

Sicherheitsanforderung

Microsoft IIS

Deutsche Telekom Gruppe

Version	8.0
Datum	01.07.2025
Status	Freigegeben

Impressum

Herausgeber
Deutsche Telekom AG
Vorstandsbereich Technology & Innovation
Chief Security Officer

Reuterstrasse 65, 53315 Bonn
Deutschland

Dateiname	Dokumentennummer 3.32	Dokumententyp Sicherheitsanforderung
Version 8.0	Stand 01.07.2025	Status Freigegeben
Fachlicher Ansprechpartner Telekom Security psa.telekom.de	Gültigkeitsdauer 01.07.2025 - 30.06.2030	Freigegeben von Stefan Pütz, Leiter SEC-T-TST

Zusammenfassung

Diese Sicherheitsanforderung basiert auf den im Konzern gültigen Sicherheitsrichtlinien und definiert die Anforderungen an eine sichere Implementierung von Microsoft IIS Webservern. Die hier beschriebenen Anforderungen müssen erfüllt werden, um sicherzustellen, dass ein Microsoft IIS Webserver nicht ohne weiteres von sachkundigen Angreifern missbraucht werden kann.

Copyright © 2025 by Deutsche Telekom AG.

Alle Rechte, auch die des auszugsweisen Nachdrucks, der fotomechanischen Wiedergabe (einschließlich Mikrokopie) sowie der Auswertung durch Datenbanken oder ähnliche Einrichtungen, vorbehalten.

Inhaltsverzeichnis

1.	Einleitung	4
2.	Anforderungen an die Webserver-Software	5
3.	Konfigurationsanforderungen	9
4.	HTTPS-Anforderungen	18
5.	Protokollierung	25

1. Einleitung

Dieses Dokument wurde auf Basis der Vorgaben aus den im Konzern gültigen Sicherheitsrichtlinien erstellt. Die Sicherheitsanforderung dient u.a. als Grundlage zur Freigabe im PSA-Prozess. Sie dient darüber hinaus als Umsetzungsempfehlung für Einheiten, die nicht am PSA-Verfahren teilnehmen. Diese Anforderungen müssen bereits während der Planungs- und Entscheidungsprozesse berücksichtigt werden. Bei der Umsetzung der Sicherheitsanforderung ist das jeweils vorrangige nationale, internationale und supranationale Recht zu beachten.

2. Anforderungen an die Webserver-Software

Req 1	Software und Hardware des Systems müssen von einem Sicherheitsschwachstellen-Support seitens des Herstellers abgedeckt werden.
-------	--

In einem System dürfen nur Software- und Hardware-Produkte eingesetzt werden, für die ein Sicherheitsschwachstellen-Support durch den Hersteller besteht.

Ein solcher Support muss umfassen, dass der Hersteller

- das Produkt kontinuierlich auf eine Betroffenheit von Sicherheitsschwachstellen beobachtet und analysiert,
- über Art, Schweregrad und Ausnutzbarkeit von im Produkt entdeckten Schwachstellen umgehend informiert
- zeitnah Produktaktualisierungen oder wirksame Workarounds zur Behebung der Schwachstellen bereitstellt.

Der Sicherheitsschwachstellen-Support muss für die gesamte Dauer bestehen, in der das betroffene Produkt im Einsatz verbleibt.

Support-Phasen mit eingeschränktem Leistungsumfang

Viele Hersteller bieten für ihre Produkte als Option einen zeitlich ausgedehnten Support an, welcher über die für den allgemeinen Markt vorgesehene Support-Phase hinausgeht, aber oftmals mit Einschränkungen verbunden ist. Einige Hersteller definieren ihren Support grundlegend in Abstufungen, die Einschränkungen schon während der letzten Phase vor dem absoluten Endtermin des regulären Supports enthalten können.

Wird ein Produkt innerhalb von Support-Phasen eingesetzt, die Einschränkungen unterliegen, muss explizit sichergestellt werden, dass diese Einschränkungen die Verfügbarkeit des Sicherheitsschwachstellen-Support nicht beeinträchtigen.

Open Source Software und Hardware

Open Source Produkte werden häufig von freien Organisationen oder Communities entwickelt; entsprechend ist ein vertraglich vereinbarter Sicherheitsschwachstellen-Support unter Umständen nicht verfügbar. Grundsätzlich muss auch hier sichergestellt werden, dass von der Organisation/Community (oder einem offiziell durch diese eingesetzten Dritten) für das betroffene Produkt ein umfassendes Sicherheitsschwachstellen-Management betrieben wird, welches die oben genannten Kriterien erfüllt und als zuverlässig etabliert gilt.

Motivation: Hardware- und Software-Produkte, für die kein vollumfassender Sicherheitsschwachstellen-Support durch den Hersteller besteht, stellen ein Risiko dar. Dies bedeutet, dass ein Produkt nicht ausreichend auf eine Betroffenheit durch weiterentwickelte Angriffsformen oder neu entdeckte Schwächen technischer Umsetzungen geprüft wird. Analog werden bei vorhandenen Sicherheitsschwachstellen in einem Produkt keine Nachbesserungen (z. B. Updates, Patches) bereitgestellt. Hieraus resultiert ein System, dessen Schwachstellen nicht behoben werden können, so dass sie für einen Angreifer ausnutzbar bleiben, um das System zu kompromittieren oder es negativ zu beeinträchtigen.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigter Zugang zum System
- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Unberechtigte Modifikation von Daten
- Unberechtigte Nutzung von Diensten oder Ressourcen
- Störung der Verfügbarkeit
- Abstreiten von durchgeführten Aktionen
- Unbemerkt durchführbare Angriffe
- Angriffe motiviert und erleichtert durch Informationspreisgabe oder erkennbare Sicherheitsschwachstellen

ID: 3.01-1/9.0

Req 2	Eingesetzte Software muss aus vertrauenswürdigen Quellen bezogen und auf Integrität geprüft werden.
-------	---

Die auf dem System eingesetzte Software muss aus vertrauenswürdigen Quellen bezogen und vor der Installation auf ihre Integrität geprüft werden.

Diese Anforderung bezieht sich auf sämtliche Arten von Software:

- Firmware sowie Microcode für Hardware-Komponenten

- Betriebssysteme
- Software-Bibliotheken
- Applikationen und Anwendungssoftware
- vorintegrierte Anwendungslösungen, wie z. B. Software Appliances oder Container

sowie weitere Software, die ggf. zum Einsatz kommt.

Vertrauenswürdige Quellen

Als vertrauenswürdige Bezugsquellen gelten allgemein:

- die offiziellen Vertriebs- und Bereitstellungs Kanäle des Herstellers
- dritte Distributoren, sofern sie durch den Hersteller autorisiert und legitimer Bestandteil von dessen Bereitstellungs Kanälen sind
- Internet-Downloads, sofern diese von offiziellen Bereitstellungs-Servern des Herstellers oder autorisierter Distributoren erfolgen
 - (1) Bietet der Bereitstellungs-Server verschiedene Formen von Downloads an, müssen durch Verschlüsselung bzw. kryptografische Signaturen geschützte Downloads jenen ohne einen solchen Schutz vorgezogen werden.
 - (2) Sichert der Bereitstellungs-Server die Transport-Schicht mittels kryptografischer Protokolle (z. B. HTTPS, sftp), müssen die zugehörig präsentierten Server-Zertifikate oder Server-Keys/Fingerprints bei jedem Download validiert werden, um die Identität des Bereitstellungs-Servers zu bestätigen; schlägt die Validierung fehl, ist der Download abubrechen und der Bereitstellungs-Server als nicht vertrauenswürdige Quelle zu bewerten.

Integritätsprüfung

Die Integritätsprüfung soll sicherstellen, dass die erhaltene Software frei von Manipulationen und Schadsoftware-Befall ist. Sofern vorhanden, müssen zur Prüfung die vom Hersteller implementierten Mechanismen verwendet werden.

Valide Mechanismen sind:

- physikalische Siegel bzw. fest aufgebrachte Echtheitszertifikate (bei Bereitstellung der Software auf physikalischen Datenträgern)
- Abgleich von kryptografischen Hash Werten (z. B. SHA256, SHA512) der erhaltenen Software gegen Soll-Werte, die der Hersteller separat benennt
- Prüfung von kryptografischen Signaturen (z. B. GPG, Zertifikate), mit denen der Hersteller seine Software versieht

Zusätzlich ist eine Überprüfung der Software mittels eines Anti-Virus- bzw. Anti-Malware Scanners empfohlen (falls der Hersteller keinen der zuvor erwähnten Integritätsschutz-Mechanismen für seine Software implementiert hat, ist diese Überprüfung verpflichtend).

Erweiterte Integritätsprüfung bei Bezug von Software aus öffentlichen Registries

Öffentliche Registries ermöglichen es Entwicklern, beliebige eigene Software-Projekte zur Nutzung bereitzustellen. Die Bandbreite umfasst sowohl Projekte von namhaften Anbietern mit kontrollierten Entwicklungsprozessen als auch von kleineren Anbietern oder Amateur-Entwicklern.

Beispiele für solche Registries sind:

- Code Registries (z. B. GitHub, Bitbucket, SourceForge, Python Package Index)
- Container Registries (z. B. Docker Hub)

Software aus öffentlichen Registries muss vor dem Einsatz einer erweiterten Integritätsprüfung unterzogen werden. Zusätzlich zu den im vorherigen Abschnitt beschriebenen Bestandteilen der Integritätsprüfung soll die erweiterte Prüfung explizit sicherstellen, dass die Software ihre Funktion tatsächlich wie beschrieben erfüllt, keine inhärenten Sicherheits-Risiken wie etwa gezielt implementierte Malware-Funktionen enthält und nicht von bekannten Sicherheits-Schwachstellen betroffen ist. Sofern die Software direkte Abhängigkeiten zu dritten Software-Projekten aufweist (sog. "Dependencies", sehr typisch in Open Source Software), die zum Einsatz der Software ebenfalls bezogen und installiert werden müssen, sind diese in die erweiterte Integritätsprüfung mit einzubeziehen.

