VidraSec

Penetrationstest der internen IT-Infrastruktur

Ergebnisbericht

KLASSIFIZIERUNG: PUBLIC

Organisation:	Muster GmbH
An:	Alice Argyle (alice@example.com) Bob Bright (bob@example.com)
Datum:	26. Mai 2026
Version:	1.0
Projekt ID:	2025-06-20
Autor:	Martin Grottenthaler (martin@vidrasec.com)

Inhaltsverzeichnis

- Executive Summary
  - Schwachstellenübersicht
  - Schwachstellenverteilung nach Schweregrad
- Testabdeckung
- Methodologie
  - Bewertung des Schweregrads
- Schwachstellen
  - 1. Schwachstelle in Active Directory Certificate Services (ADCS)
  - 2. Privilege Escalation: Beschreibbare Service Executable
  - 3. Gleiches lokales Admin Passwort auf verschiedenen Maschinen
  - 4. Fehler in der Implementierung vom Tier Modell
  - 5. Keine Festplattenverschlüsselung
  - 6. Verbesserungswürdige Passwortqualität
  - 7. Built-In Admin wird benutzt
  - 8. Zu viele Domain Admins
  - 9. KRBTGT Passwort lange nicht geändert
  - 10. Authenticated Users in Pre-Windows 2000 Compatible Access
- Impressum

Executive Summary

Dieser Bericht beschreibt die Ergebnisse eines Penetrationstests der internen Infrastruktur. Dabei wurden ausgehend von einem Standardclient nach Schwachstellen in der internen Infrastruktur gesucht.

Im Zuge des Tests wurden mindestens zwei Wege gefunden, über die das komplette Active Directory übernommen werden kann. Die problematischste Schwachstelle ist eine kritische Schwachstelle in Active Directory Certificate Services. Über diese Schwachstelle kann eine bösartige Person mit minimalen Rechten und Netzwerkzugang das komplette Active Directory übernehmen. Diese Schwachstelle ist besonders problematisch, weil die Ausnutzung sehr einfach ist.

Der zweite Weg zur Übernahme des kompletten Active Directory ist eine Verkettung von mehreren Schwachstellen. Hier sollten insbesondere die Fehler in der Implementierung des Tier Modells behoben werden, da diese dieses eigentlich sehr sinnvolle Sicherheitskonzept komplett aushebeln. Des Weiteren gibt es eine Schwachstelle, dass die lokalen Adminpasswörter von verschiedenen Clients dieselben sind. Das ermöglicht nach Übernahme eines Clients die Übernahme von weiteren Clients. In diesem Fall war es darüber möglich, den Terminalserver zu übernehmen.

Zwei weitere wichtige Schwachstellen in der Angriffskette sind fehlende Festplattenverschlüsselung und die Möglichkeit der Privilegienausweitung auf dem Standardclient. Diese machen Angriffe um einiges einfacher und sollten daher behoben werden.

Auch die restlichen Schwachstellen sollten behoben werden. Bei denen handelt es sich großteils um empfohlene Verbesserungsmaßnahmen, die die Sicherheit weiter steigern würden.

Das getestete System wies ein unzureichendes Sicherheitsniveau auf, da es möglich war, das komplette Active Directory zu übernehmen. Grundsätzlich ist dieses Ergebnis aber nicht unüblich für ein Unternehmen, das zum ersten Mal getestet wird. Es wird empfohlen, die gefundenen Schwachstellen zu beheben

und in Zukunft einen weiteren Test durchzuführen. Nur durch regelmäßiges Testen kann das Sicherheitsniveau nachhaltig gesteigert werden.

