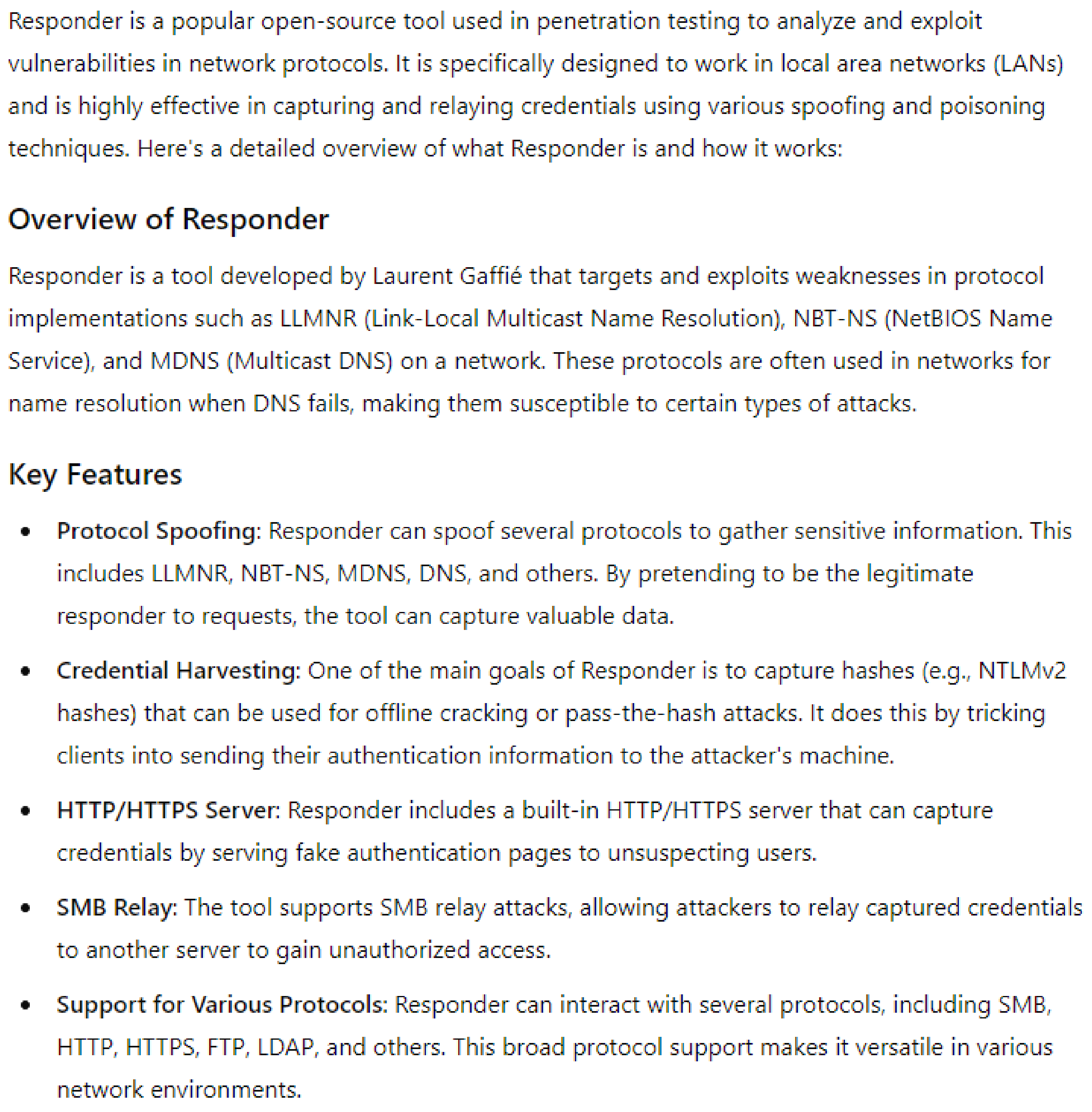
LLMNR - Responder Overview



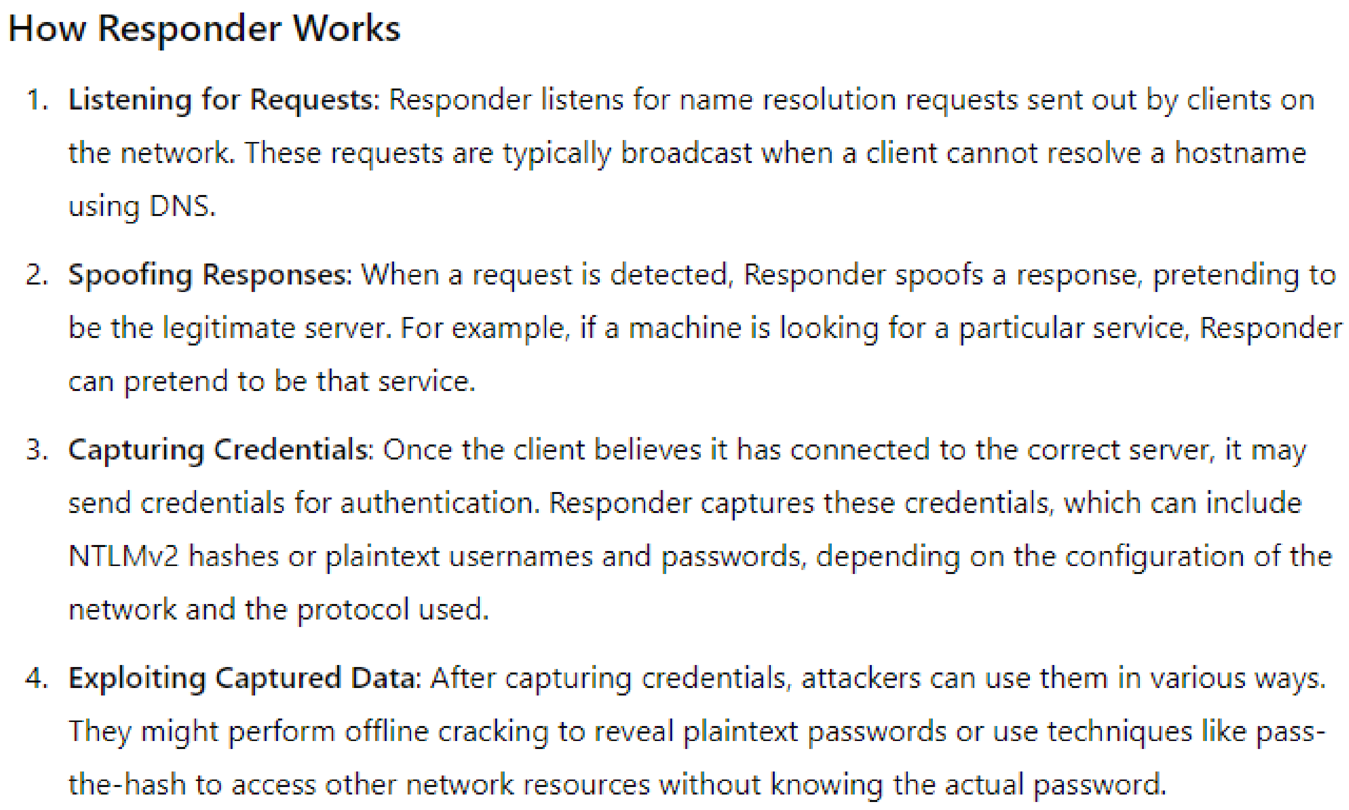
Responder is a popular open-source tool used in penetration testing to analyze and exploit vulnerabilities in network protocols. It is specifically designed to work in local area networks (LANs) and is highly effective in capturing and relaying credentials using various spoofing and poisoning techniques. Here's a detailed overview of what Responder is and how it works:

Overview of Responder

Responder is a tool developed by Laurent Gaffié that targets and exploits weaknesses in protocol implementations such as LLMNR (Link-Local Multicast Name Resolution), NBT-NS (NetBIOS Name Service), and MDNS (Multicast DNS) on a network. These protocols are often used in networks for name resolution when DNS fails, making them susceptible to certain types of attacks.

