VidraSec

Penetration Test of the Internal Infrastructure

Findings Report

CLASSIFICATION: PUBLIC

Organization:	Sample LLC
Recipients:	Alice Argyle (alice@example.com) Bob Bright (bob@example.com)
Date:	May 26, 2026
Version:	1.0
Project ID:	2025-06-20
Author:	Martin Grottenthaler (martin@vidrasec.com)

Table of Contents

Executive Summary

This report describes the results of a penetration test of the internal infrastructure. The test involved searching for vulnerabilities in the internal infrastructure starting from a standard client.

During the test, at least two methods were found through which the entire Active Directory could be taken over. The most problematic vulnerability is a critical vulnerability in Active Directory Certificate Services. Through this vulnerability, a malicious person with minimal rights and network access can take over the entire Active Directory. This vulnerability is particularly problematic because it is very easy to exploit.

The second method for taking over the entire Active Directory involves a combination of multiple vulnerabilities. In particular, the errors in the implementation of the Tier Model should be addressed, as they completely undermine this otherwise very useful security concept. Additionally, there is a vulnerability where the local admin passwords of various clients are the same. This allows for the takeover of additional clients after one client has been compromised. In this case, it was possible to take over the terminal server.

Two other important vulnerabilities in the attack chain are the lack of disk encryption and the possibility of privilege escalation on the standard client. These make attacks much easier and should therefore be addressed.

The remaining vulnerabilities should also be fixed. Most of these are recommended improvements that would further enhance security.

The tested system exhibited an insufficient level of security, as it was possible to take over the entire Active Directory. However, this result is not unusual for a company being tested for the first time. It is recommended to address the identified vulnerabilities and conduct further tests in the future. Only through regular testing can the security level be sustainably increased.

