IV1013 VT17 – Introduction to Computer Security

TCP/IP Attacks

Overview

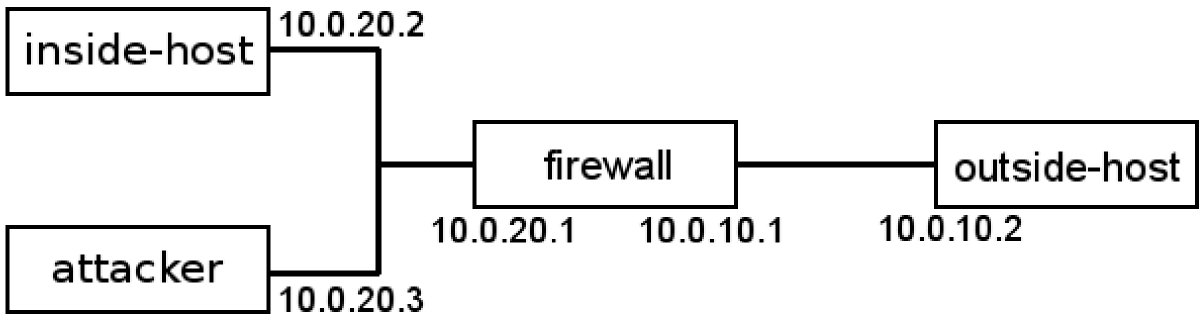
The learning objective of this lab is for you to gain the first-hand experience on the vulnerabilities of TCP/IP protocols, as well as on attacks against these vulnerabilities. The vulnerabilities in the TCP/IP protocols represent a special genre of vulnerabilities in protocol designs and implementations; they provide an invaluable lesson as to why security should be designed in from the beginning, rather than being added as an afterthought. Moreover, studying these vulnerabilities help you understand the challenges of network security and why many network security measures are needed. Vulnerabilities of the TCP/IP protocols occur at several layers.

Lab environment

Network setup

To conduct this lab, you need four machines: a firewall, an inside host, an outside host, and an attacker. You will use an LXC container for each of these machines. Let the outside network be 10.0.10.0/24, and the inside network be 10.0.20.0/24.

Topology



Tools

Wireshark - Sniffer and protocol analyzer
Tcpdump - Command-line based sniffer
Netwox - Tools to generate packets and spoof network traffic
Netcat (nc) - Lots of different tools, used to create a TCP server.

Tasks

Setting up the hosts

On the host VM, create a new LXC container for the attacker.

```
sudo lxc-clone -o lxc-template -n attacker -B overlayfs -s
```

Make sure you have an internet connection enabled on the host VM. Start the attacker:

```
sudo lxc-start –n attacker
```

Note: If the previous command does not work, use this instead:

```
sudo lxc-start --name attacker
```

At the login prompt, login to the attacker (same username and password as on the host VM), install the netwox tool, and then exit the container by shutting it down:

```
sudo apt-get update
sudo apt-get install netwox
sudo poweroff
```

On the host VM, change the attackers network link to the internal bridge.

edit `/var/lib/lxc/attacker/config` and change:

```
lxc.network.link = lxcbr0
```

to:

```
lxc.network.link = lxc-int-br
```

Start the LXC containers. Open one terminal window for each, and run these commands, one in each window:

```
sudo lxc-start -n attacker
sudo lxc-start -n inside-host
sudo lxc-start -n outside-host
sudo lxc-start -n firewall
```

Diasble DHCP on the LXC containers. In each container, edit `/etc/network/interfaces` and change:

```
      iface eth0 inet dhcp
```

to:

```
      iface eth0 inet manual
```

Then restart the network:

```
      sudo /etc/init.d/networking restart
```

Note: You might get a warning similar to "sudo: unable to resolve host attacker". You can ignore this warning or edit `/etc/hosts` and change:

```
      127.0.0.1 localhost
```

to:

```
      127.0.0.1 localhost, attacker
```

Similarly, you can edit all lxc containers.

Next configure IP addresses and routes on all of the containers.

On firewall:

```
      sudo ip address add 10.0.10.1/24 dev eth0
      sudo ip address add 10.0.20.1/24 dev eth1
```

On inside host:

```
      sudo ip address add 10.0.20.2/24 dev eth0
      sudo ip route add 10.0.10.0/24 via 10.0.20.1
```

On outside host:

```
      sudo ip address add 10.0.10.2/24 dev eth0
      sudo ip route add 10.0.20.0/24 via 10.0.10.1
```

On attacker:

```
      sudo ip address add 10.0.20.3/24 dev eth0
      sudo ip route add 10.0.10.0/24 via 10.0.20.1
```

Lastly, test connectivity by pinging each host from all others.

