UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Fakulti Teknologi Maklumat dan Komunikasi

Network Security Administration and Management

BITS 3353

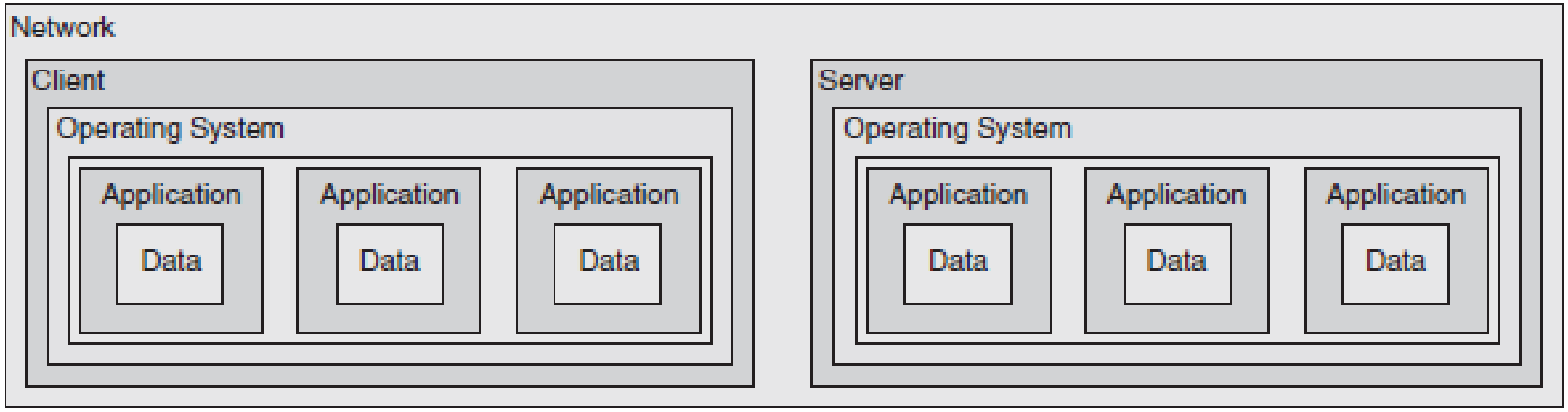
Lecture 5: Application and Network Attacks

Objectives

- List and explain the different types of Web application attacks
- Define client-side attacks
- Explain how a buffer overflow attack works
- List different types of denial of service attacks
- Describe interception and poisoning attacks

Application Attack

- A *network* is used to connect different *clients* and *servers* together.



Conceptual networked computer system

Attacks

WHAT?

any attempt to expose, alter, disable, destroy, steal or gain unauthorized access to or make unauthorized use of an asset.

APPLICATION ATTACKS (Attacks that target applications)

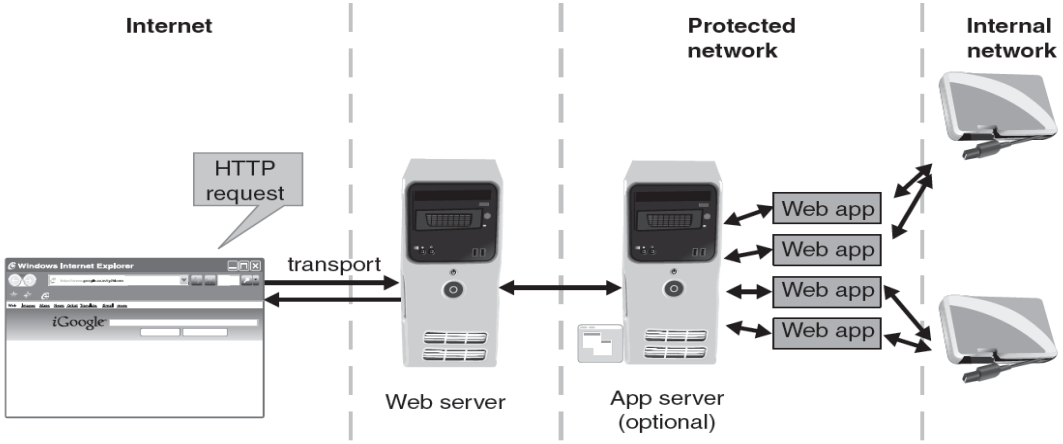
1. Server-side Web application attacks
2. Client-side attacks
3. Buffer overflow attacks

