THOMAS SIEGBERT

OFFENSIVE SECURITY · SECURE DEVELOPMENT

Musterbericht

Web- und API-Penetrationstest

Fiktives B2B-Vertragsportal · vollständig synthetisches Szenario

ÖFFENTLICHE ARBEITSPROBE · SYNTHETISCH

Berichtsdatum	15. Juli 2026
Prüfzeitraum	15. bis 19. Juni 2026

siegbert-security.de

0 Dokumenthinweise

Einordnung, Nutzung und Grenzen dieser öffentlichen Arbeitsprobe

Dokumenttyp	Öffentlicher, vollständig synthetischer Musterbericht
Fiktiver Auftraggeber	Example Systems GmbH
Fiktiver Prüfgegenstand	B2B-Vertragsportal mit REST-API und drei Benutzerrollen
Prüfart	Authentifizierter Web- und API-Penetrationstest mit gezielter Angriffskettenanalyse
Testzeitraum	15. bis 19. Juni 2026
Berichtsdatum	15. Juli 2026
Klassifizierung	Öffentliche Arbeitsprobe - keine Kundendaten

Wichtiger Hinweis
Sämtliche Organisationen, Systeme, Personen, Hostnamen, IP-Adressen, Zugangsdaten, Kennzahlen, Screenshots und Befunde in diesem Dokument sind frei erfunden. Die verwendeten Domains enden auf .test; IP-Adressen stammen aus für Dokumentationszwecke reservierten Bereichen. Das Dokument ist kein Auszug aus einem Kundenprojekt und enthält keine Inhalte aus Prüfungsumgebungen Dritter.

Zweck der Arbeitsprobe

Der Bericht zeigt, wie technische Schwachstellen für Management, Entwicklung und Betrieb aufbereitet werden können: mit nachvollziehbarer Risikobegründung, belastbarer Evidenz, einer zusammenhängenden Angriffskette und einem priorisierten Maßnahmenplan. Werkzeugausgaben werden auf die entscheidungsrelevanten Ausschnitte begrenzt.

Rahmenbedingungen

Der fiktive Test wurde nicht destruktiv durchgeführt. Nachweise wurden jeweils nach einer kleinen synthetischen Stichprobe beendet. Denial-of-Service, Social Engineering, physische Prüfungen sowie Tests des externen Identity-Providers waren ausgeschlossen.

0.1 Inhaltsübersicht

Leselogik

Für eine schnelle Management-Entscheidung genügen die Seiten 4, 6, 7 und 15. Technische Teams finden die reproduzierbare Evidenz und die konkreten Behebungsansätze auf den Seiten 8 bis 14.

1 Management Summary

Verdichtete Entscheidungsgrundlage für Auftraggeber, Produktverantwortung und technische Leitung

KRITISCH

Gesamtbewertung

Durch die Verkettung der SSRF mit dem überprivilegierten Service-Token konnte ein reguläres Kundenkonto bis zur mandantenweiten Exportfunktion eskalieren. Die BOLA blieb ein eigenständiger hoher Befund. Der Nachweis endete nach fünf Datensätzen.

PRÜFUMFANG
3 Rollen
Web + REST-API

BEFUNDE
5
1 kritisch · 2 hoch
1 mittel · 1 niedrig

TESTAUFWAND
7 PT
inkl. Bericht

Wesentliche Auswirkung

Die zentrale Gefährdung betrifft die Vertraulichkeit und Mandantentrennung des Portals. Ein Angreifer mit einem gewöhnlichen Kundenkonto könnte interne Dienste ansprechen, ein zu weit berechtigtes Service-Token erlangen und damit Exporte über mehrere Mandanten anstoßen. In einer realen Umgebung wären Vertragsdokumente, Ansprechpartner, Abrechnungsdaten und interne Referenzen betroffen.

Entscheidung	Empfehlung	Zeitpunkt
URL-Import absichern	Funktion bis zur wirksamen Zielvalidierung deaktivieren oder auf freigegebene Quellen begrenzen.	Sofort
Service-Token rotieren	Token sperren, Logs prüfen und Berechtigungen auf den minimalen technischen Zweck reduzieren.	Sofort
Mandantenprüfung zentralisieren	Objektzugriffe serverseitig an Identität und Mandant binden; Negativtests ergänzen.	Vor Release
Retest durchführen	Korrekturen gezielt gegen die dokumentierte Angriffskette verifizieren.	Nach Fix

Management-Fazit: Auf Grundlage des geprüften Scopes erscheint eine vollständige Neuentwicklung zur Behebung der dokumentierten Befunde nicht erforderlich. Die kritische Angriffskette kann durch wenige, gezielte Architektur- und Berechtigungsmaßnahmen unterbrochen werden. Die Freigabe des nächsten Releases sollte jedoch erst nach Behebung und Retest erfolgen.

