Bundesamt
für Sicherheit in der
Informationstechnik

Deutschland
Digital•Sicher•BSI•

Technische Richtlinie TR-03174: Anforderungen an Anwendungen im Finanzwesen

Teil 2: Web-Anwendungen

Version 3.0

Änderungshistorie

Version	*Datum*	*Name*	*Beschreibung*
1.0	18.05.2022	Referat DI 24	Erste Version
2.0	25.03.2024	Referat DI 24	Überarbeitung Einleitung Überarbeitung Kapitel 3 Überarbeitung Kapitel 4
3.0	16.09.2024	Referat D24	Finalisierung

Bundesamt für Sicherheit in der Informationstechnik
Postfach 20 03 63
53133 Bonn
Tel.: +49 22899 9582-0
E-Mail: referat-d24@bsi.bund.de
Internet: https://www.bsi.bund.de

Inhalt

Tabellenverzeichnis

1 Einleitung

1.1 Gegenstand der Technischen Richtlinie

Nicht erst seit dem Inkrafttreten der Payment Service Directive 2 **Fehler! Verweisquelle konnte nicht g efunden werden.** ist es für Fintechs und ähnliche Akteure möglich, Dienstleistungen im Bezahl- und Paymentumfeld anzubieten. Es ist jedoch ein konkretes Ziel der PSD2/3 den Wettbewerb im Bereich der Paymentdienstleistungen europaweit zu fördern. Zu diesen Dienstleistungen zählen insbesondere die Möglichkeit, Zugriffe auf Konten oder spezielle Kontoinformationen oder die Initiierung von Zahlungen mittels digitaler Anwendungen zu ermöglichen. Mit den Regulatory Technical Standards for Strong Customer Authentication **Fehler! Verweisquelle konnte nicht gefunden werden.** gibt die EU einen g roben Rahmen vor, wie diese Dienstleistungen sicher gestaltet werden können. Dieser grobe Rahmen soll durch die Verwendung der hier vorliegenden Technischen Richtlinie konkretisiert werden.

Die TR richtet sich an Hersteller von Web-Anwendungen im Finanzwesen. Zusätzlich kann sie als Richtlinie für Web-Anwendungen betrachtet werden, welche sensible Daten verarbeiten oder speichern.

1.2 Zielsetzung der Technischen Richtlinie

Die Digitalisierung aller Lebensbereiche, sei es im Beruf, in Heimumgebungen, im Individual- oder im öffentlichen Personenverkehr, schreitet stetig voran. Bereits im Jahr 2018 überschritt die Anzahl der Internetnutzer die Grenze von vier Milliarden Menschen. Zwei Drittel der zurzeit 8,2 Milliarden Menschen zählenden Weltbevölkerung nutzen ein Mobiltelefon. Mehr als drei Milliarden Menschen nutzen soziale Netzwerke und tun dies in neun von zehn Fällen über ihr Smartphone (vgl. [GDR18]). Diese Entwicklung lässt sich auch im Finanzwesen beobachten. So überprüfen Nutzer ihren Kontostand während Sie unterwegs sind, lösen Überweisungen aus oder Zahlen am Point of Sale ganz intuitiv mit dem Smartphone oder einer Smartwatch. Dabei sind Daten zur finanziellen Situation eines Menschen besonders schützenswert. Ein kompromittiertes Smartphone kann somit das gesamte digitale Leben des Nutzers ungewollt offenlegen und zu hohem finanziellen Schaden führen. Das Einhalten von geeigneten Sicherheitsstandards, gerade im Bereich der mobilen Anwendungen, kann dies wesentlich erschweren und möglicherweise sogar verhindern. Schon während der Entwicklungsphase sollten Hersteller sehr verantwortungsvoll planen, wie eine mobile Anwendung personenbezogene und andere sensible Daten verarbeitet, speichert und schützt.

Die IT-Sicherheit verfolgt im Wesentlichen drei Schutzziele: Vertraulichkeit, Integrität und Verfügbarkeit.

Gerade bei Anwendungen im Finanzwesen ist die Einhaltung dieser Anforderungen von besonderer Wichtigkeit. Der Verlust der Vertraulichkeit von Finanzdaten kann für das Opfer negative Auswirkungen sowohl im privaten, wie auch im beruflichen Kontext haben.

Sollte ein Angreifer in der Lage sein, sensible Daten eines Dritten zu manipulieren und damit deren Integrität zu verletzen, könnte er wesentlichen Einfluss auf das Leben des Betroffenen haben. Im Bereich der Finanzanwendungen könnte die Verletzung der Integrität dazu führen, dass Zahlungen, die zwar der rechtmäßige Kontoinhaber auslöst, auf ein Konto überwiesen werden, welches jedoch nicht dem beabsichtigten Zahlungsempfänger zugeordnet ist. Daher können unsichere Finanzanwendungen großen wirtschaftlichen Schaden bei natürlichen und juristischen Personen anrichten.

Eine Verletzung der Verfügbarkeit kann bei Betroffenen dazu führen, dass notwendige Zahlungen oder Überweisungen nicht durchgeführt werden können. Dies kann dann die Grundlage für weitere Probleme sein, wenn zum Beispiel der getätigte Einkauf nicht bezahlt oder eine fällige Rechnung nicht beglichen werden kann.

Diese Technische Richtlinie soll als Leitfaden dienen, um Entwickler von Web-Anwendungen bei der Erstellung sicherer Softwarelösungen zu unterstützen. Greift die Anwendung auf Funktionalitäten eines Hintergrundsystems zurück, ist für eine vollumfängliche sicherheitstechnische Begutachtung ebenfalls die Sicherheit des Hintergrundsystems unerlässlich (vgl. Kapitel. 2.3)

1.3 Übersicht der Technischen Richtlinie

1.3.1 Methodik

Anwendungen im Sinne dieser TR sind Web-Anwendungen im Finanzwesen. Hierzu zählen insbesondere Anwendungen, die es ermöglichen, auf Konten oder Kontoinformationen zu zugreifen und/oder Zahlungen auszulösen. Der Betrieb kann autonom durch die Anwendung auf dem Endgerät oder in Kombination mit einem sicheren Hintergrundsystem umgesetzt werden. Wird im Folgenden der Begriff Hintergrundsystem verwendet, ist insbesondere auch der Einsatz von Cloud-Computing gemeint. Auf Grund des rasanten technischen Fortschritts und der Diversität von Plattformen, erhebt die Technische Richtlinie keinen Anspruch auf Vollständigkeit. Sie kann als Mindestanforderung für den sicheren Betrieb einer Web-Anwendung betrachtet werden.

Die Technische Richtlinie formuliert eine Security Problem Definition (SPD), welche potenzielle Bedrohungsszenarien aufweist. Aus der SPD werden Prüfaspekte für Web-Anwendungen und deren Plattformen bzw. Einsatzumgebungen abgeleitet, um vor diesen Bedrohungen zu schützen.

Die in dieser Technischen Richtlinie formulierten Bedrohungsszenarien und Prüfaspekte basieren auf Erfahrungen, die das BSI bei bisherigen Untersuchungen von Web-Anwendungen gesammelt hat. Darüber hinaus orientiert sie sich an internationalen Standards, wie den „Application Security Verification Standard“ [ASVS], und dem „Web Security Testing Guide“ [WSTG].

