2. Management Summary

DAVISEC GmbH wurde von Max Mustermann beauftragt einen externen Penetrationstest im Zeitraum 14. April 2025 bis 09. Mai 2025 durchzuführen. Der Test wurde unter der Leitung von David Hofer gemeinsam mit Viktoria Horvathova durchgeführt.

Es wurden insgesamt 144 erreichbare Hosts und mehrere Webapplikationen auf Schwachstellen überprüft. Zur Unterstützung des Tests wurden dem Pentest-Team Benutzerkonten mit den jeweils niedrigsten verfügbaren Berechtigungsstufen für drei zentrale Webanwendungen bereitgestellt. Durch diese Zugriffsmöglichkeiten konnten gezielt Tests aus der Perspektive eines „Low-Privilege"-Benutzers durchgeführt werden, um realistische Angriffsszenarien zu simulieren.

Im Rahmen des Tests wurden 11 Schwachstellen identifiziert, sowie 3 rein informative Beobachtungen, welche in Tabelle 1 nach ihrem Schweregrad gelistet und in den folgenden Kapiteln ausführlicher beschrieben werden.

Kritisch	Hoch	Mittel	Niedrig	Informativ
1	3	5	2	3

Tabelle 1: Identifizierte Schwachstellen nach Schweregrad

Zwei Schwachstellen verdienen besondere Erwähnung aufgrund ihrer potenziellen Auswirkungen:

1. **Benutzerkonten-Übernahme mittels „Passwort zurücksetzen"**
   Ermöglichte in Kombination mit weiteren Schwachstellen die Übernahme aller Benutzerkonten in der Mitarbeiter Web-Anwendung.

2. **Fehlende Zugriffskontrolle bei Verkaufsanzeigen**
   Jeder authentifizierte Benutzer kann beliebige Anzeigen bearbeiten, wodurch ein Angreifer in der Lage ist, Anzeigen anderer Benutzer unautorisiert zu manipulieren.

3. **Fehlende Zugriffsbeschränkung auf Benutzerdaten**
   Führte zur vollständigen Offenlegung personenbezogener Informationen anderer Nutzer.

Besonders positiv hervorzuheben ist die reibungslose Kommunikation mit den Ansprechpartnern sowie die schnelle und zielgerichtete Reaktion auf kritische Schwachstellen. Das kritischste Finding wurde bereits während des Testzeitraums behoben – ein deutliches Zeichen für ein ausgeprägtes Sicherheitsbewusstsein und effiziente interne Prozesse. Auch die hohe Anzahl an Hosts, bei denen im Testzeitraum keine Schwachstellen festgestellt werden konnten, unterstreicht diesen positiven Eindruck.

6.2. Vollständige Kompromittierung des Unternehmensnetzwerks über unsichere Zertifikatstemplates (Domain-Admin)

CVSS-Wert	CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:H/SI:H/SA:H	9,4 Kritisch
Referenzen	https://attack.mitre.org/techniques/T1649/	
Status	DAVISEC verified	

Generelle Beschreibung

In Active Directory Certificate Services (AD CS) können falsch konfigurierte oder zu weitreichend berechtigte Zertifikatstemplates missbraucht werden, um privilegierte Benutzerkonten, einschließlich Domain-Administratoren, zu kompromittieren.

Ein häufiges Risiko entsteht, wenn Templates die Einstellung „ENROLLEE_SUPPLIES_SUBJECT“ (Benutzer kann den Subject-Namen selbst angeben) in Kombination mit „Client Authentication“ zulassen. Dadurch können Angreifer Zertifikate für beliebige AD-Konten anfordern und sich mit diesen über Kerberos (PKINIT) authentifizieren.

Besonders kritisch ist dies, wenn ein Angreifer ein Konto kompromittiert, das entweder über ENROLLMENT-Berechtigungen für das betreffende Zertifikatstemplate verfügt oder selbst als Besitzer des Templates eingetragen ist. Als Besitzer kann ein Benutzer sich selbst die erforderlichen Enroll-Rechte gewähren oder die zuvor genannte gefährliche Berechtigungskombination aktivieren.

Details

Das Testteam stellte im Rahmen der Testaktivitäten fest, dass insgesamt **vier Zertifikatstemplates** die kritische Kombination aus **„ENROLLEE_SUPPLIES_SUBJECT“** und **„Client Authentication“** aufweisen (siehe Abbildung 5).