2 Scope und Vorgehensweise

In Scope	portal.example.test - Weboberfläche api.example.test - REST-API auth.example.test - Authentifizierungsintegration innerhalb der Anwendung
Testrollen	Kunde Alpha, Kunde Beta und Support-Benutzer mit bereitgestellten Testkonten
Testansatz	Authenticated Gray Box: funktionale Dokumentation und Testkonten vorhanden, kein Quellcodezugriff
Ausgeschlossen	Denial-of-Service, Social Engineering, physische Prüfungen, externer Identity-Provider, produktive Drittsysteme
Testdaten	Ausschließlich synthetische Mandanten, Verträge, Dokumente und Personen

Methodik

1	**Planung und Testregeln**	Scope, Rollen, Abbruchkriterien, Datenhandling und Kommunikationswege festlegen.
2	**Angriffsflächen erfassen**	Web-Routen, API-Endpunkte, Rollen, Objektbeziehungen, Uploads, Importe und Integrationen inventarisieren.
3	**Authentifizierung und Autorisierung**	Rollenwechsel, Mandantentrennung, Objektzugriffe, Session-Lifecycle und administrative Funktionen prüfen.
4	**Eingaben und Geschäftslogik**	Serverseitige Requests, Datei- und URL-Importe, Massenzuweisung, Workflow-Manipulation und Fehlerbehandlung testen.
5	**Angriffsketten validieren**	Einzelbefunde kontrolliert kombinieren, um die tatsächlich erreichbare Auswirkung zu bestimmen.
6	**Bereinigung und Bericht**	Testartefakte entfernen, Evidenz minimieren, Risiken begründen und Maßnahmen priorisieren.

Risikobewertung

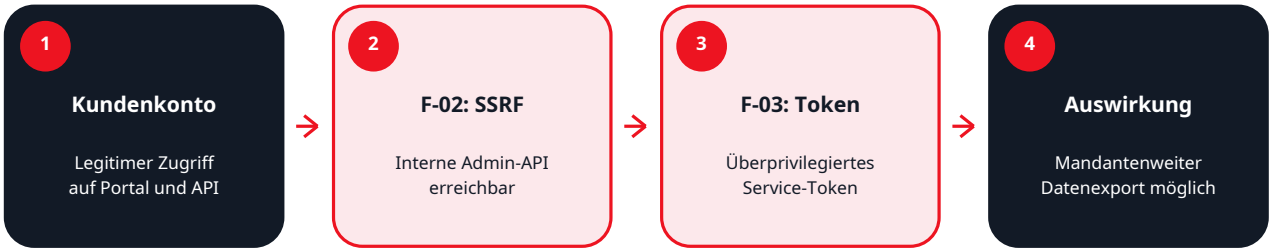
Die Einstufung verbindet Ausnutzbarkeit und Voraussetzungen mit Datenumfang, Zugriffstiefe und geschäftlicher Auswirkung.

Auswirkung / Wahrscheinlichkeit	Niedrig	Mittel	Hoch
Kritisch	**Hoch**	**Kritisch**	**Kritisch**
Hoch	**Mittel**	**Hoch**	**Kritisch**
Mittel	**Niedrig**	**Mittel**	**Hoch**
Niedrig	**Niedrig**	**Niedrig**	**Mittel**

Die geschäftliche Einstufung ist führend. CVSS kann ergänzend angegeben werden, ersetzt aber nicht die projektspezifische Begründung.

3 Angriffskette

Vom regulären Kundenkonto bis zur mandantenweiten Exportfunktion



Der Nachweis wurde nach einer kleinen synthetischen Stichprobe beendet; kein vollständiger Datenexport wurde durchgeführt.