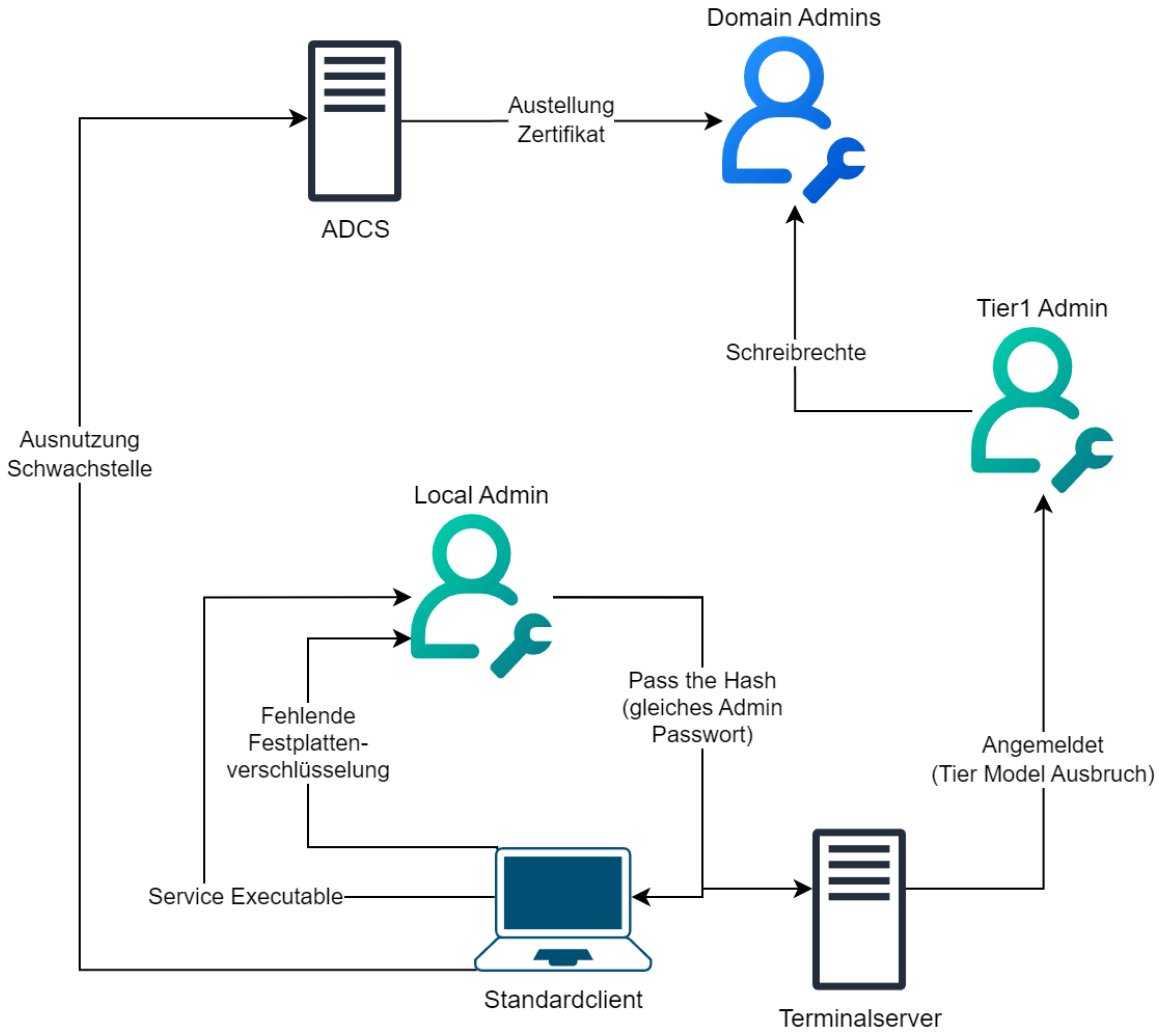
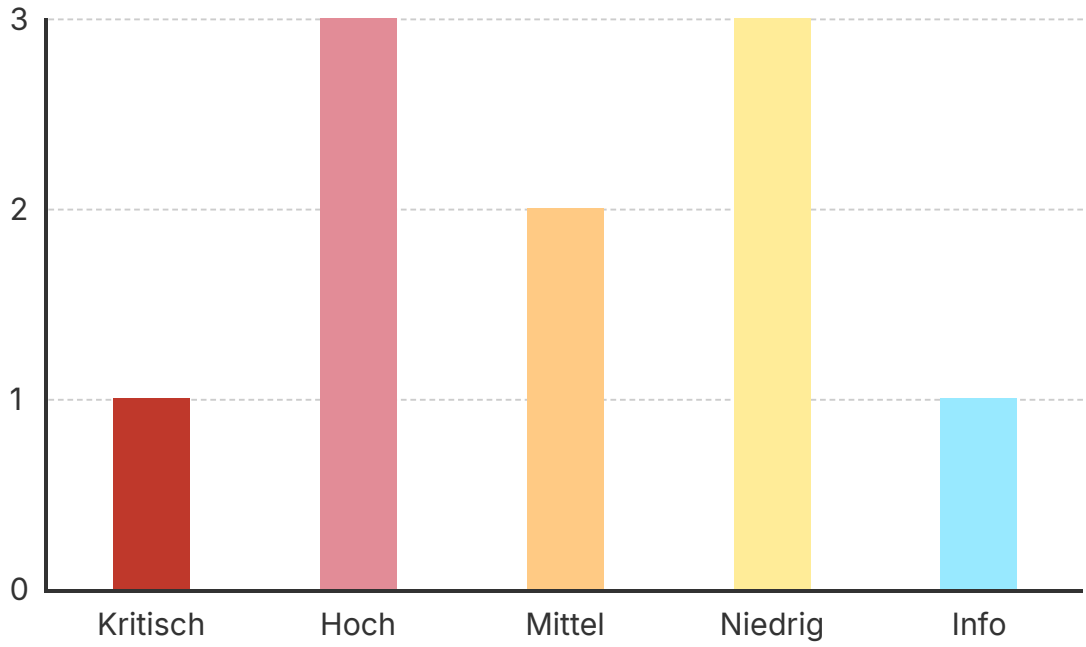