Eine Grundanforderung an Anwendungen im Sinne der Technischen Richtlinie ist die Orientierung an Best-Practice-Empfehlungen und anderen allgemeinen Anforderungen an sichere, verteilte Anwendungen. Dazu zählen die Durchführung intensiver funktionaler Tests, Integrationstests sowie insbesondere Positiv-/Negativ-Tests von Sicherheitsleistungen der Anwendung. Die TR stellt darüber hinaus zusätzliche, spezifische Anforderungen.

1.3.2 Begriffe

Diese Technische Richtlinie verwendet folgende Begriffe:

Tabelle 1: Begriffe der Technischen Richtlinie

Begriff	Beschreibung
MUSS	Die Anwendung muss eine bestimmte Eigenschaft zwingend aufweisen.
DARF NICHT / DARF KEIN(E)	Die Anwendung darf eine bestimmte Eigenschaft unter keinen Umständen aufweisen.
SOLL	Die Anwendung muss eine bestimmte Eigenschaft aufweisen, außer es wird dargelegt, dass durch ein Nicht-Umsetzen kein Risiko für den sicheren Betrieb besteht, bzw. eine Umsetzung, aufgrund von technischen Einschränkungen, derzeit nicht möglich ist.
KANN	Die Anwendung kann eine bestimmte Eigenschaft aufweisen, wobei ein Umsetzen dieser Eigenschaft vom Lösungsanbieter anzuzeigen ist.

Begriff	Beschreibung
primärer Zweck	Der primäre Zweck einer Anwendung im Sinne der Technischen Richtlinie ist ein Zweck des bestimmungsgemäßen Gebrauchs sowie alle Zwecke, die unmittelbar auf die Verankerung der Anwendung im geltenden Rechtsrahmen abzielen.
rechtmäßiger Zweck	Der rechtmäßige Zweck einer Anwendung im Sinne der Technischen Richtlinie ist ein Zweck, der durch geltendes Recht als Grundlage zur Verarbeitung personenbezogener Daten zulässig ist.

2 Überblick der Sicherheitsanforderungen an Anwendungen im Finanzwesen

2.1 Anwendungskonzepte auf mobilen Endgeräten

Der Begriff „mobile Anwendung“ bezeichnet ein Programm, das auf einer mobilen Plattform ausgeführt wird. Grundsätzlich lassen sich solche Anwendungen in drei Kategorien unterteilen. Die erste Kategorie bilden die nativen Anwendungen (Kapitel 2.1.1), welche direkt auf die Plattform, auf der sie ausgeführt werden, zugeschnitten sind, ab. Dem gegenüber stehen die Web-Anwendungen (Kapitel 2.2). Ihre Implementierung ist völlig unabhängig von der Plattform und sie laufen innerhalb des Web-Browsers des Endgeräts. In die dritte Kategorie fallen die hybriden Ansätze (Kapitel 2.1.2). Sie spiegeln alle möglichen Kombinationen aus nativen Anwendungen und Web-Anwendungen wider.

Da mobile Anwendungen weit über den Einsatz von Web-Anwendungen in mobilen Browsern oder hybriden Anwendungen hinausgehen, liegt der Fokus dieser Publikation allein auf Web-Anwendungen und dem Web-Teil von hybriden Ansätzen. Für zusätzliche Hinweise zur sicheren Entwicklung und zum sicheren Betrieb von nativen-Anwendungen empfiehlt das BSI „TR-03161 Anforderungen an Anwendungen im Finanzwesen Teil 1: Mobile Anwendungen“ [TR03174-1] zu Rate zu ziehen.

2.1.1 Native-Anwendungen

Eine native Anwendung ist passend auf eine Plattform und deren Betriebssystem zugeschnitten. Sie basiert auf den von der Plattform (beispielsweise Android oder iOS) bereitgestellten Programmierwerkzeugen (Software Development Kits - SDKs). Diese ermöglichen einen direkten Zugriff auf Gerätekomponenten, wie beispielsweise GPS, Kamera oder Mikrofon. Aufgrund ihrer Nähe zum Betriebssystem können sie eine sehr gute Performanz, eine hohe Zuverlässigkeit und eine intuitive Bedienbarkeit erreichen. Die Anwendungen werden beispielsweise über den plattformeigenen App-Store installiert und können oft auch offline betrieben werden.

Mit der Nähe zum Betriebssystem sind allerdings auch Nachteile verbunden. Änderungen am Betriebssystem, beispielsweise durch Updates, können dazu führen, dass Anpassungen an der Anwendung vorgenommen werden müssen. Sollte dies nicht erfolgen, kann es zu Beeinträchtigungen der Funktionsfähigkeit der Anwendung kommen. Darüber hinaus ist es nicht möglich sie auf anderen Betriebssystemen zu installieren. Soll die gleiche Anwendung auf mehreren Betriebssystemen publiziert werden, so muss jeweils eine eigene Codebasis[1] existieren. Dies ist häufig mit einem hohen Aufwand und somit auch hohen Kosten verbunden.

2.1.2 Hybride Ansätze

Hybride Anwendungen verbinden sowohl die Vor-, als auch die Nachteile von nativen Anwendungen und Web-Anwendungen. Mit Hilfe des SDKs wird eine Rahmen-Anwendung geschaffen, welche alle Vor- und Nachteile von nativen Anwendungen aufweist. Sie kann auf Gerätekomponenten zugreifen und über einen App-Store bezogen werden, jedoch nicht auf anderen Plattformen installiert werden, ohne Anpassungen am Quellcode vorzunehmen. Darüber hinaus beinhalten die Rahmen-Anwendungen einen eingebetteten Web-Browser (WebView genannt), mit dessen Hilfe Web-Anwendungen in native Anwendungen eingebunden werden können. Dadurch ist es auch Web-Anwendungen möglich, auf die sonst nur den nativen Anwendungen vorbehaltenen Gerätekomponenten zuzugreifen. Allerdings müssen dabei Einbußen bei der

[1] Es existieren auch plattformübergreifende Implementierungsansätze, welche die Entwicklung einer Anwendung für verschiedene Plattformen gleichzeitig unterstützen. Allerdings verschiebt sich dadurch die Abhängigkeit lediglich in diese sehr umfangreiche Middleware, die alle Zielplattformen abdecken muss.

Performanz und Stabilität in Kauf genommen werden. Darüber hinaus kann es durch den Einsatz unterschiedlicher Benutzerschnittstellen zu einer negativen Beeinträchtigung der User-Experience kommen. Die Plattformabhängigkeit der Anwendung bezieht sich nun lediglich auf die Rahmen-Anwendung, womit der Aufwand für eine Migration auf andere Plattformen deutlich reduziert wird.

2.2 Web-Anwendungen

Web-Anwendungen sind Anwendungsprogramme, meist Webseiten, die in Kombination mit einem Hintergrundsystem (Kapitel 2.3) ohne Installation auf einem lokalen System betrieben werden können. Solche Webseiten sind oft so programmiert, dass sie wie eine native Anwendung für klassische Desktop-Systeme oder mobile Endgeräte aussehen und sich vergleichbar verhalten. Im Gegensatz zu nativen Anwendungen basieren sie nicht auf einem SDK der zugrundeliegenden Plattformen, sondern auf klassischen Programmierwerkzeugen der Web-Entwicklung. In den meisten Fällen kommen HTML5 und JavaScript zum Einsatz. Aus diesem Grund ist mit ihnen nur ein sehr eingeschränkter Zugriff auf Gerätekomponenten möglich. Ihr größter Vorteil besteht darin, dass sie unabhängig vom Betriebssystem sind. Da die Anwendungen innerhalb eines Web-Browsers laufen, können sie auf jeder Plattform gleichermaßen eingesetzt werden, ohne Anpassungen an der Codebasis vornehmen zu müssen.

