|  |  |
| --- | --- |
| Version: | 1.0 |
| Status: | Freigegeben |
| Dokumentenklassifizierung: | intern |

Sicherheitsrichtlinie "Webanwendungen"

1. Januar 2020

Allgemeine Informationen zum vorliegenden Dokument

|  |  |  |
| --- | --- | --- |
| Bezeichnung | Inhalt | Bearbeitungshinweis |
| Eigentümer |  | [verantwortlich für die Erstellung und Pflege des Dokuments = Abteilungsleitung] |
| Autor |  | [operative Verantwortung für das Dokument] |
| Status | Freigegeben | [Einstufung des aktuellen Dokumentenstatus <Entwurf, Finaler Entwurf, Final/Freigegeben>] |
| Klassifizierung | intern | [Einstufung der Dokumentenvertraulichkeitoffen, intern, vertraulich, streng vertraulich] |
| Dokumen­tenkennung | ISMS300049 | [Die Dokumenten-Kennung wird von der Dokumentenlenkung vergeben] |
| Name des Dokuments | Sicherheitsrichtlinie "Webanwendungen" | [Bezeichnung des Dokuments wie auf dem Titelblatt beschrieben.] |
| Version  | 1.0 | [zweistellige Versionsnummer] |
| Veröffentlichungsform | digital | [Veröffentlichungsform Papier, digital] |
| Speicherort |  | [Ablageort des Dokumentes] |
| Freigabe am | <TT.MM.YYYY> | [Datum der Freigabe durch den Eigentümer] |
| Freigabe bis | <TT.MM.YYYY> | [Datum der Freigabe bis durch den Eigentümer] |
| Revisionszyklus | Alle zwei Jahre | [Revisionszyklus alle 1, 2 Jahre] |
| Archivierungszeitraum | 10 Jahre | [Archivierungszeitraum nach Ablauf 5, 10 Jahre] |

# Dokumentenhistorie

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Beschreibung | Autor | Datum |
| 0.1 | initiale Erstellung |  |  |
| 0.2 – 0.8 | draft |  |  |
| 0.9 | final draft |  |  |
| 1.0 | final/freigegeben |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Inhaltsverzeichnis

[Allgemeine Informationen zum vorliegenden Dokument 2](#_Toc82449734)

[Dokumentenhistorie 3](#_Toc82449735)

[Inhaltsverzeichnis 4](#_Toc82449736)

[Allgemeine Festlegungen 6](#_Toc82449737)

[Ziel / Zweck 6](#_Toc82449738)

[Geltungsbereich 6](#_Toc82449739)

[Zuständigkeiten 6](#_Toc82449740)

[Genehmigungs- und Änderungsverfahren 7](#_Toc82449741)

[Aufbau des Dokuments 7](#_Toc82449742)

[Sicherheitsrichtlinie „Webanwendungen" 8](#_Toc82449743)

[Basismaßnahmen 8](#_Toc82449744)

[Authentisierung bei Webanwendungen (APP.3.1.A1) 8](#_Toc82449745)

[Zugriffskontrolle bei Webanwendungen (APP.3.1.A2) 8](#_Toc82449746)

[Sicheres Session-Management (APP.3.1.A3) 9](#_Toc82449747)

[Kontrolliertes Einbinden von Daten und Inhalten bei Webanwendungen (APP.3.1.A4) 9](#_Toc82449748)

[Protokollierung sicherheitsrelevanter Ereignisse von Webanwendungen (APP.3.1.A5) 9](#_Toc82449749)

[Zeitnahes Einspielen sicherheitsrelevanter Patches und Updates (APP.3.1.A6) 9](#_Toc82449750)

[Schutz vor unerlaubter automatisierter Nutzung von Webanwendungen (APP.3.1.A7) 10](#_Toc82449751)

[Standardmaßnahmen 10](#_Toc82449752)

[Systemarchitektur einer Webanwendung (APP.3.1.A8) 10](#_Toc82449753)

[Beschaffung, Entwicklung und Erweiterung von Webanwendungen (APP.3.1.A9) 10](#_Toc82449754)

[Test und Freigabe von Webanwendungen (APP.3.1.A10) 11](#_Toc82449755)

