**Executive Summary**

This case study concentrates on the ICMP Redirect Attack in a virtualized lab environment. This is a sort of network attack that exploits ICMP vulnerabilities to alter the routing of IP packets. We demonstrated our attack in a controlled, virtualized network environment through a series of safely designed experiments.

Our reports reveal that an attacker, using a malicious router, can intercept and alter IP packets, affecting the integrity and confidentiality of data being transported. This attack primarily exploits the secure connections that exist between routers and hosts in a network, potentially exposing critical information.

To reduce the threat of such attacks, we propose establishing robust network security measures such as avoiding ICMP redirection whenever possible, using secure communication protocols like HTTPS, and updating and patching networking devices on a regular basis to address known vulnerabilities. Furthermore, network monitoring and intrusion detection systems can detect abnormal network activity that may indicate such attacks.

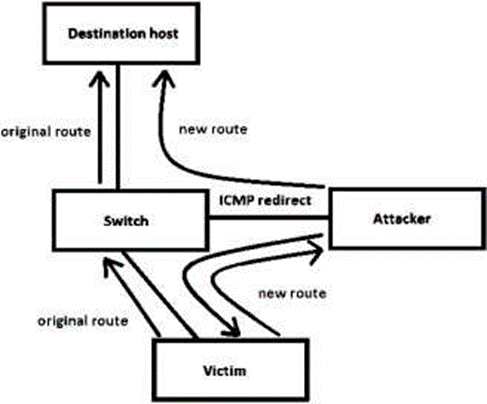
A diagram of a server

Description automatically generated

**Attack Description**

The Internet Control Message Protocol (ICMP) Redirect Attack is a sophisticated network attack that alters a target's routing information. This attack takes advantage of ICMP, a protocol employed by networking equipment to transmit status updates and error notifications. These messages can indicate various issues, such as an unavailable service or an unreachable host or router.

In the context of an ICMP Redirect Attack, a malicious actor crafts a deceptive ICMP redirect message and sends it to the unexpecting target. This fraudulent message tricks the recipient into rerouting its network traffic through a path that leads to a compromised system under the attacker's control. Once the traffic flows through the attacker's machine, they gain the ability to intercept, alter, or even block the target's network communications, effectively compromising the security and integrity of the data in transit.



**Vulnerabilities Exploited**

The ICMP Redirect Attack primarily takes advantage of the inherent trust that systems place in ICMP redirect messages. Many devices, including hosts and routers, are often configured by default to trust and act upon these messages without proper validation. Attackers can exploit this misplaced trust to manipulate the routing of a target's network traffic.

**Network Topology**

The attack scenario involves the following components:

- Target machine (10.9.0.5)

- Legitimate router (192.168.60.11)

- Attacker's machine (10.9.0.105)

- Compromised router (10.9.0.111) under attacker's control

- Destination server (192.168.60.5) the target communicates with

The attacker's goal is to inject a fraudulent ICMP redirect message, deceiving the target into rerouting this traffic through the compromised router at 10.9.0.111, which is under the attacker's control.

A computer system setup with information

Description automatically generated with medium confidence

**Attack Execution**

The ICMP Redirect Attack was carried out using Scapy, a powerful Python library for crafting and manipulating network packets.

**The attack involves the following key steps:**

* The attacker monitors network traffic, waiting for the target to communicate with the destination server.
* Upon detecting traffic, the attacker constructs a forged ICMP redirect message that appears to originate from the target's legitimate router.

1. The source IP address is spoofed to match the legitimate router.
2. The compromised router's IP is set as the new gateway for redirected traffic.
3. A copy of the intercepted packet is included to bypass OS validation checks.

* The attacker sends the fabricated ICMP redirect message to the target.
* If the target accepts the redirect, it updates its routing cache with the malicious entry.
* Subsequent traffic from the target to the destination server is now routed through the attacker's compromised machine.

A screenshot of a computer

Description automatically generated

**Victim’s Routing Cache Showing Redirected**

The target's routing cache now contains a redirected entry, indicating that traffic destined for the destination server (192.168.60.5) should be routed through the attacker's machine (10.9.0.111) acting as the gateway.

A screenshot of a computer

Description automatically generated

**Traceroute Output Showing Redirected Path**

A traceroute from the target to the destination server reveals the modified routing path, with traffic flowing through the malicious gateway.

The attacker's machine is now able to intercept, inspect, and potentially alter the redirected packets in transit, effectively positioning itself as a man-in-the-middle.