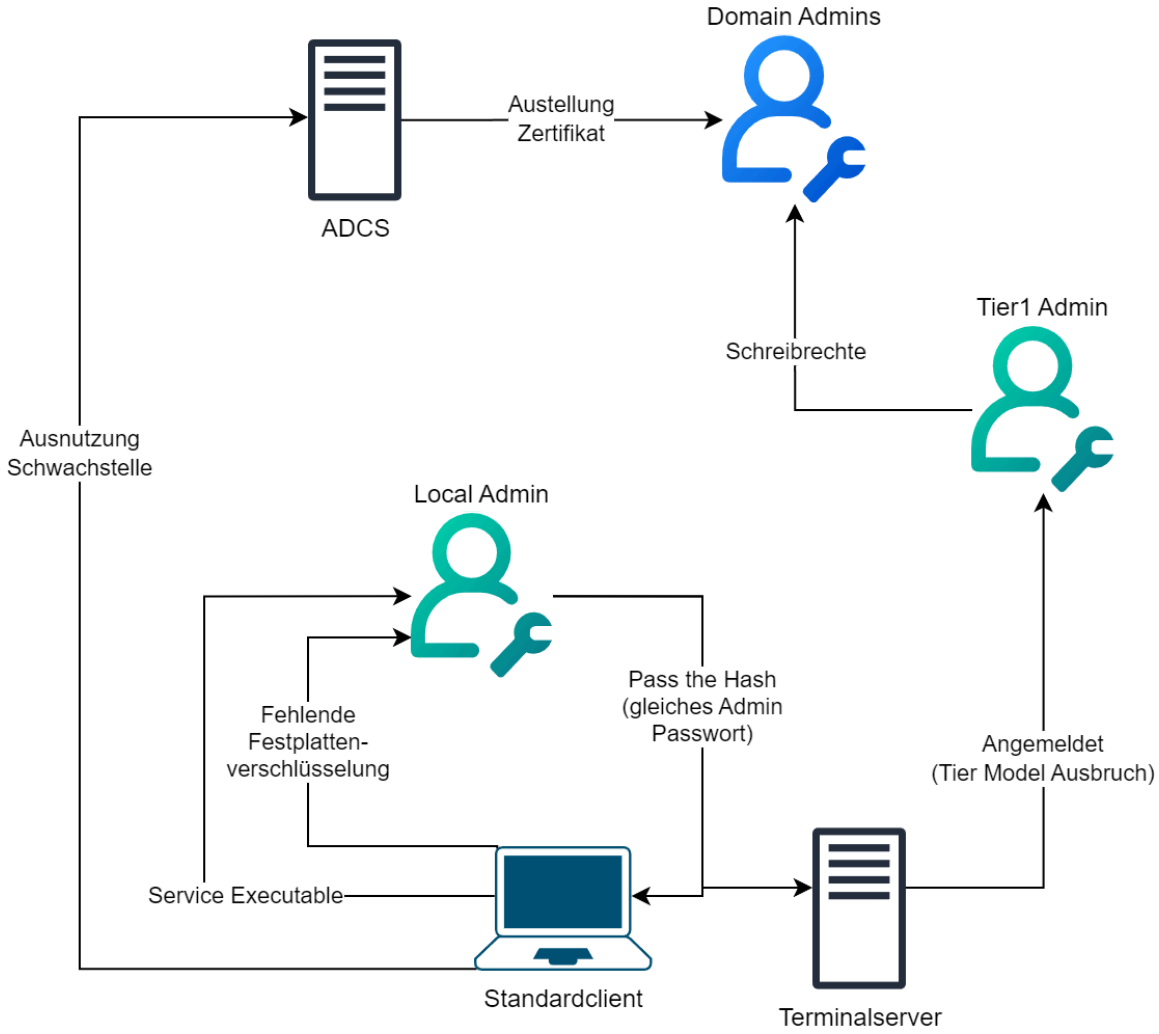
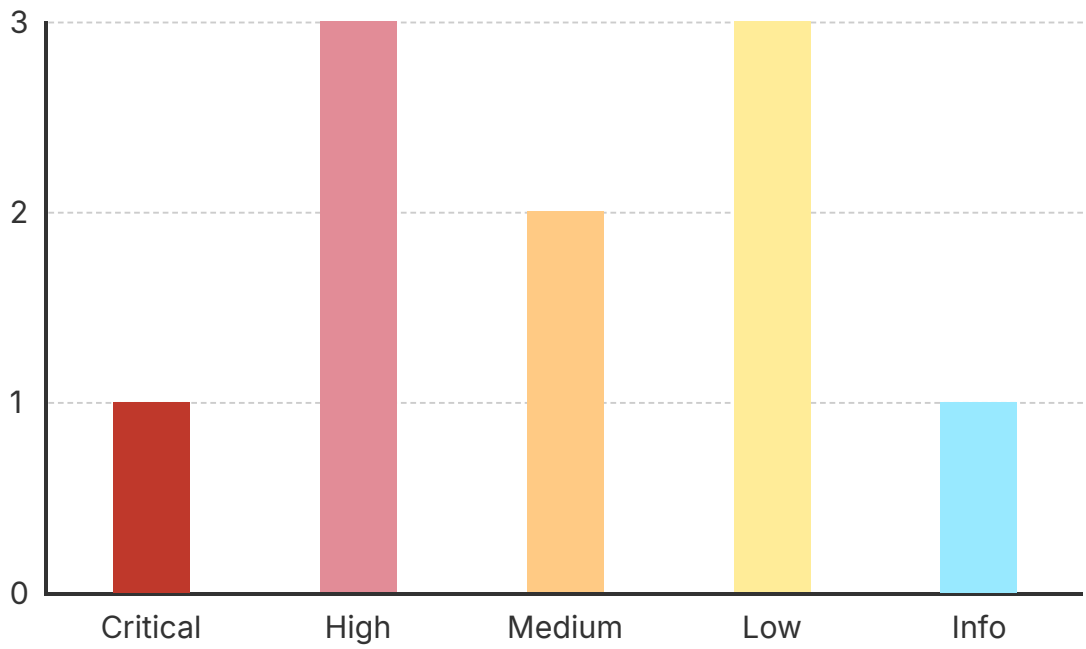


Figure 2: Visualization of attack paths

Vulnerability Overview

Vulnerability	Severity
1. Vulnerability in Active Directory Certificate Services (ADCS)	Critical
2. Privilege Escalation: Writable Service Executable	High
3. Same Local Admin Password on Different Machines	High
4. Errors in the Implementation of the Tier Model	High
5. No Disk Encryption	Medium
6. Password Quality Needs Improvement	Medium
7. Built-In Admin is in Use	Low
8. Too Many Domain Admins	Low
9. KRBTGT Password Not Changed for a Long Time	Low
10. Authenticated Users in Pre-Windows 2000 Compatible Access	Info

Vulnerability Distribution by Severity



Test Coverage

A penetration test of the internal infrastructure was conducted at Sample LLC with a **scope of 3 person-days**. The test was carried out using a time-box approach, meaning that as many vulnerabilities as possible were identified within the available time.

