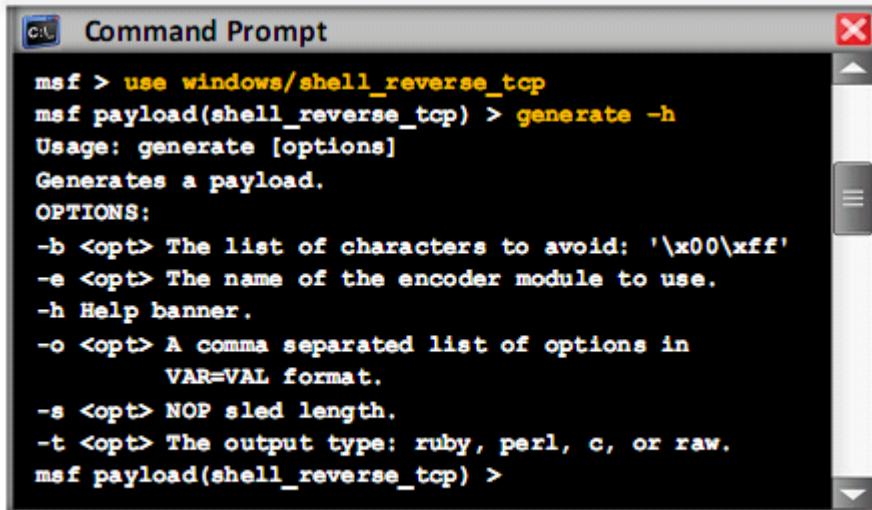
Steps to exploit a system follow the Metasploit Framework

- 1 Configuring Active Exploit
- 2 Verifying the Exploit Options
- 3 Selecting a Target
- 4 Selecting the Payload
- 5 Launching the Exploit

Metasploit Payload and Auxiliary Module

Payload Module

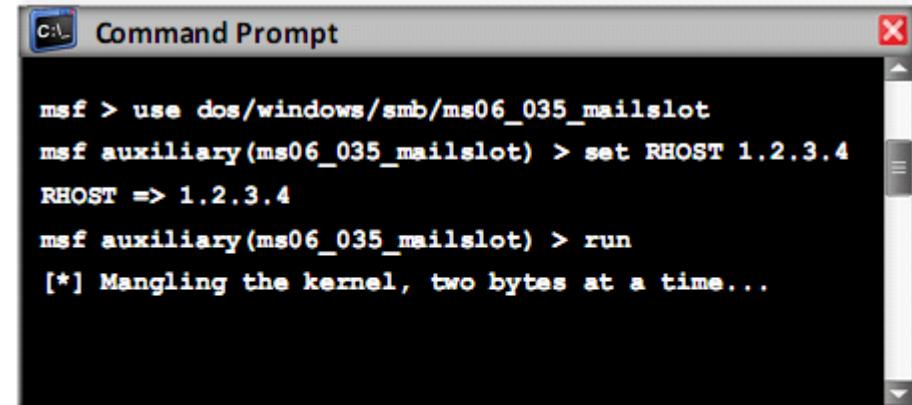
- Payload module establishes a **communication channel** between the Metasploit framework and the victim host
- It combines the **arbitrary code** that is executed as a result of an exploit succeeding
- To generate **payloads**, first select a payload using the command as shown in the screenshot



```
c:\msf> use windows/shell_reverse_tcp
msf payload(shell_reverse_tcp) > generate -h
Usage: generate [options]
Generates a payload.
OPTIONS:
-b <opt> The list of characters to avoid: '\x00\xff'
-e <opt> The name of the encoder module to use.
-h Help banner.
-o <opt> A comma separated list of options in
        VAR=VAL format.
-s <opt> NOP sled length.
-t <opt> The output type: ruby, perl, c, or raw.
msf payload(shell_reverse_tcp) >
```

Auxiliary Module

- Auxiliary modules can be **used to perform arbitrary**, one-off actions such as port scanning, denial of service, and even fuzzing
- To run the auxiliary module, either use the **run** command, or use the **exploit** command



```
c:\msf> use dos/windows/smb/ms06_035_mailslot
msf auxiliary(ms06_035_mailslot) > set RHOST 1.2.3.4
RHOST => 1.2.3.4
msf auxiliary(ms06_035_mailslot) > run
[*] Mangling the kernel, two bytes at a time...
```

Metasploit NOPS Module

- NOP modules generate a no-operation instruction used for blocking out buffers
- Use **generate** command to generate a NOP sled of an arbitrary size and display it in a given format

OPTIONS:

- b <opt>: The list of characters to avoid: '\x00\xff'
- h: Help banner
- s <opt>: The comma separated list of registers to save
- t <opt>: The output type: ruby, perl, c, or raw

```
msf nop(opty2) >
```

**Generates a NOP sled of a given length**

```
msf > use x86/opty2
msf nop(opty2) > generate -h
Usage: generate [options] length
```

**Command to generate a 50 byte NOP sled**

```
msf nop(opty2) > generate -t c 50
unsigned char buf[] =
"\xf5\x3d\x05\x15\xf8\x67\xba\x7d\x08\xd6\x66\x9f\xb8\x2d\xb6"
"\x24\xbe\xb1\x3f\x43\x1d\x93\xb2\x37\x35\x84\xd5\x14\x40\xb4"
"\xb3\x41\xb9\x48\x04\x99\x46\xa9\xb0\xb7\x2f\xfd\x96\x4a\x98"
"\x92\xb5\xd4\x4f\x91";
msf nop(opty2) >
```