Abbildung 2: Visualisierung der Angriffswege

Schwachstellenübersicht

Schwachstelle	Schweregrad
1. Schwachstelle in Active Directory Certificate Services (ADCS)	Kritisch
2. Privilege Escalation: Beschreibbare Service Executable	Hoch
3. Gleiches lokales Admin Passwort auf verschiedenen Maschinen	Hoch
4. Fehler in der Implementierung vom Tier Modell	Hoch
5. Keine Festplattenverschlüsselung	Mittel
6. Verbesserungswürdige Passwortqualität	Mittel
7. Built-In Admin wird benutzt	Niedrig
8. Zu viele Domain Admins	Niedrig
9. KRBTGT Passwort lange nicht geändert	Niedrig
10. Authenticated Users in Pre-Windows 2000 Compatible Access	Info

Schwachstellenverteilung nach Schwere-grad



Testabdeckung

Es wurde ein Penetrationstest der internen Infrastruktur bei der Muster GmbH im **Umfang von 3 Personentagen** durchgeführt. Der Test wurde im Time-Box-Verfahren durchgeführt. Das heißt, dass in der zur Verfügung stehenden Zeit so viele Schwachstellen wie möglich gefunden wurden.

Der Penetrationstest wurde ausgehend von einem Standard-Windows-Client des getesteten Unternehmens durchgeführt (`Notebook1`). Zur Authentifizierung wurde dem Tester ein Active Directory-Konto mit typischen User-Rechten zur Verfügung gestellt (`User1`). Zusätzlich wurde der Test von einer vom Tester mitgebrachten Kali-Linux-Maschine ausgeführt, der direkter Netzwerkzugriff erlaubt wurde.

Das Ziel des Penetrationstests war das Finden von Schwachstellen in folgenden Komponenten:

- Active Directory
- Intern erreichbare IT-Infrastruktur
- Windows-Client-Konfiguration

Folgende Systeme waren Ziel des Penetrationstests:

System	Beschreibung
example.com	Active Directory
192.168.0.0/16	Internes Netzwerk

Folgende Systeme wurden explizit vom Penetrationstest ausgenommen:

System	Beschreibung
192.168.100.0/24	OT-Netzwerk

Der Test wurde im Zeitraum vom 01. bis 03. Oktober 2024 in den Räumlichkeiten der Muster GmbH durchgeführt.

Methodologie

Dieser Penetrationstest wurde mit einem *time-box*-Verfahren durchgeführt. Das bedeutet, dass in der zur Verfügung stehenden Zeit so viele Schwachstellen wie möglich aufgedeckt wurden. Dabei wurde priorisiert vorgegangen und zuerst auf Schwachstellen getestet, die typischerweise problematischer sind.

