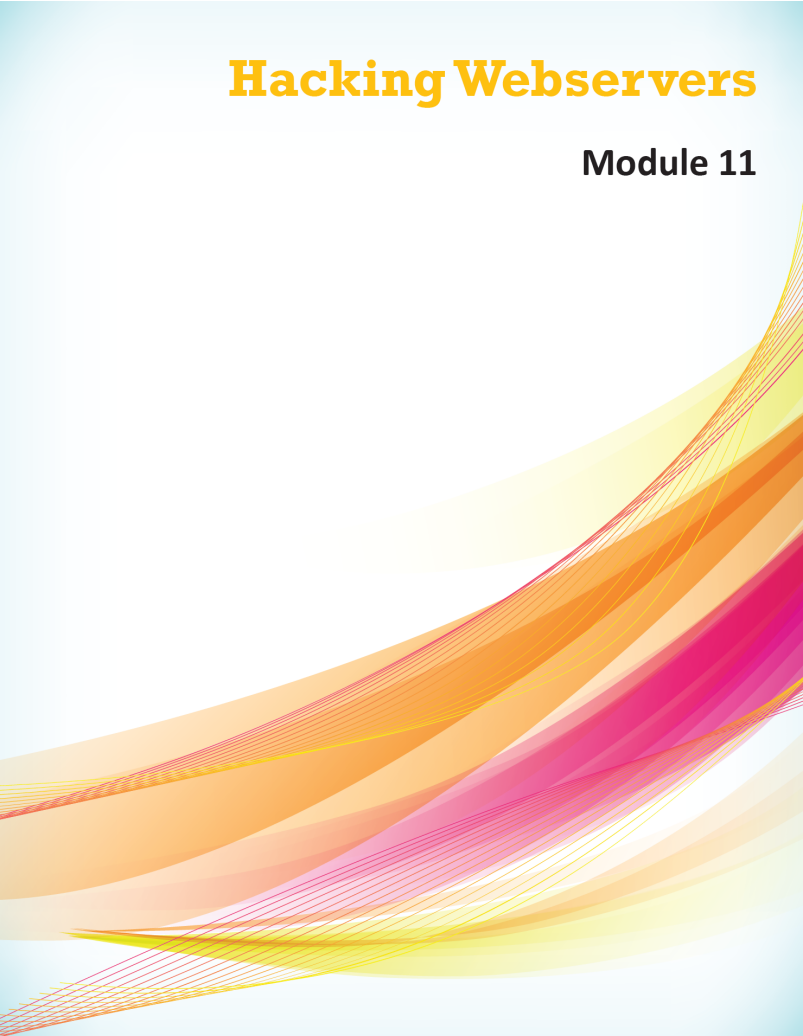
Hacking Webservers

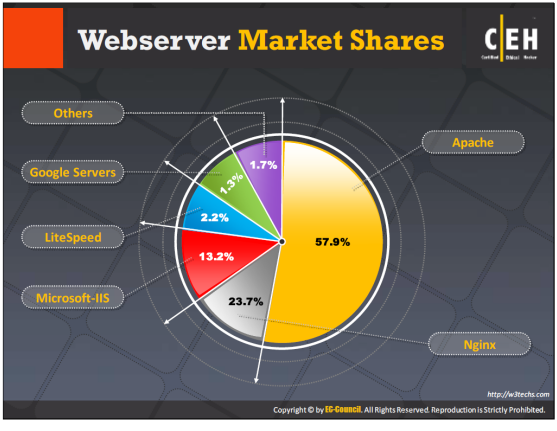
Module 11



Ethical Hacking and Countermeasures v9

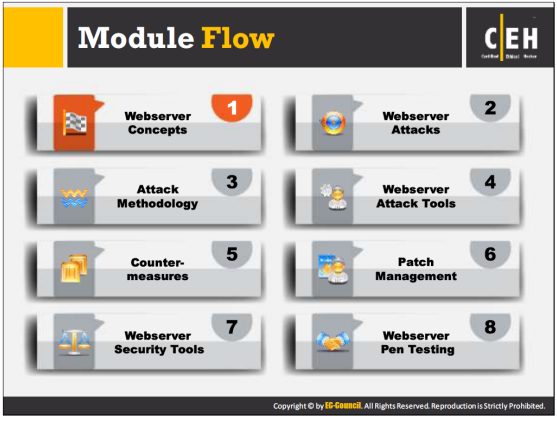
Module 11: Hacking Webservers

Exam 312-50



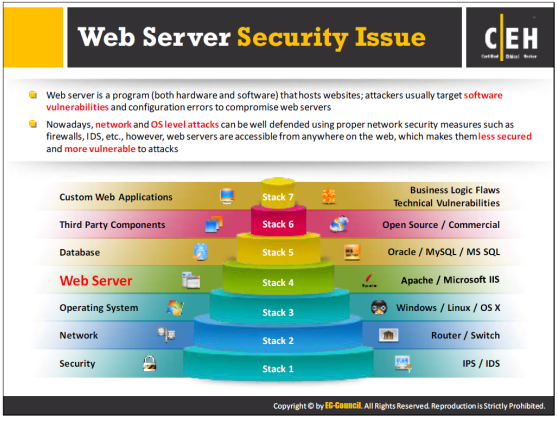
Most organizations consider their web presence to be an extension of themselves. Organizations create their web presence on the World Wide Web using websites associated with their business. Webservers are a critical component of a web infrastructure. A single vulnerability in webserver configuration may lead to a security breach on websites. This makes webserver security critical to the normal functioning of an organization.

This module starts with an overview of webserver concepts. The module provides an insight into various webserver attacks, webserver attack methodology, and webserver attack tools. Later the module describes countermeasures against webserver attacks, patch management, and webserver security tools. The module ends with an overview of pen testing steps an ethical hacker should follow to perform the security assessment of the target.



To understand webserver hacking, first you should understand webserver concepts: what a webserver is, how it functions, and the other elements associated with it.

This section gives a brief overview of the webserver and its architecture. It will also explain common reasons or mistakes made that allow attackers to hack a webserver successfully. This section also describes the impact of attacks on the webserver.



A webserver is a hardware/software application that hosts websites and make them accessible over the internet. A webserver, along with a browser, successfully implements client-server model architecture in which the webserver plays the server part in the model and the browser acts as the client. To host websites, a webserver actually stores various web pages of the websites and delivers the particular web page upon request. Each webserver has a domain name and the IP address associated with that domain name. A webserver can host more than one website. Any computer can act as a webserver if it has specific server software (a webserver program) installed and is connected to the internet. Webservers are chosen based on their capability to handle server-side programming, security characteristics, publishing, search engine, and site building tools. Apache, Microsoft Internet Information Services (IIS), Nginx, Google, and Tomcat are some of today's most widely used webservers.

An attacker may exploit the vulnerability that exists in the software component of the webserver implementation to compromise website security.

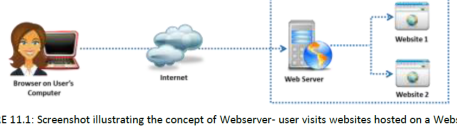


FIGURE 11.1: Screenshot illustrating the concept of Webserver- user visits websites hosted on a Webserver

Organizations can defend most Network level and OS level attacks by using network security measures such as firewalls, IDS, IPS etc., and by following security standards and guidelines. This forces attackers to turn their attention to carry out webserver and web application-level attacks. As webserver hosting web applications is accessible from anywhere over the internet. This makes webservers an attractive target. A poorly configured webserver can punch a hole in the most carefully designed firewall system. Attackers can exploit a poorly configured webserver with known vulnerabilities to compromise web application security. A leaky server can harm an organization. Today, 75% of cyber attacks are caused by vulnerabilities that exist in webservers and web applications.

Common Goals behind Webserver Hacking

Attackers carry out webserver attacks with certain goals in mind. These goals may be either technical or non-technical. For example, attackers may breach security of the webserver and steal sensitive information for financial gains, or only for the sake of curiosity.

Some goals behind a webserver attack

- Stealing credit cards or other sensitive credentials using phishing techniques
- Integrating the server in a botnet in order to perform Denial of Service (DoS) or Distributed Denial of Service (DDoS) attack
- Compromising a database
- Obtaining closed-source applications
- Hiding and redirecting traffic
- Escalating privileges

Some attacks are not made to attain financial gains, but for personal reasons:

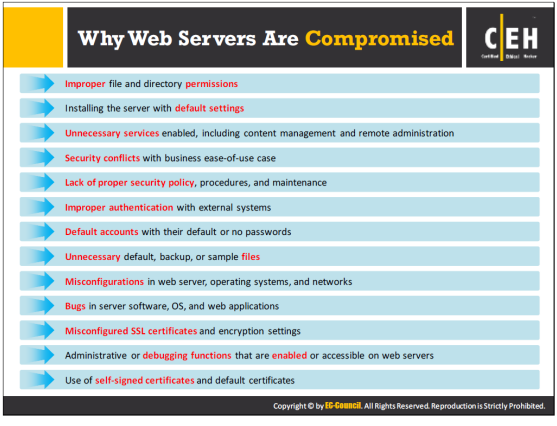
- For the sake of pure curiosity
- For the sake of achieving a self-set intellectual challenge
- To damage the target organization's reputation

Dangerous Security Flaws Affecting Webserver Security

Webserver configuration by poorly trained system administrators may leave security vulnerabilities in the webserver. Inadequate knowledge, negligence, laziness, and inattentiveness towards security can pose the biggest threats to webserver security.

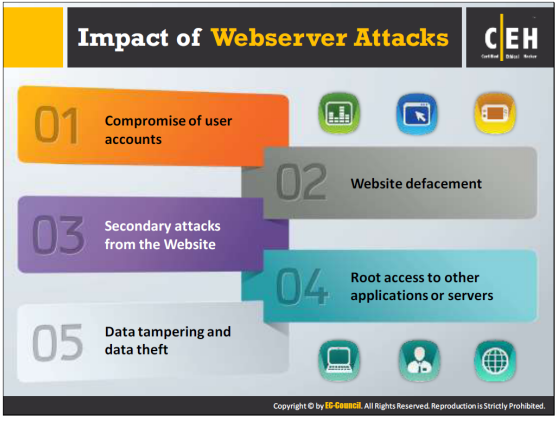
Some common oversights that make a webserver vulnerable to attacks include:

- Not updating the webserver with the latest patches
- Using the same sys admin credentials everywhere
- Allowing unrestricted internal and outbound traffic
- Running unhardened applications and servers
- Complacency



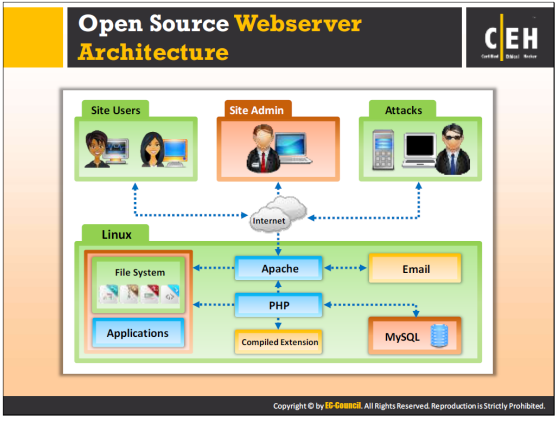
There are inherent security risks associated with webservers, the local area networks that host websites, and the users who access these websites using browsers.

- **Webmaster's Concern:** From a webmaster's perspective, the biggest security concern is that the webserver can expose the Local area network (LAN) or the corporate intranet to threats the Internet poses. These may be in the form of viruses, Trojans, attackers, or the compromise of information itself. Bugs in software programs are often the source of security lapses. Webservers that are large complex devices also come with these inherent risks. In addition, the open architecture of the webservers allows arbitrary scripts to run on the server side while replying to the remote requests. Any CGI script installed at the site may contain bugs that are potential security holes.
- **Network Administrator's Concern:** From a network administrator's perspective, a poorly configured webserver poses another potential hole in the local network's security. While the objective of a web is to provide controlled access to the network, too much control can make a web almost impossible to use. In an intranet environment, the network administrator has to be careful about configuring the webserver, so that the legitimate users are recognized and authenticated, and groups of users are assigned distinct access privileges.
- **End User's Concern:** Usually, the end user does not perceive any immediate threat, as surfing the web appears both safe and anonymous. However, active content, such as ActiveX controls and Java applets, make it possible for harmful applications, such as viruses, to invade the user's system. In addition, active content from a website browser can be a conduit for malicious software to bypass the firewall system and permeate the local area network.