2.3 Hintergrundsysteme

Die meisten Anwendungen verlassen sich für die Verarbeitung und Speicherung von Daten nicht ausschließlich auf die von der Laufzeitumgebung bereitgestellten Ressourcen. Sie lagern diese Aufgaben auf ein Server-System aus. Weil diese Server aus Nutzersicht nicht sichtbar sind, werden sie auch Hintergrundsysteme oder Backend-Services genannt (als Abgrenzung zu der Anwendung, die der Nutzer sieht, welche Frontend genannt wird). Neben der fachspezifischen Verarbeitung und Speicherung von Daten übernehmen diese Systeme oft Aufgaben zur Authentifizierung und Autorisierung von Nutzern oder andere zentrale Tätigkeiten. Dies erlaubt es, dass nicht alle Funktionalitäten der Anwendungen auf den Endgeräten umgesetzt werden müssen. Oft beschränken sie sich lediglich auf eine grafische Nutzerführung. Eine generelle Aussage darüber, wie viel Funktionalität in der Anwendung selbst umgesetzt und wie viel auf einen Server ausgelagert wird, kann nicht getroffen werden. Die Ausprägungen können von Anwendung zu Anwendung variieren. Daher ist bei vollumfänglicher sicherheitstechnischer Betrachtung der gesamten Anwendung die Sicherheit des Hintergrundsystems ein essentieller Teil.

Für die Nutzung von Anwendungen, die an ein Hintergrundsystem angeschlossen sind, ist meistens eine aktive Internetverbindung zwingend erforderlich. Dabei wird für die Kommunikation zwischen Front- und Backend-Systemen meist eine über TLS gesicherte Transportverbindung eingesetzt. Der Einsatz von Hintergrundsystemen beschränkt sich nicht nur auf den Bereich der Web-Anwendungen, sondern spiegelt den aktuellen Stand der Technik für fast alle Anwendungen wider. Hierbei werden im wesentlich drei Szenarien unterschieden:

- Der Hersteller der Anwendung verwaltet die Infrastruktur des Hintergrundsystems selbst (siehe Kapitel 2.3.1).
- Der Hersteller der Anwendung lässt die Infrastruktur von einem externen Dienstleister verwalten (siehe Kapitel 2.3.2).
- Das gesamte Hintergrundsystem der Anwendung wird bei einem Cloud-Dienstleister gehostet (siehe Kapitel 2.3.3).

Abhängig von der Art des Betriebs und den damit verbundenen unterschiedlichen Angriffsvektoren stehen dem Hersteller unterschiedliche Möglichkeiten zur Verfügung, die Sicherheit der Gesamtlösung und der gespeicherten und zu verarbeitenden Daten zu gewährleisten.

Eine Trennung zwischen der Web-Anwendung und dem Hintergrundsystem ist nicht einfach, da die Web-Anwendung durch das Hintergrundsystem zur Nutzung durch den Benutzer auf dessen Plattform mit Web-Browser ausgeliefert wird. Damit wird aber auch deutlich, dass eine Web-Anwendung ohne Hintergrundsystem gar nicht denkbar ist. Das bedeutet, dass bei einer solchen Architektur zusätzlich zu

dieser Technischen Richtlinie immer auch „TR-03174 Anforderungen an Anwendungen im Finanzwesen Teil 3: Hintergrundsysteme“ [TR03174-3] beachtet werden muss.

2.3.1 Selbst gehostete Systeme

Bei selbst gehosteten Systemen agiert der Entwickler des Hintergrundsystems auch als Betreiber. Damit hat er den direkten Zugriff auf die Systeme und deren Umgebung. Die Server, auf denen das Hintergrundsystem betrieben wird, sind innerhalb der Betriebsumgebung des Herstellers untergebracht und die physische, technische und organisatorische Absicherung der Systeme erfolgt durch ihn alleine. Der größte Vorteil dieser Lösung besteht darin, dass der Hersteller die alleinige Hoheit über die Systeme hat und schnell und direkt auf jegliche Vorgänge reagieren kann. Da er die Systeme auch selber verwaltet und die Softwarekomponenten darauf selber auswählt bzw. entwickelt, hat er auch am meisten Wissen über die mögliche Verwundbarkeit der Systeme. Allerdings lastet in diesem Fall auch die alleinige Verantwortung auf dem Hersteller, sodass er z.B. dauerhaft Personal abstellen muss, um Sicherheitsvorfälle zu überwachen und angemessen darauf zu reagieren. Je nach geschäftlicher Ausrichtung des Herstellers, besitzt dieser möglicherweise zwar viel Wissen im fachlichen Bereich seiner jeweiligen Anwendung, aber weniger im Bereich der IT-Sicherheit.

2.3.2 Extern gehostete Systeme

Bei dieser Variante werden die Server in einem Datacenter eines externen Dienstleisters gehostet, der sich üblicherweise auf Hosting spezialisiert hat. Die sicherheitstechnischen Vorteile bei dieser Lösung bestehen darin, dass der Dienstleister in der Regel mehr Erfahrung mit dem Betrieb solcher Systeme hat, was einen positiven Einfluss vor allem auf die Verfügbarkeit hat. Je nach Ausgestaltung der Dienstleitung übernimmt der externe Hoster auch weitergehende Aufgaben, wie z.B. die Versorgung der Betriebssysteme mit Sicherheitsupdates, Datensicherung und Backups, sowie Überwachung und Monitoring, um rechtzeitig auf verdächtige Aktivitäten reagieren zu können.

Der Betreiber muss dem Hoster ein gewisses Maß an Vertrauen entgegenbringen. So kann er selber z.B. die Integrität der Hardware nicht überwachen, weil mit einem direkten physischen Zugang Software-Überwachungsmaßnahmen immer umgangen werden können. Außerdem besitzt der Hoster in der Regel viele Kunden, die alle auf technischer Ebene voneinander separiert werden müssen um unbeabsichtigten Informationsabfluss, etwa zu Konkurrenten oder an die Öffentlichkeit, zu verhindern. Nicht zuletzt können durch die Aufteilung der Zuständigkeitsbereiche Reibungsverluste entstehen, die gerade bei kritischen Situationen wertvolle Zeit kosten können.

2.3.3 Cloud Computing

Cloud Computing beschreibt ein Modell, das bei Bedarf – meist über das Internet und geräteunabhängig – zeitnah und mit wenig Aufwand geteilte Computerressourcen als Dienstleistung, etwa in Form von Servern, Datenspeicher oder Anwendungen, bereitstellt und nach Nutzung abrechnet. Je nach Bedarf des Kunden können Ressourcen flexibel angepasst werden. Dadurch hat der Hersteller der Anwendung weniger Einfluss auf die Ausführungsumgebung als bei einem einfachen Hosting. Es ist beispielsweise nicht mehr möglich, zu erkennen, auf welchem Gerät eine bestimmte Operation ausgeführt wird. Der Betreiber muss sich hier voll und ganz auf den Anbieter der Cloud-Lösung verlassen können. Deswegen empfiehlt das BSI beim Einsatz von Cloud Computing für Anwendungen im Sinne der TR auf Anbieter zurückzugreifen, welche die Anforderungen aus dem „Kriterienkatalog Cloud Computing“ des BSI [KCC-C5] erfüllen. Hierbei muss der Betreiber auf Basis des vorgelegten Testats prüfen, ob die Anforderungen der TR durch die genutzten Cloud-Dienste erfüllt werden. Alternativ zum C5-Testat sind auch Anbieter mit vergleichbaren Testaten oder Zertifikaten zulässig (vgl. [TR03174-3]).