Methoden für eine erweiterte Integritätsprüfung können beispielsweise sein:

- strikte Validierung der Projekt-/Paket-Namen (Vermeidung der Verwechslung mit maliziösen, gezielt nachgeahmten Software-Projekten)
- dynamische Code Analyse / strukturierte Funktionsprüfung in einer Test-Umgebung
- statische Code Analyse mittels eines Linters (z. B. Splint, JSLint, Pylint)
- Untersuchung mittels eines Sicherheitsschwachstellen-Scanners (z. B. Qualys, Nessus)
- Untersuchung mittels eines Container Security Scanners (z. B. JFrog Xray, Harbor, Clair, Docker Scan)
- Untersuchung mittels eines SCA (Software Composition Analysis) Tools bzw. Dependency-Scanners (z. B. OWASP Dependency Check, Snyk)

Die Prüfungsmethoden sind passend zur exakten Form der Software-Bereitstellung (Quellcode, Binaries/Artefakte, Container) auszuwählen und angemessen zu kombinieren.

Motivation: Softwarelieferketten enthalten diverse Angriffsvektoren. Ein Angreifer kann an verschiedenen Stellen ansetzen, um Software zu manipulieren oder gezielt eigene Routinen einzubringen und die Zielumgebung, in der die Software anschließend zum Einsatz gebracht wird, zu schädigen oder unter seine Kontrolle zu bringen. Der Angriff kann auf dem Transport- bzw. Übertragungsweg geschehen oder auch auf die Bereitstellungsquelle selbst ausgeübt werden. Entsprechend erleichtert wird ein Angriff, falls Software nicht über offizielle und kontrollierte Quellen bezogen wird oder eine Integritätsprüfung ausgelassen wird.

Besonderes Risiko zeigt sich für Software, die aus öffentlichen Registries bezogen wird, da diese jedermann für die Bereitstellung von Software-Projekten offenstehen. Perfide Angriffsmethoden sind inzwischen bekannt, in denen der Angreifer zunächst eine Zeitlang sicherheitstechnisch völlig unauffällige, funktionale Software bereitstellt und sobald diese sich etabliert und eine gewisse Verbreitung gefunden hat, in zukünftigen Versionen gezielt versteckten Schadcode integriert. Andere Methoden setzen auf ähnlich klingende Projektnamen zu weit verbreiteten Bestandsprojekten oder überstimmende Versionsnummern, um manipulierte Software in darauf aufbauende Lösungen zu injizieren.

Umsetzungsbeispiel: Bezug der Software über die offiziellen Bereitstellungskanäle des Herstellers. Nach Erhalt der Software umgehend Prüfung auf Integrität mittels kryptografischer Checksummen, wie vom Hersteller bereitgestellt, sowie Freiheit von bekannter Schadsoftware mittels Anti-Malware- / Anti-Virus-Scanner. Ablage der geprüften Software auf einer internen, geschützten Dateiablage und weitere Nutzung (z. B. Rollout auf die Zielsysteme) nur noch von dort.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigte Modifikation von Daten
- Unbemerkt durchführbare Angriffe
- Angriffe motiviert und erleichtert durch Informationspreisgabe oder erkennbare Sicherheitsschwachstellen

ID: 3.01-2/9.0

Req 3	Bekannte Schwachstellen in der Software oder Hardware des Systems müssen behoben oder gegen Missbrauch abgesichert werden.
-------	--

Bekannte Schwachstellen in Software- und Hardware Komponenten müssen mittels Installation von verfügbaren Systemaktualisierungen des Herstellers (z. B. Patches, Updates/Upgrades) behoben werden. Alternativ ist der Einsatz von Workarounds (Akut-Lösungen, welche die Schwachstelle nicht beheben, aber die Ausnutzung wirksam verhindern) zulässig. Workarounds sollten grundlegend nur temporär Verwendung finden und sobald möglich durch eine reguläre Systemaktualisierung ersetzt werden, um die Schwachstellen vollständig zu schließen.

Komponenten, die bekannte, nicht behebbare Schwachstellen enthalten, dürfen in einem System nicht eingesetzt werden.

Die Behandlung von neu bekannt werdenden Schwachstellen muss zudem für die gesamte Einsatzphase des Systems fortlaufend sichergestellt und in den kontinuierlichen Betriebsprozessen eines Security Patch Managements umgesetzt werden.

Motivation: Der Einsatz von Komponenten ohne eine Behebung enthaltener Schwachstellen erhöht deutlich das Risiko einer erfolgreichen Kompromittierung. Der Angreifer wird zusätzlich dadurch begünstigt, dass i. d. R. nicht nur detaillierte Informationen zu bereits bekannt gewordenen Schwachstellen offen verfügbar sind, sondern oft auch bereits

angepasste Angriffswerkzeuge, die eine aktive Ausnutzung erleichtern.

Umsetzungsbeispiel: Im Anschluss an die initiale Installation eines Betriebssystems von einem offiziellen Installationsmedium werden alle aktuell verfügbaren Patches und Sicherheits-Updates eingespielt.

Zusätzliche Informationen:

Primäre Quellen für bereits bekannt gewordene Schwachstellen in einer Software/Hardware sind Auflistungen in den Versionshinweisen sowie die Sicherheitshinweise aus den offiziellen Meldekanälen des Herstellers oder unabhängiger CERTs. Insbesondere die Meldekanäle werden sinnvollerweise fest in kontinuierliche Prozesse des Security Patch Managements für ein System eingebunden, damit neu bekanntwerdende Schwachstellen zeitnah registriert und in operative Behebungsmaßnahmen geführt werden können.

Als ergänzende Maßnahme zur Aufdeckung von potenziell noch enthaltenen, grundsätzlich bereits bekannt gewordenen Arten von Schwachstellen können gezielte Schwachstellenuntersuchungen des Systems durchgeführt werden. Zu diesem Zweck eignen sich besonders spezialisierte Werkzeuge wie automatisierte Schwachstellen-Scanner. Beispiele hierfür sind: Tenable Nessus, Qualys Scanner Appliance.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigter Zugang zum System
- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Unberechtigte Modifikation von Daten
- Unberechtigte Nutzung von Diensten oder Ressourcen
- Störung der Verfügbarkeit
- Abstreiten von durchgeführten Aktionen
- Unbemerkt durchführbare Angriffe
- Angriffe motiviert und erleichtert durch Informationspreisgabe oder erkennbare Sicherheitsschwachstellen

ID: 3.01-10/9.0

3. Konfigurationsanforderungen

Req 4 Die Identität eines Anwendungspools darf kein Benutzerkonto mit Systemprivilegien sein.

Motivation: Wenn der Webserverprozess mit administrativen Berechtigungen läuft, dann kann ein Angreifer, der die Kontrolle über diesen Prozess erlangt, das gesamte System kontrollieren.

Umsetzungsbeispiel: Unter IIS 8, IIS 8.5 und IIS 10 kann ein Anwendungspool im IIS-Manager konfiguriert werden. Nachdem der zu konfigurierende Anwendungspool selektiert wurde klickt man unter „Aktionen“ auf „Erweiterte Einstellungen...“. Bei den erweiterten Einstellungen ist im Bereich „Prozessmodell“ die „Identität“ zu finden. Diese kann durch das Anklicken von „...“ geändert werden. Hierbei darf weder unter „Integriertes Konto“ „LocalService“ oder „LocalSystem“ noch unter „Benutzerdefiniertes Konto“ ein entsprechend privilegiertes Konto gewählt werden.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Unberechtigte Modifikation von Daten

ID: 3.32-4/8.0

Req 5 Der Webserver-Dienst darf nur an die Schnittstellen gebunden werden, über die er erreichbar sein muss.

In den meisten Fällen muss der Webserver-Dienst nur an eine Schnittstelle gebunden werden.

Motivation: Je mehr Schnittstellen Zugriff zum Webserver erlauben, desto höher ist ein Angriffsrisiko.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigter Zugang zum System
- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Unberechtigte Nutzung von Diensten oder Ressourcen

ID: 3.03-5/8.0

Req 6 Nicht benötigte HTTP-Methoden müssen deaktiviert werden.

Standard-Requests an Webserver verwenden nur GET und POST. Werden andere Methoden benötigt, müssen diese sicher verarbeitet werden. TRACE bzw. TRACK muss deaktiviert werden.

Motivation: HTTP TRACE könnte von einem Angreifer missbraucht werden. Diese Methode ermöglicht das Debugging und die Trace-Analyse für Verbindungen vom Client zum Webserver. Der Microsoft IIS-Webserver verwendet den Alias TRACK für diese Methode. Andere HTTP-Methoden könnten auch verwendet werden, um Informationen über den Server zu erhalten oder könnten direkt von einem Angreifer missbraucht werden.

