Sagorski

PENETRATIONSTEST-BERICHT · BEISPIEL

Externer Penetrationstest einer öffentlichen Verwaltung

Anonymisiertes Beispiel eines Abschlussberichts. Alle Namen, Adressen, Hostnamen, IP-Adressen und Zugangsdaten wurden entfernt oder verfremdet. Technische Nachweise (PoC) sind nicht enthalten.

AUFTRAGGEBER

Öffentliche Verwaltung (Kommune, ca. 260.000 Einwohner)

AUFTRAGNEHMER

Sagorski · patchletter UG (haftungsbeschränkt)

TESTART

Externer Black-Box-Pentest (Remote)

SCOPE

Öffentlich erreichbare Dienste der Domäne

BERICHTSSTAND

Beispielversion

FINDINGS

25 (2 kritisch, 9 hoch, 10 mittel, 4 niedrig)

● VERTRAULICH · BEISPIEL & ANONYMISIERT · NUR ZU DEMONSTRATIONSZWECKEN

ÜBER DIESES DOKUMENT

Ein echter Bericht – vollständig anonymisiert

Dieser Beispielbericht beruht auf einem realen, beauftragten Penetrationstest. Er zeigt, wie ein Pentest-Bericht aufgebaut ist: nachvollziehbares Vorgehen, klar priorisierte Findings und zu jedem Finding eine Sofortmaßnahme sowie eine Empfehlung zur dauerhaften Behebung.

Was wurde für die Veröffentlichung entfernt?

Organisationsname, Domains und Hostnamen, interne IP-Adressen, System-IDs, Datei- und Speicherpfade, Benutzernamen, erbeutete Passwörter und Hashes sowie sämtliche personenbezogenen Daten (Beschäftigte und Bürger). Ebenfalls entfernt: funktionierende Exploit-Payloads und Reproduktions-Befehle. Ein realer Bericht an den Auftraggeber enthält diese Nachweise vollständig und wird ausschließlich verschlüsselt und vertraulich übergeben.

MANAGEMENT SUMMARY

Gesamtbild: kritische Schwächen in zentralen Systemen

Aus dem Internet – ohne gültige Zugangsdaten – ließen sich mehrere zentrale Fachverfahren der Verwaltung kompromittieren. Im Ergebnis war unter anderem eine administrative Übernahme des zentralen ERP-Systems möglich. Daneben traten Schwächen bei Authentifizierung, Zugriffskontrolle (IDOR), SSRF, SQL-Injection und Konfiguration auf.

2	9	10	4	25
KRITISCH	HOCH	MITTEL	NIEDRIG	FINDINGS GESAMT



KRITISCH HOCH MITTEL NIEDRIG

Die wichtigsten Erkenntnisse

\ Das zentrale ERP-System war über eine offene Programmierschnittstelle (RFC/SOAP) ohne Anmeldung ansprechbar. Darüber ließen sich Passwort-Hashes auslesen, schwache Standardpasswörter offline zurückrechnen und privilegierte Konten (Vollberechtigung) übernehmen.

- Über die gleiche Schnittstelle waren personenbezogene Massendaten (Einwohner- und Personaldaten) sowie im Klartext gespeicherte Anwendungspasswörter erreichbar – ein meldepflichtiger Datenschutzvorfall (DSGVO Art. 33/34).
- Im Geodatendienst erlaubten SSRF und eine SQL-Injection den Zugriff auf interne, eigentlich gesperrte Dienste bzw. das Ausforschen der Datenbank.
- Der zentrale Anmeldedienst (SSO) und mehrere Self-Service- und Bürgerportale wiesen Schwächen bei Authentifizierung, Zugriffskontrolle und Eingabevalidierung auf.

Dringende Empfehlung

Die als kritisch und hoch bewerteten Findings sollten umgehend behandelt werden – beginnend mit der Abschottung der offenen ERP-Schnittstelle vom Internet und dem Zurücksetzen aller betroffenen bzw. mit Standardpasswörtern versehenen Konten. Die Sofortmaßnahmen je Finding sind so gewählt, dass sie das akute Risiko kurzfristig senken, auch bevor die dauerhafte Behebung umgesetzt ist.

VORGEHEN

Was wurde wie getestet?

Durchgeführt wurde ein externer Penetrationstest aus Sicht eines nicht authentifizierten Angreifers im Internet. Geprüft wurden ausschließlich die öffentlich erreichbaren Dienste des Auftraggebers; Tests liefen ratenbegrenzt (≈ 1 Request/Sekunde) und nicht-destruktiv. Ausnutzungen erfolgten nur so weit, wie es zum belastbaren Nachweis eines Findings nötig war.