If you need to open an additional terminal for a container, use:

```
      sudo lxc-attach -n <container_name>
```

Task 1 (for 50 point level) - ARP cache poisoning

Example execution:

- Use the Inside host as victim and the firewall as recipient.
- Inspect victim's ARP table with the `arp -n` command.
- Start continuous `ping <ip_recipient>` command on victim, pinging the recipient.
- Start wireshark on the host VM and capture packets on the inside bridge `lxc-int-br`.
- Make sure that routing is turned off on the attacker:
  sudo `sysctl -w net.ipv4.ip_forward=0`
- Use `netwox` on the attacker with command 33 to forge a broadcast ARP request from the recipient's IP address to the victim's IP address using the attacker's MAC address. (`sudo netwox 33 --eth-dst ff:ff:ff:ff:ff:ff --arp-ipsrc <ip_recipient> --arp-ipdst <ip_victim>`)
- Monitor `wireshark` and the output of the `ping` command to see if and how long the traffic gets redirected to the attacker.
- Issue the forged packet again, and look at the `arp -n` command output on the victim immediately.

Expected result:
The recipient's (firewall) IP address will show up on the ARP table as having the attacker's MAC address. The ping requests will be redirected to the attacker and the victim will not receive any replies, until the victim sends out a new ARP request.

Task 2 (for 100-point level) - ICMP redirect attack

Example execution:

- Use the Inside host as victim and the outside host as recipient.
- Start wireshark on the host VM and capture packets on the inside bridge `lxc-int-br`.
- Accepting redirects might be off by default, so it might have to be turned on on the inside host:
  - `sysctl -w net.ipv4.conf.all.accept_redirects=1`
  - `sysctl -w net.ipv4.conf.eth0.accept_redirects=1`
- Start continuous `ping <ip_recipient>` command on victim, pinging the recipient host.
- Start `netwag` or `netwox` command 86 to sniff for traffic and send forged ICMP Redirect messages with recipient source IP and attacker IP as the new gateway. (`sudo netwox 86 --spoofip raw --filter "dst host 10.0.10.2" --gw 10.0.20.3 --src-ip 10.0.20.1`)

- Do the previous ARP attack to get the victim to send data to the attacker. (`sudo netwox 33 --eth-dst ff:ff:ff:ff:ff:ff --arp-ipsrc 10.0.20.1 --arp-ipdst 10.0.20.2`)
- Monitor `wireshark` and the output of the `ping` command.
- Terminate the `netwox` ICMP Redirect after a redirect message has been sent.

Expected result:
Messages from the victim to the recipient are sent to the attacker instead, but with the recipient's IP address. This will last much longer than with the ARP attack.

Task 3 (for 50-point level) - TCP session hijacking

The objective of the TCP Session Hijacking attack is to hijack an existing TCP connection (session) between two victims by injecting malicious contents into this session. If this connection is a telnet session, attackers can inject malicious commands into this session, causing the victims to execute the malicious commands. In this task you can instead use netcat to see what happens on a server when the attacker hijacks the session. You can start a netcat server daemon on the outside host through the following command:

```
nc -l <outside_host> 1024
```

Connecting to the server from the inside host can be done with this command:

```
nc <outside_host> 1024
```

Anything typed in the client terminal should now appear on the server terminal.

Note: If you use Wireshark to observe the network traffic, you should be aware that when Wireshark displays the TCP sequence number, by default, it displays the relative sequence number, which equals to the actual sequence number minus the initial sequence number. If you want to see the actual sequence number in a packet, you need to right click the TCP section of the Wireshark output, and select "Protocol Preference". In the popup window, uncheck the "Relative Sequence Number and Window Scaling" option.
Finally, the client needs to be on the same network as the attacker, otherwise netwox seems to have issues sending the packet.

Expected result:
The server executes the command and sends the first packet of output to the client. Since the client's sequence number is lower than the acknowledgement number of received data, the client does not accept it and does not send an ACK. The telnet session becomes unresponsive on the client’s side since the client attempts to send data with a lower sequence number than the server expects.

Organization

This assignment is done individually.

Submission

Upload your solution as a single ZIP archive in Canvas. The ZIP archive should contain the following:

- A detailed lab report that describes what you have done and what you have observed; you also need to provide explanation to the observations that are interesting or surprising.
- Packet traces to prove that your attacks have been successful. You need to include the actual pcap files here – do not use screenshots of the packet captures.
- A file called ”AUTHOR”, with full name and KTH email address of the author.

Requirements and points

For this assignment, you can get a maximum of 100 points. The grading scale is as follows:

- 50 points: In addition to the above, Task 1 and 3 correctly solved.
- 100 points: In addition to the above, Task 2 correctly solved.

Note that there can be deduction in points depending on the quality of your report. If you aim for 100 points, for instance, but we assess that your solution only solves half of the challenges in a satisfactory manner, you might get 80 points.