NETWORK ATTACKS

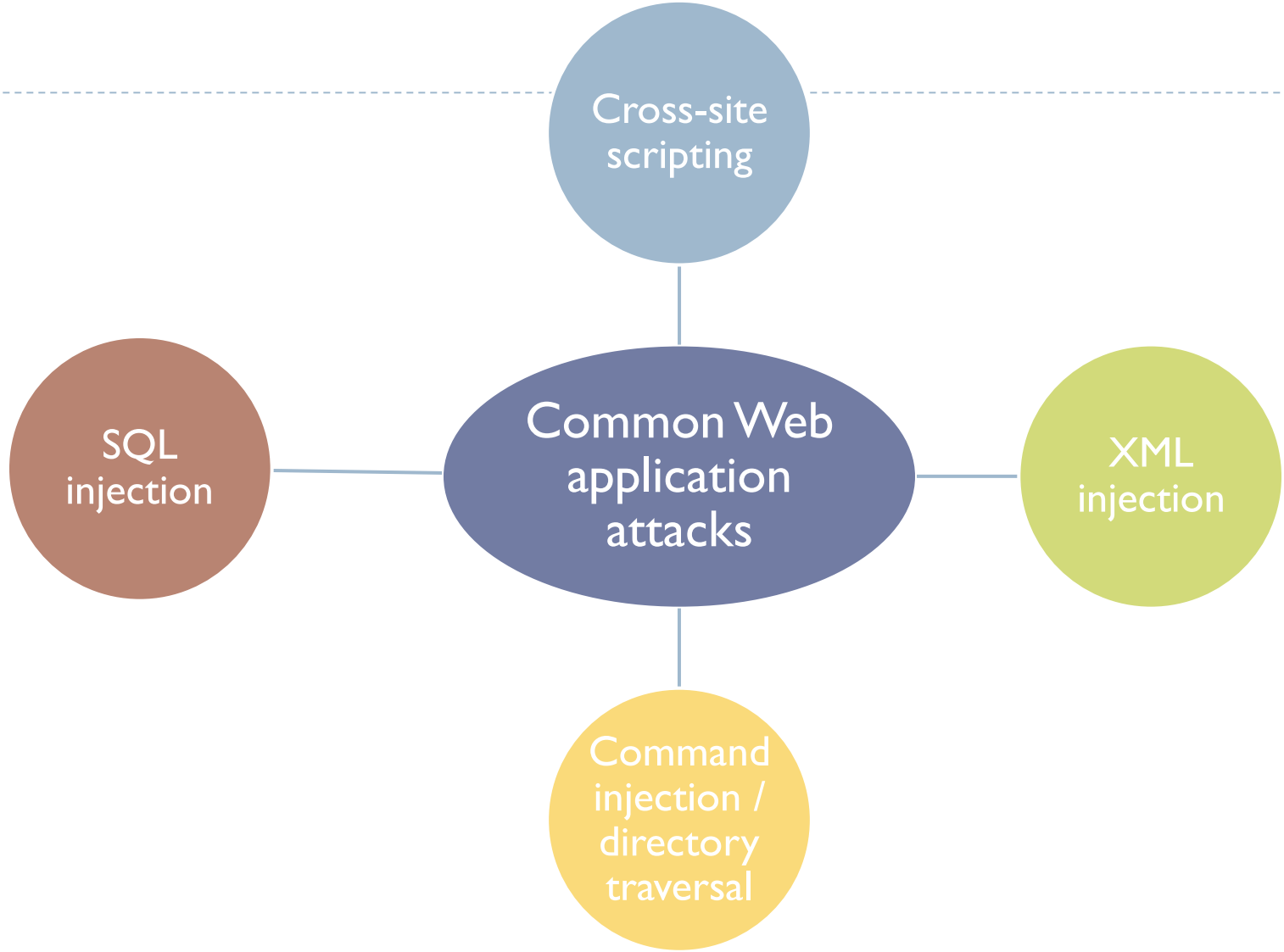
1. Denial of service (DoS)
2. Distributed denial of services (DDoS)

Web Application Attacks

- Web application is an application that runs on a web server and users access it using a web browser.
- Any security loop hole in browser will lead to exploiting vulnerabilities in web application.
- Approach to securing Web applications
  - –Hardening the Web server
  - –Protecting the network

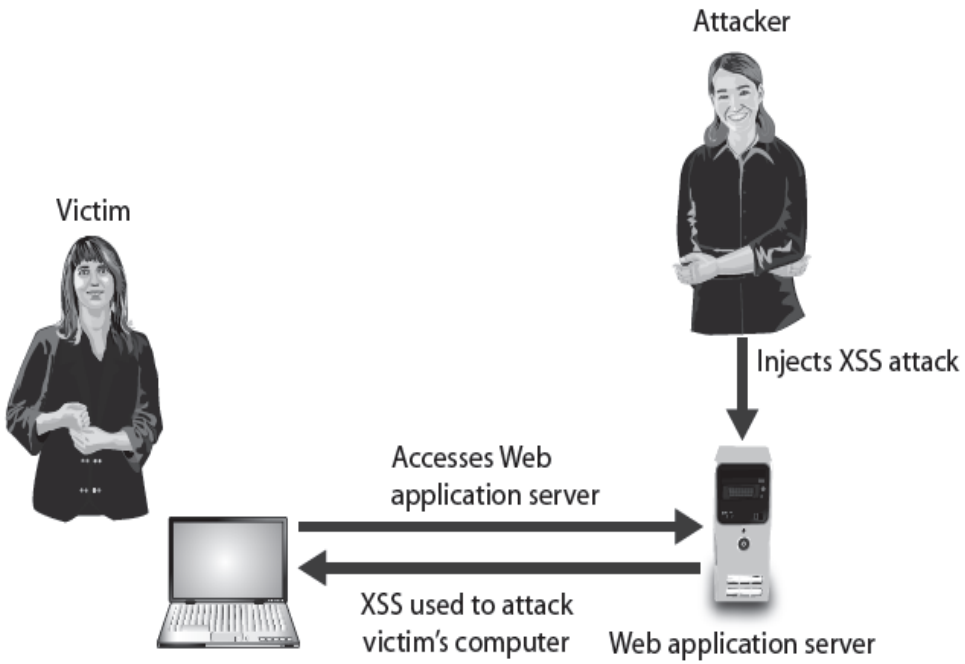


Web application infrastructure


Cross-site scripting
SQL injection
Common Web application attacks
XML injection
Command injection / directory traversal

Web application attacks:
1. Cross-Site Scripting (XSS)

- Injecting scripts into a Web application server
  - –Directs attacks at clients