Schritt	Technischer Nachweis	Warum der Schritt relevant ist
1	Anmeldung als Kunde Alpha mit regulärem Testkonto.	Kein administratives oder privilegiertes Ausgangskonto erforderlich.
2	URL-Import ruft eine interne Admin-Schnittstelle ab und zeigt die Antwort im Job-Preview.	Überschreitet die vorgesehene Netzwerkgrenze der Anwendung.
3	Interne Antwort enthält ein Service-Token mit zu breiten Exportrechten.	Macht eine technische Identität zur Privilegieneskalation nutzbar.
4	Token autorisiert eine mandantenweite Exportabfrage; Nachweis nach fünf Datensätzen beendet.	Belegt die geschäftliche Gesamtauswirkung der Verkettung.

Eigenständiger hoher Befund

F-01 (BOLA) ermöglichte unabhängig davon den Zugriff auf Dokumente und Metadaten anderer Mandanten. Die Schwachstelle war keine Voraussetzung für die SSRF und wird separat behoben und retestet.

Gesamtrisiko der Kette

F-02 ist wegen der nachgewiesenen Verkettung kritisch eingestuft; isoliert belegt die SSRF den Zugriff auf interne Dienste. Das offengelegte Token ermöglicht anschließend den mandantenweiten Export.

4 Befundübersicht

ID	Befund	Risiko	Kernauswirkung	Priorität
F-01	Fehlende objektbezogene Autorisierung Vertragsdokumente anderer Mandanten abrufbar	Hoch	Verletzung der Mandantentrennung und Vertraulichkeit	Vor Release
F-02	Server-Side Request Forgery URL-Import erreicht interne Dienste	Kritisch	Zugriff auf interne Admin-Schnittstellen und Konfiguration	Sofort
F-03	Überprivilegiertes Service-Token Keine Mandantenbindung, zu breite Scopes	Hoch	Mandantenweiter Export und administrative API-Nutzung	Sofort
F-04	Unvollständige Sitzungswiderrufung Access Token bleibt nach Passwortwechsel gültig	Mittel	Verlorene oder gestohlene Sitzungen bleiben nutzbar	Nächster Zyklus
F-05	Unvollständige Browser-Härtung CSP fehlt, Server-Header zu detailliert	Niedrig	Erhöhte Angriffsfläche und Informationsgewinn	Härtung

Interpretation der Prioritäten

KRITISCH	Unmittelbare Gefährdung zentraler Daten oder administrativer Funktionen. Vor Freigabe beheben.
HOCH	Erhebliche Auswirkung mit realistischen Voraussetzungen. Kurzfristig beheben und retesten.
MITTEL	Relevantes Risiko, dessen Ausnutzung zusätzliche Bedingungen erfordert oder begrenzter wirkt.
NIEDRIG	Härtungs- oder Informationsgewinn ohne unmittelbar nachgewiesene wesentliche Auswirkung.

Die nachfolgenden Detailseiten zeigen exemplarisch, wie ein Befund technisch reproduzierbar, für das Management verständlich und für die Umsetzung konkret dokumentiert wird.

5 F-01 - Fehlende objektbezogene Autorisierung

HOCH

Betroffener Bereich: Vertragsdokument-API
Voraussetzung: Beliebiges angemeldetes Kundenkonto
Auswirkung: Zugriff auf Dokumente und Metadaten anderer Mandanten

Zusammenfassung

Die API prüfte, ob ein Benutzer angemeldet war, band das angeforderte Dokument jedoch nicht an den Mandanten der Sitzung. Durch Austausch der Dokument-ID konnte Kunde Alpha ein Vertragsdokument des fiktiven Kunden Beta abrufen. Die Benutzeroberfläche zeigte ausschließlich eigene Dokumente; die serverseitige API setzte diese Einschränkung nicht durch.

Bereinigte Beispielanfrage

```
GET /api/v1/contracts/7cfa9b4a/documents/2e91d170 HTTP/1.1
Host: api.example.test
Authorization: Bearer <TOKEN_TENANT_ALPHA>
Accept: application/json
```

Beobachtete Antwort

```
HTTP/1.1 200 OK
Content-Type: application/json

{
  "documentId": "2e91d170",
  "tenantId": "tenant-beta",
  "fileName": "Rahmenvertrag-2026.pdf",
  "downloadUrl": "/api/v1/documents/2e91d170/content"
}
```

Bereinigte Inhaltsanfrage

```
GET /api/v1/documents/2e91d170/content HTTP/1.1
Authorization: Bearer <TOKEN_TENANT_ALPHA>
```

Beobachtete Inhaltsantwort

```
HTTP/1.1 200 OK
Content-Type: application/pdf
Content-Length: 48317
X-Synthetic-Document-SHA256: <REDACTED>
```

Erwartetes Verhalten

Die API muss für ein objektfremdes Dokument und dessen Inhalt unabhängig von der Kenntnis der ID mit 404 Not Found oder 403 Forbidden antworten. Die Entscheidung darf nicht allein auf UI-Filtern oder schwer erratbaren Identifikatoren beruhen.

