Session Hijacking

Module 11

Engineered by Hackers. Presented by Professionals.

SECURITY NEWS

December 06, 2010

eSecurity Planet

Firesheep Fix as Easy as HTTPS

November 2010 will be remembered as the month that Firesheep exploded onto the computing scene, much to the delight of college students everywhere. The Firefox browser add-on makes it trivial to **gain access to anyone's Facebook account while they're connected to the Internet** using an open, unsecured Wi-Fi connection.

Of course, the session hijacking attack vulnerability that Firesheep exploits has been well-known in hacking and security circles for ages – all that Firesheep does is make the attack spectacularly easy. And it's a bit unfair to highlight Facebook as being susceptible to the attack, if only because many other popular sites, including Flickr, Foursquare and Wordpress are just as susceptible to it, too.

http://www.esecurityplanet.com

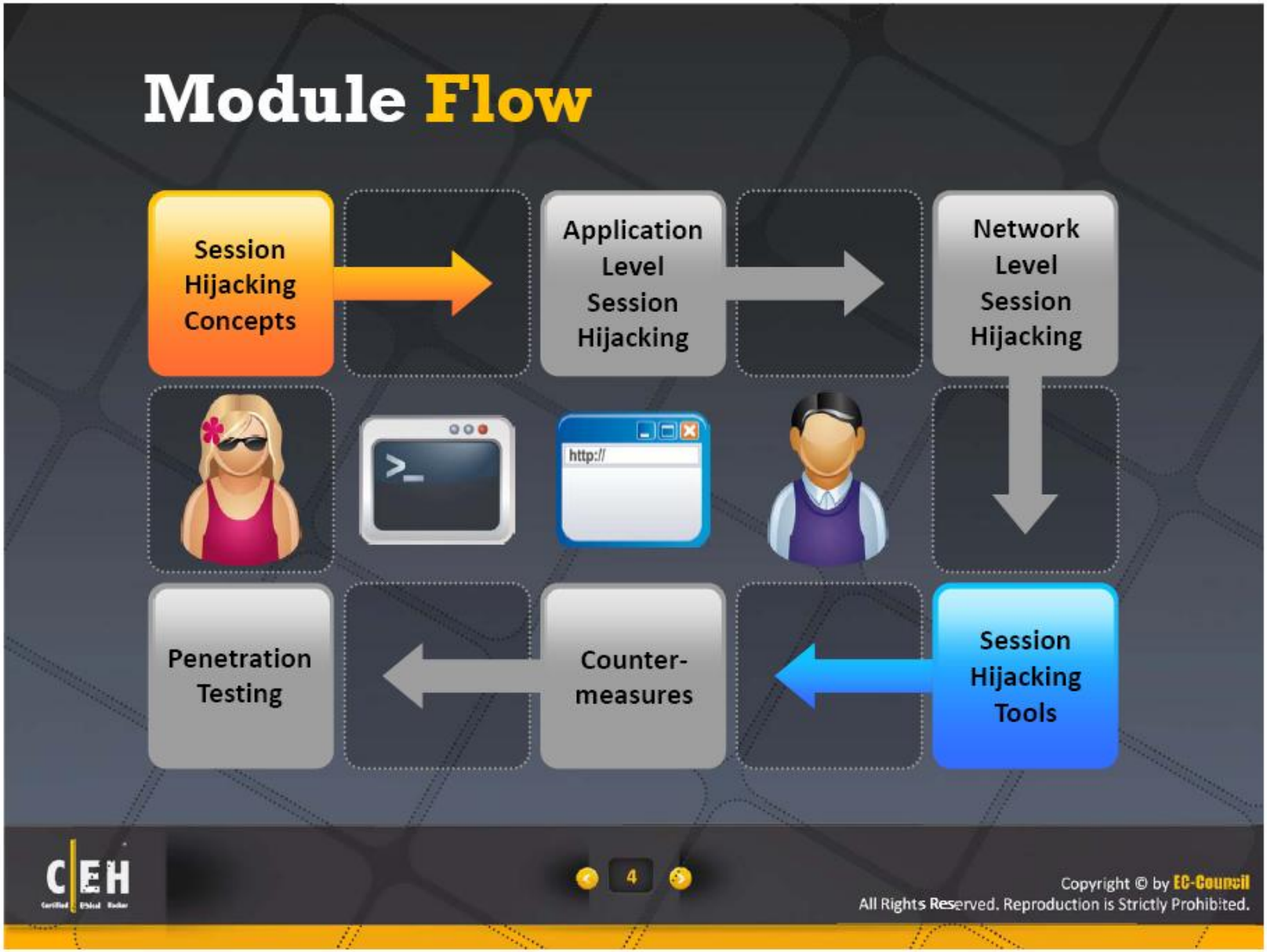
Module Objectives

- What is Session Hijacking?
- Key Session Hijacking Techniques
- Brute Forcing
- Spoofing vs. Hijacking
- Session Hijacking Process
- Types of Session Hijacking
- Session Hijacking in OSI Model

- Application Level Session Hijacking
- Network Level Session Hijacking
- TCP/IP Hijacking
- Session Hijacking Tools
- Countermeasures
- IPSec Architecture
- Penetration Testing



Module Flow


Session Hijacking Concepts
Application Level Session Hijacking
Network Level Session Hijacking
Session Hijacking Tools
Counter-measures
Penetration Testing
http://
4
Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

What is Session Hijacking?

Session Hijacking refers to the exploitation of a valid computer session where an attacker takes over a session between two computers

The attacker steals a valid session ID which is used to get into the system and snoop the data

In TCP session hijacking, an attacker takes over a TCP session between two machines

Since most authentication only occurs at the start of a TCP session, this allows the attacker to gain access to a machine

Dangers Posed by Hijacking

Most countermeasures do not work unless you use encryption

Hijacking is simple to launch

Threat of identity theft, information loss, fraud, etc.

Most computers using TCP/IP are vulnerable

You can do little to protect against it unless you switch to another secure protocol


Why Session Hijacking is Successful?
No Account Lockout For Invalid Session Ids
Weak Session ID Generation Algorithm
Insecure Handling
Indefinite Session Expiration Time
Small Session Ids
Clear Text Transmission
101010
010101
101
CEH
7
Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Key Session Hijacking Techniques

Brute Forcing

The attacker attempts different IDs until he succeeds

Stealing

Attacker uses different techniques to steal Session IDs

Calculating

Using non-randomly generated IDs, an attacker tries to calculate the Session IDs

Brute Forcing

Using **brute force attacks**, an attacker tries to guess **session ID** until he guesses the session ID

For example, in the URL's, an attacker is trying to guess the session ID

http://www.my site.com/view/VW30422101518909

http://www.mysite.com/view/VW30422101520803

http://www.mysite.com/view/VW30422101522507

Session ID's can be stolen using different techniques such as:

1. Using the HTTP referrer header
2. Sniffing the network traffic
3. Using the Cross-Site Scripting attacks
4. Sending Trojans on client PCs

1. Using a "**referrer attack**", an attacker tries to lure a user to click on a link to another site (a mysite link, say www.mysite.com)
2. For example, GET /index.html HTTP/1.0 Host: www.mysite.com Referrer: www.mywebmail.com/viewmsg.asp?msgid=689645&SID=2556X54VA75
3. The attacker obtains the session ID of the user by sending when the **browser sends the referrer URL** that contains the session ID of the user to the attacker's site (www.mysite.com)

CEH Certified Ethical Hacker

9

Brute Forcing Attack

- Using brute force attacks, an attacker tries to guess session ID until he finds the correct session ID
- Possible range of values for the session ID must be limited to perform a successful bruteforce attack

Note: Session ID brute forcing attack is known as session prediction attack if the predicted range of values for a session ID is very small

HTTP Referrer Attack

In a referrer attack, attacker tries to **lure a user** to click on a link to another site (a mysite link, say www.mysite.com)

For example, `GET /index.html HTTP/1.0 Host: www.mysite.com Referrer: www.mywebmail.com/viewmsg.asp?msgid=689645&SID=2556X54VA75`

The browser **sends the referrer URL** containing the session ID to the attacker's site - www.hostile.com, and the attacker now has the session ID of the user

Spoofing vs. Hijacking

Spoofing Attack

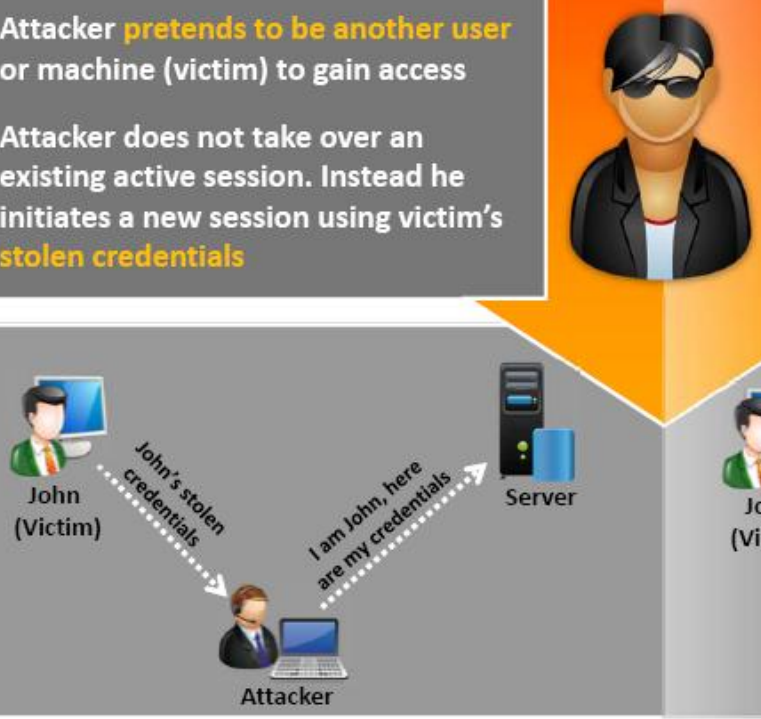
Attacker pretends to be another user or machine (victim) to gain access

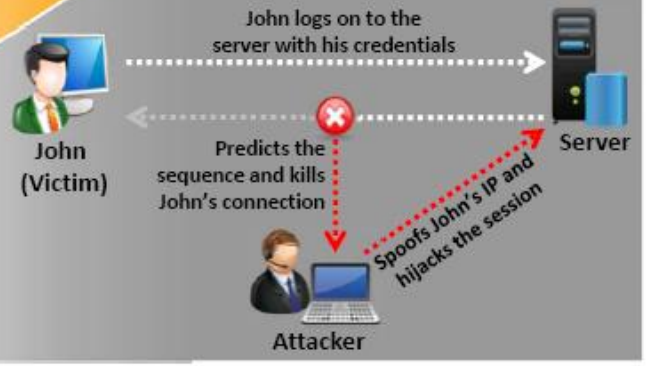
Attacker does not take over an existing active session. Instead he initiates a new session using victim's stolen credentials

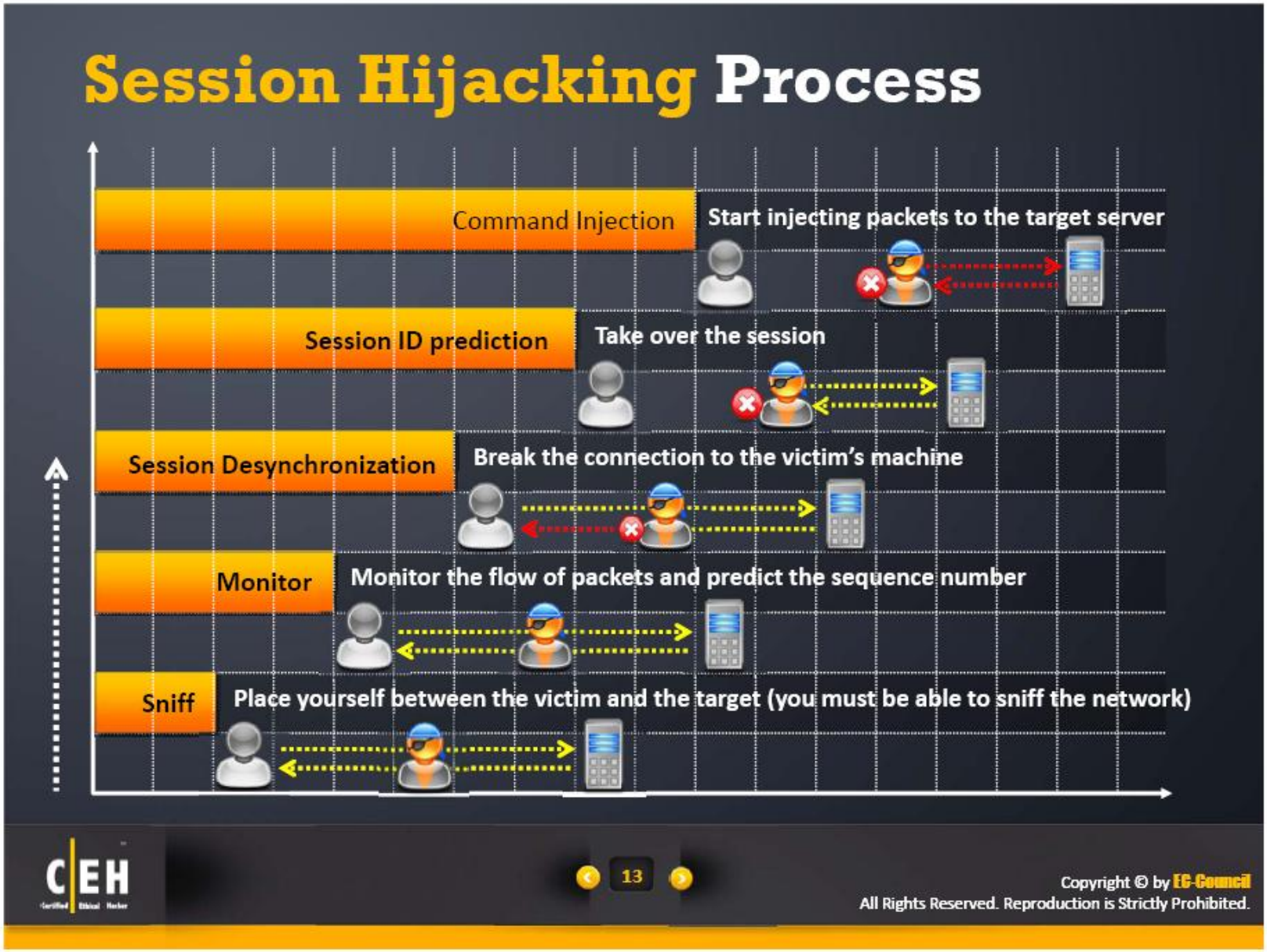
Hijacking

Session hijacking is the process of taking over an existing active session

Attacker relies on the legitimate user to make a connection and authenticate






Session Hijacking Process
Command Injection
Start injecting packets to the target server
Session ID prediction
Take over the session
Session Desynchronization
Break the connection to the victim's machine
Monitor
Monitor the flow of packets and predict the sequence number
Sniff
Place yourself between the victim and the target (you must be able to sniff the network)
CEH
13
Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Packet Analysis of a Local Session Hijack

