<Client Name>

Sample Internal Penetration Test Report

Dd Month yyyy v1.0

Presented by

WhiteRook Cyber

P: 1300 794 777
E: info@whiterookcyber.com.au
W: whiterookcyber.com.au

Document Control

Document Identification

File Name	WRC-SampleClient-SampleInternalPenetrationTestReport
Version	1.0

Document Contributors

Name	Role	Phone Number	Email Address
Billy Cody	Assessment Lead	[redacted]	[redacted]

Revision History

Version	Date Released	Author	Changes
0.1	20 February 2024	BillyCody	Initial draft
0.2	21 February 2024	[redacted]	Technical QA
0.3	21 February 2024	[redacted]	Presentation QA
1.0	22 February 2024	Billy Cody	First release

Document Distribution

Name	Company	Date	Version
Sample Contact	Sample Client	22 February 2024	1.0

Table of Contents

Executive Summary

WhiteRookCyber performed a penetration test on Sample Client's internal and wireless infrastructure. Testing was performed between 05 February 2024 and 15 February 2024 at the Sample Client head office at 123 Fake St, Brisbane.

A physical intrusion was performed on 10 February 2024 to test staff awareness and physical security controls at 123 Fake St, Brisbane. The consultant was able to enter the office and set up in a meeting room unchallenged. From here, they were able to access the internal network through a secondary Ethernet port on an IP phone. After an hour, the consultant entered the wider office and worked at a desk for an additional hour before being questioned by staff.

The internal network was able to be fully compromised by an unauthenticated attacker with physical network access. The consultant was able to obtain credentials, move laterally through the network, escalate their privilege, and retrieve the credential data of all Sample Client corporate users.
Many of the internal network attacks were possible due to the default settings within Active Directory, which is highly vulnerable without hardening. It is recommended that the prescribed remediation advice for each vulnerability is applied. It was possible for the consultant to laterally move through the network with limited restrictions due to limited network segregation. This allowed access to store and server subnets without restriction.

Of particular importance, everyday user accounts with high-privileges were compromised and used to facilitate further attacks into the network. It is highly recommended to continue Sample Client's efforts to enact a least-privilege model and use separate administrator accounts.

Despite the compromise, Sample Client's endpoint detection and response software was able to rapidly alert the team to suspicious behaviour and largely prevented malware execution. Password hygiene was exceptional, with less than 4% of unique passwords able to be cracked and retrieved in plaintext. No issues were identified in Sample Client wireless infrastructure.

This report contains a technical summary with recommendations and detailed technical findings that describe the impact of each issue and how they can potentially be mitigated. If further clarification is required on the contents of this report, please reach out to WhiteRookCyber.

We appreciate the opportunity to help improve your security.

Billy Cody, Offensive Security Director
xxxxxxxxxxxxxxxxxxxxxxxx
[redacted]

Technical Summary

A total of thirteen (13) findings have been included in this report. This includes seven (7) high risk, one (1) medium risk, and four (4) low risk vulnerabilities. One (1) finding has been included for informational purposes.

Finding #	Title	Risk
1	User Accounts with Domain Admin Privileges	HIGH
2	Limited Network Segregation	HIGH
3	Name Resolution Protocols Poisoning Attack	HIGH
4	DHCP Poisoning	HIGH
5	DHCPv6 Poisoning	HIGH
6	NTLM Relaying Attack	HIGH
7	Reused Local Administrator Password	HIGH
8	Overpermissive File Shares	Medium
9	Kerberoasting Attack	Low
10	Domain Password Policy Analysis	Low
11	Missing RDP Hardening	Low
12	Unencrypted Telnet	Low
13	Unsupported Software	Informational

Scope

The internal network was accessed from the Sample Client head office at 123 Fake St, Brisbane. A regular client switchport was used. No credentials were provided for the internal engagement.
The following wireless networks were in scope:

- SampleClient_Staff
- SampleClient_Guest
- SampleClient_IoT

Access was provisioned for each wireless network after unauthenticated techniques failed to grant the consultant access.

Recommendations and Conclusions

The Sample Client internal network suffered from multiple Active Directory specific vulnerabilities, arising from a lack of network hardening. Many of these vulnerabilities are present in all Active Directory environments by default. These should be mitigated with the appropriate Group Policy modifications prescribed in this report.

User privileges were inconsistent, with some administrators having separate administrator accounts and some with high-privileges enabled on their user account. Sample Client should continue to implement least-privilege across the environment. Although LAPS is in use to manage local Administrator account passwords, password reuse was identified across 20+ servers and workstations. LAPS should be enabled on these machines. Multiple file shares containing sensitive information and with write privileges were able to be accessed with a low-privilege account. This access should be audited and amended.

Network segregation did not appear to be implemented. From the corporate network, full access to server and store subnets was possible. This may indicate stores also have full

corporate network access. A program of work to segregate these networks should be implemented.

Sample Client's password policy has resulted in exceptional user password hygiene. Less than 4% of unique account passwords were able to be cracked and the plaintext representation retrieved. The endpoint detection and response software in use was able to detect and prevent malicious activity and malware execution, although this was possible to bypass during the engagement.

Attack Walkthrough

The attack walkthrough demonstrated how the discovered vulnerabilities can be chained to exploit the Sample Client internal network, and the potential impacts an attacker could have.

Unprivileged

Network access

Physical network access could be achieved by physically intruding into the Sample Client head office. The consultant walked past reception after exciting the elevator without being. No secure doors prevented entry to the main office space.

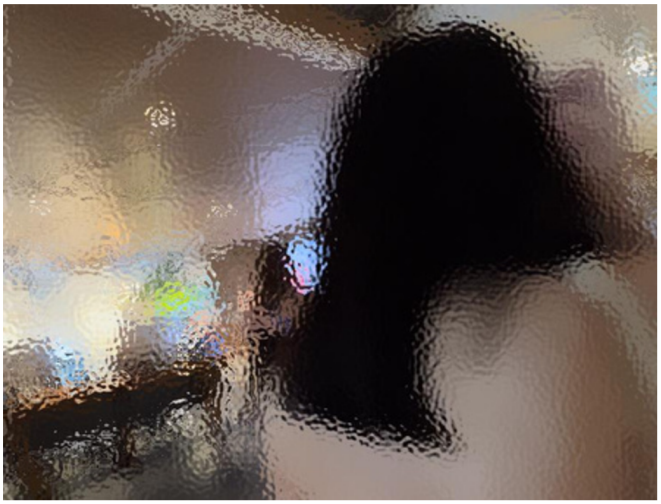

Figure 1 – Physically intruding into Sample Client head office

The consultant was able to gain internal network access by plugging into an IP phone located in a meeting room.

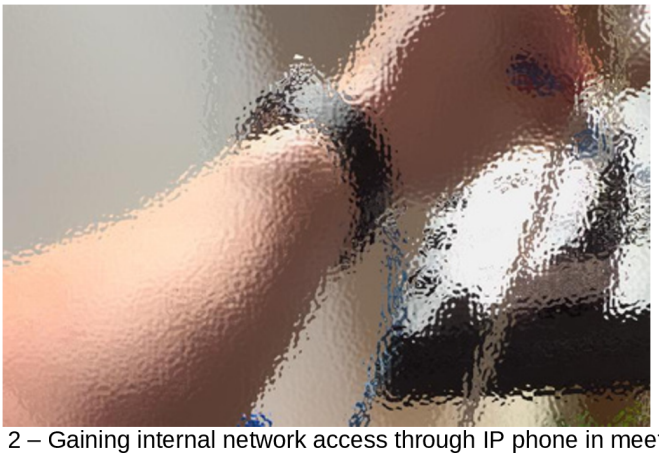

Figure 2 – Gaining internal network access through IP phone in meeting room

Obtaining credentials

Once connected to the network, an attacker will seek credentials to perform lateral movement and privilege escalation. This was able to be achieved through name resolution protocol poisoning and NTLM relaying. This exploit allowed the consultant to obtain authenticated SMB

sessions without requiring credentials and relay sessions to LDAP to create a computer account with known credentials.