[Sichere Anbindung von Hintergrundsystemen (APP.3.1.A11) 11](#_Toc82449756)

[Sichere Konfiguration von Webanwendungen (APP.3.1.A12) 11](#_Toc82449757)

[Restriktive Herausgabe sicherheitsrelevanter Informationen (APP.3.1.A13) 12](#_Toc82449758)

[Schutz vertraulicher Daten (AP.3.1.A14) 12](#_Toc82449759)

[Verifikation essenzieller Änderungen (APP.3.1.A15) 12](#_Toc82449760)

[Umfassende Ein- und Ausgabevalidierung (APP.3.1.A16) 12](#_Toc82449761)

[Fehlerbehandlung (APP.3.1.A17) 13](#_Toc82449762)

[Kontrolle der Protokolldateien (APP.3.1.A18) 13](#_Toc82449763)

[Schutz vor SQL-Injection (APP.3.1.A19) 13](#_Toc82449764)

[Sichere HTTP-Konfiguration bei Webanwendungen (APP.3.1.A21) 13](#_Toc82449765)

[Überprüfung von Webanwendungen (APP.3.1.A22) 14](#_Toc82449766)

[Verhinderung von Cross-Site Request Forgery (APP.3.1.A23) 14](#_Toc82449767)

[Maßnahmen bei erhöhtem Schutzbedarf 14](#_Toc82449768)

[Einsatz von Web Application Firewalls (APP.3.1.A20 - CIA) 15](#_Toc82449769)

[Verhinderung der Blockade von Ressourcen (APP.3.1.A24 - A) 15](#_Toc82449770)

# Allgemeine Festlegungen

## Ziel / Zweck

Während Webserver die Webseiten ausliefern, stellen Webanwendungen Funktionen zur Verfügung und bereiten dynamische Inhalte vor, die durch den Webserver ausgeliefert werden. Um eine Webanwendung zu betreiben, sind in der Regel mehrere IT-Systemkomponenten notwendig. Hierzu gehören üblicherweise ein Webserver, um Daten auszuliefern, ein Applikationsserver, um die eigentliche Anwendung zu betreiben, und zusätzliche Hintergrundsysteme, die als Datenquellen über unterschiedliche Schnittstellen angebunden sind (z. B. Datenbank oder Verzeichnisdienst). In dieser Sicherheitsrichtlinie werden nicht Vorgaben für die redaktionelle Planung des Webauftritts sowie das Notfallmanagement sowie die sicherheitsrelevanten Aspekte einer serviceorientierten Architektur (SOA) betrachtet.

Ziel dieser Sicherheitsrichtlinie ist der sichere Betrieb von Webanwendungen für die <Institution> sowie der Schutz von Informationen, die durch eine Webanwendung verarbeitet werden. Bei der Erstellung dieser Sicherheitsrichtlinie wurden die Vorgaben des BSI Bausteines APP.3.1 „Webanwendungen" beachtet.

## Geltungsbereich

Die Vorgaben des Dokumentes sind für alle Prozessverantwortlichen der <Institution> verbindlich und entsprechend durch die zuständigen Rollenträger umzusetzen.

Anzuwenden sind die Vorgaben für alle durch die <Institution> verantworteten Geschäftsprozesse, Hard- und Softwarekomponenten sowie ihren Konfigurationen. Die Umsetzung dieser Arbeitsanweisung ist durch die entsprechenden Führungskräfte sicherzustellen.

Die im Folgenden beschriebenen Vorgaben sind hingegen nicht bindend für Prozessverantwortliche von Geschäftsprozessen, die nicht durch die <Institution> wahrgenommen werden. In diesen Fällen besitzen die beschriebenen Vorgaben einen empfehlenden Charakter, auf eine Einhaltung muss durch die <Institution> hingewirkt werden.

Interne Regelungen sind geschlechterneutral zu formulieren. Aus Gründen der besseren Lesbarkeit wird auf die gleichzeitige Verwendung männlicher und weiblicher Sprachformen verzichtet. Sämtliche personenbezogenen Bezeichnungen in männlicher Form werden verallgemeinernd verwendet und beziehen sich stets auf alle Geschlechter.

