Types of attacks- II

Security countermeasure- Lecture 3

Information Security Department

2nd Class- Second Course

feb-2024

Learning Objectives

- HTTP Post DoS.
- **DDoS cont**. Configure a system to defend against buffer overflow attacks.
- Defending Against IP Spoofing.
- Defending Against Session Hijacking

HTTP Post DoS

- An HTTP Post DoS attack sends a legitimate HTTP post message.
  - Part of the post message is the ‘content-length’. This indicates the size of the message to follow.
  - The attacker sends legitimate HTTP POST headers to a Web server.
  - In these headers, the sizes of the message body that will follow are correctly specified.
  - However, the message body is sent at a painfully low speed.
  - These speeds may be as slow as one byte every two minutes.
  - Because the server keeps the connection open in anticipation of additional data, genuine users are prevented from accessing the server. The servers would appear to have a large number of connected clients but the actual processing load would be very low.

PDoS

- A permanent denial of service (PDoS) :
  - It is an attack that damages the system so badly that the victim machine needs either an operating system reinstallation or even new hardware. This is sometimes called phlashing.
  - This will usually involve a DoS attack on the device's firmware.

DoS (Denial of Service) and DDoS (Distributed Denial of Service)

- The main difference between DoS (Denial of Service) and DDoS (Distributed Denial of Service) lies in how they are executed and the scale of the attack.
- DoS (Denial of Service):

1. DoS attacks are typically carried out by a single malicious user or a small group of attackers.
2. In a DoS attack, the attacker overwhelms a targeted server or network with a flood of traffic, such as requests for information or connection attempts.
3. The goal of a DoS attack is to make a service or network unavailable to its intended users by consuming all available resources or by exploiting vulnerabilities in the system.

DoS (Denial of Service) and DDoS (Distributed Denial of Service)

- DDoS (Distributed Denial of Service):

1. DDoS attacks involve multiple sources or systems to orchestrate the attack.
2. These multiple sources are often compromised computers or devices, forming what is known as a botnet.
3. Each compromised device, under the control of the attacker, sends traffic to the target simultaneously, amplifying the impact and making it more difficult to mitigate.
4. DDoS attacks are typically much larger in scale and harder to defend against than DoS attacks because of the distributed nature of the attack.

Distributed Reflection Denial of Service

- The distributed reflection denial of service (DRDoS) attack is a special type of DoS attack.
- The attack works a bit differently than other DoS attacks. Rather than getting computers to attack the target, this method tricks Internet routers into attacking a target.
- What makes this attack particularly wicked is that it does not require the routers in question to be compromised in any way. The attacker does not need to get any sort of software on the router to get it to participate in the attack.
- Instead, the hacker sends a stream of packets to the various routers requesting a connection. The packets have been altered so that they appear to come from the target system's IP address. The routers respond by initiating connections with the target system.
- What occurs is a flood of connections from multiple routers, all hitting the same target system. This has the effect of rendering the target system unreachable.