SYN <Clt ISN 4000><WIN 512>

SYN <Svr ISN 5000><WIN 1024> /ACK 4001

ACK 4001

DATA=128 <Clt SEQ 4001>

ACK (Clt SEQ + DATA) 4129

DATA=91 <Clt SEQ 4129>

ACK (Clt SEQ + DATA) 4220

Types of Session Hijacking

Session Hijacking in OSI Model

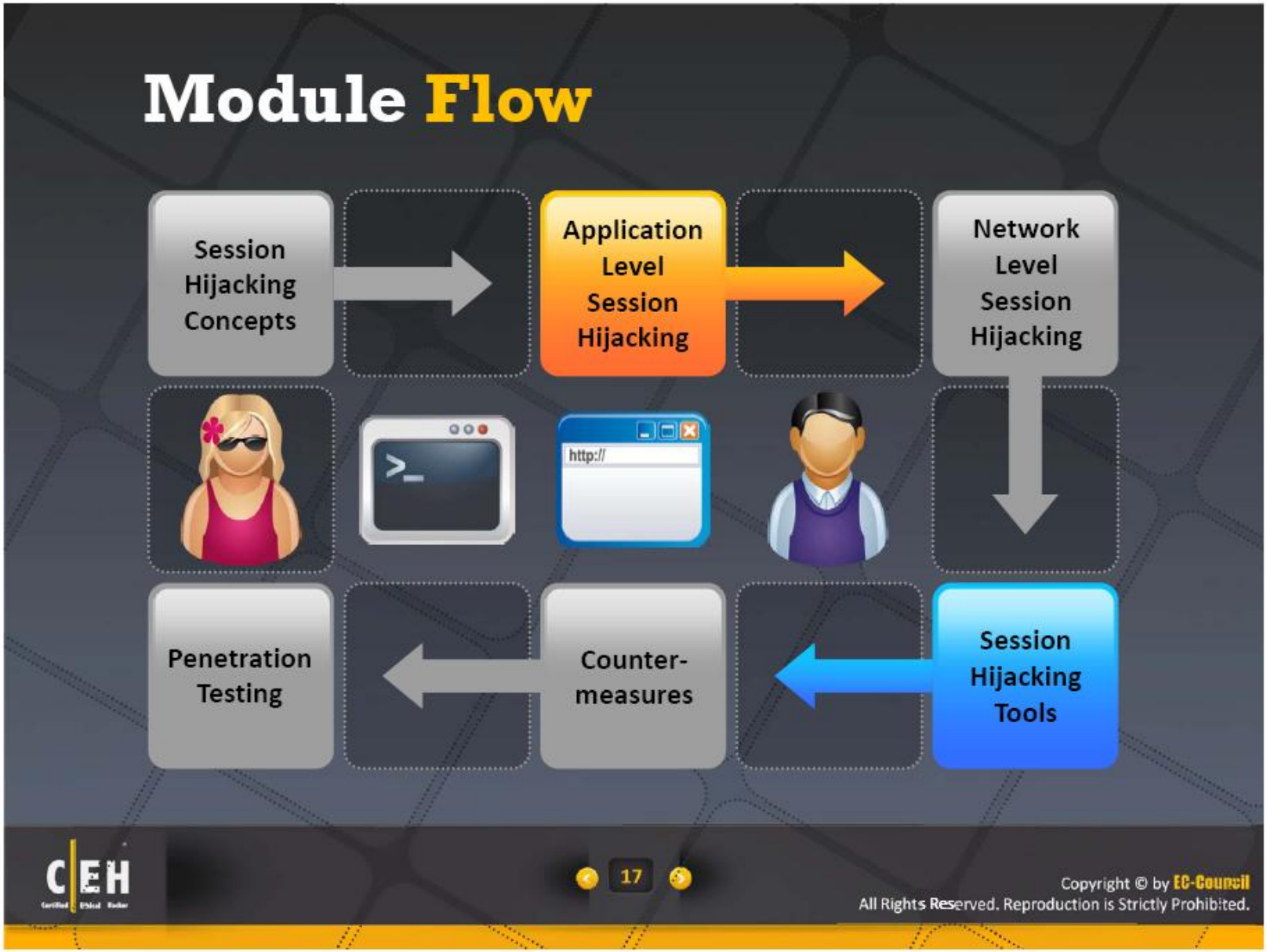
Network Level Hijacking

- Network level can be defined as the **interception of the packets** during the transmission between the client and the server in a TCP and UDP session

Application Level Hijacking

- Application level is about **gaining control** on the **HTTP's user session** by obtaining the session ID's

Module Flow


Session Hijacking Concepts
Application Level Session Hijacking
Network Level Session Hijacking
Session Hijacking Tools
Counter-measures
Penetration Testing
http://
17
Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Application Level Session Hijacking

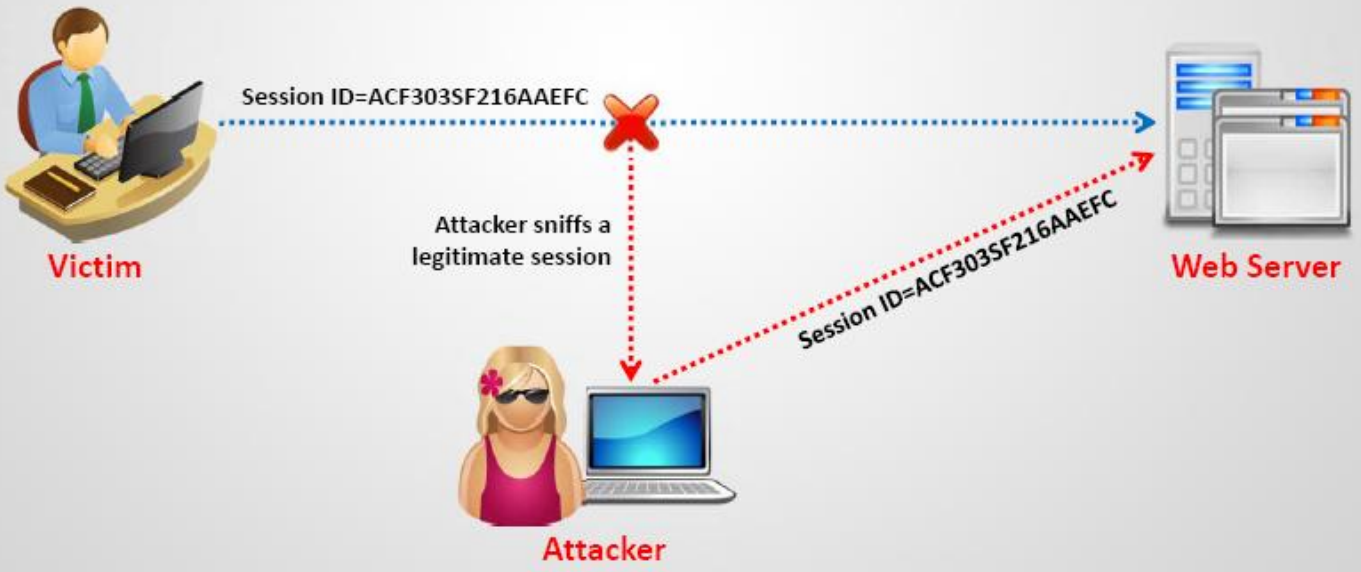
In a Session Hijacking attack, a session token is stolen or a valid session token is predicted to **gain unauthorized access** to the web server

A session token can be compromised in various ways

- Session Sniffing
- Predictable session token
- Man-in-the-middle attack
- Man-in-the-browser attack
- Client-side attacks

Session Sniffing

- Attacker uses a sniffer to capture a valid session token called "Session ID"
- Attacker then uses the valid token session to gain unauthorized access to the web server



Predictable Session Token

It is a method used for predicting a session ID or to **impersonate a web site user**

Predicting a session ID is also known as **Session Hijacking**

Using session hijacking technique, an attacker gets the ability to **ping web site requests** with compromised user's privileges

By guessing the unique **session value or deducing** the session ID accomplishes the attack

How to Predict a Session Token?

Most of the webservers use custom **algorithms** or a predefined pattern to generate sessions IDs

Captures

Attacker captures several session IDs and analyzes the pattern

```
http://www.juggyboy.com/view/JBEX21092010152820
http://www.juggyboy.com/view/JBEX21092010153020
http://www.juggyboy.com/view/JBEX21092010160020
http://www.juggyboy.com/view/JBEX21092010164020
```

Constant Date Time

Predicts

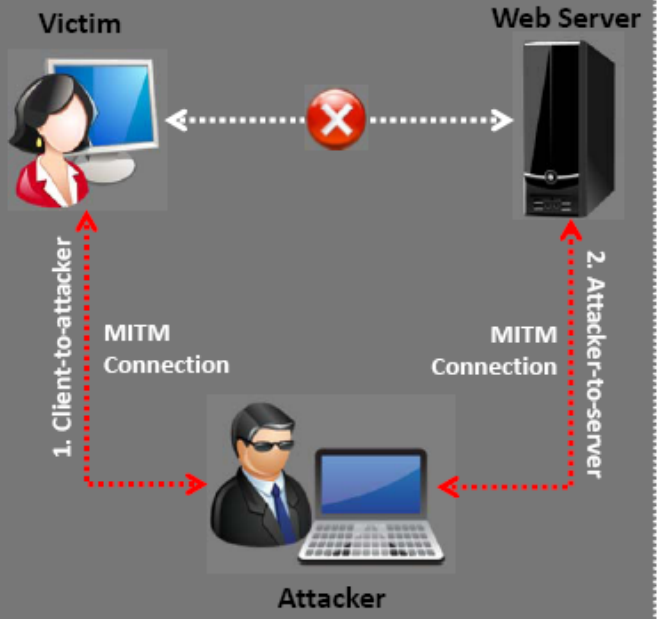
At 16:25:55 on Sep-25, 2010, attacker can successfully predict the session ID to be

```
http://www.juggyboy.com/view/JBEX25092010162555
```

Constant Date Time

Man-in-the-Middle Attack

The man-in-the-middle attack is used to **intrude into an existing connection** between systems and to intercept messages being exchanged



Attackers use different techniques and **split the TCP connection** into two connections

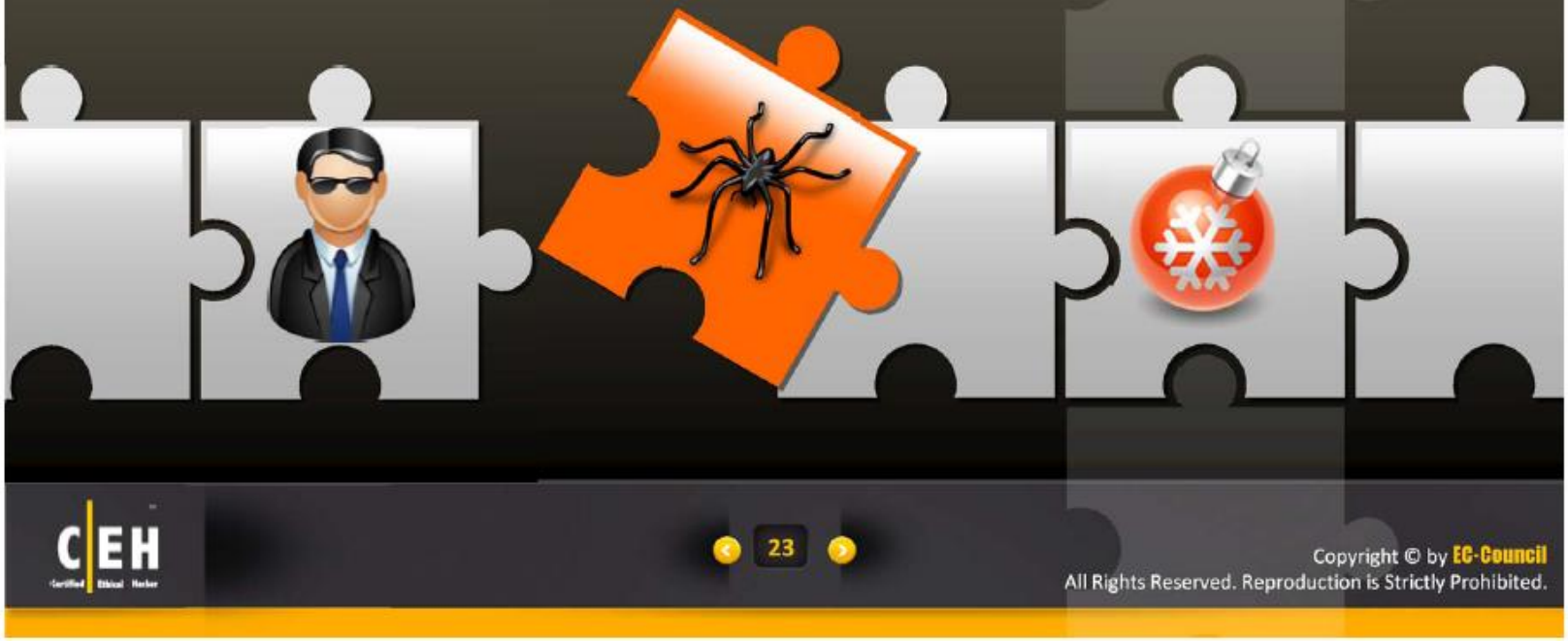
1. Client-to-attacker connection
2. Attacker-to-server connection

