PENTERA | Pentera Labs™ Research Series

Credential Harvesting Using DHCP Spoofing

The short path from DHCP spoofing to forced authentication and credential harvesting and how best to mitigate against it.

Yuval Lazar

DHCP may be famous for being an essential Windows networking protocol, but it is also infamous, or at least it should be, for falling victim to cyber attacks and leading adversaries to dangerous achievements.

DHCP spoofing gained recognition back in 2008 with Trojan. Flush.M (source), now a commonly known APT attack vector. Since then, the vector has diversified into several variants that all use different DHCP spoofing attacks to change DNS server entries for their malicious intents.

On-premise poisoning techniques continue to be one of the leading mid-stage attacks used by adversaries to collect data, enable lateral movement, and perform privilege escalation after gaining an initial foothold in the network. In this post, we describe the DHCP spoofing attack vector and explain how it could be abused by attackers.

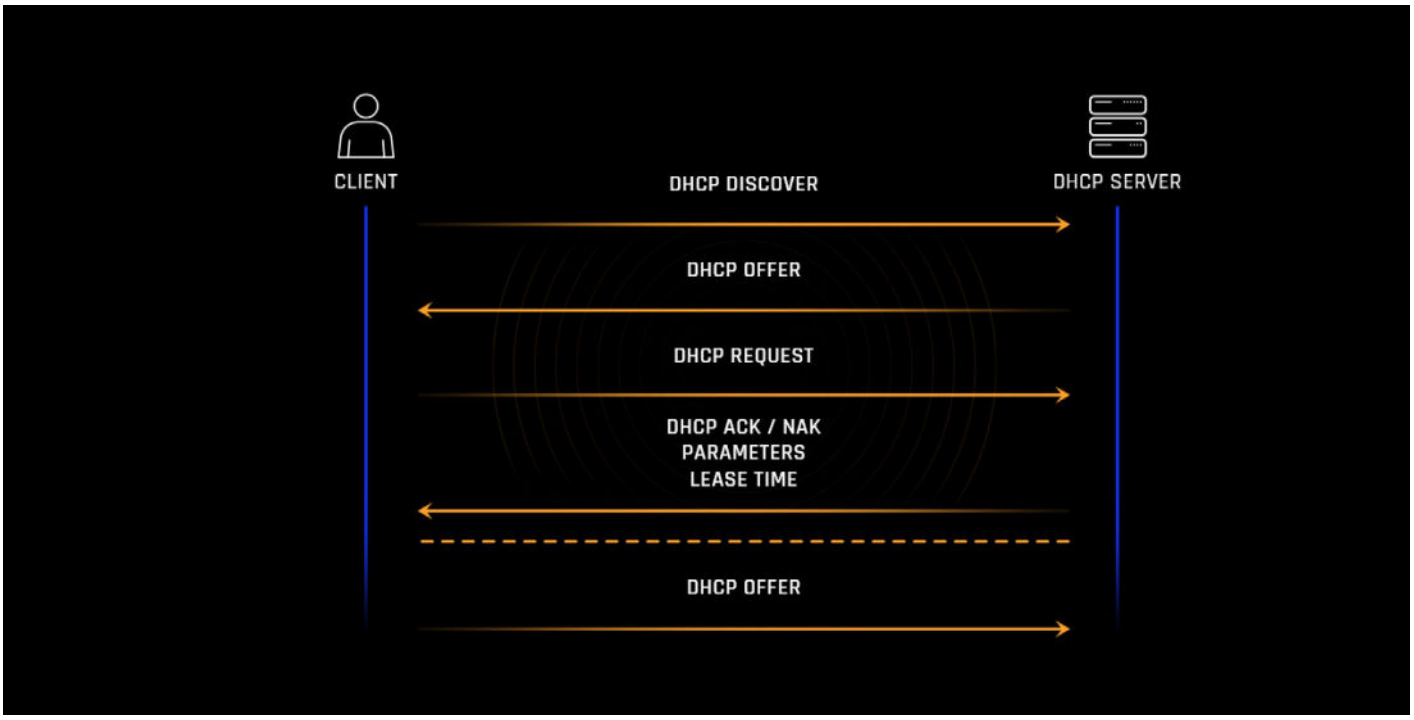
What is DHCP?

DHCP (Dynamic Host Configuration Protocol) is an application layer network management protocol that provides a quick, automatic and central system for the distribution and assignment of IP addresses and TCP/IP configuration information for clients in a network.