Distributed Reflection Denial of Service

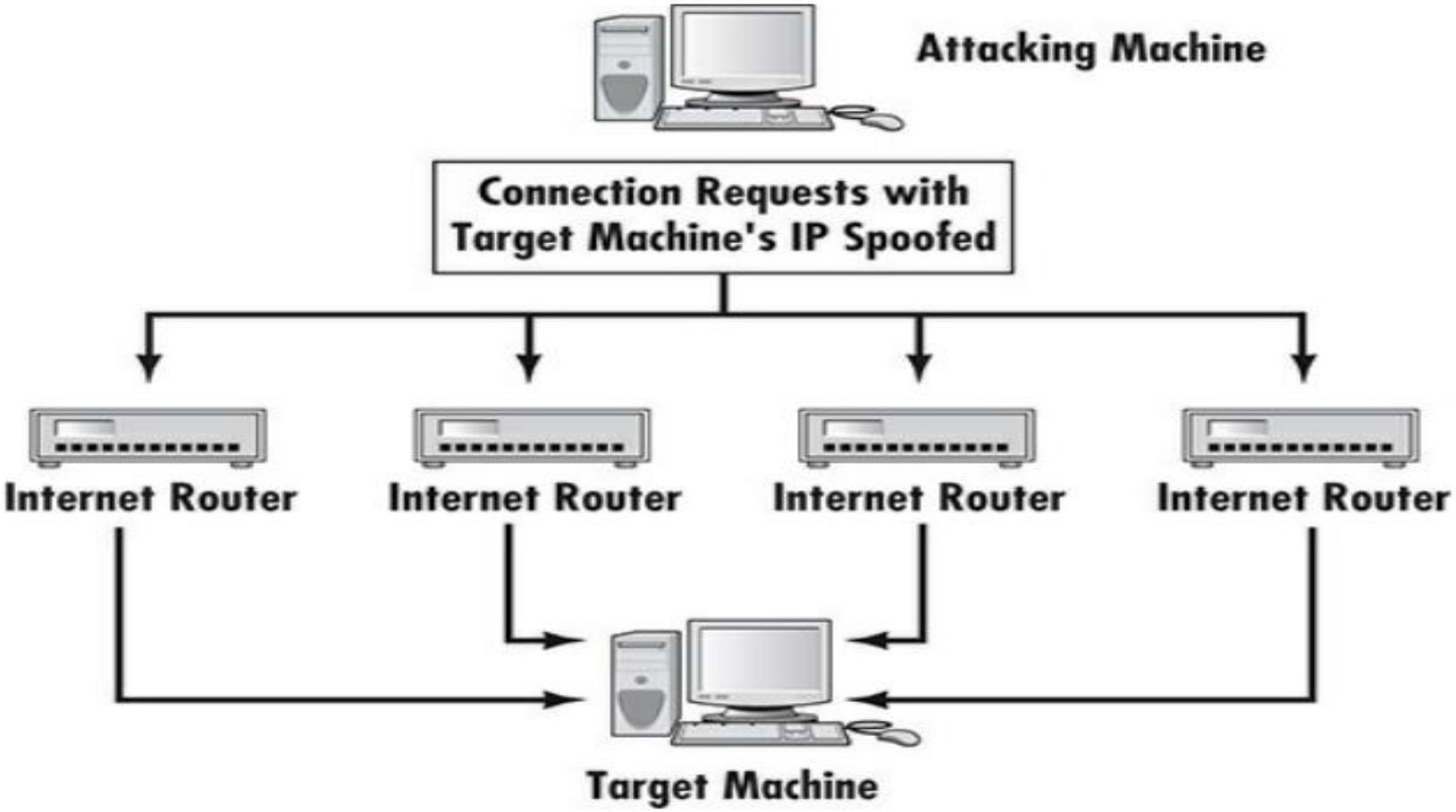


FIGURE 2-5 Distributed reflection denial of service

DoS Tools

- One reason that DoS attacks are becoming so common is that a number of tools are available for executing DoS attacks.
- These tools are widely available on the Internet, and in most cases are free to download.
- This means that any prudent administrator should be aware of them. In addition to their obvious use as an attack tool, they can also be useful for testing your anti-DoS security measures.

Buffer Overflow Attacks

- Buffer overflow attacks are a type of security vulnerability that occurs when a program or process tries to store more data in a buffer (temporary storage area) than it was intended to hold.
- This extra data can overwrite adjacent memory locations, causing unexpected behavior and potentially allowing an attacker to execute arbitrary code or take control of the affected system.
- Any program that communicates with the Internet or a private network must receive some data. This data is stored, at least temporarily, in a space in memory called a buffer.

Buffer Overflow Attacks

How buffer overflow attacks work:

1. Buffer: A buffer is a temporary storage area in a computer's memory used to hold data while it's being transferred between two locations or processed by a program. Buffers are often used in functions that handle input from users or external sources.
2. Overflow: A buffer overflow occurs when more data is written to a buffer than it can hold, causing the excess data to spill over into adjacent memory locations. This can corrupt the integrity of the program's execution or data structures.
3. Exploitation: In a buffer overflow attack, an attacker intentionally sends or injects malicious data into a program's buffer to trigger a buffer overflow. By carefully crafting the input data, the attacker can overwrite critical data structures, such as the function's return address or other variables, with their own malicious code or data.
4. Execution of Arbitrary Code: If the attacker successfully overwrites the program's execution flow with their own code, they can potentially execute arbitrary commands or instructions with the privileges of the compromised process. This could lead to unauthorized access, privilege escalation, or the execution of further attacks.