Bewertung des Schweregrads

Für jede Schwachstelle wurde ein technischer Schweregrad ermittelt. Dieser spiegelt nicht immer das tatsächliche Unternehmensrisiko wider, das eine Schwachstelle darstellt. Für die tatsächliche Risikobewertung sollten zusätzlich zum technischen Schweregrad weitere Parameter wie die Wichtigkeit des betroffenen Services berücksichtigt werden. Auf Basis des Risikos können anschließend geeignete Risikobehandlungsmaßnahmen abgeleitet werden.

Die folgende Tabelle gibt eine Übersicht über die möglichen Schweregrade:

Schweregrad	**Bedeutung**
Kritisch	Die Schwachstelle ist sehr problematisch und sollte umgehend behoben werden. Beispielsweise ist darüber die vollständige Übernahme eines kritischen Systems möglich.
Hoch	Die Schwachstelle ist problematisch und sollte schnellstmöglich behoben werden. Beispielsweise ist darüber die Übernahme eines Systems möglich.
Mittel	Diese Schwachstelle kann unter Umständen problematisch sein und sollte analysiert werden. Beispielsweise können darüber sensible Informationen ausgelesen werden.
Niedrig	Diese Schwachstelle sollte analysiert werden. Beispielsweise kann sie in Kombination mit anderen Schwachstellen ausgenutzt werden oder interne Informationen preisgeben.

Schwe-regrad	Bedeutung
Info	Diese Art von Schwachstelle ist an sich nicht problematisch. Beispiele wären Maßnahmen, die die Sicherheit noch weiter steigern könnten.

Schwachstellen

1. Schwachstelle in Active Directory Certificate Services (ADCS) Kritisch

Betroffene Systeme

- ADCS.EXAMPLE.COM

Beschreibung

Es konnte eine schwerwiegende Schwachstelle in Active Directory Certificate Services (ADCS) gefunden werden. Diese Schwachstelle ist besonders problematisch, weil es dadurch einer bösartigen Person möglich ist, das komplette Active Directory zu übernehmen. Dafür benötigt sie nur Zugriff auf das interne Netzwerk und die Anmeldedaten eines beliebigen Active Directory-Kontos. Das ist möglich, weil ADCS Zertifikate ausstellen kann, die zur Authentifizierung verwendet werden können. Eine Person, die also ADCS übernehmen kann, kann sich Zertifikate für beliebige Konten im Active Directory ausstellen.

ESC8

Die gefundene Schwachstelle wird auch als ESC8 bezeichnet. Dabei handelt es sich um eine Schwachstelle in der Web Enrollment-Funktionalität des ADCS. Diese Funktionalität ist auf einen *Machine-in-the-Middle*-Angriff anfällig. Das gewährt einer bösartigen Person Zugriff auf die so ausgestellten Zertifikate.

Im Zuge des Tests wurde ein Domain Controller dazu gebracht, sich zu einem bösartigen System zu verbinden. Die Verbindung wurde dann zu ADCS weitergeleitet, was dazu führt, dass das bösartige System den Netzwerkverkehr mitlesen kann. Der Domain Controller kommuniziert nun mit dem ADCS und

fragt ein Zertifikat zur Authentifizierung an. Da das bösartige System mitlesen kann, kann es das Zertifikat stehlen und selbst zur Authentifizierung benutzen.

Der Angriff wird auch in [1] im Detail beschrieben.

Gegenmaßnahmen

ESC8

Die einfachste und sicherste Gegenmaßnahme ist das vollständige Deaktivieren der Web Enrollment. Diese Funktionalität wird in vielen Fällen nicht benötigt. Um die Funktionalität komplett zu entfernen, sollte die `Certification Authority (CA) Web Enrollment`-Rolle entfernt werden.

Sollte die Web Enrollment-Funktionalität unbedingt benötigt werden, bietet Microsoft weiterführende Informationen, wie der Service sicher konfiguriert werden kann: [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: Beschreibbare Service Executable

Hoch

Betroffene Systeme

- CLIENT1.EXAMPLE.COM
- InventoryService

Beschreibung

Auf dem Windows 11 Standardclient war es möglich, die Rechte zum lokalen Admin auszuweiten, weil ein dort installierter Service fehlkonfiguriert ist. Die Service Executable ist mit normalen Userrechten schreibbar.