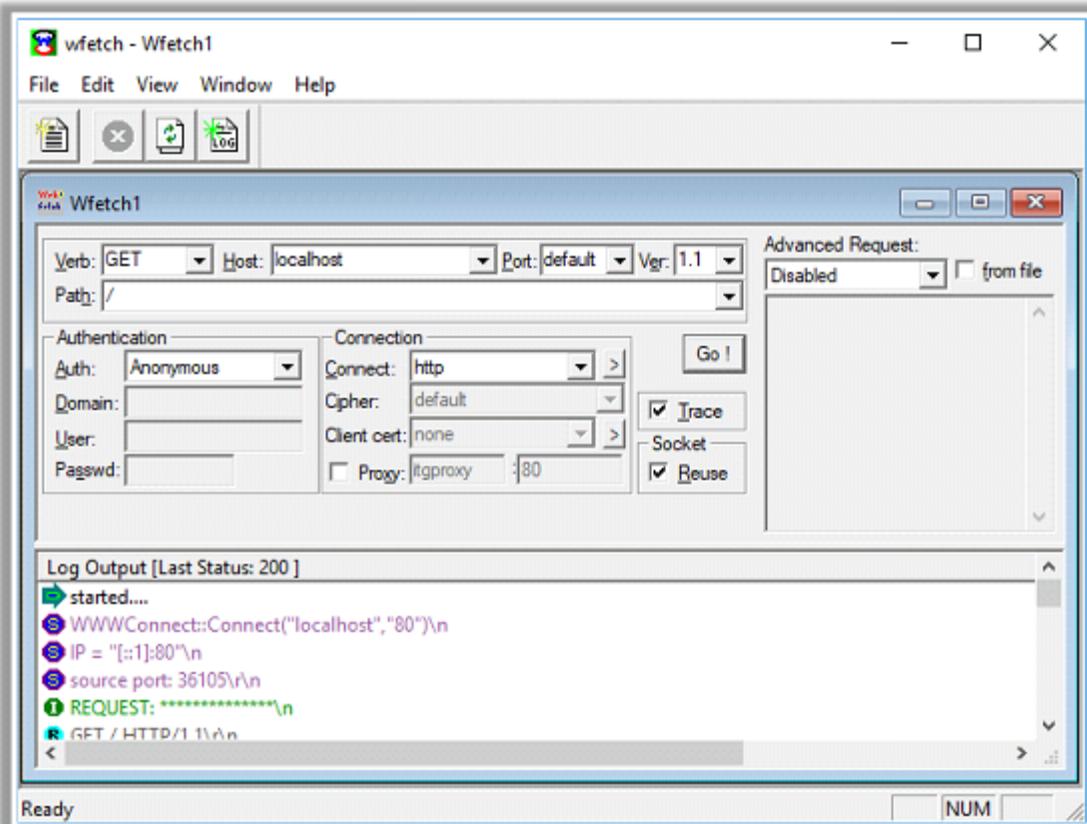
Web Server Attack Tools

Wfetch

- **Wfetch** allows the attacker to fully customize an **HTTP request** and send it to a Web server to see the raw HTTP request and response data
- It allows the attacker to test the performance of Web sites that contain new elements such as **Active Server Pages (ASP)** or wireless protocols

Web Server Attack Tools

- THC Hydra (<https://www.thc.org>)
- HULK DoS (<https://github.com>)
- MPack (<https://sourceforge.net>)
- w3af (<http://w3af.org>)



Module Flow

1

Web Server Concepts

2

Web Server Attacks

3

Web Server Attack Methodology

4

Web Server Attack Tools

5

Counter-measures

6

Patch Management

7

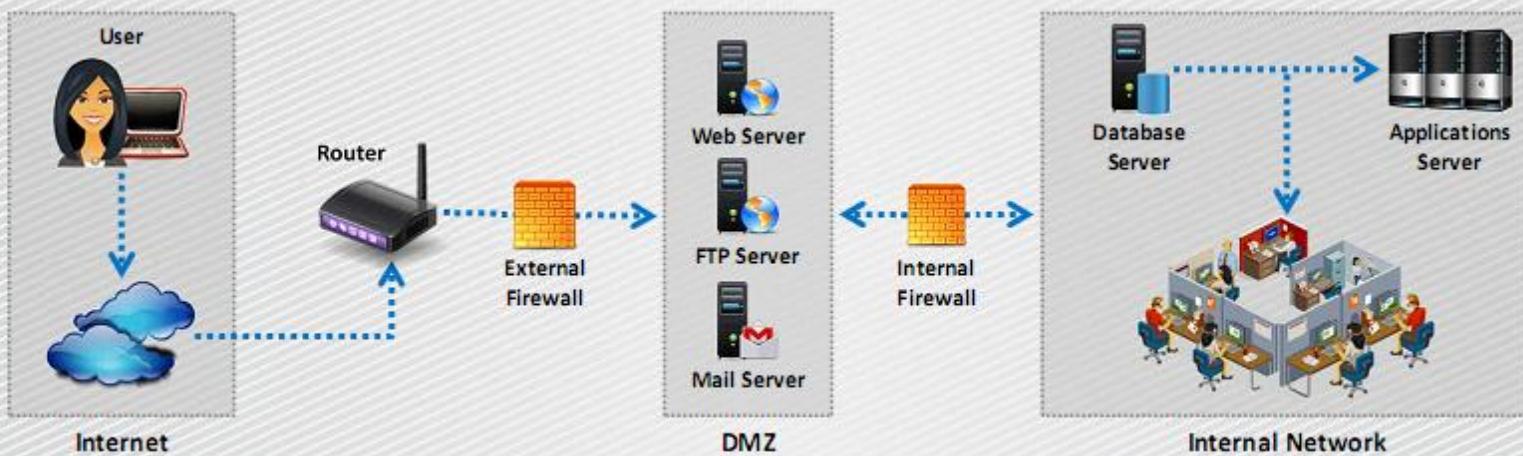
Web Server Security Tools

8

Web Server Pen Testing

Place Web Servers in Separate Secure Server Security Segment on Network

- An ideal **web hosting network** should be designed with at least **three segments** namely Internet segment, secure server security segment often called demilitarized zone (DMZ), and internal network
- Place the web server in **Server Security Segment** (DMZ) of the network, isolated from public network as well as internal network
- Firewalls should be in place for **internal network** as well as **Internet traffic** going towards DMZ



Countermeasures: Patches and Updates

01

Scan for existing vulnerabilities, patch, and update the **server software regularly**

05

Ensure that service packs, hotfixes, and security patch levels are consistent on **all Domain Controllers (DCs)**

02

Before applying any service pack, hotfix, or security patch, **read and peer review** all relevant documentation

06

Ensure that **server outages** are scheduled and a complete set of **backup tapes** and emergency repair disks are available

03

Apply all updates, regardless of their type on an "**as-needed**" basis

07

Have a **back-out plan** that allows the system and enterprise to return to their original state, prior to the failed implementation

04

Test the service packs and hotfixes on a representative **non-production environment** prior to being deployed to production

08

Schedule periodic service pack upgrades as part of operations maintenance and never try to have **more than two service packs behind**

Countermeasures: Protocols

01 Block all unnecessary ports, Internet Control Message Protocol (ICMP) traffic, and unnecessary protocols such as NetBIOS and SMB



02 Harden the TCP/IP stack and consistently apply the latest software patches and updates to system software



03 If using insecure protocols such as Telnet, POP3, SMTP, FTP, take appropriate measures to provide secure authentication and communication, for example, by using IPSec policies



04 If remote access is needed, make sure that the remote connection is secured properly, by using tunneling and encryption protocols



05 Disable WebDAV if not used by the application or keep secure if it is required



Countermeasures: Accounts

1

Remove all unused modules and application extensions

2

Disable unused default user accounts created during installation of an operating system

3

When creating a new web root directory, grant the appropriate (least possible) NTFS permissions to the anonymous user being used from the IIS web server to access the web content

4

Eliminate unnecessary database users and stored procedures and follow the principle of least privilege for the database application to defend against SQL query poisoning

5

Use secure web permissions, NTFS permissions, and .NET Framework access control mechanisms including URL authorization