DHCP can be used to assign subnet mask information, default gateway IP addresses, domain name system (DNS) addresses, and more. Another feature of DHCP configurations is that they are time-bound by the DHCP Lease Time, which determines how often they must be renewed.

The Basics of DHCP Allocation

Before we move on to describe DHCP spoofing and poisoning techniques, let's review the basic workings of the DHCP networking protocol.



In a normal scenario, when a client first connects to a new DHCP network, the process is as follows:

- The client sends a DHCPDISCOVER request by broadcast. The request includes information about the client, such as the MAC Address, so that the server knows which client sent the request.
- The DHCP server responds with a DHCPOFFER. The DHCPOFFER offers the client an IP address from the DHCP server's pool of available addresses and sets the destination IP to the offered address.
- The client replies with a DHCPREQUEST to confirm the address provided by the server and requests additional details from the server.
- The DHCP server responds with a DHCPACK to acknowledge the Client's request and confirm the requested IP address. Additional information is provided, including the DNS server address, domain name, and lease time.
- When the time for the lease is over, the Client will send a DHCPRELEASE message to inform the server that the address can now be re-distributed to another client.

The "classic" scenario we just described can take many variations. Here's a short list of a few other possible vectors:

- If the DHCP server receives an invalid or unsupported request, it will send a DHCPNAK response to the client. This can happen even if the request is valid but the server's address pool is depleted and no addresses are available.
- A client already familiar with the DHCP server address can skip some of the steps and send a DHCPREQUEST by unicast directly to the server. This is standard for DHCP lease renewal.

- If the client has a static IP address configured, it can send a DHCPINFORM message to notify the server and avoid a conflict with another client.
- If a DHCP relay agent is involved, the process is a bit different. A relay agent can be any TCP/IP host used to forward requests and replies between the DHCP server and clients and is used when the DHCP server is on a different LAN than the clients. A relay agent will receive DHCP messages and then generate a new DHCP message to send out on another interface.
- DHCP messages can be sent by either broadcast or unicast traffic, depending on the configuration.

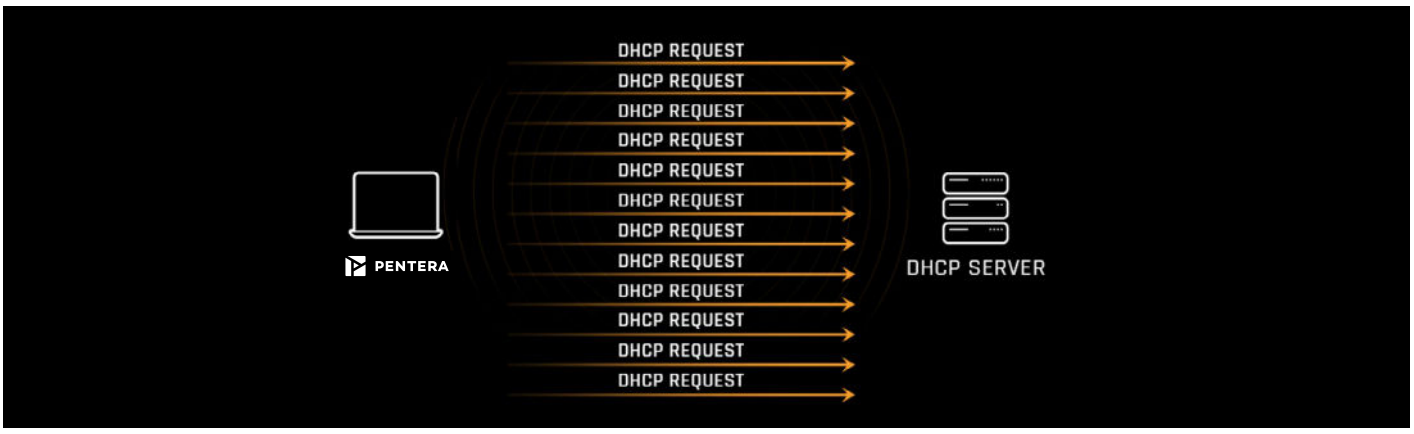
DHCP spoofing attack methods

Needless to say, the DHCP protocol is a powerful network configuration tool that can simplify life for network administrators. A problem arises when unsuspecting network administrators are not aware of all that makes the DHCP protocol susceptible to attack. By default, the DHCP protocol uses no form of authentication and is sent on broadcast, so potentially any device on the network could receive and possibly tamper with the messages.

Let's consider what could happen if an attacker were to combine attacks - for example, DHCP starvation and Rogue DHCP - to launch a Man-In-The-Middle attack (MITM).