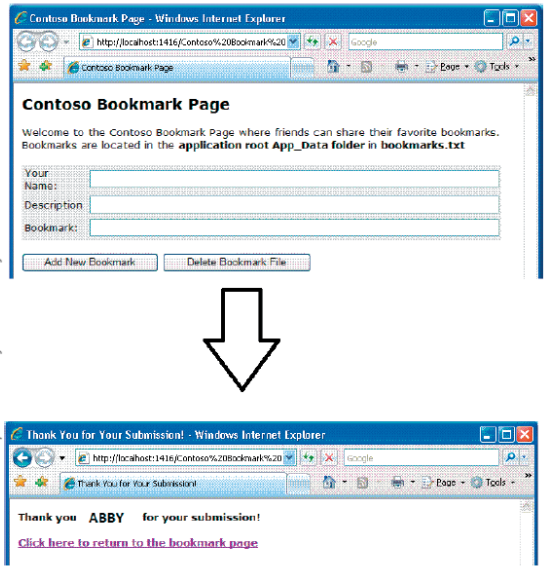
XSS attacks

When victim visits injected Web site:

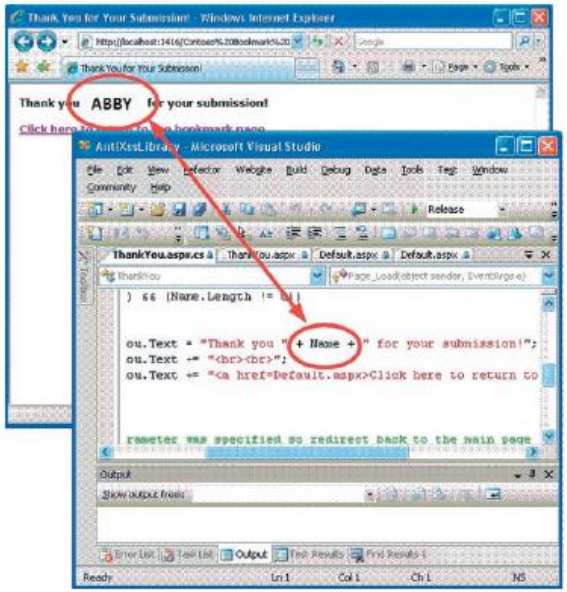
–Malicious instructions sent to victim's browser

Browser cannot distinguish between valid code and malicious script

Web application attacks:
1. Cross-Site Scripting (XSS)



Source: Used with permissions from Microsoft



Source: Used with permissions from Microsoft

Exploits applications that echo raw, unfiltered input to Web pages

- Input from <form> fields
- Input from query strings

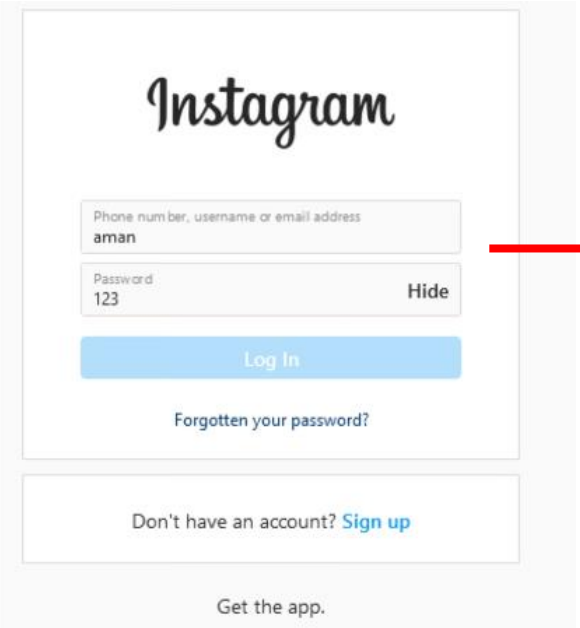
The technique:

- Find a <form> field or query string parameter whose value is echoed to the Web page
- Enter malicious script and get an unwary user to navigate to the infected page

Web application attacks:
2. SQL Injection

SQL (Structured Query Language)

–Used to manipulate data stored in relational database



select * from users where username='aman' and password='123';

SQL query TRUE

Successful LOGIN

```
MariaDB [haleluya]> select * from users;
+---------+-----------+-------------+
| user_id | username  | password    |
+---------+-----------+-------------+
|       1 | aman      | 123456      |
|       2 | stranger  | ucantseeme  |
|       3 | silk_rest | soobscribe  |
|       4 | shubham   | Imgrate     |
|       5 | somanya   | unHek8b0    |
+---------+-----------+-------------+
5 rows in set (0.000 sec)

MariaDB [haleluya]> select * from users where username="aman" and password="123456";
```

```
+---------+----------+----------+
| user_id | username | password |
+---------+----------+----------+
|       1 | aman     | 123456   |
+---------+----------+----------+
1 row in set (0.000 sec)
```

```
MariaDB [haleluya]> select * from users where username="aman" and password="1234532";
Empty set (0.000 sec)

MariaDB [haleluya]>
```

```
MariaDB [haleluya]> select * from users where username="aman" or password="1234532";
+---------+----------+----------+
| user_id | username | password |
+---------+----------+----------+
|       1 | aman     | 123456   |
+---------+----------+----------+
1 row in set (0.000 sec)
```

https://juice-shop.herokuapp.com/#/login

Username: ' or '1'='1'; --
Password:

SQL injection statement	Result
'whatever' AND email IS NULL;	Determine the names of different fields in the database
'whatever' AND 1= (SELECT COUNT(*) FROM tabname);	Discover the name of the table
'whatever' OR full name LIKE '%Mia%';	Find specific users
'whatever'; DROP TABLE members;	Erase the database table
'whatever'; UPDATE members SET email = ' attacker-email@ evil.net' *WHERE email =* ' Mia@good.com';	Mail password to attacker's email account

SQL injection statements

Web application attacks:
3. XML Injection

XML (eXtensible Markup Language)

- defines a set of rules for encoding documents in a format that is both human-readable and machine-readable
- Used to store and transport data.