Attackers can cause various kinds of damage to an organization by attacking a webserver. The damage includes:

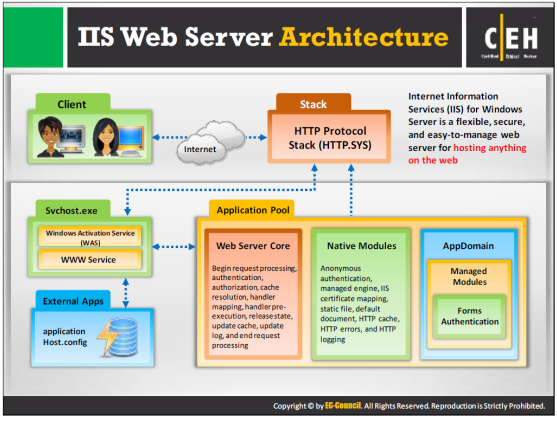
- **Compromise of user accounts:** Webserver attacks are mostly concentrated on user account compromise. If the attacker is able to compromise a user account, then the attacker can gain a lot of useful information. Attacker can use the compromised user account to launch further attacks on the webserver.
- **Data tampering:** An attacker can alter or delete the data, and can even replace the data with malware in order to compromise whoever connects to the webserver.
- **Website defacement**: Attackers completely change the appearance of the website by replacing the original data. They change the website look by changing the visuals and displaying different pages with messages of their own.
- **Secondary attacks from the website:** An attacker who compromises a webserver can use the server to launch further attacks on various websites or client systems.
- **Data theft:** Data is one of the main assets of the organization. Attackers can get access to sensitive data like financial records, future plans, or the source code of a program.
- **Root access to other applications or server:** Root access is the highest privilege one gets to log in to a network, be it a dedicated server, semi-dedicated, or virtual private server. Attackers can perform any action once they get root access to the server.



Open-source webserver architecture typically uses Linux, Apache, MySQL, and PHP (LAMP) as principal components.

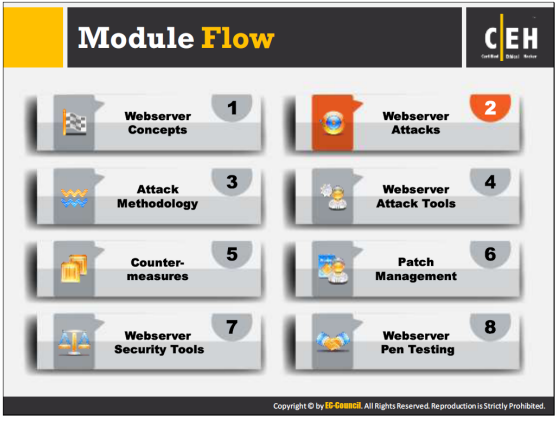
Functions of principal components in open source webserver architecture:

- Linux is a the server's OS that provides secure platform for the webserver
- Apache is a the webserver component that handles each HTTP request and response
- MySQL is a relational database used to store the webserver's content and configuration information
- PHP is the application layer technology used to generate dynamic web content

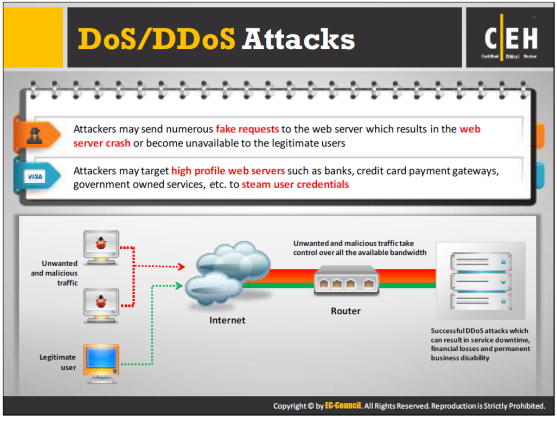


Internet Information Service (IIS) is a webserver application developed by Microsoft for Windows. It supports HTTP, HTTPS, FTP, FTPS, SMTP, and NNTP.

IIS includes several components, including a protocol listener such as HTTP.sys, and services, such as World Wide Web Publishing Service (WWW Service) and Windows Process Activation Service (WAS). Each component functions in application and webserver roles. These functions may include listening for requests, managing processes, and reading configuration files, etc.



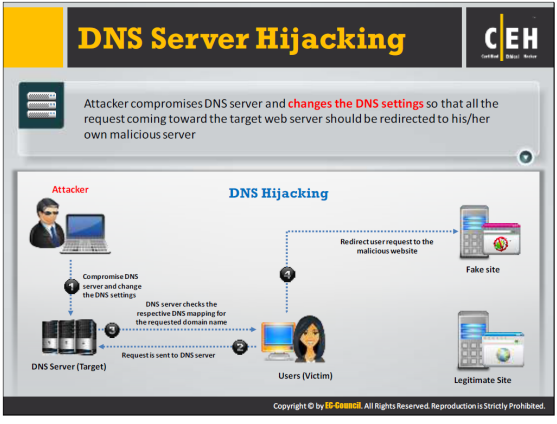
An attacker can use many attack techniques to compromise a webserver such as DoS/DDoS, DNS server hijacking, directory traversal, Man-in-the-Middle (MITM)/sniffing, phishing, website defacement, HTTP response splitting, web cache poisoning, HTTP response hijacking, SSH brute force, webserver password cracking, etc. This section describes these possible attacks in detail.



A DoS/DDoS attack involves flooding targets with numerous fake requests so that the target stops functioning and will be unavailable to legitimate users. Attackers also perform DoS/DDoS attacks on the webserver in the same way. Using a webserver DoS/DDoS attack, an attacker attempts to take the webserver down or make it unavailable to legitimate users. A webserver DoS/DDoS attack often targets high-profile webservers such as banks, credit card payment gateways, and even root name servers.

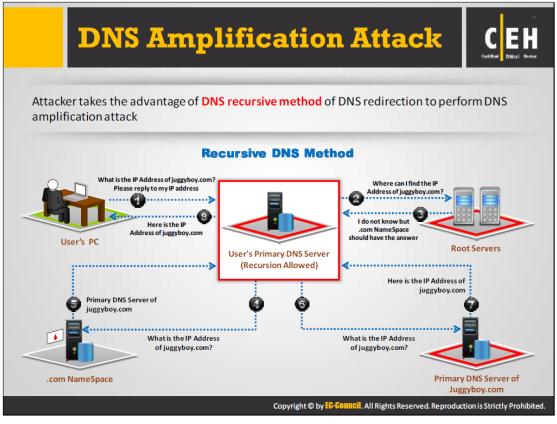
To crash the webserver running the application, attacker targets the following services by consuming the webserver with fake requests.

- Network bandwidth
- Server memory
- Application exception handling mechanism
- CPU usage
- Hard disk space
- Database space



Domain Name System (DNS) resolves a domain name to its corresponding IP address. A user queries the DNS server with a domain name and it delivers the corresponding IP address.

In a DNS server hijacking, an attacker changes the mapping settings of the target DNS server to redirect towards a rogue DNS server so that it would redirect the user's requests to the attacker's rogue server. Thus, when the user types the legitimate URL in a browser, the settings will redirect to the attacker's fake site.



Recursive DNS Query is a method of requesting DNS mapping. The query goes through domain name servers recursively until it fails to find the specified domain name to IP address mapping.

Steps involved in processing Recursive DNS request:

- **Step1:**

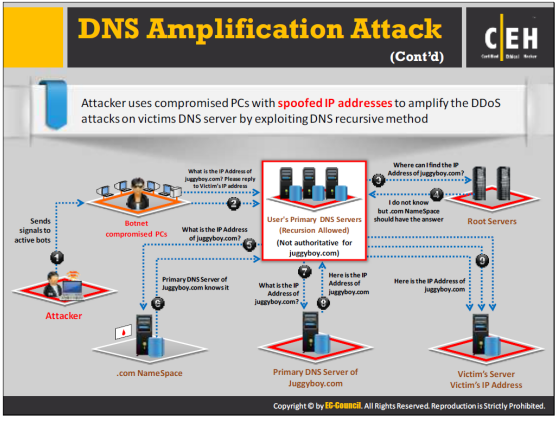
  Users who want to resolve the IP address for a specific domain send a DNS query to the Primary DNS Server specified in its TCP/IP properties.

- **Step 2 to 7:**

  If the requested DNS mapping is not present on the user's Primary DNS Server, then it will forward the request to the root server. The root server will forward the request to .com namespace where the user could find DNS mappings. This process repeats recursively until DNS mapping is resolved.

- **Step 8:**

  Ultimately, when the system finds the primary DNS server for the requested DNS mapping, then it generates a cache for the IP address in the user's primary DNS server.



Attackers exploit recursive DNS queries to perform a DNS amplification attack that results in DDoS attacks on the victim's DNS server.

Steps involved in DNS Amplification Attack:

- **Step1**:

  The attacker instructs compromised hosts (bots) to make DNS queries in the network.

- **Step2**:

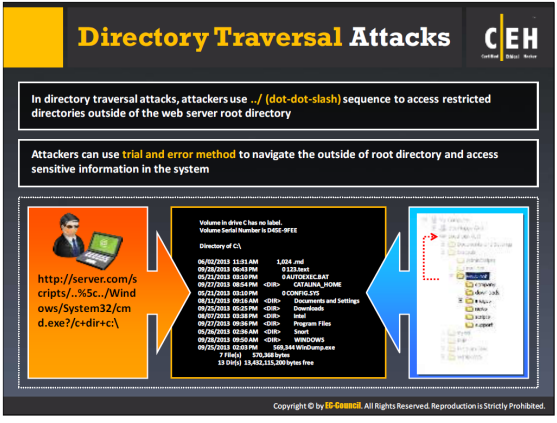
  All the compromised hosts use spoofed victim IP addresses and send DNS query requests to the victim's primary DNS server configured in its TCP /IP settings.

- **Step 3 to Step 8**:

  If the requested DNS mapping is not present on the victim's primary DNS server, the server forwards the requests to the root server. The root server will forward the request to .com or respective TLD namespaces. This process repeats recursively until the victim's primary DNS server resolves the DNS mapping request.

- **Step 9**:

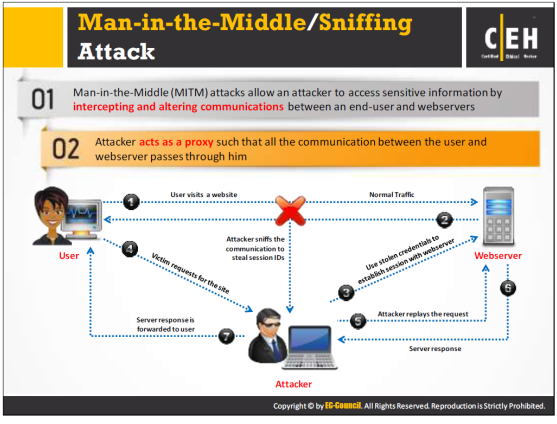
  After the primary DNS server finds the DNS mapping for the victim's request, it sends a DNS mapping response to the victim's IP address. This response goes to the victim as bots are using the victim's IP address. The replies to a large number of DNS mapping requests from the bots results in DDoS on the victim's DNS server.



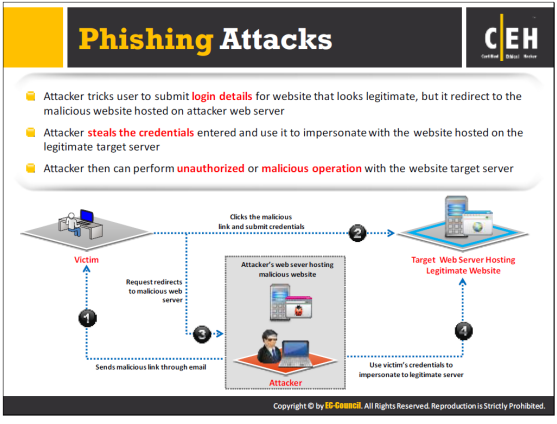
An attacker may be able to perform a directory traversal attack due to a vulnerability present in the code of the web application. In addition to this, poorly patched or configured webserver software can make the webserver itself vulnerable to a directory traversal attack.

The design of webservers limits public access to some extent. Directory traversal is exploitation of HTTP through which attackers are able to access restricted directories and execute commands outside of the webserver root directory by manipulating a URL. Attackers can use the trial-and-error method to navigate outside of the root directory and access sensitive information in the system.

An attacker exploits the software (webserver program) on a webserver to carry out directory traversal attacks. The attacker usually performs directory traversal attacks with the help of a browser. A webserver is vulnerable to a directory traversal attack if it accepts input data from a browser without proper validation.



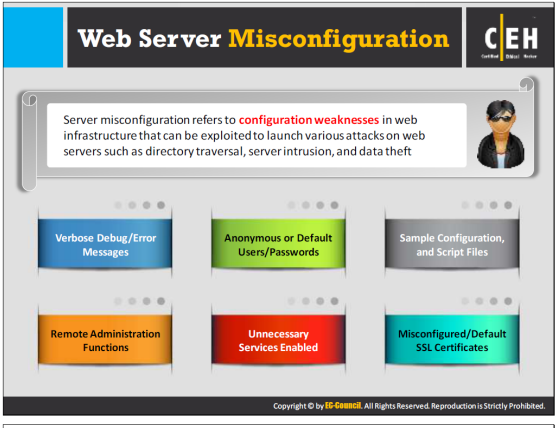
In an MITM attack or sniffing attack, an intruder intercepts or modifies the messages exchanged between the user and webserver through eavesdropping or intruding into a connection. This allows an attacker to steal sensitive user information such as online banking details, user names, passwords, etc., transferred over the Internet to the webserver. The attacker lures the victim to connect to the webserver by pretending to be a proxy. If the victim believes and agrees to the attacker's request, then all the communication between the user and the webserver passes through the attacker. In this way, the attacker can steal sensitive user information.