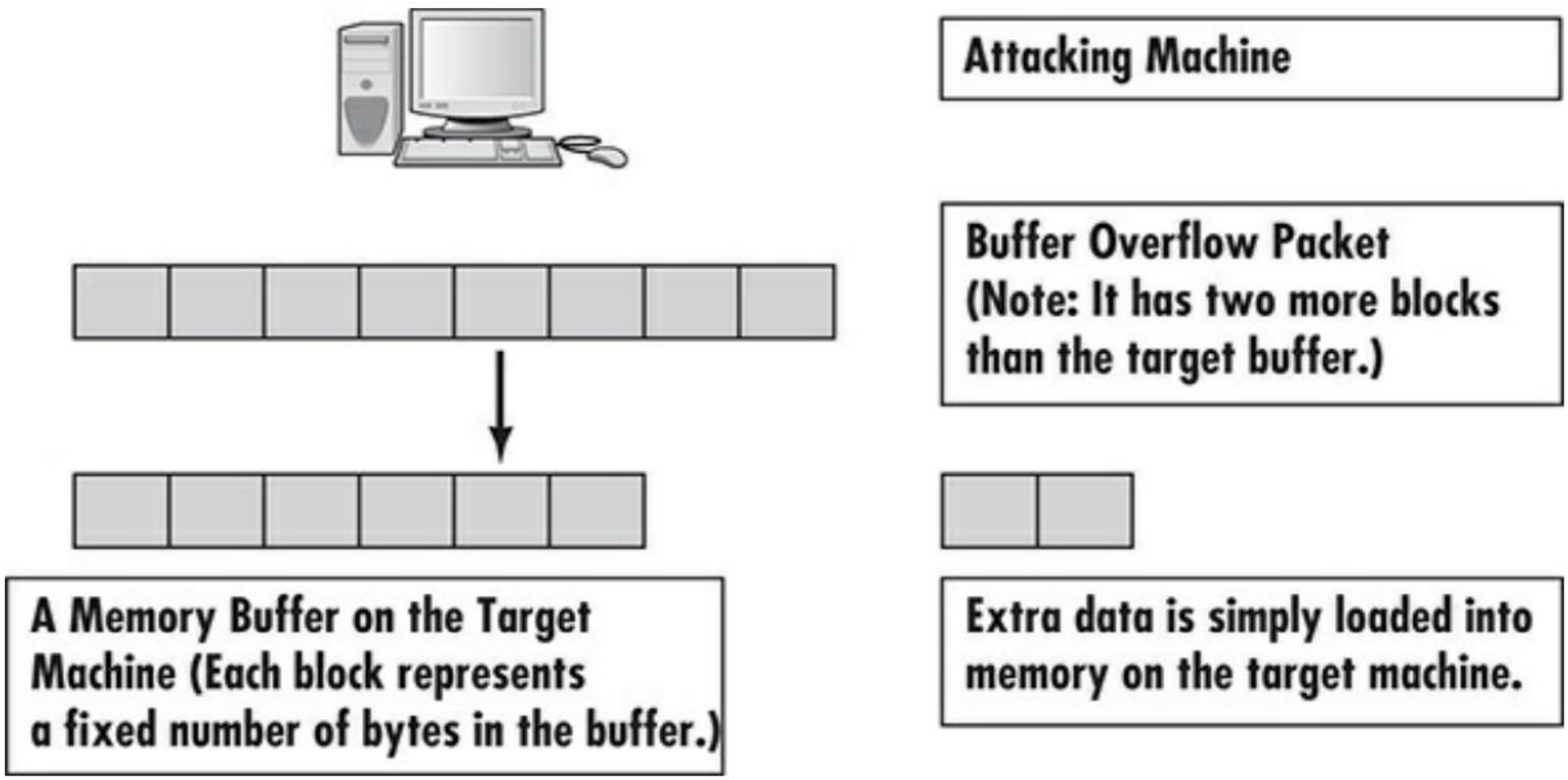


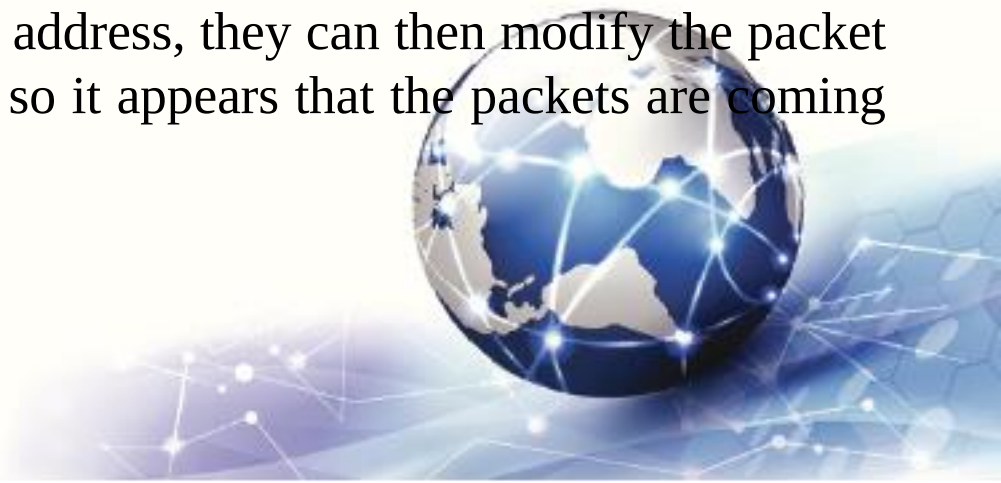
FIGURE 2-7 Buffer overflow attack

Defending Against Buffer Overflow Attacks

- A person moderately skilled in programming can write a program that purposefully writes more data into the buffer than it can hold.
  - For example, if the buffer can hold 1024 bytes of data and you try to fill it with 2048 bytes, the extra 1024 bytes is then simply loaded into memory.
- Susceptibility to a buffer overflow attack is entirely contingent on software flaws.
  - A perfectly written program would not allow buffer overflows. Because perfection is unlikely, the best defense against buffer overflow attacks is to routinely patch software so that flaws are corrected when the vendor discovers a vulnerability.

IP Spoofing

- IP spoofing is essentially a technique used by hackers to gain unauthorized access to computers.
  - Although this is the most common reason for IP spoofing,
  - it is occasionally done simply to mask the origins of a DoS attack.
    - In fact, DoS and DDoS attacks often mask the actual IP address from which the attack is originating.
- To successfully perpetrate an IP spoofing attack,
  - The hacker must first find the IP address of a machine that the target system considers a trusted source.
    - Hackers might employ a variety of techniques to find an IP address of a trusted host.
  - After they have that trusted IP address, they can then modify the packet headers of their transmissions so it appears that the packets are coming from that host.



IP spoofing

- The danger from IP spoofing is :
  - that some **firewalls** do not examine packets that appear to come from an internal IP address.
  - Routing packets through filtering routers is possible if they are not configured to filter incoming packets whose source address is in the local domain.

IP spoofing

- Examples of router configurations that are potentially vulnerable include Routers to external networks that support multiple internal interfaces Proxy firewalls where the proxy applications use the source IP address for authentication Routers with two interfaces that support subnetting on the internal network
- Routers that do not filter packets whose source address is in the local domain

Defending Against IP spoofing

- The best method of preventing IP spoofing is to install a filtering router.
  - Filtering routers filter incoming packets by not allowing a packet through if it has a source address from your internal network.
  - In addition, you should filter outgoing packets that have a source address different from your internal network to prevent a source IP spoofing attack from originating at your site.