After the successful interception of TCP connection, an attacker can read, modify, and insert fraudulent data into the **intercepted communication**

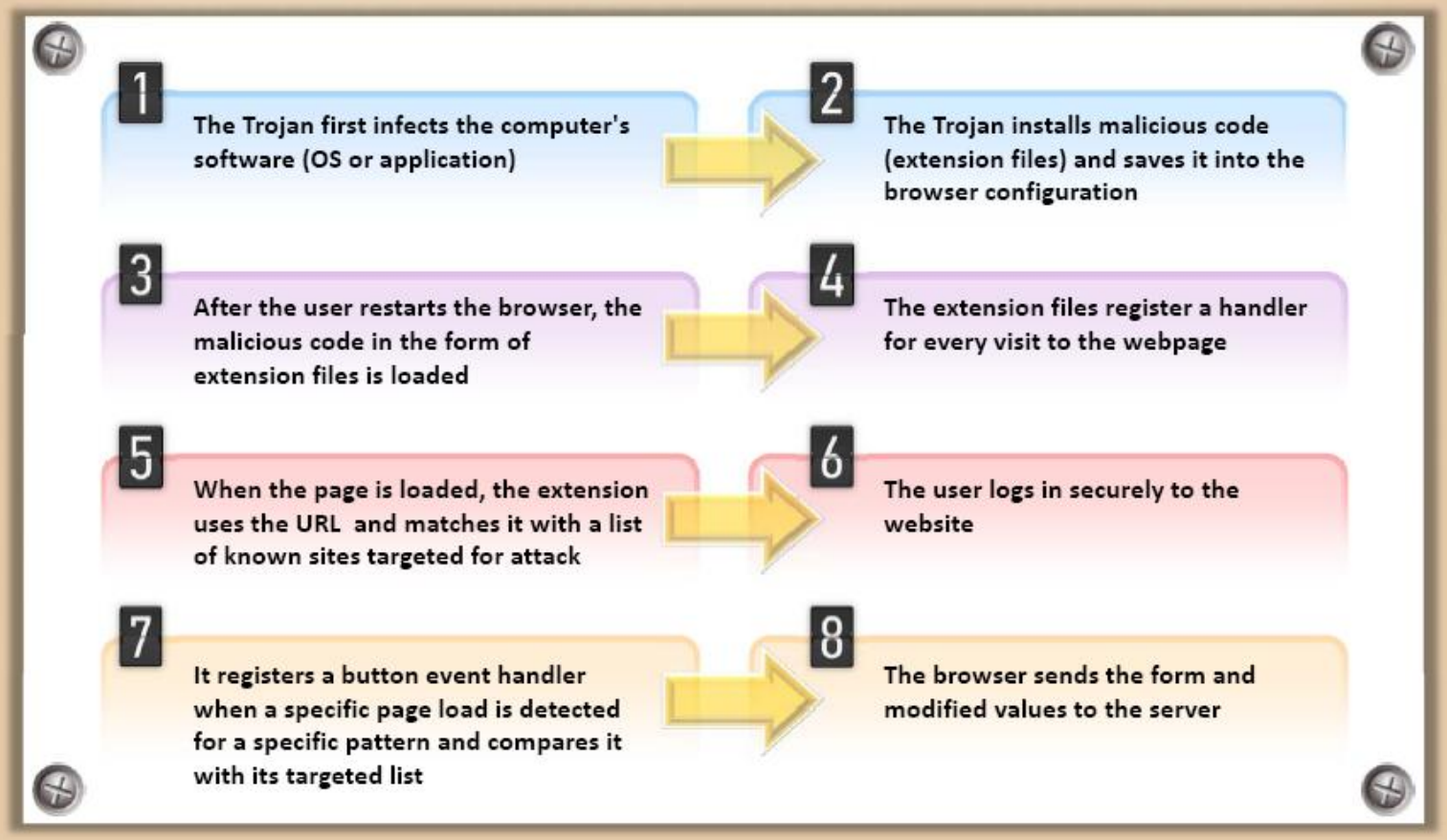
In the case of an **http transaction**, the TCP connection between the client and the server becomes the target

Man-in-the-Browser Attack

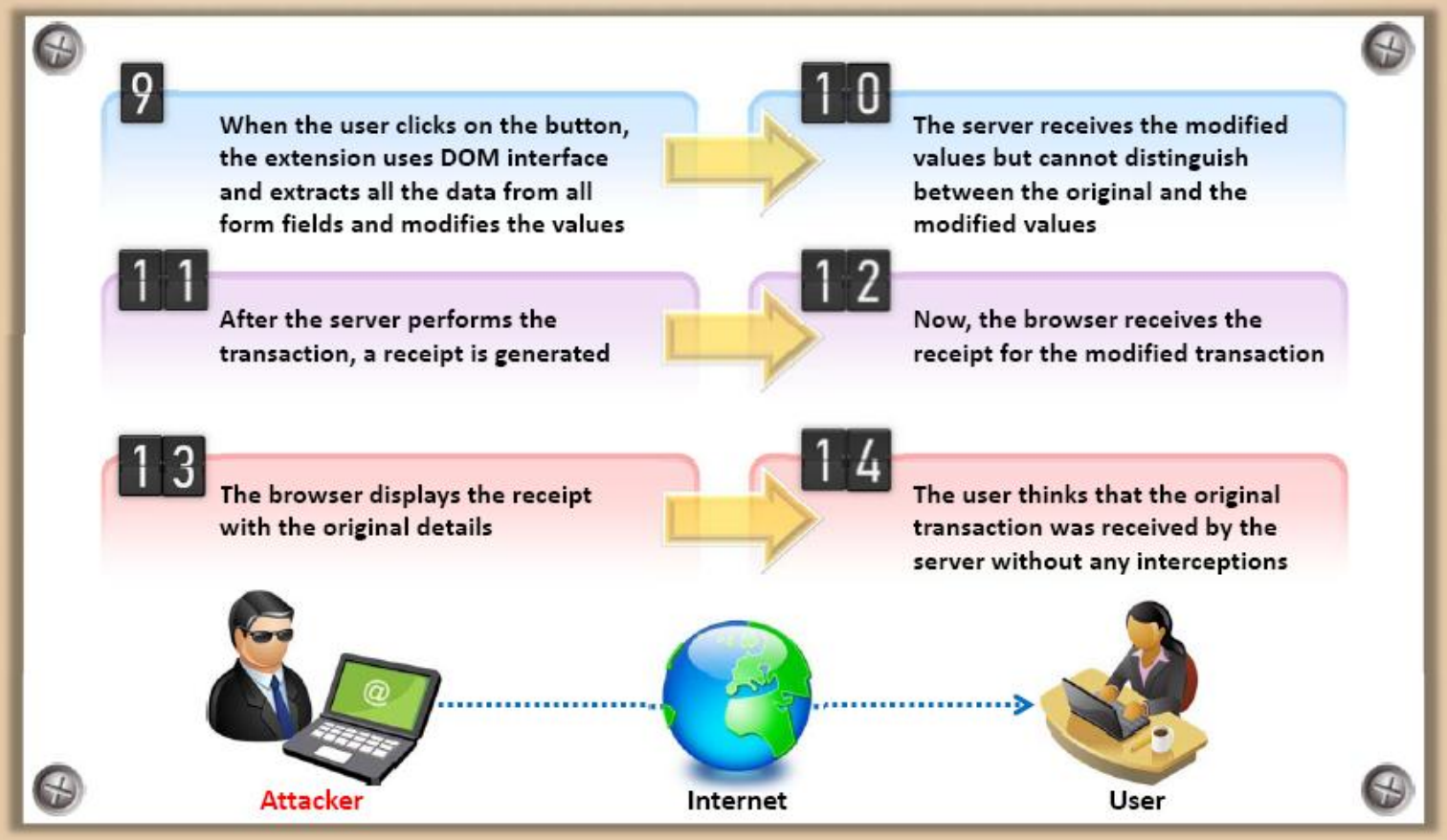
- Man-in-the-browser attack **uses a Trojan Horse** to intercept the calls between the browser and its security mechanisms or libraries
- It works with an already installed Trojan horse and acts between the **browser and its security mechanisms**
- Its main objective is to cause financial deceptions by manipulating transactions of Internet Banking systems



Steps to Perform Man-in-the-Browser Attack



Steps to Perform Man-in-the-Browser Attack



Client-side Attacks

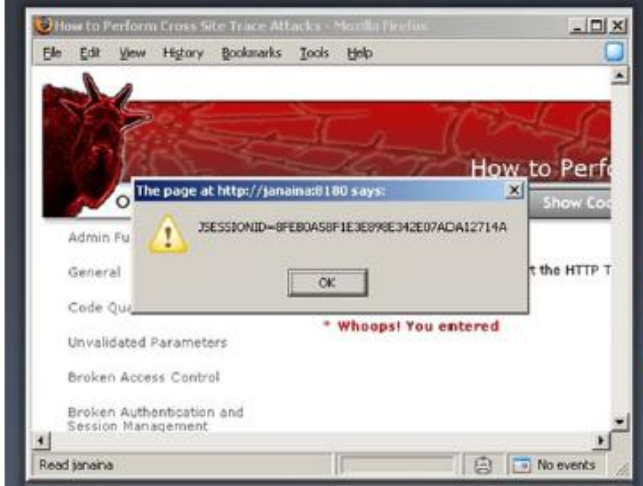
XSS	Malicious JavaScript Codes	Trojans
Cross-Site Scripting attacks are a type of injection attacks, in which the malicious scripts are injected into the web sites	A malicious script can be embedded in a web page and does not generate any type of warnings when the page is viewed in any browser	Trojan horse is a program in which the malicious code is contained inside apparently harmless programming or data in such a way that it can get control and cause damage

Cross-site Script Attack

The attacker can compromise the session token by sending malicious code or programs to the client-side programs

The example here shows how the attacker steals the session token using XSS attack

If an attacker sends a crafted link to the victim with the malicious JavaScript, when the victim clicks on the link, the JavaScript will run and complete the instructions made by the attacker



The example here uses an XSS attack to show the cookie value of the current session

Using the same technique, it is possible to create a specific JavaScript code that will send the cookie to the attacker `<SCRIPT>alert (document.cookie);</SCRIPT>`

Session Fixation

Session Fixation is an attack that allows an attacker to hijack a **valid user session**

The attack tries to lure a user to authenticate himself with a known session ID and then hijacks the **user-validated session** by the knowledge of the used session ID

The attacker has to provide a **legitimate web application** session ID and try to lure victim browser to use it

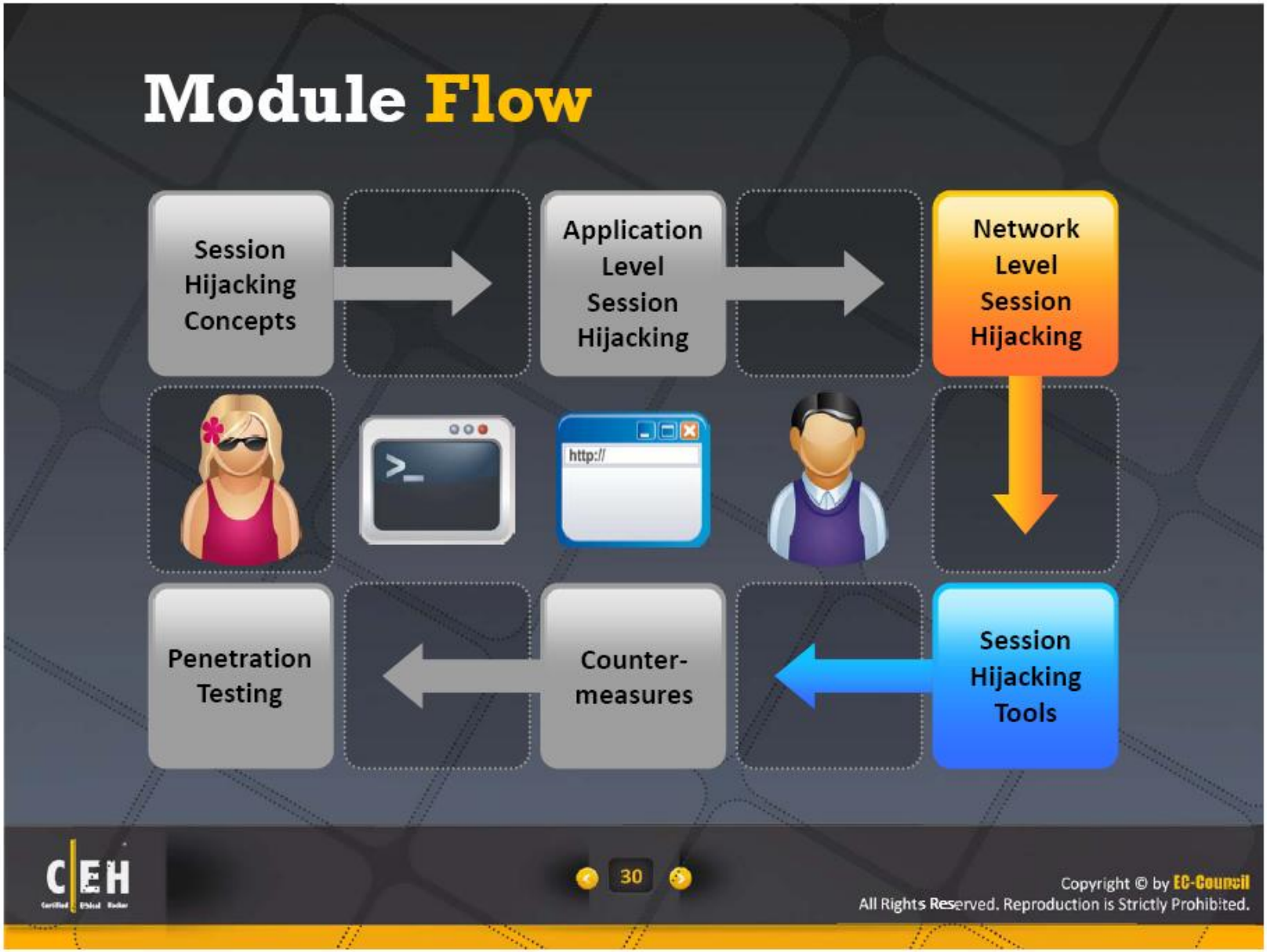
Several techniques to **execute Session Fixation** attack are