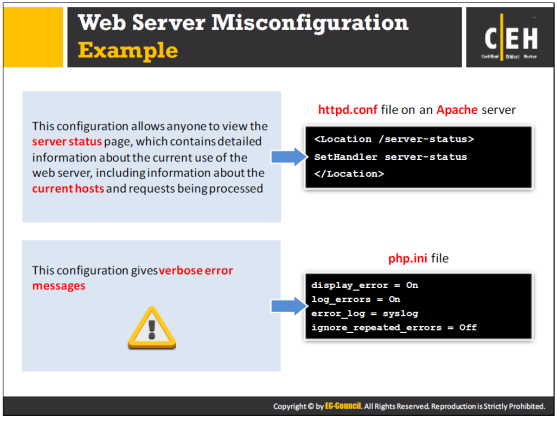


Attackers perform a phishing attack by sending an email containing a malicious link, and urging the user to click it. Clicking the link will redirect the user to a fake website that looks similar to the legitimate website. The attackers create such websites using their address hosted on webservers. A victim clicks on the malicious link believing the link is a legitimate website address, but it redirects to the malicious website hosted on the attacker's server. The website prompts the user to enter sensitive information such as username, passwords, financial account information, social security numbers, etc., and divulges the data to the attacker. Later, the attacker may be able to establish a session with the legitimate website with the victim's stolen credentials in order to perform a malicious operation on the target legitimate website.

Website defacement refers to unauthorized changes made to the content of a single webpage or an entire site, resulting in a change to the visual appearance of the site or a webpage. Hackers break into webservers and alter the hosted website by injecting code in order to add images, popups, or text to a page in such a way that the visual appearance of the page changes. In some cases, the attacker may replace the entire website instead of just changing single pages.

Besides changing the visual appearance of the target website, attackers deface websites for the purpose of infecting the computers of visitors by making the website vulnerable to virus attacks. Thus, website defacement not only embarrasses the target organization by changing the appearance of its website, but is also intended to harm its visitors.

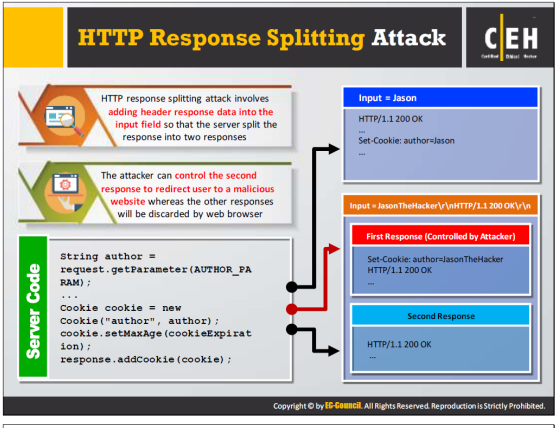


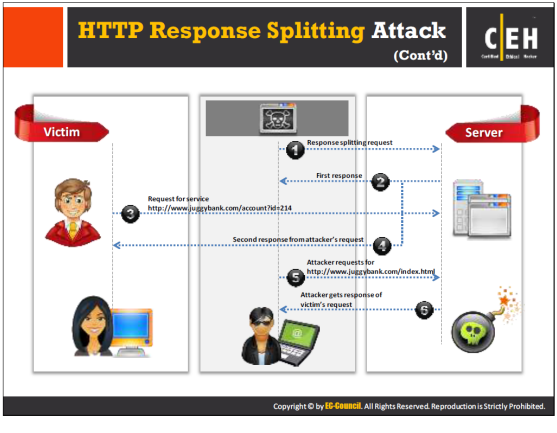


Web Server Misconfiguration

"**Keeping the server configuration secure requires vigilance**"- OWASP

Administrators who configure webservers improperly may leave serious loopholes in the webserver that may give an attacker the chance to exploit the misconfigured webserver to compromise its security and obtain sensitive information. The vulnerabilities of improperly configured webservers may be related to configuration, applications, files, scripts, or web pages. An attacker looks for such vulnerable webservers to launch attacks. The misconfiguration of a webserver gives the attacker a path to enter into the target network of an organization. These loopholes in the server can also help an attacker to bypass user authentication. Once detected, these problems can be easily exploited and result in the total compromise of a website hosted on the target webserver.



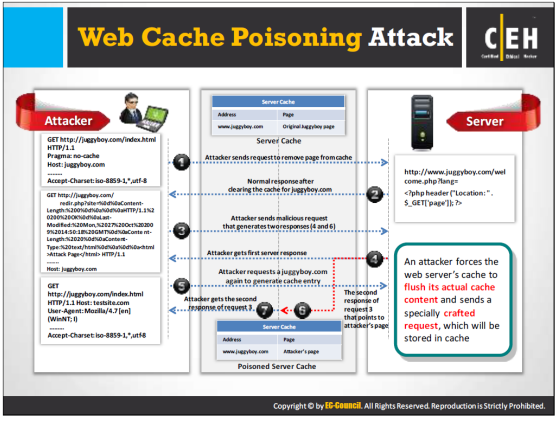


HTTP Response-Splitting Attack

An HTTP response attack is a web-based attack in which the attacker tricks the server by injecting new lines into response headers, along with arbitrary code. This type of attack exploits vulnerabilities in input validation. Cross-Site Scripting (XSS), Cross Site Request Forgery (CSRF), and SQL Injection are some of the examples for this type of attack. In this attack, the attacker controls the input parameter and cleverly constructs a request header that causes two responses from the server. The attacker alters a single request to appear as two requests by adding header response data into the input field. The webserver in turn responds to each request. The attacker can pass malicious data to a vulnerable application, and the application includes the data in an HTTP response header. The attacker can control the first response to redirect the user to a malicious website, whereas the web browser will discard other responses.

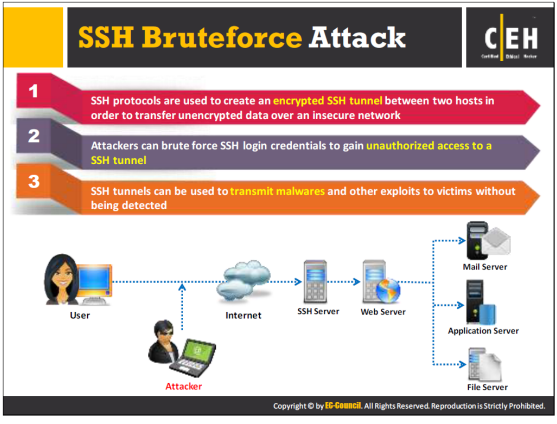
HTTP Response-Splitting Attack Example

In this attack, the attacker sends a response-splitting request to the webserver. The server splits the response into two and sends the first response to the attacker and the second response to the victim. After receiving the response from webserver, the victim requests service by providing credentials. At the same time, the attacker requests the index page. Then the webserver sends the response to the victim's request to the attacker and the victim remains uninformed.

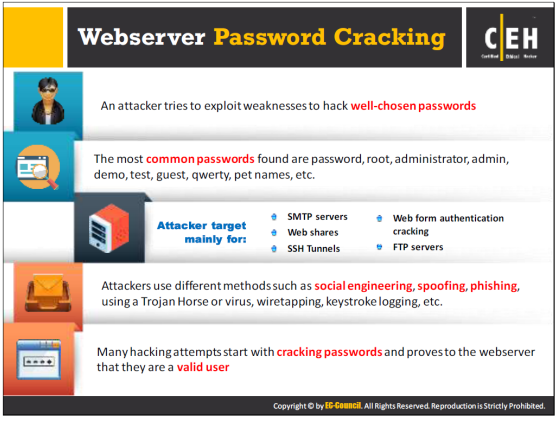


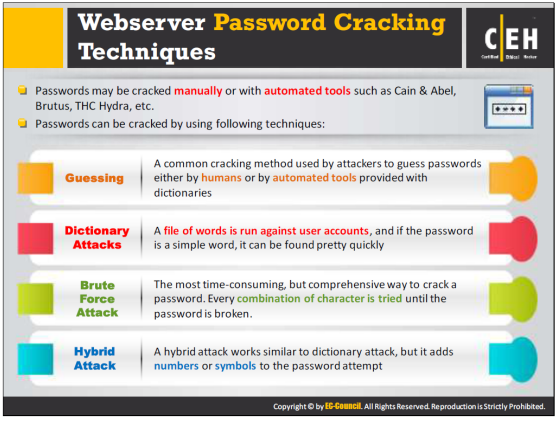
Web cache poisoning attacks the reliability of an intermediate web cache source. In this attack, the attackers swap cached content for a random URL with infected content. Users of the web cache source can unknowingly use the poisoned content instead of true and secured content when requesting the required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

An attacker forces the webserver's cache to flush its actual cache content and sends a specially crafted request to store in cache. In this case, all the users of that webserver cache will get malicious content until the servers flush the web cache. Web cache poisoning attacks are possible if the webserver and application has HTTP Response-Splitting flaws.



Attackers use the SSH protocols to create an encrypted SSH tunnel between two hosts in order to transfer unencrypted data over an insecure network. Usually SSH runs on TCP port 22. In order to conduct an attack on SSH, the attacker scans the entire SSH server using bots (performs TCP port 22 port scan) to identify possible vulnerabilities. With the help of a brute force attack, the attacker gains the login credentials. An attacker who gains the login credentials of SSH can use the same SSH tunnels to transmit malware and other means of exploitation to victims. Attackers use tools such as Nmap and ncrack on a Linux platform to carry out an SSH brute force attack.





Webserver Password Cracking Techniques

Cracking a password is the most common way of gaining unauthorized access to the webserver by exploiting its flawed and weak authentication mechanism. Once the password is cracked, an attacker can use those passwords to launch further attacks.

Attackers can use the following password cracking techniques to extract passwords from webservers, FTP servers, SMTP servers, etc. Let us get into the detail of various password cracking techniques and tools used by the attacker to crack passwords. Attackers can crack passwords either manually or with automated tools such as Cain & Abel, Brutus, THC Hydra, etc.

Attackers use various techniques to crack passwords:

- **Guessing:** This is most common method of cracking passwords, in which the attacker guesses possible passwords either manually or by using automated tools provided with dictionaries. Most people tend to use their pets' names, loved ones' names, license plate numbers, dates of birth, or other weak pass words such as "QWERTY," "password," "admin," etc. so that they can remember them easily. The attacker exploits this human behavior of keeping things simple to crack passwords.
- **Dictionary Attack:** A dictionary attack has predefined words of various combinations, and an automated program tries entering these words one at a time to see if any of them are the password. But this might not be effective if the password includes special

characters and symbols. Compared to a brute force attack, a dictionary attack is less time-consuming.

- **Brute Force Attack:** In the brute force method, all possible characters are tested, for example, uppercase from A to Z, numbers from 0 to 9, or lowercase a to z. This type of method is useful to identify one-word or two-word passwords. If a password consists of uppercase and lowercase letters and special characters, it might take months or years to crack the password using a brute force attack.
- **Hybrid Attack:** A hybrid attack is more powerful as it uses both a dictionary attack and brute force attack. It also uses symbols and numbers. Password cracking becomes easier with this method.

Even if webservers are configured securely or secured using network security measures such as firewalls, a poorly coded web application deployed on the webserver may give a path to an attacker to compromise the webserver's security. If the web developers do not adopt secure coding practices while developing web applications, it may give attackers the chance to exploit vulnerabilities and compromise web applications and webserver security. An attacker can perform different types of attacks on vulnerable web applications to breach webserver security.

- **Directory Traversal**

  Directory traversal is the exploitation of HTTP through which attackers are able to access restricted directories and execute commands outside of the webserver root directory by manipulating a URL.

- **Parameter/Form Tampering**

  In this type of tampering attack, the attacker manipulates the parameters exchanged between client and server in order to modify application data, such as user credentials and permissions, price and quantity of products, etc.

- **Cookie Tampering**

  Cookie tampering attacks occur when sending a cookie from the client side to the server. Different types of tools help in modifying persistent and non-persistent cookies.

- **Command Injection Attacks**

  Command injection is an attack method in which a hacker alters the content of the web page by using html code and by identifying the form fields that lack valid constraints.

- **Buffer Overflow Attacks**

  The design of most web applications helps them in sustaining some amount of data. If that amount exceeds the storage space available, the application may crash or may exhibit some other vulnerable behavior. The attacker uses this advantage and floods the application with too much data, which in turn causes a buffer overflow attack.

- **Cross-Site Scripting (XSS) Attacks**

  Cross-site scripting is a method where an attacker injects HTML tags or scripts into a target website.

- **Denial-of-Service (DoS) Attack**

  A DoS attack is intended to terminate the operations of a website or a server and make it unavailable for access by intended users.

- **Unvalidated Input and File injection Attacks**

  Unvalidated input and file injection attacks are carried out by supplying an unvalidated input or by injecting files into a web application.

- **Cross-Site Request Forgery (CSRF) Attack**

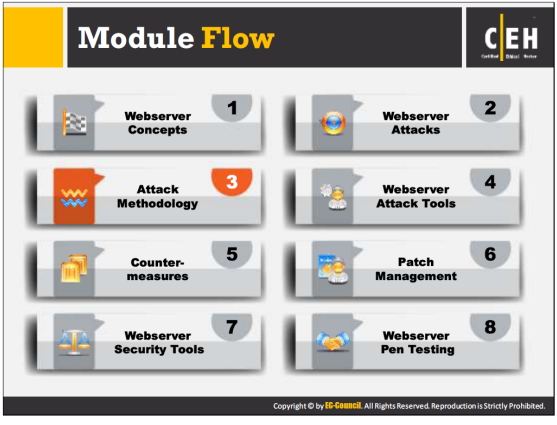
  An attacker exploits the trust of an authenticated user to pass malicious code or commands to the webserver.