6

Slow down brute force and dictionary attacks with strong password policies, and then audit and be alert for logon failures

7

Run processes using least privileged accounts as well as least privileged service and user accounts

Countermeasures: Files and Directories

Eliminate unnecessary files within the **.jar files**



Disable serving of **directory listings**

Eliminate **sensitive configuration** information within the **byte code**



Eliminate the **presence of non-web files** such as archive files, backup files, text files, and header/include files

Avoid mapping **virtual directories** between two different servers, or over a network



Disable serving certain **file types** by creating a resource mapping

Monitor and check all **network services logs**, **website access logs**, **database server logs** (e.g., Microsoft SQL Server, MySQL, Oracle) and OS logs frequently



Ensure the presence of **web application or website files** and **scripts** on a separate partition or drive other than that of the operating system, logs, and any other system files

Detecting Web Server Hacking Attempts



Use **Website Change Detection System** to detect hacking attempts on the web server

Website Change Detection System involves:



Running specific script on the server that detects any changes made in the existing executable file or new file included on the server



Periodically comparing the **hash values** of the files on the server with their respective master hash value to detect the changes made in codebase



Alerting the user upon any change detection on the server



For example: **WebsiteCDS** is a script that goes through your entire web folder and detects any changes made to your code base and alerts you using email

How to Defend Against Web Server Attacks

01

Ports

- Audit the ports on the server regularly to ensure that an **insecure** or unnecessary service is not active on your web server
- Limit inbound traffic to **port 80 for HTTP** and **port 443 for HTTPS (SSL)**
- Encrypt or restrict **intranet traffic**

02

Server Certificates

- Ensure that **certificate data ranges** are valid and that certificates are used for their intended purpose
- Ensure that any certificate has not been revoked and **certificate's public key** is valid all the way to a trusted root authority

03

Machine.config

- Ensure that protected resources are mapped to **HttpForbiddenHandler** and unused **HttpModules** are removed
- Ensure that **tracing is disabled** <trace enable="false"/> and **debug compiles** are turned off

04

Code Access Security

- Implement **secure coding** practices
- Restrict **code access security policy** settings
- Configure IIS** to reject URLs with "../" and install new patches and updates

How to Defend Against Web Server Attacks

(Cont'd)

1

- UrlScan is a security tool that **restricts** the types of HTTP requests that IIS will process

2

- By blocking specific HTTP requests, the UrlScan security tool helps to **prevent potentially harmful requests** from reaching applications on the server

3

- UrlScan screens all incoming requests to the server by filtering the requests based on **rules** that are set by the administrator

4

- UrlScan can be configured to filter HTTP query string values and other HTTP headers to **mitigate SQL injection** attacks while the root cause is being fixed in the application

5

- It provides **W3C formatted logs** for easier log file analysis through log parsing solutions like Microsoft Log Parser 2.2

How to Defend Against Web Server Attacks

(Cont'd)

01

- Apply **restricted ACLs** and block remote registry administration
- Secure the **SAM** (Stand-alone Servers Only)



02

Ensure that security related settings are **configured appropriately** and access to the metabase file is restricted with hardened **NTFS permissions**



03

Remove unnecessary ISAPI filters from the web server



04

- Remove all unnecessary file shares including the **default administration shares** if not required
- Secure the shares with restricted **NTFS permissions**



05

Relocate sites and virtual directories to **non-system partitions** and use IIS Web permissions to restrict access



06

Remove all unnecessary **IIS script mappings** for optional file extensions to avoid exploiting any bugs in the **ISAPI** extensions that handle these types of files



07

Enable a **minimum level of auditing** on your web server and use NTFS permissions to protect the **log files**



How to Defend Against Web Server Attacks (Cont'd)

Do use a **dedicated machine** as a web server

Do physically protect the **web server machine** in a secure machine room

Create **URL mappings** to internal servers cautiously

Do not connect an IIS Server to the **Internet** until it is fully hardened

Do not install the **IIS server** on a domain controller

Do not allow anyone to **locally log on** to the machine except for the administrator

Use server side **session ID tracking** and match connections with time stamps, IP addresses, etc.

Do configure a **separate anonymous user account** for each application, if you host multiple web applications

If a database server, such as **Microsoft SQL Server**, is to be used as a backend database, install it on a **separate server**

Limit the **server functionality** in order to support the web technologies that are going to be used

Use **security tools** provided with web server software and **scanners** that automate and make the process of securing a web server easy

Screen and filter the **incoming traffic request**

How to Defend against HTTP Response Splitting and Web Cache Poisoning



Server Admin

- Use latest **web server software**
- Regularly **update/patch OS** and web server
- Run **web Vulnerability Scanner**



Application Developers

- Restrict web application access to **unique IPs**
- Disallow **carriage return** (%0d or \r) and line feed (%0a or \n) characters
- Comply to **RFC 2616** specifications for HTTP/1.1



Proxy Servers

- Avoid sharing **incoming TCP connections** among different clients
- Use different TCP connections with the proxy for different **virtual hosts**
- Implement "**maintain request host header**" correctly

How to Defend against DNS Hijacking



Choose an ICANN accredited registrar and encourage them to set Registrar-Lock on the domain name



Safeguard the registrant account information



Include DNS hijacking into incident response and business continuity planning



Use DNS monitoring tools/services to monitor DNS server IP address and alert



Avoid downloading audio and video codecs and other downloaders from untrusted websites



Install antivirus program and update it regularly



Change the default router password that comes with the factory settings

Module Flow

1

Web Server Concepts

2

Web Server Attacks

3

Web Server Attack Methodology

4

Web Server Attack Tools

5

Counter-measures

6

Patch Management

7

Web Server Security Tools

8

Web Server Pen Testing

Patches and Hotfixes

Hotfixes are an **update to fix a specific customer issue** and not always distributed outside the customer organization

A patch is a **small piece of software designed to fix problems**, security vulnerabilities, and bugs and improve the performance of a computer program or its supporting data

Users may be notified through **emails** or through the **vendor's website**

A patch can be considered as a **repair job to a programming problem**

Hotfixes are sometimes packaged as a set of fixes called a **combined hotfix** or **service pack**

What is Patch Management?