DHCP Starvation Attack

In a DHCP starvation attack, an attacker sends the DHCP server multiple DHCPREQUEST messages with spoofed source MAC addresses within a short time span to deplete the server's pool of available IP addresses. Hence, the DHCP server is "starved" and will not respond to new DHCP requests until a new address is available. This prevents a race condition.



A DHCP starvation attack sets the stage for the attacker to pass itself off as the DHCP server and send out spoofed messages to trick other clients on the network.

Rogue DHCP Attack

Now the attacker can set up its own rogue DHCP server, listen for incoming broadcast requests, and send out spoofed responses with malicious configurations. Usually, the attacker will aim to set itself as the DNS server and default gateway for the clients.

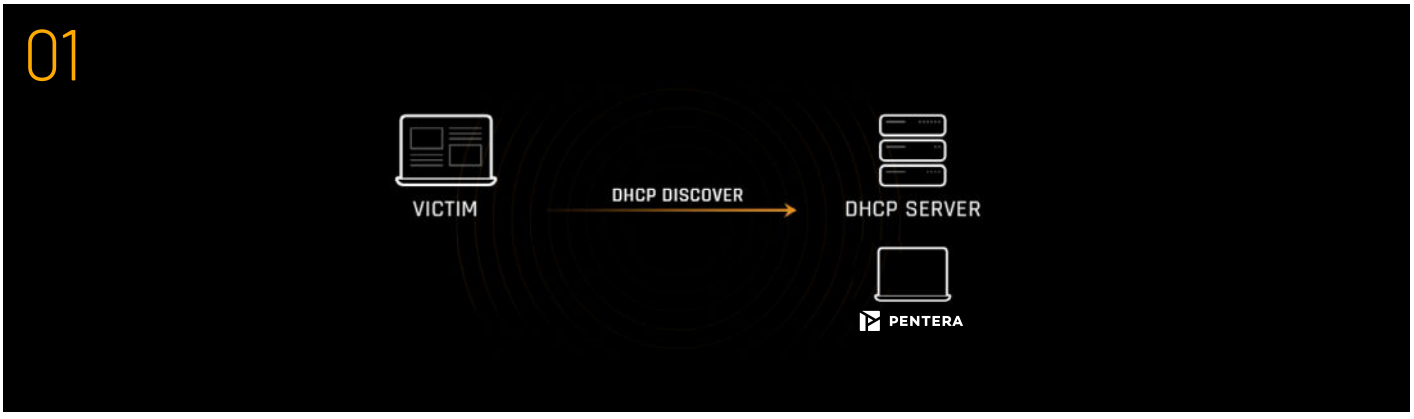
The attacker will open port 53 on its machine for DNS activity, so that every DNS resolution request will reach its machine, allowing it to choose when to answer with its own hostname.

Testing your vulnerability to DHCP spoofing with Pentera

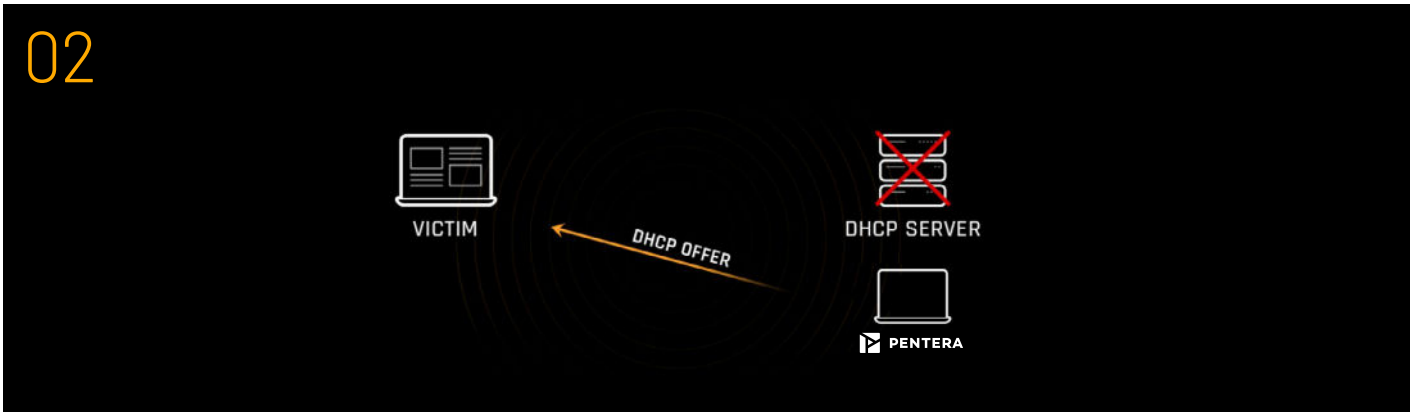
To test this scenario on a network in a safe and controlled environment, we could use Pentera, a fully-automated, safe by-design penetration-testing platform that runs orchestrated, multi-step attack scenarios.