Beim anfälligen Service handelt es sich um `InventoryService`. Der Service selbst startet automatisch und läuft mit `LocalSystem`-Rechten. Die Executable liegt unter `C:\Inventory\IventoryService.exe` und ist mit den Rechten eines normalen, eingeschränkten Users schreibbar.

Eine bösartige Person kann also einfach die `IventoryService.exe` gegen eine beliebige andere Executable austauschen. Wird der Service gestartet, wird auch diese Executable mit `LocalSystem`-Rechten ausgeführt. In diesem Fall wurde eine Executable abgelegt, die einen neuen lokalen Admin hinzufügt. Der Service darf zwar vom User selbst nicht neu gestartet werden, ein Reboot der Maschine hat allerdings einen Service-Neustart zur Folge.

Diese Schwachstelle hat dazu geführt, dass im Zuge des Tests die Rechte auf dem Standardclient zum lokalen Admin ausgeweitet werden konnten.

Gegenmaßnahmen

Die Installation des Services muss angepasst werden. Die Service Executable darf nur von Usern mit lokalen Adminrechten schreibbar sein. Alternativ können dem Service selbst die Rechte entzogen werden. Würde der Service mit normalen Userrechten laufen, wäre hier keine Privilege Escalation möglich.

3. Gleiches lokales Admin Passwort auf verschiedenen Maschinen

Hoch

Betroffene Systeme

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

Beschreibung

Es konnte festgestellt werden, dass unterschiedliche Systeme dasselbe lokale Admin-Passwort gesetzt haben. Das führte im Zuge des Tests dazu, dass ausgehend von einer Maschine andere Systeme im Netzwerk übernommen werden konnten.

Eine Eigenheit von Windows ist, dass zur Authentifizierung über das Netzwerk meist ein Passworthash ausreicht und das Passwort im Klartext nicht benötigt wird („Pass the Hash“). Passworthashes können mit lokalen Admin-Rechten aus dem Speicher (LSASS) bzw. von der Festplatte (SAM) ausgelesen werden.

Daher ist es besonders wichtig, dass verschiedene Systeme verschiedene lokale Admin-Passwörter haben. Die Ursache für dieselben Admin-Passwörter ist oft, dass mehrere Systeme von demselben Image erstellt wurden und die Admin-Passwörter danach nicht geändert wurden.

Gegenmaßnahmen

Verschiedene Systeme dürfen nicht dasselbe lokale Admin-Passwort haben. Die Passwörter können theoretisch alle manuell geändert werden, besser ist aber, dafür eine automatisierte Lösung zu benutzen. Microsoft bietet hierfür die kostenlose Software „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. Fehler in der Implementierung vom Tier Modell

Hoch

Betroffene Systeme

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

Beschreibung

Es konnten mehrere Fehler bei der Implementierung des Tier Modells gefunden werden. Diese Fehler führen dazu, dass eine Privilege Escalation zwischen den Tiers möglich ist.

Tier 2 → Tier 1

Terminalserver im falschen Tier

Der Terminalserver `TS1.EXAMPLE.COM` ist aktuell im **Tier 1**. Das heißt, dass er von Tier 1 Admins administriert wird. Da sich auf diesem System normale User einloggen, sollte dieser Server im **Tier 2** platziert werden. Ist es einem bösartigen User möglich, eine Privilege Escalation auf dem Terminalserver zu erreichen, kann der User eingeloggte Adminkonten kompromittieren. Eine erfolgreiche Privilege Escalation ist nur eine Frage der Zeit, da laufend neue Schwachstellen veröffentlicht werden und eine bösartige Person daher nur warten muss.

Systeme, auf denen sich normale User einloggen, sollten immer im **Tier 2** sein.

Tier 1 → Tier 0

Tier 1 Admins haben Schreibrechte auf Tier 0 Gruppe

Mitglieder der Gruppe `ADMINS-T1@EXAMPLE.COM` haben Schreibrechte auf die Gruppe `EXCHANGE TRUSTED SUBSYSTEM@EXAMPLE.COM`. Diese Gruppe wiederum hat Schreibrechte auf die Gruppe `DOMAIN ADMINS@EXAMPLE.COM` (Tier 0).