**Mitigation Strategies**

Risk mitigation associated with ICMP Redirect attacks:

* Disable ICMP redirect acceptance on systems unless operationally necessary.
* Restrict ICMP redirects to trusted routers only, where required.
* Implement network perimeter filtering to block ICMP redirects.
* Employ passive monitoring techniques to detect suspicious redirect messages.
* Ensure the integrity of sensitive traffic using cryptographic mechanisms.
* Proper configuration management and continuous monitoring are essential to prevent the misuse of ICMP redirects for malicious purposes.

**Case Reflection**

* This hands-on exercise demonstrated the potential for exploiting ICMP to manipulate a target's routing behavior. Although ICMP has legitimate uses, its functionality must be carefully controlled to prevent abuse.

Key takeaways from this study include:

* Even fundamental networking protocols can be exploited if not properly secured.
* Routing, as a critical infrastructure component, requires robust protection.
* Attackers with a presence within a network can leverage redirects to expand their access.
* A multi-layered defense approach, encompassing hardening, filtering, and monitoring, is crucial.
* Future investigations could explore redirect attacks on different platforms, compare mitigation strategies, or combine redirects with other MITM techniques.

**Additional experiments**

**Question 1**: modify the icmp.gw value in the attack script to a remote IP address (not on the local LAN) and observe the results.

A screenshot of a computer program

Description automatically generated

A screenshot of a computer

Description automatically generated

* The target's routing cache will not contain a redirected entry for the destination host (192.168.60.5) pointing to the remote IP.
* The traceroute from the target to the destination will not include the remote IP address in the path.
* The attack attempt will be unsuccessful because ICMP redirect messages are generally restricted to the local network and are ignored for remote IP addresses.

***Explanation***: ICMP redirect messages are intended to optimize routing within a local network. Routers typically discard ICMP redirect messages that specify a gateway outside the local network scope. As a result, attempting to redirect traffic to a remote IP address will have no impact on the target's routing behavior.

**Question 2**: change icmp.gw to an IP address of a non-existing or offline machine on the same network and analyze the outcome.

A screenshot of a computer code

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

* The target's routing cache will include a redirected entry for the destination host (192.168.60.5) pointing to the non-existent or offline machine's IP.
* The traceroute from the target to the destination will display the non-existent or offline machine's IP address in the path.
* However, when the target attempts to send traffic to the destination host, the traffic will fail to reach its intended destination because the specified gateway is unavailable.
* The target may encounter connectivity issues or timeouts when trying to communicate with the destination host.

***Explanation***: Despite the specified gateway being non-existent or offline, the target will still process the ICMP redirect message and update its routing cache accordingly. However, since the gateway is unreachable, the traffic will not be forwarded successfully, leading to communication failures.

**Question 3**: modify the send\_redirects settings in the docker-compose.yml file for the malicious router container, by setting them to 1. Launch the attack again and observe the results.A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

* The attack attempt will be unsuccessful, and the target's routing cache will not be updated with a redirected entry for the destination host (192.168.60.5) pointing to the malicious router.
* The traceroute from the target to the destination will not include the malicious router's IP address in the path.

***Explanation***: The send\_redirects settings determine whether the malicious router is permitted to send ICMP redirect messages. By default, these settings are disabled (set to 0) to prevent the router from sending redirects. When these settings are enabled by setting them to 1, the malicious router will start sending its own ICMP redirect messages, potentially interfering with the attack.

In this scenario, the malicious router's redirect messages may take precedence over the attacker's crafted redirect messages, causing the attack to fail. The target will not update its routing cache based on the attacker's redirect messages.

**ICMP\_REDIRECT py source code**

*#!/usr/bin/python3*

*# Importing necessary modules from Scapy*

**from scapy.all import \***

*# the outer IP packet with source and destination addresses*

**ip = IP(src='10.9.0.11', dst='10.9.0.5')**

*# the ICMP redirect message with specific type and code*

**icmp = ICMP(type=5, code=1)**

*# Set the gateway IP address in the ICMP redirect message*

**icmp.gw = '10.9.0.111'**

*# the inner IP packet that triggers the redirect message*

**ip2 = IP(src='10.9.0.5', dst='192.168.60.5')**

*# Send the composed packet, including both IP packets and ICMP message*

**send(ip/icmp/ip2/ICMP())**

References

[1] W. Du, SEED Labs – ICMP Redirect Attack Lab, Syracuse University, [Online] Available: https://seedsecuritylabs.org/Labs\_20.04/Networking/ICMP\_Redirect/

[2] Scapy documentation, [Online] Available: https://scapy.readthedocs.io/

[3] https://github.com/seed-labs/seed-labs/blob/master/category-network/ICMP\_Redirect/Old\_IP\_Attacks.tex

[4] https://medium.com/@sherishrat/icmp-redirect-attack-18755cd0897