Example Of How Pentera Tests For Vulnerability To DHCP Spoofing

The victim's host sends a broadcast DHCP DISCOVER request –



Since the real DHCP server has been starved and cannot reply, Pentera answers with a DHCP OFFER response –



The victim's host completes the negotiation flow against Pentera, receives fake configurations, and sets the DNS server as the Pentera host. Pentera now masquerades as the victim's DNS server, effectively performing a MITM attack.

At this point, if the victim sends a DNS request for a resource, Pentera can provide its own address and test many attack scenarios, one of them is the forced authentication scenario.

From DHCP spoofing to Forced Authentication

In a forced authentication scenario, Pentera - if you're lucky, or a savvy attacker, if you're less lucky - aims to obtain the victim's NetNTLM hash. Here's how it can happen:

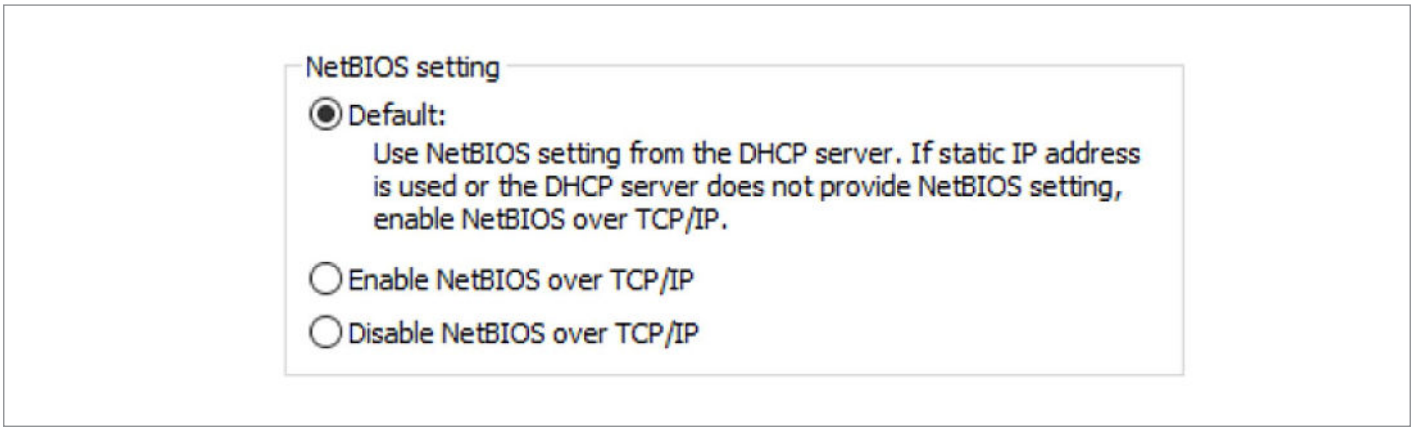
- Let's say the victim tries to reach "www.SomethingThatDoesntExist.com", which is to say a non-existing site, an incorrect URL or an address with a typo or error.
- The attacker's host acting as the victim's DNS server will receive the DNS request for the missing address "SomethingThatDoesntExist.com".
- The attacker will reply that "SomethingThatDoesntExist.com" is the address of the attacker's very own host.
- The victim will send an HTTP request to the attacker.
- The attacker will return an HTTP 401 – unauthenticated response, which will require the client to initiate an authentication process.
- Bingo! The attacker will obtain the client's NetNTLM hash.

Not only NTLM – the risk of enabling NetBIOS using DHCP

We've already seen that an attacker can attach malicious configuration data to DHCP responses. Typically, this means attackers will aim to set themselves as the DNS server and default gateway for the victims. As if that weren't enough, DHCP has control over other configurations as well, including the option to enable NetBIOS.
NetBIOS is a Microsoft Windows protocol that works on top of the TCP/IP transport layer, which provides three distinct services:

- Name service (NBNS) for name registration and resolution - UDP port 137
- Datagram distribution service (NBDGM) for connectionless communication - UDP port 138
- Session service (NBSS) for connection-oriented communication - TCP port 139

On Windows machines, the NetBIOS settings are derived from the DHCP server by default. So unless the client has a static IP address or the DHCP server has an explicit configuration for this option, NetBIOS is enabled by default. Here's a screenshot of the NetBIOS default settings:



In other words, once attackers perform a DHCP spoofing attack, they are in a position to enable the NetBIOS settings and open the port group UDP 137, UDP 138, and TCP 139, thereby exposing the victim to be exploited.