Dadurch kann eine bösartige Person, die ein Konto im Tier 1 übernehmen kann, ins Tier 0 ausbrechen und das komplette Active Directory übernehmen.

Gegenmaßnahmen

Terminalserver im falschen Tier

Der Terminalserver sollte ins Tier 2 verschoben werden. Dadurch wird er von Tier 2 Admins administriert und eine Übernahme hat geringere Auswirkungen als eine Übernahme eines Tier 1 Admins.

Zusätzlich sollte überlegt werden, ob es sinnvoll ist, das lokale Adminpasswort dieses Servers mit einer Lösung wie LAPS[1] zu verwalten. Dadurch wäre es möglich, dass Tier 2 Admins das (temporäre) lokale Adminpasswort dieses Servers auslesen und den lokalen Account verwenden, um sich einzuloggen. Das würde eine Übernahme des Tier 2 Accounts verhindern, allerdings die Nachvollziehbarkeit von Aktionen erschweren (nur generischer Adminaccount in Logs).

Tier 1 Admins haben Schreibrechte auf Tier 0 Gruppe

Die Rechte auf die Tier 0 Gruppe sollten der Tier 1 Gruppe entzogen werden. Zusätzlich sollten Split Permissions für Exchange implementiert werden. Dadurch werden den Exchange Server Admins Rechte auf das Tier 0 entzogen. [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. Keine Festplattenverschlüsselung Mittel

Betroffene Systeme

- Windows Clients

Beschreibung

Auf den betroffenen Systemen wird keine Festplattenverschlüsselung verwendet. Im Falle eines Gerätediebstahls könnten bösartige Personen ohne großen Aufwand sensible Daten, wie etwa Active Directory-Anmeldedaten, auslesen.

Ein oft übersehener Aspekt ist, dass ohne Festplattenverschlüsselung, eine lokale Privilege-Escalation möglich ist. Eine bösartige Person könnte den Festplatteninhalt manipulieren und so ein weiteres lokales Adminkonto erstellen.

Daher sollten selbst Geräte, die in gesicherten Räumen betrieben werden, immer mit aktiver Festplattenverschlüsselung ausgestattet sein.

Gegenmaßnahmen

Es wird empfohlen, eine Lösung zur Festplattenverschlüsselung einzusetzen. Microsoft bietet hierfür beispielsweise *BitLocker*[1] als kostenlose Option an.

Bei der Implementierung sollte eine Verschlüsselungslösung gewählt werden, die das im Gerät verbaute *Trusted Platform Module (TPM)* nutzt, da dies die Sicherheit erhöht. Auch wenn einige Konfigurationen ohne ein zusätzliches „Pre-Boot-Passwort“ auskommen, wird empfohlen, dieses dennoch zu verwenden, da der reine *TPM-only-Modus* durch verschiedene Angriffe angreifbar ist.[2]

---

1. Microsoft. BitLocker overview: https://learn.microsoft.com/en-us/windows/security/operating-system-security/data-protection/bitlocker/

2. VidraSec. BitLocker absichern: Initiales Setup und Absicherung gegen Angriffe: https://www.vidrasec.com/de/blog/setup-bitlocker/

6. Verbesserungswürdige Passwortqualität

Mittel

Betroffene Systeme

- Active Directory

Beschreibung

Eine schlechte Passwortqualität kann zur Übernahme von Konten führen. Speziell bei höher privilegierten Konten sollten zufällige und eindeutige Passwörter vergeben werden. Allerdings sollten alle Personen im Unternehmen darauf geschult werden, lange und sichere Passwörter zu vergeben und diese nicht mehrmals zu verwenden.

Accounts mit demselben Passwort

Folgende Konten hatten dasselbe Passwort. Das konnte festgestellt werden, da Passwörter im Active Directory ohne Salt abgespeichert werden. Daher ergibt eine gewisse Zeichenfolge immer den gleichen Hash.

Jeweils dasselbe Passwort haben:

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

Besonders problematisch ist, dass ein Account in verschiedenen Tiers dasselbe Passwort hat. Eine Person, die das Passwort des Kontos herausfindet, könnte darüber in höhere Tiers ausbrechen.