Umsetzungsbeispiel: Unter IIS 8, IIS 8.5 und IIS 10 können nicht benötigte HTTP-Methoden für eine Website wie folgt deaktiviert werden:

Im IIS-Manager wählt man die zu konfigurierende Website aus und öffnet dann in der Ansicht „Features“ unter „IIS“ die „Anforderungsfilerung“. Bei den „Aktionen“ wählt man nun „Featureeinstellungen bearbeiten ...“ und deaktiviert ggf. den Punkt „Nicht aufgelistete Verben zulassen“. Nun wählt man den Tab „HTTP-Verben“ aus und trägt über die Aktion „Verb zulassen ...“ alle zulässigen HTTP-Methoden ein.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Unberechtigte Modifikation von Daten
- Angriffe motiviert und erleichtert durch Informationspreisgabe oder erkennbare Sicherheitsschwachstellen

ID: 3.32-6/8.0

Falls sie nicht benötigt werden, dürfen die Webserver-Rollendienste

- FTP-Server
- IIS-Verwaltungsdienst für die Remoteverwaltung
- Alle Anwendungsentwicklungsfeatures (insbesondere ASP, CGI, ISAPI, Serverseitige Include-Dateien)
- WebDAV-Veröffentlichung

nicht aktiviert werden.

Motivation: Jede Windows-Funktion kann Sicherheitsschwachstellen aufweisen und sollte daher deaktiviert werden, wenn sie nicht benötigt wird.

Umsetzungsbeispiel: In der Powershell kann der Status von Webserver-Rollendiensten abgefragt werden mit dem Befehl

Get-WindowsFeature Web*

Der Ausgabe des Befehls lässt sich auch der Name der Rollendienste entnehmen, wie er in weiteren Befehlen verwendet werden kann.

Unerwünschte Features können mit

Uninstall-WindowsFeature <Name>

entfernt werden. Namen zu deaktivierender Rollendienste sind insbesondere

- Web-Ftp-Server
- Web-Mgmt-Service
- Web-App-Dev
- Web-DAV-Publishing

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigter Zugang zum System
- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Unberechtigte Modifikation von Daten
- Unberechtigte Nutzung von Diensten oder Ressourcen
- Unbemerkt durchführbare Angriffe

ID: 3.32-7/8.0

Motivation: Die Technik „Server Side Includes (SSI)“, die als zusätzliches ladbares Modul implementiert ist, kann potenziell von Angreifern verwendet werden. Insbesondere die „exec“-Funktion von SSI könnte zur Ausführung von Systembefehlen verwendet werden, was ein Risiko darstellt.

Umsetzungsbeispiel: Die Ausführung von Systembefehlen kann für eine Website wie folgt deaktiviert werden: Im IIS-Manager wählt man die zu konfigurierende Website aus und öffnet dann in der Ansicht „Features“ unter „Verwaltung“ den „Konfigurationseditor“. Im Abschnitt „system.webServer/serverSideInclude“ ändert man dann den Wert von „ssiExecDisable“ auf „True“.

Alternativ kann die folgende Befehlszeile verwendet werden:

Set-WebConfigurationProperty -PSPath 'IIS:\' -location "<web_site>" -filter 'system.webServer/serverSideInclude' -name 'ssiExecDisable' -value 'True'

"<web_site>" ist hierbei durch die zu konfigurierende Web Site zu ersetzen. Die Einstellung kann auch für alle Websites geändert werden:

Set-WebConfigurationProperty -PSPath 'IIS:\' -filter 'system.webServer/serverSideInclude' -name 'ssiExecDisable' -value 'True'

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigter Zugang zum System
- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Unberechtigte Modifikation von Daten
- Unberechtigte Nutzung von Diensten oder Ressourcen
- Störung der Verfügbarkeit
- Unbemerkt durchführbare Angriffe

ID: 3.32-8/8.0

Req 9	CGI soll nicht verwendet werden.
-------	----------------------------------

Motivation: Durch eine unsachgemäße CGI-Konfiguration kann sich eine Vielzahl von Angriffsmöglichkeiten auf einen Webserver ergeben. Da bei modernen Webservern verschiedene sowohl sicherere als auch performantere Alternativen zu CGI existieren ist die Verwendung von CGI weder nötig noch zu empfehlen.

Umsetzungsbeispiel: In der Powershell kann CGI mit

Uninstall-WindowsFeature Web-CGI

deaktiviert werden.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigter Zugang zum System
- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Unberechtigte Modifikation von Daten

ID: 3.32-9/8.0

Req 10	Falls WebDAV für das Schreiben von Dateien verwendet wird, darf Zugriff nicht ohne erfolgreiche Authentifizierung gewährt werden.
--------	---

Motivation: WebDav ermöglicht den Online-Update von Content, der durch den Webserver zur Verfügung gestellt wird. Diese Funktion könnte daher zur Veränderung von Website-Inhalten missbraucht werden.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigte Modifikation von Daten

ID: 3.03-10/8.0

Req 11	Falls WebDAV verwendet wird, muss der Zugriff auf erforderliche Verzeichnisse und bezüglich der berechtigten Anwender beschränkt werden.
--------	--

Die Zugriffsrechte auf alle Dateien, auf die über WebDAV zugegriffen werden kann, müssen möglichst restriktiv gesetzt werden. Zusätzlich muss der WebDAV-Zugriff auf die benötigten Verzeichnisse beschränkt werden.

Motivation: WebDav ermöglicht den Online-Update von Content, der durch den Webserver zur Verfügung gestellt wird. Diese Funktion könnte daher zur Veränderung von Website-Inhalten missbraucht werden.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigte Einsichtnahme oder Abgriff von Daten

- Unberechtigte Modifikation von Daten

ID: 3.03-11/8.0

Req 12 Zugriffsrechte für Webserver-Konfigurationsdateien dürfen nur dem Eigentümer des Webserver-Prozesses oder einem Benutzer mit Systemprivilegien gewährt werden.

Motivation: Konfigurationsdateien dürfen nur vom Eigentümer des Webserver-Prozesses oder einem Benutzer mit Systemprivilegien geschrieben werden. Andernfalls können unbefugte Benutzer die Konfiguration des Webserver ändern oder Konfigurationsinformationen erhalten, die für einen Angriff verwendet werden könnten.

Umsetzungsbeispiel: „Lese-“ und „Schreib“-Zugriffsrechte für „andere“ entfernen. „Schreib“-Zugriff nur dem Benutzer erlauben, der den Webserver konfiguriert.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Unberechtigte Modifikation von Daten
- Unberechtigte Nutzung von Diensten oder Ressourcen
- Störung der Verfügbarkeit

ID: 3.03-12/8.0

Req 13 Falls die „Default Web Site“ nicht verwendet wird, muss diese entfernt werden.

Motivation: Die "Default Website" wird in einer Standard-Konfiguration mit Beispieldateien ausgeliefert. Wenn ein Angreifer auf die Default Website Zugriff erhält, kann er daher Rückschlüsse auf das verwendete System ziehen.

Umsetzungsbeispiel: Im IIS Manager klickt man mit der rechten Maustaste auf die "Default Website" und wählt "Entfernen".

Alternativ kann man in der Powershell den folgenden Befehl eingeben

Remove-Website -Name " Default Web Site"

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Angriffe motiviert und erleichtert durch Informationspreisgabe oder erkennbare Sicherheitsschwachstellen

ID: 3.32-13/8.0

Req 14 Default-Dateien im Verzeichnis einer Website müssen gelöscht werden.

Bei einer neu angelegten Website können sich Default-Dateien im Verzeichnis der Website befinden. Üblicherweise handelt es sich um eine Index-HTML-Datei und eine Image-Datei, es können aber auch weitere Beispieldateien und Tutorials vorhanden sein. Diese Dateien müssen gelöscht werden. Dies betrifft insbesondere die „Default Web Site“, falls diese verwendet wird.

Motivation: Durch die Verwendung von Beispielen könnten Informationen über die installierte Software (Version) erlangt werden. Beispiele können Sicherheitsschwachstellen enthalten.

Umsetzungsbeispiel: Nach dem Erstellen einer Website alle Dateien im Wurzelverzeichnis einer Web Site löschen.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Angriffe motiviert und erleichtert durch Informationspreisgabe oder erkennbare Sicherheitsschwachstellen

ID: 3.32-14/8.0

Req 15	Die Windows-Funktion „Verzeichnis durchsuchen“ der Internetinformationsdienste bzw. der Webserver-Rollendienst „Verzeichnissuche“ muss deaktiviert werden.
--------	--

Motivation: Verzeichnislisten enthalten Informationen über Dateien und Verzeichnisstrukturen, die missbraucht werden könnten.

Umsetzungsbeispiel: In der Powershell deaktiviert der folgende Befehl das Erstellen von Verzeichnislisten:

Uninstall-WindowsFeature Web-Dir-Browsing

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Angriffe motiviert und erleichtert durch Informationspreisgabe oder erkennbare Sicherheitsschwachstellen

ID: 3.32-15/8.0

Req 16	Der HTTP-„Server“-Header darf Informationen zu Software und Version des Webserver nicht enthalten.
--------	--

Motivation: Alle Informationen über den Webserver könnten über Sicherheitsschwachstellen Aufschluss geben.

Umsetzungsbeispiel: Für IIS 10 kann die Ausgabe des "Server"-Headers in der PowerShell durch den folgenden Befehl serverweit deaktiviert werden:

```
Set-WebConfigurationProperty -PSPath 'IIS:\' -filter  
'system.webServer/security/requestFiltering' -name  
'removeServerHeader' -value 'True'
```

Bei älteren IIS-Versionen kann unter Verwendung des Tools UrlScan im Bereich „[Options]“ der UrlScan- Konfigurationsdatei entweder der Eintrag

```
RemoveServerHeader=1
```

oder

```
AlternateServerName=<Webserver>
```

vorgenommen, wobei <Webserver> etwa durch den String „Webserver“ ersetzt werden kann.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Angriffe motiviert und erleichtert durch Informationspreisgabe oder erkennbare Sicherheitsschwachstellen

ID: 3.32-16/8.0

Req 17	Zusätzliche HTTP-Antwortheader, die Informationen zu Software und Version des Webserver oder verwendeter Komponenten beinhalten, dürfen nicht gesetzt sein.
--------	---

Motivation: Alle Informationen über den Webserver oder verwendete Komponenten könnten über Sicherheitsschwachstellen Aufschluss geben.