Let's see what this might look like in a real-world example. Compare the network settings before & after a DHCP attack.

BEFORE

```
Ethernet adapter Ethernet0:

   Connection-specific DNS Suffix  . : lab.local
   Description . . . . . . . . . . . : Intel(R) 82574L Gigabit Network Connection
   Physical Address. . . . . . . . . : 00-0C-29-D0-DA-5D
   DHCP Enabled. . . . . . . . . . . : Yes
   Autoconfiguration Enabled . . . . : Yes
   IPv4 Address. . . . . . . . . . . : 192.168.153.5(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Lease Obtained. . . . . . . . . . : Sunday, April 11, 2021 11:26:22 AM
   Lease Expires . . . . . . . . . . : Monday, April 12, 2021 11:26:22 AM
   Default Gateway . . . . . . . . . : 192.168.153.2
   DHCP Server . . . . . . . . . . . : 192.168.153.100
   DNS Servers . . . . . . . . . . . : 192.168.153.100
   NetBIOS over Tcpip. . . . . . . . : Disabled
```

AFTER

```
Ethernet adapter Ethernet0:

   Connection-specific DNS Suffix  . : Attack.me
   Description . . . . . . . . . . . : Intel(R) 82574L Gigabit Network Connection
   Physical Address. . . . . . . . . : 00-0C-29-D0-DA-5D
   DHCP Enabled. . . . . . . . . . . : Yes
   Autoconfiguration Enabled . . . . : Yes
   IPv4 Address. . . . . . . . . . . : 192.168.153.6(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.0.0
   Lease Obtained. . . . . . . . . . : Sunday, April 11, 2021 12:46:09 PM
   Lease Expires . . . . . . . . . . : Sunday, April 11, 2021 12:51:09 PM
   Default Gateway . . . . . . . . . : 172.16.1.1
   DHCP Server . . . . . . . . . . . : 192.168.153.4
   DNS Servers . . . . . . . . . . . : 192.168.153.4
   NetBIOS over Tcpip. . . . . . . . : Enabled
```

Note that after the attack, all of the following have changed:

- DNS suffix
- IP address
- Subnet mask
- Default gateway
- DHCP server – changed to the attacker's address
- DNS server
- NetBIOS over TCP/IP is now enabled

DHCP spoofing can open dangerous ports

In our example, we see that the malicious DHCP used the spoofing attack to open ports 139,138 & 137. Compare the before & after states:

BEFORE

```
C:\Users\yuval>netstat -anot | findstr 13[7-9]

C:\Users\yuval>_
```

AFTER

```
C:\Users\yuval>netstat -anot | findstr 13[7-9]
  TCP    192.168.153.5:139      0.0.0.0:0              LISTENING       4       InHost
  UDP    192.168.153.5:137      *:*                                    4
  UDP    192.168.153.5:138      *:*                                    4
```

Here's what's so significant about the fact that the attack opens ports 139,138 & 137:
The attacker could exploit NBNS on port 137 using tools such as Responder, that respond to broadcast requests with false information.

PORT 445 VS. PORT 139 – MORE SIMILAR THAN YOU MIGHT THINK

Another considerable risk is that the attacker will exploit a host via port 139. Port 445 is often blocked by firewalls due to its popular use in exploitation techniques. A little known fact is that the SMB protocol can also be used on top of the NBSS protocol over port 139. Thus, any SMB vulnerability that can be exploited using port 445, can also be used and potentially exploited over port 139.