Alte Passwörter

Da es aus verschiedensten Gründen passieren kann, dass ein Passwort bekannt wird, ist es speziell für höher privilegierte Konten und Service-Accounts empfehlenswert, das Passwort regelmäßig (z.B. einmal pro Jahr) zu ändern.

Folgende Konten haben ein altes Passwort:

Konto	Geändert
Service1	2010-08-01
Service2	2015-01-15
AliceT0	2013-04-13

Gegenmaßnahmen

Die oben beschriebenen Ergebnisse sollten analysiert werden, und die Passwörter gegebenenfalls geändert werden.

Grundsätzlich sollten die Passwörter von Service-Accounts und hoch privilegierten Konten regelmäßig geändert werden. Bei Service-Accounts hat das den Nebeneffekt, dass genau dokumentiert werden muss, wo die Passwörter benutzt werden. Des Weiteren sollten Passwörter im Normalfall eindeutig sein. Speziell hoch privilegierte Konten sollten zufällige Passwörter benutzen, was versehentlich gleichgesetzte Passwörter unmöglich machen sollte.

7. Built-In Admin wird benutzt Niedrig

Betroffene Systeme

- Active Directory

Beschreibung

Es konnte festgestellt werden, dass sich der Built-In Admin der Domain (RID 500) erst vor Kurzem erfolgreich eingeloggt hat. Daher wird davon ausgegangen, dass dieser User für administrative Tätigkeiten verwendet wird. Dieser User sollte nur für das initiale Aufsetzen des Active Directory und für Notfälle verwendet werden und speziell abgesichert werden.

Grundsätzlich sollten nur personalisierte Admin-Konten verwendet werden, da sonst keine Nachvollziehbarkeit der getätigten Aktionen möglich ist. Der Built-In Admin ist besonders problematisch, weil er spezielle Eigenschaften hat. So kann dieser User durch einen Brute-Force-Angriff nicht gesperrt werden. Dieses Verhalten macht diesen User zwar besonders angreifbar für Brute-Force-Angriffe, macht ihn aber den perfekten User für Notfälle (z.B. alle anderen Domain Accounts wurden durch Brute-Force-Angriffe gesperrt).

Gegenmaßnahmen

Der Built-In Admin sollte im normalen Betrieb nicht genutzt werden. Sein Passwort sollte möglichst lang und zufällig gesetzt werden und beispielsweise im Safe abgelegt werden. Zur Administration sollten nur personalisierte Domain-Admin-Konten verwendet werden.

Zusätzlich sollte der Built-In Admin speziell abgesichert werden. Mehr dazu in der Microsoft-Dokumentation. [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. Zu viele Domain Admins

Niedrig

Betroffene Systeme

- Active Directory

Beschreibung

In dem getesteten Active Directory gibt es unüblich viele Konten mit Domain Admin-Rechten. Das ist gefährlich, weil die Übernahme nur eines der Konten zur kompletten Übernahme des Active Directory führt.

Es konnten folgende Konten mit Domain Admin-Rechten identifiziert werden:

- AliceT0
- GustavT0
- HedwigT0
- CarolT0
- FileAdmin

Laut Auskunft werden eigentlich nur `AliceT0` und `GustavT0` regelmäßig benutzt. Die anderen Accounts sollten aus der Gruppe der Domain Admins entfernt werden.

Gegenmaßnahmen

Alle Konten sollten nur die Rechte haben, die sie wirklich benötigen. Das ist besonders bei den höchst privilegierten Konten wichtig.

In der Gruppe der Domain Admins sollten außer dem eingebauten Admin nur personalisierte Konten sein. Es sollte regelmäßig überprüft werden, ob die Personen, die Domain Admin-Konten haben, diese immer noch benötigen.

Service Accounts sollten im Normalfall keine Domain Admin-Rechte haben.

9. KRBTGT Passwort lange nicht geändert Niedrig

Betroffene Systeme

- Active Directory

Beschreibung

Das Active Directory-Konto `krbtgt` ist essenziell für das Kerberos-Authentifizierungsprotokoll. Alle Kerberos-Tickets werden mit dem Passwort-Hash dieses Kontos signiert. Kennt eine bösartige Person diesen Hash, kann sie beliebige gültige Kerberos-Tickets erstellen und sich so z. B. als Domain Admin ausgeben. Ein solcher Missbrauch wird als *Golden Ticket Attack* bezeichnet[1].