The penetration test was performed starting from a standard Windows client of the tested company (`Notebook1`). For authentication purposes, the tester was provided with an Active Directory account with typical user rights (`User1`). Additionally, the test was executed from a Kali Linux machine brought in by the tester, which was granted direct network access.

The goal of the penetration test was to find vulnerabilities in the following components:

- Active Directory
- Internally accessible IT infrastructure
- Windows client configuration

The following systems were targets of the penetration test:

System	Description
example.com	Active Directory
192.168.0.0/16	Internal Network

The following systems were explicitly excluded from the penetration test:

System	Description
192.168.100.0/24	OT Network

The test was conducted from October 1 to October 3, 2024, at the premises of Sample LLC.

Methodology

This penetration test was conducted using a *time-box* approach. This means that as many vulnerabilities as possible were uncovered within the available time. The process was prioritized, focusing first on vulnerabilities that are typically more problematic.

Severity Assessment

For each vulnerability, a technical severity level was determined. This does not always reflect the actual business risk posed by a vulnerability. For an accurate risk assessment, additional parameters such as the importance of the affected service should be considered alongside the technical severity level. Based on the risk, appropriate risk management measures can then be derived.

The following table provides an overview of the possible severity levels:

Severity	Meaning
Critical	The vulnerability is very problematic and should be addressed immediately. For example, it could allow for the complete takeover of a critical system.
High	The vulnerability is problematic and should be addressed as soon as possible. For example, it could allow for the takeover of a system.
Medium	This vulnerability could potentially be problematic and should be analyzed. For example, it could allow sensitive information to be read.
Low	This vulnerability should be analyzed. For example, it could be exploited in combination with other vulnerabilities or disclose internal information.
Info	This type of vulnerability is not problematic in itself. Examples would include measures that could further enhance security.

Vulnerabilities

1. Vulnerability in Active Directory Certificate Services (ADCS) Critical

Affected Systems

- ADCS.EXAMPLE.COM

Description

A severe vulnerability was found in Active Directory Certificate Services (ADCS). This vulnerability is particularly problematic because it allows a malicious person to take over the entire Active Directory. To do this, they only need access to the internal network and the login credentials of any Active Directory account. This is possible because ADCS can issue certificates that can be used for authentication. Therefore, a person who can take over ADCS can issue certificates for any accounts in the Active Directory.

ESC8

The discovered vulnerability is also known as ESC8. It is a vulnerability in the Web Enrollment functionality of ADCS. This functionality is susceptible to a *Man-in-the-Middle* attack, which grants a malicious person access to the certificates issued in this way.

During the test, a Domain Controller was tricked into connecting to a malicious system. The connection was then forwarded to ADCS, allowing the malicious system to eavesdrop on the network traffic. The Domain Controller communicates with ADCS and requests a certificate for authentication. Since the malicious system can eavesdrop, it can steal the certificate and use it for authentication itself.

The attack is also described in detail in [1].

Countermeasures

ESC8

The simplest and safest countermeasure is to completely disable Web Enrollment. This functionality is often not needed. To completely remove the functionality, the `Certification Authority (CA) Web Enrollment` role should be removed.

If Web Enrollment functionality is absolutely necessary, Microsoft provides further information on how the service can be securely configured: [2]

---

1. Will Schroeder. Certified Pre-Owned: https://posts.specterops.io/certified-pre-owned-d95910965cd2
2. Microsoft. KB5005413: Mitigating NTLM Relay Attacks on Active Directory Certificate Services (AD CS): https://support.microsoft.com/en-gb/topic/kb5005413-mitigating-ntlm-relay-attacks-on-active-directory-certificate-services-ad-cs-3612b773-4043-4aa9-b23d-b87910cd3429

2. Privilege Escalation: Writable Service Executable

High

Affected Systems

- CLIENT1.EXAMPLE.COM
- InventoryService

Description

On the Windows 11 standard client, it was possible to escalate privileges to local admin because a service installed there was misconfigured. The service executable is writable with normal user rights.