"Patch management is a process used to ensure that the **appropriate patches** are installed on a system and help fix known vulnerabilities"

An automated patch management process

Detect

Use tools to detect missing security patches

Assess

Asses the issue(s) and its associated severity by mitigating the factors that may influence the decision

Acquire

Download the patch for testing

Test

Install the patch first on a testing machine to verify the consequences of the update

Deploy

Deploy the patch to the computers and make sure the applications are not affected

Maintain

Subscribe to get notifications about vulnerabilities as they are reported

Installation of a Patch

Identifying Appropriate Sources for Updates and Patches

- First make a **patch management plan** that fits the operational environment and business objectives
- Find appropriate **updates and patches** on the home sites of the applications or operating systems' vendors
- The recommended way of tracking issues relevant to **proactive patching** is to register with the home sites to **receive alerts**

Installation of a Patch

- Users can access and install security patches via the **World Wide Web**
- Patches can be installed in two ways

Manual Installation

- In this method, the user has to **download the patch** from the vendor and fix it

Automatic Installation

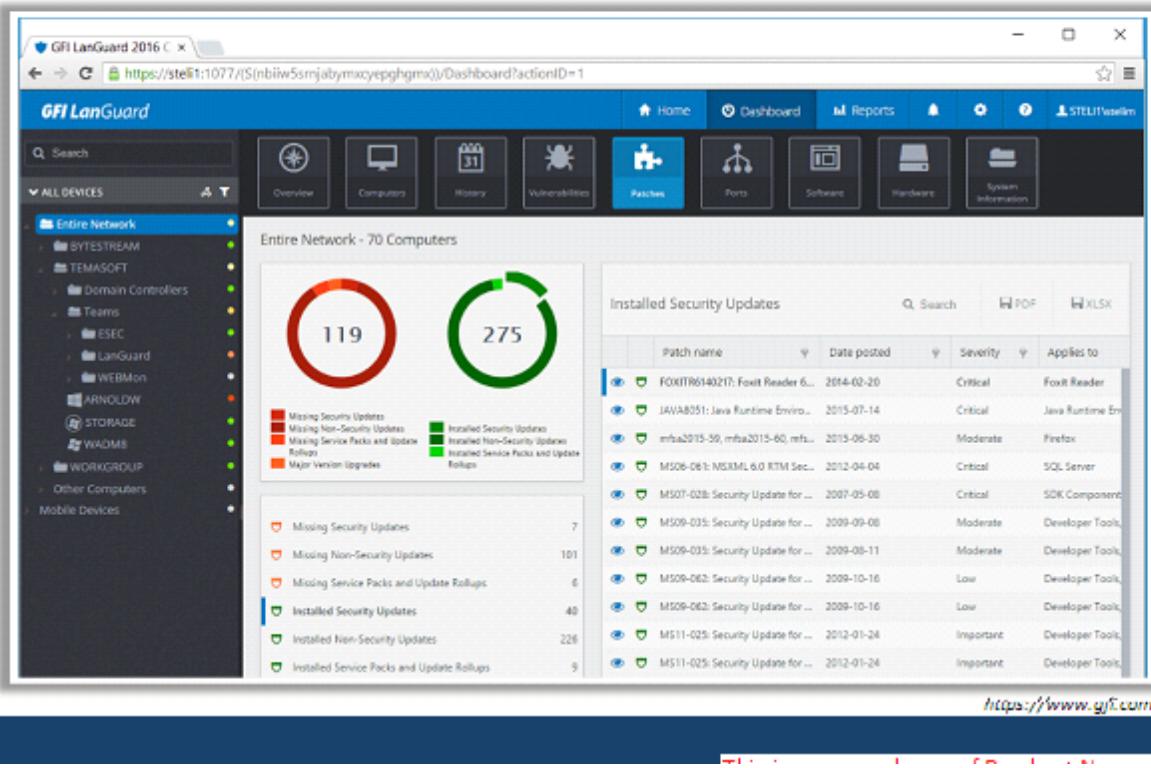
- In this method, the applications use the **Auto Update** feature to update themselves

Implementation and Verification of a Security Patch or Upgrade

- Before installing any patch, **verify the source**
- Use proper **patch management program** to validate files versions and checksums before deploying security patches
- The patch management tool must be **able to monitor the patched systems**
- The **patch management team** should check for updates and patches regularly

GFI LanGuard

GFI LanGuard's patch management scans your network automatically and also **installs and manages security and non-security patches**



The screenshot shows the GFI LanGuard 2016 Dashboard. On the left, a sidebar lists network devices under 'Entire Network'. The main area displays two large circular metrics: '119' in red and '275' in green. Below these are four categories: Missing Security Updates, Missing Non-Security Updates, Missing Service Packs and Update Rollups, and Installed Security Updates. A table titled 'Installed Security Updates' lists various updates with columns for Patch name, Date posted, Severity, and Applies to.

Patch name	Date posted	Severity	Applies to
FOXITR6140217: Foxit Reader 6.0.1.140217	2014-02-20	Critical	Foxit Reader
JAV/A8051: Java Runtime Environment 7.0_79	2015-07-14	Critical	Java Runtime Env.
mrsa2015-59; mrsa2015-60; mrsa2015-61	2015-06-30	Moderate	Firefox
MS08-061: MSXML 6.0 RTM Security Update	2012-04-04	Critical	SQL Server
MS10-028: Security Update for Microsoft .NET Framework 3.5.1	2009-05-08	Critical	SDK Component
MS10-035: Security Update for Microsoft .NET Framework 3.5.1	2009-09-08	Moderate	Developer Toolkit
MS10-035: Security Update for Microsoft .NET Framework 3.5.1	2009-08-11	Moderate	Developer Toolkit
MS10-062: Security Update for Microsoft .NET Framework 3.5.1	2009-10-16	Low	Developer Toolkit
MS10-062: Security Update for Microsoft .NET Framework 3.5.1	2009-10-16	Low	Developer Toolkit
MS11-025: Security Update for Microsoft .NET Framework 3.5.1	2012-01-24	Important	Developer Toolkit
MS11-025: Security Update for Microsoft .NET Framework 3.5.1	2012-01-24	Important	Developer Toolkit

<https://www.gfi.com>

Patch Management Tools

Symantec Client Management Suite

<https://www.symantec.com>


MaaS360 Patch Analyzer

<https://www.ibm.com>


Solarwinds Patch Manager

<https://www.solarwinds.com>


Kaseya Security Patch Management

<https://www.kaseya.com>


Software Vulnerability Manager

<https://www.flexerasoftware.com>



Module Flow

1

Web Server Concepts

2

Web Server Attacks

3

Web Server Attack Methodology

4

Web Server Attack Tools

5

Counter-measures

6

Patch Management

7

Web Server Security Tools

8

Web Server Pen Testing

Web Application Security Scanners

Syhunt Hybrid

Syhunt Hybrid helps to automate **web application security** testing and guard organization's **web infrastructure** against various web application security threats

The screenshot shows the Syhunt Hybrid interface. At the top, there's a toolbar with File, Edit, View, Recent Hosts, Advanced, Tools, Help. Below it is a navigation pane with 'Scanned Hosts' expanded, showing 'demo.syhunt.com:80' with sub-options like 'Host Information', 'Emails', 'JS-Based Pages', 'Site Errors', 'Source Structure', 'Vulnerable Source', 'Vulnerable URLs', 'Web Structure', 'detection', 'file.php', 'file_sample.php', 'x_basic.php', 'x_basic_plunker.php', 'x_form.php', 'exploitation', 'index_hidden.php', and 'intelligence'. The main panel displays 'Application Checker' results for 'demo.syhunt.com' with a progress bar at 40%, 'Requests: 27 seconds ago', 'Timeover: 00:00:01', 'Retries: 0', 'Optimized: 0%', and 'Vulnerabilities: 19'. Below this is an 'Event List' tab showing logs of the scan process, including launching SandicCS.exe, crawling the site, and finding various vulnerabilities like 'x_basic_plunker.php:05' and 'index_hidden.php:XSS'. At the bottom, there's a status bar with 'Check (Open Site Scripting) / 55-28... 1 of 28!' and a browser address bar showing 'http://www.syhunt.com'.