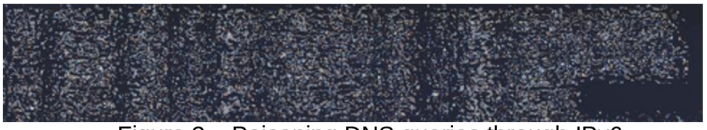

Figure 3 – Poisoning DNS queries through IPv6

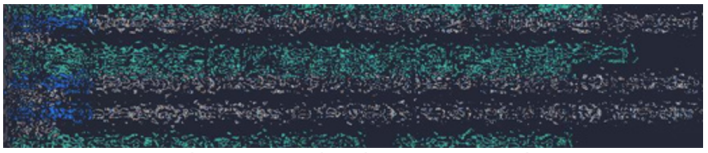

Figure 4 – Poisoning DHCP queries

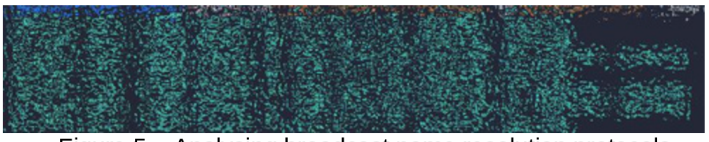

Figure 5 – Analysing broadcast name resolution protocols

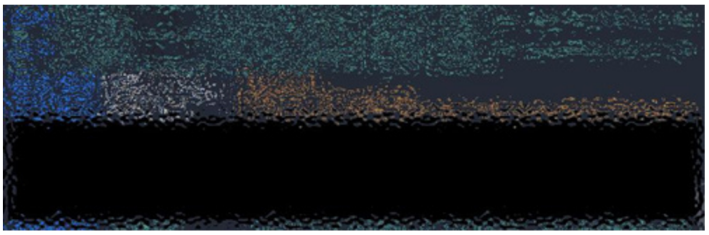

Figure 6 – Password hash received from broadcast name resolution protocol poisoning

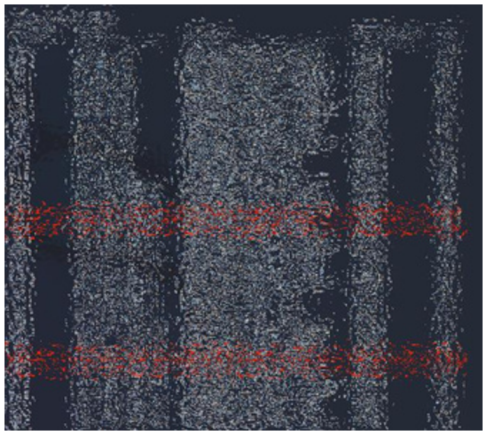

Figure 7 – Authenticated SMB sessions after NTLM relay attack, some with local Administrator privilege

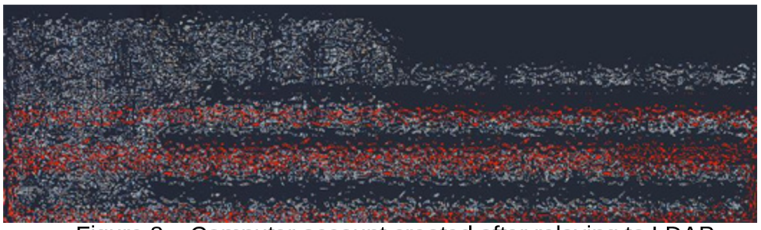

Figure 8 – Computer account created after relaying to LDAP

With credentials in hand, the attacker could begin analysing the Active Directory environment. The password policy was retrieved, a list of users, and detailed information about Active Directory.

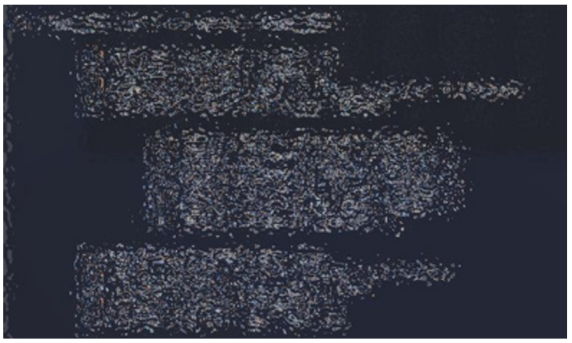

Figure 9 – Retrieved domain password policy

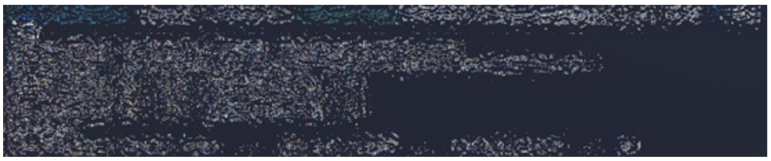

Figure 10 – Retrieving list of domain users through relayed SMB session

Figure 11 – Enumerating Active Directory information with a compromised account through bloodhound

Low-privilege credentials were able to be used to access numerous file shares containing potentially sensitive information, with some shares offering read/write access.

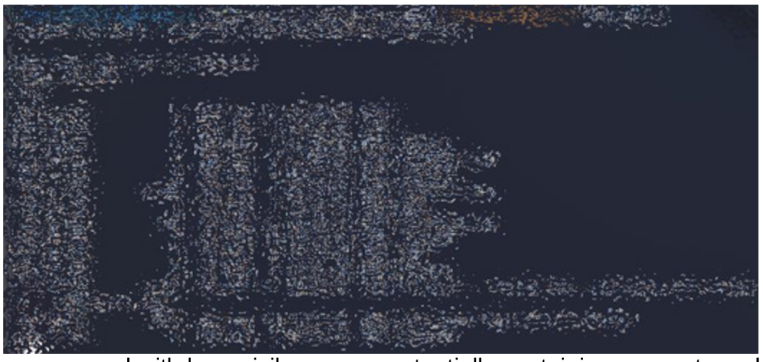

Figure 12 – Share accessed with low-privilege user, potentially containing accounts and payroll information

Privilege Escalation

Using relayed accounts with local Administrator privilege, it was possible to retrieve password hashes and plaintext passwords.

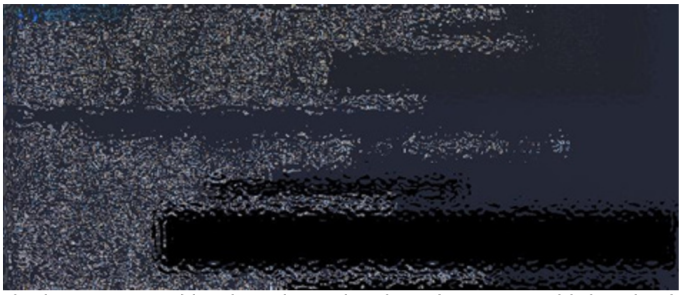

Figure 13 – Retrieving password hashes through relayed account with local Administrator privilege

A subset of servers and workstations were found to reuse local Administrator passwords.

Using pass-the-hash, it was possible to use compromised password hashes to gain access to additional systems.

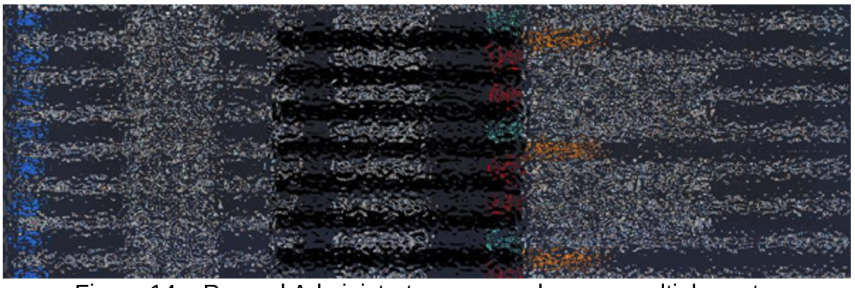

Figure 14 – Reused Administrator password across multiple systems

Due to the strong password policy, it was unlikely for the consultant to retrieve Domain Admin credentials in plaintext through password cracking. Direct compromise of a Domain Admin session on a machine was required. Active logins were discovered through bloodhound.