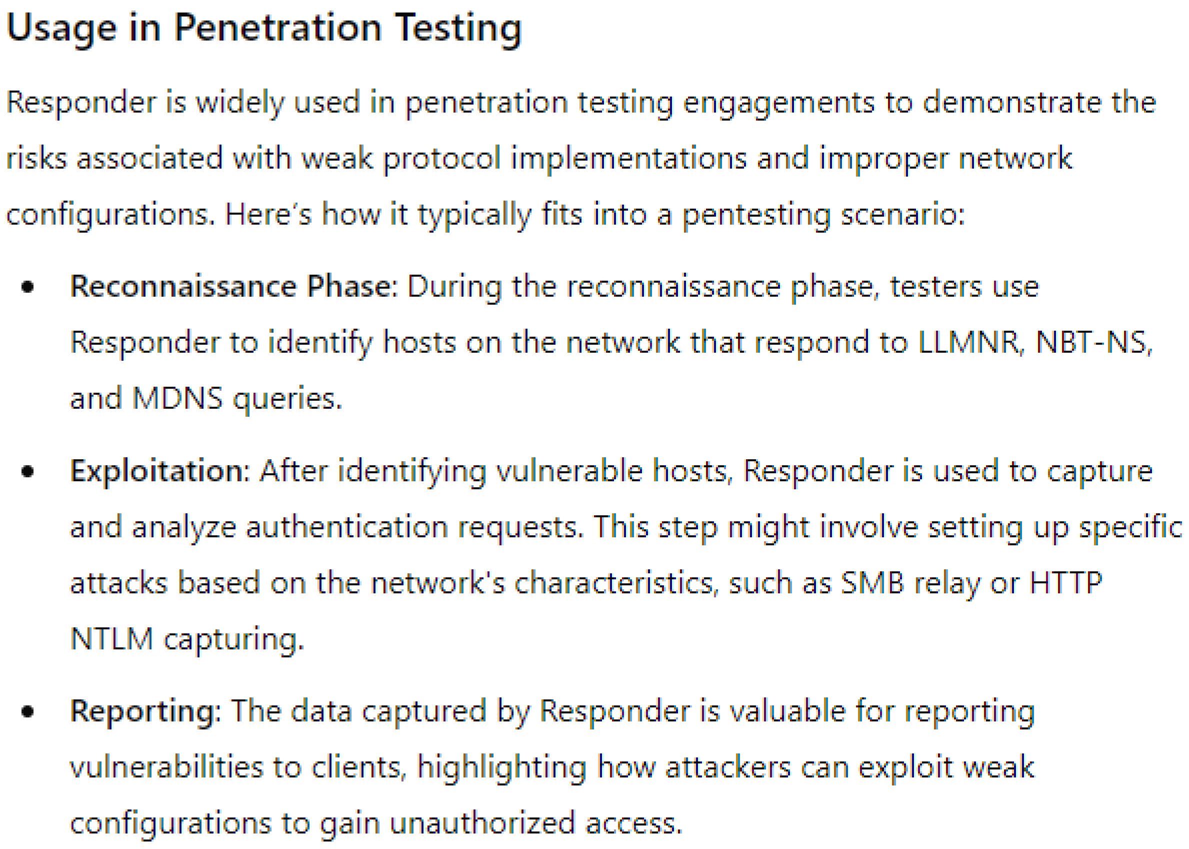
Key Features

- **Protocol Spoofing**: Responder can spoof several protocols to gather sensitive information. This includes LLMNR, NBT-NS, MDNS, DNS, and others. By pretending to be the legitimate responder to requests, the tool can capture valuable data.
- **Credential Harvesting**: One of the main goals of Responder is to capture hashes (e.g., NTLMv2 hashes) that can be used for offline cracking or pass-the-hash attacks. It does this by tricking clients into sending their authentication information to the attacker's machine.
- **HTTP/HTTPS Server:** Responder includes a built-in HTTP/HTTPS server that can capture credentials by serving fake authentication pages to unsuspecting users.
- **SMB Relay**: The tool supports SMB relay attacks, allowing attackers to relay captured credentials to another server to gain unauthorized access.
- **Support for Various Protocols**: Responder can interact with several protocols, including SMB, HTTP, HTTPS, FTP, LDAP, and others. This broad protocol support makes it versatile in various network environments.



How Responder Works

1. **Listening for Requests:** Responder listens for name resolution requests sent out by clients on the network. These requests are typically broadcast when a client cannot resolve a hostname using DNS.
2. **Spoofing Responses:** When a request is detected, Responder spoofs a response, pretending to be the legitimate server. For example, if a machine is looking for a particular service, Responder can pretend to be that service.
3. **Capturing Credentials**: Once the client believes it has connected to the correct server, it may send credentials for authentication. Responder captures these credentials, which can include NTLMv2 hashes or plaintext usernames and passwords, depending on the configuration of the network and the protocol used.
4. **Exploiting Captured Data:** After capturing credentials, attackers can use them in various ways. They might perform offline cracking to reveal plaintext passwords or use techniques like pass-the-hash to access other network resources without knowing the actual password.

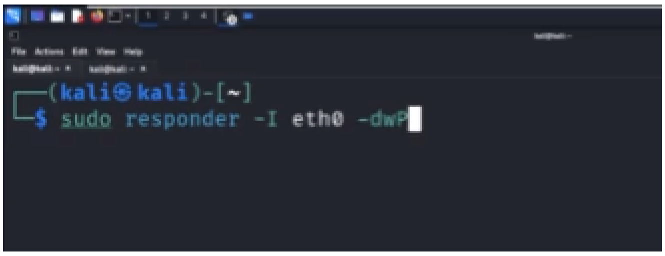


Usage in Penetration Testing

Responder is widely used in penetration testing engagements to demonstrate the risks associated with weak protocol implementations and improper network configurations. Here's how it typically fits into a pentesting scenario:

- **Reconnaissance Phase**: During the reconnaissance phase, testers use Responder to identify hosts on the network that respond to LLMNR, NBT-NS, and MDNS queries.
- **Exploitation**: After identifying vulnerable hosts, Responder is used to capture and analyze authentication requests. This step might involve setting up specific attacks based on the network's characteristics, such as SMB relay or HTTP NTLM capturing.
- **Reporting**: The data captured by Responder is valuable for reporting vulnerabilities to clients, highlighting how attackers can exploit weak configurations to gain unauthorized access.

We are going to use Responder:



```
┌──(kali㉿kali)-[~]
└─$ sudo responder -I eth0 -dwP
```