## Zuständigkeiten

Zuständig für die Einhaltung der in diesem Dokument aufgeführten Pflichten und Anforderungen sind:

* Eigene Mitarbeitende und beauftragte Dienstleister, welche administrative Arbeiten an IT- Systemen und Anwendungen von der <Institution> durchführen,
* Eigene Mitarbeitende und beauftragte Dienstleister, welche Applikationsbetreuung mit administrativem Charakter (z. B. Versionspflege, Benutzerverwaltung) betreiben. Die Kontrolle der korrekten Umsetzung der Vorgaben erfolgt durch den <Bereich ???> bei der <Institution>.

## Genehmigungs- und Änderungsverfahren

Die Sicherheitsrichtlinie „Webanwendungen“ wird durch den <Informationssicherheitsbeauftragter> verantwortet. Die Pflege dieses Dokuments unterliegt dem <Bereich ???> vertreten durch den <Informationssicherheitsbeauftragter>. Änderungen werden ausschließlich von dieser Person oder seinem Stellvertreter vorgenommen. Eine Genehmigung und Freigabe erfolgt durch den <Informationssicherheitsbeauftragter>.

## Aufbau des Dokuments

Das vorliegende Dokument ist wie folgt aufgebaut:

* Kapitel Basismaßnahmen: Beschreibung der Kernmaßnahmen, die für das Anforderungsmanagement zwingend erforderlich sind.
* Kapitel Standardmaßnahmen: Definition von Maßnahmen zur Erreichung eines vollumfänglichen Standardabsicherungsschutzniveaus für einen Schutzbedarf von „Normal“ in den Informationssicherheitsschutzzielen Vertraulichkeit, Integrität und Verfügbarkeit.
* Kapitel Maßnahmen bei erhöhtem Schutzbedarf: Erläuterung von Maßnahmen die einen erhöhten Schutzbedarf (Schutzbedarfe „Hoch“, „Sehr hoch“) gewährleisten. Der Einsatz ist je Anwendungsfall im Rahmen einer Verhältnismäßigkeitsprüfung abzuwägen.

# Sicherheitsrichtlinie „Webanwendungen"

## Basismaßnahmen

Die nachfolgenden Basismaßnahmen sind vorrangig zur Gewährleistung der sicherheitstechnischen Anforderungen aus der Leitlinie umzusetzen.

### Authentisierung bei Webanwendungen (APP.3.1.A1)

Um auf geschützte Ressourcen zugreifen zu können, sind alle Benutzer verpflichtet sich zu authentisieren. Die Authentisierungsmethode und der Auswahlprozess wurden hierfür dokumentiert. Bei dem Einsatz der Digest-Authentisierungsmethode, werden die Passwortdateien auf dem Webserver ausreichend geschützt. Wird Basic-Auth verwendet, wird eine Transportverschlüsselung mittels TLS mindestens in Version 1.2 besser in Version 1.3 eingesetzt. Es sollte eine zentrale Authentisierungskomponente verwendet werden. Diese wird mit den etablierten Standardkomponenten umgesetzt. Die Authentisierungskomponente zwingt die Benutzer der <Institution> dazu, sichere Passwörter gemäß der Passwort-Richtlinie zu benutzen. Speichert eine Webanwendung Authentisierungsdaten auf einem Client, stimmt der Benutzer explizit zu ("Opt-In") und wird auf die Risiken der Funktion hingewiesen. Sofern Personen bezogene Daten in der Webanwendung verarbeitet werden, wird zum Nachweis der Benutzerintegrität eine Authentisierung mittels Multi-Faktor-Authentifizierung (MFA) sichergestellt.

Um sicherzugehen, dass eine gültige Sitzung (Session-ID) nicht von einem Angreifer übernommen wurde, authentisieren sich bei kritischen Funktionen die Benutzer erneut. Zusätzlich wurden in der Webanwendung Grenzwerte für fehlgeschlagene Anmeldeversuche definiert. Alle angebotenen Authentisierungsverfahren der Webanwendung haben das gleiche Sicherheitsniveau. Darüber hinaus werden die Benutzer sofort informiert, wenn das Passwort zurückgesetzt wurde.