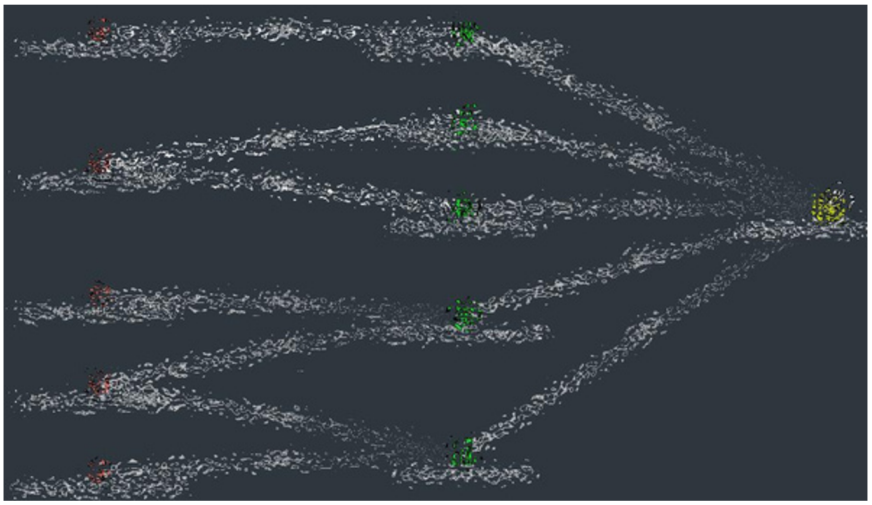

Figure 15 – Active Domain Admin logins in bloodhound

An NTLM relay attack was used to target these machines. Once a local Administrator session was compromised, it was possible to gain a semi-interactive shell using impacket-wmiexec. To compromise a Domain Admin session, it was necessary to bypass endpoint detection and response software to upload malware. This was done by creating an exclusion in the local antivirus policy.

Figure 16 – Creating exclusion in antivirus policy

Figure 17 – Uploading open-source malware

Figure 18 – Received C2 malware session

With command and control achieved on the machine, it was possible to use the permissions of a Domain Admin to create a backdoor Domain Admin user.

Figure 19 – Process running as a Domain Admin

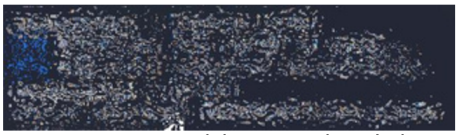

Figure 20 – Compromising Domain Admin process

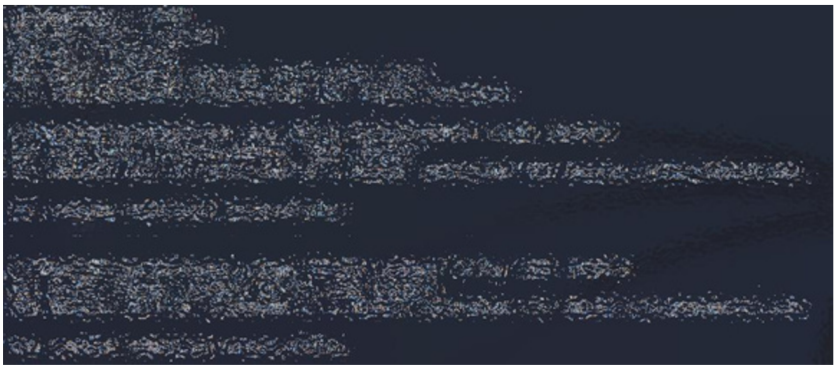

Figure 21 – Creating backdoor Domain Admin using compromised Domain Admin process

The attacker has now fully compromised the Active Directory domain, granting full access to any domain-joined asset.

Post Exploitation

A Domain Admin has full control over an Active Directory environment, meaning these accounts are actively hunted for by attackers. After compromising credentials, an attacker can simply disable antivirus across the organisation and deploy ransomware.

With the domain fully compromised, it was possible to retrieve the password hashes for all Sample Client corporate users. Less than 4% of these passwords were cracked due to the strong password policy in use.

Technical Findings

Finding 1User Accounts with Domain Admin Privileges

Description

Risk	HIGH
Impact	Severe
Likelihood	Likely
Type	Network

The implementation of least-privilege is important for preventing compromise and limiting the scope of a successful compromise. By separating everyday user accounts from high-privilege administrator accounts, even the direct compromise of an administrator user can prevent privilege escalation.

Users accounts were found to have Domain Admin privilege, indicating least-privilege is not implemented widely within the environment.

Impact

Three user accounts were found to be members of the Domain Admins group. Using a name resolution protocol poisoning attack combined with an NTLM relay attack (detailed in separate findings), it was possible to relay the sessions of Domain Admin accounts. These sessions were used to gain privileged access to servers and eventually led to total domain compromise.

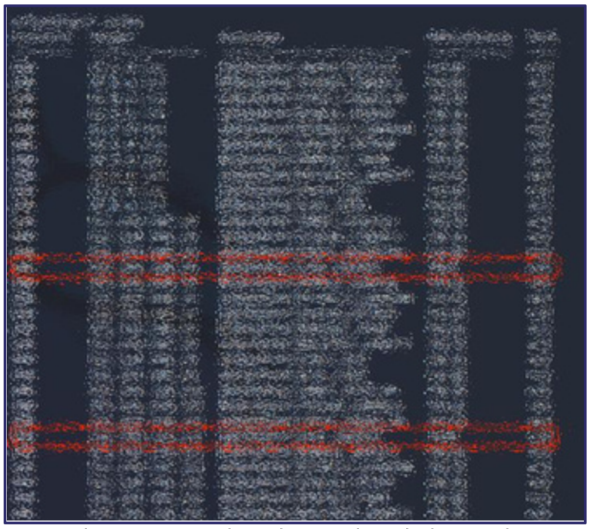

Figure 22 – Relayed Domain Admin sessions

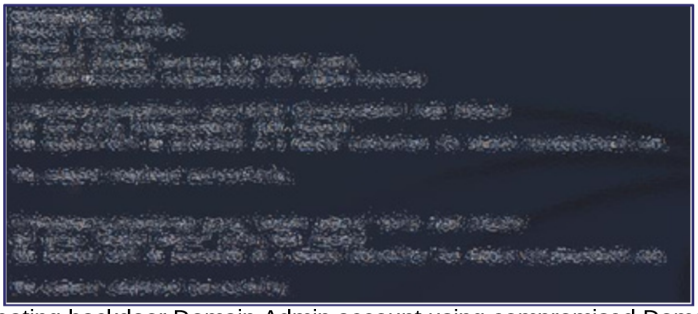

Figure 23 – Creating backdoor Domain Admin account using compromised Domain Admin session

Mitigation

Least-privilege should be required for the implementation of all high-privilege administrator accounts.
Group Policy Objects should be created and linked to these accounts to enable the following:

- Deny access to this computer from the network
- Deny log on as a batch job
- Deny log on as a service

This will create additional difficulties for an attacker to abuse a high-privilege account. Administrators will require an interactive low-privilege session to elevate their privilege. Additionally, add the Domain Admins group to the Protected Users group.

References

- What Is Least Privilege Access? | Okta Australia
- Implementing Least-Privilege Administrative Models | Microsoft Learn
- Appendix G: Securing Administrators Groups in Active Directory | Microsoft Learn
- Guidance about how to configure protected accounts | Microsoft Learn

Affected

The following user accounts are Domain Admins:

- User_1
- User_2
- User_3

Finding 2Limited Network Segregation

Description

Risk	HIGH
Impact	Moderate
Likelihood	Likely
Type	Network

When securing an organisation's network, a defence-in-depth approach is best. A single vulnerability or misconfiguration in the external infrastructure (or a successful phishing attack) could allow an attacker access to the internal network. In this case, logical segmentation and segregation of the network can limit the access and options available to an attacker to a minimum.
The in-scope internal network was determined to have limited network segregation in place.

Impact

The consultant was able to access the following high risk networks:

- Server subnets (10.0.0.0/24, 10.0.1.0/24, 10.0.2.0/24, 10.0.3.0/24)
- Store subnets (192.168.0.0/16)

This allowed access to sensitive and high-risk systems.

These networks are not required by the majority of the organisation's users, and they should not be accessible to them.

Mitigation

Adopt a zero-trust networking model. No internal networks should be able to communicate with each other (deny all inbound traffic) unless there is a clear requirement. Instead of granting access to entire subnets, limit access to the host IPs and ports that are required.

For users that routinely require access to these networks, consider a management VLAN accessible only to administrators (such as a separate WiFi network, certain Ethernet ports within locked offices, or an internal VPN) or implement bastion hosts that are hardened and bridge the required networks.