Umsetzungsbeispiel: In der PowerShell können alle zusätzlichen Antwortheader durch die Eingabe des folgenden Befehls entfernt werden:

Remove-WebConfigurationProperty -PSPath 'IIS:\' -filter 'system.webServer/httpProtocol/customHeaders' -Name .

Alternativ wählt man bei der Nutzung des IIS-Managers den Server oder die zu konfigurierende Website aus und öffnet dann in der Ansicht „Features“ unter „IIS“ die „HTTP-Antwortheader“. Header, die unzulässige Informationen enthalten wie etwa ein „X-powered-by“-Header, entfernt man nun.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Angriffe motiviert und erleichtert durch Informationspreisgabe oder erkennbare Sicherheitsschwachstellen

Req 18	Informationen über den Webserver in Fehlerseiten, die durch den Webserver ausgeliefert werden, müssen entfernt werden.
--------	--

Die Standardfehlerseiten müssen durch benutzerspezifisch definierte Fehlerseiten ersetzt werden.

Benutzerspezifisch definierte Fehlerseiten dürfen keine Versionsinformationen des Webserver und verwendeter Module/Erweiterungen enthalten. Die Fehlermeldungen dürfen keine internen Informationen, wie z. B. interne Servernamen, Fehlercodes usw. enthalten.

Motivation: Alle Informationen über den Webserver könnten Aufschluss über Sicherheitsschwachstellen geben.

Umsetzungsbeispiel: Im IIS-Manager wählt man den Server oder die zu konfigurierende Website aus und öffnet dann in der Ansicht „Features“ unter „IIS“ die „Fehlerseiten“. Entweder editiert man nun die Fehlerdateien unter dem dort angegebenen Pfad oder gibt einen neuen Pfad zu eigenen Fehlerseiten ohne Informationen über Webserver- Produkt und –Version ein.

Standardmäßig werden die Fehlerseiten auf Serverebene definiert und dann an die einzelnen Websites vererbt. Soweit dies nicht geändert wird ist es also ausreichend, die auf Serverebene definierten Fehlerdateien zu ändern.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Angriffe motiviert und erleichtert durch Informationspreisgabe oder erkennbare Sicherheitsschwachstellen

Req 19	Unautorisierte Änderungen von Web.config-Dateien müssen unterbunden werden.
--------	---

Die Konfiguration eines Microsoft IIS kann in Web.config-Dateien für Sites, Anwendungen oder Verzeichnisse verteilt werden. Die Zugriffsrechte auf diese Dateien müssen so restriktiv wie möglich gesetzt sein. Insbesondere dürfen die Dateien für Benutzer ohne administrative Privilegien nicht schreibbar sein.

Web.config-Dateien, die vom IIS Manager erzeugt werden, sind bereits mit restriktiven Zugriffsrechten versehen. Insbesondere auch bei eingebundenen (virtuellen) Verzeichnissen ist aber darauf zu achten, dass für keinen unberechtigten Benutzer die Möglichkeit besteht, diese Dateien zu erzeugen oder zu modifizieren.

Motivation: Die Konfiguration einer Website könnte von anderen Benutzern mithilfe von Web.config-Dateien, geändert werden. Ein Benutzer kann sich auf diese Weise beispielsweise unautorisierten Zugriff auf Dateien des Webserver verschaffen.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Unberechtigte Modifikation von Daten
- Unberechtigte Nutzung von Diensten oder Ressourcen
- Störung der Verfügbarkeit

Req 20	Die in der Web.Config möglichen Einstellungen müssen so weit wie möglich eingeschränkt werden.
--------	--

Alle Sektionen in der zentralen Konfigurationsdatei ApplicationHost.config, die sich auf Sicherheit oder kritische Aspekte beziehen, müssen gesperrt sein, soweit eine individuelle Einstellung auf Ebene von Websites nicht zwingend erforderlich ist. Standardmäßig ist dies der Fall, Änderungen dürfen nicht ohne zwingenden Grund durchgeführt werden.

Motivation: Die Einstellungen in nicht gesperrten Sektionen können durch verteilte Web.config-Dateien überschrieben werden, die sich in Dokumentenverzeichnissen befinden. Einem Benutzer wird es hierdurch gegebenenfalls erleichtert, unautorisierte Konfigurationsänderungen durchzuführen.

Umsetzungsbeispiel: Die Datei ApplicationHost.config befindet sich im Verzeichnis %windir%\system32\inetsrv\config. Die Eigenschaften der Sektionen werden in dem Bereich <configSections> definiert. Eine Sektion ist dann gesperrt, wenn das Attribut „overrideModeDefault“ auf “Deny” gesetzt wird. Bei den meisten Sektionen ist der Standardwert in Ordnung, sollte aber kontrolliert werden. Eine Änderung ist erforderlich für die Sektion "directoryBrowse". Hier muss der Eintrag korrigiert werden zu

```
<section name="directoryBrowse" overrideModeDefault="Deny" />
```

Diese Änderung kann auch in der PowerShell durch den folgenden Befehl vorgenommen werden

```
Set-WebConfigurationProperty -PSPath 'IIS:\' -filter 'system.webServer' -name 'sections[directoryBrowse].overrideModeDefault' -value Deny
```

Die aktuelle Einstellung kann mit dem folgenden Befehl angezeigt werden

```
Get-WebConfigurationProperty -PSPath 'IIS:\' -filter 'system.webServer' -name 'sections[directoryBrowse].overrideModeDefault'
```

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Unberechtigte Modifikation von Daten
- Unberechtigte Nutzung von Diensten oder Ressourcen
- Störung der Verfügbarkeit

ID: 3.32-20/8.0

Req 21 Der Webserver darf nur Dateien ausliefern, die für die Auslieferung bestimmt sind.

Alle Dateien, die sich direkt oder indirekt (beispielsweise über Links oder in virtuellen Verzeichnissen) im Dokumentenverzeichnis des Webserver befinden, müssen mit restriktiven Zugriffsrechten versehen werden. Insbesondere darf der Webserver nicht auf Dateien zugreifen können, die nicht für die Auslieferung bestimmt sind.

Beim IIS muss hier insbesondere auf virtuelle Verzeichnisse geachtet werden, über die beispielsweise der Content anderer Anwendungen eingebunden werden kann.

Motivation: Insbesondere wenn zusätzliche Dateien oder Verzeichnisse über Links oder virtuelle Verzeichnisse in das Dokumentenverzeichnis des Webserver eingebunden werden besteht die Möglichkeit, dass ein Benutzer über den Webserver auf Dateien zugreifen kann, die er nicht einsehen dürfte. Dies muss durch eine sorgfältige Konfiguration verhindert werden.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigte Einsichtnahme oder Abgriff von Daten

ID: 3.32-21/8.0

Req 22 Der Webserver muss sich gegen Überlastsituationen schützen.

Ein Webserver muss über Schutzmechanismen verfügen, die Überlastsituationen soweit wie möglich verhindern. Insbesondere ist eine partielle oder komplette Beeinträchtigung der Verfügbarkeit des Webserver zu vermeiden. Beispiele für mögliche Schutzmaßnahmen sind:

- Begrenzung der maximalen Anzahl von HTTP Sessions pro IP-Adresse
- Festlegen der maximalen Größe eines HTTP Requests
- Festlegen eines Timeouts für HTTP Requests

Begrenzungen müssen unter Berücksichtigung der zu schützenden Anwendung und ihrer Besonderheiten festgelegt werden. Als Richtlinie können hierbei die folgenden Werte dienen.

Falls über den Webserver keine Uploads erfolgen sollen:

- Maximale Anzahl von HTTP Sessions pro IP Adresse: 50

- Maximale Größe eines HTTP Requests: 20000 Bytes
- Timeout für HTTP Requests: 30 Sekunden

Falls über den Webserver auch Uploads erfolgen sollen:

- Maximale Anzahl von HTTP Sessions pro IP Adresse: 50
- Maximale Größe eines HTTP Requests: 10000000 Bytes oder, falls bekannt, maximale Größe des zu erwartenden Uploads
- Timeout für HTTP Requests: 60 Sekunden oder, falls bekannt, für maximalen Upload benötigte Zeit

Motivation: Angreifer versuchen oftmals mittels Denial-of-Service-Angriffen einen Webserver in eine Überlastsituation zu bringen. Ist ein solcher Angriff erfolgreich kann dies zur Beeinträchtigung der Verfügbarkeit oder Integrität des Webservers führen.

Umsetzungsbeispiel: Um eine Einschränkung der HTTP Sessions pro IP Adresse vornehmen zu können muss die hierzu erforderliche Windows-Funktion bzw. der IIS-Rollendienst "IP- und Domäneneinschränkungen" installiert sein. Die Installation ist in der PowerShell über den folgenden Befehl möglich

Install-WindowsFeature Web-IP-Security

Alle Einstellungen können im IIS-Manager vorgenommen werden. Die Anzahl der HTTP Sessions pro IP Adresse und die maximale Größe eines HTTP Requests können für den gesamten Server oder einzelne Web Sites festgelegt werden. Der Timeout kann nur für einzelne Web Sites konfiguriert werden.