How

injecting command

- Similar to SQL injection attack
- Attacker discovers Web site that does not filter user data
- Injects XML tags and data into the database

Web application attacks:
3. XML Injection

- attacker manipulates an XML (eXtensible Markup Language) document to gain unauthorized access, modify data, or perform other malicious actions.

The application's XML processing code might look like this:

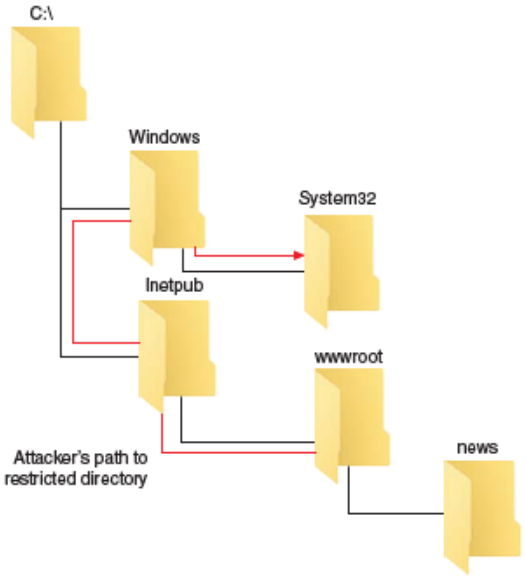
```xml
<product>
  <name>Product Name</name>
  <price>100</price>
</product>
```

A user submits the following XML data as part of a request:

```xml
<product>
  <name>Malicious Product</name>
  <price>999</price>
</product>
```

Web application attacks:
4. Command Injection / Directory Traversal

- HTTP attack which allows attackers to access restricted **directories** and execute commands outside of the web server's root **directory**.
- In this type of attack, an authenticated or unauthenticated user can request and view or execute files that they should not be able to access.
- A directory traversal uses malformed input or takes advantage of a vulnerability to move from the root directory to restricted directories. Once the attacker has accessed a restricted directory, she can enter (inject) commands to execute on a server (called command injection) or view confidential files.



In this case, "example.txt" is the name of the file the user wants to download.

```arduino
http://example.com/download?file=example.txt
```

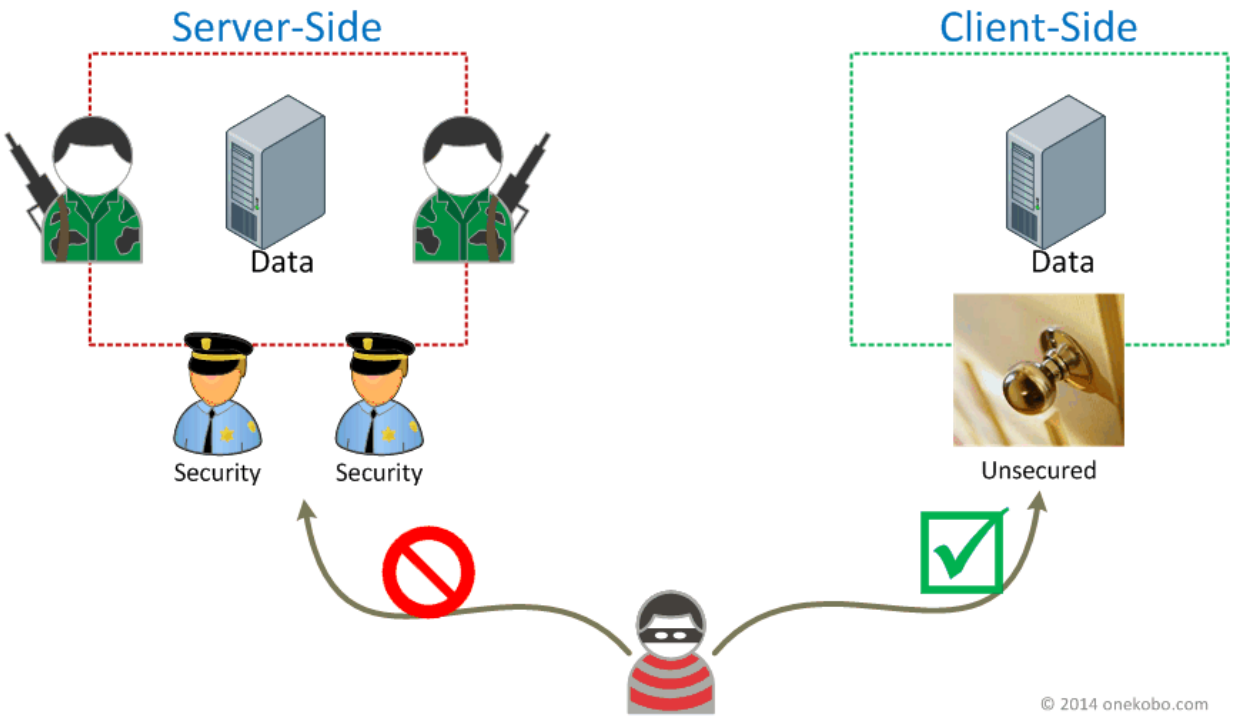
The application doesn't properly validate or restrict user input, and it constructs the file path based on the user-provided "file" parameter without adequate checks

The attacker is attempting to traverse the directory structure upwards and access the "/etc/passwd"

```bash
http://example.com/download?file=../../../../../etc/passwd
```

Client-Side Attacks

- Web application attacks are server-side attacks
- Client-side attacks target vulnerabilities in client applications



Client-Side Attacks

- A client can be a standard web browser such as Internet Explorer or Google Chrome or it can be an embedded browser object in an application such as an email client, media players, e-book reader, etc.