References

- What is Zero Trust security? | Cloudflare
- Implementing Network Segmentation and Segregation | Cyber.gov.au

Finding 3Name Resolution Protocols Poisoning Attack

Description

Risk	HIGH
Impact	Severe
Likelihood	Possible
Type	Network

Link-Local Multicast Name Resolution (LLMNR) and Netbios Name Service (NBT-NS) are name resolution protocols enabled by default in Microsoft Windows. These protocols serve as backup name resolution protocols if DNS resolution fails. These protocols broadcast to the local network to resolve the requested name.

An attacker able to intercept this traffic (usually by being on the same local network) is able to exploit this behaviour by returning malicious IP addresses to the victim in a name resolution protocol poisoning attack. As LLMNR and NBT-NS are usually used to resolve names for Microsoft processes (such as file shares), the victim will generally attempt to authenticate to the malicious IP, allowing the attacker to retrieve password hashes or perform NTLM relay attacks.

Successful attacks can result in password hash disclosure, or when combined with other vulnerabilities can result in privilege escalation and remote code execution. Open source tools, such as responder, exist to perform these attacks.

Impact

The local network was analysed, revealing the use of LLMNR and NBT-NS protocols.

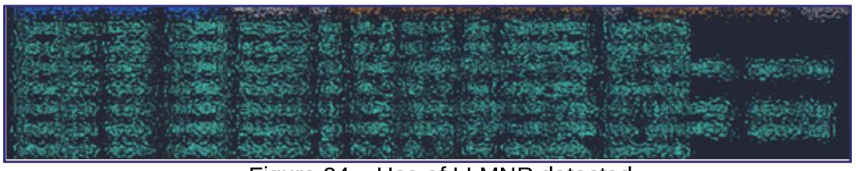

Figure 24 – Use of LLMNR detected

Various user hashes were able to be intercepted during a poisoning attack.

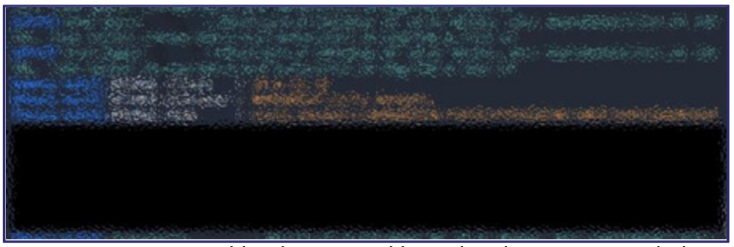

Figure 25 – User password hash captured by poisoning name resolution protocols

This vulnerability was also used to perform an NTLM relay attack (detailed in a separate finding).

Mitigation

LLMNR can be disabled via Group Policy, under the following policy:

Computer Configuration > Administrative Templates > Network > DNS Client

Set the following policy:

Turn off Multicast Name Resolution - Enabled

NBT-NS is unable to be directly disabled via Group Policy, necessitating the use of PowerShell. The following can be added to a logon script to automatically disable NBT-NS domain-wide. PowerShell Scripts can be added under:

Computer Configuration > Policies > Windows Settings > Scripts > Startup > PowerShell Scripts

The following script iterates over each network interface and disables NBT-NS:

```
$regkey = "HKLM:SYSTEM\CurrentControlSet\services\NetBT\Parameters\Interfaces"

Get-ChildItem $regkey | foreach { Set-ItemProperty -Path "$regkey\$($_.pschildname)" -Name NetbiosOptions -Value 2 -
Verbose}
```

References

- Local Network Attacks: LLMNR and NBT-NS Poisoning | MITRE ATT&CK
- GitHub - lganx/Responder
- Network Sniffing | MITRE ATT&CK
- Adversary-in-the-Middle: LLMNR/NBT-NS Poisoning and SMB Relay | MITRE ATT&CK

Finding 4DHCP Poisoning

Description

Risk	HIGH
Impact	Severe
Likelihood	Likely
Type	Network

Dynamic Host Configuration Protocol (DHCP) is a broadcast protocol used by to automatically configure hosts with IP addresses and settings, such as DNS servers and the domain name. An attacker on the same network can reply to DHCP requests from clients joining the network (a DHCP poisoning attack), allowing the attacker to set the victim's DNS server to any IP address of their choosing. This effectively creates a person-in-the-middle position, allowing an attacker to perform various attacks.

This can be used to create the conditions required for an NTLM relay attack. Open source tools, such as Responder, exist to perform this attack.

Impact

It was possible to poison DHCP requests to trick Windows machines into authenticating to the consultant's machine. This resulted in the disclosure of password hashes, which were cracked. The plaintext passwords were used to perform further attacks on the environment. It was possible to perform NTLM and LDAPS relay attacks due to disabled SMB and LDAPS signing, detailed in separate findings.

Figure 26 – Performing a DHCP poisoning attack with responder

Mitigation

Enable DHCP snooping on all compatible switches.

References

- DHCP Poisoning | Ethical Hacking - GreyCampus
- Adversary-in-the-Middle: LLMNR/NBT-NS Poisoning and SMB Relay | MITRE ATT&CK
- GitHub - lgandx/Responder: Responder is a LLMNR, NBT-NS and MDNS poisoner, with built-in HTTP/SMB/MSSQL/FTP/LDAP rogue authentication server supporting NTLMv1/NTLMv2/LMv2, Extended Security NTLMSSP and Basic HTTP authentication.
- Configuring DHCP Snooping - Cisco

Finding 5DHCPv6 Poisoning

Description

Risk	HIGH
Impact	Severe
Likelihood	Likely
Type	Network

Modern Windows hosts prefer to communicate over IPv6. To configure IPv6, DHCPv6 configuration requests are broadcast to the local network. An attacker on the same network can reply to these requests, allowing them to set the victim's DNS server to any IP address of their choosing. This effectively creates a person-in-the-middle position, allowing an attacker to perform various attacks.

This can be used to create the conditions required for an NTLM relay attack. Open source tools, such as mitm6, exist to perform this attack.

Impact

The mitm6 tool was used to poison IPv6 DHCP requests.

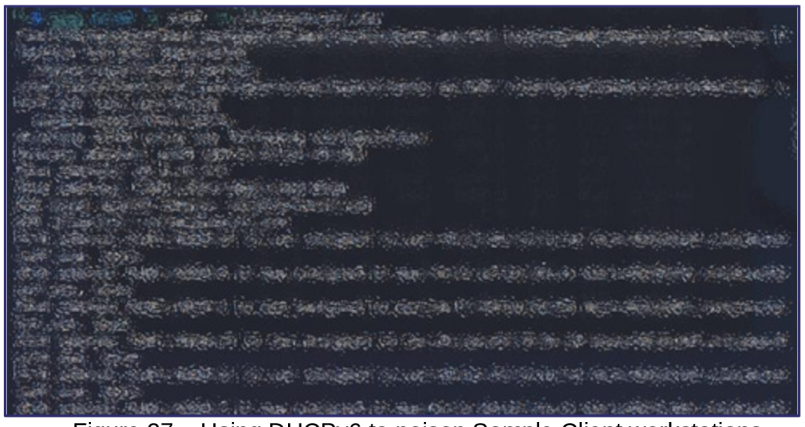

Figure 27 – Using DHCPv6 to poison Sample Client workstations

DNS requests for the internal domain were intercepted and the IP of the consultant's laptop was returned instead.

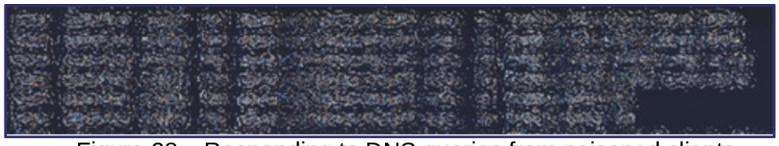

Figure 28 – Responding to DNS queries from poisoned clients

This was used to perform an NTLM relaying attack (detailed in a separate finding).

Mitigation

Certain networking devices (such as Cisco) implement protections against this attack, such as DHCPv6 Guard. Where possible, enable this for all networks.