Reproduzierbarkeit

Der Zugriff auf Metadaten und Dokumentinhalt war mit zwei bereitgestellten Kundenkonten reproduzierbar. Insgesamt wurden drei synthetische Dokumentobjekte geprüft. Weitere Zugriffe wurden nach Bestätigung des systematischen Fehlers beendet.

5.1 F-01 - Bewertung und Behebung

Risikobegründung

Ausnutzbarkeit	Auswirkung	Gesamt
Niedrige Komplexität; gültiges Kundenkonto genügt. Objekt-IDs waren in regulären API-Antworten sichtbar.	Mandantenübergreifender Zugriff auf Vertragsdokumente und Metadaten. Potenziell erhebliche Vertraulichkeits- und Datenschutzfolgen.	Hoch

Wahrscheinliche Ursache

Das Dokument wurde zunächst ausschließlich über seine globale ID geladen. Die nachgelagerte Berechtigungsprüfung kontrollierte lediglich die Rolle „customer“, nicht aber die Zuordnung des Dokuments zum Mandanten des angemeldeten Benutzers.

Empfohlene Behebung

1. Objekt und Mandant in derselben serverseitigen Abfrage binden, beispielsweise `findByIdAndTenant(documentId, principal.tenantId)`.

2. Berechtigungen standardmäßig verweigern und sämtliche API-Endpunkte mit Objektbezug in einer zentralen Autorisierungsschicht absichern.

3. Opaque IDs oder UUIDs nur als zusätzliche Erschwernis betrachten; sie ersetzen keine Autorisierung.

4. Automatisierte Negativtests für fremde Mandanten, fremde Rollen und direkte Objekt-URLs ergänzen.

5. Wiederholte objektfremde Zugriffe protokollieren und als mögliches Enumeration-Signal überwachen.

Beispiel für eine sichere Abfrage

```
document = repository.find_by_id_and_tenant(
    document_id=request.document_id,
    tenant_id=principal.tenant_id
)
if document is None:
    return 404
```

Retest-Kriterium

Alle dokumentbezogenen Endpunkte liefern für objekt- oder mandantenfremde IDs konsistent 404/403. Negativtests decken mindestens Kunden-, Support- und Administratorrollen ab.

6 F-02 - Server-Side Request Forgery im URL-Import

KRITISCH

Betroffener Bereich: Dokumentimport aus URL
Voraussetzung: Angemeldetes Kundenkonto
Auswirkung: Zugriff auf interne HTTP-Dienste und deren Antworten

Zusammenfassung

Die Funktion „Dokument aus URL importieren“ akzeptierte beliebige HTTP- und HTTPS-Ziele. Der Backend-Dienst führte die Anfrage aus seinem internen Netzsegment aus und übernahm den Antworttext in eine Vorschau des Importjobs. Eine Prüfung auf interne, lokale oder reservierte Adressbereiche fand nicht statt.

Bereinigte Beispielanfrage

```
POST /api/v1/imports HTTP/1.1
Host: api.example.test
Authorization: Bearer <TOKEN_TENANT_ALPHA>
Content-Type: application/json

{
  "sourceUrl": "http://admin-api.internal.test/v1/runtime"
}
```

Beobachtete Antwort

```
HTTP/1.1 201 Created
Content-Type: application/json

{
  "jobId": "imp-8341",
  "status": "preview-ready",
  "preview": "{ \"service\": \"export-worker\", \"token\": \"svc_REDACTED_SYNTHETIC\" }"
}
```

Nachweisgrenze

Es wurden ausschließlich eigens für das Szenario vorgesehene interne Testendpunkte aufgerufen. Portscans, Cloud-Metadaten-Endpunkte und produktive Drittsysteme waren ausgeschlossen. Nach Sichtbarkeit der synthetischen Konfiguration wurde der Test beendet.

Geschäftliche Relevanz

Die Schwachstelle durchbricht die vorgesehene Netzwerksegmentierung. Interne Verwaltungs-, Monitoring- oder Metadaten-Schnittstellen können Informationen und Zugangsdaten liefern, die extern nicht erreichbar sein sollten.