Header Manipulation

Header manipulation involves modifying HTTP headers sent by the client's web browser to deceive or exploit a web server.

Cookies

Cookies can be stolen and used to impersonate the user
Third-party cookies can be used to track the browsing or buying habits of a user

Session Hijacking

Attacker takes control of a user's active session on a website or application, typically by stealing session identifiers.

Malicious add-ons

Plug-in and add-ons or extensions manipulate and compromise user's browser experience, often with harmful intent

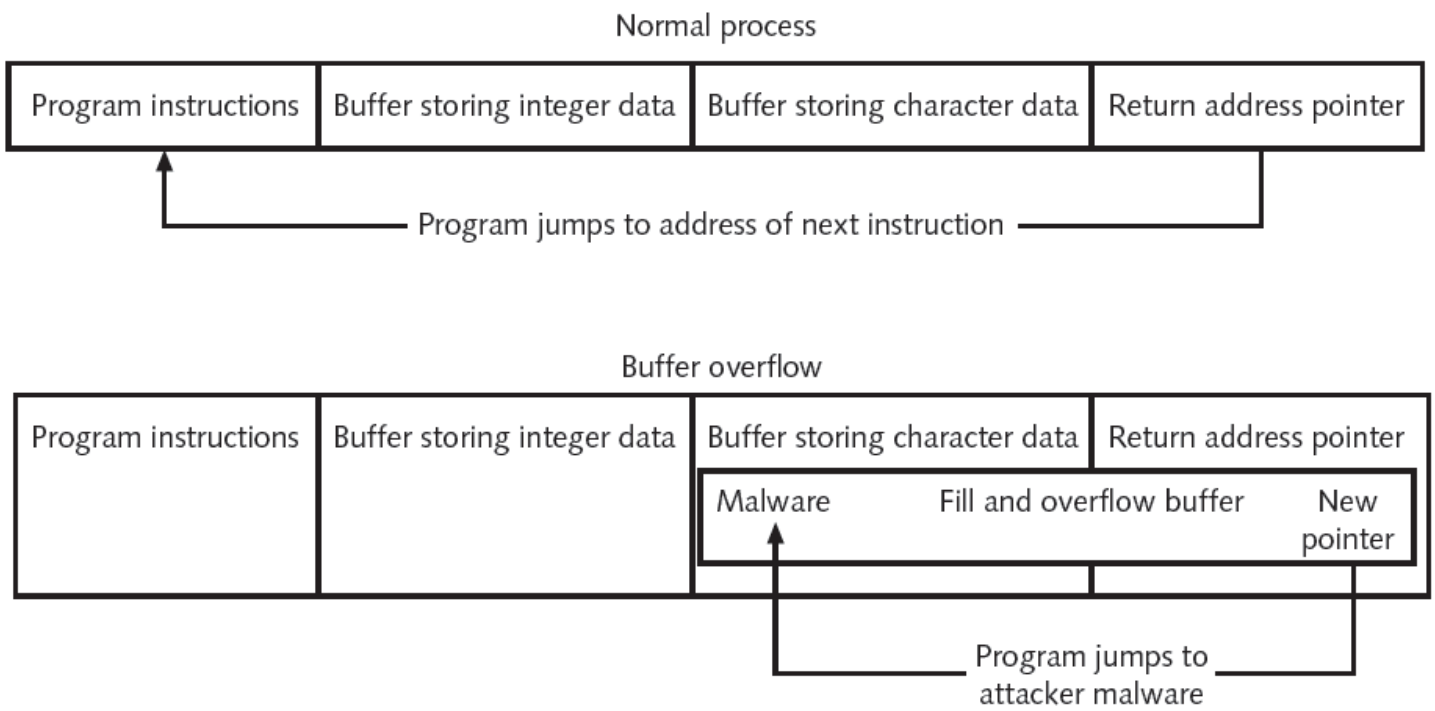
- Two reasons why client applications are more inviting targets for attackers include:
  1. Clients generally undergo less rigorous security testing than server applications
  2. Clients are more difficult to patch due to the diverse range of client versions, owners and environments

How it happen?

- When a user downloads malicious content
- They require user-interaction such as clicking a malicious link or running executable payload.

Client-Side Attacks: Buffer Overflow Attacks

Process attempts to store data in RAM beyond boundaries of fixed-length storage buffer → **Data overflows into adjacent memory locations** → **May cause computer to stop functioning**

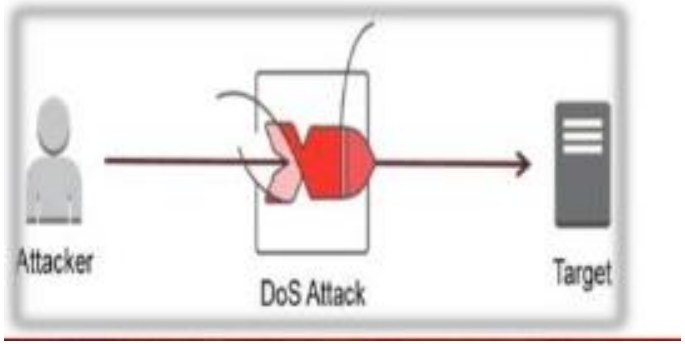


Buffer overflow attack

Network Attacks

- Attempts to prevent system from performing normal functions

- In a denial-of-service (DoS) attack, an attacker attempts to prevent legitimate users from accessing information or services.