Alternatively, IPv6 can be disabled on all domain-joined hosts by creating a PowerShell logon script through Group Policy:

```
Set-NetIPInterface -AddressFamily IPv6 -InterfaceIndex $(Get-NetIPInterface -AddressFamily IPV6 | Select-Object -ExpandProperty InterfaceIndex) -RouterDiscovery Disabled -Dhcp Disabled
```

This may cause connectivity issues with certain Azure functionality, testing this script before deployment is advised.

References

- Adversary-in-the-Middle: LLMNR/NBT-NS Poisoning and SMB Relay | MITRE ATT&CK
- GitHub - dirkjanm/mitm6: pwning IPv4 via IPv6
- mitm6 - compromising IPv4 networks via IPv6 - Fox-IT International blog
- IPv6 First-Hop Security Configuration Guide - DHCP—DHCPv6 Guard [Cisco Cloud Services Router 1000V Series] - Cisco
- Mitigating IPv6 Poisoning Attacks | LMG Security

Finding 6NTLM Relaying Attack

Description

Risk	HIGH
Impact	Severe
Likelihood	Likely
Type	Network

SMB signing is a security mechanism within the SMB protocol that prevents tampering of the connection between a server and client. Without SMB signing, it is possible to perform an NTLM relay attack against SMB.

An NTLM relay attack occurs when an attacker is able to intercept a legitimate authentication request (usually through a person-in-the-middle position, achievable through attacks like LLMNR or NBT-NS protocol poisoning). The attacker can forward the tampered authentication requests to the victim server, authenticating themselves rather than the victim client.

An LDAP relay attack is a specialised NTLM relay attack that is possible when LDAP channel binding and signing are disabled (the default policy). This can be abused to create accounts, escalate the privilege of accounts, and retrieve domain information (depending on the privilege of the relayed account).

Depending on the privileges of the relayed user, a successful NTLM relay attack can result in access to privileged file shares, credential dumping attacks, and remote code execution. Open source tools, such as impacket-ntlmrelayx, exist to perform this attack.

Impact

128 servers and workstations were identified without SMB signing enabled, indicating this may be a domain-wide issue. This was exploited by the consultant to retrieve credentials from dozens of servers to escalate their privilege within the domain.

At first, the consultant was only able to relay user credentials. This was able to be exploited by retrieving a full list of usernames within the domain for password spraying attacks.

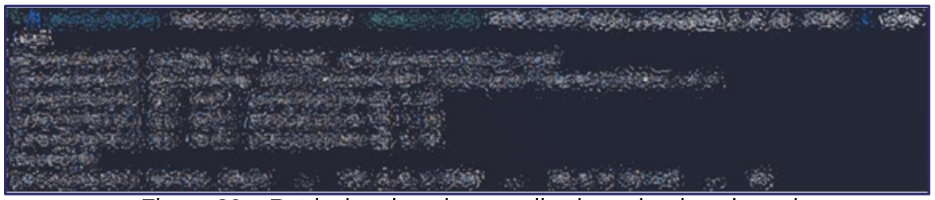

Figure 29 – Retrieving domain users list through relayed session

A user account with local administrator privileges was also able to be relayed due to excessive privileges on user accounts, detailed in another finding. This was exploited to perform credential dumping attacks against several servers.

Figure 30 – Relayed local administrator session

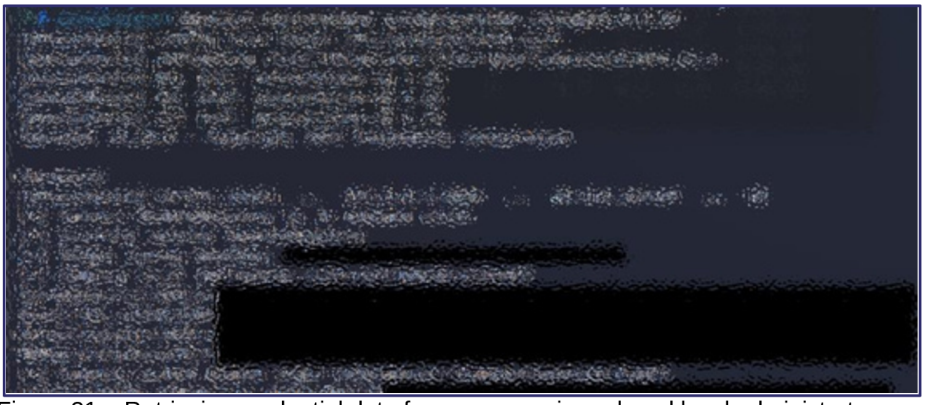

Figure 31 – Retrieving credential data from server using relayed local administrator session

LDAP signing and channel binding was not enforced on any Domain Controllers.

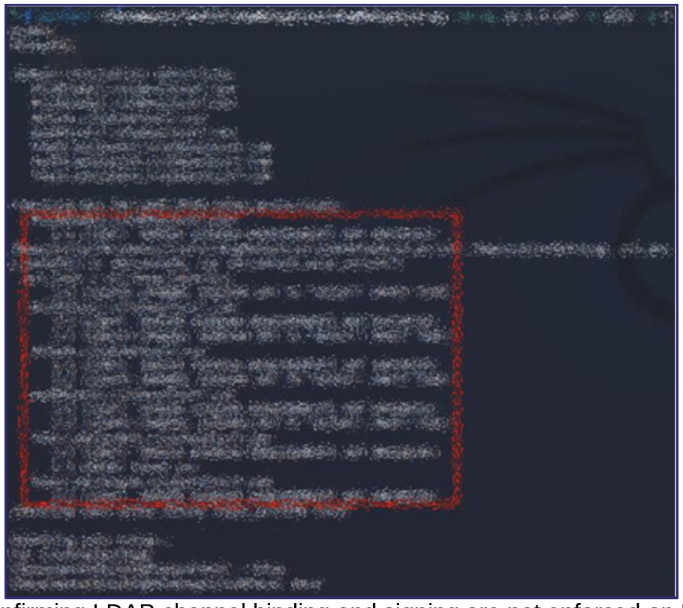

Figure 32 – Confirming LDAP channel binding and signing are not enforced on Domain Controllers

A successful LDAP relay attack resulted in a domain computer account with a known password being created, allowing for credentialed attacks.

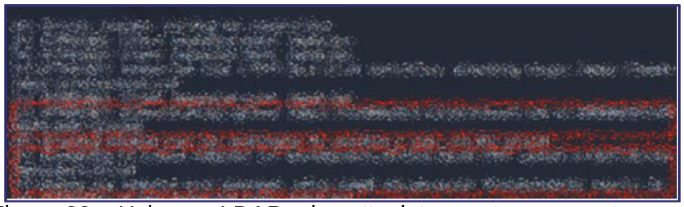

Figure 33 – Using an LDAP relay attack to create a computer account

Mitigation

SMB signing is able to be enabled through Group Policy to apply to all domain-joined computers. The relevant policies are found under:

Computer Configuration > Windows Settings > Security Settings > Local Policies > Security Options

Set the following policies and values:

- Microsoft network client: Digitally sign communications (always) - Enabled
- Microsoft network client: Digitally sign communications (if server agrees) - Enabled
- Microsoft network server: Digitally sign communications (always) - Enabled
- Microsoft network server: Digitally sign communications (if client agrees) - Enabled

Additionally, for environments using Active Directory Certificate Services (ADCS), Microsoft has additional guidance for mitigating NTLM relay attacks targeting ADCS.

LDAP channel binding and signing can be enabled through Group Policy to apply to all Domain Controllers. The relevant policies are found under:

Computer Configuration > Policies > Windows Settings > Local Policies > Security Options

Set the following policies and values:

- Domain controller: LDAP server signing requirements - Require Signing
- Domain controller: LDAP server channel binding token requirements – Always

References

- Overview of Server Message Block signing - Windows Server | Microsoft Learn
- The worst of both worlds: Combining NTLM Relaying and Kerberos delegation - dirkjanm.io
- GitHub - SecureAuthCorp/impacket/ntlmrelayx.py
- KB5005413: Mitigating NTLM Relay Attacks on Active Directory Certificate Services (AD CS)
- Relaying credentials everywhere with ntlmrelayx – Fox-IT International blog
- Adversary-in-the-Middle: LLMNR/NBT-NS Poisoning and SMB Relay | MITRE ATT&CK

Finding 7 Reused Local Administrator Password

Description

Risk	HIGH
Impact	Severe
Likelihood	Possible
Type	Host

Local Administrator accounts can, by default, perform any action on a server or workstation. As part of privilege escalation and lateral movement, these accounts are invaluable to attackers.