Name	Destination	Manager approval	Enrollee can supply subject	Issuance requirements	Vulnerable ACL	For Authentication
	Computer	NO	YES	NO	NO	YES
	User	NO	YES	NO	NO	YES
Template1	User	NO	YES	NO	NO	YES
	Computer	NO	YES	NO	NO	YES

Abbildung 5: Zertifikatstemplates mit kritischen Berechtigungen

Alle vier Templates wären unter geeigneten Umständen für eine Kompromittierung missbrauchbar, jedoch zeigte die Analyse der Access Control Lists (ACLs), dass aktuell nur

hochprivilegierte Benutzer, Computer oder Gruppen – mit Berechtigungen auf dem Niveau von Domain Admins – über Enrollment-Rechte verfügen. Daher erschien ein direkter Missbrauch zunächst nicht möglich.

Allerdings berücksichtigen viele Tools zur Enumeration verwundbarer Zertifikatstemplates – wie beispielsweise PingCastle oder Certify.exe – nicht, wer als Besitzer (Owner) eines Templates eingetragen ist. Der Besitzer eines Active-Directory-Objekts besitzt jedoch umfassende Rechte und kann nahezu alle Änderungen an diesem Objekt vornehmen – einschließlich der Vergabe eigener Enroll-Rechte oder dem Anpassen kritischer Template-Einstellungen.

Das Testteam überprüfte daher zusätzlich den Besitzstatus der identifizierten Templates. Dabei stellte sich heraus, dass mehrere Templates dem bereits im Finding 6.1 beschriebenen, zuvor erfolgreich kompromittierten, Benutzerkonto *fischer* gehörten. Auf Basis dieser Informationen wählte das Testteam ein spezifisches Zertifikatstemplate „Template1“ aus und änderte dessen Berechtigungen mithilfe des Benutzerkontos *fischer*, sodass dieser sich Zertifikate selbst ausstellen konnte.

Im nächsten Schritt wurde ein Zertifikat für das Domain-Admin-Konto *domadm* ausgestellt und erfolgreich importiert. Anschließend konnte mit den erlangten Berechtigungen exemplarisch auf das C:-Laufwerk des Domain Controllers zugegriffen werden. Dies demonstriert, dass die Kompromittierung des Domain Admins erfolgreich war.

Empfehlung

Es wird empfohlen, die Konfiguration der identifizierten Zertifikatstemplates in Active Directory Certificate Services (AD CS) zeitnah zu überprüfen und anzupassen:

- **Besitzrechte korrigieren**: Ausschließlich hoch privilegierte Gruppen (z. B. Enterprise Admins oder Domain Admins) sollen als Besitzer von Zertifikatstemplates eingetragen sein. Entfernen Sie alle unprivilegierten Benutzerkonten (fischer) aus den Besitzrechten.
- **Enrollment-Berechtigungen restriktiv vergeben**: Gewähren Sie Enrollment-Rechte nur vertrauenswürdigen Sicherheitsgruppen und vermeiden Sie, dass Standardbenutzer oder nicht privilegierte Accounts diese Rechte erhalten.
- **Gefährliche Kombinationen deaktivieren**: Entfernen oder ersetzen Sie Templates, die gleichzeitig „*ENROLLEE_SUPPLIES_SUBJECT*“ und „*Client Authentication*“ zulassen. Wo diese Funktionalität benötigt wird, sollte eine strikte Einschränkung der berechtigten Benutzergruppe erfolgen.
- **Regelmäßige Überprüfung**: Führen Sie wiederkehrend eine technische Überprüfung der Zertifikatstemplates und deren Berechtigungen durch (z. B. mit Tools wie Certify.exe oder PSPKI), um Fehlkonfigurationen frühzeitig zu erkennen.

5.1. Benutzerkonten-Übernahme mittels „Passwort zurücksetzen“

CVSS-Wert	CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:H/SI:H/SA:H	9,4 Kritisch
Referenzen	https://portswigger.net/web-security/access-control/idor https://cwe.mitre.org/data/definitions/639.html	
Status	DAVISEC verified	

Generelle Beschreibung

Wenn Änderungen an Benutzerkonten nicht ausreichend gegen unautorisierte Manipulation abgesichert sind, besteht die Gefahr, dass Benutzerkonten von anderen Benutzern übernommen oder verändert werden. Dadurch können Angreifer u. a. eine fremde E-Mail-Adresse hinterlegen, das Passwort zurücksetzen und so Konten übernehmen. In Verbindung mit SSO kann ein kompromittiertes Konto zudem Zugang zu weiteren Diensten erhalten.

Details

Beim Speichern von Benutzerprofilen kann durch fehlende Autorisierung ein beliebiger, angemeldeter Benutzer über die Mitarbeiter Web-Applikation eine neue E-Mail-Adresse in einem beliebigen anderen Benutzerkonto hinterlegen. Diese Adresse lässt sich anschließend über die Passwort-zurücksetzen-Funktion nutzen, um das Zielkonto zu übernehmen. Da die Anmeldung zu Dritt-Diensten automatisch über SSO erfolgt, ermöglicht eine Kontoübernahme auch Zugriff auf diese Dienste.