2.4 Security Problem Definition

Die Security Problem Definition beschreibt Annahmen, Bedrohungen und organisatorische Sicherheitspolitiken, die für Anwendungen im Finanzwesen zur Erbringung der Sicherheitsleistung relevant sind.

2.4.1 Annahmen

A.Backend Das Hintergrundsystem befindet sich in einer geschützten Umgebung. Es ist durch organisatorische und technische Maßnahmen sichergestellt, dass Angreifer sich keinen physischen Zugriff auf die Infrastruktur des Hintergrundsystems verschaffen können. Das Hintergrundsystem erfüllt die Anforderungen der „TR-03174 Anforderungen an Anwendungen im Finanzwesenwesen Teil 3: Hintergrundsysteme“ [TR03174-3].

A.Browser Der durch den Benutzer verwendete Web-Browser ist frei von Schwachstellen. Damit wird angenommen, dass ein Zugriff auf ungeschützte Datenstrukturen im Speicher, welche etwa den Zugriff auf Schlüssel und sensible Daten ermöglicht, ausgeschlossen ist. Der Browser implementiert das TLS-Protokoll zur Absicherung der Kommunikation mit dem Hintergrundsystem entsprechend dem aktuellen Stand der Technik auf sichere Art und Weise. Dies schließt eine korrekte und vollständige Zertifikatsprüfung mit ein.

A.Device Die Plattform, auf der die Web-Anwendung genutzt wird, wird vom Nutzer selbst betrieben und ist vor Schwachstellen geschützt.

A.DevRNG Es wird angenommen, dass der Web-Browser des Nutzers für die gesicherte Verbindungsaufnahme zum Hintergrundsystem Zufall ausreichender Qualität benutzt, sodass die Vertraulichkeit und Integrität der übertragenen Daten gesichert ist.

A.Updates Die Plattform bestehend aus Betriebssystem und Web-Browser, auf der die Web-Anwendung genutzt wird, wird vom Nutzer selbst betrieben und vor Schwachstellen geschützt, etwa durch Aktualisieren des Betriebssystems und des Web-Browsers nach Bereitstellung von Updates. Ihre Sicherheit wurde nicht vorsätzlich durch den Nutzer beeinträchtigt (z.B. „Roots“ oder „Jailbreaks“[2]). Der Bezug des genutzten Web-Browsers erfolgt ausschließlich über offizielle und vertrauenswürdige Quellen.

A.User Der Benutzer der Web-Anwendung prüft im Browser, ob er mit der korrekten Internetadresse verbunden ist. Der Nutzer verwendet die standardmäßigen Sicherheitseinstellungen des genutzten Web-Browsers.

Anwendungshinweis: Im Falle der Web-Anwendung gibt es a priori keine Möglichkeit Informationen dem Nutzer authentisch anzuzeigen. Die Web-Anwendung ist darauf angewiesen, dass der Nutzer die richtige Webpräsenz über den Web-Browser nutzt. Eine gefälschte Webpräsenz kann ebenfalls eine gültige TLS-Verbindung anbieten und wird dann die notwendigen Sicherheitshinweise dem Benutzer einfach vorenthalten oder verfälscht präsentieren. Daher kann die Etablierung des initialen Vertrauensankers über die Serverauthentisierung im Rahmen der TLS-Verbindung insbesondere mit der korrekten Webpräsenz lediglich als Annahme formuliert sein. Besteht zwischen dem Anbieter der Web-Anwendung und dem Nutzer ein unabhängiger Kanal zur Informationsübermittlung bspw. über eine Krankenkasse kann dieser natürlich genutzt werden, um Informationen zur Web-Präsenz des Anbieters dem Nutzer authentisch zur Verfügung zu stellen.

[2] Das Aufweichen der betriebssystemeigenen Sicherheitsfunktionalitäten durch Erlangen von erhöhten Zugriffsrechten und Ermöglichen der Installation von Applikationen aus unbekannten Quellen.

2.4.2 Bedrohungen

T.SensitiveData	Sensible Daten in der Technischen Richtlinie sind im Sinne des Anhang A zu verstehen. Ein Unbefugter erhält Zugriff auf solche sensiblen Daten der Web-Anwendung, etwa auf unverschlüsselt gespeicherte Daten im Dateisystem (z.B. Browsercache) oder Arbeitsspeicher. Dies umfasst auch, dass ein Angreifer auf verschlüsselte, sensible Daten, nach Analyse des für die Übertragung zwischen Web-Browser und Hintergrundsystem genutzten Verschlüsselungsmechanismus, im Klartext zugreifen kann.
T.Auth	Ein Angreifer erhält unter einer fremden Nutzerkennung oder der Verwendung fremder Rollen- oder Gruppenzugehörigkeit Zugriff auf sensible Daten anderer Nutzer.
T.DevFunctions	Ein Angreifer nutzt in der Web-Anwendung versteckte oder verbliebene Entwickler- bzw. Debuggingfunktionen zur Unterwanderung von Sicherheitsmaßnahmen.
T.Expense	Die Web-Anwendung verursacht unvorhergesehene, zusätzliche Kosten für den Nutzer oder Betreiber.
T.Impersonation	Ein Angreifer erhält durch fehlende oder fehlerhafte Zugriffskontrollen oder durch Erraten von Zugriffsparametern unberechtigten Zugriff auf sensible Daten oder kostenpflichtige Funktionen eines anderen Nutzers.
T.InfoDisclosure	Ein Angreifer führt eine Analyse der Web-Anwendung durch und findet Referenzen auf z.B. Entwicklerinstanzen, festkodierte Testaccounts oder Daten zur Verwendung in Verschlüsselungsroutinen.
T.Integrity	Ein Angreifer ist in der Lage, Daten innerhalb des Arbeitsspeichers oder auf dem Transportweg unbemerkt zu manipulieren oder zu löschen.
T.VisibleAsset	Der Angreifer kann durch „Schulter-Surfen“[3] sensible Daten, die auf der Web-Anwendung dargestellt werden, mitlesen.

2.4.3 Organisatorische Sicherheitspolitiken

OSP.Authorization	Der Hersteller entwickelt ein Autorisierungskonzept, welches sowohl den lesenden, als auch den schreibenden Zugriff auf sensible Daten steuert. Die Zugriffsberechtigungen müssen so gewählt werden, dass ausschließlich für die Erfüllung des primären bzw. rechtmäßigen Zwecks erforderliche Rechte erteilt werden. Das Autorisierungskonzept muss unabhängig von der Authentifizierung implementiert werden.
OSP.BrowserCred	Aktuelle Plattformen mit Web-Browser bieten in aller Regel an, Login-Informationen zu Web-Angeboten gesichert zu speichern, um beim erneuten Besuch des Web-Angebots den Nutzern einen vereinfachten Zugang anbieten zu können. Die Web-Anwendung weist den Nutzer auf die damit verbundenen Restrisiken geeignet hin.
OSP.User	Der Nutzer ist über die Nutzungsbedingungen der Web-Anwendung auf seine Mitwirkungspflichten hinzuweisen. Stimmt der Nutzer diesen Nutzungsbedingungen nicht zu, MUSS er von der Nutzung des Angebots ausgeschlossen werden. Mit der Zustimmung des Benutzers hingegen wird davon ausgegangen, dass der Nutzer sich entsprechend den Nutzungsbedingungen verhält.