Flags used : -dwP

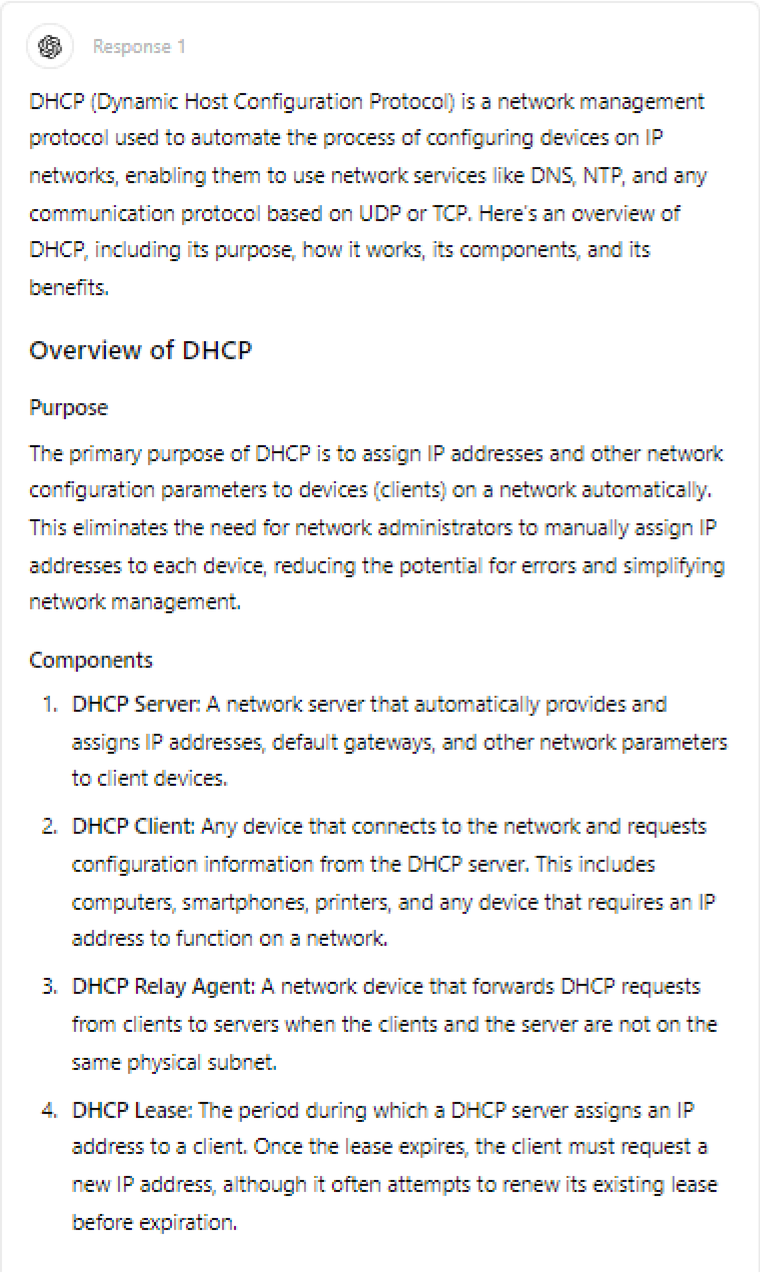
-d stands for DHCP

-w stands for WPAD

-P stands for "Proxy Auth". Description is "Forces NTLM (transparently)/Basic (prompt) authentication for the proxy. WPAD doesn't need to be ON.

-v this is for verbose. If you have already captured a particular hash, Responder wont show the captured hash, only that it is already stored. So, lets add this flag as well.

DHCP Overview:



Response 1

DHCP (Dynamic Host Configuration Protocol) is a network management protocol used to automate the process of configuring devices on IP networks, enabling them to use network services like DNS, NTP, and any communication protocol based on UDP or TCP. Here's an overview of DHCP, including its purpose, how it works, its components, and its benefits.

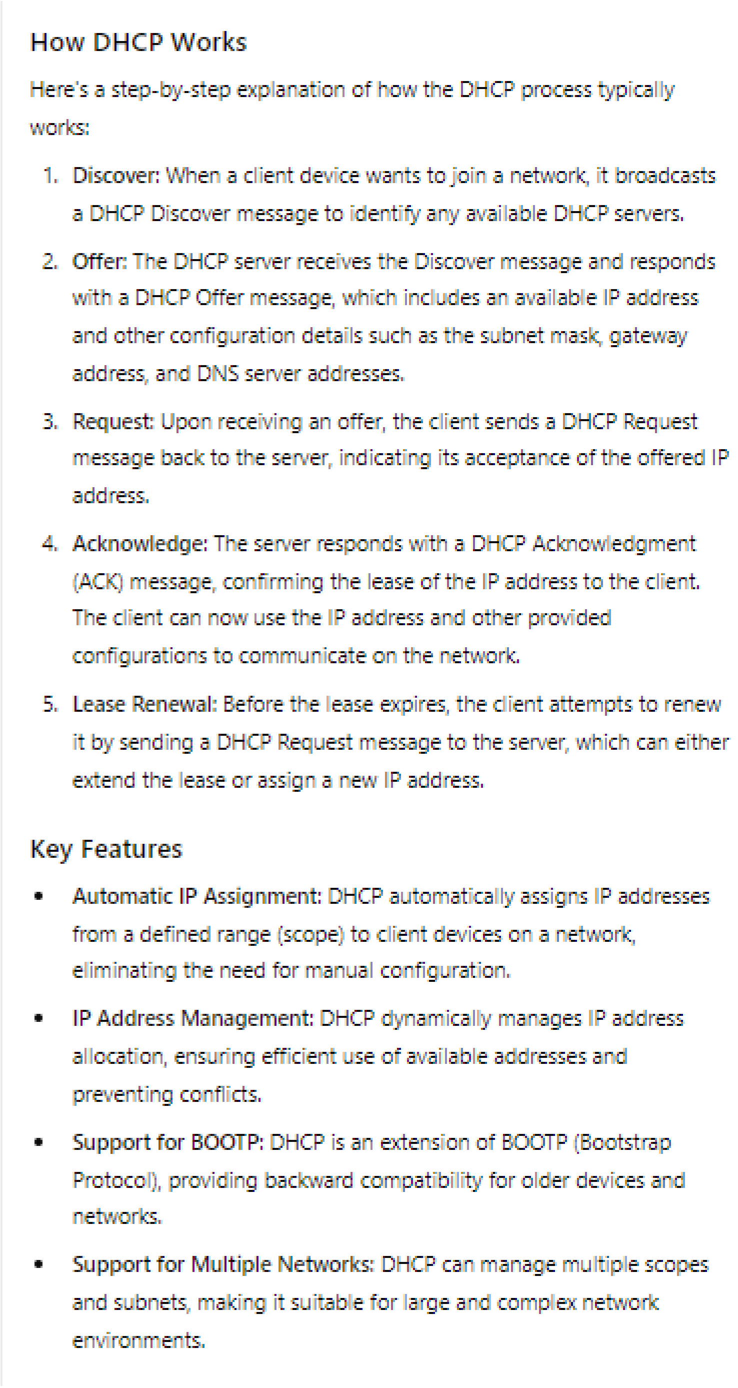
Overview of DHCP

Purpose

The primary purpose of DHCP is to assign IP addresses and other network configuration parameters to devices (clients) on a network automatically. This eliminates the need for network administrators to manually assign IP addresses to each device, reducing the potential for errors and simplifying network management.

Components

1. **DHCP Server:** A network server that automatically provides and assigns IP addresses, default gateways, and other network parameters to client devices.
2. **DHCP Client:** Any device that connects to the network and requests configuration information from the DHCP server. This includes computers, smartphones, printers, and any device that requires an IP address to function on a network.
3. **DHCP Relay Agent:** A network device that forwards DHCP requests from clients to servers when the clients and the server are not on the same physical subnet.
4. **DHCP Lease:** The period during which a DHCP server assigns an IP address to a client. Once the lease expires, the client must request a new IP address, although it often attempts to renew its existing lease before expiration.