The vulnerable service is `InventoryService`. The service itself starts automatically and runs with `LocalSystem` privileges. The executable is located at `C:\Inventory\InventoryService.exe` and is writable with the rights of a normal, restricted user.

Therefore, a malicious person can simply replace the `InventoryService.exe` with any other executable. When the service starts, this executable is also run with `LocalSystem` privileges. In this case, an executable was placed that adds a new local admin. Although the user cannot restart the service themselves, a reboot of the machine results in a service restart.

This vulnerability led to the ability to escalate privileges to local admin on the standard client during the test.

Countermeasures

The installation of the service must be adjusted. The service executable should only be writable by users with local admin rights. Alternatively, the service itself can have its rights revoked. If the service were to run with normal user rights, privilege escalation would not be possible here.

3. Same Local Admin Password on Different Machines

High

Affected Systems

- CLIENT1.EXAMPLE.COM
- CLIENT2.EXAMPLE.COM
- CLIENT3.EXAMPLE.COM
- TS1.EXAMPLE.COM

Description

It was determined that different systems had the same local admin password set. During the test, this led to the ability to take over other systems on the network starting from one machine.

A characteristic of Windows is that for network authentication, a password hash is usually sufficient, and the plaintext password is not required (“Pass the Hash”). Password hashes can be read from memory (LSASS) or from the hard drive (SAM) with local admin rights.

Therefore, it is particularly important that different systems have different local admin passwords. The cause of identical admin passwords is often that multiple systems were created from the same image, and the admin passwords were not changed afterward.

Countermeasures

Different systems must not have the same local admin password. While the passwords can theoretically all be changed manually, it is better to use an automated solution. Microsoft offers free software for this purpose called “Local Administrative Password Solution (LAPS)”[1].

1. Microsoft. What is Windows LAPS?: https://learn.microsoft.com/en-us/windows-server/identity/laps/laps-overview

4. Errors in the Implementation of the Tier Model

High

Affected Systems

- TS1.EXAMPLE.COM
- ADMINS-T1@EXAMPLE.COM

Description

Multiple errors were found in the implementation of the Tier Model. These errors allow for privilege escalation between the tiers.

Tier 2 → Tier 1

Terminal Server in the Wrong Tier

The terminal server `TS1.EXAMPLE.COM` is currently in **Tier 1**, meaning it is administered by Tier 1 admins. Since normal users log into this system, this server should be placed in **Tier 2**. If a malicious user can achieve privilege escalation on the terminal server, they can compromise logged-in admin accounts. Successful privilege escalation is only a matter of time, as new vulnerabilities are continually being published, and a malicious person only needs to wait.

Systems where normal users log in should always be in **Tier 2**.

Tier 1 → Tier 0

Tier 1 Admins Have Write Permissions on Tier 0 Group

Members of the group `ADMINS-T1@EXAMPLE.COM` have write permissions on the group `EXCHANGE TRUSTED SUBSYSTEM@EXAMPLE.COM`. This group, in turn, has write permissions on the group `DOMAIN ADMINS@EXAMPLE.COM` (Tier 0).

This allows a malicious person who can take over an account in Tier 1 to break into Tier 0 and take over the entire Active Directory.

Countermeasures

Terminal Server in the Wrong Tier

The terminal server should be moved to Tier 2. This way, it will be administered by Tier 2 admins, and a takeover will have lesser impacts than a takeover of a Tier 1 admin.

Additionally, it should be considered whether it makes sense to manage the local admin password of this server with a solution like LAPS[1]. This would allow Tier 2 admins to read the (temporary) local admin password of this server and use the local account to log in. This would prevent the takeover of the Tier 2 account but make the traceability of actions more difficult (only generic admin account in logs).

Tier 1 Admins Have Write Permissions on Tier 0 Group

The rights to the Tier 0 group should be revoked from the Tier 1 group. Additionally, split permissions for Exchange should be implemented. This will revoke Tier 0 rights from Exchange server admins.[2]

---

1. Microsoft. What is Windows LAPS?: https://learn.microsoft.com/en-us/windows-server/identity/laps/laps-overview
2. Microsoft. Configure Exchange Server for split permissions: https://learn.microsoft.com/en-us/exchange/permissions/split-permissions/configure-exchange-for-split-permissions

5. No Disk Encryption

Medium

Affected Systems

- Windows Clients

Description

The affected systems do not use disk encryption.