Umfang (Scope)

Der Scope umfasste die aus einer Domäne erreichbare Angriffsfläche. Die automatisierte und manuelle Erkundung erfasste:

Kategorie	Größenordnung
Subdomains	> 1.000
Root-Domains	> 100
Erreichbare Endpunkte	> 2.200
Dienste / Ports	> 500

Kategorie	Größenordnung
IP-Adressen	> 400
Web-Anwendungen	ca. 40

Prüfschritte

- Recon & Asset-Discovery – Ermittlung von Subdomains, Diensten und Endpunkten; Bestimmung der Angriffsfläche.
- API-Endpunkt-Entdeckung – Auffinden undokumentierter bzw. ungeschützter Schnittstellen und Funktionsmodule.
- Authentifizierung & Session-Management – Prüfung von Login-Mechanik, Session-Cookies, Passwort-Reset und SSO/OAuth-Abläufen.
- Zugriff auf Admin- & Dashboard-Funktionen – Prüfung auf fehlende/umgehbare Zugriffskontrollen.
- Schwachstellenanalyse – gezielte Tests auf SQL-Injection, XSS, IDOR, SSRF, LDAP-Injection, Auth-Bypass und Fehlkonfiguration.
- Kontrollierter Nachweis (PoC) – belastbarer Beweis je Finding, inkl. Negativkontrollen, ohne produktive Daten zu verändern.

Methodik & Bewertung

Die Durchführung orientiert sich an etablierten Leitfäden (u. a. OWASP WSTG, OWASP Top 10 & API Security Top 10, PTES, NIST SP 800-115). Die Schweregrade folgen der Logik von CVSS/Risiko (Eintrittswahrscheinlichkeit × Auswirkung) und sind in vier Stufen zusammengefasst:

Stufe	Bedeutung
● KRITISCH	Unmittelbare, gravierende Kompromittierung möglich (z. B. Systemübernahme, Massenabfluss personenbezogener Daten). Sofort handeln.
● HOCH	Erheblicher Schaden mit geringem Aufwand realisierbar. Kurzfristig beheben.
● MITTEL	Relevantes Risiko, oft mit Vorbedingung oder begrenzter Reichweite. Geplant beheben.
● NIEDRIG	Geringes Einzelrisiko, meist Information Disclosure / Härtung. Im Zuge der Pflege beheben.

Hinweis zur Zählung: Es wurden 27 Rohbefunde erhoben; zwei Duplikat-Paare (gleiche Ursache) wurden zu je einem Finding zusammengeführt. Es verbleiben 25 Einzel-Findings.

DETAILBEFUNDE

Findings im Detail

Nach Zielsystem gruppiert, innerhalb der Gruppe nach Schweregrad sortiert. Jedes Finding nennt eine Sofortmaßnahme (kurzfristige Risikominderung) und eine Empfehlung (dauerhafte Behebung).

A · Zentrales ERP-System

9 FINDINGS · 2 KRITISCH · 5 HOCH · 2 MITTEL

Getestet wurde die aus dem Internet erreichbare Programmierschnittstelle (SOAP/RFC) des ERP. Sie nahm Funktionsaufrufe ohne jede Authentifizierung entgegen. Schutzmechanismen der Standardsoftware wurden teils durch kundeneigene Funktionsmodule umgangen, die keine Berechtigungsprüfung durchführen.

A-01 ● KRITISCH

Übernahme privilegierter ERP-Konten über Standard-/Initialpasswörter

SYSTEM	KATEGORIE	STATUS
ERP-System	Broken Authentication	Bestätigt

BESCHREIBUNG

Über die offene Schnittstelle ließen sich die Passwort-Hashes der Benutzerkonten auslesen. Ein Teil der Konten – darunter hochprivilegierte Dienst- und Administratorkonten mit Vollberechtigung – nutzte weiterhin unveränderte Standard-/Initialpasswörter und zusätzlich ein veraltetes, schwaches Hash-Verfahren. Die Hashes ließen sich offline in kurzer Zeit zurückrechnen; die so gewonnenen Zugangsdaten wurden gegen die Schnittstelle als gültige Anmeldung bestätigt.

AUSWIRKUNG

Vollständige administrative Kontrolle über das produktive ERP aus dem Internet – ohne vorherige Zugangsdaten. Das umfasst Benutzerverwaltung, Zugriff auf Fach- und Personaldaten sowie einen Ausgangspunkt für weitere Ausbreitung ins interne Netz.

SOFORTMASSNAHME

Betroffene Dienst- und Administratorkonten sofort sperren bzw. Passwörter zurücksetzen und erzwungen ändern. Die Schnittstelle unverzüglich vom Internet trennen (Netzsegmentierung / IP-Allowlisting / VPN).

EMPFEHLUNG

Veraltete Hash-Verfahren deaktivieren (nur aktuelle, salted Verfahren zulassen), Passwortrichtlinie und -historie erzwingen, privilegierte Konten auf das Nötigste reduzieren und mit starken, rotierten Secrets versehen. Schnittstelle nur authentifiziert und netzseitig eingeschränkt betreiben.

A-02 ● KRITISCH

Auslesen von Passwort-Hashes und Klartext-Zugangsdaten über ein unsicheres Kundenmodul

SYSTEM	KATEGORIE	STATUS
ERP-System	Broken Access Control / Info Disclosure	Bestätigt

BESCHREIBUNG

Ein kundeneigenes Funktionsmodul wirkte als generischer Tabellen-Leser und war ohne Authentifizierung aufrufbar. Damit ließen sich beliebige Datenbanktabellen auslesen – u. a. die Benutzerstammtabelle mit Passwort-Hashes, die Berechtigungstabelle (welche Konten Vollrechte halten) sowie kundeneigene Tabellen mit im Klartext gespeicherten Anwendungspasswörtern namentlich benannter Beschäftigter. Auch personenbezogene Massendaten waren erreichbar.