How DHCP Works

Here's a step-by-step explanation of how the DHCP process typically works:

1. **Discover:** When a client device wants to join a network, it broadcasts a DHCP Discover message to identify any available DHCP servers.
2. **Offer:** The DHCP server receives the Discover message and responds with a DHCP Offer message, which includes an available IP address and other configuration details such as the subnet mask, gateway address, and DNS server addresses.
3. **Request:** Upon receiving an offer, the client sends a DHCP Request message back to the server, indicating its acceptance of the offered IP address.
4. **Acknowledge:** The server responds with a DHCP Acknowledgment (ACK) message, confirming the lease of the IP address to the client. The client can now use the IP address and other provided configurations to communicate on the network.
5. **Lease Renewal:** Before the lease expires, the client attempts to renew it by sending a DHCP Request message to the server, which can either extend the lease or assign a new IP address.

Key Features

- **Automatic IP Assignment:** DHCP automatically assigns IP addresses from a defined range (scope) to client devices on a network, eliminating the need for manual configuration.
- **IP Address Management:** DHCP dynamically manages IP address allocation, ensuring efficient use of available addresses and preventing conflicts.
- **Support for BOOTP:** DHCP is an extension of BOOTP (Bootstrap Protocol), providing backward compatibility for older devices and networks.
- **Support for Multiple Networks:** DHCP can manage multiple scopes and subnets, making it suitable for large and complex network environments.

Benefits of DHCP

1. **Simplicity:** DHCP simplifies the process of connecting devices to a network by automating IP address assignment and configuration, reducing administrative overhead.
2. **Efficiency:** Automatic IP address management prevents IP address conflicts and ensures efficient use of network resources.
3. **Scalability:** DHCP can support networks of any size, from small home networks to large enterprise environments with thousands of devices.
4. **Flexibility:** DHCP allows network administrators to change network configurations without manually updating each device, enabling easy adaptation to changing network requirements.
5. **Centralized Management:** DHCP servers centralize network configuration, making it easier to manage and monitor network settings.

Example Configuration

Here's an example of a basic DHCP configuration for a home network:

- **IP Address Range:** 192.168.1.100 to 192.168.1.200
- **Subnet Mask:** 255.255.255.0
- **Default Gateway:** 192.168.1.1
- **DNS Servers:** 8.8.8.8, 8.8.4.4
- **Lease Time:** 24 hours

DHCP Server Configuration Example

Here's a basic example of how to configure a DHCP server using a popular server like **ISC DHCP** on a Linux system. This example sets up a DHCP server for a simple local network.

ISC DHCP Configuration Example

1. **Install the DHCP Server**

   Install the ISC DHCP server package using a package manager. On a Debian-based system, use the following command:

   ```bash
   sudo apt-get install isc-dhcp-server
   ```

2. Edit the DHCP Configuration File

   Open the DHCP configuration file in a text editor. The file is usually located at `/etc/dhcp/dhcpd.conf`.

   ```bash
   sudo nano /etc/dhcp/dhcpd.conf
   ```

3. Define Global Settings

   Define global settings that apply to all subnets.

   ```plaintext
   # Global settings
   option domain-name "example.com";
   option domain-name-servers ns1.example.com, ns2.example.com;
   default-lease-time 600;
   max-lease-time 7200;
   ```

4. Configure a Subnet

   Configure a subnet, specifying the range of IP addresses to be assigned, the network mask, the default gateway, and other options.

   ```plaintext
   # Subnet configuration
   subnet 192.168.1.0 netmask 255.255.255.0 {
       range 192.168.1.100 192.168.1.200;
       option routers 192.168.1.1;
       option subnet-mask 255.255.255.0;
       option broadcast-address 192.168.1.255;
       option domain-name-servers 8.8.8.8, 8.8.4.4;
       default-lease-time 600;
       max-lease-time 7200;
   }
   ```

5. Assign Static IP Addresses (Optional)

   Optionally, assign static IP addresses to specific devices using their MAC addresses.

   ```plaintext
   # Static IP assignment
   host Printer {
       hardware ethernet 00:11:22:33:44:55;
       fixed-address 192.168.1.50;
   }
   ```

6. Start the DHCP Server

   Start the DHCP server using the following command:

   ```bash
   sudo service isc-dhcp-server start
   ```

7. Check the Status

   Check the status of the DHCP server to ensure it's running correctly:

   ```bash
   sudo service isc-dhcp-server status
   ```

8. Configure the DHCP Relay (Optional)

   If your DHCP server is on a different subnet than your clients, you may need to configure a DHCP relay agent to forward requests.

   ```plaintext
   # DHCP relay configuration
   interface eth0 {
       server 192.168.2.1;  # DHCP server IP address
       listen on 192.168.1.0;  # Subnet where the relay listens
   }
   ```

This configuration example demonstrates how to set up a DHCP server to provide dynamic IP address allocation for a simple network, with optional static IP assignments for specific devices.

Security Considerations

While DHCP is a convenient and powerful protocol, it does have some security considerations:

- **DHCP Spoofing:** Attackers can set up rogue DHCP servers to provide malicious configurations to clients, potentially redirecting traffic or capturing sensitive information. Implementing DHCP Snooping on network switches can mitigate this risk.
- **IP Address Exhaustion:** Without proper management, a network could run out of available IP addresses if leases are not reclaimed promptly. Proper lease management and monitoring can help prevent this issue.
- **Lack of Authentication:** DHCP lacks built-in authentication mechanisms, making it vulnerable to unauthorized devices requesting IP addresses. Network Access Control (NAC) solutions can help enforce authentication before assigning IP addresses.