- In a distributed denial-of-service (DDoS) attack, an attacker may use your computer to attack another computer.

Network Attacks

Denial of service (DoS)

- Attempts to prevent system from performing normal functions
- Ping flood attack
  - -Ping utility used to send large number of echo request messages
  - -Overwhelms Web server
- Smurf attack
  - -Ping request with originating address changed
  - -Appears as if target computer is asking for response from all computers on the network
- SYN flood attack
  - -Takes advantage of procedures for establishing a connection

Distributed denial of service (DDoS)

- Attacker uses many zombie computers in a botnet to flood a device with requests
- Virtually impossible to identify and block source of attack

DoS ATTACK VERSUS DDoS ATTACK

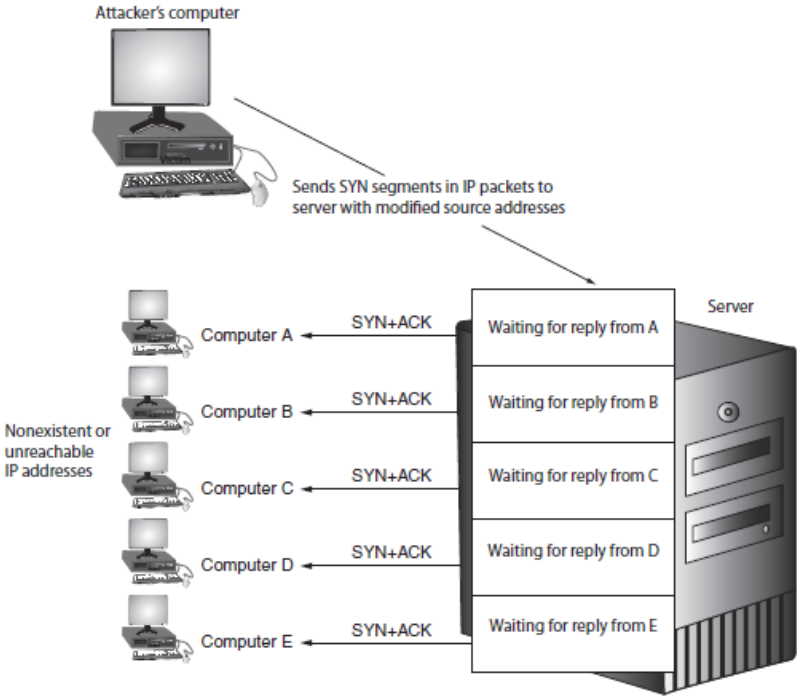
DoS ATTACK	DDoS ATTACK
A cyber-attack in which the perpetrator seeks to make a machine or network resource unavailable to its intended users by temporarily or indefinitely disrupting services of a host connected to the internet	A cyber-attack in which the incoming traffic flooding the victim originates from many different sources
Stands for Denial of Service	Stands for Distributed Denial of Service
A single machine is used to launch an attack	Multiple machines are used to launch an attack
Comparatively less complicated	More complicated and difficult to prevent
There is no malware involvement	Uses malware to affect multiple machines