- Session token in the **URL argument**
- Session token in a **hidden form field**
- Session ID in a **cookie**

Session Fixation Attack

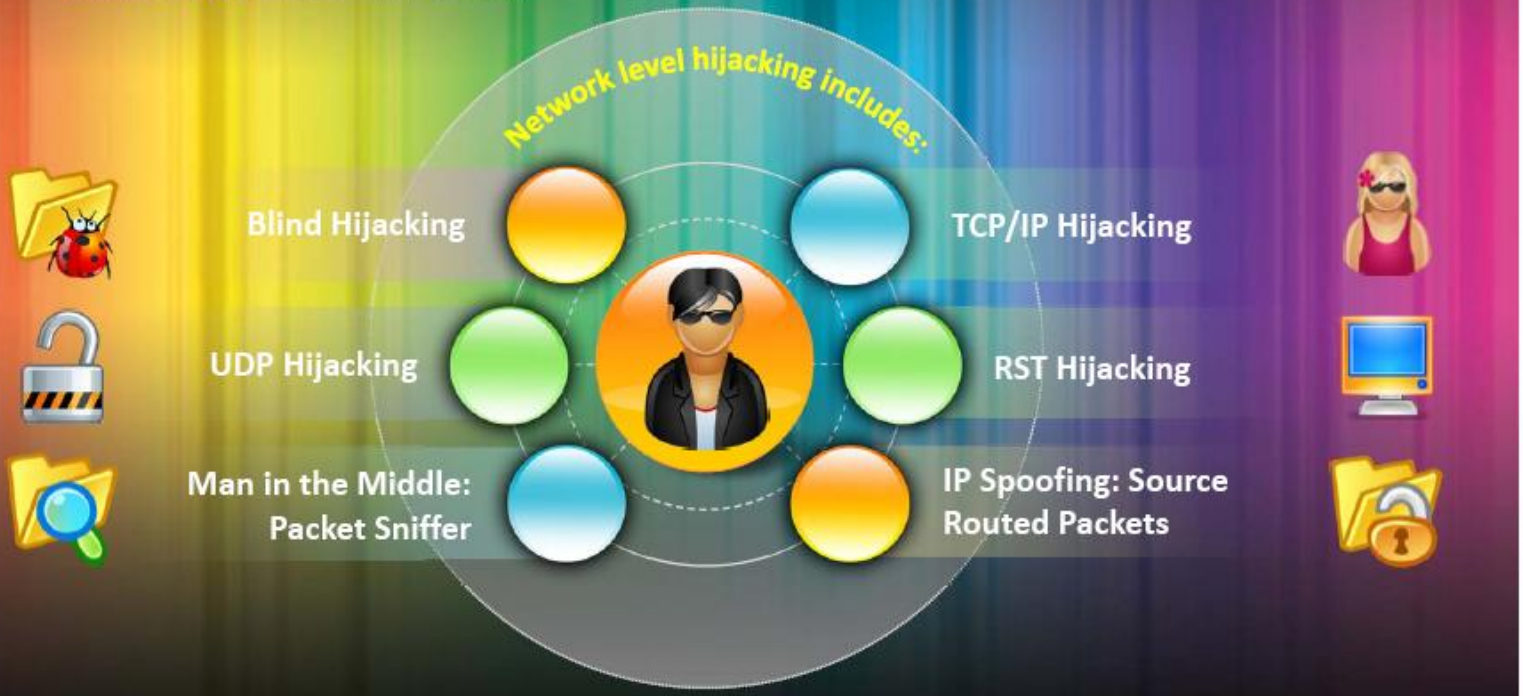
- Attacker exploits the **vulnerability of a server** which allows a user to use fixed SID
- Attacker provides a **valid SID** to a victim and lures him to **authenticate himself** using that SID

Module Flow



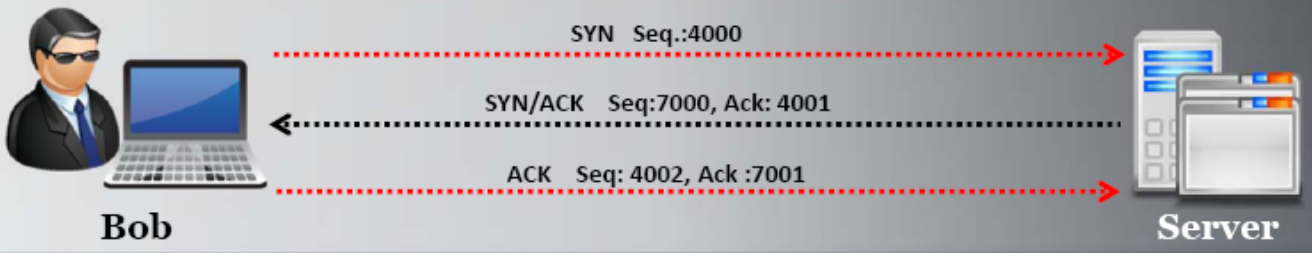
Network Level Session Hijacking

- The network level hijacking is **implemented on the data flow of the protocol** shared by all web applications
- By attacking the network level sessions, the attacker gathers some **critical information** which is used to **attack the application level sessions**



The 3-Way Handshake

If the attacker can anticipate the next sequence and ACK number that Bob will send, he/she will spoof Bob's address and start a communication with the server



1. Bob initiates a connection with the server and sends a packet to the server with the SYN bit set
2. The server receives this packet and sends back a packet with the SYN/ACK bit and an ISN (Initial Sequence Number) for the server
3. Bob sets the ACK bit acknowledging the receipt of the packet and increments the sequence number by 1
4. Now, the two machines successfully established a session

Sequence Numbers

Sequence numbers are important in providing a reliable communication and are also crucial for hijacking a session

They are a 32-bit counter. Therefore, the possible combinations can be over 4 billion

They are used to tell the receiving machine in what order the packets should go when they are received

Therefore, an attacker must successfully guess the sequence numbers in order to hijack a session

Sequence Number Prediction

After a client sends a connection request (SYN) packet to the server, the server responds (SYN-ACK) with a sequence number of choosing, which must be acknowledged by the client

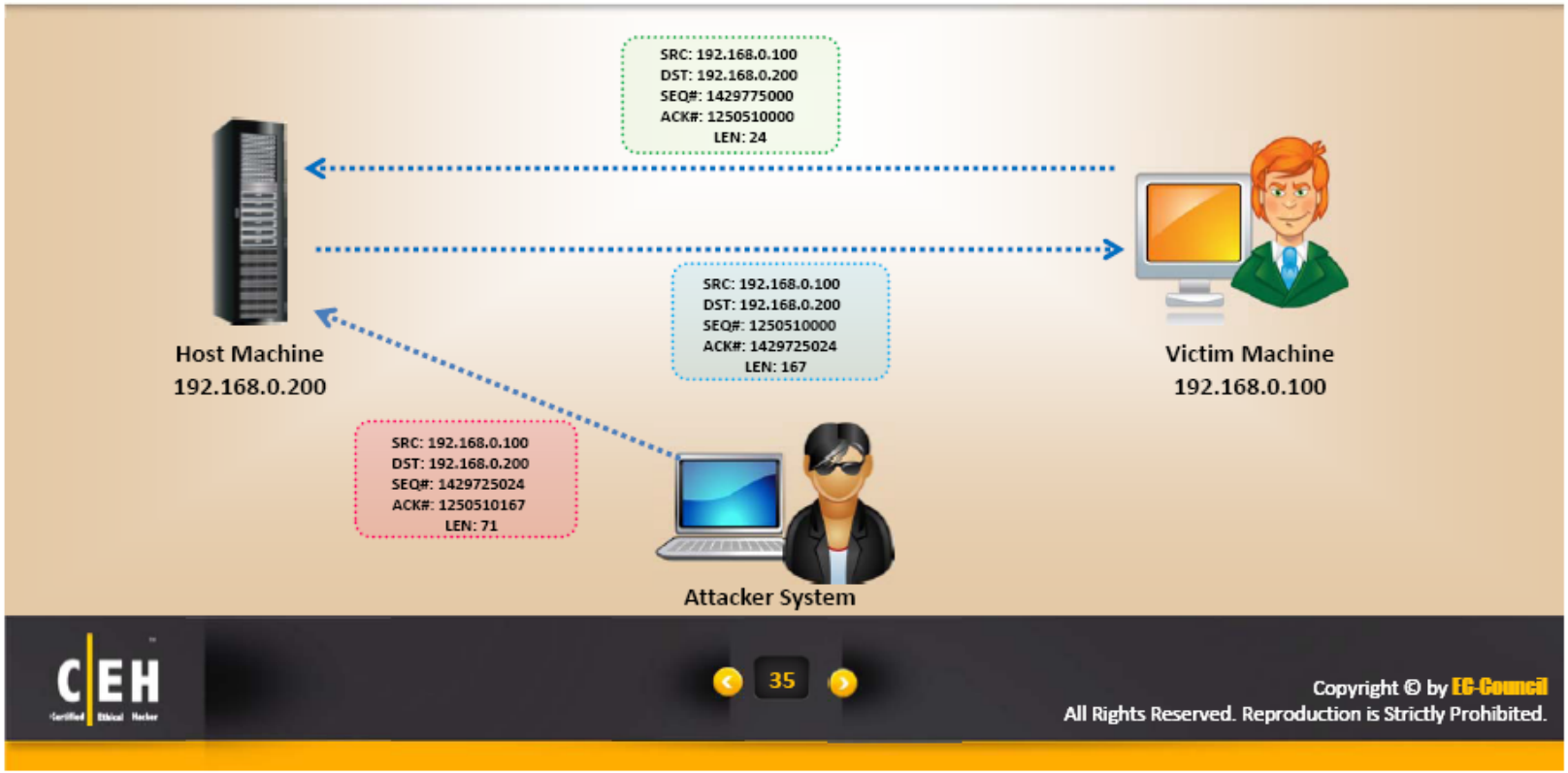
This sequence number is predictable; the attack connects to a server first with its own IP address, records the sequence number chosen, and then opens a second connection from a forged IP address

If the source IP address is used for authentication, then the attacker can use one-sided communication to break into the server

The attack does not see the SYN-ACK (or any other packet) from the server, but can guess the correct response

TCP/IP Hijacking

- TCP/IP hijacking is a hacking technique that uses **spoofed packets** to take over a connection between a victim and a target machine
- The victim's connection hangs and the attacker is then able to **communicate with the host's machine** as if the attacker is the victim
- To launch a TCP/IP hijacking attack, the **attacker must be on the same network as the victim**
- The target and the victim machines can be anywhere



TCP/IP Hijacking

1. Attacker sniffs the victim's connection and uses victim's IP to send a spoofed packet with the predicted sequence number
2. Host processes the spoofed packet, increments the sequence number and sends acknowledgement to the victim's IP
3. Victim machine is unaware of the spoofed packet, so it ignores the host machine's ACK packet and turns sequence number count off
4. Therefore, the host receives packets with the incorrect sequence number
5. The attacker forces the victim's connection with the host machine to a desynchronized state
6. The attacker tracks sequence numbers and continuously spoofs packets that comes from the victim's IP
7. The attacker continues to communicate with the host machine while the victim's connection hangs

IP Spoofing: Source Routed Packets

Source Routed Packets technique is used for gaining unauthorized access to the computer with the aid of the trusted host's IP address