### Zugriffskontrolle bei Webanwendungen (APP.3.1.A2)

Darf nur ein beschränkter Benutzerkreis die Webanwendung nutzen, wird mittels einer Autorisierungskomponente sichergestellt, dass Benutzer nur für sie berechtigte Aktionen ausführen dürfen. Jeder Zugriff auf geschützte Inhalte und Funktionen wird kontrolliert, bevor er ausgeführt wird.

Allen Benutzern werden restriktive Zugriffsrechte nach dem „Need-to-know-Prinzip“ zugewiesen. Jegliche Änderungen der Zugriffsrechte werden durch die Verantwortlichen für den Betrieb und das Verfahren geprüft, bestätigt und dokumentiert. Die Dokumentationen werden immer aktuell gehalten. Es wurde ein Verfahren zum Entziehen der Zugriffsrechte etabliert. Sollte es nicht möglich sein Rechte zuzuweisen, werden zusätzliche Sicherheitsprodukte eingesetzt.

Es werden alle von der Webanwendung verwalteten Ressourcen von der Autorisierungskomponente berücksichtigt. Die Benutzer autorisieren sich serverseitig und zentral auf einem vertrauenswürdigen IT-System. Bei fehlerhafter Zugriffskontrolle werden Zugriffe abgelehnt. Es findet ebenfalls eine Zugriffskontrolle bei URL-Aufrufen und Objekt-Referenzen statt. Der Zugriff auf Dateien durch die Benutzer mit restriktiven Dateisystemberechtigungen werden beschränkt. Ein sicherer Umgang mit temporären Dateien ist vorgesehen.

### Sicheres Session-Management (APP.3.1.A3)

Die Session-IDs werden angemessen geschützt. Diese werden mit ausreichender Entropie zufällig erzeugt. Kann das Framework der Webanwendung Session-IDs generieren, wird die Funktion des Frameworks genutzt. Das Framework wird sicher konfiguriert. Bei einer Übertragung und clientseitiger Speicherung wird die Session-ID ebenfalls ausreichend geschützt.

Den Benutzern wird die Möglichkeit gegeben eine bestehende Sitzung explizit zu beenden. Bei einer Anmeldung wird die bereits bestehende Session-ID durch eine neue ersetzt. Die Sitzungsdauer wird beschränkt. Nach dem Ablauf einer Sitzung werden alle Sitzungsdaten sowohl server- als auch clientseitig ungültig und gelöscht.

### Kontrolliertes Einbinden von Daten und Inhalten bei Webanwendungen (APP.3.1.A4)

Es wird sichergestellt, dass eine Webanwendung ausschließlich vorgesehene Daten und Inhalte einbindet und an den Benutzer ausliefert. Bietet eine Webanwendung eine Upload-Funktion für Dateien an, wird diese eingeschränkt. Zugriffs- und Ausführungsrechte werden restriktiv vergeben. Zusätzlich wird sichergestellt, dass die Benutzer nur im vorgegebenen Pfad speichern können. Der Ablageort der Uploads ist durch den Benutzer nicht beeinflussbar.

Die Weiterleitungsfunktionen von Webanwendungen werden so eingeschränkt, dass Benutzer ausschließlich auf vertrauenswürdige Webseiten weitergeleitet werden. Wird die Vertrauensdomäne verlassen, wird der Benutzer umgehend informiert.

### Protokollierung sicherheitsrelevanter Ereignisse von Webanwendungen (APP.3.1.A5)

Die sicherheitsrelevanten Ereignisse werden mit den erforderlichen Merkmalen nachvollziehbar protokolliert. Der Zugriff auf die Protokolldaten ist nur auf wenige Mitarbeitende beschränkt. Bei der Auswertung der Protokolldaten wird sichergestellt, dass Schadcode in Protokoll-Einträgen vom Auswertungsprogramm nicht interpretiert wird.

### Zeitnahes Einspielen sicherheitsrelevanter Patches und Updates (APP.3.1.A6)

Die verantwortlichen Administratoren bzw. Entwickler sind verpflichtet sich regelmäßig über aktuelle Schwachstellen zu informieren und sicherheitsrelevante Updates zeitnah einzuspielen. Die Updates werden nur aus vertrauenswürdigen Quellen bezogen und werden vor dem Roll-Out ausreichend getestet. Vor dem Einspielen von Updates oder Patches wird sichergestellt, dass der ursprüngliche Zustand der Webanwendung wiederhergestellt werden kann. Der aktuelle Patch-Level wird dokumentiert.