AUSWIRKUNG

Massenhafte Offenlegung von Zugangsdaten und personenbezogenen Daten; direkte Grundlage für Kontoübernahmen. Meldepflichtiger Datenschutzvorfall (DSGVO Art. 33/34).

SOFORTMASSNAHME

Das Kundenmodul aus der extern erreichbaren Schnittstelle entfernen bzw. sperren, Schnittstelle vom Internet trennen, betroffene Zugangsdaten rotieren.

EMPFEHLUNG

Generische Tabellen-Leser und Fernaufruf-Module grundsätzlich deaktivieren oder streng berechtigen; in allen kundeneigenen Modulen eine serverseitige Berechtigungsprüfung (Authority-Check) nachrüsten; keine Anwendungspasswörter im Klartext speichern (Hashing bzw. Secret-Store).

A-03 HOCH

Unauthentifizierte RFC/SOAP-Schnittstelle: System-, Schema- und Benutzer-Preisgabe

SYSTEM	KATEGORIE	STATUS
ERP-System	Missing Authentication	Bestätigt

BESCHREIBUNG

Die Schnittstelle nahm Funktionsaufrufe von beliebigen, nicht authentifizierten Clients entgegen. Darüber ließen sich der interne System-Fingerabdruck (Systemkennung, Datenbank, Host, Patch-Stand, interne Adressen), beliebige Tabellen-Schemata und ein zuverlässiges „Konto existiert ja/nein"-Orakel abfragen.

AUSWIRKUNG

Preisgabe interner Infrastruktur- und Identitätsinformationen, die gezielte Folgeangriffe (passgenaue Exploits, Credential-Angriffe) erheblich erleichtern. Zugleich die gemeinsame Ursache mehrerer weiterer ERP-Findings.

SOFORTMASSNAHME

Zugriff auf die Schnittstelle aus dem Internet unterbinden (Reverse-Proxy-/Firewall-Regel, Allowlisting). Anonymen Funktionsaufruf abschalten.

EMPFEHLUNG

Die Schnittstelle grundsätzlich authentifizieren, auf die wirklich benötigten Funktionsbausteine beschränken und in ein internes Netzsegment verlagern. Fehlermeldungen vereinheitlichen, um das Benutzer-Orakel zu schließen.

A-04 HOCH

Unauthentifizierte Code-Ausführung (ABAP) über Kundenmodule

SYSTEM	KATEGORIE	STATUS
ERP-System	Remote Code Execution	Bestätigt

BESCHREIBUNG

Kundeneigene Funktionsmodule erlaubten es, serverseitig eigenen Programmcode zu hinterlegen und auszuführen. Im Nachweis wurden selbst geschriebene Routinen ausgeführt und lieferten serverseitig berechnete Werte zurück. Zugriffe auf Betriebssystemebene waren durch Berechtigungen des Dienstkontos begrenzt.

AUSWIRKUNG

Ausführung von Angreifer-Code in der Anwendungsschicht des produktiven ERP ohne Anmeldung. Die weitere Reichweite hängt allein von den Rechten des ausführenden Dienstkontos ab.

SOFORTMASSNAHME

Die betroffenen Module extern sperren und die Schnittstelle vom Internet trennen; ausgeführte/hinterlegte Fremdobjekte durch die Administration prüfen und bereinigen.

EMPFEHLUNG

Module, die Quellcode entgegennehmen oder ausführen, entfernen bzw. strikt berechtigen; Rechte des Dienstkontos minimieren; Freigabe-/Transportprozess für kundeneigenen Code etablieren und regelmäßig auditieren.

A-05 ● HOCH

Anlegen eines anmeldefähigen, privilegierten Kontos ohne Authentifizierung

SYSTEM	KATEGORIE	STATUS
ERP-System	Authentication Bypass / Persistence	Bestätigt

BESCHREIBUNG

Aufbauend auf der Code-Ausführung (A-04) ließ sich ein vollständig neues, anmeldefähiges Benutzerkonto mit selbst gewähltem Passwort und hoher Berechtigungszuordnung anlegen. Die anschließende Anmeldung mit diesem Konto gelang.

AUSWIRKUNG

Ein Angreifer kann sich einen dauerhaften, von echten Beschäftigtenkonten kaum unterscheidbaren Zugang zum ERP schaffen – unabhängig vom Patchen der Weboberfläche.

SOFORTMASSNAHME

Benutzerstamm auf unbekannte/kürzlich angelegte Konten prüfen und diese sperren; Schnittstelle vom Internet trennen (siehe A-03/A-04).

EMPFEHLUNG

Schreibende Operationen auf Benutzer-/Berechtigungstabellen ausschließlich über geprüfte, berechtigte Pfade zulassen; Alarmierung bei Neuanlage privilegierter Konten; regelmäßiger Abgleich des Benutzerstamms.