Conclusion

DHCP is an essential protocol for modern networks, providing efficient and automated management of IP addresses and configurations. By understanding its operation and benefits, network administrators can effectively manage their networks and ensure seamless connectivity for all devices.

Key Features of DHCP

1. **Automatic IP Address Assignment:** DHCP automatically assigns IP addresses to devices (clients) on a network, reducing the need for manual configuration.
2. **Centralized Management:** Network administrators can manage all IP address allocations from a central DHCP server, simplifying the administration of large networks.
3. **Configuration Parameters Distribution:** Besides IP addresses, DHCP can also assign other network configuration parameters, such as subnet masks, default gateways, DNS servers, and more.
4. **Lease Concept:** DHCP assigns IP addresses based on leases, meaning each IP address is assigned for a specific period. When the lease expires, the client must request a renewal, which helps in the efficient reuse of IP addresses.

How DHCP Works

The DHCP process consists of several steps, often summarized by the acronym DORA:

1. **Discovery:** When a device connects to a network, it broadcasts a DHCPDISCOVER message to identify available DHCP servers.
2. **Offer:** DHCP servers on the network respond with a DHCPOFFER message, offering an IP address to the client.
3. **Request:** The client responds to the first DHCPOFFER it receives with a DHCPREQUEST message, indicating that it has accepted the offer.
4. **Acknowledgment:** The DHCP server confirms the lease assignment with a DHCPACK message. At this point, the client can use the assigned IP address and network configuration parameters.

DHCP Packet Structure

A DHCP packet is structured similarly to a BOOTP packet, with additional options to support DHCP functionalities. Key fields in a DHCP packet include:

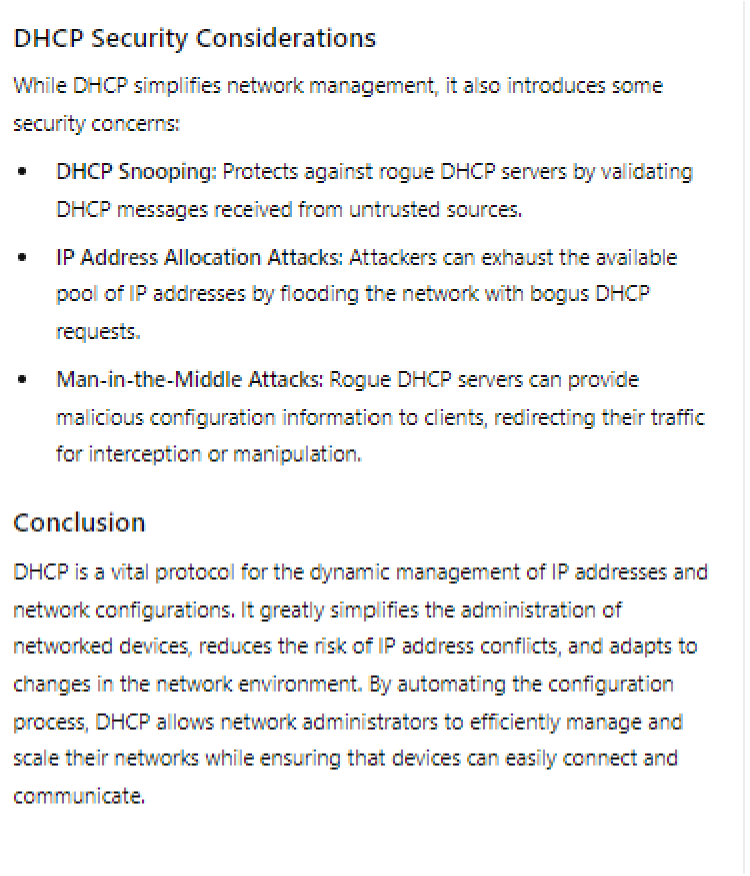
- **op:** Message type (1 for request, 2 for reply)
- **htype:** Hardware address type (e.g., Ethernet)
- **hlen:** Hardware address length (e.g., 6 for Ethernet)
- **hops:** Number of relay agent hops from client to server
- **xid:** Transaction ID, a random number chosen by the client
- **secs:** Seconds elapsed since client began address acquisition
- **flags:** Flags, including a broadcast flag
- **ciaddr:** Client IP address (if already assigned)
- **yiaddr:** 'Your' (client) IP address
- **siaddr:** IP address of the next server to use in the bootstrap process
- **giaddr:** Relay agent IP address
- **chaddr:** Client hardware address
- **sname:** Optional server host name
- **file:** Boot file name

Benefits of DHCP

- **Simplifies Management:** Reduces the administrative burden of manually configuring IP addresses.
- **Prevents IP Conflicts:** Automatically manages IP address allocation, minimizing conflicts.
- **Flexibility:** Easily accommodates network changes, such as adding or removing devices.
- **Scalability:** Supports large networks with dynamic address allocation and reallocation.

Common DHCP Configuration Options

- **IP Address:** The unique address assigned to the client.
- **Subnet Mask:** Defines the network and host portions of the IP address.
- **Default Gateway:** The router that forwards traffic to destinations outside the local network.
- **DNS Servers:** Servers that resolve domain names to IP addresses.
- **Lease Time:** Duration for which the IP address is leased to the client.



DHCP Security Considerations

While DHCP simplifies network management, it also introduces some security concerns:

- **DHCP Snooping:** Protects against rogue DHCP servers by validating DHCP messages received from untrusted sources.
- **IP Address Allocation Attacks:** Attackers can exhaust the available pool of IP addresses by flooding the network with bogus DHCP requests.
- **Man-in-the-Middle Attacks:** Rogue DHCP servers can provide malicious configuration information to clients, redirecting their traffic for interception or manipulation.