### Schutz vor unerlaubter automatisierter Nutzung von Webanwendungen (APP.3.1.A7)

Durch geeignete Schutzmechanismen werden die Webanwendungen vor automatisierten Zugriffen geschützt. Hierfür wurden Grenzwerte definiert. Die Auswirkungen der Grenzwerte auf die Webanwendungen wurden hierbei berücksichtigt.

## Standardmaßnahmen

Gemeinsam mit den Basismaßnahmen sind die folgenden Standardmaßnahmen zum Erzielen eines normalen Schutzbedarfs zu betrachten und sollten grundsätzlich umgesetzt werden.

### Systemarchitektur einer Webanwendung (APP.3.1.A8)

Die Sicherheitsaspekte einer Webanwendung werden bereits in der Entwurfsphase betrachtet. Ebenfalls wird darauf geachtet, dass die Architektur der Webanwendung die Geschäftslogik der <Institution> exakt erfasst und korrekt umsetzt.

Es ist vorgesehen die Serverdienste durch jeweils separate IT-Systeme voneinander zu trennen. Zusätzlich sind jeweils eigene Benutzerkonten für die unterschiedlichen Prozesse der Systemkomponenten zu verwenden. Die Rechte dieser Dienstekonten werden auf Systemebene soweit eingeschränkt, dass nur auf erforderliche Ressourcen und Dateien des Betriebssystems zugegriffen werden kann.

Die Netzarchitektur verfolgt einen mehrschichtigen Ansatz (Multi-Tier-Architektur). Dabei sollten mindestens die Sicherheitszonen Webschicht, Anwendungsschicht und Datenschicht berücksichtigt werden. Aus diesen Zonen darf nicht auf Systeme im Intranet (Office-IT) zugegriffen werden können.

Die Softwarearchitektur wird mit allen Bestandteilen und Abhängigkeiten dokumentiert. Die Dokumentation wird bereits während des Projektverlaufs so aktualisiert und angepasst, dass sie schon in der Entwicklungsphase benutzt werden kann und Entscheidungsfindungen nachvollziehbar sind. Alle für den Betrieb notwendigen Komponenten, die nicht Bestandteil der Webanwendung sind, werden als solche innerhalb der Dokumentation gekennzeichnet. Die Dokumentation beinhaltet auch Informationen dazu, welche Komponenten welche Sicherheitsmechanismen umsetzen, wie die Webanwendung in eine bestehende Infrastruktur integriert wird und welche kryptografischen Funktionen und Verfahren eingesetzt werden.

### Beschaffung, Entwicklung und Erweiterung von Webanwendungen (APP.3.1.A9)

Für die Beschaffung von Produkten für eine Webanwendung sollte ein Anforderungskatalog erstellt werden. Die verschiedenen Produkte sollten anhand einer innerhalb der <Institution> entwickelten Bewertungsskala verglichen.

Findet eine Eigenentwicklung einer Webanwendung oder Erweiterung hierzu statt, wird nach einem geeigneten Vorgehensmodell vorgegangen. Vor der Inbetriebnahme werden alle Phasen des Modells durchlaufen. Den Entwicklern werden zudem Programmierrichtlinien der <Institution> vorgegeben, um ein einheitliches Usability- und Sicherheitsniveau zu etablieren.

Die Sicherheitsmechanismen der Webanwendungen werden so entworfen und entwickelt, dass diese möglichst auch zukünftigen Standards und Angriffstechniken berücksichtigen. Bei der Anwendungsentwicklung sind mindestens die Entwicklungs- und Testumgebung von der Umgebung der Produktivsysteme getrennt.

Falls für die Entwicklung der Webanwendung auch auf Dienstleister zurückgegriffen wird, ist sicherzustellen, dass dieser Dienstleister die nötigen Sicherheitsanforderungen bei der Entwicklung umsetzt und die <Institution> jederzeit auf den Quelltext zugreifen kann.

### Test und Freigabe von Webanwendungen (APP.3.1.A10)

Alle Anwendungen und Erweiterungen werden vor der Übernahme in den produktiven Betrieb getestet. Die Ergebnisse des Softwaretests werden dokumentiert. Nach erfolgreichen Tests werden die Anwendungen formal freigegeben. Werden Fehler festgestellt, wird das etablierte Verfahren zur Fehlerbehebung eingeleitet.