A-06 ● HOCH

Unauthentifizierte Enumeration des Server-Dateisystems

SYSTEM	KATEGORIE	STATUS
ERP-System	Broken Access Control / Info Disclosure	Bestätigt

BESCHREIBUNG

Ein Kundenmodul listete beliebige Verzeichnisse des Betriebssystems auf (Dateinamen, Größen, Zeitstempel, Rechte). Datei-inhalte blieben durch Berechtigungen gesperrt; offengelegt wurde die komplette Dateisystem-Struktur inkl. Dienstkonten, Schlüsselspeicher-Dateinamen und Software-Bestand.

AUSWIRKUNG

Detaillierte Aufklärung des Zielsystems (Konten-, Schlüssel- und Software-Inventar) als Steilvorlage für Folgeangriffe.

SOFORTMASSNAHME

Das Verzeichnis-Modul extern sperren; Schnittstelle vom Internet trennen.

EMPFEHLUNG

Dateisystem-Zugriffsmodule entfernen oder streng berechtigen; Prinzip der minimalen Rechte für das Dienstkonto.

A-07 ● HOCH

Unauthentifizierter Zugriff auf personenbezogene Daten (Einwohner- und Personaldaten)

SYSTEM	KATEGORIE	STATUS
ERP-System	Broken Access Control / PII Exposure	Bestätigt

BESCHREIBUNG

Kundeneigene Fachmodule gaben ohne Anmeldung personenbezogene Datensätze zurück – u. a. einen Melderegister-Auszug (Name, Geburtsdatum, Anschrift, weitere Merkmale) mit gezielter Einzelabfrage je Person sowie einen Personaldatensatz mit dienstlicher Kennung und Verzeichnis-Identität.

AUSWIRKUNG

Offenlegung besonders schutzwürdiger Daten in großem Umfang; meldepflichtiger Datenschutzvorfall und Grundlage für gezielte Angriffe auf Beschäftigte.

SOFORTMASSNAHME

Die datenliefernden Module extern sperren; Schnittstelle vom Internet trennen; Datenschutz-Meldeprozess prüfen/auslösen.

EMPFEHLUNG

Jedes personenbezogene Fachmodul mit einer verbindlichen Authentifizierungs- und Berechtigungsprüfung versehen; Zugriff protokollieren; Datenminimierung.

A-08 HOCH

Anonymes Anlegen und Ändern von Benutzerkonten eines Fachportals

SYSTEM	KATEGORIE	STATUS
ERP-System (Fachportal)	Broken Access Control	Bestätigt

BESCHREIBUNG

Ein Funktionsmodul für die Benutzerverwaltung eines Fachportals führte Anlegen/Ändern/Löschen in der Konten-Tabelle ohne jede Berechtigungsprüfung aus. Im Nachweis wurde ein Testkonto angelegt, in die höchste Benutzergruppe gehoben, das Passwort gesetzt und anschließend wieder gelöscht – ausschließlich das eigene Testobjekt.

AUSWIRKUNG

Vollständige Kontrolle über den Authentifizierungsbestand des Fachportals (Konto-Neuanlage, Passwort-Überschreiben, Rechte-Eskalation, Löschung).

SOFORTMASSNAHME

Schreibende Modul-Aufrufe extern sperren; Schnittstelle vom Internet trennen.

EMPFEHLUNG

Authentifizierung und Rollenprüfung für alle Verwaltungsfunktionen erzwingen; Passwörter nur salted/gehasht speichern; Audit-Logging für Kontoänderungen.

A-09 ● MITTEL

Umgehung der Passwortprüfung eines Kundenmoduls (optionaler Parameter)

SYSTEM	KATEGORIE	STATUS
ERP-System (Fachportal)	Broken Authentication	Bestätigt

BESCHREIBUNG

Ein Modul schützte seine Ausgabe mit einer optionalen Passwortprüfung. Wurde der Passwort-Parameter weggelassen, entfiel die Prüfung und das vollständige Konto (inkl. Passwort-Material und personenbezogenen Daten) wurde zurückgegeben; mit falschem Passwort wurde korrekt abgelehnt.

AUSWIRKUNG

Unauthentifiziertes Auslesen beliebiger Konten des Fachportals inkl. schwach geschütztem Login-Geheimnis.

SOFORTMASSNAHME

Modul extern sperren, bis die Prüfung verpflichtend ist.

EMPFEHLUNG

Sicherheitskritische Parameter nie optional führen; Prüfungen serverseitig erzwingen und bei Fehlen hart ablehnen (fail-closed).

B · Geodatendienst (GIS) 3 FINDINGS · 2 HOCH · 1 NIEDRIG

Getestet wurden die öffentlichen Kartendienste (WFS/WMS) hinter einem Relay. Geprüft wurde, ob clientseitige Parameter serverseitig ungefiltert verarbeitet werden.

B-01 HOCH

Server-Side Request Forgery (SSRF) im Druck-/Export-Dienst

SYSTEM	KATEGORIE	STATUS
Geodatendienst	SSRF	Bestätigt

BESCHREIBUNG

Ein anonym aufrufbarer Geoverarbeitungs-Dienst lud eine vom Client angegebene URL serverseitig nach. Dadurch ließen sich vom Server aus frei wählbare Hosts ansprechen – inklusive eines nur lokal erreichbaren internen Kartendienstes, dessen Daten über das öffentliche Relay eigentlich per Zugriffskontrolle gesperrt sind.