[3] Beim Schulter-Surfen blickt der Angreifer unbemerkt über die Schulter auf das Gerät um Informationen zu erhalten.

OSP.CriticalUpdates Der Hersteller überprüft und überwacht die für die Web-Anwendung nutzbaren Browser sowie die für die Web-Anwendung genutzten Drittanbieter-Software[4] dauerhaft auf ausnutzbare Schwachstellen. Der Hersteller muss beim Bekanntwerden von Schwachstellen kurzfristig ein Update der Web-Anwendung bereitstellen, welches die Schwachstelle unzugänglich macht oder deren Ausnutzbarkeit verringert. Das Hintergrundsystem MUSS die Benutzung der Anwendung mit einem veralteten Browser bzw. Browser in einer veralteten Version unterbinden.

OSP.DataSovereignty Die Web-Anwendung stellt die Datenhoheit des Nutzers sicher. Die Anwendung weist den Nutzer auf bestehende Risiken durch die Konfiguration seines Endgeräts hin und lässt ihn entscheiden, die Nutzung abzubrechen. Auf Anforderung des Nutzers löscht die Web-Anwendung bereits erfasste Daten im Hintergrundsystem und weist den Nutzer darauf hin, dass ggf. lokal gespeicherte Daten (z.B. gespeicherte Formulardaten od. Browsercache) durch den Nutzer selbst gelöscht werden müssen. Exportiert der Nutzer sensible Daten unverschlüsselt und entzieht diese damit der Überwachung durch die Web-Anwendung, wird er durch die Web-Anwendung darauf aufmerksam gemacht, dass der Nutzer selbst die Verantwortung für die Datensicherheit dieser exportierten Daten übernimmt.

OSP.Disclosure Der Hersteller bietet einen niederschwelligen Prozess zum Melden von Schwachstellen an. Das heißt, er stellt leicht auffindbare Kontaktinformationen zur Sicherheits-Abteilung bereit und bietet eine Möglichkeit, um Schwachstellen anonym zu melden.

OSP.Purpose Jegliche Datenerhebung, -verarbeitung, -speicherung und -weitergabe darf nur mit einer Zweckbindung erfolgen. Der Hersteller veröffentlicht dafür den rechtmäßigen Zweck der Web-Anwendung und darüber hinaus welche Daten wie verarbeitet werden und wo und wie lange sie gespeichert werden. Ausgehend vom rechtmäßigen Zweck ist das zulässige Kommunikationsverhalten sowie die verwendete interne und externe Sensorik auszuwählen.

Anwendungshinweis: Ortungsdaten wie IP-Adressen, GPS u. ä. dürfen nur verarbeitet werden, wenn diese für die Funktion der Web-Anwendung von essentiellem Nutzen sind. Die so erhobenen Daten dürfen ausschließlich zweckgebunden verarbeitet werden. Sie dürfen nicht direkt oder indirekt im Gerät persistiert werden, sofern dies nicht unmittelbar für den Verwendungszweck erforderlich ist.

OSP.SecurityLifeCycle Der Hersteller realisiert einen Entwicklungszyklus, dessen Teilschritte darauf ausgelegt sind, die Sicherheit der Web-Anwendung zu stärken. Darunter fallen Maßnahmen, mit denen bösartige Aktivitäten erkannt werden und der Betreiber angemessene Gegenmaßnahmen einleiten kann.

2.4.4 Restrisiken

Der Betrieb von Anwendungen im Finanzwesen hat besonders hohe Anforderungen, die mit bestehenden Endgeräten und Cloud-Lösungen nur unzureichend abzudecken sind. Daher weist die Technische Richtlinie auf bestehende Restrisiken hin:

Mobile Endgeräte sind besonders anfällig für Diebstahl. Auch bei der Nutzung der sicheren Quellen (s.o.) ist nicht ausgeschlossen, dass auf diesen Schadsoftware zum Download angeboten wird. Installierte Schadsoftware kann bestehende Schwachstellen ausnutzen.

[4] Unter einer Drittanbieter-Software soll die Zusammenfassung von Funktionalitäten verstanden werden, die nicht in der Hoheit des Entwicklers der Anwendung entstanden sind und die auch nicht Teil der Funktionalität der verwendeten Betriebssystemplattform ist.

Der Betrieb des Hintergrundsystems bei Public Cloud-Anbietern beinhaltet besondere Risiken für die sensiblen Daten der Nutzer. Während hohe Entropie, sichere Kommunikations- und Verschlüsselungsverfahren Risiken abmildern, sind Daten während der Verarbeitung in der Cloud potentiell ungeschützt angesehen werden. Dies stellt höchste Anforderungen an den Betreiber der Cloud, sowie an andere Anwender, die eventuell gleichzeitig Ressourcen derselben physischen Maschine benutzen dürfen. Durch Überwinden von Trennungsmechanismen erhält ein Angreifer Zugriffsmöglichkeiten außerhalb seines Mandantenbereichs und kann unter Umständen sensible Daten eines anderen Mandanten (hier: der Anwendungen im Finanzwesen), während deren Verarbeitung, einsehen und manipulieren. Eine Verletzung der Vertraulichkeit kann ggf. auch ohne ein Überwinden von Trennungsmechanismen möglich sein, wenn es einem bösartigen Mandanten auf der gleichen physischen Maschine gelingt Seitenkanäle auszunutzen, die bei der Verarbeitung sensibler Daten in einer anderen virtuellen Maschine entstehen können.

Der Schutz von Kommunikationsverbindungen zwischen der Web-Anwendung und dem Hintergrundsystem erfolgt mittels des kryptographisch gesicherten TLS-Protokolls. Im vorliegenden Szenario geht die TR von einer einseitigen Authentisierung aus, wobei der genutzte Web-Browser die Authentizität des Hintergrundsystems prüft. Zufallszahlen auf Smartphone-Plattformen erreichen im Allgemeinen jedoch nicht die notwendige Qualität, die für den Schutz sensibler Daten innerhalb einer Anwendung im Finanzwesen notwendig sind. Das Restrisiko während des Verbindungsaufbaus besteht darin, dass der Angreifer die Authentizität eigener Nachrichten vortäuschen kann. Dadurch könnte der Angreifer sensible Daten, welche von der Anwendung an das Hintergrundsystem übermittelt werden, einsehen und manipulieren. Anders als im Falle der nativen Applikation, besteht für eine Web-Anwendung keine Möglichkeit, zusätzlichen Zufall in die Verbindung einzubringen, um dieses Restrisiko abzumildern.

Mit der Annahme A.Browser wurde im Sinne einer geschlossenen Sicherheitsaussage in dieser Technischen Richtlinie angenommen, dass der genutzte Web-Browser frei von Schwachstellen ist. Dies kann in der Realität nicht vollständig vorausgesetzt werden. Einige Prüfkriterien versuchen dieses Restrisiko zu minimieren, insbesondere z.B. O.Arch_8. Gleichwohl kann das Restrisiko von Schwachstellen im genutzten Web-Browser damit nicht gänzlich ausgeschlossen werden.