6.1 F-02 - Bewertung und Behebung

Risikobegründung

Ausnutzbarkeit	Auswirkung	Gesamt
Niedrige Komplexität; die Funktion war für reguläre Kunden vorgesehen. Eine einfache URL genügte.	Interne Dienste und Konfigurationen waren erreichbar. Der Befund ermöglichte den Übergang zu F-03 und damit eine mandantenweite Exportberechtigung.	Kritisch

Wahrscheinliche Ursache

Die Implementierung validierte nur das URL-Schema. DNS-Auflösung, Ziel-IP, Weiterleitungen und Netzbereich wurden nicht geprüft. Der Importdienst verfügte außerdem über weitreichenden ausgehenden Netzwerkzugriff und gab interne Antwortinhalte an den Benutzer zurück.

Empfohlene Behebung

1. Wenn fachlich möglich, nur explizit freigegebene Quell-Domains zulassen. Eine Positivliste ist einer Blockliste vorzuziehen.

2. Nach jeder DNS-Auflösung sämtliche Zieladressen prüfen und Loopback-, Link-Local-, private, reservierte sowie interne Netze ablehnen.

3. HTTP-Weiterleitungen deaktivieren oder jedes Weiterleitungsziel erneut vollständig validieren.

4. Ausgehende Requests über einen dedizierten Egress-Proxy mit Netzwerkregeln, Größenlimit, Zeitlimit und erlaubten Content-Types führen.

5. Antwortinhalte interner Requests niemals ungefiltert an den Benutzer spiegeln. Importergebnis und Diagnoseausgabe strikt trennen.

6. Auffällige Ziele, wiederholte Fehler und Zugriffe auf interne Namensräume zentral protokollieren.

Retest-Kriterium

Interne Hostnamen und alle reservierten Adressbereiche werden vor und nach Weiterleitungen zuverlässig blockiert. Zulässige externe Dokumentquellen funktionieren weiterhin. Antwortinhalte interner Ziele sind weder direkt noch über Fehlermeldungen sichtbar.

Sofortmaßnahme
Bis zur vollständigen Korrektur sollte der URL-Import deaktiviert oder auf wenige kontrollierte Quellen begrenzt werden. Parallel sind die Zugriffsprotokolle des Importdienstes auf interne Zieladressen zu prüfen.

7 F-03 - Überprivilegiertes Service-Token

HOCH

Betroffener Bereich: Extern erreichbarer Export-Endpunkt

Voraussetzung: Kenntnis des Tokens, im Szenario über F-02 erlangt

Auswirkung: Mandantenübergreifende Export- und Leserechte

Zusammenfassung

Das für einen Hintergrundprozess vorgesehene Service-Token war 24 Stunden gültig, enthielt keine Mandantenbindung und besaß zusätzlich zum vorgesehenen Job-Statuszugriff den Scope admin:exports. Ein extern erreichbarer Export-Endpunkt akzeptierte das Token, ohne die aufrufende Workload oder den fachlichen Zweck zu prüfen.

Bereinigte Beispielanfrage

```
GET /v1/admin/exports?tenant=* HTTP/1.1
Host: api.example.test
Authorization: Bearer svc_REDACTED_SYNTHETIC
Accept: application/json
```

Beobachtete Antwort

```
HTTP/1.1 200 OK
Content-Type: application/json

{
  "availableRecords": 1842,
  "availableDocuments": 312,
  "sample": [
    {"tenantId":"tenant-alpha","contractId":"c-1001"},
    {"tenantId":"tenant-beta","contractId":"c-2007"}
  ]
}
```

Nachweisgrenze

Es wurden fünf synthetische Datensätze abgerufen, um die mandantenübergreifende Wirkung zu belegen. Ein vollständiger Export wurde nicht gestartet. Das Token wurde nach Abschluss des Szenarios als kompromittiert markiert und aus der Testumgebung entfernt.

Zusammenhang mit F-02

Der Tokenbefund ist unabhängig relevant; das Token wurde jedoch erst durch F-02 offengelegt. Der extern erreichbare Export-Endpunkt akzeptierte es anschließend direkt als Bearer-Token. Die SSRF war damit nicht für die Erreichbarkeit des Exports, aber für die Erlangung des Tokens erforderlich.