When passwords for the local Administrator are reused across multiple servers or workstations, attackers can move laterally between these affected machines. Due to technical flaws in the NTLM protocol, only the local Administrator password hash is required to authenticate between these machines (known as a "Pass-the-Hash" attack). After gaining SYSTEM level access and retrieving the Security Account Manager (SAM) database, an attacker can retrieve the Administrator's NTLM password hash and use it to further compromise the environment without needing to crack the password. Open source tools, such as impacket-psexec, can be used to perform Pass-the-Hash attacks, leading to remote code execution on machines using the local Administrator password hash.

Impact

Multiple users with local Administrator privileges fell victim to a name resolution poisoning and NTLM relay attack (detailed in separate findings), providing the consultant with a local Administrator session. This was exploited to perform an LSASS credential dump, returning the local Administrator's NTLM password hash.

Figure 34 – Relayed user account with local Administrator privileges

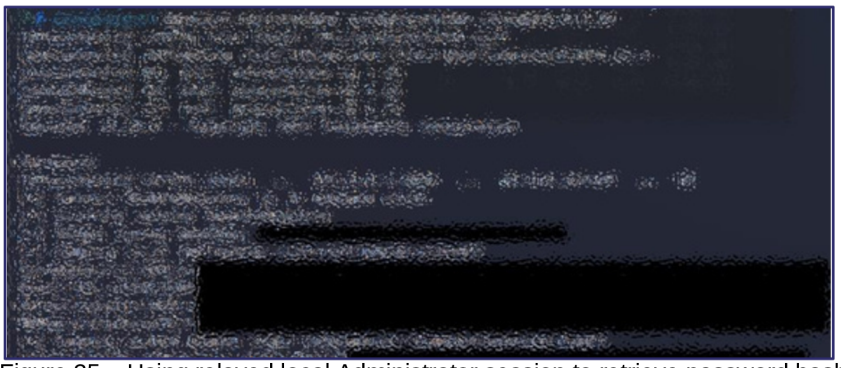

Figure 35 – Using relayed local Administrator session to retrieve password hashes

The reuse of four different local Administrator passwords resulted in the compromise of one machine leading the compromise of an additional four (4) to thirteen (13) machines, depending on the account.

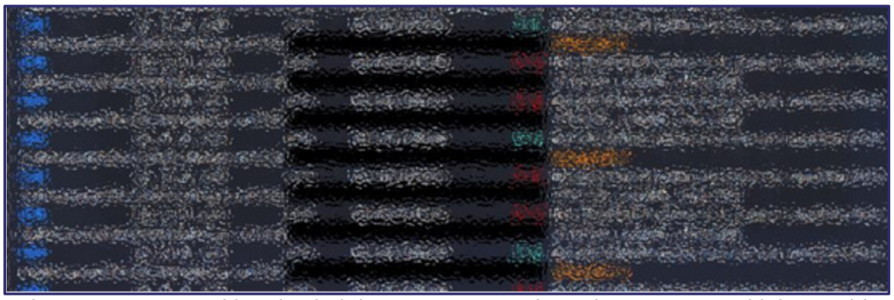

Figure 36 – Reused local Administrator password used to access multiple machines

Mitigation

Microsoft Local Administrator Password Solution (LAPS) is a free password manager provided by Microsoft that integrates the management of local Administrator accounts across the domain with Active Directory. These passwords are randomised and unique across machines, preventing the compromise of one endpoint leading to lateral movement. Administrators requiring local Administrator passwords can consult the LAPS UI or request the password through PowerShell. Consult the referenced Microsoft documentation on how to install, configure, and use LAPS.

References

- How to Configure Microsoft Local Administrator Password Solution (LAPS)
- GitHub - SecureAuthCorp/impacket/psexec.py
- OS Credential Dumping: Security Account Manager | MITRE ATT&CK
- Remote Services: SMB/Windows Admin Shares | MITRE ATT&CK
- Brute Force: Password Spraying | MITRE ATT&CK
- Use Alternate Authentication Material: Pass the Hash | MITRE ATT&CK

Affected

The following local Administrator accounts reused passwords

- Administrator (variant 1: 5 uses
  - 10.0.0.1-5
- Administrator (variant 2): 8 uses
  - 10.0.0.6-13
- Administrator (variant 3): 13 uses
  - 10.0.0.14-26

Finding 8 Overpermissive File Shares

Description

Risk	Medium
Impact	Moderate
Likelihood	Likely
Type	Network

File shares within an organisation generally limit which users are able to access them through an access control list. When an access control list is poorly defined or non-existent, an attacker with compromised credentials may be able to compromise sensitive data. Additionally, if the shares have write permissions, an attacker may be able to overwrite configuration files and achieve remote code execution. Additionally, malicious files may be uploaded to gather credentials or spread malware.

The impact of this vulnerability is dependent on the contents of the affected shares. Open source tools, such as crackmapexec and impacket-smbclient, can be used to discover and exploit access to file shares.

Impact

The user "relay-comp$" was used to perform a file share audit within the Sample Client domain. This was due to its membership only to the "Domain Users" group, a default group that all users belong to.

Access to sensitive files was possible, as well as read/write access to several sensitive shares. An example is shown below.

Figure 37 – Share containing potential payroll and accounting information

Mitigation

An audit of all file shares within the domain should be conducted. Access control lists containing "Everyone" or "Domain Users" should be remediated as a high priority. Additionally, sensitive files should be removed or moved to more secure shares, and files containing passwords should be stored in the organisation's password manager.

References

- GitHub - Porchetta-Industries/CrackMapExec
- GitHub - SecureAuthCorp/impacket/smbclient.py
- Export Remote Shares and Folder permissions using PowerShell – TheSleepyAdmins
- Taint Shared Content | MITRE ATT&CK
- Network Share Discovery | MITRE ATT&CK

Affected

The following shares were able to be accessed:

IP address	Share	Permissions	Description
10.0.0.1	Finance	READ	Finance Department Document Storage
10.0.0.2	Backups	READ, WRITE	Document and server image backups
10.0.0.3	Database	READ	Database storage

Finding 9Kerberoasting Attack

Description

Risk	Low
Impact	Severe
Likelihood	Rare
Type	Network

When a user wants to interact with a Windows domain service, one method of authenticating to the service is through Kerberos. The user makes an authentication request to a domain controller and receives a Kerberos ticket encrypted with service account's password. This ticket is then passed from the user's computer to the service. The service itself inspects the ticket, allowing or denying the user access.

Due to the architecture of this authentication scheme, any domain user may request a Kerberos ticket for a service if a Service Principal Name (SPN) is configured for it. As it is encrypted with the service account's password, it is possible to crack the ticket and retrieve the plaintext password in a Kerberoasting attack.

The effects of Kerberoasting depend on the privilege of the service logon account mapped to the affected SPN. A Kerberoasting attack also depends on the strength of the service's password, as with most password cracking attacks. Additionally, the encryption used can affect the time taken for a successful password crack. By default, the weak RC4 encryption scheme is used. Open source tools, such as impacket-GetUserSPNs, exist to perform this attack.

Impact

Four (4) active accounts with SPNs were identified within the domain. Tickets with RC4 encryption for each service were retrieved and underwent password cracking. However, no accounts were able to have their password retrieved in plaintext.

Mitigation

It is not possible to prevent Kerberoast attacks explicitly, however it is possible to make it near impossible to crack the password and limit the effects of a successful compromise.

The encryption types available to Kerberos can be defined in Group Policy under:

Computer Configuration > Windows Settings > Security Settings > Local Policies > Security Options

The following policy and values should be configured:

Network security: Configure encryption types allowed for Kerberos

- AES128_HMAC_SHA1
- AES256_HMAC_SHA1
- Future encryption types

Accounts with SPNs should be set with complex, long, and randomly generated passwords. A minimum password length of 24 is advised. Additionally, SPNs should be reviewed and

removed when no longer required. The privileges associated with the service logon account should also be reviewed and extraneous privileges removed.

It is also possible to replace service account with Group Managed Service Accounts to benefit from automatic password management and simpler management.