Visit www.PEDIAA.com

Network Attacks

SYN Flood

- Takes advantage of the procedures for initiating a session

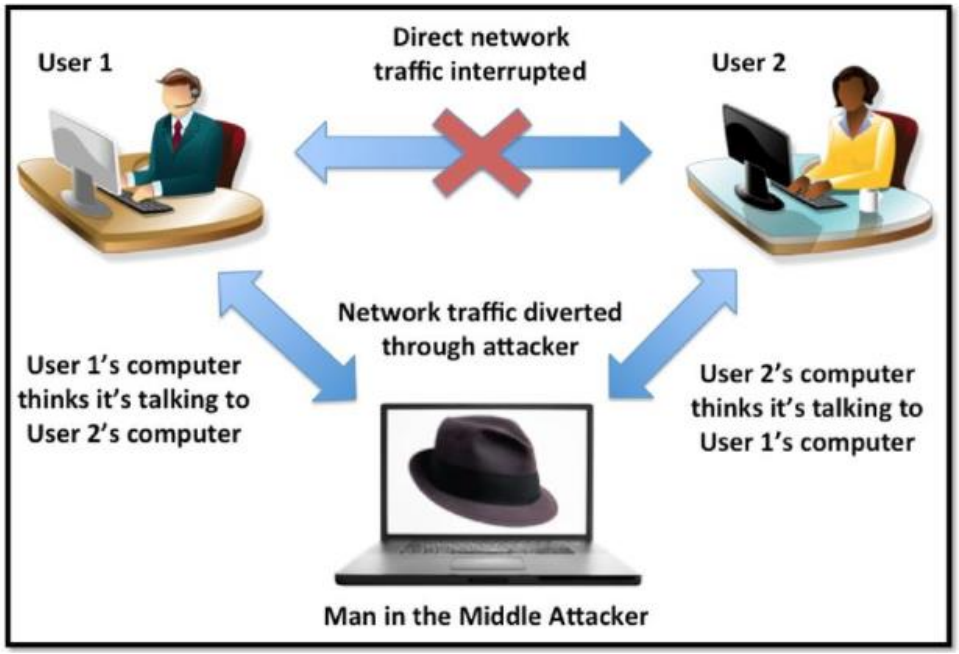


SYN flood attack

Interception

Man-in-the-middle

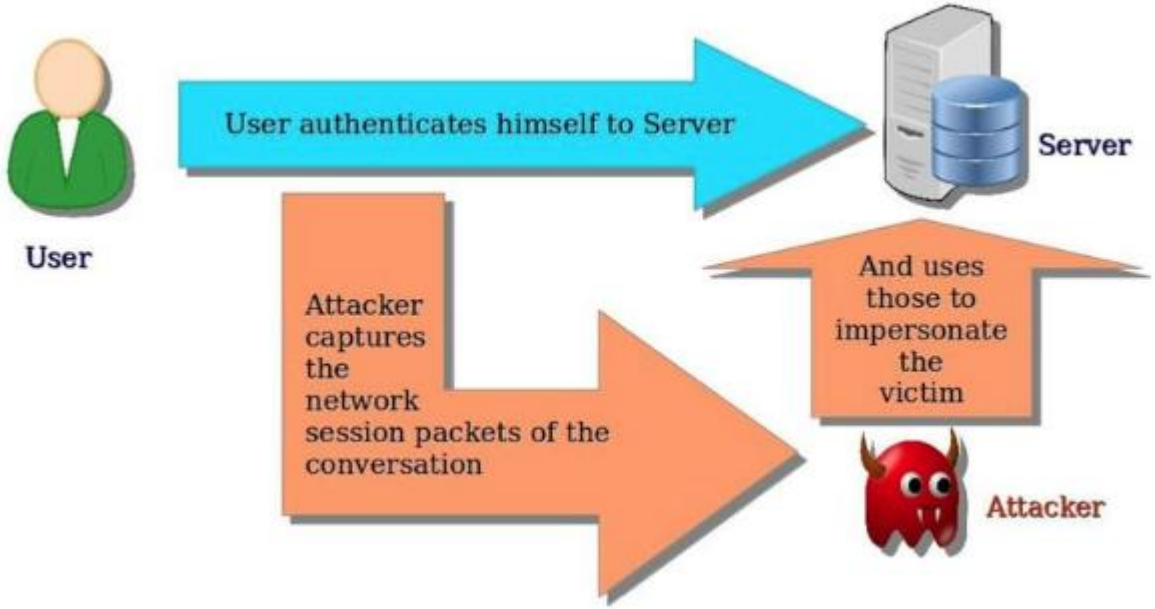
- –Interception of legitimate communication
- –Forging a fictitious response to the sender
- –Passive attack records transmitted data
- –Active attack alters contents of transmission before sending to recipient



Interception

Replay Attack

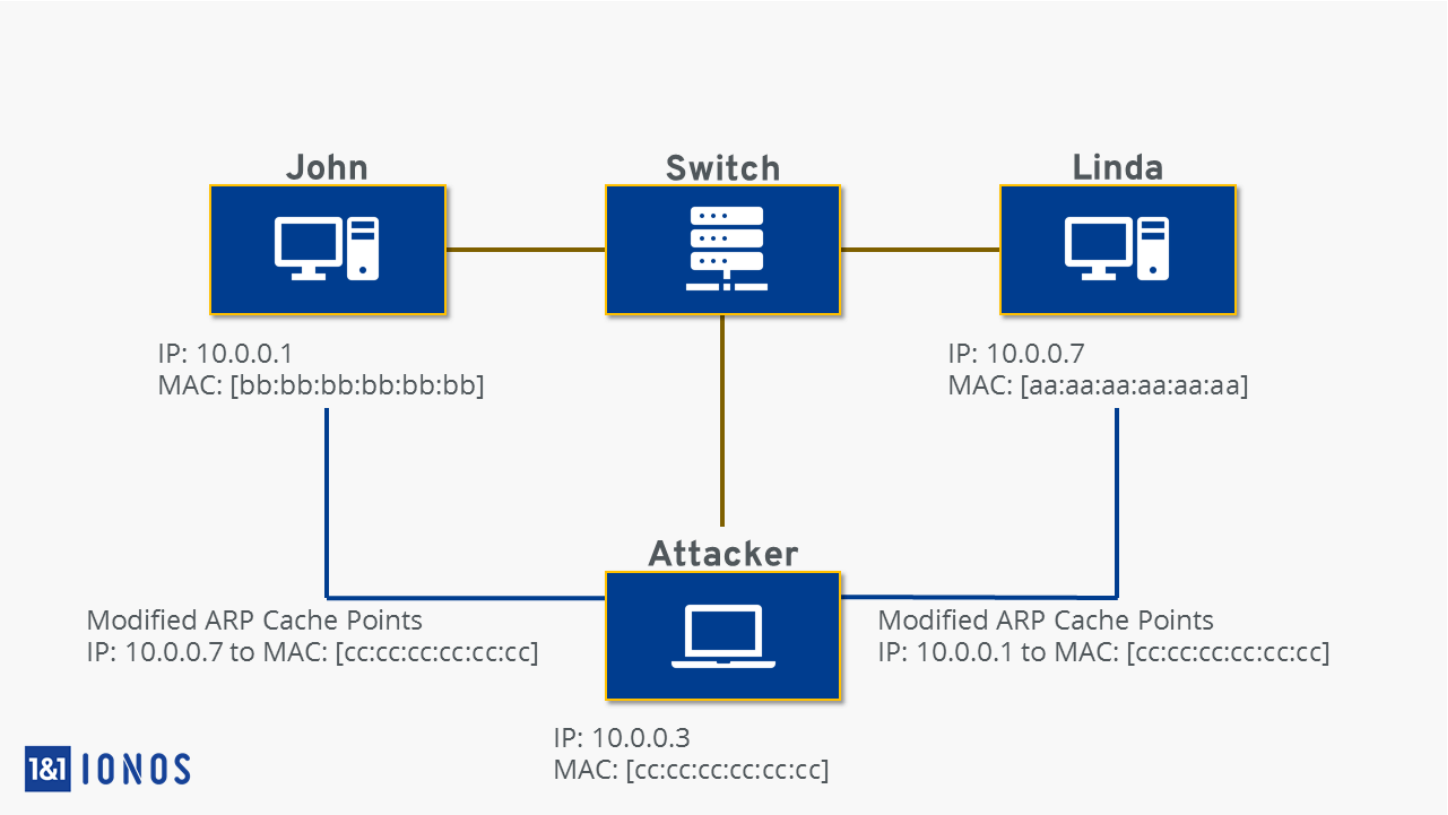
- –Hacker sniffs packets to get authentication info
- –Then hacker uses info to connect to server