Weitere Restrisiken bestehen darin, dass der Nutzer bestimmte Eigenschaften der Plattform bzw. des Web-Browsers nutzt, die wiederum mit einer Gefährdung sensibler Daten verbunden sein können. Als Beispiel sei hier auf O.Auth_9 verwiesen. Diese oder ähnlich Prüfaspekte sollen sicherstellen, dass der Nutzer über die Nutzungsbedingungen umfassend über seine Verantwortlichkeiten aufgeklärt wird. Mithin eliminieren die Nutzungsbedingungen das Restrisiko für die Web-Anwendung formal. Es obliegt dem Nutzer mit diesen Restrisiken verantwortlich umzugehen.

Im Allgemeinen ist auf Grund der in Kapitel 2.1 und 2.2 im Teil 1 dieser Familie von Technischen Richtlinien [TR03174-1] beschriebenen Einschränkungen, bezogen auf den Umfang der Technischen Richtlinie, eine gesamtheitliche Aussage über die Sicherheit der mobilen Web-Anwendung, selbst unter Berücksichtigung aller aufgeführten Prüfaspekte, nicht möglich. Um die Sicherheit der gesamten Web-Anwendung zu erhöhen, ist es erforderlich weitere Literatur zu studieren. Dies gilt insbesondere für den Schutz vor Angriffen, welche direkt das eingesetzte Hintergrundsystem als Ziel haben und bei der Verbindung von Anwendungen im Finanzwesens mit IoT-Geräten.

3 Prüfaspekte für Anwendungen im Finanzwesen

3.1 Prüfaspekte

Die Prüfung nach der Technischen Richtlinie deckt die minimalen Sicherheitseigenschaften von Anwendungen im Finanzwesen ab. Die zu prüfende Sicherheitsfunktionalität lässt sich in folgende Prüfaspekte gliedern:

(1) Prüfung des Anwendungszwecks

(2) Prüfung der Architektur

(3) Prüfung des Quellcodes

(4) Prüfung der Drittanbieter-Software

(5) Prüfung der Kryptographische Umsetzung

(6) Prüfung der Authentisierung und Authentifizierung

(7) Prüfung von Datensicherheit

(8) Prüfung auf Kostenpflichtige Ressourcen

(9) Prüfung der Netzwerkkommunikation

(10) Prüfung auf Plattformspezifische Interaktionen

(11) Prüfung der Resilienz

Der Hersteller dokumentiert für jeden Prüfaspekt, sofern die zu schützende Funktionalität verwendet wird, wie dessen Anforderung durch die Implementierung sichergestellt wird.

3.1.1 Prüfaspekt (1): Anwendungszweck

O.Purp_1	Der Hersteller MUSS die rechtmäßigen Zwecke der Web-Anwendung und die Verarbeitung von personenbezogenen Daten offenlegen (etwa in der Beschreibung der Nutzungsbedingungen der Web-Anwendung) und den Nutzer spätestens bei der erstmaligen Nutzung der Anwendung darüber informieren.
O.Purp_2	Die Web-Anwendung DARF KEINE Daten erheben und verarbeiten, die nicht dem rechtmäßigen Zweck der Anwendung dienen.
O.Purp_3	Die Web-Anwendung MUSS vor jeglicher Erfassung oder Verarbeitung personenbezogener Daten eine aktive und eindeutige Einwilligungserklärung des Nutzers einholen.
O.Purp_4	Daten, deren Verarbeitung der Nutzer nicht ausdrücklich zugestimmt hat, DÜRFEN NICHT von der Web-Anwendung oder dem Hintergrundsystem erfasst, erhalten oder genutzt werden.
O.Purp_5	Die Web-Anwendung MUSS ermöglichen, dass der Nutzer eine bereits erteilte Einwilligung wieder entziehen kann. Der Nutzer MUSS vor der Einwilligung über die Möglichkeit des Widerrufs und die sich daraus ergebenden Veränderungen im Verhalten der Anwendung informiert werden.
O.Purp_6	Der Hersteller MUSS ein Verzeichnis führen, welches erkennen lässt, welche Nutzereinwilligungen vorliegen. Der nutzerspezifische Teil des Verzeichnisses MUSS für den Nutzer automatisiert einsehbar sein. Es SOLL eine Historie dieses Verzeichnisses angefordert werden können.
O.Purp_7	Setzt die Web-Anwendung Drittanbieter-Software ein, MÜSSEN alle verwendeten Funktionen für die rechtmäßigen Zwecke der Anwendung erforderlich sein. Die

Anwendung SOLL anderweitige Funktionen sicher deaktivieren. Wird nur eine einzige oder sehr wenige Funktionen der Drittanbieter-Software benötigt, MUSS abgewogen werden, ob die Einbindung des gesamten Drittanbieter-Software im Verhältnis zur Vergrößerung der Angriffsoberfläche durch die verwendete Drittanbieter-Software steht.

O.Purp_8 Sofern es nicht für den vorgesehenen primären oder rechtmäßigen Zweck einer Web-Anwendung erforderlich ist, DÜRFEN sensible Daten NICHT mit Dritten geteilt werden. Die Anwendung MUSS den Nutzer über die Konsequenzen einer eventuellen Weitergabe von Anwendungsdaten vollumfänglich informieren und sein Einverständnis einholen (OPT-IN).

O.Purp_9 Die Web-Anwendung DARF sensible Daten NICHT auf dem Bildschirm darstellen, außer dies ist für den primären Zweck der Anwendung erforderlich.

3.1.2 Prüfaspekt (2): Architektur

O.Arch_1 „Security“ MUSS ein fester Bestandteil des Softwareentwicklungs- und Lebenszyklus‘ für die gesamte Web-Anwendung und das Hintergrundsystem sein.

O.Arch_2 Bereits in der Designphase von Web-Anwendung und Hintergrundsystem MUSS berücksichtigt werden, dass die Anwendung in der Produktivphase sensible Daten verarbeiten wird. Die Architektur der Anwendung MUSS dafür die sichere Erhebung, Verarbeitung, Speicherung und Löschung der sensiblen Daten in einem Datenlebenszyklus gewährleisten.

O.Arch_3 Der Lebenszyklus von kryptographischem Schlüsselmaterial MUSS einer ausgearbeiteten Richtlinie folgen, die Eigenschaften wie die Zufallszahlenquelle, detaillierte Angaben zur Aufgabentrennung von Schlüsseln, Ablauf von Schlüsselzertifikaten, Integritätssicherung durch Hash-Algorithmen etc., umfasst. Die Richtlinie SOLL auf anerkannten Standards wie [TR02102-2] und [NIST80057] basieren.

O.Arch_4 In Backups gespeicherte sensiblen Daten MÜSSEN gemäß dem aktuellen Stand der Technik verschlüsselt sein. Dies schließt das Persistieren sensibler Daten durch den Browser, etwa in dessen Cache, mit ein.

O.Arch_5 Nutzt die Web-Anwendung Drittanbieter-Software, MUSS der Hersteller sicherstellen[5], dass nur solche Drittanbieter-Software zum Einsatz kommen, deren zu nutzenden Funktionen sicher genutzt werden können und dem Nutzer Informationen über den Nutzungsumfang und die eingesetzten Sicherheitsmechanismen klar darstellen. Die Anwendung MUSS diese Funktionen sicher nutzten. Der Hersteller MUSS darüber hinaus sicherstellen[5], dass ungenutzte Funktionen durch Dritte nicht aktiviert werden können.

O.Arch_6 Die Architektur der Web-Anwendung SOLL einem minimalistischen Ansatz folgen und mit einer serverseitig lokalisierten Verarbeitungslogik realisiert sein, d.h. es SOLLEN keine komplexen aktiven Inhalte (Java Applets, ActiveX-Plugin, o.ä.) verwendet werden.