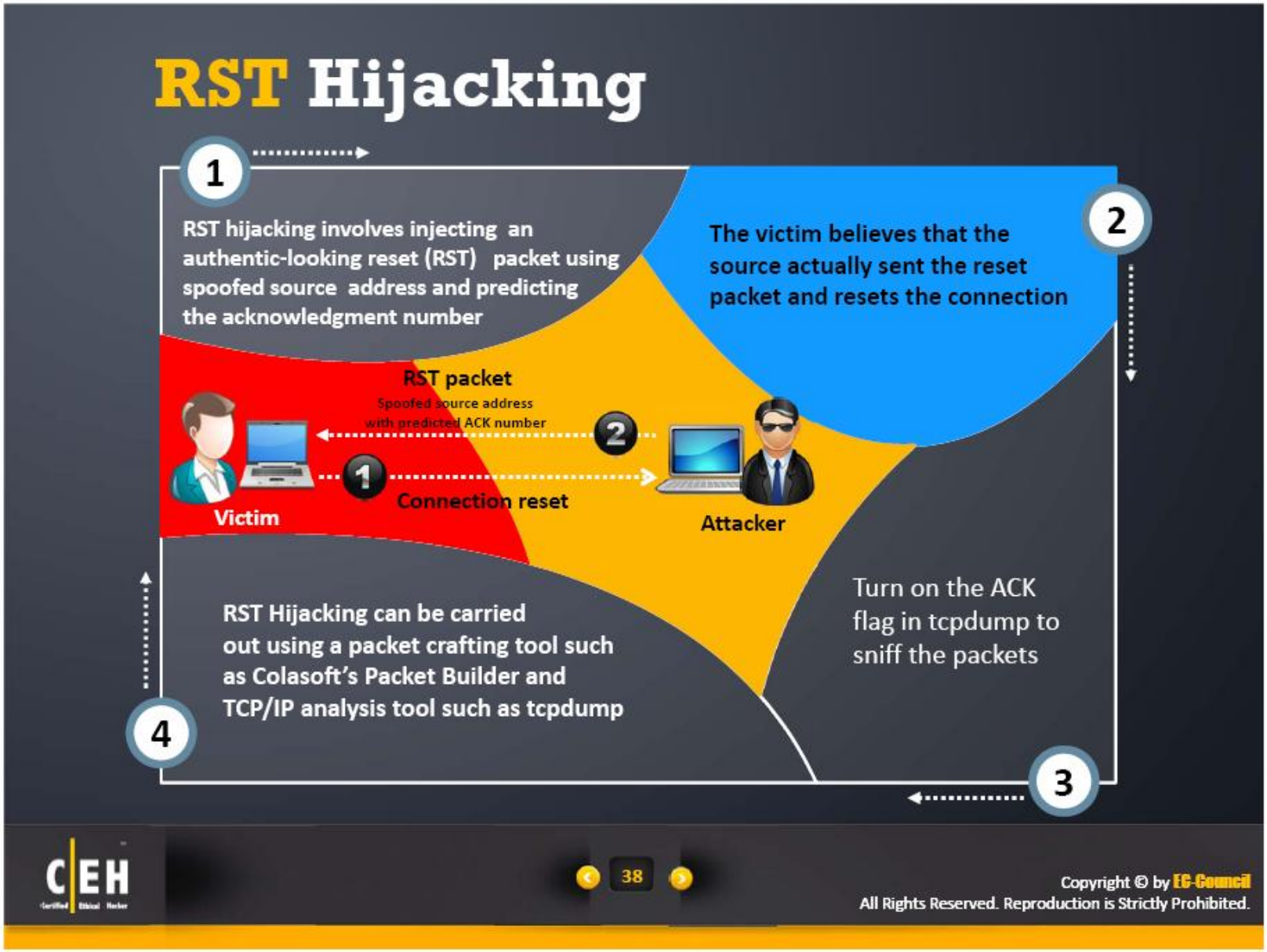
The host's IP address spoofs the packets so that the server managing a session with the client, accepts the packets

When the session is established, the hijacker injects the forged packets before the client responds

The original packet is lost as the server gets the packet with a different sequence number

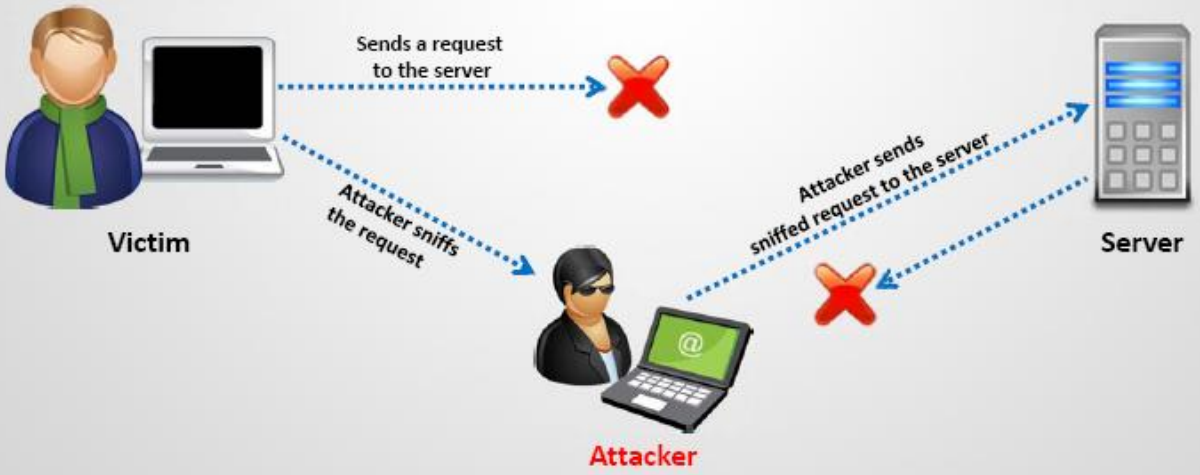
The packets are source-routed where the patch to the destination IP can be specified by the attacker

RST Hijacking



Blind Hijacking

- The attacker can inject the malicious data or commands into the intercepted communications in the TCP session even if the source-routing is disabled
- The attacker can send the data or comments but has no access to see the response



Man-in-the-Middle Attack using Packet Sniffer

In this attack, the packet sniffer is used as an interface between the client and the server

The packets between the client and the server are routed through the hijacker's host by using two techniques

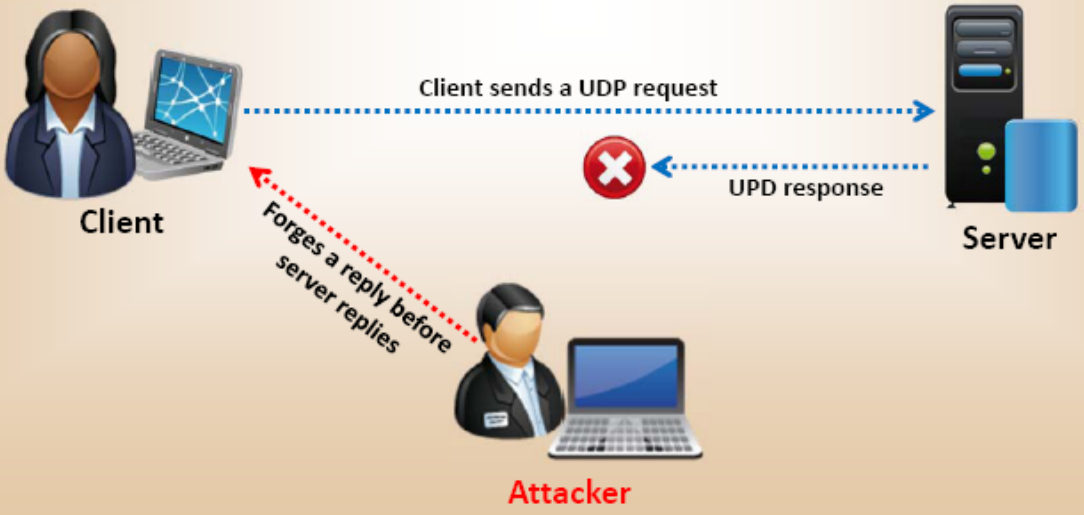
Using forged Internet Control Message Protocol (ICMP) – It is an extension of IP to send error messages where the attacker can send messages to fool the client and the server

Using Address Resolution Protocol(ARP) spoofing – ARP is used to map the local IP addresses to hardware addresses or MAC addresses

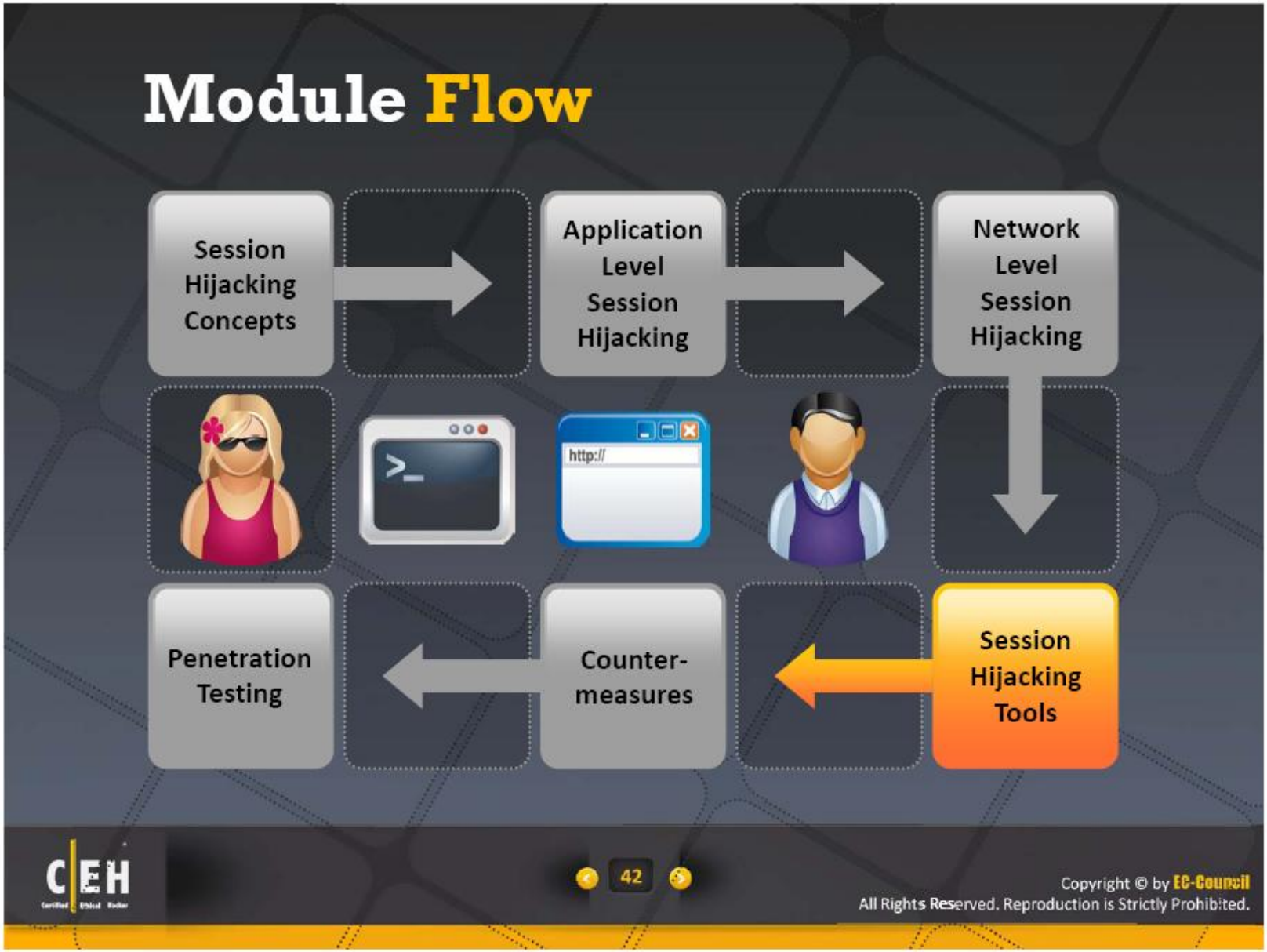
ARP spoofing involves fooling the host by **broadcasting the ARP request** and changing its ARP tables by sending the forged ARP replies

UDP Hijacking

1. Attacker sends a forged server reply to the client's UDP request before the server responds to it
2. Attacker uses Man-in-the-Middle attack to intercept server's response to the client and sends its own forged reply

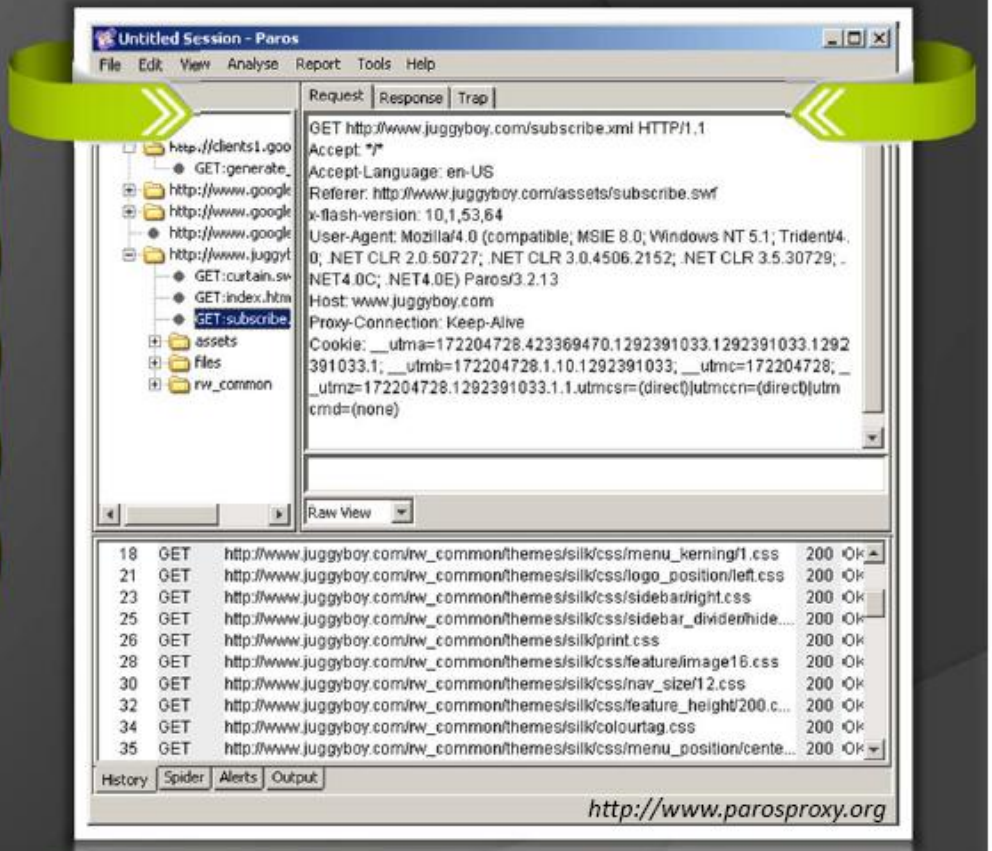


Module Flow


Session Hijacking Concepts
Application Level Session Hijacking
Network Level Session Hijacking
Session Hijacking Tools
Counter-measures
Penetration Testing
http://
42
CEH
Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