Poisoning

ARP Poisoning

–Attacker modifies MAC address in ARP cache to point to different computer



Poisoning

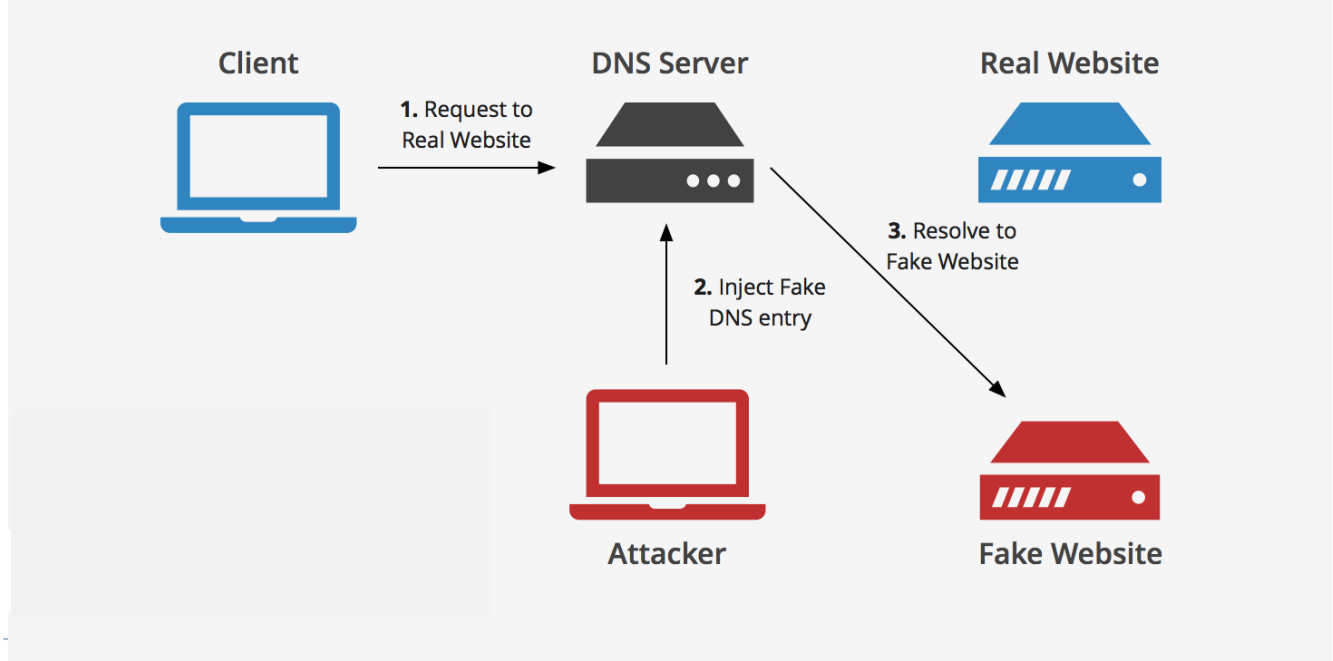
Attack	Description
Steal data	An attacker could substitute their own MAC address and steal data intended for another device
Prevent Internet access	An attacker could substitute an invalid MAC address for the network gateway so that no users could access external networks
Man-in-the-middle	A man-in-the-middle device could be set to receive all communications by substituting that MAC address
DoS attack	The valid IP address of the DoS target could be substituted with an invalid MAC address, causing all traffic destined for the target to fail

Attacks from ARP poisoning

Poisoning

DNS Poisoning

- Domain Name System is current basis for name resolution to IP address
- DNS poisoning substitutes DNS addresses to redirect computer to another device
- Two locations for DNS poisoning
  - –Local host table
  - –External DNS server



Attacks on Access Rights

- Privilege escalation
  - –Exploiting software vulnerability to gain access to restricted data
  - –Lower privilege user accesses functions restricted to higher privilege users
  - –User with restricted privilege accesses different restricted privilege of a similar user

- Transitive access
  - –Attack involving a third party to gain access rights
  - –Has to do with whose credentials should be used when accessing services
    - •Different users have different access rights

Summary

- Web application flaws are exploited through normal communication channels
- XSS attack uses Web sites that accept user input without validating it
  - Uses server to launch attacks on computers that access it
- Client-side attack targets vulnerabilities in client applications
  - Client interacts with compromised server
- Session hijacking
  - Attacker steals session token and impersonates user
- Buffer overflow attack
  - Attempts to compromise computer by pushing data into inappropriate memory locations
- Denial of service attack attempts to overwhelm system so that it cannot perform normal functions
- In ARP and DNS poisoning, valid addresses are replaced with fraudulent addresses
- Access rights and privileges may also be exploited