O.Arch_7 Der Hersteller MUSS dem Nutzer eine barrierearme Möglichkeit bereitstellen, um Sicherheitsprobleme zu melden. Die Kommunikation SOLL über einen verschlüsselten Kanal stattfinden.

[5] Sicherstellen meint das Abfragen einer Eigenschaft oder eines Zustands und anschließende Prüfen der Abfrage auf ein positives Ergebnis.

O.Arch_8	Die Web-Anwendung MUSS beim Start die Aktualität des genutzten Web-Browsers prüfen. Wenn die Installation eines sicherheitsrelevanten Updates noch nicht erfolgt ist, DARF die Web-Anwendung KEINEN Zugriff auf sensible Daten ermöglichen.
O.Arch_9	Die Web-Anwendung SOLL HTTP-Server-Header nutzen, die dem aktuellen Stand der Technik entsprechen und die Sicherheit der Anwendung erhöhen. Dazu gehören unter anderem HTTP Strict Transport Security (HSTS), Content Security Policy (CSP) und X-Frame-Options.

3.1.3 Prüfaspekt (3): Quellcode

O.Source_1	Die Anwendung MUSS alle Eingaben vor deren Verarbeitung prüfen, um potenziell bösartige Werte vor der Verarbeitung herauszufiltern.
O.Source_2	Die Anwendung MUSS eingehende und ausgehende Daten maskieren beziehungsweise von potenziell schadhaften Zeichen bereinigen oder deren Verarbeitung ablehnen.
O.Source_3	Fehlermeldungen und Log-Dateien DÜRFEN KEINE sensiblen Daten (z. B. User Identifier oder Session-IDs) enthalten.
O.Source_4	Potenzielle Ausnahmen im Programmablauf (Exceptions) MÜSSEN abgefangen, kontrolliert behandelt und dokumentiert werden. Technische Fehlerbeschreibungen (z.B. Stack Traces) DÜRFEN dem Nutzer NICHT angezeigt werden.
O.Source_5	Bei Ausnahmen im Programmablauf (Exceptions) SOLL die Web-Anwendung Zugriffe auf sensible Daten abbrechen und diese im Speicher sicher löschen.
O.Source_6	Alle Optionen zur Unterstützung der Entwicklung (z. B. Entwickler-URLs, Testmethoden, Überreste von Debugmechanismen etc.) MÜSSEN in der Produktiv-Version vollständig entfernt sein.
O.Source_7	Vor der produktiven Bereitstellung der Anwendung SOLLEN moderne Sicherheitsmechanismen, wie beispielsweise Obfuskation und Bundler, verwendet werden.
O.Source_8	Für die Entwicklung der Anwendung SOLLEN Werkzeuge zur statischen Codeanalyse eingesetzt werden.
O.Source_9	Nutzt die Web-Anwendung URL-Weiterleitungen (URL-Redirects), MUSS diese kontrolliert erfolgen.
O.Source_10	Die Web-Anwendung MUSS Maßnahmen vorsehen, die verhindern, dass Funktionalitäten, die nicht in der Entwicklungshoheit des Herstellers liegen, in die Web-Anwendung eingeschleust und zur Ausführung gebracht werden.
O.Source_11	Sensible Daten DÜRFEN NICHT in der URL vorkommen. Die Web-Anwendung MUSS solche Daten in HTTP Request Headern oder POST-Parametern verarbeiten.

3.1.4 Prüfaspekt (4): Drittanbieter-Software

O.TrdP_1	Der Anbieter MUSS eine zentrale und vollständige Liste von Abhängigkeiten durch Drittanbieter-Software führen.
O.TrdP_2	Drittanbieter-Software MUSS in der neusten oder der ihr vorhergehenden, für die Veröffentlichung vorgesehenen Version verwendet werden.
O.TrdP_3	Drittanbieter-Software MUSS durch den Hersteller regelmäßig (durch Auswertung öffentlich verfügbarer Informationen oder durch statische/dynamische Testmethoden) auf Schwachstellen überprüft werden. Überreste von Optionen zur Unterstützung der Entwicklung (vgl.O.Source_6) sind hierbei als Schwachstelle zu werten. Der Hersteller

	MUSS für alle öffentlich bekannten Schwachstellen analysieren, inwieweit die Schwachstelle die Sicherheit des Gesamtsystems beeinträchtigt. Software, bzw. Funktionen aus Drittanbieter-Software DÜRFEN bei bekannten Schwachstellen, die die Sicherheit des Gesamtsystems betreffen NICHT eingesetzt werden.
O.TrdP_4	Sicherheitsupdates für Drittanbieter-Software MUSS zeitnah integriert und per Update dem Nutzer zur Verfügung gestellt werden. Der Hersteller MUSS ein Sicherheitskonzept vorlegen, das anhand der Kritikalität ausnutzbarer Schwachstellen die geduldete Weiternutzung für die Web-Anwendung, bzw. das Hintergrundsystem festlegt. Nachdem die Übergangsfrist (Grace Period) abgelaufen ist, DARF die Web-Anwendung NICHT mehr zur Benutzung angeboten werden.
O.TrdP_5	Vor der Verwendung von Drittanbieter-Software MUSS deren Quelle auf Vertrauenswürdigkeit geprüft werden.
O.TrdP_6	Die Anwendung SOLL sensible Daten nicht an Drittanbieter-Software weitergeben.
O.TrdP_7	Über Drittanbieter-Software eingehende Daten MÜSSEN validiert werden.
O.TrdP_8	Drittanbieter-Software, die nicht mehr gewartet wird, DARF NICHT verwendet werden.

3.1.5 Prüfaspekt (5): Kryptographische Umsetzung

O.Cryp_1	Beim Einsatz von Verschlüsselung in der Web-Anwendung DÜRFEN KEINE fest einprogrammierten geheimen, bzw. privaten Schlüssel eingesetzt werden.
O.Cryp_2	Die Anwendung MUSS auf bewährte Implementierungen zur Umsetzung kryptographischer Primitive und Protokolle zurückgreifen (vgl. [TR02102-2]).
O.Cryp_3	Die Wahl kryptographischer Primitive MUSS passend zum Anwendungsfall sein und dem aktuellen Stand der Technik (siehe [TR02102-1]) entsprechen.
O.Cryp_4	Kryptographische Schlüssel DÜRFEN NICHT für mehr als genau einen Zweck eingesetzt werden.
O.Cryp_5	Die Stärke der kryptographischen Schlüssel MUSS dem aktuellen Stand der Technik entsprechen (siehe [TR02102-1]).

3.1.6 Prüfaspekt (6): Authentisierung und Authentifizierung

O.Auth_1	Der Hersteller MUSS ein Konzept zur Authentisierung auf angemessenem Vertrauensniveau [TR03107-1], zur Autorisierung (Rollenkonzept) und zum Beenden einer Anwendungssitzung dokumentieren.
O.Auth_2	Die Anwendung SOLL Authentisierungsmechanismen und Autorisierungsfunktionen separat realisieren. Sind für die Anwendung verschiedene Rollen notwendig, MUSS eine Autorisierung bei jedem Datenzugriff separat realisiert werden.
O.Auth_3	Jeder Authentifizierungsvorgang des Nutzers MUSS in Form einer Zwei-Faktor-Authentifizierung umgesetzt werden.
O.Auth_4	Für die Bewertung eines Authentisierungsvorgangs SOLLEN zusätzliche Informationen (z. B. das verwendete Endgerät, die verwendete IP-Adresse oder die Zeit des Zugriffs) mit einbezogen werden.
O.Auth_5	Dem Nutzer SOLL eine Möglichkeit gegeben werden, sich über ungewöhnliche Anmeldevorgänge informieren zu lassen.
O.Auth_6	Die Anwendung MUSS Maßnahmen umsetzen, die ein Ausprobieren von Login-Parametern (z. B. Passwörter) erschweren.