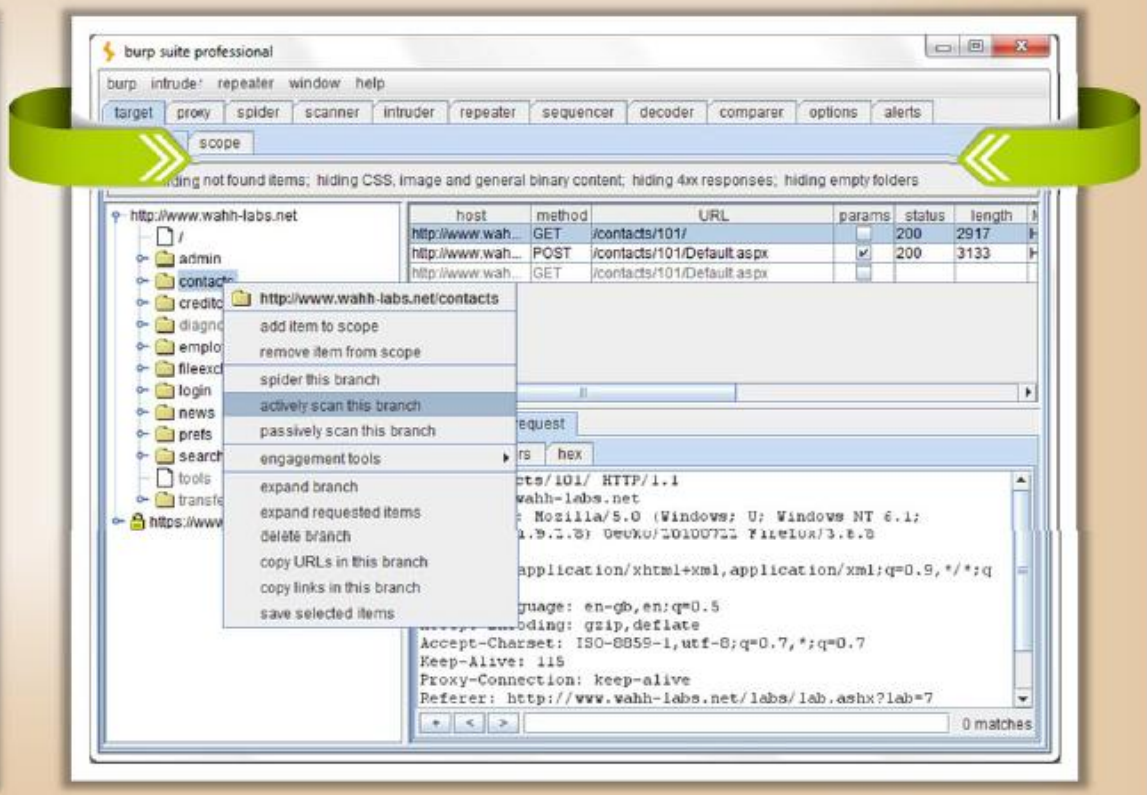
Session Hijacking Tool: Paros

- Paros is a man-in-the-middle proxy and application vulnerability scanner
- It allows attacker to intercept, modify, and debug HTTP and HTTPS data on-the-fly between a web server and a client browser
- It also supports spidering, proxy-chaining, filtering, and application vulnerability scanning



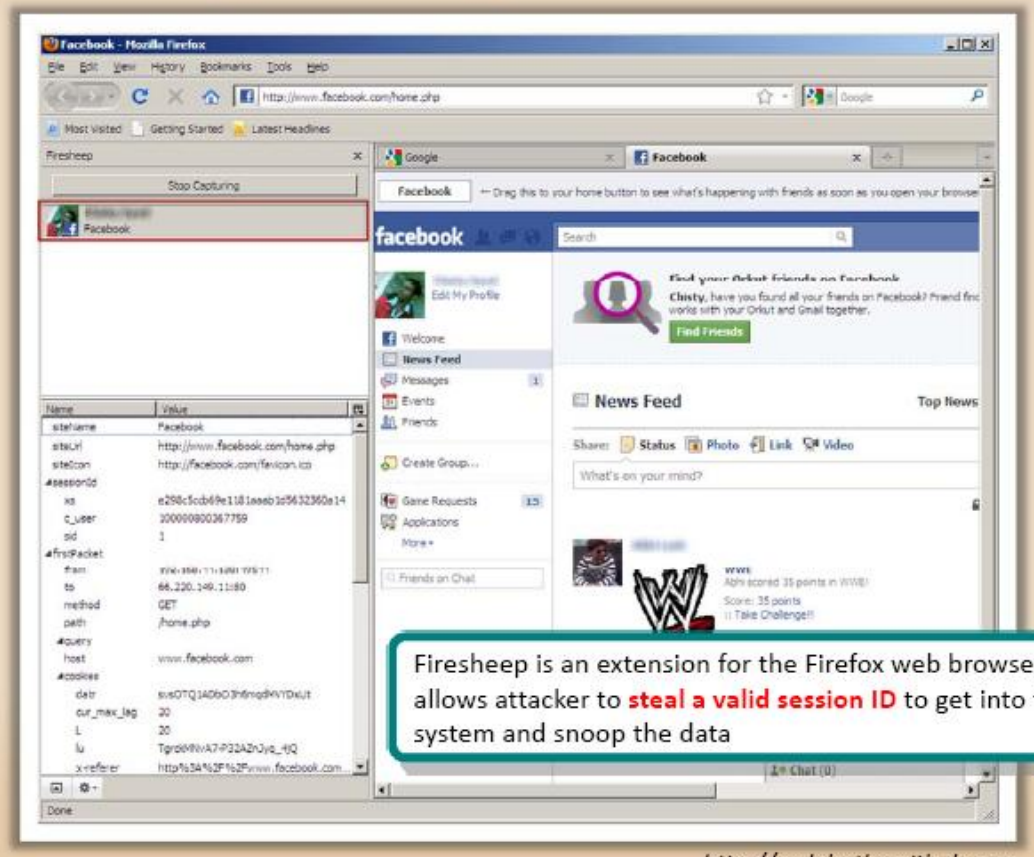
Session Hijacking Tool: Burp Suite

- Burp suite allows attacker to **inspect and modify traffic** between browser and the target application
- It **analyzes** all kinds of content, with automatic colorizing of request and response syntax



http://portswigger.net

Session Hijacking Tool: Firesheep



http://codebutler.github.com

Session Hijacking Tools

Hamster
http://hamster.erratasec.com

Hunt
http://packetstormsecurity.org

Session Thief
http://scriptjunkie1.110mb.com

JHijack
http://jhijack.sourceforge.net

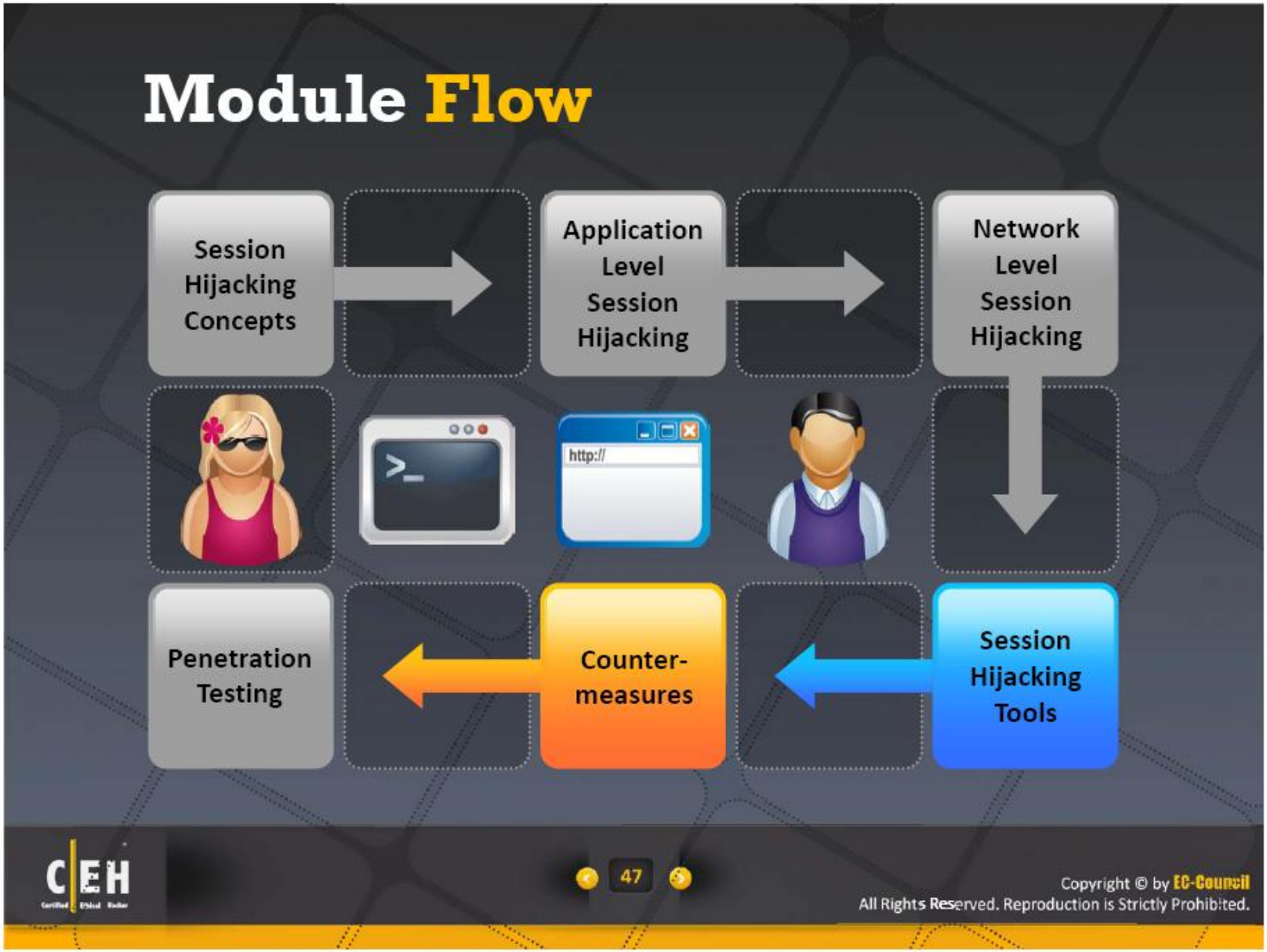
Surf Jack
http://surfjack.googlecode.com

TamperIE
http://www.bayden.com

Ettercap
http://ettercap.sourceforge.net

Ferret
http://www.erratasec.com

Module Flow


Session Hijacking Concepts
Application Level Session Hijacking
Network Level Session Hijacking
http://
Session Hijacking Tools
Counter-measures
Penetration Testing
47
Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Countermeasures

- Use secure shell (SSL) to create a secure communication channel
- Pass the authentication cookies over HTTPS connection
- Implement the logout functionality for user to end the session
- Generate the session ID after successful login
- Use string or long random number as a session key
- Pass the encrypted data between the users and the webservers


Protecting against Session Hijacking
101010
010101
101
Use encryption
Minimize remote access
Use a secure protocol
Educate the employees
Limit incoming connections
Regenerate the session ID after login
http://
CEH
49
Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Methods to Prevent Session Hijacking:
To be Followed by Web Developers

- Reduce the life span of a session or a cookie
- Create session keys with lengthy strings or random number so that it is difficult for an attacker to guess a valid session key
- Expire the session as soon as the user logs out
- Encrypt the data and session key that is transferred between the user and the web servers
- Regenerate the session id after a successful login to prevent session fixation attack
- Prevent Eavesdropping within the network

Methods to Prevent Session Hijacking:
To be Followed by Web Users

- Do not click on the links that are received through **mails or IM's**
- Use Firewalls to prevent the **malicious content** from entering the network
- Use firewall and browser settings to **restrict cookies**
- Make sure that the website is certified by the **certifying authorities**
- Make sure you clear **history, offline content, and cookies** from your browser after every confidential and sensitive transaction
- Prefer https, a secure transmission, rather than http when transmitting **sensitive and confidential data**
- Logout from the browser by **clicking on logout** button instead of closing the browser

Defending against Session Hijack Attacks

Use encrypted protocols that are available at OpenSSH suite

Use strong authentication (like Kerberos) or peer-to-peer VPN's

Use IDS products or ARPwatch for monitoring ARP cache poisoning

Configure the appropriate internal and external spoof rules on gateways

Session Hijacking Remediation

1

Defense in depth is a key component of a **comprehensive security plan**

2

Defense in depth is also a key component in **protecting a network** from session hijack attacks

Defense in depth is defined as the practice of using **multiple security systems or technologies** to prevent network intrusions