N-Stalker

N-Stalker is a **WebApp Security Scanner** to search for vulnerabilities such as SQL injection, XSS, and known attacks

The screenshot shows the N-Stalker interface. It has tabs for 'N-Stalker Scanner' and 'Scan Options'. Under 'Scan Options', there are sections for 'Start Scan', 'Start Proxy', 'Close Session', 'Session Control', 'Threads # (5)', 'Encode URI (WAF)', 'HTTP Settings', 'Control Options', 'Timeout (15)', 'Track Spider', 'Debug HTTP', 'IP Keyword Filter', and 'False-Positive Control'. The main area has tabs for 'Website Tree', 'Scanner Events', and 'Scanner Dashboard'. The 'Scanner Events' tab shows a progress bar for a 'Spider' task that is 'Completed'. The 'Scanner Dashboard' tab provides detailed statistics: 'Scan Session' (Start Time: Dec 21, 2017 01:28:47, Duration: 0 hours 3 minutes), 'Spider Engine' (Crawled URLs: 11, Crawled Hosts: 1, Default Page Size: 53,379 bytes), 'Vulnerabilities' (found 1 vulnerability at http://www.goodshopping.com), and 'Scan Engine' (Total Requests: 179, Failed Requests: 0, Attacks Sent: 64, 404 Errors: 49, 302 Redirection: 0). It also includes a 'Progress Details' section with a bar chart for 'High (1)', 'Mid (2)', 'Low (1)', and 'Info (5)' vulnerabilities, and a 'Network' section with metrics like Bytes Sent: 87,064, Bytes Received: 809,453, Avg Response Time: 8.93 s, Avg Transfer Rate: 496.10 Kbps, and Requests/Minute: 29.00 req/min. At the bottom, there's a 'Scan Modules' table with rows for 'Sensitive File Search Assess 1', 'WebServer Infrastructure Ass 4', '3rd-Party Package Scanner', and 'N-Stalker Spider Module', all showing 100% progress. A browser address bar at the bottom shows 'https://www.nstalker.com'.

Web Server Security Scanners

ScanMyServer

- ScanMyServer is used to **find security vulnerabilities** in a web site or a web server
- It can **generate comprehensive test reports** and also can assist in fixing security problems that might exist in company's website or web server

SCAN MY SERVER

Secure Site
Nov-8-2017

Scan About Contact FAQ
ISPs, MSPs, Security Cons...

Test the Security of Your Website or Blog - Free

http:// Scan
(Enter correct URL: Is it yoursite.com or www.yoursite.com?)
<https://www.scanmyserver.com>

QualysGUARD Malware Detection Service

MDS Dashboard Scans Reports Assets KnowledgeBase Help Demo User Log Out Add Site

Dashboard
Last login: 12 Feb 2012
0 scans since last login

Total Sites	Infected Sites	Total Infections
13	4	45

Infections Detected
Date range: 7 days, 14 days

Infections by total scans

Day	Infections
Tue	1
Wed	1
Thu	1
Fri	1
Sat	3
Sun	2

Infected Sites

Site	Infections	Severity
My Test Site Pt.2	1	MED
http://www.mwtest.info/malware-demos-named/Kill.../	1	MED
http://www.mwtest.info/malware-demos-named/sm.../	16	HIGH
Demo Ascr Scan	3	HIGH
http://www.mwtest.info/malware-demos-named/sm.../	3	HIGH
Demos Owned	25	HIGH
http://www.mwtest.info/malware-demos-named/	25	HIGH

Your Last Scans

Scan	Scan Date	Status	Severity
Lite Scan-Schedule	12 Feb 2012	Host No... -	-
Bank of Qualys-Schedule	12 Feb 2012	Finished	SAFE
Relaunch My Test Site Pt.2-Schedule	12 Feb 2012	Finished	MED
My Test Site Pt.2-Schedule	11 Feb 2012	Finished	MED
Lite Scan-Schedule	11 Feb 2012	Host No... -	-

Your Upcoming Scans

Scan name	Starts	Occurs
My Test Site Pt.2-Schedule	12 Feb 2012	DAILY
http://www.mwtest.info/malware-demos-named/Kill.../	13 Feb 2012	DAILY
http://www.mwtest.info/malware-demos-named/sm.../	13 Feb 2012	DAILY
Bank of Qualys-Schedule	13 Feb 2012	DAILY
Lite Scan-Schedule	13 Feb 2012	WEEKLY

<https://www.qualys.com>

QualysGUARD Malware Detection Service proactively scan their web sites for **malware**, providing automated alerts and in-depth reporting to enable prompt identification and resolution

Web Server Security Tools

Acunetix WVS

Acunetix WVS **checks web applications** for SQL injections, cross-site scripting, etc. and creates professional security audit and regulatory compliance reports

The screenshot shows the Acunetix WVS interface. On the left is a sidebar with navigation links: Dashboard, Targets, Vulnerabilities, Scans, Reports, and Settings. The main window has tabs at the top: Back, Stop Scan, Generate Report, WAF Export..., Group By: None, and Filter. The 'Vulnerabilities' tab is selected. Below it is a table with columns: Severity, Vulnerability, and URL. A red box highlights the first few rows of the table, which list various security issues found on the target website.

Severity	Vulnerability	URL
Info	Blind SQL Injection	http://www.moviescope.com/
Info	Blind SQL Injection	http://www.moviescope.com/
Info	Microsoft IIS tilde directory enumeration	http://www.moviescope.com/
Info	Unencrypted __VIEWSTATE parameter	http://www.moviescope.com/
Info	Vulnerable Javascript library	http://www.moviescope.com/
Info	ASP.NET debugging enabled	http://www.moviescope.com/
Info	ASP.NET version disclosure	http://www.moviescope.com/
Info	Clickjacking: X-Frame-Options header missing	http://www.moviescope.com/
Info	Login page password-guessing attack	https://www.moviescope.com/
Info	OPTIONS method is enabled	http://www.moviescope.com/

<https://www.acunetix.com>



Fortify WebInspect
<https://software.microfocus.com>



Retina CS
<https://www.beyondtrust.com>



Nscan
<https://nscan.hypermart.net>



NetIQ Secure Configuration Manager
<https://www.netiq.com>



SAINT Scanner
<http://www.saintcorporation.com>

Module Flow

1

Web Server Concepts

2

Web Server Attacks

3

Web Server Attack Methodology

4

Web Server Attack Tools

5

Counter-measures

6

Patch Management

7

Web Server Security Tools

8

Web Server Pen Testing

Web Server Penetration Testing

- Web server pen testing is used to **identify, analyze, and report vulnerabilities** such as authentication weaknesses, configuration errors, protocol related vulnerabilities, etc. in a web server
- The best way to perform penetration testing is to **conduct a series of methodical and repeatable tests**, and to work through all of the different application vulnerabilities

Why Web Server Pen Testing?