- **SQL Injection Attacks**

  SQL injection t exploits the security vulnerability of a database for attacks. The attacker injects malicious code into the strings, later passed on to the SQL Server for execution.

- **Session Hijacking**

  Session hijacking is an attack in which the attacker exploits, steals, predicts, and negotiates the real valid web session control mechanism to access the authenticated parts of a web application.



The previous section described attacks that an attacker can carry out to compromise webserver security. This section explains exactly how the attacker moves forward in carrying out a successful attack on a webserver. A webserver attack typically involves preplanned activities called an attack methodology that an attacker follows to reach the goal of breaching the target webserver's security.

Attackers hack a webserver in multiple stages. At each stage, the attacker tries to gather more information about loopholes and tries to gain unauthorized access to the webserver. The stages of webserver attack methodology include:

- **Information Gathering**

  Every attacker tries to collect as much information as possible about the target webserver. The attacker gathers the information and then analyzes the information in order to find lapses in the current security mechanism of the webserver.

- **Webserver Footprinting**

  The purpose of footprinting is to gather more information about security aspects of a webserver with the help of tools or footprinting techniques. The main purpose is to know about the webserver's remote access capabilities, its ports and services, and other aspects of its security.

- **Mirroring Website**

  Website mirroring is a method of copying a website and its content onto another server for offline browsing. With a mirrored website, an attacker can view the detailed structure of the website.

- **Vulnerability Scanning**

  Vulnerability scanning finds vulnerabilities and misconfigurations of a webserver. Attackers scan for vulnerabilities with the help of automated tools known as vulnerability scanners.

- **Session Hijacking**

  Attackers can perform session hijacking after identifying the current session of the client. The attacker takes over complete control of the user session by means of session hijacking.

- **Hacking Webserver Passwords**

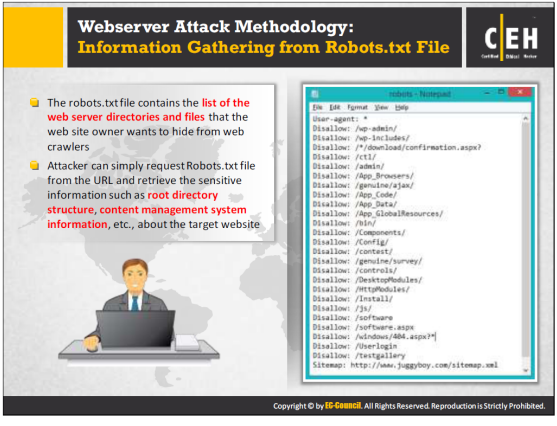
  Attackers use password cracking methods like brute force attacks, hybrid attacks, dictionary attacks, etc., to crack webserver passwords.

Information gathering is the first and one of the important steps towards hacking a target webserver. An attacker collects as much information as possible about the target server by using various tools and techniques. The information obtained from this step helps the attacker in assessing the security posture of the webserver. Attackers may search the Internet, newsgroups, bulletin boards, etc. for information about the target organization. Some of the following tools help the attacker to extract information such as the target's domain name, IP address, autonomous system number etc.

Whois

Source: *http://www.whois.net*

Whois performs a domain whois search and a whois IP lookup and searches the whois database for relevant information on domain registration and availability. This can provide insight into a domain's history and additional information. It performs a search to see who owns a domain name, how many pages from a site the owner has listed with Google, and can even search the Whois address listings for a website's owner, etc.



A website owner creates a robots.txt file to list for a web crawler those files or directories it should index in search results. Poorly written robots.txt files can cause complete indexing of website files and directories. In this case, an attacker may easily get information such as passwords, email addresses, hidden links, and membership areas if there are indexed confidential files and directories in the search results.

If the owner of the target website writes the robots.txt file and does not allow indexing of restricted pages in the search results, an attacker can still easily view the robots.txt file of that site to discover restricted files, and then view them to gather information.

An attacker types URL/robots.txt in the address bar of a browser to view the target website's robots.txt file. An attacker can also download the robots.txt file of a target website using the Wget tool.

Webserver Attack Methodology: Webserver Footprinting

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

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

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

http://toolbar.netcraft.com

Webserver footprinting tools ID Serve and httprecon can extract information from the target server. Let us look at the features and the type information these tools are able to collect from the target server.

httprecon

Source: *http://www.computec.ch*

httprecon is a tool for advanced webserver fingerprinting. This tool performs banner-grabbing attacks, status code enumeration and header ordering analysis on the target webserver. This tool provides accurate webserver fingerprinting information.

httprecon performs the following header analysis test cases on the target webserver:

- legitimate GET request for an existing resource
- very long GET request (>1024 bytes in URI)
- common GET request for a non-existing resource
- common HEAD request for an existing resource
- allowed method enumeration with OPTIONS
- usually not permitted http method DELETE
- not defined http method TEST
- non-existing protocol version HTTP/9.8
- GET request including attack patterns (e.g., : ../ and %%)

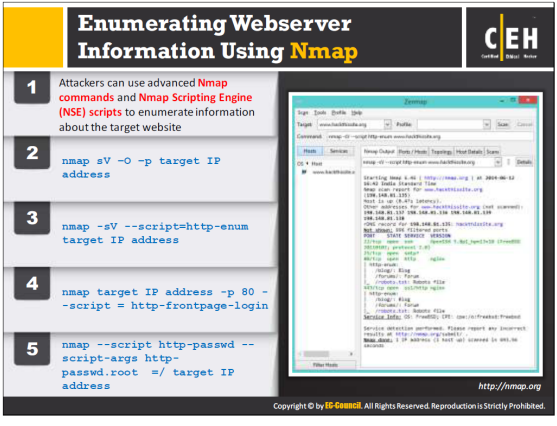
ID Serve

Source: http://www.grc.com

ID Serve is a simple Internet server identification utility. Its capabilities include:

ID Serve's capabilities include:

- **HTTP Server Identification:** ID Serve can identify the make, model, and version of a website's server software. ID Serve sends this information in the preamble of replies to web queries, but the information is not visible to the user.
- **Non-HTTP Server Identification:** Most non-HTTP (non-web) Internet servers (like FTP, SMTP, POP, NEWS, etc.) are required to transmit a line containing a numeric status code and a human-readable greeting to any connecting client. Therefore, ID Serve can also connect with non-webservers to receive and report that server's greeting message. This generally reveals the server's make, model, version, and other potentially useful information.
- **Reverse DNS Lookup:** When ID Serve users enter a site's or server's domain name or URL, the application will use DNS to determine the IP address for that domain. However, sometimes it is useful to go in the other direction to determine the domain name associated with a known IP address. This process, known as reverse DNS lookup, is also built into ID Serve. ID Serve will attempt to determine the associated domain name or any entered IP address.



Nmap along with Nmap Scripting Engine can extract lot of valuable information from the target webserver. In addition to Nmap commands, Nmap Scripting Engine (NSE) provides scripts that reveal to an attacker all sorts of useful information from the target webserver.

An attacker uses the following Nmap commands and NSE scripts to extract information.

- Discover virtual domains with hostmap

  ```
  $nmap --script hostmap <host>
  ```

- Detect a vulnerable server that uses the TRACE method

  ```
  nmap --script http-trace -p80 localhost
  ```

- Harvest email accounts with http-google-email

  ```
  $nmap --script http-google-email <host>
  ```

- Enumerate users with http-userdir-enum

  ```
  nmap -p80 --script http-userdir -enum localhost
  ```

- Detect HTTP TRACE

  ```
  $nmap -p80 --script http-trace <host>
  ```

- Check if webserver is protected by a WAF/IPS

  ```
  $nmap -p80 --script http-waf-detect --script-args="http-waf-detect.uri=/testphp.vulnweb.com/artists.php,http-waf-detect.detectBodyChanges" www.modsecurity.org
  ```

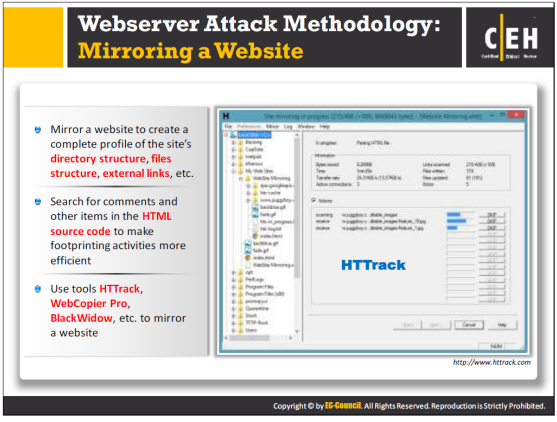
- Enumerate common web applications

```
$nmap --script http-enum -p80 <host>
```

- Obtain robots.txt

```
$nmap -p80 --script http-robots.txt <host>
```

Source: *http://nmap.org*

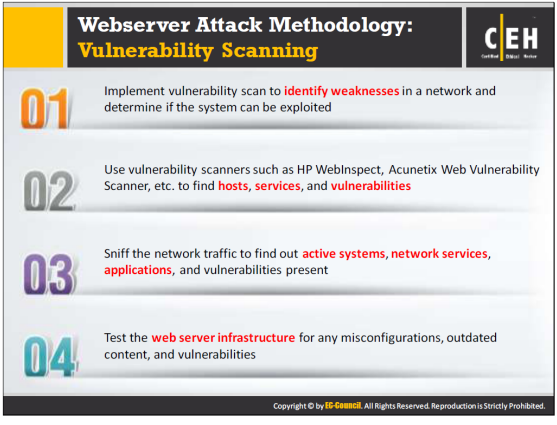


Website mirroring copies an entire website and its content onto the local drive. The mirrored web site reveals the complete profile of the site's directory structure, file structure, external links, images, webpages etc. With a mirrored target website, an attacker can easily trace out the website's directories and gain valuable information. An attacker who copies the website does not need to be online to go through the target website. The attacker can trace out the website at any time. The attacker can gain valuable information by searching the comments and other items in the HTML source code of downloaded webpages. There are many website mirroring tools available to copy a target website onto a local drive, such as HTTrack, Webripper 2.0, WinWSD, Webcopier, Blackwidow, etc.

HTTrack

Source: *http://www.httrack.com*

HTTrack is an offline browser utility. It downloads a Website from the Internet to a local directory, building recursively all directories, getting HTML, images, and other files from the server. HTTrack arranges the original site's relative link-structure. Simply open a page of the "mirrored" website in a browser, browse the site from link to link, as if viewing it online.



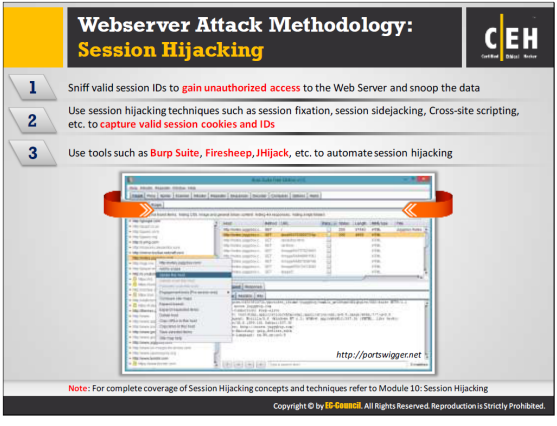
Vulnerability scanning determines vulnerabilities and misconfigurations of a target webserver or network. Vulnerability scanning finds possible weaknesses in a target server to exploit in a webserver attack. An attacker uses various automated tools to perform vulnerability scanning on a target server.

Attackers use sniffing techniques to obtain data about network traffic to find out active systems, network services and applications in the vulnerability scanning phase. Some of the tools that attacker can use for vulnerability scanning include Acunetix Web Vulnerability Scanner, HP WebInspect, Nessus, and Paros proxy.

Acunetix Web Vulnerability Scanner

Source: http://www.acunetix.com

Acunetix Web Vulnerability Scanner automatically looks for known security vulnerabilities in web applications, including SQL Injection, XSS, arbitrary file creation/deletion, and weak password strength on authentication pages. It creates professional security audit and compliance reports.

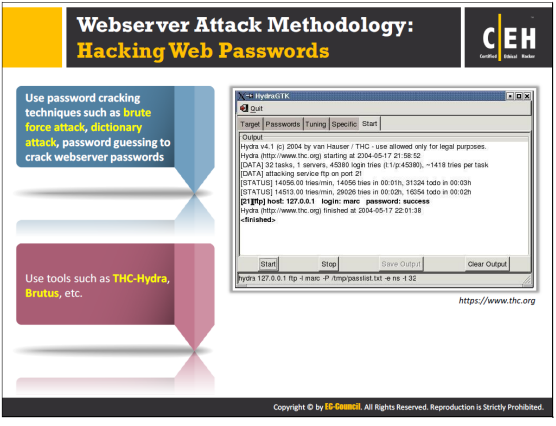


There are many techniques through which an attacker can hijack or steal valid session content such as session token prediction, session replay, session fixation, sidejacking, cross-site scripting, etc. Using these techniques, an attacker tries to capture valid capture valid session cookies and IDs in established sessions.