Wird bei den Benutzerkontoeinstellungen auf „Speichern“ der Änderungen geklickt, so wird im Hintergrund ein POST-Request an den „/api/user/save“ API-Endpunkt abgesetzt. Der Body dieses HTTP-Requests besteht dabei aus den gesamten Daten des Benutzers (siehe Abbildung 1).

Allerdings wird am Server nicht verifiziert, dass es sich bei dem Benutzer im HTTP-Request-Body (identifiziert durch seine „employeeNr“) auch tatsächlich um denselben Benutzer handelt, der am Service angemeldet ist und sich durch seinen Bearer-Token im Authorization-Header identifiziert.

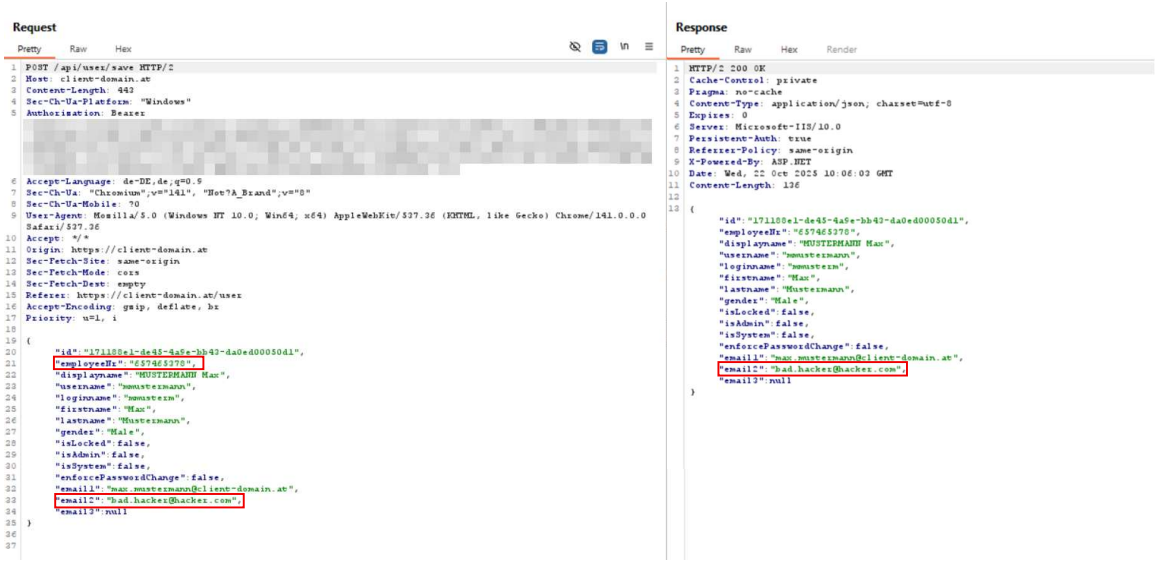

Abbildung 1: Hinterlegen einer neuen E-Mail-Adresse bei einem beliebigen anderen Benutzerkonto

Das erlaubt es einem beliebigen, angemeldeten Benutzer, die Daten eines beliebigen anderen Benutzers zu bearbeiten und überschreiben, sofern dessen „employeeNr“ bekannt ist. In Verbindung mit den anderorts gelisteten Schwachstellen bezüglich der Preisgabe beliebiger Benutzerinformationen (inklusive „employeeNr“) und mittels der „Passwort zurücksetzen“-Funktion kann praktisch das Benutzerkonto eines beliebigen Mitarbeiters übernommen werden, indem einfach eine neue E-Mail-Adresse mittels dieses API-Endpunkts hinterlegt wird und diese dann zum Zurücksetzen des Passworts verwendet wird.

Die „Passwort zurücksetzen“-Funktion kann einfach über den „/api/auth/ForgotPassword“ API-Endpunkt verwendet werden. Dorthin muss die „employeeNr“, die hinterlegte E-Mail-Adresse und um welche der hinterlegten E-Mails es sich handelt, gesendet werden.

Empfehlung

Folgende Maßnahmen werden empfohlen:

- Die Server-seitige Autorisierung muss bei jeder Änderung sicherstellen, dass nur der aktuell authentifizierte Benutzer auf sein eigenes Benutzerobjekt zugreifen und dieses ändern kann.
- Der Bearer-Token im Authorization-Header muss mit der im Request-Body übermittelten Benutzer-ID (employeeNr) abgeglichen werden.
- Der E-Mail-Änderungsprozess soll so gestaltet sein, dass neue E-Mail-Adressen nur dann hinzugefügt werden können, wenn die Änderung über eine bestehende, verifizierte E-Mail-Adresse bestätigt wird.