### Sichere Anbindung von Hintergrundsystemen (APP.3.1.A11)

Alle Hintergrundsysteme der Webanwendungen werden entsprechend dem Schutzbedarf der verarbeiteten Informationen ausreichend geschützt. Der Zugriff ist ausschließlich über definierte Schnittstellen sowie von definierten Systemen möglich und erfolgt nur mit minimalen Rechten. Der Datenverkehr zwischen den Benutzern und der Webanwendung bzw. den Anwendungen und weiteren Diensten sowie den Hintergrundsystemen wird durch Sicherheitsgateways geregelt. Der Datenverkehr (Data in transit) wird zusätzlich verschlüsselt.

Wird ein Enterprise Service Bus (ESB) eingesetzt, wird sichergestellt, dass sich alle Dienste gegenüber dem ESB authentisieren, bevor ihnen ein Zugriff erlaubt wird. Es können nur angeschlossene Anwendungen und Dienste auf den ESB zugreifen. Alle Zugriffe auf den ESB werden authentisiert und bei der Kommunikation über Standort- und Netzgrenzen hinweg verschlüsselt. Ein logisches Netzsegment für den ESB ist vorhanden.

### Sichere Konfiguration von Webanwendungen (APP.3.1.A12)

Alle Webanwendungen werden so konfiguriert, dass auf ihre Ressourcen und Funktionen ausschließlich über die vorgesehenen, abgesicherten Kommunikationspfade zugegriffen werden kann. Der Zugriff auf nicht benötigte Ressourcen und Funktionen wird daher eingeschränkt. Aus Security-Sicht fehlerhafte Konfigurationen finden sich schon seit langem in den OWASP-Top-Ten ([Open Web Application Security Project](https://www.owasp.org/index.php/Category%3AOWASP_Top_Ten_Project)). Die aktuelle OWASP-Top-Ten sollte als Grundlage für eine sichere Konfiguration herangezogen werden. Bei der Konfiguration von Webanwendungen muss mindestens Folgendes berücksichtigt werden:

* Deaktivierung nicht benötigter HTTP-Methoden,
* Zeichencodierungskonfiguration,
* Festlegung von Grenzwerten,
* Administration einer Webanwendung.

### Restriktive Herausgabe sicherheitsrelevanter Informationen (APP.3.1.A13)

Webseiten und Rückantworten von Webanwendungen der <Institution> dürfen keine Informationen beinhalten, die einem Angreifer Hinweise geben, mit denen er etablierte Sicherheitsmechanismen umgehen kann. Hierzu gehören unter anderem:

* Neutrale Fehlermeldungen,
* Keine sicherheitsrelevanten Kommentare oder Produkt- und Versionsangaben,
* Eingeschränkter Zugriff auf sicherheitsrelevante Dokumentation,
* Regelmäßiges Löschen nicht benötigter Dateien,
* Sichere Erfassung durch externe Suchmaschinen sowie der Verzicht auf absolute Pfadangaben.

Die Webanwendungen werden ausschließlich aus sicheren Netzen administriert. Die Administrationszugänge hierauf werden auf vertrauenswürdige Netzsegmente und IT-Systeme beschränkt. Konfigurationsdateien der Webanwendung werden außerhalb des Web-Root-Verzeichnisses gespeichert.

### Schutz vertraulicher Daten (AP.3.1.A14)

Vertrauliche Daten der Webanwendungen werden durch Data-at-Rest-Verschlüsselung sowie der verschlüsselten Übertragung geschützt. Bei Verbindungsfehlern innerhalb eines verschlüsselten Kanals wird nicht auf einen unverschlüsselten Kanal gewechselt.

Durch Direktiven wird gewährleistet, dass clientseitig keine schützenswerten Daten zwischengespeichert werden. Weiterhin werden in Formularen keine vertraulichen Formulardaten im Klartext angezeigt und auch nicht vom Browser gespeichert. Zugangsdaten der Webanwendung werden serverseitig mithilfe von kryptographischen Algorithmen vor unbefugtem Zugriff geschützt (Salted Hash). Ein Abruf von Dateien mit Quelltext der Webanwendung ist nicht möglich. Auch die Konfigurationsdateien von Webanwendungen werden ausschließlich außerhalb des Web-Root-Verzeichnisses gespeichert.