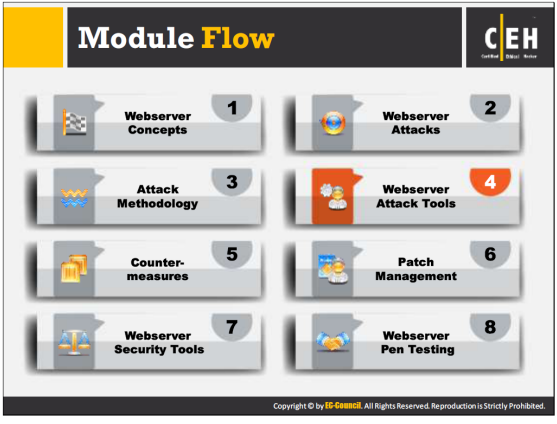
Burp Suite

Source: *http://portswigger.net*

Burp Suite is web security testing tool that can hijack the session identifiers in established sessions. The Sequencer tool in Burp Suite tests the randomness of session tokens. With this tool, an attacker can predict the next possible session ID token, and use that to take over a valid session.



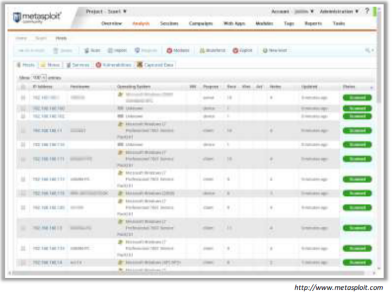
In this phase of webserver hacking, an attacker tries to crack webserver passwords. An attacker tries all possible techniques of password cracking to extract passwords, including password guessing, dictionary attacks, brute force attacks, hybrid attacks, pre-computed hashes, rule-based attacks, distributed network attacks, rainbow attacks, etc. An attacker needs patience, as some of these techniques are tedious and time-consuming. An attacker can also use automated tools such as Brutus, THC-Hydra, etc. to crack web passwords.



Now you are familiar with the methodology that an attacker uses to hack a webserver. This section will introduce webserver hacking tools that an attacker may use in the webserver hacking methodology described in the previous section. These tools extract critical information during the hacking methodology.

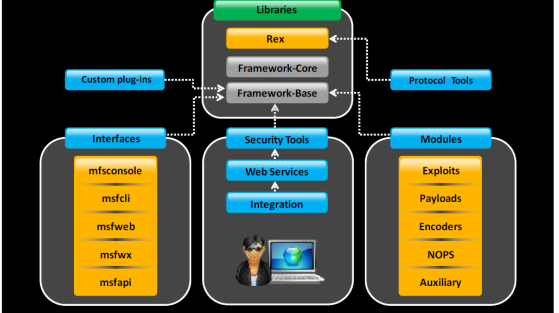
Webserver Attack Tool: Metasploit

- The Metasploit Framework is a **penetration testing toolkit**, exploit development platform, and **research tool** that includes hundreds of working remote exploits for a variety of platforms
- It supports fully automated **exploitation of web servers**, by abusing known vulnerabilities and leveraging weak passwords via Telnet, SSH, HTTP, and SNM



http://www.metasploit.com

Metasploit Architecture



Webserver Attack Tool: Metasploit

Features of Metasploit that an attacker may use to perform webserver attack include:

- Closed-loop Vulnerability Validation
- Phishing Simulations
- Social Engineering
- Manual Brute Forcing
- Manual Exploitation
- Evade leading defensive solutions

Metasploit enables pen testers to:

- Complete pen test assignments faster by automating repetitive tasks and leveraging multi-level attacks
- Assess the security of web applications, network and endpoint systems, as well as email users
- Tunnel any traffic through compromised targets to pivot deeper into the network
- Customize the content and template of executive, audit, and technical reports

Metasploit Architecture

The Metasploit framework is an open-source exploitation framework that provides security researchers and pen testers with a uniform model for rapid development of exploits, payloads, encoders, NOP generators, and reconnaissance tools. The framework reuses large chunks of code that a user would have to otherwise copy or re-implement on a per-exploit basis. The framework is modular in architecture and encourages the reuse of code across various projects. The framework itself is broken down into a few different pieces, the most low-level being the framework core. The framework core is responsible for implementing all of the required interfaces that allow for interacting with exploit modules, sessions, and plugins. It supports vulnerability research, exploit development, and the creation of custom security tools.

Source: http://www.metasploit.com

Metasploit Exploit Module

- It is the basic module in Metasploit used to **encapsulate an exploit** using 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 Module

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

Command Prompt

```
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) >
```

Metasploit Payload Module

An exploit carries the payload in its backpack when it breaks into the system and then leaves the backpack there.

There are three types of payload modules provided by the Metasploit:

- **Singles**: It is self-contained and completely standalone
- **Stagers**: It sets up a network connection between the attacker and victim
- **Stages**: It is downloaded by stagers modules

Metasploit Payload Module can upload and download files from the system, take screenshots, and collect password hashes. It can even take over the screen, mouse, and keyboard to control a remote computer.

Metasploit Auxiliary Module

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

Command Prompt

```
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...
```

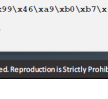
Metasploit NOPS Module

- NOP modules generate a no-operation instructions 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\x
66\x9f\xb8\x2d\xb6"
"\x24\xbe\xb1\x3f\x43\x1d\x93\xb2\x37\x35\x
84\xd5\x14\x40\xb4"
"\xb3\x41\xb9\x48\x04\x99\x46\xa9\xb0\xb7\x
2f\xfd\x96\x4a\x98"
"\x92\xb5\xd4\x4f\x91";
msf nop(opty2) >
```

Metasploit Auxiliary Module

The Auxiliary Module of Metasploit includes tools and modules that assess the security of the target, auxiliary modules such as scanners, denial of service modules, fuzzers, etc. To list all the available auxiliary modules in Metasploit, use `show auxiliary` command in Metasploit. All the other modules in Metasploit are auxiliary modules except modules used to exploit. The tool uses the auxiliary modules as an extension for a variety of purposes other than exploitation. Auxiliary modules reside in the modules/auxiliary/ directory of the framework main directory.

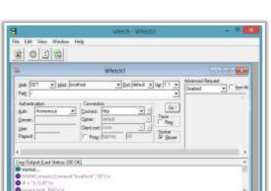
The basic definition of an auxiliary module is:

```
require 'msf/core'
p "My Auxiliary Module"
class Metasploit3 < Msf::Auxiliary
end         # for the class definition
```

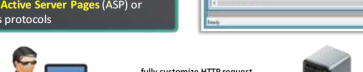
Webserver Attack Tool: Wfetch

WFetch allows 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 attacker to test the performance of Web sites that contain new elements such as **Active Server Pages** (ASP) or wireless protocols



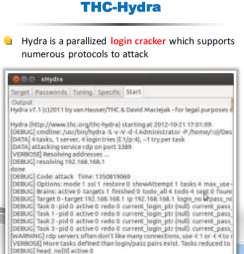
http://www.microsoft.com



Web Password Cracking Tools: THC-Hydra and Brutus

THC-Hydra

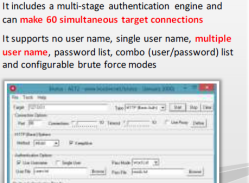
- Hydra is a parallized login cracker which supports numerous protocols to attack



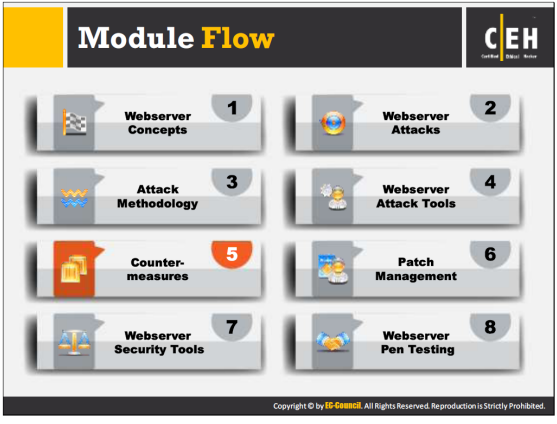
http://www.thc.org

Brutus

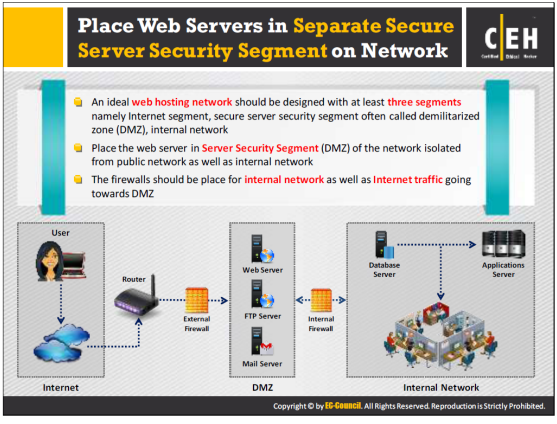
- It includes a multi-stage authentication engine and can make 60 simultaneous target connections
- It supports no user name, single user name, multiple user name, password list, combo (user/password) list and configurable brute force modes



http://www.hoobie.net



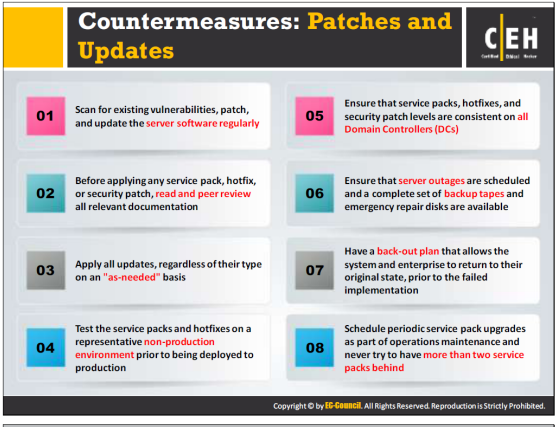
We have discussed the benefits of a well-versed webserver security posture, the danger posed by webserver attacks, the methodology used in webserver attacks, and the tools that assist an attacker to conduct webserver attacks. Now we will discuss tools and techniques used in securing webservers. This section discusses various webserver attack detection methods, countermeasures, and defense techniques.

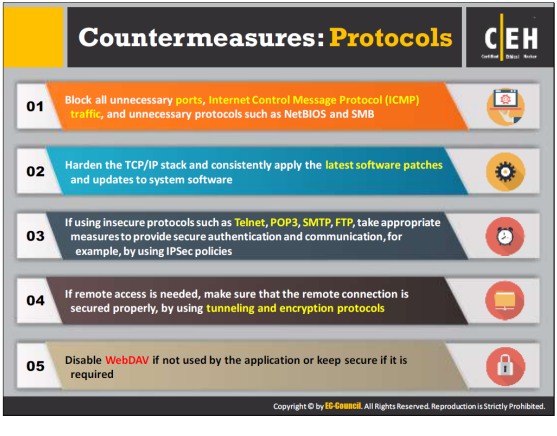


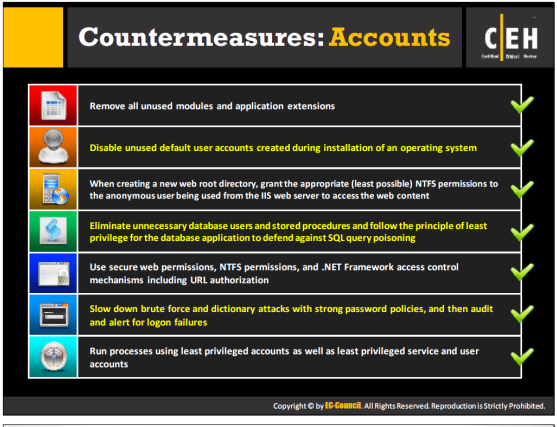
The first step in securing webservers is to place them in a separate segment in the web hosting network called the De-Militarized Zone (DMZ). Placing webservers in a separate segment adds security barriers to the webserver from the internal network as well as the outside public network. This separation allows the administrator to place firewalls and apply access control based on security rules for the internal network as well as Internet traffic going towards the DMZ. A web hosting network such as this can prevent attacks to the webserver from an outside attacker as well as from malicious insiders

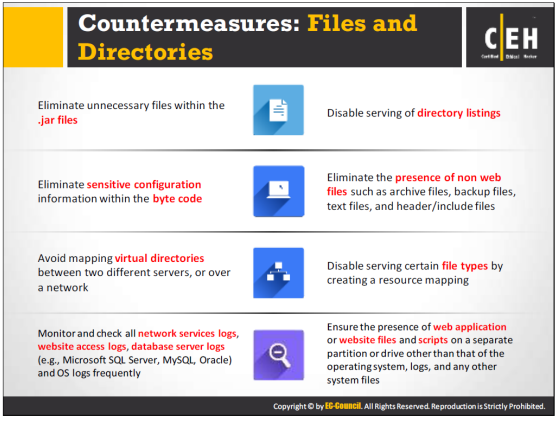
Network segmentation divides a network into different segments, each is having its own hub or switch. Network segmentation allows network administrators to protect one segment from others by imposing firewalls and security rules depending on the level of security desired. In a segmentation network, an attacker who compromises even one segment of the network will not able to compromise the security of other segments of the network.

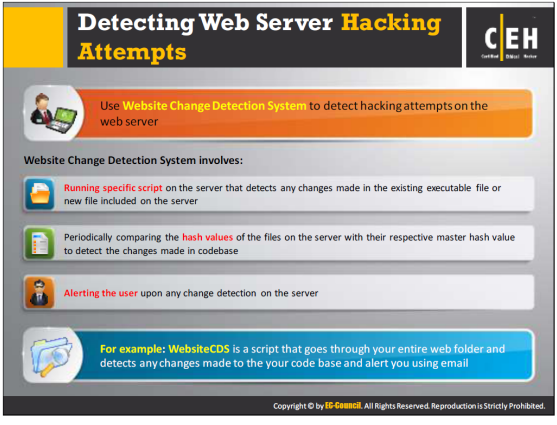
Let us take look at a sample web hosting network. Here the administrator segments the network and places the webserver in a separate server security segment known as the DMZ.





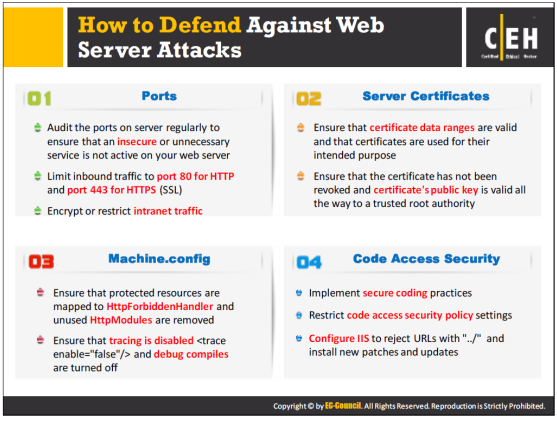






An attacker who gains access to the webserver by compromising security through known vulnerabilities present in the webserver may try to plant backdoors (scripts). These allow the attacker to gain access, launch phishing attacks, or send spam emails. The victim will be unaware of the webserver attack until the server is blacklisted on spam mails, or if the attacker redirects the visitors of the target site hosted on the webserver to some other site. Thus, a webserver attack is hard to detect unless such malicious things happen to the victim, but by that time, it will be too late to react, the as attacker would have already succeeded. There should be a mechanism that detects a webserver hacking attempt in its early stages, to prevent the harm that an attacker could do to the webserver.

When an attacker installs a backdoor on the webserver, the size of files infected with the backdoor will increase automatically. Website Change Detection System (WDS) is a script that runs on the server to detect changes made to any executable file, or the presence of any new file on the webserver such as HTML, JS, PHP, ASP, Perl, Python files, etc. It works by periodically comparing the hash values of the files on the server with their respective master hash value to detect any changes made in codebase. If it detects any change on the server, and alerts the user to take necessary action. Thus, WDS helps in detecting webserver hacking attempts in the early stages of an attack.



Defenses against webserver attacks include:

Ports

Monitor all ports on the webserver regularly to prevent unnecessary traffic going towards the target webserver. If traffic is not monitored, the target webserver will leave the door open for malware attacks. Do not allow public access to port 80 for HTTP and to port 443 for HTTPS; traffic for these ports should be limited. If port 80 is kept open, it may lead to DoS attacks by consuming server resources. Intranet traffic should either be encrypted or restricted to secure the webservers.

Attackers try to hide their identity by spoofing the IP address of a legitimate user. By processing the security log file, either by using the "deny this IP address" rule in the firewall rule set file or by creating a "routed blackhole" command, the target system can defend against webserver attacks.

Server Certificates

Server certificates guarantee security by providing certificates signed from a trusted authority. However, an attacker may compromise certified servers using forged certificates in order to intercept the secure communication by performing MITM attacks. There are various ways to avoid such MITM attacks. Some of the techniques to avoid such attacks include:

- Using direct validation of certificates
- Using a novel protocol that does not depend on third parties for certificate validation

- Allowing domains to directly and securely examine their certificates by using previously established user authentication credentials
- Using robust cryptographic construction that enhances server identity validation and also resolves limitations of third party solutions
- Ensuring that the certificate data ranges are valid and that certificates are used for their intended purpose
- Ensuring that the certificate has not been revoked and certificate's public key is valid all the way to a trusted root authority

Machine.config

Machine.config is the mechanism of securing information by changing the machine level settings. This affect applies to all other applications. Machine.config file includes machine settings for the .Net framework that affects the security.

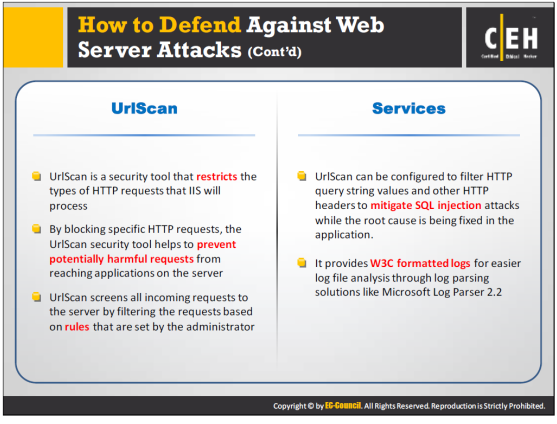
Machine.config file can:

- 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
- Validate that ASP.NET errors are not reverted back to the client
- Verify session state settings

Code Access Security

- Implement secure coding practices to avoid source code disclosure and input validation attack.
- Restrict code access security policy settings to ensure that code downloaded from the Internet or intranet has no permissions to execute.
- Configure IIS to reject URLs with "../" to prevent path traversal, lock down system commands and utilities with restrictive access control lists (ACLs), and install new patches and updates.

If targets do not implement code access security in their webservers, then there is a possibility for execution of malicious code.



UrlScan is a security tool that restricts the types of HTTP requests that Microsoft Internet Information Services (IIS) will process. By blocking specific HTTP requests, the UrlScan security tool helps prevent potentially harmful requests from reaching the server. UrlScan is implemented as an ISAPI filter that screens and analyzes HTTP requests as IIS receives them. When properly configured, UrlScan is effective at reducing the exposure of IIS to potential Internet attacks.

Administrators may configure UrlScan to reject HTTP requests based on the following criteria:

- The HTTP request method or verb
- The file name extension of the requested resource
- Suspicious URL encoding
- Presence of non-ASCII characters in the URL
- Presence of the specified character sequences in the URL
- Presence of specified headers in the request

UrlScan Features

- Create "deny" rules independently to the query string, all headers, or a particular header.
- A global DenyQueryString section in configuration lets you add deny rules for query strings with the option of checking the un-escaped version of the query string.

- A global AlwaysAllowedUrls section in configuration lets you specify safe URLs that will bypass all URL based checks.
- A global AlwaysAllowedQueryStrings section in configuration lets you specify safe query strings that will bypass all query string checks.
- Escape sequences (e.g., %0A%0D) can be used in deny rules so it is possible to deny CRLF and other sequences involving non-printable characters.
- Multiple UrlScan instances can be installed as site filters, each with its own configuration and rules (UrlScan.ini).
- Configuration (UrlScan.ini) change notifications are propagated to IIS worker processes.
- Enhanced W3C formatted logging gives descriptive configuration errors in the Remarks header.

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 webserver

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

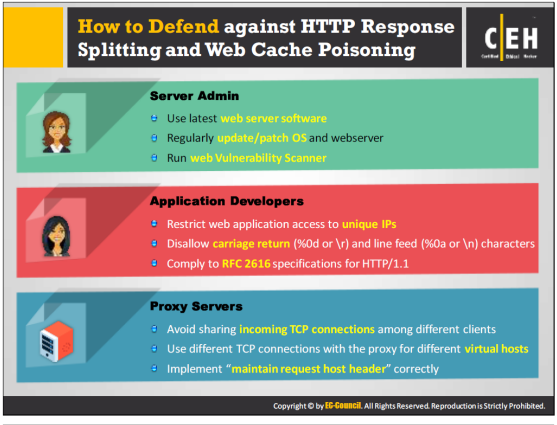
05 Relocate sites and virtual directories to non-system partitions and use IIS Web permissions to restrict access

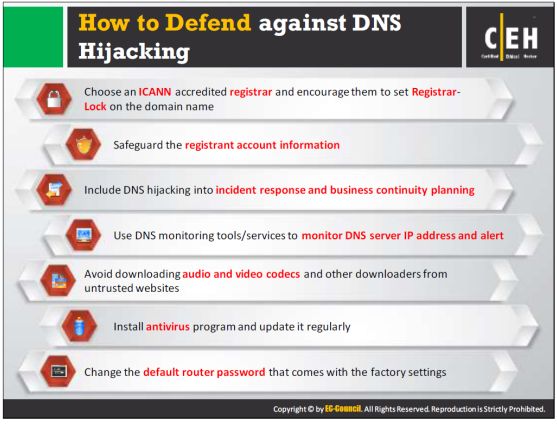
06 Remove all unnecessary IIS script mappings for optional file extensions to avoid exploiting any bugs in the ISAPI extensions that handle these types of files

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 webserver 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





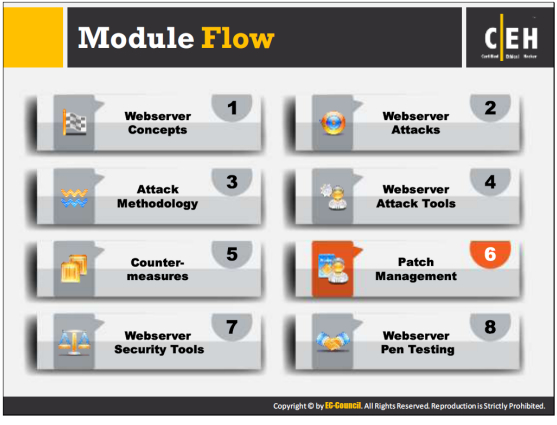
How to Defend against HTTP Response-Splitting and Web Cache Poisoning

While setting cookies, remove carriage returns (CRs) and linefeeds (LFs) before inserting data into any HTTP response header. It is best practice to use third party products to test for the existence of security holes and defend against CRLF injection. Ensure that data application engines are up to date.

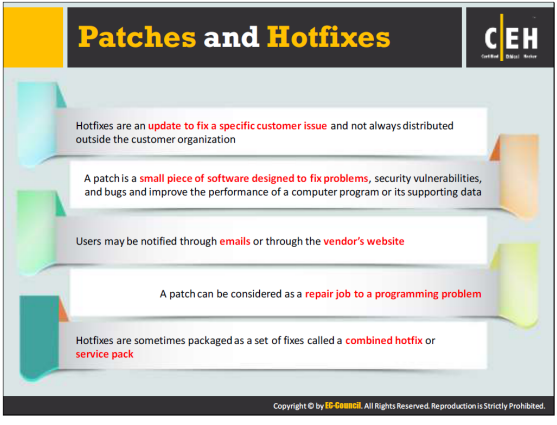
UDP source port randomization technique defends severs against blind response forgery. Limit the number of simultaneous recursive queries and increase the Times-to-Live (TTLs) of legitimate records.

Defenses against DNS Hijacking include:

- **Domain Name System Security Extensions (DNSSEC):** It adds an extra layer to DNS that prevents DNS from being hacked.
- **Strong Password Policies and User Management:** Use of strong passwords further enhances the security.
- **Better Service Level Agreements (SLAs) from DNS Service Providers**: When signing up to DNS servers with DNS service providers, learn who to contact when there is an issue, how to receive better quality of reception and support, and whether the DNS server's infrastructure is hardened against attack, etc.