Im vorliegenden Fall wurde das Passwort des `krbtgt`-Kontos zuletzt am `01. März 2010` geändert.

Gegenmaßnahmen

Das Konto `krbtgt` hat eine Passwort-History von `2`. Das bedeutet, dass Kerberos-Tickets, die mit dem alten Passwort signiert wurden, weiterhin akzeptiert werden. Eine einfache Passwortrotation reicht daher nicht aus, um einen Missbrauch zu verhindern. Es ist notwendig, das Passwort **zweimal** zu rotieren, wobei zwischen den Änderungen die Synchronisierung der Domain Controller abgewartet werden muss. Andernfalls könnten Synchronisierungsprobleme auftreten.

Jede Person mit Domain-Admin-Rechten hat Zugriff auf den Hash des `krbtgt`-Kontos. Sollte eine solche Person das Unternehmen verlassen, muss das Passwort des `krbtgt`-Kontos aus Sicherheitsgründen ebenfalls **zweimal** geändert werden.

Bei einem vermuteten Angriff auf das Active Directory ist es ebenfalls unerlässlich, nach Schließen der Eintrittsschwachstelle das Passwort des `krbtgt`-

Kontos **zweimal** zu rotieren. Dies verhindert, dass Angreifer weiterhin gültige Kerberos-Tickets ausstellen können.

Es wird außerdem empfohlen, das Passwort des `krbtgt`-Kontos regelmäßig (z. B. monatlich) zu ändern. Bei regelmäßiger Rotation wird sichergestellt, dass ein möglicherweise kompromittierter Hash nach spätestens zwei Änderungen nicht mehr gültig ist.

Eine Anleitung zur manuellen Änderung des `krbtgt`-Passworts finden Sie in der Microsoft-Dokumentation[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

Betroffene Systeme

- Active Directory

Beschreibung

Active Directory, wie wir es kennen, wurde mit Windows 2000 eingeführt. Zuvor gab es bereits ähnliche Funktionalitäten, aber sie waren viel begrenzter. Ein Unterschied besteht darin, dass es keine hierarchische Struktur war, sondern flach. Das bedeutet, dass jeder alle Attribute jedes Objekts lesen konnte. Active Directory begrenzt dies. Ein normaler User kann nicht mehr sensible Attribute anderer User lesen, wie das `pwdLastSet`-Attribut. Das Problem war und ist, dass einige Anwendungen Zugriff auf diese Informationen benötigen. Deshalb hat Microsoft die `Pre-Windows 2000 Compatible Access`-Gruppe eingeführt, die wieder Leseberechtigungen für alle Attribute aller Objekte gewährt. Und um sicherzugehen, wurden `Authenticated Users` dieser Gruppe hinzugefügt.

Wenn `Authenticated Users` nicht in dieser Gruppe sind, kann es für Angreifer sehr ärgerlich sein. Wie bereits gesagt, ist eines der interessanten Attribute, das ein Angreifer nicht mehr lesen kann, `pwdLastSet`. Das bedeutet, es wird für Angreifer viel schwieriger, Konten mit alten und möglicherweise schwachen Passwörtern zu finden.

Es gibt jedoch einen wichtigen Punkt! Da dies eine Standard-Schwachstelle im AD ist, verlassen sich einige Tools auf diese „Funktionalität“. Deshalb muss eine Änderung vorher ausführlich getestet werden!

Gegenmaßnahmen

`Authenticated Users` sollten aus der `Pre-Windows 2000 Compatible Access`-Gruppe entfernt werden. Es sollte aber ausführlich getestet werden,

> ob diese Änderung negative Auswirkungen auf irgendwelche Software hat. Weiterführende Informationen finden sich in [1].

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

Impressum

VidraSec e.U. – https://www.vidrasec.com/
Anschrift: Almesbergerweg 3, 4160 Aigen-Schlägl, AUSTRIA
Firmenbuchnummer: 623204b
Firmenbuchgericht: Landesgericht Linz
Gewerbeaufsichtbehörde: BH Rohrbach
UID-Nummer: ATU80334229

Kontaktdaten
E-Mail: martin@vidrasec.com
Telefon: +43 670 3081275

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