Verification of Vulnerabilities

To exploit the vulnerability in order to test and fix the issue

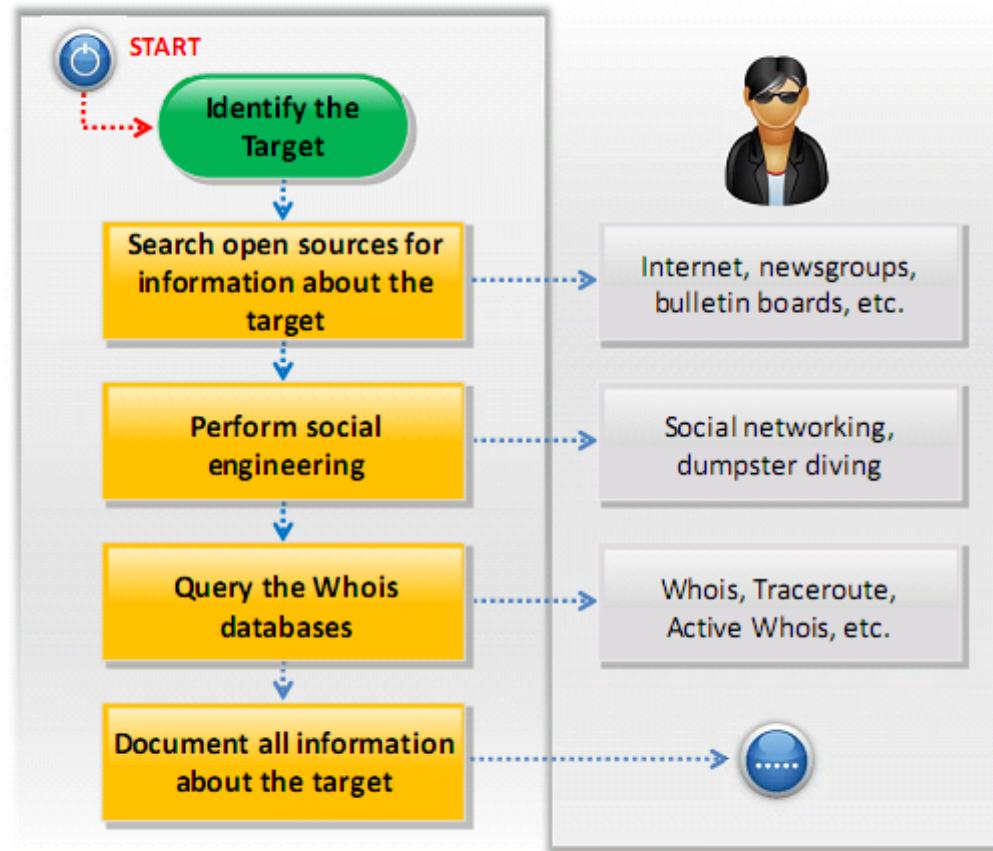
Remediation of Vulnerabilities

To retest the solution against vulnerability to ensure that it is completely secure

Identification of Web Infrastructure

To identify make, version, and update levels of web servers; this helps in selecting exploits to test for associated published vulnerabilities

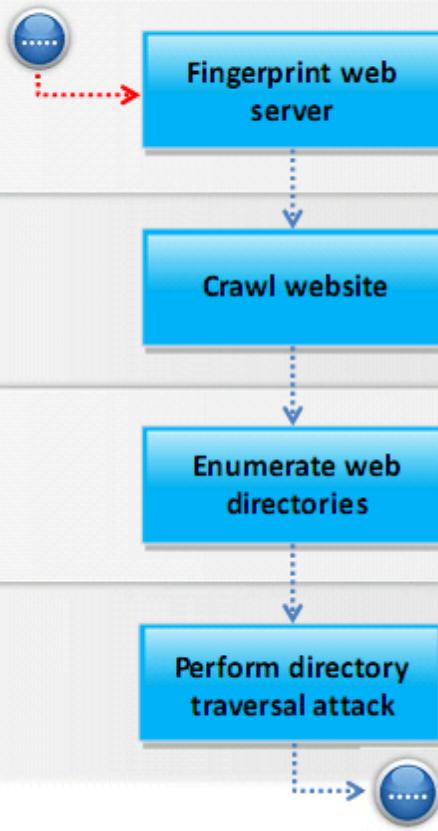
Web Server Penetration Testing (Cont'd)



- Web server penetration testing starts with **collecting as much information** as possible about an organization ranging from its physical location to operating environment
- Use **social engineering techniques** to collect information such as human resources, contact details, etc. that may help in **web server authentication testing**
- Use **Whois database query tools** to get the details about the target such as domain name, IP address, administrative contacts, Autonomous System Number, DNS, etc.
- Note:** Refer Module 02: Footprinting and Reconnaissance for more information gathering techniques