Um die Anzahl der HTTP Sessions pro IP Adresse zu konfigurieren wählt man den Server oder eine Web Site und öffnet dann das Feature "Einschränkungen für IP-Adressen und Domänen". Unter "Aktionen" klickt man dann auf "Einstellungen für die dynamische Einschränkung bearbeiten...". In dem sich öffnenden Fenster aktiviert man nun "IP-Adresse auf Grundlage der Anzahl gleichzeitiger Anforderungen verweigern".

Um die maximale Größe eines HTTP Requests zu konfigurieren wählt man den Server oder eine Web Site und öffnet dann das Feature "Anforderungsfilterung". Unter "Aktionen" klickt man dann auf "Featureeinstellungen bearbeiten...". In dem sich öffnenden Fenster kann nun unter "Maximal zulässige Inhaltslänge (Bytes)" die Größe des Requests festgelegt werden.

Um einen Timeout für HTTP Requests zu konfigurieren wählt man eine Web Site und klickt dann unter "Aktionen" bei "Konfigurieren" auf "Limits...". In dem sich öffnenden Fenster kann nun unter "Verbindungstimeout (in Sekunden)" der Timeout festgelegt werden.

Die Werte können auch über die PowerShell gesetzt werden. In den folgenden Befehlen ist hierbei <web_site> durch den Namen der Web Site und <value> durch den zu setzenden Wert zu ersetzen.

Einstellungen systemweit setzen:

```
Set-WebConfigurationProperty -PSPath 'IIS:\' -filter system.webServer/security/dynamicIpSecurity/denyByConcurrentRequests -name Enabled -value True
```

```
Set-WebConfigurationProperty -PSPath 'IIS:\' -filter system.webServer/security/requestFiltering/requestLimits -name maxAllowedContentLength -value <value>
```

```
Set-WebConfigurationProperty -PSPath 'IIS:\' -filter 'system.applicationHost/sites/siteDefaults/limits' -name connectionTimeout -value <value>
```

Einstellungen für eine Web Site setzen:

```
Set-WebConfigurationProperty -PSPath 'IIS:\' -location "<web_site>" -filter system.webServer/security/dynamicIpSecurity/denyByConcurrentRequests -name maxConcurrentRequests -value <value>
```

```
Set-WebConfigurationProperty -PSPath 'IIS:\' -location "<web_site>" -filter sy-
```

`stem.webServer/security/requestFiltering/requestLimits -name maxAllowedContentLength -value <value>`

`Set-WebConfigurationProperty -PSPath 'IIS:\' -filter 'system.applicationHost/sites/site[@name = "<web_site>"]/limits' -name connectionTimeout -value <value>`

Für diese Anforderung sind folgende Gewährleistungsziele relevant:

- Verfügbarkeit
- Integrität

ID: 3.32-22/8.0

4. HTTPS-Anforderungen

Req 23	Schutzbedürftige Daten müssen während der Übertragung vor nicht autorisierter Einsichtnahme und Veränderung geschützt werden.
--------	---

Der Schutzbedarf von zu übertragenden Daten ist abhängig von deren Einstufung (beispielsweise nach geltenden gesetzlichen Datenschutzvorgaben, regulatorischen Auflagen, vertraglichen Verpflichtungen), dem Schadenspotential bei deren Missbrauch, sowie weiteren relevanten Faktoren (beispielsweise der Übertragung über öffentliche Netze). Art und Umfang der Schutzmaßnahmen müssen entsprechend angemessen gewählt werden. Authentisierungsmerkmale wie beispielsweise Passwörter oder Tokens gelten grundsätzlich als schutzbedürftige Daten. Ebenfalls grundsätzlich als schutzbedürftig gelten Daten, die Funktionalität und sicherheitsrelevantes Verhalten eines Systems bestimmen (z. B. Updates & Patches, Konfigurationsparameter, Fernwartungen, Steuerung über APIs).

Die Einhaltung der Schutzziele Vertraulichkeit und Integrität muss während der Übertragung von schutzbedürftigen Daten durchgängig gewährleistet werden.

I.d.R. erfordert dies eine Implementation von kryptografischen Verfahren (z. B. Verschlüsselung, Signaturen, Hashes). Kryptografische Verfahren können

- vor der Übertragung direkt auf die Daten angewendet werden, was eine anschließende Übertragung der Daten auch über unsichere Kanäle vertretbar machen kann
- auf dem Übertragungs-Kanal eingesetzt sein, um einen sicheren Kanal zu schaffen und beliebige Daten, die durch diesen geleitet werden, zu schützen
- oder auch als Kombination aus beidem umgesetzt sein.

Bei der Übertragung von Daten eingesetzte kryptografische Verfahren müssen für diesen Zweck anerkannt wirksam sein und dürfen keine bekannten Schwachstellen aufweisen.

Motivation: Die Übertragung von Daten ohne einen adäquaten Schutz ermöglicht es einem Angreifer, diese unberechtigt abzufangen, zu nutzen, zu verbreiten, zu verändern oder aus der Übertragung zu entfernen. Dies eröffnet potenziell weitere Angriffsvektoren auf die unmittelbaren Zielsysteme sowie angeschlossene weitere Systeme und kann zu erheblichen Ausfällen, Kontrollverlusten und Schäden sowie resultierenden Strafzahlungsforderungen und Reputationsverlusten gegenüber Kunden und Geschäftspartnern führen.

Umsetzungsbeispiel: [Beispiel 1]

Vertrauliche Dokumente werden vor dem Versand per E-Mail an den Kunden verschlüsselt.

[Beispiel 2]

Ein Administrator konfiguriert eine neue Cloud-Anwendung über das Internet. Der Zugriff erfolgt über eine TLS-verschlüsselte Verbindung ("https").

[Beispiel 3]

Ein System bezieht automatische Software-Updates von einem Update-Server. Der Update-Server liefert die Software Updates kryptografisch signiert aus. Das System kann die empfangenen Software-Updates somit validieren und sicher ausschließen, dass diese während der Übertragung manipuliert wurden.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Unberechtigte Modifikation von Daten
- Störung der Verfügbarkeit
- Unbemerkt durchführbare Angriffe
- Angriffe motiviert und erleichtert durch Informationspreisgabe oder erkennbare Sicherheitsschwachstellen

ID: 3.01-15/9.0

Req 24	Für die Verschlüsselung mit HTTPS muss das TLS-Protokoll in Version 1.2 oder höher verwendet werden.
--------	--

SSL und TLS 1.0/1.1 sind als veraltet anzusehen und dürfen daher nicht aktiviert werden bzw. müssen deaktiviert werden. TLS in der Version 1.2 besitzt eine ausreichende Designsicherheit und bietet ebenfalls Cipher mit Authenticated Encryption Associated Data (AEAD)

Motivation: Die aktuellen Versionen von TLS schließen bisherige Sicherheitsschwachstellen und Angriffsflächen auf den TLS Protokoll-Handshake.

Umsetzungsbeispiel: Die Konfiguration von SSL/TLS wird durch entsprechende Registry-Einträge vorgenommen. Für eine ausschließliche Verwendung des TLS-Protokolls ab Version 1.2 sind folgende Einträge durchzuführen:

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\SSL
2.0\Server] "Enabled"=dword:00000000
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\SSL
3.0\Server] "Enabled"=dword:00000000
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\TLS
1.0\Server] "Enabled"=dword:00000000
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\TLS
1.1\Server] "Enabled"=dword:00000000
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\TLS
1.2\Server] "Enabled"=dword:ffffff
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\TLS
1.3\Server] "Enabled"=dword:ffffff
```

Der Wert von „Enabled“ gibt jeweils an, ob das Protokoll verwendet werden kann (Wert ungleich 0) oder nicht (Wert 0). Nicht vorhandene Schlüssel müssen gegebenenfalls erzeugt werden.

TLS 1.3 wird gegebenenfalls auf der eingesetzten Windows Server Version nicht unterstützt. In diesem Fall entfällt der letzte Eintrag.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Unberechtigte Modifikation von Daten

ID: 3.32-24/8.0

Req 25	Der Webserver muss so konfiguriert werden, dass die Nutzung der aktuellen Version des TLS-Protokolls möglich ist.
--------	---

Motivation: Die aktuelle Version des Protokolls bietet den höchsten Schutz und enthält Behebungen von Schwachstellen älterer Versionen.

Umsetzungsbeispiel: Ab Windows Server 2008 R2 muss die Registry diese Einträge enthalten:

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\TLS
1.2\Server] "Enabled"=dword:ffffff
```