AUSWIRKUNG

Umgehung der Zugriffskontrolle und Lesen interner, gesperrter Geodaten; zudem ein Werkzeug, um das interne Netz hinter dem Dienst zu erkunden.

SOFORTMASSNAHME

Das Nachladen frei wählbarer URLs unterbinden; ausgehende Verbindungen des Dienstes auf benötigte Ziele per Allowlist beschränken.

EMPFEHLUNG

Server-seitiges Nachladen nur gegen eine feste Allowlist zulassen; interne Adressen/Metadaten-Endpunkte netzseitig blockieren; den internen Dienst nicht allein durch Relay-ACL, sondern eigenständig absichern.

B-02 ● HOCH

Blinde SQL-Injection in einem Geo-Filterparameter

SYSTEM	KATEGORIE	STATUS
Geodatendienst	SQL-Injection	Bestätigt

BESCHREIBUNG

Ein Filter-Literal einer anonymen WFS-Abfrage wurde ungefiltert in eine SQL-Bedingung des internen Backends eingefügt. Über einen zuverlässigen Wahr/Falsch-Indikator (Treffermenge) ließ sich die WHERE-Bedingung steuern, die Existenz nicht-öffentlicher Spalten prüfen und Spalteninhalte blind ausforschen.

AUSWIRKUNG

Unauthentifizierte Kontrolle über die Abfrage, Schema-Erkundung und schrittweises Auslesen von Daten. (UNION/Stacked Queries waren durch ein vorgelagertes Filter blockiert.)

SOFORTMASSNAHME

Den betroffenen Filterpfad vorübergehend abschalten oder strikt validieren (nur erlaubte Werte).

EMPFEHLUNG

Alle Filterwerte als Parameter binden (Prepared Statements), nie in SQL-Strings konkatenieren; Eingaben serverseitig gegen eine feste Grammatik validieren.

B-03 ● NIEDRIG

Datenbank-Schema-Preisgabe über ausführliche Fehlermeldungen

SYSTEM	KATEGORIE	STATUS
Geodatendienst	Information Disclosure	Bestätigt

BESCHREIBUNG

Ein fehlerhaft veröffentlichter Layer gab rohe Datenbank-Fehlertexte an anonyme Clients zurück und legte damit interne Schema- und Tabellennamen sowie den Datenbank-/Framework-Stack offen.

AUSWIRKUNG

Reine Aufklärung: erleichtert gezielte Folgeangriffe, liefert für sich allein keine Daten.

SOFORTMASSNAHME

Ausführliche Fehlerausgaben für externe Clients abschalten; den defekten Layer korrigieren oder deaktivieren.

EMPFEHLUNG

Generische Fehlerseiten ausliefern, interne Details nur serverseitig protokollieren; Konfiguration auf „produktiv" härten.

C · Zentraler Anmeldedienst (SSO)

3 FINDINGS · 1 HOCH · 1 MITTEL · 1 NIEDRIG

Getestet wurde der zentrale Identity-Provider (OAuth/OIDC & SAML), über den sich mehrere Fachanwendungen anmelden.

C-01 ● HOCH

OAuth/OIDC: Wildcard-Redirect-URI und aktiver Implicit-Flow

SYSTEM	KATEGORIE	STATUS
Anmeldedienst (SSO)	OAuth Misconfiguration	Bestätigt

BESCHREIBUNG

Ein OIDC-Client akzeptierte beliebige, vom Angreifer gewählte Redirect-Ziele und erlaubte zusätzlich den Implicit-Flow. Dadurch können Zugriffs-/ID-Tokens eines angemeldeten Opfers an eine Angreifer-Domain ausgeliefert werden. Ein Vergleichs-Client im selben System lehnte beides korrekt ab.

AUSWIRKUNG

Token-Abgriff und damit Sitzungsübernahme für die angebundene Anwendung, ausgelöst durch einen einzigen Klick des Opfers.

SOFORTMASSNAHME

Redirect-URIs des Clients auf exakte, vollständige URLs beschränken; Implicit-Flow deaktivieren.

EMPFEHLUNG

Authorization-Code-Flow mit PKCE erzwingen; Redirect-URIs ohne Wildcards registrieren; auch die Post-Logout-Redirects validieren; Client-Konfiguration regelmäßig auditieren.

C-02 ● MITTEL

Umgehung einer Zugriffssperre durch Groß-/Kleinschreibung im Pfad

SYSTEM	KATEGORIE	STATUS
Anmeldedienst (SSO)	Access Control Bypass	Bestätigt

BESCHREIBUNG

Eine vorgelagerte Sperre für den administrativen Bereich (Master-Realm) war case-sensitiv, die Anwendung dahinter jedoch nicht. Durch abweichende Groß-/Kleinschreibung im Pfad ließ sich die Sperre umgehen und die geschützte Admin-/Login-Oberfläche erreichen (ohne dass eine Anmeldung gelang).

AUSWIRKUNG

Die bewusst gesetzte Schutzschicht vor dem Super-Admin-Bereich ist wirkungslos; die Login-Oberfläche wird aus dem Internet angreifbar.