For example, a machine with an open 139 port may be susceptible to an EternalBlue attack, which exploits a vulnerability in the SMBv1 protocol. We were able to validate this using Wireshark, the popular network protocol analyzer. Here's a Wireshark screenshot of an EternalBlue attack executed over port 139:

	Destination	Protocol	Source Port	Info
3.4	192.168.153.5	NBSS	43046	Session request, to from
3.5	192.168.153.4	NBSS	139	Positive session response
3.4	192.168.153.5	SMB	43046	Negotiate Protocol Request
3.5	192.168.153.4	SMB	139	Negotiate Protocol Response
3.4	192.168.153.5	SMB	43046	Session Setup AndX Request, NTLMSSP_NEGOTIATE
3.5	192.168.153.4	SMB	139	Session Setup AndX Response, NTLMSSP_CHALLENGE,
3.4	192.168.153.5	SMB	43046	Session Setup AndX Request, NTLMSSP_AUTH, User:
3.5	192.168.153.4	SMB	139	Session Setup AndX Response
3.4	192.168.153.5	SMB	43046	Tree Connect AndX Request, Path: \\
3.5	192.168.153.4	SMB	139	Tree Connect AndX Response
3.4	192.168.153.5	SMB	43046	NT Create AndX Request, FID: 0x4000, Path: netl
3.5	192.168.153.4	SMB	139	NT Create AndX Response, FID: 0x4000

Recommendations

Implementing DHCP Snooping To Mitigate Dhcp Spoofing

Unfortunately, there isn't a simple fix that can hermetically block DHCP spoofing. Protecting your network involves a methodology known as DHCP snooping, a set of techniques aimed at reducing and mitigating the impact of DHCP spoofing attacks. It can be configured on LAN switches to prevent malicious or malformed DHCP traffic and block rogue DHCP servers.

DHCP snooping involves monitoring your DHCP traffic. This can be done by compiling information on hosts which have successfully completed a DHCP transaction in a database of "bindings" and use security or accounting features to monitor the traffic.

First And Foremost - Firewall Rules

Since you know that SMB is a highly vulnerable protocol, you probably already have a firewall rule to block its use. If you do, make sure that this rule also includes the NetBIOS port group - 137, 138 and 139.

If you don't have a firewall rule to block port 445, consider adding one and make sure to add the NetBIOS port group.

Disabling Netbios - The Right Way

Group policy

Group policies don't offer us a quick fix to block DHCP from turning on unwanted NetBIOS settings. In fact, Microsoft does not currently offer the option to disable NetBIOS through Group Policy. Instead, we recommend using GPO scripts to disable NetBIOS on domain clients.
As a network administrator, you can create a GPO setting and use a bat file to run WMIC commands on local and remote computers. WMIC (Windows Management Instrumentation Command-line) is a Windows scripting interface for accessing and managing data from the Command Prompt.
For example, this WIC command searches for all network interfaces that have NetBIOS enabled and disables them:

```
wmic nicconfig where
(TcpIpNetbiosOptions!=Null and
TcpIpNetbiosOptions!=2) call
SetTcpIpNetbios 2
```

Registry

The registry value used to disable NetBIOS is dependent on the interface UUID, so it may not be as simple to manage it using GPOs. However, if you want to use the registry on a specific client, here is the value:

```
HKLM\SYSTEM\CurrentControlSet\Services\NetBT\
Parameters\Interfaces\<Interface_UUID>
```

If you go this route, you'll need to create a new REG_DWORD value named "NetbiosOptions" and set it to 0x00000002 for each interface you want to disable.

Conclusion

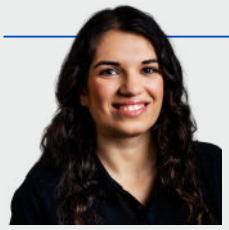
We hope you're now ready to review your DHCP policies and firewall rules, apply DHCP snooping in your network systems and devices, and validate your security posture for compliance.

Contact Pentera to schedule a Proof of Value in your environment and begin your road to security posture validation today.

Sources And Additional Reading

- https://www.techrepublic.com/blog/data-center/dns-changer-trojan-latest-variant-is-certainly-unique/
- https://isc.sans.org/diary/Rogue+DHCP+servers/5434
- https://attack.mitre.org/techniques/T1557/002/

About the author



Yuval Lazar has earned recognition among the Top 25 Women in Cybersecurity for 2020. After leading research projects in the elite IDF 8200 Unit, Yuval joined Pentera as a Senior Security Researcher. Driven by her love for offensive cyber research, Yuval is constantly searching for new complex attack vectors.

About Pentera

PENTERA

Pentera is the category leader for Automated Security Validation, allowing every organization to test with ease the integrity of all cybersecurity layers, unfolding true, current security exposures at any moment, at any scale. Thousands of security professionals and service providers around the world use Pentera to guide remediation and close security gaps before they are exploited.

For more info visit: pentera.io

PENTERA | Pentera Labs™ Research Series

Credential Harvesting Using DHCP Spoofing