Conclusion

DHCP is a vital protocol for the dynamic management of IP addresses and network configurations. It greatly simplifies the administration of networked devices, reduces the risk of IP address conflicts, and adapts to changes in the network environment. By automating the configuration process, DHCP allows network administrators to efficiently manage and scale their networks while ensuring that devices can easily connect and communicate.

WPAD:

WPAD (Web Proxy Auto-Discovery) is a protocol that allows network clients to automatically discover proxy configuration settings, such as the address of a proxy server. WPAD is commonly used in enterprise environments to simplify the process of configuring web browsers to use proxy servers.

However, WPAD is vulnerable to certain attacks, particularly when combined with DHCP and DNS spoofing. A rogue WPAD proxy server attack can intercept and manipulate network traffic, leading to severe security risks. Here's a detailed explanation of how WPAD works, the risks involved, and how to mitigate these risks.

How WPAD Works

1. **Automatic Proxy Discovery:** WPAD allows devices to automatically locate and configure a proxy server by looking for a specific file, typically called `wpad.dat`, hosted on a web server. This file contains the proxy configuration script.
2. **Discovery Mechanisms:**
   - **DHCP:** WPAD clients can request the location of the `wpad.dat` file using DHCP option 252.
   - **DNS:** WPAD clients perform DNS lookups for the hostname `wpad` in their local domain to find the proxy configuration file.
3. **Proxy Auto-Configuration (PAC) File:** The `wpad.dat` file is essentially a Proxy Auto-Configuration (PAC) file containing JavaScript that instructs the browser on how to route traffic through the proxy server.

WPAD Rogue Proxy Server Attack

In a rogue proxy server attack, an attacker exploits WPAD to redirect network traffic through a malicious proxy server. Here's how such an attack might occur:

Attack Steps

1. **Network Access:** The attacker gains access to a target network, often via a wireless connection or through compromised devices.
2. **DHCP Spoofing:** The attacker sets up a rogue DHCP server to provide WPAD information (using option 252) that points to a malicious proxy server.
3. **DNS Spoofing:** Alternatively, the attacker uses DNS spoofing to respond to DNS queries for `` `wpad.<domain>` `` with the IP address of the rogue server.
4. **Serving Malicious PAC File:** The attacker's server hosts a malicious `` `wpad.dat` `` file that contains instructions to route traffic through the attacker's proxy.
5. **Interception and Manipulation:** As clients unknowingly configure themselves to use the rogue proxy, the attacker intercepts, monitors, and potentially manipulates network traffic.

Potential Impacts

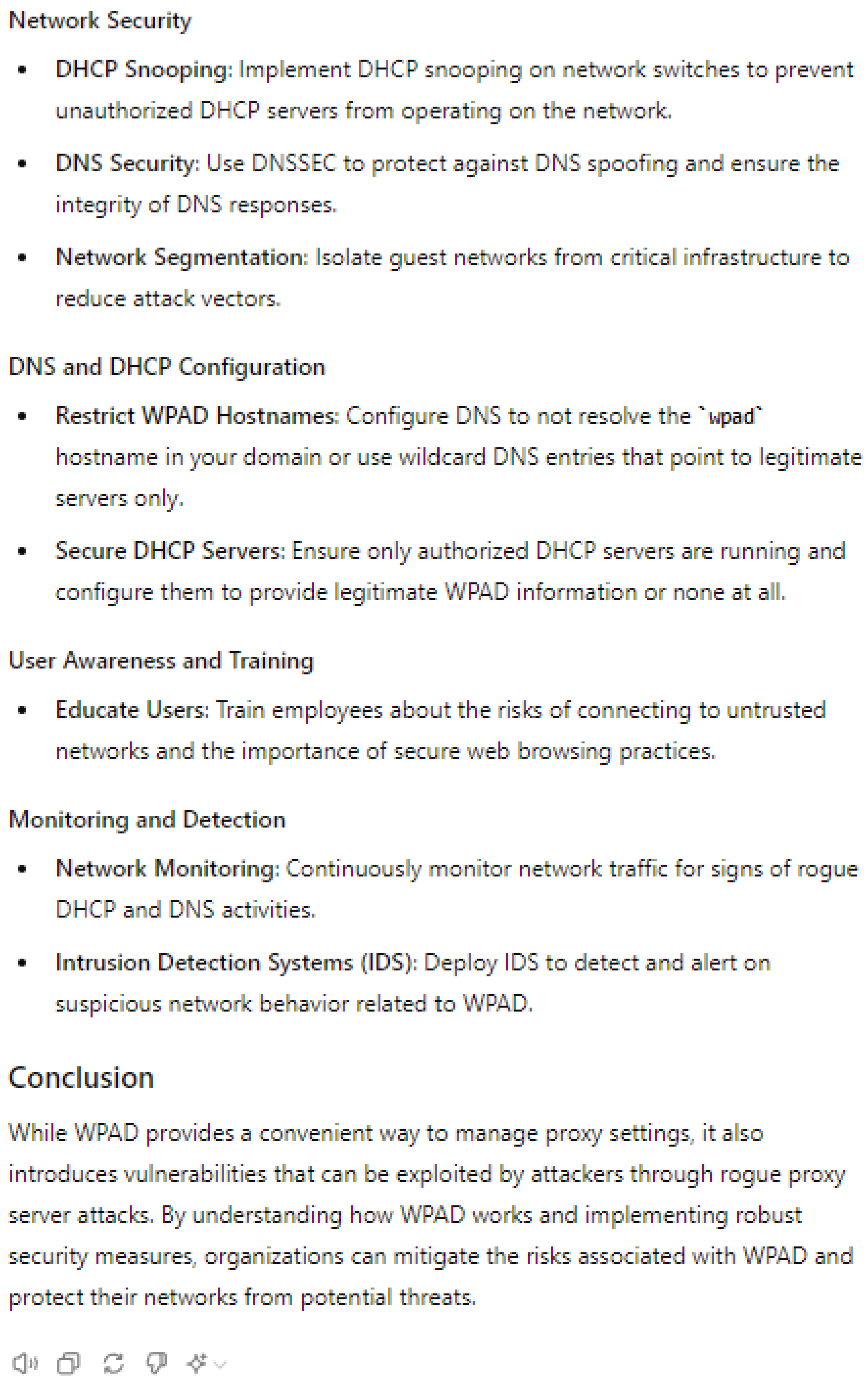
- **Data Interception:** Sensitive information such as usernames, passwords, and cookies can be captured.
- **Traffic Manipulation:** The attacker can alter the content of web pages, inject malware, or redirect users to phishing sites.
- **Credential Harvesting:** Attackers may capture authentication credentials for various services.
- **Malware Distribution:** The rogue proxy can deliver malware to clients by redirecting legitimate downloads to malicious files.