SOFORTMASSNAHME

Sperrregeln case-insensitiv formulieren und auf alle Pfadvarianten anwenden.

EMPFEHLUNG

Autorisierung nicht allein per Pfad-Pattern im Reverse-Proxy, sondern in der Anwendung durchsetzen; administrative Realms grundsätzlich nicht aus dem Internet exponieren.

C-03 ● NIEDRIG

Client-Enumeration über unterschiedliche Fehlermeldungen

SYSTEM	KATEGORIE	STATUS
Anmeldedienst (SSO)	Information Disclosure	Bestätigt

BESCHREIBUNG

Ein SAML-Endpunkt beantwortete Anfragen je nach Client-Kennung mit unterschiedlichen Fehlertexten. Daraus ließ sich ableiten, welche Clients existieren und welchen Protokoll-/Flow-Status sie haben (bekannte Schwachstellenklasse des eingesetzten Produkts).

AUSWIRKUNG

Aufklärung des angebundenen Anwendungsbestands; Vorbereitung gezielter Angriffe.

SOFORTMASSNAHME

Fehlermeldungen vereinheitlichen (generische Antwort unabhängig von Existenz/Status).

EMPFEHLUNG

Produkt auf die gepatchte Version aktualisieren; Endpunkte, die nicht benötigt werden, deaktivieren.

D · Mitarbeiter-Self-Service & Verwaltungsportale

5 FINDINGS · 4 MITTEL · 1 NIEDRIG

Getestet wurden interne Self-Service-Anwendungen (Passwort, 2FA, Kontosuche) und mehrere ASP.NET-Verwaltungsportale, jeweils auf Authentifizierung, Session und Eingabevalidierung.

D-01 ● MITTEL

LDAP-/Verzeichnis-Injection in der Kontosuche

SYSTEM	KATEGORIE	STATUS
Mitarbeiter-Self-Service	LDAP-Injection	Bestätigt

BESCHREIBUNG

Das Benutzernamen-Feld eines anonymen Self-Service-Ablaufs wurde ungefiltert in einen LDAP-/Verzeichnis-Suchfilter eingesetzt. Dadurch ließen sich mit Wildcards und Attribut-Prädikaten gültige Konten und deren Verzeichnis-Attribute unauthentifiziert erkunden.

AUSWIRKUNG

Unauthentifizierte Enumeration des Active-Directory-Kontenbestands und ein blinder Attribut-Orakel – wertvolle Vorarbeit für Credential-Angriffe.

SOFORTMASSNAHME

Sonderzeichen im Suchfeld serverseitig escapen/zurückweisen (LDAP-Filter-Escaping).

EMPFEHLUNG

Verzeichnis-Abfragen über sichere APIs mit parametrisierten Filtern; Eingaben auf eine feste Zeichen-/Längen-Grammatik begrenzen; Rate-Limiting.

D-02 ● MITTEL

Passwort-Reset ohne Identitätsprüfung mit vorhersagbarem Reset-Code

SYSTEM	KATEGORIE	STATUS
Mitarbeiter-Self-Service	Broken Authentication	Bestätigt

BESCHREIBUNG

Der Passwort-Reset-Ablauf gab den „Identitätsnachweis-Code" für jede Eingabe aus – auch für erfundene Beschäftigte – und der Code war pro Eingabe deterministisch (dieselben Daten erzeugten denselben Code). Er hängt nur von wenigen, auf einer Gehaltsabrechnung sichtbaren Datenpunkten ab.

AUSWIRKUNG

Der Reset-Code lässt sich offline aus bekannten Personendaten reproduzieren; zwischen Angreifer und Passwort-Reset steht nur noch der telefonische Service-Center-Prozess.

SOFORTMASSNAHME

Code nur nach erfolgreicher Identitätsprüfung ausgeben; nicht-deterministisch erzeugen und serverseitig an eine kurzlebige, einmalige Sitzung binden.

EMPFEHLUNG

Reset über einen zufälligen, einmaligen Token mit kurzer Gültigkeit und sicherem Zustellkanal; Service-Center-Prozess um einen unabhängigen Rückruf/Zweitfaktor ergänzen.

D-03 ● MITTEL

Anmeldung über frei wählbaren Session-Cookie (CMS)

SYSTEM	KATEGORIE	STATUS
Verwaltungsportal (CMS)	Broken Authentication / Session	Bestätigt

BESCHREIBUNG

Die Anwendung leitete ihre Autorisierung aus einem client-seitig frei setzbaren, ungeschützten Session-Cookie ab. Ein bestimmter, raterbarer Wert wurde als gültige Sitzung akzeptiert und gab login-geschützte CMS-Inhalte frei (die Editor-/Admin-Bereiche blieben gesperrt).

AUSWIRKUNG

Zugriff auf eigentlich angemeldungspflichtige Inhalte durch beliebige Clients; strukturelle Schwäche des Session-Konzepts.

SOFORTMASSNAHME

Keine Autorisierung aus client-gelieferten Werten ableiten; Sitzungen serverseitig führen und nach Login neu ausstellen.