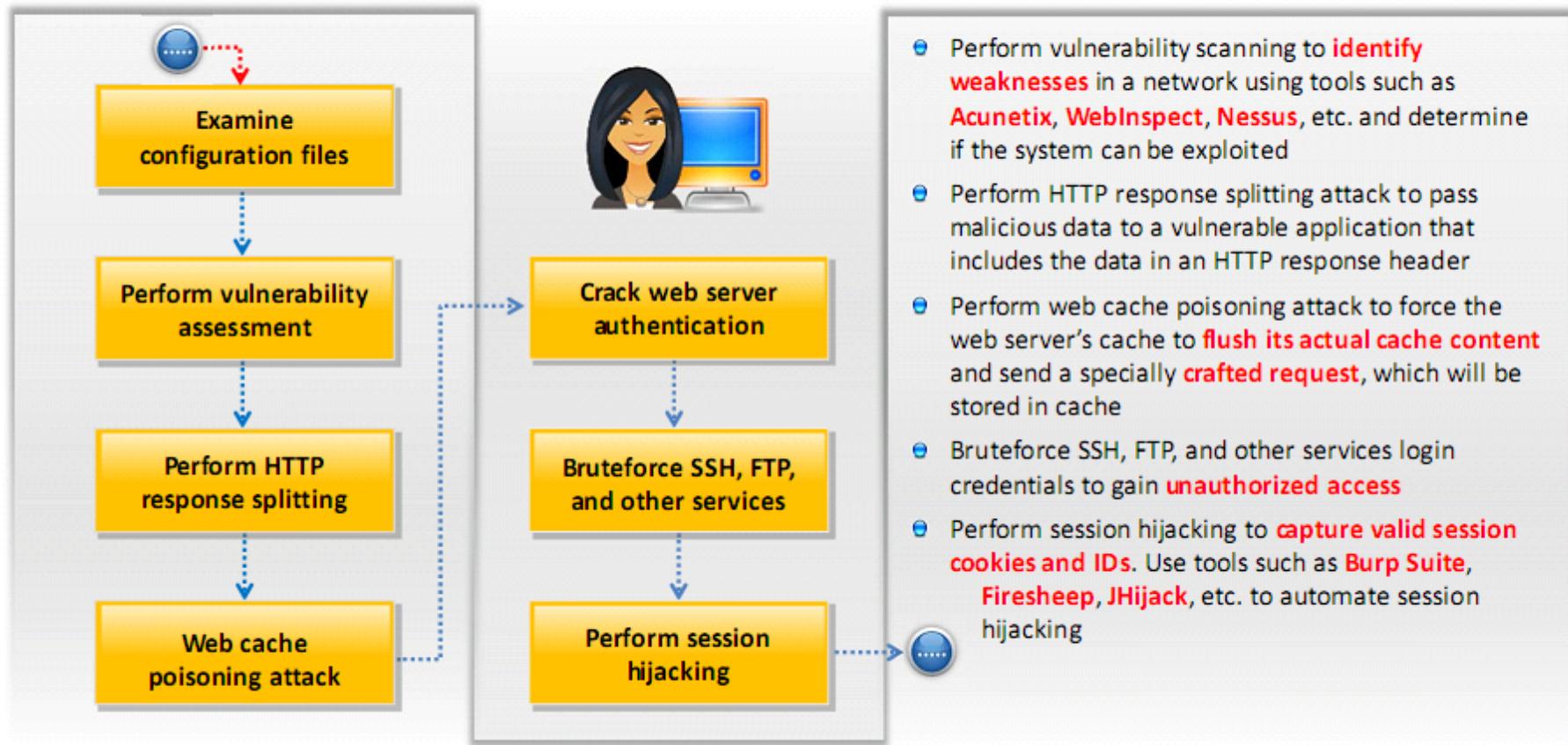


Web Server Penetration Testing (Cont'd)



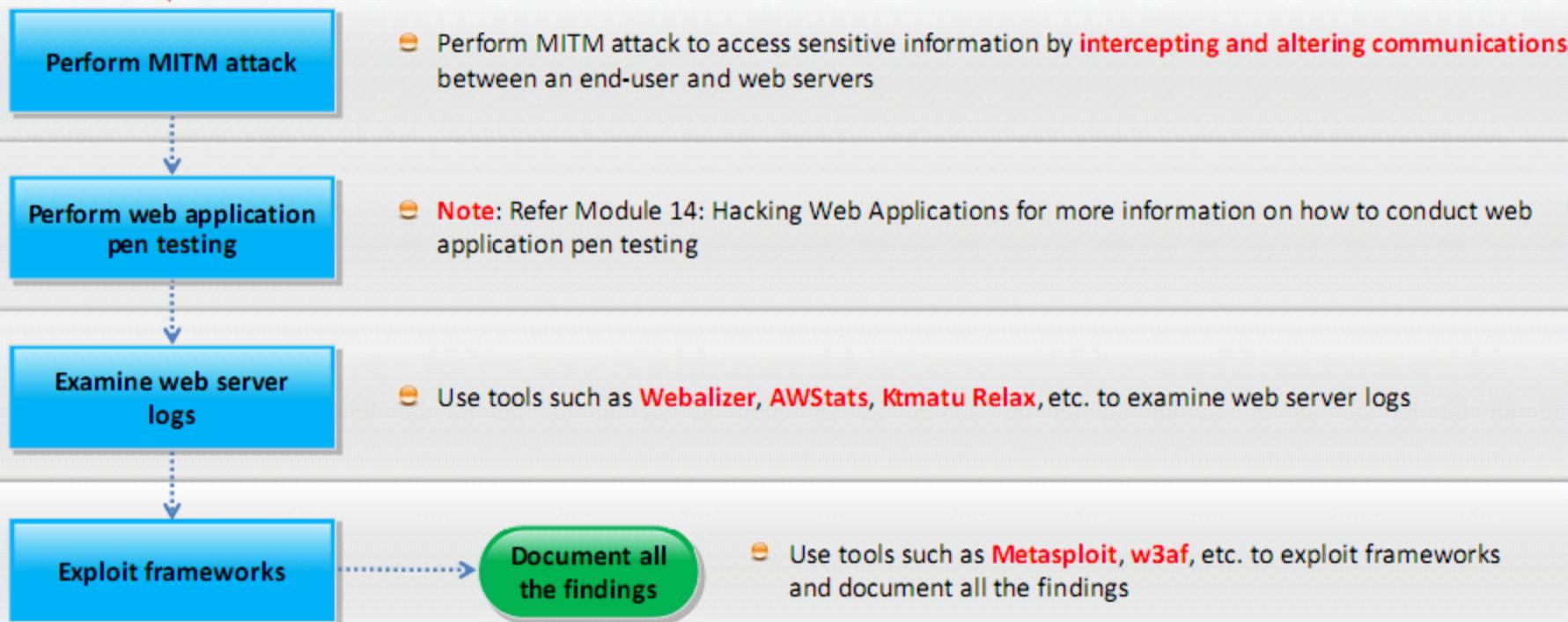
- Fingerprint web server to gather information such as server name, server type, operating systems, applications running, etc. using tools such as **Netcraft**, **httprecon**, and **ID Serve**
- **Crawl website** to gather specific types of information from web pages, such as email addresses
- Enumerate **web server directories** to extract important information such as web functionalities, login forms, etc.
- Perform **directory traversal** attack to access restricted directories and execute commands outside of the web server's root directory

Web Server Penetration Testing (Cont'd)



- Perform vulnerability scanning to **identify weaknesses** in a network using tools such as **Acunetix**, **WebInspect**, **Nessus**, etc. and determine if the system can be exploited
- Perform HTTP response splitting attack to pass malicious data to a vulnerable application that includes the data in an HTTP response header
- Perform web cache poisoning attack to force the web server's cache to **flush its actual cache content** and send a specially **crafted request**, which will be stored in cache
- Bruteforce SSH, FTP, and other services login credentials to gain **unauthorized access**
- Perform session hijacking to **capture valid session cookies and IDs**. Use tools such as **Burp Suite**, **Firesheep**, **JHijack**, etc. to automate session hijacking

Web Server Penetration Testing (Cont'd)



Web Server Pen Testing Tools

CORE Impact

- CORE Impact finds vulnerabilities on an organization's web server
- This tool allows a user to evaluate the security posture of a web server using the same techniques employed by today's cyber-criminals