- **Configuring a Master-Slave DNS within your Network:** Use a Master-Slave DNS and configure the master without internet access. Maintain two slave servers instead, so that even if someone hacks a slave, it will update only when it receives an update from the master.
- **Constant Monitoring of DNS Servers:** Constant monitoring of DNS server ensures that a website is returning the correct IP address.



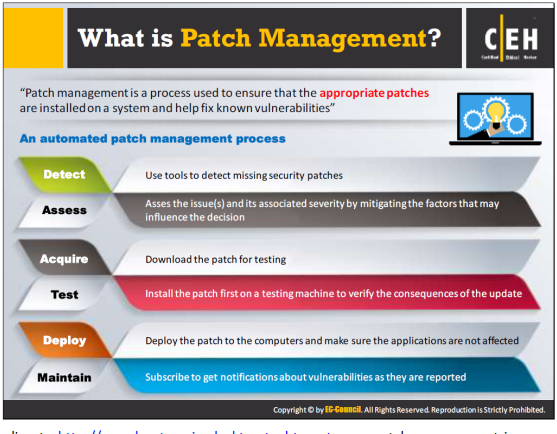
Developers always try to find bugs in the webserver and fix them. The bug fixes are available in the form of patches. These patches provide protection against known vulnerabilities. Unpatched or vulnerable patches can create a security hole in the webserver. This section describes the role of patches, upgrades, and hotfixes in securing webservers. This section also provides guidance in choosing proper patches, upgrades, hotfixes, and their appropriate sources for secure patch management.



A patch is a small piece of software designed to fix problems, security vulnerabilities, and bugs, and improve the usability or performance of a computer program or its supporting data. A patch can be considered as a repair job done to a programming problem. Software vulnerability is the weakness of a software program that makes it susceptible to malware attacks. Software vendors provide patches that prevent exploitations and reduce the probability of threats exploiting a specific vulnerability. Patches include fixes and updates for multiple issues, and are available for all customers. A patch is a publicly released update to fix a known bug or issue. Without patches, any system is much more vulnerable to attack. If an attacker can identify a vulnerability before it is fixed, then a system might be susceptible to malware attacks.

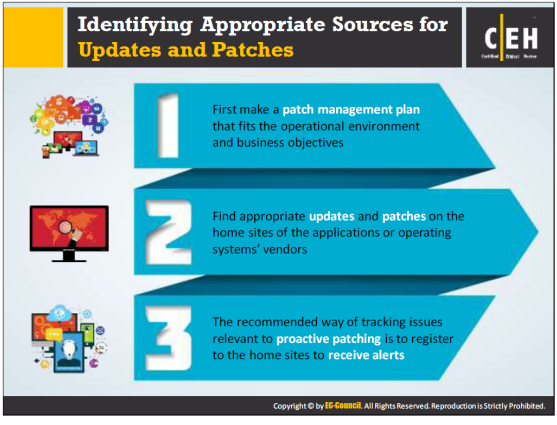
A hotfix is a package used to address a critical defect in a live environment, and contains a fix for a single issue. It updates a specific version of a product. Hotfixes provide solutions faster and ensure that the issues are resolved. Apply hotfixes to software patches on production systems.

Vendors update users about the latest hotfixes through email, or make them available on their official website. Hotfixes are an update to fix a specific customer issue and not always distributed outside the customer organization. Vendors occasionally deliver hotfixes as a set of fixes called a combined hotfix or service pack.



According to http://searchenterprisedesktop.techtarget.com, patch management is an area of systems management that involves acquiring, testing, and installing multiple patches (code changes) to an administered computer system. Patch management is a defense against vulnerabilities that cause security weakness, or corrupts data. It is a process of scanning for network vulnerabilities, detecting the missed security patches and hotfixes and then deploying the relevant patches as soon as they are available to secure the network. It involves the following:

- Choosing, verifying, testing, and applying patches
- Updating previously applied patches with current patches
- Listing patches applied previously to the current software
- Recording repositories, or depots, of patches for easy selection
- Assigning and deploying the applied patches



It is important to identify the appropriate source for updates and patches. Patches and updates that are not installed from trusted sources can render the target server even more vulnerable to attacks instead of hardening the security of the target server. Thus, the selection of appropriate sources for updates and patches plays a vital role in securing webservers.

Installation of a Patch

01 Users can access and install security patches via the World Wide Web

Patches can be installed in two ways

Manual Installation

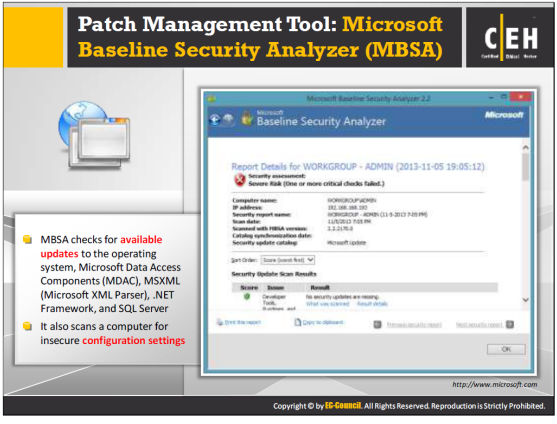
In this method, the user has to download the patch from the vendor and fix it

Automatic Installation

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

Implementation and Verification of a Security Patch or Upgrade

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



The Microsoft Baseline Security Analyzer (MBSA) identifies missing security updates and common security misconfigurations. Designed for the IT professional in small and medium-sized businesses, the tool helps them to determine their security state in accordance with Microsoft security recommendations, and offers specific remediation guidance. Improve your security management process by using MBSA to detect common security misconfigurations and missing security updates on computers. The MBSA can scan one or more computers by domain, IP address range, or other grouping. Once the scan is complete, the MBSA provides a detailed report and instructions on how to help turn a system into a more secure working environment. The MBSA will create and store individual XML security reports for each computer scanned.

Source: *http://www.microsoft.com*

In addition to the MBSA patch management tool, there are various patch management tools available to fix security vulnerabilities in a webserver. These tools include:

Altiris Client Management Suite

Source: *http://www.symantec.com*

This tool support multiple platforms, implements a single management framework for distributed, heterogeneous client environments, even for a system over the Internet. This tool creates advanced analytics and reporting to explore data, and includes network discovery and inventory tools.

GFI LanGuard

Source: *http://www.gfi.com*

GFI LanGuard's patch management scans a network automatically or on demand, allows auto-download of missing patches for operating systems such as Microsoft, Mac OS X, and Linux; and many third-party applications. It also automates patching of all major browsers running on Windows and offers a rollback feature to provide consistency.

Kaseya Security Patch Management

Source: *http://www.kaseya.com*

Kaseya Security Patch Management offers cloud-based architecture to identify and update patches for every system with date and systems. The application takes systems that are not updated offline and flags them for remediation. It monitors and maintains patch compliance

and uses filters to customize security. It allows automated patch development by scheduling a time for user, computer, or group, controlled by email notification.

ZENworks® Patch Management

Source: http://www.novell.com

Novell ZENworks Patch Management automates the process of discovering, retrieving, and deploying patches. It uses new Linux agent security to monitors all Windows, Red Hat Linux, and SUSE Linux operating systems and recommends patches, which are tested and verified prior to release. This tool offers centralized management of all devices in a single, unified web-based console that generate reports for security compliance on a device.

Security Manager Plus

Source: http://www.manageengine.com

Security Manager Plus supports Red Hat Linux, CetOS Linux, Debian Linux, and multi-language patching for Windows. Without any third-party involvement, it deploys missing patches from the vendor, which are required for all systems and networks. The deployment status comes with configurable timeout that can notify by email, and completion leads to automatic reboot or shutdown of a system.

Prism Suite

Source: http://www.newboundary.com

Prism Suite® is a fully integrated Windows configuration management solution featuring simple and automated software deployment, IT asset management and patch management. This tool allows installation, updates and patches in a single step with no end-user distraction. Smart update configuration automatically targets computers with the correct software, updates, and patches, and continuously monitors and rebuilds computer environments in real-time.

MaaS360® Patch Analyzer Tool

Source: http://www.maas360.com

MaaS360 Patch Analyzer will scan any Windows machine and report on all the installed and missing patches. This tool provides access to detailed information about product, title, severity, category, vendor URL, etc. Compare all the missing patches installed by this tool with reports generated by Windows Update Service and Windows Server Update Services (WSUS) Server.

Secunia CSI

Source: http://secunia.com

The Secunia Corporate Software Inspector (CSI) is an authenticated internal vulnerability scanner, capable of assessing the security of all programs that run on Windows PCs, and patches the vulnerabilities. This tool is compatible with any type of security software such as firewalls, IDS, or IPS, and runs on VMware. In each advisory presented within the Secunia CSI, there is always a link to the Common Vulnerabilities and Exposures (CVE) reference.

Lumension Patch and Remediation

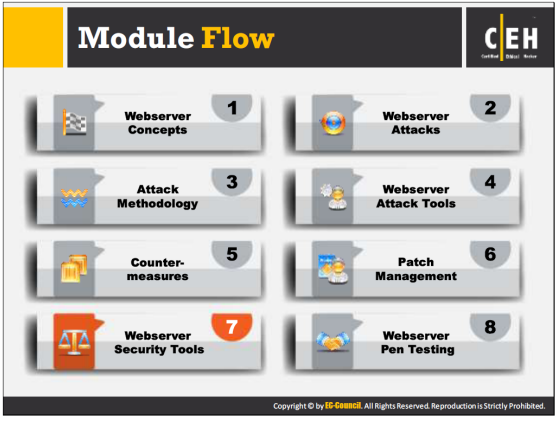
Source: http://www.lumension.com

Lumension Patch and Remediation is an endpoint management and security suite that provides patch content for Microsoft operating systems and applications; and offers a patch repository for Sun, Oracle, Linux, Adobe, Apple, etc. This tool includes configuration management, antivirus, application control, device control, disk encryption, mobile device management, and allows customization of patch deployments with advanced options such as set custom flags, control reboot or dismiss, etc. It automates scheduling of disk defragmentation tasks; and policy enforcement for account, device control, domain, network, and system policy security settings.

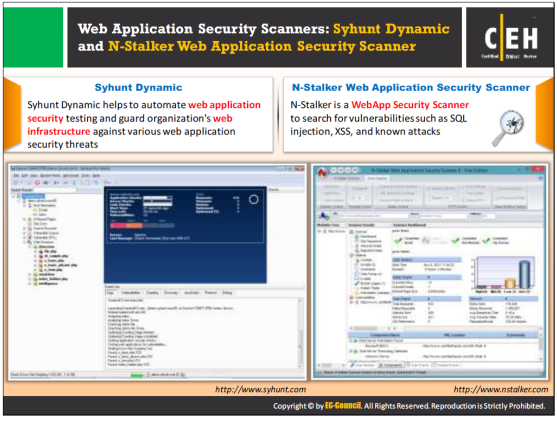
VMware vCenter Protect

Source: http://www.vmware.com

VMware vCenter Protect Essentials provides centralized Windows patch management and asset-inventory management for virtual and physical systems. This tool continuously scans and deploys all available security patches and enables users to manage physical and virtual machines, deploy software, discover assets, simplify configuration, control power usage remotely and ensure endpoint security.



This section describes common webserver security tools that keep a webserver secure from possible attacks. These tools scan for vulnerabilities in a target server and web applications, send alerts on hacking attempts, scan for malware in the webserver, and perform many more security assessment activities.



Syhunt Dynamic

Source: http://www.syhunt.com

Syhunt Dynamic automates web application security testing and guards the organization's web infrastructure against web application security threats. Syhunt Dynamic crawls web sites and detects cross-site scripting, directory transversal problems, fault Injection, SQL Injection, attempts to execute commands and multiple other attacks This tool works with UNIX, Linux etc. and devices such as routers, firewalls etc. Syhunt Dynamic creates signatures to detect application vulnerabilities and prevents logout. It analyzes JavaScript and logs suspicious responses, and tests errors for review.