EMPFEHLUNG

Signierte/zufällige Session-IDs aus einem serverseitigen Store; Cookies mit HttpOnly, Secure und SameSite; Session-Fixation-Schutz.

D-04 ● MITTEL

Benutzernamen-Enumeration und Kontostatus-Preisgabe am Login

SYSTEM	KATEGORIE	STATUS
Verwaltungsportal (Statistik)	Information Disclosure	Bestätigt

BESCHREIBUNG

Das Login beantwortete jede Benutzernamen-Eingabe mit einer eindeutigen Meldung je Zustand (unbekannt / falsches Passwort / Konto deaktiviert). Damit lassen sich gültige Konten und ihr Aktiv-Status ohne Rate-Limit ermitteln.

AUSWIRKUNG

Aus „Benutzer und Passwort raten" wird „Passwort für ein bekanntes Konto raten" – deutlich erleichterte Credential-Angriffe.

SOFORTMASSNAHME

Einheitliche Fehlermeldung unabhängig vom Kontozustand; Rate-Limiting/Lockout aktivieren.

EMPFEHLUNG

Generische Login-Antworten, serverseitiges Throttling, Monitoring auf Enumerations-Muster.

D-05 ● NIEDRIG

Ausführliche .NET-Fehlerseiten mit internen Pfaden und IP-Adressen

SYSTEM	KATEGORIE	STATUS
Mehrere ASP.NET-Portale	Information Disclosure	Bestätigt (2 Hosts, zusammengeführt)

BESCHREIBUNG

Mehrere ASP.NET-Anwendungen liefen mit abgeschalteter Fehlerunterdrückung (customErrors off). Eine manipulierte Eingabe genügte, um eine vollständige Fehlerseite mit Stacktrace, internem Dateipfad, Framework-Version und teils interner IP-Adresse zu erhalten.

AUSWIRKUNG

Preisgabe interner Pfade, Versionen und Topologie – nützlich für die Auswahl passgenauer Folgeangriffe.

SOFORTMASSNAHME

Fehlerunterdrückung aktivieren (customErrors on / RemoteOnly) und generische Fehlerseiten ausliefern.

EMPFEHLUNG

Einheitliches, produktives Error-Handling fleetweit erzwingen; Diagnose-Endpunkte (Tracing) deaktivieren; interne Details nur serverseitig loggen.

E · Bürgerportale

5 FINDINGS · 4 MITTEL · 1 NIEDRIG

Getestet wurden bürgernahe Online-Dienste (Melde-/Anliegen-Portal, Kita-Portal, Bildungsportal) auf Zugriffskontrolle, Anti-Automatisierung und Eingabevalidierung.

E-01 MITTEL

IDOR: schreibender Fremdzugriff auf fremde Upload-Bündel

SYSTEM	KATEGORIE	STATUS
Melde-/Anliegen-Portal	IDOR / Broken Access Control	Bestätigt

BESCHREIBUNG

Der Datei-Upload prüfte nur das (öffentlich gelistete) Bündel-Token, aber keine Bindung an den Aufrufer. So ließen sich ohne Anmeldung fremde Meldungen um beliebige Dateien ergänzen. Zum Nachweis wurde eine eigene Marker-Datei in ein fremdes Bündel gelegt und anschließend wieder entfernt; Fremdinhalte blieben unberührt.

AUSWIRKUNG

Manipulation fremder Bürger-Meldungen (z. B. Unterschieben irreführender Inhalte), Speichermissbrauch. Lese-/Lösch-Pfade waren korrekt geschützt.

SOFORTMASSNAHME

Schreibzugriff an die Eigentümer-Session binden; unbekannte/fremde Token für Schreibzugriffe ablehnen.

EMPFEHLUNG

Object-Level-Authorization konsequent auf allen Pfaden; nicht erratbare, sitzungsgebundene Referenzen; Upload-Limits und MIME-Allowlist.

E-02 ● MITTEL

IDOR: Offenlegung von Melder-Kontaktdaten (Name & E-Mail)

SYSTEM	KATEGORIE	STATUS
Melde-/Anliegen-Portal	IDOR / PII Exposure	Bestätigt

BESCHREIBUNG

Eine Hilfs-Schnittstelle gab zu jeder veröffentlichten Meldung ohne Anmeldung den Namen und die private E-Mail-Adresse der meldenden Person sowie die zuständige Dienst-Mailadresse zurück – Daten, die die öffentliche Ansicht derselben Meldung bewusst verbirgt.

AUSWIRKUNG

Massenhafte Re-Identifikation meldender Bürger (Name + E-Mail) – Datenschutzvorfall und Grundlage für gezieltes Phishing.

SOFORTMASSNAHME

Die Schnittstelle gegen anonyme Aufrufe absichern (nur berechtigtes Personal) bzw. Kontaktdaten aus der Antwort entfernen.

EMPFEHLUNG

Zugriffskontrolle für jede Objekt-/Hilfsschnittstelle; personenbezogene Felder nur an berechtigte Rollen; Datenminimierung in API-Antworten.