- Many commercial firewall vendors, such as Cisco, FortiGate, D-Link, and Juniper, offer this option

Session Hijacking

- Session hijacking, also known as session hacking, is a type of attack where an attacker takes control of a valid user's session on a computer system or network service.
- The attacker aims to impersonate the legitimate user and perform unauthorized actions within the hijacked session.
- This attack typically targets web sessions, where users are authenticated through a login process and assigned a session identifier (such as a session cookie) to maintain their logged-in state.

Session Hijacking

- Here's how session hijacking generally works:
- Session Establishment: When a user logs into a website or a network service, they are assigned a session identifier (e.g., a session cookie) that is used to authenticate subsequent requests.
- Capture Session Identifier: The attacker intercepts the session identifier belonging to a valid user. This can be done through various means, such as eavesdropping on network traffic, stealing session cookies stored on the user's device, or exploiting vulnerabilities in the web application.
- Impersonation: With the stolen session identifier, the attacker masquerades as the legitimate user by injecting the identifier into their requests. This tricks the server into believing that the attacker's requests are coming from the legitimate user's session.
- Unauthorized Actions: The attacker can now perform various unauthorized actions within the hijacked session, depending on the privileges associated with the user's account.

Man-in-the-middle Attack.

- The most common sort of session hacking is the "**man-in-the-middle attack**."
  - In this scenario a hacker uses some sort of packet-sniffing program to :
    - simply listen in on the transmissions between two computers,
    - taking whatever information he or she wants but not actually disrupting the conversation.
  - A common component of such an attack is to execute a DoS attack against one endpoint to stop it from responding.
    - Because that endpoint is no longer responding;
      - the hacker can now interject his own machine to stand in for that endpoint.
- The point of hijacking a connection is
  - to **exploit the trust,**
  - **and** to gain access to a system to which one would not otherwise have access.

Defending Against Session Hijacking

- Should implement secure session management practices, such as using HTTPS to encrypt communication, employing secure cookies with appropriate attributes (e.g., Secure, Http Only, Same Site), implementing mechanisms for session expiration and re authentication, and regularly auditing and monitoring for suspicious activity related to user sessions.