References

- A Guide to Kerberoasting | RedTeam Talks Kerberos
- GitHub - SecureAuthCorp/impacket/GetUserSPNs.py
- Service Principal Names | Microsoft Learn
- Network security: Configure encryption types allowed for Kerberos | Microsoft Learn
- Getting Started with Group Managed Service Accounts | Microsoft Learn
- Steal or Forge Kerberos Tickets: Kerberoasting | MITRE ATT&CK

Affected

The following accounts are vulnerable to Kerberoasting:

- User_1
- User_2
- User_3

Finding 10 Domain Password Policy Analysis

Description

Risk	Low
Impact	Moderate
Likelihood	Rare
Type	Network

Active Directory (AD) is often the source of truth for user credentials within a domain. It can be configured to provide Single Sign-On (SSO) to extend a user's access to third party applications and services. The domain's password policy is crucial to ensuring strong credentials are used across the organisation.

Weak password policies can result in users using easy-to-guess or common passwords, making applications that sync with AD vulnerable to password attacks, such as bruteforcing (trying a list of passwords against a single account) and spraying (trying a single password against a list of accounts). The impact is limited to the privilege of the compromised account, but could result in sensitive information exposure or remote code execution. Open source tools, such as hydra and spray.sh, exist to perform these attacks.

Impact

Using the sample account, a member of Domain Users, the password policy for the domain was retrieved. The retrieved password policy is shown in the table below.

Field	Value
Minimum password length	14 characters
Enforce password history	4 passwords
Maximum password age	90 days
Password must meet complexity requirements	Enabled
Minimum password age	Not defined
Reset account lockout counter after	30 minutes
Account lockout threshold	3

After gaining Domain Admin privileges, the Active Directory domain database was taken from the domain controller to perform password analysis. This involved rounds of password cracking to ascertain the strength of the configured password policy.

	Total hashes	Total cracked	Total percentage cracked	Unique hashes	Unique cracked	Unique percentage cracked
Total accounts	2063	401	19.44%	1723	67	3.89%
Enabled accounts	2094	364	17.38%	1433	50	3.49%

As shown in the table above, less than 18% of active passwords and less than 4% of unique passwords used within the organisation's Active Directory were able to be cracked and have the plaintext password retrieved. Notably, only one privileged user (User_DA) was able to be

compromised. The large disparity between total cracked and unique cracked passwords is due to password reuse for POS accounts, which use one of two (2) passwords. Enforcing a minimum 14-character password is likely the main attribute for the low percentage of cracked passwords.

The following are the most commonly used cracked passwords in the organisation:

Password	Count
Password1	284
Password123	24
Letmein123	5
Welcome123	3
Ilovemykids123	2
Sampleclient2022	2

Mitigation

Reduce the reuse of domain passwords across accounts. For POS accounts, consider using randomised passwords and implementing a standard Windows PIN across devices. This will allow local access to machines without reusing passwords across the domain.

References

- Password policy recommendations for Microsoft 365 passwords | Microsoft Leanr
- NIST Special Publication 800-63B OS Credential Dumping: NTDS | MITRE ATT&CK
- Brute Force: Password Guessing | MITRE ATT&CK Brute Force: Password Cracking |
- MITRE ATT&CK Brute Force: Password Spraying | MITRE ATT&CK
- Brute Force: Credential Stuffing | MITRE ATT&CK
- GitHub - vanhauser-thc/thc-hydra GitHub - Greenwolf/Spray

Finding 11 Missing RDP Hardening

Description

Risk	Low
Impact	Moderate
Likelihood	Rare
Type	Host

Remote Desktop Protocol can be configured to use Network Layer Authentication (NLA) and TLS encryption to prevent eavesdropping and person-in-the-middle attacks. Without these protections, it is possible to interact with the remote machine without authentication in a limited manner, as well as intercept credentials.

Open source tools, such as Seth, exist to perform these attacks.

Impact

Due to a lack of NLA, it was possible to login to servers without authentication to view current and previously logged in users.

As the affected servers were not contained within the same local network as users, a person-in-the-middle attack to retrieve RDP credentials was not attempted.

Mitigation

Enable and enforce Network Layer Authentication and require encryption on all domain-joined computers through Group Policy. The relevant policies are under:

Computer Configuration > Policies > Administrative Templates > Windows Components > Remote Desktop Settings > Remote Desktop Session Host > Security

Set the following policies and values:

- Require user authentication for remote connections by using Network Level Authentication - Enable
- Require use of specific security layer for remote (RDP) connections - High/SSL (TLS 1.0)

References

- Performing RDP Man in the Middle (MitM) Attacks Using Seth.sh to Steal Passwords | Infinite Logins
- GitHub - SySS-Research/Seth: Perform a MitM attack and extract clear text credentials from RDP connections
- Configure Network Level Authentication for Remote Desktop Services Connections - TechNet Articles - United States (English) - TechNet Wiki
- Forcing RDP to use TLS Encryption | The Dispel Blog

Affected

The following RDP-enabled machines did not enforce the use of NLA:

- 10.0.0.1
- 10.0.0.2
- 10.0.0.3
- 10.0.0.4
- 10.0.0.5
- 10.0.0.6

Finding 12 Unencrypted Telnet

Description

Risk	Low
Impact	Moderate
Likelihood	Rare
Type	Host

Telnet is often used for the management of network and embedded devices. It is unencrypted, allowing an attacker with a person-in-the-middle position to view plaintext credentials or other sensitive information during an active session.

Impact

Five (5) instances of unencrypted telnet were found within the Sample Client internal network.

Mitigation

Where possible, disable telnet and enable SSH for remote management. Consider limiting access to these devices to a dedicated management subnet.

References

- Adversary-in-the-Middle | MITRE ATT&CK

Affected

The following machines were accessible via telnet:

- 10.0.0.1
- 10.0.0.2
- 10.0.0.3
- 10.0.0.4
- 10.0.0.5

Finding 13 Unsupported Software

Description

This finding has been included for informational purposes.

Software that is no longer supported by the vendor is more likely to contain unpatched security issues, as discovered vulnerabilities are unlikely to be applied to unsupported versions.

Impact

Over five (5) instances of unsupported software that was network accessible was identified within the Sample Client internal network. It is likely more instances of unsupported software exists on hosts within the environment.

Mitigation

Update the unsupported software to supported versions. At time of writing, this includes:

Software	Supported Version(s)
Microsoft Windows Server	2016, 2019, 2022
IIS	8.0, 8.5, 10
MSSQL	2014 SP3, 2016 SP3, 2017, 2019
FreeBSD	12.4, 13.1

References

- Windows Server release information | Microsoft Learn
- Internet Information Services (IIS) - Microsoft Lifecycle | Microsoft Learn
- Supported SQL Server versions - Configuration Manager | Microsoft Learn
- Release Information | The FreeBSD Project

Affected

The following machines run unsupported software:

IP address	Software
10.0.0.1	Microsoft Windows Server 2008 R2 Standard Service Pack 1
10.0.0.2	Microsoft Windows Server 2008 R2 Standard Service Pack 1
10.0.0.3	FreeBSD 10.3/11.0 Microsoft IIS 7.5 SQL Server 2014 GDR
10.0.0.4	SQL Server 2014 GDR SQL Server 2012 RTM
10.0.0.5	
10.0.0.6	
10.0.0.7	

Appendix A Risk Assessment Methodology

A qualitative risk assessment is performed on each vulnerability to determine the impact and likelihood of the vulnerability being exploited. An overall risk is calculated based on the table below:

Impact \ Likelihood	Rare	Unlikely	Possible	Likely
Critical	Medium	HIGH	CRITICAL	CRITICAL
Severe	Low	Medium	HIGH	HIGH
Moderate	Low	Medium	Medium	HIGH
Low	Low	Low	Low	Medium

The risk assessment methodology is derived from industry standards such as ISO 310001 and OWASP Risk Rating Methodology2.