und, falls unterstützt,

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\TLS
1.3\Server] "Enabled"=dword:ffffff
```

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Unberechtigte Modifikation von Daten

ID: 3.32-25/8.0

Req 26	Die TLS-Konfiguration muss sichere Cipher-Suites, die Perfect Forward Secrecy (PFS) sicher stellen, verwenden.
--------	--

Cipher Suites spezifizieren die kryptographischen Verfahren einer TLS-Verbindung.

Perfect Forward Secrecy (kurz PFS, auch Forward Secrecy) bedeutet, dass Verbindungsinhalte auch bei Kenntnis der Langzeit-Schlüssel der Kommunikationspartner nicht nachträglich entschlüsselt werden können.

In TLS v1.2 werden Cipher Suites wie folgt definiert: TLS_AKE_WITH_Enc_Hash.

Nachfolgend wird die Bedeutung der einzelnen Bestandteile erläutert:

- AKE (Authenticated Key Exchange): Authentisiertes Schlüsseleingangsverfahren für das Handshake-Protokoll
- Enc (Encryption): Verschlüsselungsverfahren mit Betriebsart für das Record-Protokoll
- Hash: Hashfunktion für einen HMAC zur Schlüsselableitung. Sofern Enc kein AEAD-Verschlüsselungsverfahren ist, wird der HMAC außerdem zur Integritätssicherung verwendet.

Die nachfolgende Tabelle listet die zulässigen Cipher Suites mit PFS in TLS v1.2 sowie die Referenzspezifikationen auf. Dabei wurde der Designphilosophie von TLS v1.3 gefolgt, weshalb in dieser Tabelle lediglich AEAD-Konstruktionen aufgelistet sind.

Cipher Suite	IANA-Nr.	Referenzspezifikation
TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	0xC0,0x2B	[RFC-5289]
TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	0xC0,0x2C	[RFC-5289]
TLS_ECDHE_ECDSA_WITH_AES_128_CCM	0xC0,0xAC	[RFC-7251]
TLS_ECDHE_ECDSA_WITH_AES_256_CCM	0xC0,0xAD	[RFC-7251]
TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256	0xC0,0x2F	[RFC-5289]
TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	0xC0,0x30	[RFC-5289]
TLS_ECDHE_ECDSA_WITH_CHACHA20POLY1305_SHA256	0xCC,0xA9	[RFC-7905]
TLS_ECDHE_RSA_WITH_CHACHA20POLY1305_SHA256	0xCC,0xA8	[RFC-7905]

Die folgenden Cipher Suites mit PFS in TLS v1.2 mit AES-CBC (CBC-Modus) und / oder DHE Cipher Suites sind nur in Bestandsystemen erlaubt:

Cipher Suite	IANA-Nr.	Referenzspezifikation	Hinweis
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	0xC0,0x23	[RFC-5289]	CBC

TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384	0xC0,0x24	[RFC-5289]	CBC
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256	0xC0,0x27	[RFC-5289]	CBC
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384	0xC0,0x28	[RFC-5289]	CBC
TLS_DHE_DSS_WITH_AES_128_GCM_SHA256	0x00,0xA2	[RFC-5288]	DHE
TLS_DHE_DSS_WITH_AES_256_GCM_SHA384	0x00,0xA3	[RFC-5288]	DHE
TLS_DHE_RSA_WITH_AES_128_GCM_SHA256	0x00,0x9E	[RFC-5288]	DHE
TLS_DHE_RSA_WITH_AES_256_GCM_SHA384	0x00,0x9F	[RFC-5288]	DHE
TLS_DHE_RSA_WITH_AES_128_CCM	0xC0,0x9E	[RFC-6655]	DHE
TLS_DHE_RSA_WITH_AES_256_CCM	0xC0,0x9F	[RFC-6655]	DHE
TLS_DHE_RSA_WITH_CHACHA20POLY1305_SHA256	0xCC,0xAA	[RFC-7905]	DHE
TLS_DHE_DSS_WITH_AES_128_CBC_SHA256	0x00,0x40	[RFC-5246]	CBC / DHE
TLS_DHE_DSS_WITH_AES_256_CBC_SHA256	0x00,0x6A	[RFC-5246]	CBC / DHE
TLS_DHE_RSA_WITH_AES_128_CBC_SHA256	0x00,0x67	[RFC-5246]	CBC / DHE
TLS_DHE_RSA_WITH_AES_256_CBC_SHA256	0x00,0x6B	[RFC-5246]	CBC / DHE

Hinweis zu DHE Cipher Suites:

Für das authentifizierte Schlüsseleinigungsverfahren (AKE) werden Verfahren basierend auf elliptischen Kurven (ECDHE_ECDSA) bevorzugt. DHE (diskreter Logarithmus) Schlüsselvereinbarungen sind anfälliger gegen DoS (DHeater) Angriffe als ECDHE (elliptische Kurven). Sie werden voraussichtlich vom IETF abgekündigt und vom BSI [TR-02102-2] nur bis zum Jahr 2029 empfohlen.

Hinweis zu Cipher Suites mit AES-CBC:

Cipher Suites mit AES-CBC sind nur mit der TLS-Erweiterung "Encrypt-then-MAC" vom BSI empfohlen ([TR-02102-2]). Die Erweiterung ist aber in den meisten Implementierungen nicht aktivierbar bzw. eine Überprüfung, ob sie aktiv ist, ist nicht möglich.

In TLS v1.3 werden Cipher Suites wie folgt definiert: TLS_AEAD_Hash.

Nachfolgend wird die Bedeutung der einzelnen Bestandteile erläutert:

- AEAD: Authentifiziertes Verschlüsselungsverfahren für das Record-Protokoll
- Hash: Hashfunktion für HMAC und HKDF im Handshake-Protokoll

Die nachfolgende Tabelle listet die empfohlenen Cipher Suites mit PFS in TLS v1.3 sowie die Referenzspezifikationen auf.

Cipher Suite	IANA-Nr.	Referenzspezifikation
TLS_AES_128_GCM_SHA256	0x13,0x01	[RFC-8446]
TLS_AES_256_GCM_SHA384	0x13,0x02	[RFC-8446]
TLS_CHACHA20_POLY1305_SHA256	0x13,0x03	[RFC-8446]
TLS_AES_128_CCM_SHA256	0x13,0x04	[RFC-8446]

Hinweis zu CHACHA20 Cipher Suites:

CHACHA20 Cipher Suites: werden nicht (explizit) vom BSI empfohlen, aber bisher sind keine Schwachstellen bekannt, so dass sie als sicher einzustufen sind.

Alle zukünftig spezifizierten Cipher Suites, die den in diesem Dokument definierten Anforderungen entsprechen, können ebenfalls verwendet werden. Dies gilt beispielsweise für Cipher Suites, die eine Hash-Funktion aus der SHA-3-Familie verwenden.

Motivation: Als unsicher erkannte Cipher-Suites bieten keinen ausreichenden Schutz.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Unberechtigte Modifikation von Daten

ID: 3.03-23/8.0

Req 27 Die TLS-Konfiguration muss vorsehen, dass beim Verbindungsaufbau die als am sichersten einzustufenden Cipher-Suites mit oberster Priorität ausgewählt werden.

Motivation: Beim Aufbau einer TLS-Verbindung wird auf Basis der sowohl auf Client-Seite als auch auf Server-Seite zur Verfügung stehenden Cipher-Suites eine zu verwendende Cipher-Suite ausgewählt. Um eine möglichst hohe Kompatibilität zu verschiedenen Client-Systemen sicherzustellen können auf Server-Seite nicht nur die aus Sicherheitssicht besten Cipher-Suites angeboten werden. Damit für jeden Client aus den verfügbaren Cipher-Suites dennoch die beste ausgewählt wird und dadurch die Verbindung am besten geschützt wird ist eine entsprechende Priorisierung in der Konfiguration erforderlich.

Umsetzungsbeispiel: Sowohl die zu verwendenden Cipher-Suites als auch die Reihenfolge, in der die Cipher-Suites verwendet werden, kann über den Editor für lokale Gruppenrichtlinien festgelegt werden.

1. Durch die Eingabe von gpedit.msc an einem Befehlsprompt wird der Editor für lokale Gruppenrichtlinien gestartet.
2. In der Baumstruktur öffnet man nun unter "Computerkonfiguration" erst "Administrative Vorlagen" und dann "Netzwerk". Schließlich klickt man auf "SSL-Konfigurationseinstellungen".
3. Unter "SSL-Konfigurationseinstellungen" klickt man doppelt auf "Reihenfolge der SSL-Verschlüsselungssammlungen".
4. In dem Fenster "Reihenfolge der SSL-Verschlüsselungssammlungen" kann unter "Optionen" nun die Reihenfolge der zu verwendeten Cipher-Suites angegeben werden. Hierzu sind die Hinweise unter "Hilfe" zu beachten. Insbesondere darf die eingegebene Zeichenkette nicht länger als 1023 Zeichen sein.
5. Die geänderten Einstellungen werden erst nach einem Neustart des Systems aktiv

Es sind die folgenden Cipher-Suites zu verwenden. Die Cipher-Suites sind von der höchsten Priorität oben zur niedrig-

sten Priorität unten aufgeführt. Unter Betriebssystemen vor Windows Server 2012 muss die Liste gegebenenfalls gekürzt werden.

TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384_P521
TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384_P384
TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384_P256
TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256_P521
TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256_P384
TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256_P256
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384_P384
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384_P256
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256_P384
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256_P256
TLS_DHE_DSS_WITH_AES_256_CBC_SHA256
TLS_DHE_DSS_WITH_AES_128_CBC_SHA256

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Unberechtigte Modifikation von Daten

ID: 3.32-27/8.0

Req 28	Zertifikate müssen von einer Zertifizierungsstelle ausgestellt werden, deren Zertifikate von den gängigen Webbrowsern erkannt werden.
--------	---

Für kritische Anwendungen, die aus dem Internet nutzbar sind, wird empfohlen, dass ein Extended-Validation-Zertifikat (EV-Zertifikat) verwendet wird.

Motivation: Nur wenn die Zertifizierungsstelle (Certificate Authority, CA) in der CA-Liste des verwendeten Browsers enthalten ist, kann der Browser die Authentizität eines Servers bzw. einer Web-Anwendung überprüfen. Die Ausgabe von EV-Zertifikaten ist an strengere Vergabekriterien gebunden. Wird ein EV-Zertifikat verwendet, wird dies im Browser visualisiert. Auch wenn durch EV-Zertifikate an sich kein Gewinn an Sicherheit erzielt wird, erhöht deren Verwendung die Vertrauenswürdigkeit des Servers beim Nutzer.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Angriffe motiviert und erleichtert durch Informationspreisgabe oder erkennbare Sicherheitsschwachstellen