PowerShell Empire - Core Impact

File View Modules Tools Help

Modules

- agent
- Agent Process Injector
- Blue Coat Authentication and Authorization Agent Buffer Overflow
- CA Brightline AFCeeve Backup SQL agent exploit
- Check Agent Capabilities
- Check for sensitive information using SQL Agent
- Cisco IOS Agent - Privilege Escalation
- Cisco IOS Agent - Privilege Escalation Clean Up
- Command Shell using Module Agent
- Command Shell using SQL Agent
- Connect Agent
- Create Packed Agent Using WMI
- Delete agent changing root
- Deploy Agent Link Agent
- Exploit PowerShell Empire agent
- Disconnect Agent
- Eaten ELGSoft Buffer Overflow Exploit
- FATF - Exploit a Cisco Agent R-MSR Overflow Exploit
- Heuvelink Packaged Enterprise Data Protection EEEC_B4R user Name
- HP Data Protector Call Manager Opcode 211 Buffer Overflow
- HP Data Protector Call Manager Opcode 259 Remote Code Execution
- HP Data Protector Call Manager Opcode 263 Buffer Overflow
- HP Data Protector Client EEEC_SETUP Remote Code Execution
- HP Data Protector EEEC_B4R Remote Command Execution
- HP Data Protector EEEC_CMD Exploit
- HP Data Protector OracleNet Listener Buffer Overflow Exploit
- HP Data Protector Remote Command Execution Exploit
- HP Diagnostic Server managementservice Remote Buffer Overflow
- HP OpenView Performance Agent code Opcode 0x4 Buffer Overflow
- HP OpenView Performance Agent code Opcode 0x1C Buffer Overflow
- HP OpenView Storage Data Processor Remote Buffer Overflow
- HP ProCurve Agent Sender Remote Code Execution Exploit
- HP Storage Data Protector MSG_PROTOCOL Buffer Overflow
- HTTP Proxy over TCP Proxy Plugin
- In-process Agent Backdoor Connector
- In-process Agent Backdoor Installer
- Inject Agent into Virtual Machine
- Install Agent using Access Token (Integrated)
- Install Agent using PowerShell Empire agent
- Install Agent using SAMS
- Install Agent using rsh
- Install Agent using TeamViewer
- Install Agent using telnet

Networks Client Side Web

Hosts Wireless Mobile Identities Search Folders Tags

Search... Name IP OS Arch

Visibility: Root (1)

Networks: 192.168.21.0/24

- localhost 192.168.21.131 Windows x86-64

Visibility: localhost (0)

Networks: 192.168.21.0/24

- R051MB014 192.168.21.3 Windows x86-64
- 192.168.21.2 192.168.21.2 Windows i386
- AVV-IV99-95QH-F 192.168.21.34 Windows i386
- [agent1] i386 agent(i1) i386 agent(i2) i386 agent(i3) i386 agent(i4)
- AVV-GA95-QT94S 192.168.21.36 Windows x86-64
- [agent1] 192.168.21.37 192.168.21.37 Linux i386
- 192.168.21.254 192.168.21.254 Unknown Unknown

Executed Modules

Name	Started	Finished	Status	Source Agent	Results
PowerShell Empire server ...	11/26/2016 6:58:56 PM	11/26/2016 6:58:52 PM	Finished	/localagent	/localagent
Web Server	11/26/2016 6:59:56 PM	11/26/2016 6:59:53 PM	Finished	/localagent	/localagent
Deploying PowerShell Empire agen...	11/26/2016 7:00:57 PM	11/26/2016 7:00:57 PM	Finished	/localagent	/localagent
Web Server	11/26/2016 7:00:58 PM	11/26/2016 7:00:58 PM	Running	/localagent	/localagent
Forward request to Power... T11/26/2016 7:00:12 PM	11/26/2016 7:00:12 PM	Finished	/localagent	/localagent	
Forward request to Power... T11/26/2016 7:00:13 PM	11/26/2016 7:00:13 PM	Finished	/localagent	/localagent	
Forward request to Power... T11/26/2016 7:00:14 PM	11/26/2016 7:00:14 PM	Finished	/localagent	/localagent	
Forward request to Power... T11/26/2016 7:00:20 PM	11/26/2016 7:00:20 PM	Finished	/localagent	/localagent	
Forward request to Power... T11/26/2016 7:00:25 PM	11/26/2016 7:00:25 PM	Finished	/localagent	/localagent	
Forward request to Power... T11/26/2016 7:00:31 PM	11/26/2016 7:00:31 PM	Finished	/localagent	/localagent	
Forward request to Power... T11/26/2016 7:00:36 PM	11/26/2016 7:00:36 PM	Finished	/localagent	/localagent	
Forward request to Power... T11/26/2016 7:00:41 PM	11/26/2016 7:00:41 PM	Finished	/localagent	/localagent	
Forward request to Power... T11/26/2016 7:00:47 PM	11/26/2016 7:00:47 PM	Finished	/localagent	/localagent	
Forward request to Power... T11/26/2016 7:00:52 PM	11/26/2016 7:00:52 PM	Finished	/localagent	/localagent	

Module Log

```
Connecting with the agent ...
Module "Deploy PowerShell Empire agent" (v184514) started execution on Sat Nov 26 19:00:00 2016
*** Debugging mode is enabled
Created listener "Impact-agent(1)-:40000"
Launching agent
A new PowerShell Empire agent with name "KILLUS7P4EIR2RM67" has been deployed
Module finished execution after 14 secs.
```

Module Output Module Log Module Parameters

Quick Information

agent[1]

General

Type Agent

Visibility Path (192.168.21.131\agent1) i386-64

Architecture i386-GH35QDCT94S

Host True

Crypto Channel iWVN-QN5510KTB49JACDR803V0035-46297.exe

File Name [selected]

Proxy Agent /localagent

Deployed With Install Agent using SAMS

<https://www.coresecurity.com>

CAP NUM SCR 7:05 PM 11/26/2016

Web Server Pen Testing Tools

- Immunity CANVAS (<http://www.immunitysec.com>)
- Arachni (<http://www.arachni-scanner.com>)
- WebSurgery (<http://sunrisetech.gr>)

Module Summary

- ❑ Web servers assume critical importance in the realm of Internet security
- ❑ Vulnerabilities exist in different releases of popular web servers and respective vendors patch these often
- ❑ The inherent security risks owing to the compromised web servers have impact on the local area networks that host these websites, even on the normal users of web browsers
- ❑ Looking through the long list of vulnerabilities that had been discovered and patched over the past few years, it provides an attacker ample scope to plan attacks to unpatched servers
- ❑ Different tools/exploit codes aid an attacker in perpetrating web server's hacking
- ❑ Countermeasures include scanning for the existing vulnerabilities and patching them immediately, anonymous access restriction, incoming traffic request screening, and filtering