7.1 F-03 - Bewertung und Behebung

Risikobegründung

Ausnutzbarkeit	Auswirkung	Gesamt
Tokenkenntnis erforderlich; im geprüften Szenario jedoch über F-02 erreichbar. Keine zusätzliche Workload- oder Netzwerkbindung.	Mandantenweite Lese- und Exportrechte, lange Gültigkeit und fehlende Zweckbindung. Potenziell vollständige Offenlegung zentraler Geschäftsdaten.	Hoch

Empfohlene Behebung

1. Bestehendes Token sofort widerrufen und alle hiervon abhängigen Geheimnisse kontrolliert rotieren.

2. Pro Workload eine eigene technische Identität mit minimalen Scopes und eindeutigem Audience-Claim verwenden.

3. Mandantenkontext oder konkreten Job-Kontext serverseitig erzwingen; Wildcards wie `tenant=*` für technische Standardkonten verbieten.

4. Kurzlebige Workload-Identitäten oder mTLS-gebundene Tokens anstelle langlebiger statischer Bearer-Tokens einsetzen.

5. Exportfunktionen zusätzlich durch Netzwerksegmentierung, Vier-Augen-Freigaben für Massendaten und revisionssichere Audit-Events absichern.

6. Tokennutzung nach Quelle, Scope, Datenmenge und Uhrzeit überwachen; ungewöhnliche mandantenweite Zugriffe alarmieren.

Retest-Kriterium

Ein Token des Import- oder Worker-Dienstes kann ausschließlich die vorgesehenen Job-Endpunkte und den zugewiesenen Mandantenkontext nutzen. Administrative Exporte erfordern eine separate Identität, zusätzlichen Freigabekontext und erzeugen ein vollständiges Audit-Event.

Hinweis für die Architektur
„Intern erreichbar“ ist keine Berechtigung. Jeder Dienst muss Anrufer, Zweck, Ressource und Mandantenkontext eigenständig verifizieren. Netzwerkgrenzen reduzieren Angriffsflächen, ersetzen aber keine Autorisierung.

8 Weitere Befunde

MITTEL

F-04 - Access Token bleibt nach Passwortwechsel gültig
Ein vor dem Passwortwechsel ausgestelltes Access Token konnte 35 Minuten später weiterhin auf geschützte API-Endpunkte zugreifen. Refresh Tokens wurden widerrufen; bestehende Access Tokens mit acht Stunden Laufzeit blieben jedoch gültig.

Auswirkung: Bei einem gestohlenen Token beendet ein Passwortwechsel den Zugriff nicht zuverlässig. Das Risiko ist insbesondere bei gemeinsam genutzten oder verlorenen Endgeräten relevant.

Empfehlung: Access Tokens deutlich kurzlebiger ausstellen, eine serverseitige Sitzungs- oder Token-Version prüfen und bei Passwortwechsel, MFA-Reset sowie manueller Abmeldung alle aktiven Sitzungen widerrufen. Benutzer sollten aktive Geräte und Sitzungen einsehen und beenden können.

NIEDRIG

F-05 - Unvollständige Browser- und Informationshärtung
Die Webanwendung setzte keine Content-Security-Policy und gab detaillierte Server- und Framework-Versionen in Antwort-Headern aus. Eine direkte Ausnutzung wurde im Test nicht nachgewiesen.

Auswirkung: Fehlende Browser-Härtung kann die Auswirkungen zukünftiger clientseitiger Fehler vergrößern. Detaillierte Versionsangaben erleichtern die zielgerichtete Schwachstellensuche.

Empfehlung: Eine an die Anwendung angepasste CSP zunächst im Report-Only-Modus einführen, Inline-Skripte schrittweise reduzieren und unnötige Versionsheader entfernen. Bestehende HSTS-, Secure-, HttpOnly- und SameSite-Einstellungen beibehalten.

Verifikationsgrenzen

Technisch verifiziert: Im geprüften Umfang wurden keine SQL-Injection und keine direkte Umgehung der MFA für die bereitgestellte Support-Rolle festgestellt. Die Passwortspeicherung war nicht Bestandteil der technischen Verifikation. Aussagen zu Administrationsrollen wurden nicht unabhängig technisch verifiziert. Der externe Identity-Provider war ausgeschlossen und nicht Bestandteil der technischen Verifikation. Diese Einordnung gilt ausschließlich für den vereinbarten Testzeitraum und Scope.