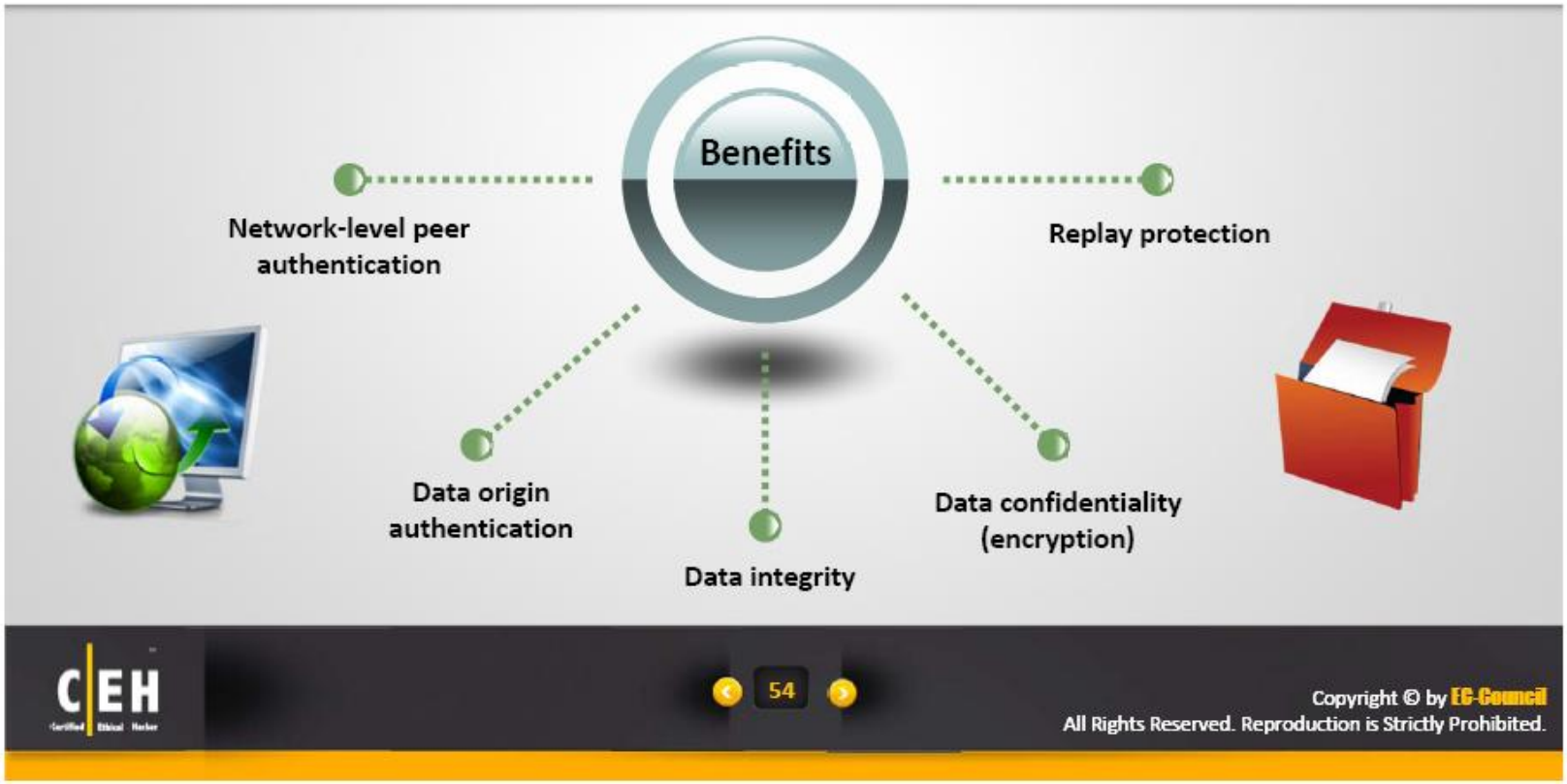
3

The central idea behind the concept is that if one counter measure fails, there are additional levels of protection to **safeguard the network**

4

IPSec

- IPSec is a set of protocols developed by the IETF to support the secure exchange of packets at the IP layer
- It is deployed widely to implement Virtual Private Networks (VPNs)



Modes of IPSec

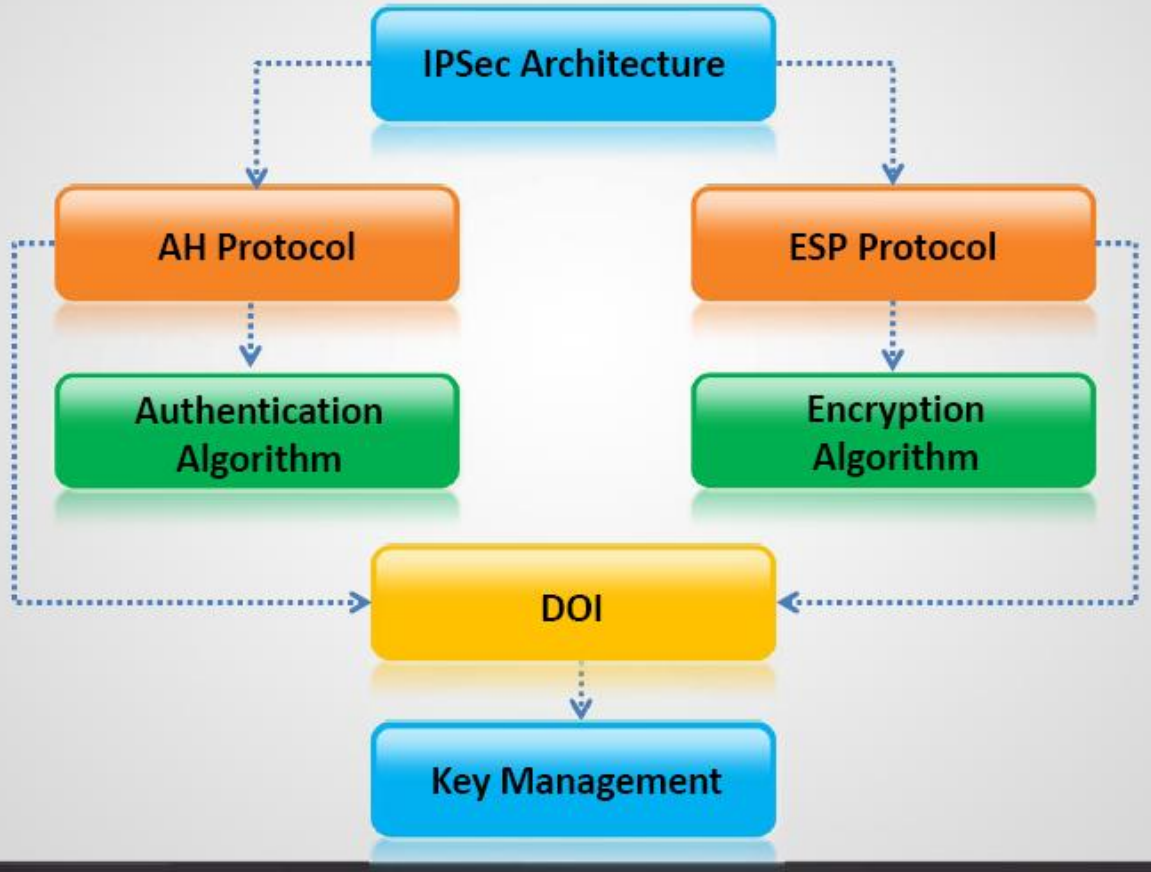
Transport Mode

- **Authenticates** two connected computers
- Has an option to **encrypt data transfer**
- Compatible with **NAT**

Tunnel Mode

- **Encapsulates** packets being transferred
- Has an option to **encrypt data transfer**
- Not compatible with **NAT**

IPSec Architecture



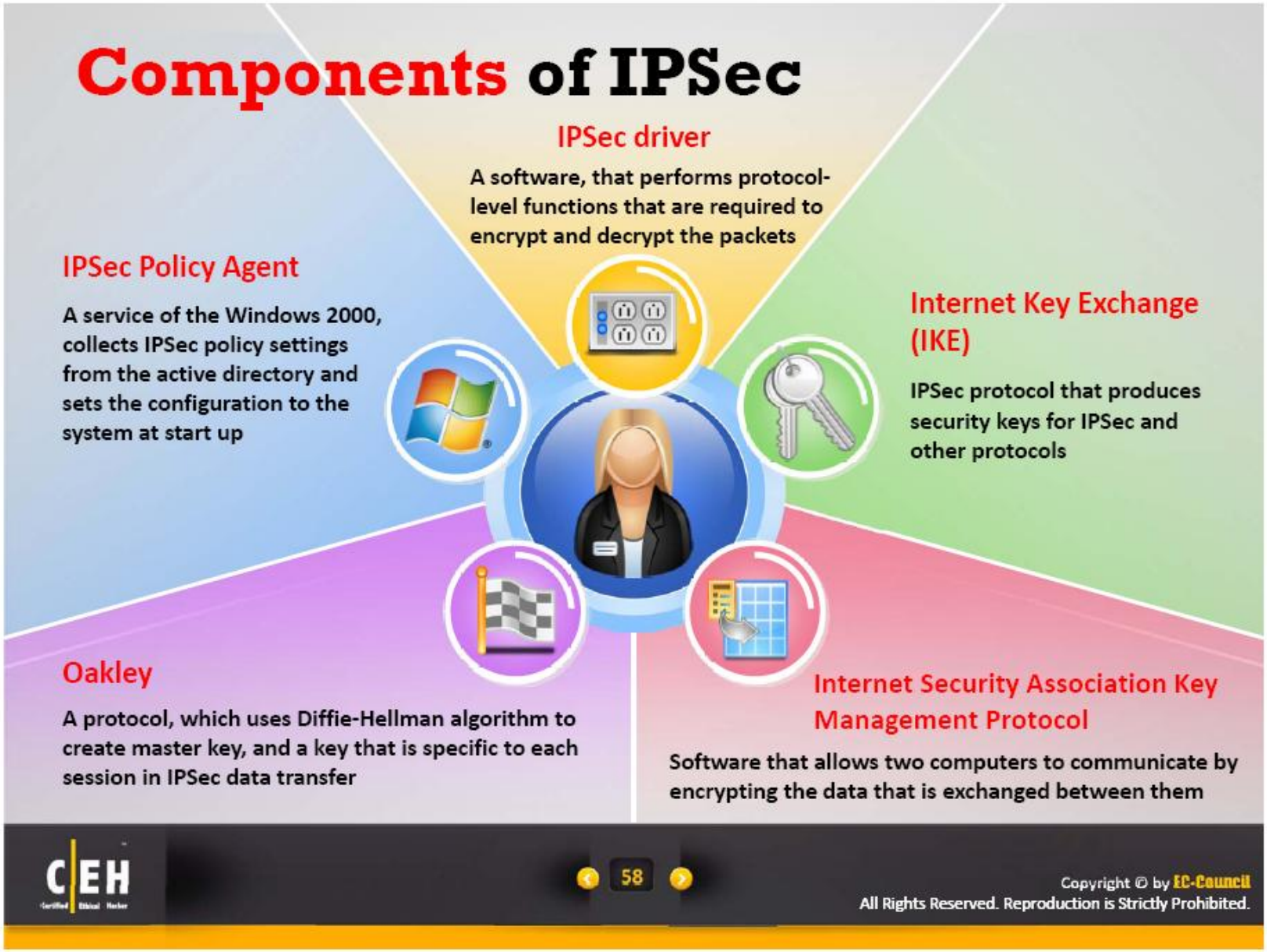
IPSec Authentication and Confidentiality

IPSec uses two different security services for authentication and confidentiality

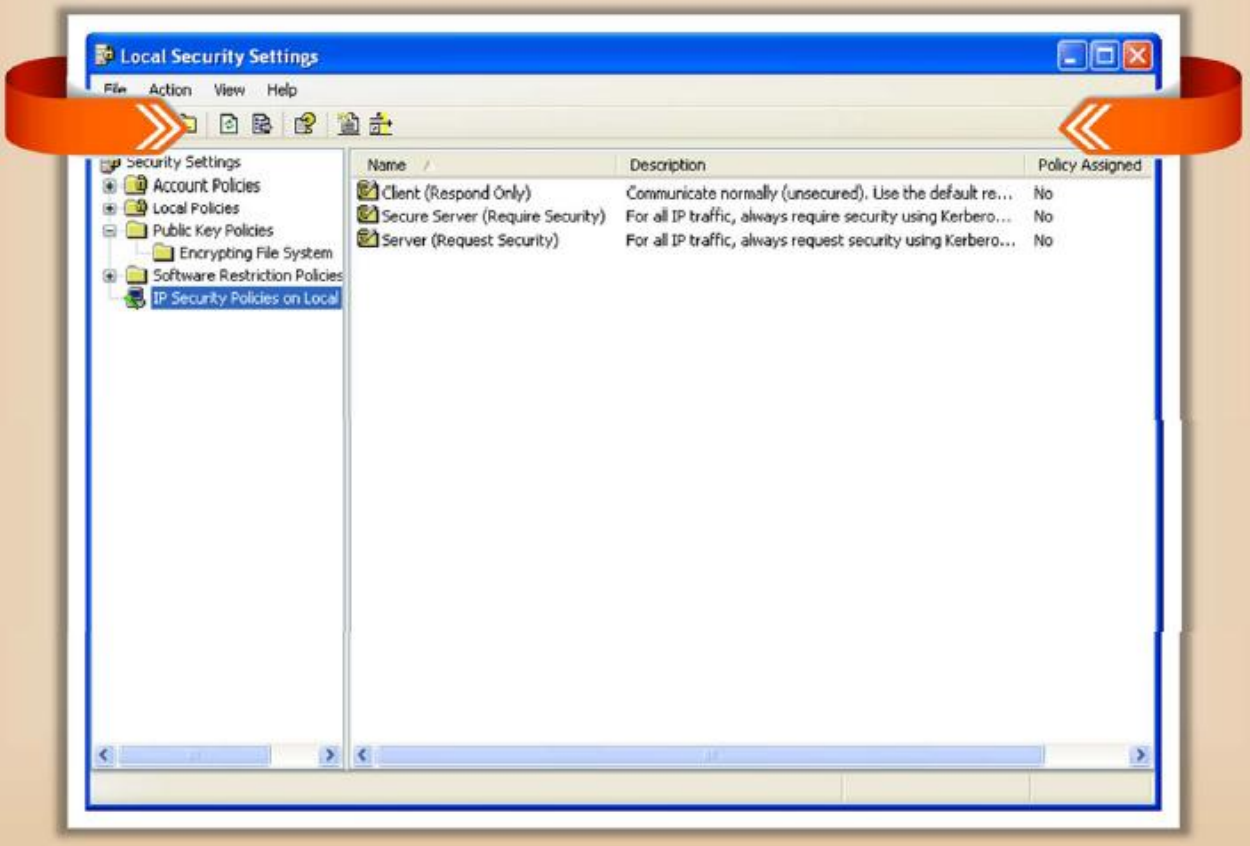
- Authentication Header (AH)
- Encapsulation Security Payload (ESP)