ID: 3.03-25/8.0

Req 29	Zertifikate müssen spätestens nach 1 Jahr ihre Gültigkeit verlieren.
--------	--

Bei Zertifikaten einer internen CA, insbesondere für Maschinenschnittstellen, darf der Zeitraum auf maximal 3 Jahre verlängert werden.

Motivation: Die Methoden und Techniken, mit denen kryptographische Verfahren analysiert und gebrochen werden, werden fortlaufend weiterentwickelt. Daher kann die Sicherheit von Zertifikaten nur für einen begrenzten Zeitraum gewährleistet werden. Bei einer ausreichenden Schlüssellänge ist nach allgemeiner Einschätzung die Sicherheit der Zertifikate für die geforderte Gültigkeitsdauer von einem Jahr aber gewährleistet.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Unberechtigte Modifikation von Daten

ID: 3.03-26/8.0

Req 30	Zertifikate müssen eine Schlüssellänge von mindestens 3072 Bit bei Verwendung von RSA oder mindestens 256 Bit bei Verwendung von ECC aufweisen.
--------	---

Hinweise zu DSA und RSA Zertifikaten:

Für DSA und RSA dürfen Schlüssellängen kleiner als 3000 Bit nur noch in Bestandssystemen [BSI TR-02102-1] bis zum Ende des Jahres 2025 [2] eingesetzt werden und sollten zum nächstmöglichen Zeitpunkt ersetzt werden. Aufgrund der besseren Performanz, sind elliptische Kurven Zertifikate (EC-DSA) zu bevorzugen (falls technisch unterstützt und realisierbar). RSA-PKCS#1 v1.5 darf nur noch in Bestandssystemen eingesetzt werden und sollte, wenn technisch möglich, zum nächstmöglichen Zeitpunkt ersetzt werden [BSI TR-02102-1].

Quellen:

[BSI TR-02102-1] Bundesamt für Sicherheit in der Informationstechnik: Kryptographische Verfahren: Empfehlungen und Schlüssellängen, TR-02102-1, Version 2022-01, 28.01.2022

Motivation: Um die Sicherheit der Zertifikate über den Gültigkeitszeitraum zu gewährleisten, müssen die kryptographischen Schlüssel eine entsprechende Länge aufweisen. Nach allgemeiner Einschätzung bietet eine Schlüssellänge von 3072 Bit einen ausreichenden Schutz für die nächsten Jahre. Bei ECC-Algorithmen bieten kürzere Schlüssellängen bereits dasselbe Maß an Sicherheit.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Unberechtigte Modifikation von Daten

ID: 3.03-27/8.0

5. Protokollierung

Req 31 Der Zugriff auf den Webserver muss protokolliert werden.

Das Protokoll muss die folgenden Informationen enthalten:

- Zugriffszeitpunkt
- Quelle (IP-Adresse)
- Konto (falls bekannt)
- URL
- Status-Code der Antwort des Webserver

Bei der Protokollierung müssen die jeweils geltenden gesetzlichen, tariflichen und betrieblichen Bestimmungen beachtet werden. Diese besagen unter anderem, dass die Protokollierung von Ereignissen nur zweckgebunden erfolgen darf. Die Protokollierung von Ereignissen, zum Zwecke der Arbeitskontrolle von Mitarbeitern, ist nicht zulässig.

Motivation: Für die Analyse von Sicherheitsvorfällen ist es sehr wichtig grundlegende Informationen darüber zu besitzen, wie ein Angriff erfolgt ist. Da ein Webserver eine Außenschnittstelle darstellt liegen gewisse Informationen auch bei dem Angriff auf ein nachgelagertes System lediglich auf dem Webserver vor und müssen dort daher zwingend protokolliert werden.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Abstreiten von durchgeführten Aktionen
- Unbemerkt durchführbare Angriffe

ID: 3.03-28/8.0

Req 32 Für sicherheitsrelevante Protokollierungsdaten, die lokal erfasst werden, müssen geltende Aufbewahrungszeiträume und Löschfristen eingehalten werden.

Aus Sicht der IT-Sicherheit wird auf einem System die lokale Erfassung von sicherheitsrelevanten Protokollierungsdaten nicht zwingend gefordert. Da lokal erfasste Protokollierungsdaten bei Systemstörungen beschädigt oder von einem erfolgreichen Angreifer gezielt manipuliert werden können, sind sie für sicherheitstechnische bzw. forensische Analysen nur bedingt belastbar. Entsprechend ist für die IT-Sicherheit relevant, dass Protokollierungsdaten an einen separaten Log Server weitergeleitet werden.

Eine lokale Erfassung kann dennoch erfolgen; beispielsweise falls bei der Erzeugung der Protokollierungsdaten eine lokale Speicherung aufgrund technischer Abläufe zunächst unabdingbar ist oder es begründete betriebliche Interessen gibt, Protokollierungsdaten auch lokal vorzuhalten.

Folgende Grundregelung ist bei einer lokalen Erfassung von Protokollierungsdaten zu berücksichtigen:

- Sicherheitsrelevante Protokollierungsdaten müssen für einen Zeitraum von 90 Tagen aufbewahrt werden.
(Diese Anforderung gilt nur, falls auf dem System keine zusätzliche Weiterleitung an einen separaten Log Server umgesetzt ist und Protokollierungsdaten daher ausschließlich lokal erfasst werden.)
- Nach Ablauf von 90 Tagen muss eine umgehende Löschung von gespeicherten Protokollierungsdaten erfolgen.

Abweichungen

Abweichende Aufbewahrungszeiträume und Löschfristen können durch gesetzliche oder regulatorische Anforderungen (insbesondere im Zusammenhang mit personenbezogenen Daten) bestehen oder durch vertragliche Vereinbarungen definiert sein. In diesen Fällen sind die geltenden Zeiträume individuell mit einem Project Security Manager (PSM) bzw. Datenschutzberater (DSB) abzustimmen oder werden durch diesen vorgegeben.

Motivation: Protokollierungsdaten sind ein immens wichtiges Werkzeug der IT-Sicherheit zur Abwendung, Detektie-

ung und Aufklärung von Systemstörungen, Sicherheits- und Datenschutzvorfällen. Auf der anderen Seite unterliegt die Erfassung von Protokollierungsdaten wie jede andere Datenverarbeitung auch gesetzlichen und regulatorischen Auflagen. Entsprechend müssen Richtlinien eingehalten werden, die beides in Einklang bringen.

Umsetzungsbeispiel: Unter Berücksichtigung der gültigen Gesetzeslage und zutreffenden Datenschutz-Regularien werden folgende Löschrufen für lokal erfasste sicherheitsrelevante Protokollierungsdaten auf einem beispielhaften TK-System umgesetzt:

- Standard System Logs: Löschung nach spätestens 90 Tagen
- Protokollierungen von öffentlichen IP-Adressen: Löschung (oder Anonymisierung) nach spätestens 7 Tagen
- Protokollierungen der Vergabe von dynamischen öffentlichen IP-Adressen durch die TK-Lösung: Löschung nach spätestens 7 Tagen
- Protokollierungen nicht abrechnungsrelevanter Call Detail Records: Löschung nach spätestens 7 Tagen
- Protokollierungen der Kommunikations-Inhalte von E-Mail und SMS: Löschung nach spätestens 24 Stunden
- Protokollierungen der behandelten Domain-Abfragen des DNS-Servers der TK-Lösung: Löschung nach spätestens 24 Stunden

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Abstreiten von durchgeführten Aktionen
- Unbemerkt durchführbare Angriffe

ID: 3.01-34/9.0

Req 33	Sicherheitsrelevante Protokollierungsdaten müssen umgehend nach deren Erzeugung an einen Log-Server weitergeleitet werden.
--------	--

Die sicherheitsrelevanten Protokollierungsdaten des Systems müssen umgehend nach deren Erzeugung an einen Log-Server weitergeleitet werden.

Im Sinne der Anforderung muss der dafür genutzte Log-Server die folgenden Kriterien erfüllen:

- Der Log-Server muss dediziert für die Vorhaltung von Protokollierungsdaten eingesetzt sein. Er darf keine weiteren Funktionen oder Dienste abbilden, die nicht in Relation zu dieser Aufgabe stehen. Eine zulässige Ausnahme ist die Kombination eines Log Servers mit einer SIEM (Security Information and Event Management) Lösung.
- Der Log-Server muss vom System separiert sein. Er darf nicht als zusätzliches Modul unmittelbar mit auf dem System betrieben werden.
- Der Log-Server muss zentral für die Einsatzumgebung bereitstehen, zu der auch das System gehört.

Zur Übertragung der Protokollierungsdaten an den Log-Server sollten bevorzugt auf diesen Zweck spezialisierte Standard-Protokolle (z. B. Syslog oder SNMPv3) zum Einsatz kommen.

Motivation: Werden Protokollierungsdaten nur lokal gespeichert, können diese im Fall einer erfolgreichen Kompromittierung des Systems durch einen Angreifer manipuliert werden, um dessen Angriff und auf dem System durchgeführte Aktionen zu verschleiern. Aus diesem Grund muss die Weiterleitung von Protokollierungsdaten an einen Log-Server sehr zeitnah nach dem Auftreten eines sicherheitsrelevanten Ereignisses im Quell-System erfolgen. So stehen die angefallenen Protokollierungsdaten separiert vom Quell-System und unabhängig von dessen Zustand jederzeit für Analysen bereit. Die Verwendung eines zentralen Log-Servers für eine Einsatzumgebung stellt sicher, dass auf die Logs sämtlicher IT/NT-Systeme dieser Einsatzumgebung schnell und einheitlich zugegriffen und der Zugriff selbst optimal in die Incident-Prozesse der Umgebung integriert werden kann.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigte Modifikation von Daten