Mitigating WPAD Rogue Proxy Attacks

To protect against WPAD rogue proxy attacks, consider the following countermeasures:

Disable WPAD

- **Disable WPAD in Browsers:** Manually configure browsers to use specific proxy settings rather than relying on automatic discovery.
- **Disable WPAD in Operating Systems:** Ensure that WPAD is disabled in network settings on all devices.



Network Security

- **DHCP Snooping:** Implement DHCP snooping on network switches to prevent unauthorized DHCP servers from operating on the network.
- **DNS Security:** Use DNSSEC to protect against DNS spoofing and ensure the integrity of DNS responses.
- **Network Segmentation:** Isolate guest networks from critical infrastructure to reduce attack vectors.

DNS and DHCP Configuration

- **Restrict WPAD Hostnames:** Configure DNS to not resolve the `` `wpad` `` hostname in your domain or use wildcard DNS entries that point to legitimate servers only.
- **Secure DHCP Servers:** Ensure only authorized DHCP servers are running and configure them to provide legitimate WPAD information or none at all.

User Awareness and Training

- **Educate Users:** Train employees about the risks of connecting to untrusted networks and the importance of secure web browsing practices.

Monitoring and Detection

- **Network Monitoring:** Continuously monitor network traffic for signs of rogue DHCP and DNS activities.
- **Intrusion Detection Systems (IDS):** Deploy IDS to detect and alert on suspicious network behavior related to WPAD.

Conclusion

While WPAD provides a convenient way to manage proxy settings, it also introduces vulnerabilities that can be exploited by attackers through rogue proxy server attacks. By understanding how WPAD works and implementing robust security measures, organizations can mitigate the risks associated with WPAD and protect their networks from potential threats.

ProxyAuth Overview:

Purpose of the `-P` Flag

The `-P` or `--ProxyAuth` flag in Responder is designed to capture proxy authentication attempts by leveraging the WPAD protocol. Here's how it works:

- **WPAD Exploitation:** WPAD is a protocol that allows clients on a network to automatically discover and configure proxy settings. When the `-P` flag is enabled, Responder sets up a rogue WPAD server to respond to WPAD queries from client machines.
- **Proxy Authentication Capture:** By responding to WPAD queries, Responder provides a proxy configuration that points to itself. When clients attempt to use the rogue proxy, they send their authentication credentials (such as NTLM hashes) to Responder.
- **Credential Harvesting:** The captured credentials can then be used for further attacks, such as offline password cracking or pass-the-hash attacks, allowing the attacker to gain unauthorized access to network resources.

How the `-P` Flag Works

Here's a step-by-step breakdown of what happens when you use the `-P` flag with Responder:

1. **Responder Setup:** You start Responder with the `-P` flag to enable proxy authentication capturing.

   ```bash
   sudo python3 Responder.py -I eth0 -P
   ```

   Replace `eth0` with your network interface.
2. **WPAD Response:** Responder listens for WPAD requests from clients and responds with a malicious proxy configuration pointing to the attacker's machine.
3. **Client Configuration:** When a client machine attempts to access the internet or network resources, it uses the provided proxy settings. The browser or application sends proxy authentication credentials to the rogue proxy (Responder).
4. **Credential Capture:** Responder captures the proxy authentication credentials, typically NTLM hashes, for further exploitation.
5. **Analysis and Exploitation:** The attacker analyzes the captured credentials to crack passwords or perform other attacks on the network.

Mitigating WPAD and Proxy Auth Attacks

To protect against attacks using WPAD and rogue proxy servers, consider the following security measures:

1. **Disable WPAD:** If WPAD is not required, disable it on all network clients and browsers to eliminate the attack surface.
2. **Secure Proxy Configurations:** Use manual proxy configurations or authenticated proxies that do not rely on automatic discovery.
3. **Implement DNS Security:** Use DNSSEC to secure DNS responses and prevent spoofing attacks.
4. **Enable Network Segmentation:** Isolate sensitive areas of the network to limit the impact of potential attacks.
5. **Educate Users:** Train users to recognize suspicious network behavior and report potential security incidents.
6. **Monitor Network Traffic:** Regularly monitor network traffic for anomalies that may indicate a rogue proxy or unauthorized WPAD responses.
7. **Deploy Security Solutions:** Use intrusion detection systems (IDS) and other security solutions to detect and alert on potential WPAD exploitation attempts.