The impact rating is deduced from multiple factors that consider both technical impact and business impact:

- Loss of confidentiality: How much sensitive information could be accessed or leaked and how sensitive was it?
- Loss of integrity: How much data could be corrupted and what degree of corruption was possible? Was it possible to perform actions on behalf of others?
- Loss of availability: How much services could be disrupted, preventing users from performing their tasks? What was the degree of impairment?
- Financial damage: How much money could be lost as a result?
  Reputational damage: How badly would the company's reputation be damaged and how much trust could customers lose?
- Non-compliance: Would the business be in breach of certain compliance standards they are obliged to comply with? (e.g. Privacy Act, PCI-DSS)

The likelihood is deduced from considering who the adversary may be and factors around the vulnerability:

- Skill of adversary: How skilful is the attacker likely to be?
- Motive: What are the motivating factors that the adversary may have?
- Resources: How much time and economic resources does the adversary have?
- Ease of discovery: How likely is the adversary to discover the vulnerability?
- Ease of exploitation: How easy is the vulnerability to exploit and are there publicly available tools to aid in doing so?
- Detection: How likely is the attack to be discovered by the organisation?

An overall rating (from Low to Critical) is given to each vulnerability. The vulnerabilities are then sorted in order from importance and urgency to remediate.

[1] https://www.iso.org/iso-31000-risk-management.html
[2] https://www.owasp.org/index.php/OWASP_Risk_Rating_Methodology

Appendix B Internal Network Testing Methodology

WhiteRookCyber follows industry standards including the Penetration Testing Execution Standard (PTES3) to perform internal network penetration testing.

The methods and feasibility of physical network access by an attacker is assessed. The consultant connects to the internal network and passively examines visible network traffic. The local network is mapped, and the existence of other networks is investigated.
Network vulnerability scans are conducted to search for known vulnerabilities in network infrastructure. Internal services are discovered and investigated. Default and weak credentials are used in attempts to gain access.
If Active Directory is available, known vulnerabilities are exploited to gain credentialed access to the environment. The domain is mapped, including privileged users and high value servers. Potential privilege escalation paths are investigated and exploited where possible. If the domain can be fully compromised, password hashes for all users are taken from the domain controller for offline password cracking. This is done as part of a password policy audit.
Low privilege credentials, if compromised, are used to access internal resources to determine if passwords or other sensitive data can be accessed. High privilege credentials are used to compromise systems for further lateral movement. Sensitive systems are attempted to be accessed to determine existing security controls. Open source malware may be used to determine antivirus effectiveness.

[3] http://www.pentest-standard.org/index.php/PTES_Technical_Guidelines

Appendix C External Network Testing Methodology

WhiteRookCyber follows industry standards including the Penetration Testing Execution Standard (PTES4) to perform external network penetration testing.

Open-source intelligence is used to determine assets that likely belong to the organisation. This includes DNS enumeration, IP block information, email addresses, and third-party applications.

The external attack surface is mapped using port scans. Accessible ports are probed to determine hosted services and their versions. Web applications are explored to determine if sensitive information can be gathered unauthenticated. Vulnerability scans are performed on infrastructure to determine exploitability.

Where feasible, brute forcing is performed on available services to determine if default or weak passwords are able to be used to gain further access.

[4] http://www.pentest-standard.org/index.php/PTES_Technical_Guidelines

Appendix D Web Application Testing Methodology

WhiteRookCyber follows industry standards including the Penetration Testing Execution Standard (PTES5) and the Open Web Application Security Project (OWASP6) to perform web application penetration testing.

Preauthentication activities involve mapping the attack surface of the application, such as by performing subdirectory bruteforcing, to identify functionality that may be accessible to an attacker without credentials. Unsecured webpages may include sensitive information or powerful functionality that could negatively impact users and the system itself. Available source code is analysed to find sensitive information in JavaScript or comments. Common files, such as robots.txt, are examined.

Where possible, the framework and underlying server technology are identified and known vulnerabilities are tested for. JavaScript versions are tested for vulnerabilities.

The login functionality is tested for injection flaws such as SQL injection that could allow an attacker to bypass authentication, and attacks such as password bruteforcing are performed to determine what mitigations may exist with the application, such as lockouts or CAPTCHA. Issues such as user enumeration through login error messages are identified.

Authentication is tested, noting the method of identifying the user (such as through a cookie or JWT).

The strength of passwords is tested, as well as the workflow of changing and resetting a user's password. Multifactor authentication, if available, is tested for bypasses.

File uploads are tested to determine if malware is able to be uploaded, and if so, if it can be executed on the web server. Dangerous file types are uploaded to determine if data filtering occurs. Files are attempted to be retrieved without credentials, and with another user's credentials.

The functionality of the application is tested. This includes generic fuzzing for input validation flaws, authorisation weaknesses, and logic flaws in workflows. The misuse of intended functionality is explored. The behaviour of the web application is investigated. This includes third party activity and web browser storage. The web server is investigated for HTTP flaws and encryption weaknesses.

[5] http://www.pentest-standard.org/index.php/PTES_Technical_Guidelines
[6] https://owasp.org/www-project-web-security-testing-guide/

Appendix E Mobile Application Testing Methodology

WhiteRookCyber follows industry standards including OWASP Mobile Application Security (MAS7) to perform mobile application penetration testing.

The mobile application, after installation on a mobile device, is retrieved and analysed. Where possible, the application's source code is decompiled and examined. Information such as supported versions, third party libraries, and permissions requested are collated and analysed.

Dynamic analysis begins with attempting to bypass any defences the app may implement, such as SSL pinning to prevent web traffic proxying or instrumentation detection. Once bypassed, the application is interacted with to generate artifacts on the mobile device. This can include log files, objects stored in keychains/keystores, and other files that may contain sensitive data. Backups and memory dumps are taken and examined for sensitive information.

Web traffic is examined and tested according to WhiteRookCyber's web application testing methodology. Mobile-specific functionality, such as biometrics, is tested and attempts made to bypass.

[7] https://mas.owasp.org/

Appendix F WiFi Network Testing Methodology

WhiteRookCyber follows industry standards including the Penetration Testing Execution Standard (PTES8) to perform wireless network penetration testing.

The WiFi network is passively analysed to determine its encryption and authentication protocols. It is determined if clients within range are connected to the targeted WiFi network.

Open WiFi networks are connected to directly. It is noted if additional authentication is required, and if it is able to be bypassed. It is determined if internal network access can be achieved pre- or post-authentication. In organisations using multifactor authentication, it is tested if connecting to the guest network allows single factor authentication as a “trusted location”.

Preshared key networks are attacked in order to retrieve an encrypted version of the preshared key. This could be through disassociating clients to retrieve four-way handshakes, or PMKID attacks. The preshared key is attempted to be retrieved through password cracking.

Enterprise wireless networks are tested for their authentication type. Username and password authentication is attempted. Where feasible, evil twin attacks are performed in order to retrieve credentials.

[8] http://www.pentest-standard.org/index.php/PTES_Technical_Guidelines

Appendix G Phishing Methodology

WhiteRookCyber attempts to emulate phishing campaigns that are successfully used by attackers across the globe. These phishing campaigns are mostly employed to gather credentials or deliver malware to victims.

Users of a victim organisation are gathered through open-source intelligence, such as LinkedIn. Names and emails of these users are gathered to create targeted lists of users.

Spam phishing utilises generic templates (such as imitating Microsoft security alerts) to entice many users into either submitting their credentials or downloading malware.

Spear phishing, using either a compromised account or an impersonated account, uses customised templates that are more likely to be interacted with by users of the victim organisation.

Credential phishing landing pages are made to either impersonate or proxy traffic to an authentication form likely used by the organisation. When submitted, the user is redirected to the genuine site.

Malware links can host a variety of payloads, such as tainted Office documents containing malicious macros or tainted containers (like ZIP, ISO and RAR) that execute commands and download C2 malware.

About WhiteRook Cyber

WhiteRook Cyber is an Australian Cyber Security organisation changing the approach to supporting clients in building digital resilience and cyber security capability through offerings that include Awareness, Advisory, Leadership and Training.

Our purpose is to simplify cyber security and increase security awareness and resilience, enabling organisations to focus on their core business.

Our focus is on understanding cyber security vulnerabilities and gaps within your environment, across business, people, processes, and technology. Linking the cyber risks to your business risks, while translating and clarifying the issues associated with your cyber security technical requirements for leaders and managers within your business to better understand.

We do this by assisting organisations to increase their security maturity and ongoing digital resilience, through cyber security professional service and solutions, embedded with enablement and upskilling.

For more information on our cyber security programs, services, and solutions, please contact us at:

contact@whiterookcyber.com.au