### Verifikation essenzieller Änderungen (APP.3.1.A15)

Sollen wichtige Einträge geändert werden, wird die Eingabe durch ein Passwort erneut verifiziert werden. Die betroffenen Benutzer sind über die Änderungen beispielsweise per Mail außerhalb der Web-Anwendung zu informieren. Änderungen der Version der Webanwendung sollten zusätzlich zur Mail-Benachrichtigung auch den Benutzer innerhalb der Anwendung angezeigt werden.

### Umfassende Ein- und Ausgabevalidierung (APP.3.1.A16)

Alle an eine Webanwendung übergebenen Daten werden als potenziell gefährlich behandelt und entsprechend gefiltert. Hierbei werden alle Ein- und Ausgabedaten sowie Datenströme und Sekundärdaten (z. B. Session-IDs) validiert. Die Daten werden serverseitig auf einem vertrauenswürdigen IT-System geprüft. Fehleingaben werden nicht automatisch behandelt. Lässt sich die nicht-automatische Behandlung nicht vermeiden, wird Sanitizing sicher umgesetzt, damit ein Missbrauch ausgeschlossen werden kann.

Ausgabedaten werden so codiert, dass schadhafter Code auf dem Zielsystem nicht interpretiert oder ausgeführt wird.

### Fehlerbehandlung (APP.3.1.A17)

Treten während des Betriebs Fehler auf, werden diese so behandelt, dass der konsistente Zustand der Webanwendung erhalten bleibt. Folgende Punkte werden bei der Fehlerbehandlung berücksichtigt:

* Fehlermeldungen müssen protokolliert werden,
* Eine veranlasste Aktion muss im Fehlerfall abgebrochen und
* In der Folge der Zugriff auf die angeforderte Ressource oder Funktion abgewiesen werden.

Innerhalb der Fehlerbehandlung werden die zuvor reservierten Ressourcen wieder freigegeben. Die auftretenden Fehler werden nach Möglichkeit von den Webanwendungen selbst behandelt.

### Kontrolle der Protokolldateien (APP.3.1.A18)

Für die Webanwendung wird ein Konzept für die Protokollierung erstellt. Dieses beinhaltet wie umfangreich die Protokollierung ist und wie die Daten auszuwerten sind. Zusätzlich wird ein Verantwortlicher für die Auswertung benannt. Die Ergebnisse der Auswertungen werden dem <Informationssicherheitsbeauftragter> übergeben. Bestehende gesetzliche bzw. vertragliche Vorgaben in Bezug auf die Protokolldaten werden innerhalb der <Institution> eingehalten.

### Schutz vor SQL-Injection (APP.3.1.A19)

Alle Eingaben und Parameter werden von den Webanwendungen sorgfältig überprüft und gefiltert, bevor diese an das Datenbanksystem weitergeleitet werden. Ebenfalls werden Stored Procedures bzw. Prepared SQL-Statements eingesetzt. Können Prepared SQL-Statements nicht eingesetzt werden, werden die SQL-Queries gesondert abgesichert. Um potenziellen Angreifern keine Anhaltspunkte für Angriffe zu liefern, werden von Webanwendungen neutrale Fehlermeldungen nach außen ausgeben.

### Sichere HTTP-Konfiguration bei Webanwendungen (APP.3.1.A21)

Zum Schutz vor Clickjacking-Angriffen ist in den HTTP-Response-Headern der Webanwendung die Direktive X-FRAME-OPTIONS mit geeigneten Parametern zu setzen.