In the event of device theft, malicious individuals could easily access sensitive data, such as Active Directory login credentials.

An often-overlooked aspect is that without disk encryption, local privilege escalation is possible. A malicious person could manipulate the disk content to create an additional local admin account.

Therefore, even devices operated in secure rooms should always be equipped with active disk encryption.

Countermeasures

It is recommended to implement a disk encryption solution.

Microsoft offers *BitLocker*[1] as a free option for this purpose.

When implementing, choose an encryption solution that utilizes the device's built-in *Trusted Platform Module (TPM)*, as this increases security.

Even though some configurations can do without an additional "pre-boot password," it is recommended to use one, as the *TPM-only mode* is vulnerable to various attacks.[2]

---

1. Microsoft. BitLocker overview: https://learn.microsoft.com/en-us/windows/security/operating-system-security/data-protection/bitlocker/
2. VidraSec. Securing BitLocker: Initial Setup and Protection Against Attacks: https://www.vidrasec.com/de/blog/setup-bitlocker/

6. Password Quality Needs Improvement Medium

Affected Systems

- Active Directory

Description

Poor password quality can lead to account takeovers. Especially for higher-privileged accounts, random and unique passwords should be assigned. However, all individuals in the company should be trained to use long and secure passwords and not to reuse them.

Accounts with the Same Password

The following accounts had the same password. This could be determined because passwords in Active Directory are stored without salt. Therefore, a certain character sequence always results in the same hash.

The following accounts have the same password:

- Service1 and Service2
- AliceT0, AliceT1, AliceT2, and Alice
- Kiosk1, Kiosk2, Kiosk3, and Kiosk4

It is particularly problematic when an account has the same password in different tiers. A person who discovers the password of the account could use it to break into higher tiers.

Old Passwords

Since it can happen for various reasons that a password becomes known, it is advisable to change the password regularly (e.g., once a year), especially for higher-privileged accounts and service accounts.

The following accounts have an old password:

Account	Changed
Service1	2010-08-01
Service2	2015-01-15
AliceT0	2013-04-13

Countermeasures

The results described above should be analyzed, and passwords should be changed if necessary.

In general, the passwords of service accounts and highly privileged accounts should be changed regularly. For service accounts, this has the side effect that the use of passwords must be precisely documented. Furthermore, passwords should generally be unique. Especially highly privileged accounts should use random passwords, which should make accidentally identical passwords impossible.

7. Built-In Admin is in Use

Low

Affected Systems

- Active Directory

Description

It was determined that the Built-In Admin of the domain (RID 500) had successfully logged in recently. Therefore, it is assumed that this user is being used for administrative tasks. This user should only be used for the initial setup of Active Directory and for emergencies, and should be specially secured.

In general, only personalized admin accounts should be used to ensure traceability of actions performed. The Built-In Admin is particularly problematic because it has special properties. For example, this user cannot be locked out by a brute-force attack. While this behavior makes this user particularly vulnerable to brute-force attacks, it also makes it the perfect user for emergencies (e.g., all other domain accounts have been locked out by brute-force attacks).

Countermeasures

The Built-In Admin should not be used during normal operations. Its password should be set as long and random as possible and stored securely, for example, in a safe. Only personalized domain admin accounts should be used for administration.

Additionally, the Built-In Admin should be specially secured. More information can be found in the Microsoft documentation. [1]

1. Microsoft. Appendix D: Securing Built-in Administrator Accounts in Active Directory: https://learn.microsoft.com/en-us/windows-server/identity/ad-ds/plan/security-best-practices/appendix-d--securing-built-in-administrator-accounts-in-active-directory

8. Too Many Domain Admins

Low

Affected Systems

- Active Directory

Description

The tested Active Directory has an unusually high number of accounts with Domain Admin rights. This is dangerous because the takeover of just one of these accounts can lead to the complete takeover of the Active Directory.

The following accounts with Domain Admin rights were identified:

- AliceT0
- GustavT0
- HedwigT0
- CarolT0
- FileAdmin

According to the information provided, only `AliceT0` and `GustavT0` are regularly used. The other accounts should be removed from the Domain Admins group.