N-Stalker Web Application Security Scanner

Source: http://www.nstalker.com

N-Stalker is a WebApp Security Scanner that searches for vulnerabilities such as Clickjacking, SQL injection, XSS, and known attacks. It allows spider crawling throughout the entire application and creation of Web macros for form authentication. It also provides proxy capabilities for "drive-thru" attacks and identifies components through reverse proxies that distribute different platforms in the same application URL. This tool checks for Web Signature attacks, Cookie Exposure, etc. and every known Web development platform is supported which interacts through the HTTP protocol.

Wikto

Source: http://www.sensepost.com

Wikto is a webserver security scanner used to perform webserver assessments such as associating data found on different search engines to social networking sites like MySpace.com and LinkedIn. It provides the ability to collect data found at one place to anywhere else.

Acunetix Web Vulnerability Scanner

Source: http://www.acunetix.com

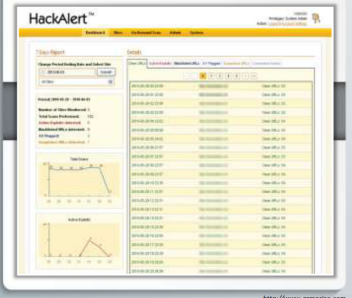
Acunetix Web Vulnerability Scanner scans web sites and detects vulnerabilities. Acunetix WVS checks web applications for SQL injections, cross-site scripting, etc. It includes advanced pen testing tools to ease manual security audit processes, and creates professional security audit and regulatory compliance reports. Based on AcuSensor Technology that detects more vulnerabilities and generates fewer false positives. It supports testing of web forms and password protected areas, pages with CAPTCHA, single sign-on and two factor authentication mechanisms. It detects application languages, webserver types and smartphone-optimized sites. Acunetix crawls and analyzes different types of websites including HTML5, SOAP and AJAX. It supports scanning of network services running on the server and port scanning of the webserver.

Web Server Malware Infection Monitoring Tool: HackAlert

HackAlert is a **cloud-based service** that identifies hidden zero-day malware and drive-by downloads in websites and online advertisements

Features

- Protects clients and customers from malware injected websites
- Identifies malware
- Displays injected code snippets
- Deploys as cloud-based SaaS
- Integrates with WAF or web server modules for instant mitigation

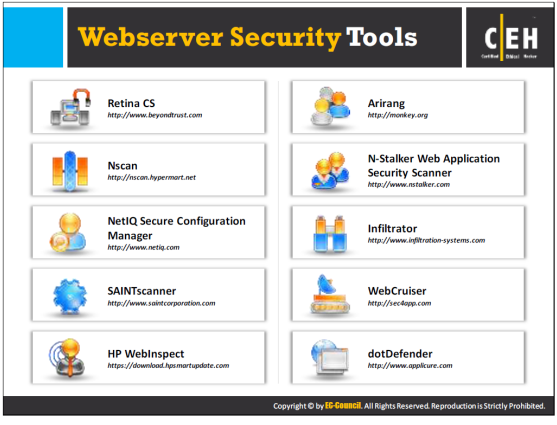


http://www.armorize.com

QualysGuard Malware Detection Service (MDS) Enterprise Edition allows organizations to proactively scan their web sites for malware, providing automated alerts and in-depth reporting to enable prompt identification and resolution. QualysGuard MDS enables organizations to protect their customers from malware infections and safeguard their brand reputations.

The QualysGuard MDS Enterprise Edition enables businesses to scan and manage a large number of sites, preventing web site blacklisting. Organizations that use MDS can quickly identify and eradicate malware that could infect their web site visitors and lead to loss of data and revenue. When MDS discovers infections, it supports regularly scheduled scanning to monitor web sites on an ongoing basis, with email alerts to quickly notify organizations. Information regarding malware infection helps the organizations in taking quick action in isolating and in removing malware.

Source: *http://www.qualys.com*



More webserver security tools to secure compromised webservers include:

Retina CS

Source: *http://www.beyondtrust.com*

Retina CS is a vulnerability management solution that can to discover network, web, mobile, cloud and virtual infrastructure, profile asset configuration; pinpoint vulnerabilities, malware, and attacks; analyze threat potential and return on remediation; remediate vulnerabilities via integrated patch management;, report on vulnerabilities, compliance, benchmarks, etc.; and protect endpoints against client-side attacks.

Nscan

Source: *http://nscan.hypermart.net*

Nscan scans large networks and gathers network/host information. It supports remote monitoring, usage of host and port lists, option profiles, etc. It also contains a traceroute, dig and whois that work together with the scanner.

NetIQ Secure Configuration Manager

Source: *http://www.netiq.com*

NetIQ Secure Configuration Manager protects sensitive data from breaches, exploits, or cyber-attacks and is an automated, continuous process that monitors systems for adherence to industry regulations.

SAINTscanner

Source: http://www.saintcorporation.com

SAINT uncovers areas of weakness and recommends fixes. It enables its users to identify vulnerabilities on network devices, operating systems, desktop applications, Web applications, databases, etc. It detects and fixes possible weaknesses in the network's security before intruders can exploit them. It anticipates and prevents common system vulnerabilities. It demonstrates compliance with current government and industry regulations such as PCI DSS, NERC, FISMA, SOX, GLBA, and HIPAA. It performs configuration audits with policies defined by FDCC, USGCB, and DISA.

HP WebInspect

Source: https://download.hpsmartupdate.com

HP WebInspect gives security professionals and security novices alike the power and knowledge to identify and validate critical, high-risk security vulnerabilities in applications running in development, QA, or production. HP WebInspect enables its users to increase modern Web technology coverage, accelerate security through more actionable information, elevate security knowledge across the business, comply with legal, regulatory, and architectural requirements, leverage automation to do more with less, and build an enterprise-wide application security program.

Arirang

Source: http://monkey.org

Arirang is a webserver security scanner for network. It supports operating system detection, webserver type scan, webserver allow scan, multiple hosts and ports scan, http port, virtual host scan, socket connect/recv timeout, html report, SSL scan, cidr IP scan, cidr domain scan, wide network IP range scan, wide network webserver type scan, wide network webserver allow scan, HTTP Proxy, SOCKS5 Proxy, Caching Only Proxy, http request injection, recv flags, automatic scan, and Arirang Ruby script.

N-Stalker Web Application Security Scanner

Source: http://www.nstalker.com

This tool provides a restricted set of free web security assessment checks to enhance the overall security of webserver infrastructure using a web attack signature database called the N-Stealth Web Attack Signature Database.

Infiltrator

Source: http://www.infiltration-systems.com

Infiltrator is a network security scanner that allows its users to audit network computers for vulnerabilities, exploits, and information enumerations. Infiltrator can reveal and catalog a plethora of information on scanned computers, such as installed software, shares, users, drives, hotfixes, NetBIOS and SNMP information, open ports, etc. Infiltrator can audit each computer's password and security policies, alerting its users when they should make changes to increase security. Infiltrator supports network utilities for footprinting, scanning, enumerating and

gaining access to machines. Utilities include ping sweep, whois lookups, email tracing, brute force cracking tools, share scanning, network enumerating, etc.

WebCruiser

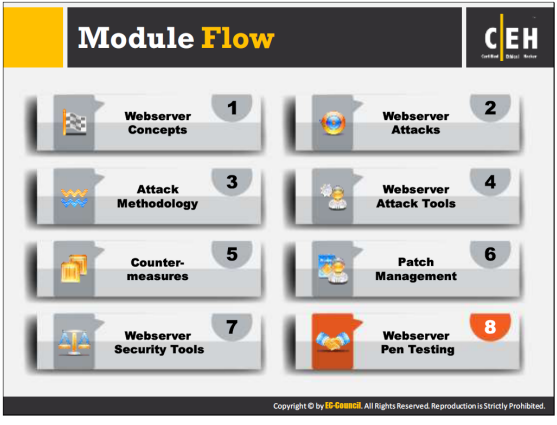
Source: http://sec4app.com

WebCruiser- Web Vulnerability Scanner is a web penetration testing tool that aids its users in auditing their website. It supports scanning the website as well as POC (Proof of concept) for web vulnerabilities. This tool supports crawler, vulnerability scanner, SQL Injection Scanner, SQL Injection Tool, SQL Injection for SQL Server, SQL Injection for MySQL, SQL Injection for Oracle, SQL Injection for DB2, SQL Injection for access, Post Data Resend, Cross Site Scripting Scanner and POC, XPath Injection Scanner and POC, as well as Auto get Cookie from Web Browser for Authentication.

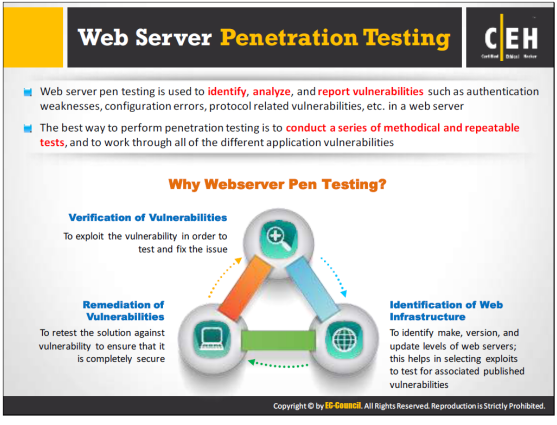
dotDefender

Source: http://www.applicure.com

dotDefender is a web application security solution (a Web Application Firewall, or WAF) that offers security for its user's websites and web applications. It uses a number of engines to detect and prevent hacking attacks, including pattern recognition, a signature knowledgebase, Data Leakage Protection (DLP), and upload inspection.

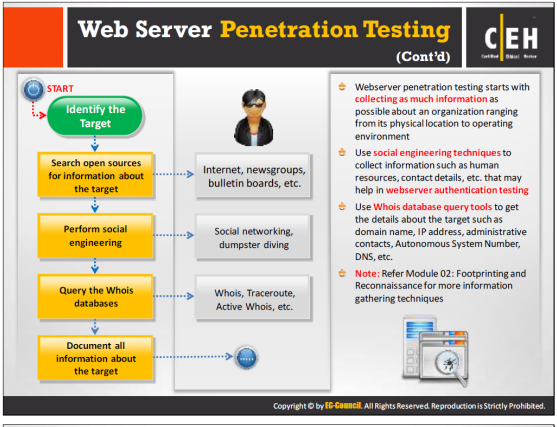


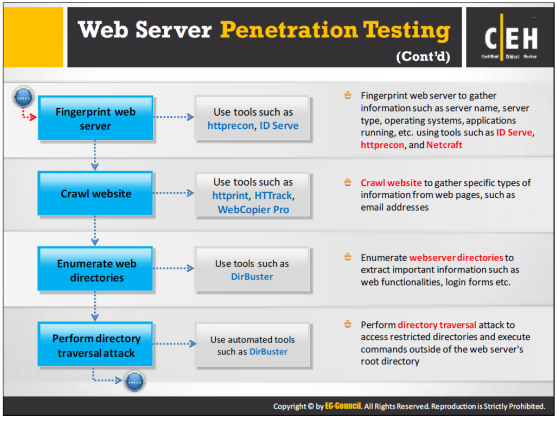
This section describes the webservers pen testing carried out to test the security posture of a webserver by simulating attacks on it much like an attacker, in order to find vulnerable areas in the webserver environment. Pen testing is a step-by-step process carried out by a pen tester with the help of pen testing tools.

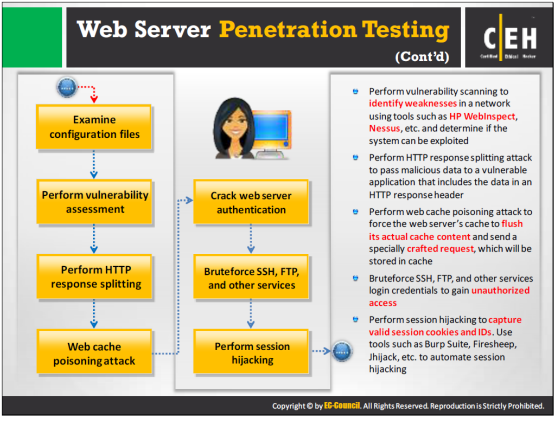


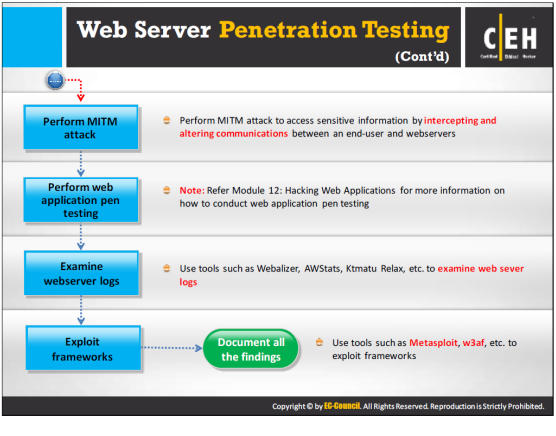
To be become successful pen tester, think like an attacker!

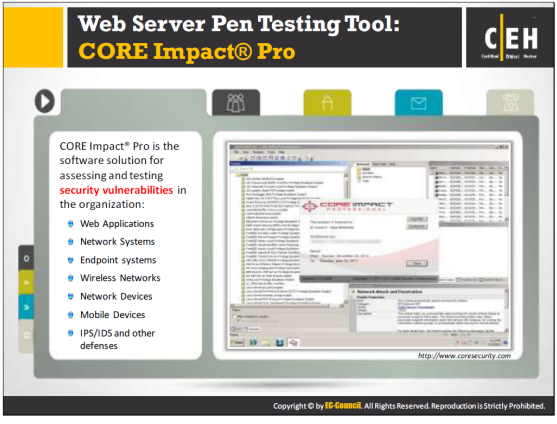
Pen or "security testing" is a methodology in which a pen tester simulates different types of attacks on an organizations' information system to assess its security. Webserver pen testing tests an organization's webserver security. Pen testing tools can automate the process.











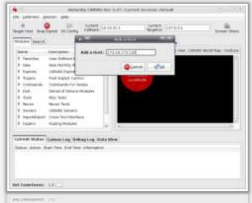
The CORE Impact® Pro finds vulnerabilities on an organization's webserver. This tool allows a user to evaluate the security posture of a webserver using the same techniques employed by today's cyber-criminals. It scans possible vulnerabilities in the webserver, imports scan results, and runs exploits to test identified vulnerabilities. The tool can also:

- Scan network servers, workstations, firewalls, routers and various applications for vulnerabilities
- Identify which vulnerabilities pose real threats to the network
- Determine the potential impact of exploited vulnerabilities
- Prioritize and execute remediation efforts

Source: http://www.coresecurity.com

Web Server Pen Testing Tool: Immunity CANVAS

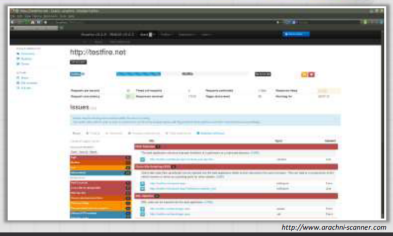
CANVAS is an automated exploitation system, and a comprehensive, reliable **exploit development framework** to security professionals and penetration testers



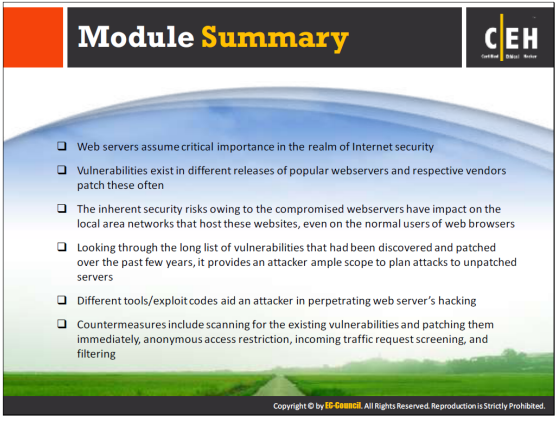
http://www.immunitysec.com

Web Server Pen Testing Tool: Arachni

Arachni is an open source, feature-full, modular, high-performance Ruby framework aimed towards helping penetration testers and administrators evaluate the **security of web applications**



http://www.arachni-scanner.com



This module provided an overview of webservers concepts, attacks, and attack methodology; attack tools, countermeasures, security tools, and pen testing. In the next module, we will see how attackers as well as ethical hackers and pen testers hack web applications.