1. Authentication Header (AH) provides data authentication of the sender
2. Encapsulation Security Payload (ESP) provides both data authentication and encryption (confidentiality) of the sender

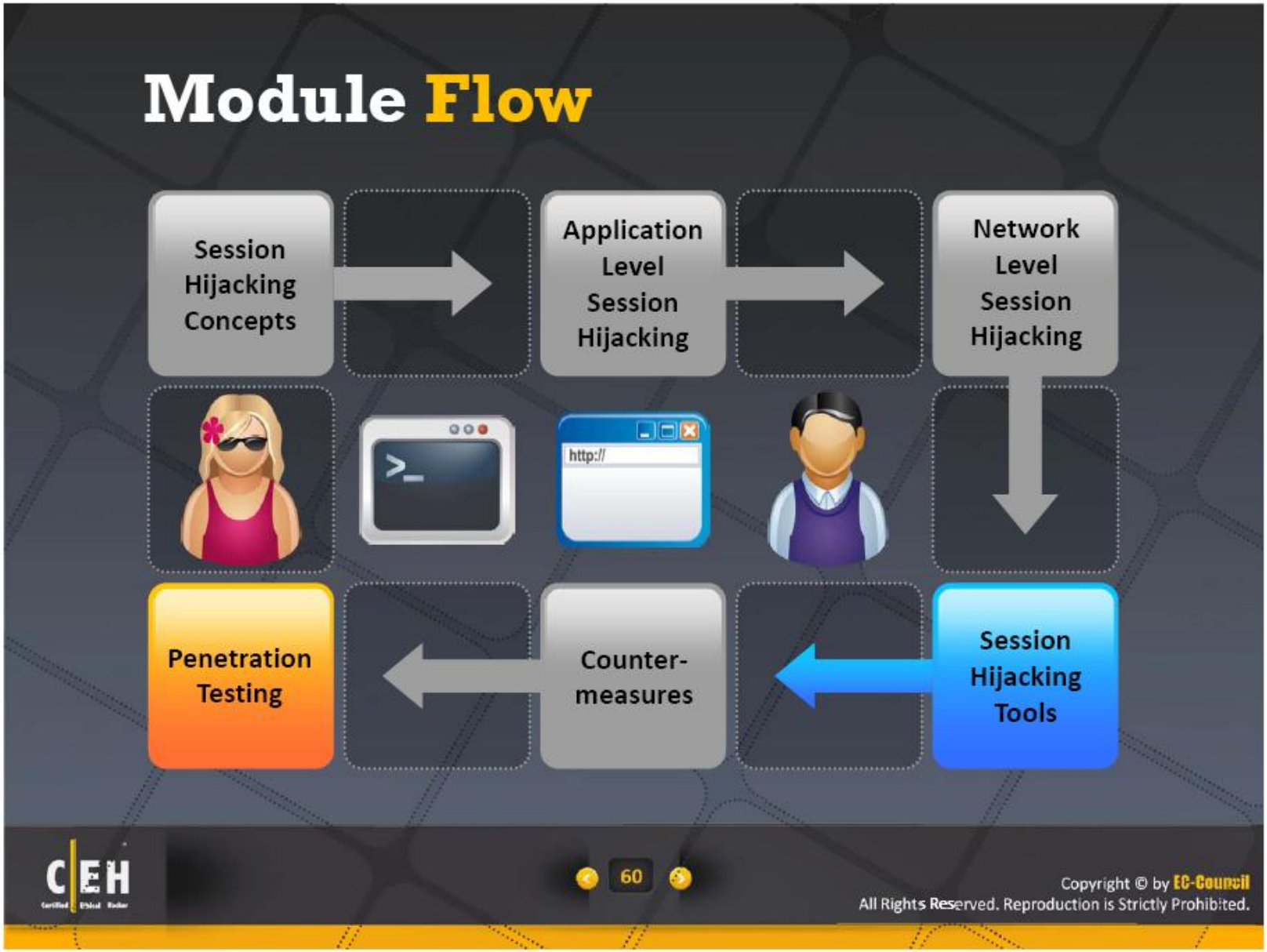
Components of IPSec

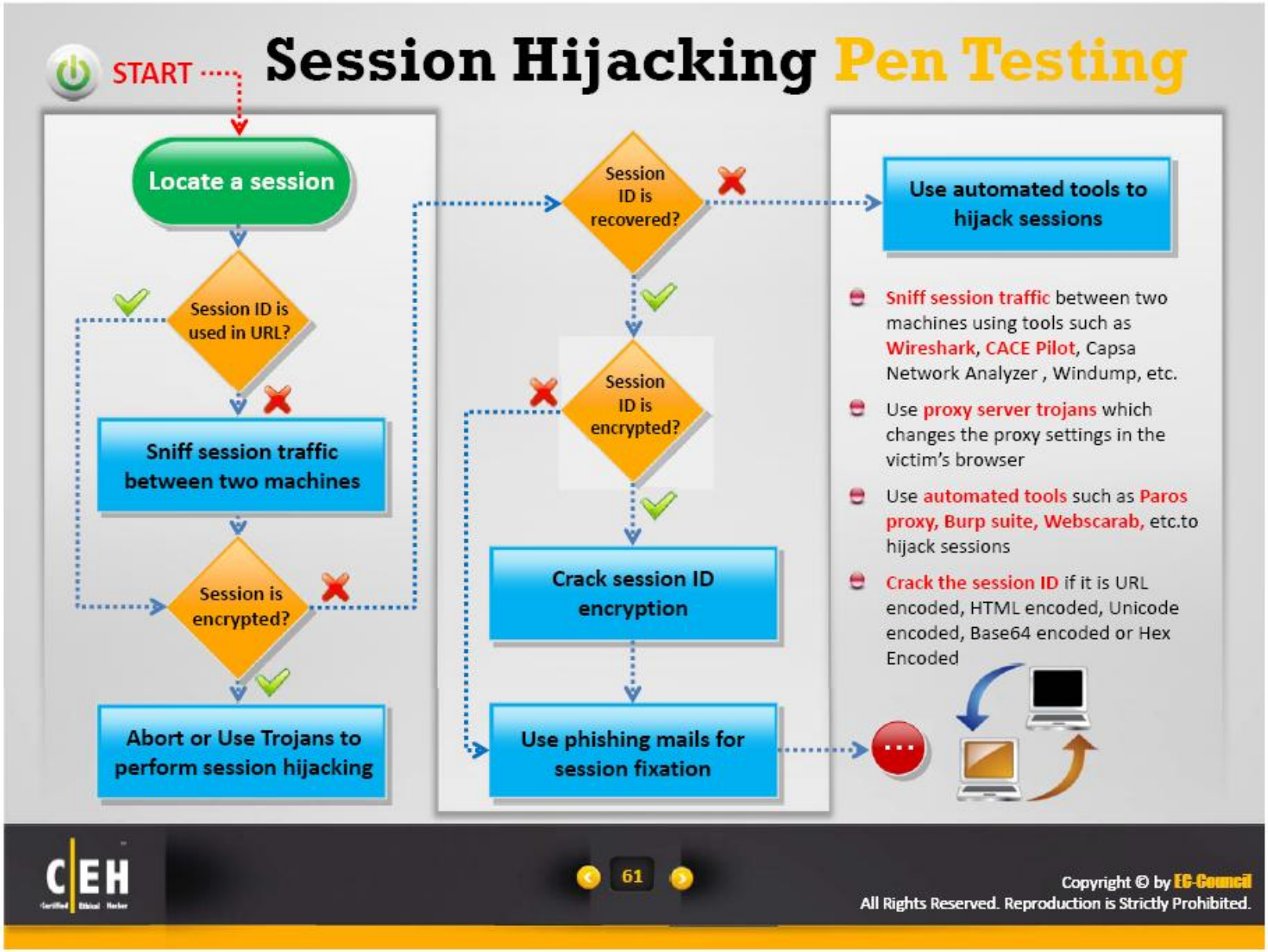


IPSec Implementation

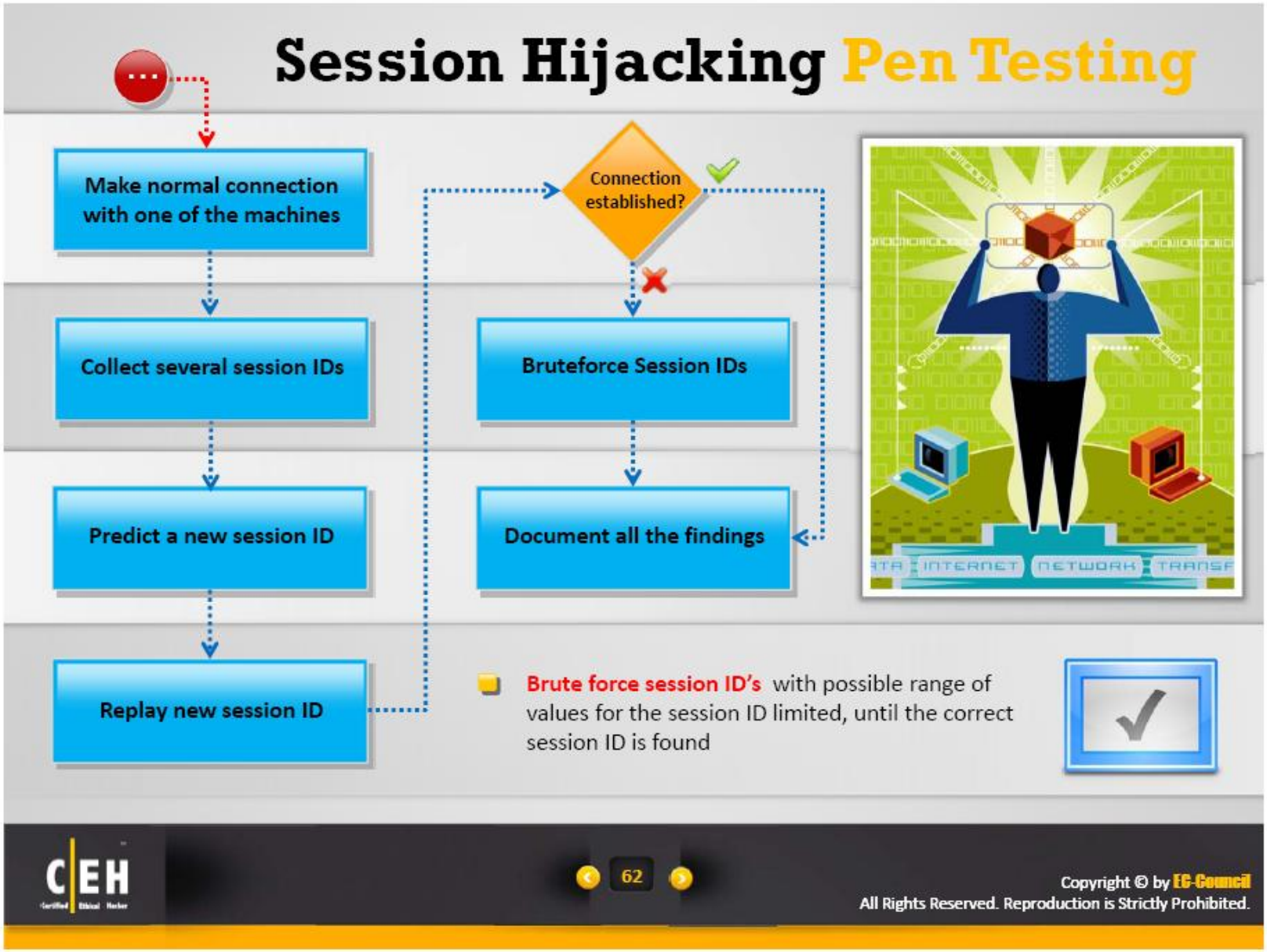


Module Flow


Session Hijacking Concepts
Application Level Session Hijacking
Network Level Session Hijacking
Session Hijacking Tools
Counter-measures
Penetration Testing
http://
60
Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.


START
Session Hijacking Pen Testing
Locate a session
Session ID is used in URL?
Sniff session traffic between two machines
Session is encrypted?
Abort or Use Trojans to perform session hijacking
Session ID is recovered?
Session ID is encrypted?
Crack session ID encryption
Use phishing mails for session fixation
Use automated tools to hijack sessions
Sniff session traffic between two machines using tools such as Wireshark, CACE Pilot, Capsa Network Analyzer , Windump, etc.
Use proxy server trojans which changes the proxy settings in the victim's browser
Use automated tools such as Paros proxy, Burp suite, Webscarab, etc.to hijack sessions
Crack the session ID if it is URL encoded, HTML encoded, Unicode encoded, Base64 encoded or Hex Encoded
C|EH
61
Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Session Hijacking Pen Testing



Module Summary

- In session hijacking, an attacker relies on the legitimate user to connect and authenticate, and will then take over the session
- In a spoofing attack, the attacker pretends to be another user or machine to gain access
- Successful session hijacking is difficult and is only possible when a number of factors are under the attacker's control
- Session hijacking can be active or passive in nature depending on the degree of involvement of the attacker
- A variety of tools exist to aid the attacker in perpetrating a session hijack
- Session hijacking could be dangerous, and therefore, there is a need for implementing strict countermeasures

Quotes

"Being able to break security doesn't make you a hacker any more than being able to hotwire cars makes you an automotive engineer."

- Eric Raymond,
An author and open source software advocate