- Störung der Verfügbarkeit
- Abstreiten von durchgeführten Aktionen

- Unbemerkt durchführbare Angriffe

ID: 3.01-35/9.0

Req 34	Für sicherheitsrelevante Protokollierungsdaten, die an den Log Server weitergeleitet werden, muss die Einhaltung geltender Aufbewahrungszeiträume und Löschfristen sichergestellt werden.
--------	---

Folgende Grundregelung ist zu berücksichtigen:

- sicherheitsrelevante Protokollierungsdaten müssen auf dem Log Server für einen Zeitraum von 90 Tagen aufbewahrt werden.
- nach Ablauf von 90 Tagen muss eine umgehende Löschung der auf dem Log Server gespeicherten Protokollierungsdaten erfolgen.

Abweichungen

Abweichende Aufbewahrungszeiträume und Löschfristen können durch gesetzliche oder regulatorische Anforderungen bestehen (insbesondere im Zusammenhang mit personenbezogenen Daten) oder durch vertragliche Vereinbarungen definiert sein. In diesen Fällen sind die geltenden Zeiträume individuell mit einem Project Security Manager (PSM) bzw. Datenschutzberater (DSB) abzustimmen oder werden durch diesen vorgegeben.

Log Server in Verantwortung eines Dritten

Sollte der ausgewählte Log Server nicht innerhalb derselben operativen Verantwortung liegen wie das Quellsystem der Protokollierungsdaten, muss sichergestellt werden, dass dem verantwortlichen Betreiber des Log Servers die gültigen Parameter für die zu empfangenden Protokollierungsdaten bekannt sind und durch diesen im Sinne der hier genannten Regelungen eingehalten werden.

Motivation: Protokollierungsdaten sind ein immens wichtiges Werkzeug der IT-Sicherheit zur Abwendung, Detektierung und Aufklärung von Systemstörungen, Sicherheits- und Datenschutzvorfällen. Auf der anderen Seite unterliegt die Erfassung von Protokollierungsdaten wie jede andere Datenverarbeitung auch gesetzlichen und regulatorischen Auflagen. Entsprechend müssen Richtlinien eingehalten werden, die beides in Einklang bringen.

Umsetzungsbeispiel: Unter Berücksichtigung der gültigen Gesetzeslage und zutreffenden Datenschutz-Regularien werden folgende Löschfristen für ausgeleitete sicherheitsrelevante Protokollierungsdaten eines beispielhaften TK-Systems auf dem separaten Log Server umgesetzt:

- Standard System Logs: Löschung nach spätestens 90 Tagen
- Protokollierungen von öffentlichen IP-Adressen: Löschung (oder Anonymisierung) nach spätestens 7 Tagen
- Protokollierungen der Vergabe von dynamischen öffentlichen IP-Adressen durch die TK-Lösung: Löschung nach spätestens 7 Tagen
- Protokollierungen nicht abrechnungsrelevanter Call Detail Records: Löschung nach spätestens 7 Tagen
- Protokollierungen der Kommunikations-Inhalte von E-Mail und SMS: Löschung nach spätestens 24 Stunden
- Protokollierungen der behandelten Domain-Abfragen des DNS-Servers der TK-Lösung: Löschung spätestens nach 24 Stunden

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Abstreiten von durchgeführten Aktionen
- Unbemerkt durchführbare Angriffe

ID: 3.01-36/9.0

Req 35	Sicherheitsrelevante Protokollierungsdaten müssen echtzeitnah an eine Erkennungs- und Reaktionslösung für Sicherheitsvorfälle bereitgestellt werden.
--------	--

Eine zulässige Erkennungs- und Reaktionslösung im Sinne dieser Anforderung muss die folgenden Kriterien erfüllen:

- Es erfolgt eine kontinuierliche Analyse bzw. Korrelation der aus angebundenen IT/NT-Systemen bereitgestellten Protokollierungsdaten.
- Die Identifizierung bedrohlicher Aktivitäten und laufender Angriffe findet auf Basis einer fortlaufend aktuell gehaltenen Erkennungsmuster-Datenbank statt.
- Integrierte Alarmierungsprozesse lösen im Fall eines Fundes die Einleitung von Gegenmaßnahmen aus.
- Sicherheitsexperten sind aktiv involviert in den Betrieb der Lösung sowie die Auswertung von Alarmen.

Eine solche Lösung ist in ihrer Funktion nicht mit einem Log Server gleichzusetzen. Entsprechend muss die Bereitstellung an eine Erkennungs- und Reaktionslösung stets zusätzlich zu einer vorhandenen Weiterleitung an einen Log Server erfolgen. Eine Ausnahme besteht, falls der genutzte Log Server seinerseits die Protokollierungsdaten umgehend nach Erhalt zusätzlich an eine Erkennungs- und Reaktionslösung bereitstellt. In diesem Fall ist es ausreichend, dass das System die Protokollierungsdaten nur an den Log Server weiterleitet.

Die Protokollierungsdaten müssen ab dem Zeitpunkt ihrer Erzeugung im System innerhalb von 5 Minuten an die Erkennungs- und Reaktionslösung bereitgestellt werden ("echtzeitnah").

Die konkret zu unterstützenden technischen Schnittstellen und Protokolle für die Bereitstellung werden durch die im Einsatzumfeld des Systems verfügbare Erkennungs- und Reaktionslösung sowie die technische Basis des Systems bestimmt.

Motivation: IT/NT-Systeme sind permanent sicherheitstechnischen Bedrohungen und Angriffen ausgesetzt. Mittels einer zentralen Analyse und der Korrelation erfasster System-Ereignisse ganzer IT-Landschaften in nahezu Echtzeit können aktuelle Bedrohungen und laufende Angriffe auch über Systemgrenzen hinaus erkannt und zeitnah passende Gegenmaßnahmen eingeleitet werden. Die unverzügerte Bereitstellung von Protokollierungsdaten an die Erkennungs- und Reaktionslösung ist essenziell für deren Effektivität.

Umsetzungsbeispiel: **Beispiel 1:** Ein System auf Basis eines klassischen Servers erfasst sicherheitsrelevante Protokollierungsdaten im lokalen Syslog oder Windows Event Log des Betriebssystems. Die Protokollierungsdaten werden kontinuierlich zur weiteren Analyse, Angriffserkennung und Alarmbehandlung an ein SIEM (Security Information and Event Management) weitergeleitet.

Beispiel 2: Auf einem System wird ein EDR ("Endpoint Detection and Response") Agent ausgeführt. Der EDR-Agent leitet die von ihm auf dem System erfassten sicherheitsrelevanten Protokollierungsdaten kontinuierlich an das zentrale Backend der EDR-Lösung aus, auf dem die weitere Analyse, Angriffserkennung und Alarmbehandlung erfolgt.

Beispiel 3: Netzwerk-Komponenten stellen Ereignisse an ein SIEM bereit, die es ermöglichen, verdächtiges Kommunikationsverhalten wie Netzwerk-Scans, unzulässige Datenflüsse zwischen Netz-Segmenten oder aktive Verbindungen bzw. Verbindungsversuche zu verdächtigen Zielen im Internet zu entdecken. Im speziellen liefern Gateway Router aus dem CNDTAG "Netflow" Logs, während Forward Proxies Zugriffe auf Domänen und/oder IP-Adressen melden und Firewalls blockierte Verbindungsversuche loggen.

Beispiel 4: Auf Basis von Cloud-nativen Funktionen entwickelte Systeme erzeugen Protokollierungsdaten mittels der Cloud-eigenen Protokollierungs- und Überwachungsinfrastruktur direkt aus diesen Funktionen heraus. In einem solchen System wird die Erzeugung sicherheitsrelevanter Protokollierungsdaten in den entsprechenden Konfigurationsoptionen der eingesetzten Cloud-nativen Funktionen gezielt aktiviert. Die entstehenden Protokollierungsdaten werden kontinuierlich zur weiteren Analyse, Angriffserkennung und Alarmbehandlung an ein SIEM (Security Information and Event Management) übertragen. Einige Cloud-Anbieter leisten eine gewisse Angriffsmustererkennung bereits in der Cloud-eigenen Überwachungsinfrastruktur und erzeugen daraus spezifische Cloud-Alarme. Diese werden zusätzlich an das SIEM übertragen.

Beispiel 5: Ein System basiert auf einer SaaS ("Software as a Service") Applikation eines Cloud-Anbieters. Typischerweise verantwortet im SaaS Modell der Cloud-Anbieter alle operativen Aspekte der Applikation, inkl. des Sicherheitsmanagements. Der Cloud-Kunde selbst hat keinen Einfluss auf die zugrunde liegende technische Infrastruktur oder den Betrieb der Applikation. Anhand von Vertragsvereinbarungen, anerkannten Zertifikaten und dokumentierten Selbstauskünften des Cloud-Anbieters wird formell überprüft, dass der Cloud-Anbieter eine Erkennungs- und Reaktionslösung für Sicherheitsvorfälle implementiert hat. Einige Cloud-Anbieter erzeugen zusätzlich dedizierte Protokollierungsdaten über sicherheitsrelevante Aktivitäten innerhalb der separaten Kunden-Mandanten ihrer SaaS Applikation (beispielsweise zu Änderungen von Konfigurationseinstellungen, Berechtigungsvergaben oder Benutzer-Konten innerhalb der Mandanten) und stellen diese den Cloud-Kunden zu Verfügung. Diese sogenannten Audit-Logs werden kontinuierlich zur weiteren Analyse, Angriffserkennung und Alarmbehandlung an ein SIEM (Security Information and

Event Management) übertragen.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unbemerkt durchführbare Angriffe

ID: 3.01-37/9.0