9 Priorisierter Maßnahmenplan und Retest

Zeitraum	Maßnahme	Verantwortlich	Erfolgskriterium
0-24 Stunden	URL-Import deaktivieren oder streng begrenzen.	Product Owner / Betrieb	Externer URL-Abruf bis zur Zielvalidierung unterbrochen.
0-24 Stunden	Service-Token widerrufen; Nutzung und Logs prüfen.	IAM / Plattformbetrieb	Token ungültig; missbräuchliche Nutzung geprüft.
1-5 Arbeitstage	SSRF-Schutz mit Zielvalidierung und Egress-Regeln implementieren.	Entwicklung / Plattform	F-02/F-03-Kette nicht mehr reproduzierbar.
1-5 Arbeitstage	Objektberechtigung zentral korrigieren.	API-Entwicklung	F-01 separat behoben; Fremdzugriffe liefern 404/403.
Nach Behebung	Retest koordinieren und Ergebnisse dokumentieren.	Security Owner	Befunde verifiziert; Freigabeentscheidung dokumentiert.
2-4 Wochen	Negativtests ergänzen; Session-Widerruf, Egress-Proxy und Audit-Events verbessern.	Entwicklung / Plattform / SecOps	Kontrollen automatisiert und auswertbar.
4-8 Wochen	Threat Model aktualisieren; Identitäten inventarisieren; CSP-/Header-Härtung ausrollen.	Architektur / Product Security	Härtung in Entwicklungs- und Betriebsprozesse integriert.

Vorgeschlagener Retest

Umfang	F-01 bis F-04 sowie die dokumentierte Angriffskette; F-05 als Konfigurationsprüfung
Voraussetzungen	Bereitgestellte Testkonten, freigegebene Testumgebung und Beschreibung der Änderungen
Verantwortlich	Security Owner koordiniert; Maßnahmenverantwortliche liefern Fixes und Evidenz
Aufwand	Voraussichtlich 1 bis 2 Personentage, abhängig von der Anzahl geänderter Endpunkte
Ergebnis	Kurzer Retest-Vermerk je Befund: behoben, teilweise behoben oder weiterhin reproduzierbar

Freigabeempfehlung

Der nächste produktive Release sollte erst erfolgen, wenn F-02 und F-03 vollständig behoben und F-01 für sämtliche objektbezogenen Endpunkte verifiziert wurde. Ein erfolgreicher Retest ist die belastbarste Grundlage für die Freigabeentscheidung.

10 Positive Beobachtungen und Fazit

Beobachtung	Nutzen
MFA für die bereitgestellte Support-Rolle (technisch verifiziert)	Reduziert das Risiko einer reinen Passwortkompromittierung.
MFA für Administrationsrollen und speicherintensives Passwort-Hash-Verfahren (dokumentationsbasiert)	Nicht technisch unabhängig verifiziert; ergänzende Angabe aus der funktionalen Dokumentation.
HSTS sowie Secure-, HttpOnly- und SameSite-Cookies (technisch verifiziert)	Stärkt Transport- und Sitzungsabsicherung im Browser.
Rate Limits und generische Fehlermeldungen am Login (beobachtet)	Begrenzt automatisierte Anmeldeversuche und Benutzer-Enumeration.
Getrennte Testdaten und klar definierte Abbruchkriterien (beobachtet)	Ermöglicht aussagekräftige Nachweise ohne unnötige operative Risiken.

Fazit

Die fiktive Anwendung verfügte über mehrere wirksame Basiskontrollen. Die kritische Angriffskette entstand am Übergang zwischen serverseitigem URL-Abruf und technischen Identitäten. Die SSRF legte ein überprivilegiertes Service-Token offen, das mandantenweite Exporte erlaubte. Zusätzlich verletzte die BOLA als eigenständiger hoher Befund die Mandantentrennung.

Die Maßnahmen sind gezielt und umsetzungsnah: Netzwerkziele des Importdienstes kontrollieren und technische Tokens nach dem Minimalprinzip zuschneiden, um die kritische Kette zu unterbrechen. Objektberechtigungen werden davon unabhängig zentral erzwungen. Negativtests sichern beide Korrekturen ab.

Thomas Siegbert
IT-Security Consultant
Offensive Security · Secure Development

siegbert-security.de

Hinweis zur Arbeitsprobe
Ein realer Kundenbericht wird an Scope, Zielgruppe, regulatorische Anforderungen und technische Umgebung angepasst. Sensible Evidenz wird getrennt behandelt; Management- und Technikteil können je nach Bedarf separat bereitgestellt werden.