Darüber hinaus sollten die folgenden HTTP-Header Konfigurationsmöglichkeiten mit geeigneten Parametern hinterlegt werden:

* Strict-Transport-Security (empfohlen "Strict-Transport-Security: max-age=31536000; includeSubDomains")
* Content Security Policy (Whitelisting von zugelassenen Quellen bspw. "Content-Security-Policy: script-src 'self")
* X-Frame-Options (empfohlen "X-Frame-Options: SAMEORIGIN")
* X-Content-Type-Options (empfohlen "X-Content-Type-Options: nosniff")
* X-XSS-Protection (empfohlen "X-XSS-Protection: 1; mode=block")
* Referrer-Policy (Die Konfiguration kann mit dem Online-Tool <https://report-uri.io/home/generate> erstellt werden)
* Feature-Policy (Möglichkeiten der Konfiguration können über diesen Link <https://github.com/w3c/webappsec-feature-policy/blob/master/README.md> und <https://w3c.github.io/webappsec-feature-policy/> nachgelesen werden)
* X-Powered-By (Hier dürft ihr dann auch gerne kreativ werden)

Die vorgenommenen Einstellungen sollten mit Hilfe dieser Tools <https://securityheaders.com>, <https://webbkoll.dataskydd.net>, <https://observatory.mozilla.org/> und <https://csp-evaluator.withgoogle.com/> verifiziert werden.

### Überprüfung von Webanwendungen (APP.3.1.A22)

Es sind regelmäßig Penetrationstests durchzuführen. Diese werden ausschließlich von zuverlässigem, vertrauenswürdigem und qualifiziertem Personal oder Dienstleistern ausgeführt. Zuvor werden mit allen Auftragnehmern für Penetrationstests detaillierte Vereinbarungen zur Durchführung und Auswertung der Tests getroffen. Es wird das Einverständnis aller zuständigen Stellen eingeholt. Für den Testzeitraum stehen die jeweiligen Ansprechpartner verbindlich fest und sind auch erreichbar. Nach den Penetrationstests werden die Ergebnisse ausreichend geschützt und vertraulich behandelt. Der Abschlussbericht wird dem <Informationssicherheitsbeauftragter> und den verantwortlichen Product Ownern sowie den verantwortlichen für den späteren Betrieb vorgelegt.

### Verhinderung von Cross-Site Request Forgery (APP.3.1.A23)

Die eingesetzten Webanwendungen unterstützen Sicherheitsmechanismen, die es ermöglichen, beabsichtigte Seitenaufrufe des Benutzers von unbeabsichtigt weitergeleiteten Befehlen Dritter zu unterscheiden. Dabei wird geprüft, ob neben der Session-ID ein geheimes Token für den Zugriff auf geschützte Ressourcen und Funktionen benötigt wird. Ebenfalls wird das Referrer-Feld im HTTP-Request als zusätzliches Merkmal geprüft, um so einen beabsichtigten Aufruf durch einen Benutzer zu erkennen.

## Maßnahmen bei erhöhtem Schutzbedarf

Gemeinsam mit den Basismaßnahmen und den Standardmaßnahmen sind zum Erzielen eines erhöhten Schutzbedarfs die hier aufgeführten Maßnahmen zu betrachten und sollten grundsätzlich umgesetzt werden. Ist dies aus wirtschaftlichen bzw. organisatorischen Gründen nicht möglich, so ist dies mit dem Sicherheitsmanagement zur weiteren Begegnung von Risiken für die Infrastruktur der <Institution> zu begründen und abzustimmen. Im Folgenden werden die Maßnahmen bei erhöhtem Schutzbedarf aufgeführt. Die jeweils in Klammern angegebenen Buchstaben zeigen an, welche Grundwerte durch die Anforderung vorrangig geschützt werden (C = Vertraulichkeit, I = Integrität, A = Verfügbarkeit).

### Einsatz von Web Application Firewalls (APP.3.1.A20 - CIA)

Innerhalb der <Institution> werden Web Application Firewalls (WAF) eingesetzt. Die Konfiguration ist auf die zu schützende Webanwendung angepasst. Nach jedem Update der Webanwendung findet eine Überprüfung der Konfiguration statt.

### Verhinderung der Blockade von Ressourcen (APP.3.1.A24 - A)

Zum Schutz vor Denial-of-Service-(DoS)-Angriffen werden ressourcenintensive Operationen vermieden und besonders abgesichert. Ein möglicher Überlauf von Protokolldaten bei Webanwendungen wird überwacht und verhindert. SOAP-Nachrichten werden anhand eines entsprechenden XML-Schemas validiert. Bei kritischen Diensten und Anwendungen wird die Entscheidung überprüft mit Anti-DoS-Dienstleistern zusammenzuarbeiten.