Countermeasures

All accounts should only have the rights that they truly need. This is especially important for the most highly privileged accounts.

In the Domain Admins group, apart from the built-in Admin, there should only be personalized accounts. It should be regularly checked whether the individuals who have Domain Admin accounts still need them.

Service accounts should normally not have Domain Admin rights.

9. KRBTGT Password Not Changed for a Long Time Low

Affected Systems

- Active Directory

Description

The Active Directory account `krbtgt` is essential for the Kerberos authentication protocol. All Kerberos tickets are signed with the password hash of this account. If a malicious person knows this hash, they can create arbitrary valid Kerberos tickets and, for example, impersonate a Domain Admin. Such misuse is known as a *Golden Ticket Attack*[1].

In this case, the password of the `krbtgt` account was last changed on `March 1, 2010`.

Countermeasures

The `krbtgt` account has a password history of `2`. This means that Kerberos tickets signed with the old password will continue to be accepted. Therefore, a simple password rotation is not sufficient to prevent misuse. It is necessary to rotate the password **twice**, waiting for the synchronization of the domain controllers between changes. Otherwise, synchronization issues may occur.

Any person with Domain Admin rights has access to the hash of the `krbtgt` account. If such a person leaves the company, the password of the `krbtgt` account must also be changed **twice** for security reasons.

In the event of a suspected attack on Active Directory, it is also essential to rotate the password of the `krbtgt` account **twice** after closing the entry vulnerability. This prevents attackers from continuing to issue valid Kerberos tickets.

> It is also recommended to change the password of the `krbtgt` account regularly (e.g., monthly). Regular rotation ensures that a potentially compromised hash will no longer be valid after at most two changes.
>
> A guide to manually changing the `krbtgt` password can be found in the Microsoft documentation[2].

---

1. Semperis. How to Defend Against Golden Ticket Attacks: AD Security 101: https://www.semperis.com/blog/how-to-defend-against-golden-ticket-attacks/
2. Microsoft. Active Directory Forest Recovery - Reset the krbtgt password: https://learn.microsoft.com/en-us/windows-server/identity/ad-ds/manage/forest-recovery-guide/ad-forest-recovery-reset-the-krbtgt-password

10. Authenticated Users in Pre-Windows 2000 Compatible Access

Info

Affected Systems

- Active Directory

Description

Active Directory, as we know it, was introduced with Windows 2000. Before that, there were similar functionalities, but they were much more limited. One difference was that it was not a hierarchical structure but flat. This meant that everyone could read all attributes of every object. Active Directory limits this. A normal user can no longer read sensitive attributes of other users, such as the `pwdLastSet` attribute. The problem was and is that some applications require access to this information. Therefore, Microsoft introduced the `Pre-Windows 2000 Compatible Access` group, which again grants read permissions for all attributes of all objects. And to be sure, `Authenticated Users` were added to this group.

If `Authenticated Users` are not in this group, it can be very annoying for attackers. As mentioned, one of the interesting attributes that an attacker can no longer read is `pwdLastSet`. This means it becomes much more difficult for attackers to find accounts with old and potentially weak passwords.

However, there is an important point! Since this is a standard vulnerability in AD, some tools rely on this “functionality.” Therefore, any changes must be thoroughly tested beforehand!

Countermeasures

`Authenticated Users` should be removed from the `Pre-Windows 2000 Compatible Access` group. However, it should be thoroughly tested whether this change has any negative impact on any software. Further information can be found in [1].

1. VidraSec. Built-in Misconfigurations - Pre-Windows 2000 Compatible Access: https://www.vidrasec.com/de/blog/built-in-insecurities-win2k/

Imprint

VidraSec e.U. – https://www.vidrasec.com/
Address: Almesbergerweg 3, 4160 Aigen-Schlägl, AUSTRIA
Commercial Register Number: 623204b
Commercial Court: Regional Court Linz
Trade Supervisory Authority: BH Rohrbach
VAT ID: ATU80334229

Contact details
Email: martin@vidrasec.com
Phone: +43 670 3081275

Bank details
Bank: REVOLUT BANK UAB
IBAN: LT76 3250 0126 5399 4118
BIC: REVOLT21