E-03 ● MITTEL

Captcha-Umgehung über ein Entschlüsselungs-Orakel

SYSTEM	KATEGORIE	STATUS
Kita-Portal	Broken Anti-Automation	Bestätigt

BESCHREIBUNG

Das Captcha wird als verschlüsselter Wert ausgegeben. Derselbe Entschlüsseler, der an anderer Stelle genutzt wird, gab den entschlüsselten Klartext in einer Fehlermeldung zurück – also die Captcha-Lösung. Damit ließ sich die Registrierung ohne menschliche Interaktion durchlaufen.

AUSWIRKUNG

Automatisierte Massen-Registrierung/-Spam möglich; die Anti-Automatisierung ist wirkungslos (keine Token-Fälschung nachgewiesen).

SOFORTMASSNAHME

Die Fehlermeldung, die den Klartext preisgibt, abschalten; den Entschlüsseler nicht an nicht vertrauenswürdige Eingaben koppeln.

EMPFEHLUNG

Captcha-Lösung ausschließlich serverseitig prüfen und nie (auch nicht indirekt über Fehler) ausgeben; Rate-Limiting ergänzen.

E-04 ● MITTEL

Reflektiertes XSS in einem Kurssuche-Parameter

SYSTEM	KATEGORIE	STATUS
Bildungsportal	Cross-Site Scripting	Bestätigt

BESCHREIBUNG

Der Suchbegriff wurde ungefiltert in einen Inline-JavaScript-Kontext (Template-Literal) der Seite übernommen. Ein passendes Zeichen brach aus der Zeichenkette aus und brachte beliebiges JavaScript zur Ausführung (im Browser als Dialog nachgewiesen); eine Content-Security-Policy fehlte.

AUSWIRKUNG

Ausführung von Angreifer-Skript im Kontext des Portals über einen präparierten Link – z. B. Diebstahl von Session-Daten oder Phishing auf vertrauter Domain.

SOFORTMASSNAHME

Den Parameter im JS-Kontext korrekt kodieren/escapen; eine Content-Security-Policy setzen.

EMPFEHLUNG

Ausgabe-Kodierung je Kontext (HTML/JS/Attribut) konsequent anwenden; nutzergesteuerte Werte nicht in Inline-Skripte schreiben; strikte CSP ausrollen.

E-05 ● NIEDRIG

Erstellen von Meldungen ohne Captcha und ohne Rate-Limit

SYSTEM	KATEGORIE	STATUS
Melde-/Anliegen-Portal	Broken Anti-Automation	Bestätigt

BESCHREIBUNG

Während die Weboberfläche für neue Meldungen ein Captcha verlangt, nahm die zugehörige JSON-API vollständige Meldungen anonym und ohne Captcha oder Rate-Limit entgegen. Serverseitige Moderation verhinderte, dass Inhalte unkontrolliert öffentlich wurden.

AUSWIRKUNG

Automatisiertes Zumüllen der Melde-Warteschlange und Speichermissbrauch; dank Moderation keine direkte Veröffentlichung.

SOFORTMASSNAHME

Captcha-/Rate-Limit-Prüfung auch auf dem API-Pfad erzwingen (nicht nur im Frontend).

EMPFEHLUNG

Anti-Automatisierung serverseitig und kanalübergreifend durchsetzen; Upload-/Erstell-Limits pro Quelle.

ÜBERGREIFENDE EMPFEHLUNGEN

Vier Hebel mit der größten Wirkung

\ Angriffsfläche reduzieren. Programmierschnittstellen und Verwaltungs-/Admin-Oberflächen gehören nicht ungeschützt ins Internet – Netzsegmentierung, Allowlisting und VPN vor Erreichbarkeit.

\ Authentifizierung & Autorisierung serverseitig erzwingen. Jede Funktion, jedes Objekt und jede Schnittstelle prüft Identität und Rechte selbst (fail-closed) – nie allein über Pfad-Pattern, Cookies oder Client-Angaben.

\ Zugangsdaten härten. Keine Standard-/Initialpasswörter, keine Klartext-Passwörter, nur aktuelle salted Hash-Verfahren, Passwortrichtlinie und Rotation privilegierter Konten.

\ Eingaben parametrisieren, Ausgaben kodieren. Prepared Statements und Filter-Escaping gegen SQL-/LDAP-Injection; kontextabhängige Ausgabe-Kodierung und eine strikte CSP gegen XSS; generische Fehlerseiten.

Nächster Schritt

Empfohlen wird ein Nachtest (Re-Test) nach Umsetzung der Sofortmaßnahmen und der dauerhaften Behebung, um die Wirksamkeit zu bestätigen. Behebungsunterstützung gemeinsam mit Ihrer IT bieten wir optional an.

Sagorski — Penetrationstests

Tested before attacked. KI-gestützte, kosteneffiziente Penetrationstests auf eigener Infrastruktur in Europa. Dieser Beispielbericht ist anonymisiert und dient ausschließlich der Veranschaulichung unseres Vorgehens und unserer Berichtsqualität.

Vertragspartner

patchletter UG
(haftungsbeschränkt)
Meisterweg 16, 45896
Gelsenkirchen

Kontakt

anfrage@sagorski.org

Vertraulichkeit

Realberichte werden ausschließlich verschlüsselt und vertraulich an den Auftraggeber übergeben.