O.Auth_7	Wurde die Anwendung unterbrochen (in den Hintergrundbetrieb versetzt), MUSS nach Ablauf einer angemessenen Frist (Grace Period) eine erneute Authentisierung durchgeführt werden.
O.Auth_8	Die Anwendung MUSS nach einer angemessenen Zeit in der sie nicht aktiv verwendet wurde (idle time) eine erneute Authentisierung fordern.
O.Auth_9	Die Anwendung MUSS nach einer angemessenen Zeit in der sie aktiv verwendet wurde (active time) eine erneute Authentisierung zur Reaktivierung der Serversitzung fordern.
O.Auth_10	Die Authentisierungsdaten DÜRFEN NICHT ohne eine erneute Authentifizierung des Nutzers geändert werden.
O.Auth_11	Die Anwendung MUSS für die Anbindung eines Hintergrundsystems eine dem Stand der Technik entsprechende Authentifizierung verwenden.
O.Auth_12	Authentisierungsdaten, wie bspw. Session-Identifier bzw. Authentisierungstoken, MÜSSEN als sensible Daten geschützt werden.
O.Auth_13	Die Anwendung MUSS es dem Nutzer ermöglichen einen oder alle zuvor ausgestellten Session-Identifier bzw. Authentisierungstoken zu invalidieren.
O.Auth_14	Wird eine Anwendungssitzung ordnungsgemäß beendet, MUSS die Anwendung das Hintergrundsystem darüber informieren, sodass Session-Identifier bzw. Authentisierungstoken sicher gelöscht werden. Dies gilt sowohl für das aktive Beenden durch den Benutzer (log-out), als auch für das automatische Beenden durch die Anwendung (vgl. O.Auth_8 und O.Auth_9).
O.Auth_15	Bei Änderung der Zugangsparameter SOLL der Nutzer über die zuletzt hinterlegten, gültigen Kontaktdaten über die Änderung informiert werden. Dem Nutzer SOLL über diesem Weg eine Möglichkeit geboten werden, die gemeldete Änderung zu sperren und nach entsprechender Authentifizierung neue Zugangsparameter zu setzen.
O.Auth_16	Der Nutzer MUSS in den Nutzungsbedingungen der Web-Anwendung auf das Restrisiko hingewiesen werden, welches mit der Speicherung der Login-Credentials im Web-Browser oder auch einem anderen externen Programm für einen komfortableren Anmeldevorgang verbunden ist.

3.1.6.1 Authentifizierung über Passwort

O.Pass_1	Bei einer Authentifizierung mittels Benutzername und Passwort MÜSSEN starke Passwortrichtlinien existieren. Diese SOLLEN sich am aktuellen Stand gängiger „Best-Practices“ orientieren.
O.Pass_2	Für die Einrichtung der Authentisierung mittels Benutzername und Passwort KANN die Stärke des verwendeten Passworts dem Nutzer angezeigt werden. Informationen über die Stärke des gewählten Passworts DÜRFEN NICHT gespeichert werden.
O.Pass_3	Der Nutzer MUSS die Möglichkeit haben, sein Passwort zu ändern.
O.Pass_4	Das Ändern und Zurücksetzen von Passwörtern MUSS protokolliert werden.
O.Pass_5	Werden Passwörter gespeichert, MÜSSEN diese mit einer den aktuellen Sicherheitsstandards entsprechenden Hash-Funktion und unter Verwendung geeigneter Salts gehasht werden.

3.1.7 Prüfaspekt (7): Datensicherheit

O.Data_1	Die Werkseinstellung der Web-Anwendung MUSS die maximale Sicherheit bieten.

O.Data_2	Exportiert der Nutzer sensible Daten unverschlüsselt MUSS der Nutzer durch die Web-Anwendung darauf aufmerksam gemacht werden, dass der Nutzer selbst die Verantwortung für die Datensicherheit dieser exportierten Daten übernimmt.
O.Data_3	Die Web-Anwendung DARF Ressourcen, die einen Zugriff auf sensible Daten ermöglichen, gegenüber Dritten NICHT verfügbar machen.
O.Data_4	Alle erhobenen sensiblen Daten DÜRFEN NICHT über die Dauer ihrer jeweiligen Verarbeitung hinaus in der Web-Anwendung gehalten werden.
O.Data_5	Die Web-Anwendung MUSS die Grundsätze der Datensparsamkeit und Zweckbindung berücksichtigen.
O.Data_6	Die Speicherung und Verarbeitung von sensiblen Daten SOLL im Hintergrundsystem erfolgen.
O.Data_7	Bei der Verwendung von Aufnahmegeräten (z. B. Kamera) MÜSSEN sämtliche Metadaten mit Datenschutz-Relevanz, wie etwa Rückschlüsse auf die GPS-Koordinaten des Aufnahmeorts, eingesetzte Hardware etc., entfernt werden.
O.Data_8	Bei der Erhebung von sensiblen Daten durch die Verwendung von Aufnahmegeräten (z. B. Kamera), MUSS vorgebeugt werden, dass andere Anwendungen darauf Zugriff erlangen könnten, etwa über eine Mediengalerie.
O.Data_9	Bei der Eingabe sensibler Daten über die Tastatur SOLL die Web-Anwendung unterbinden, dass Aufzeichnungen für Dritte erkennbar werden.
O.Data_10	Bei der Eingabe sensibler Daten SOLL der Export in die Zwischenablage unterbunden werden. Die Anwendung KANN alternativ eine eigene Zwischenablage implementieren, welche vor dem Zugriff durch andere Anwendungen geschützt ist.
O.Data_11	Sensible Daten DÜRFEN NICHT aus der Komponente, auf der sie erzeugt wurden, exportiert werden.
O.Data_12	Durch die Web-Anwendung kann der Zugriff für Dritte und die Speicherung des Bildschirms (z. B. Screenshots und Anzeigen für das App-Switching) nicht unterbunden werden. Über die Nutzungsbedingungen MUSS der Nutzer darüber informiert werden, dass sensible Daten über Screenshots oder Anzeigen für das App-Switching kompromittiert werden können.
O.Data_13	Über die Nutzungsbedingungen der Web-Anwendung MUSS der Nutzer über das Risiko informiert werden, welches damit verbunden ist, dass im gesperrten Zustand des Endgeräts die Verbindung zum Hintergrundsystem weiter geöffnet bleibt, wenn der Nutzer sich nicht explizit ausgeloggt hat.
O.Data_14	Die Web-Anwendung SOLL sicherstellen, dass bei ihrer Beendigung alle sensiblen Daten und anwendungsspezifischen Anmeldeinformationen im Web-Browser nicht mehr zugreifbar sind. Dies schließt insbesondere Cookies und Webstorage mit ein.
O.Data_15	Die Web-Anwendung MUSS dem Nutzer die Möglichkeit geben, dass bei endgültiger Beendigung der Nutzung alle sensiblen Daten und anwendungsspezifischen Anmeldeinformationen vollständig gelöscht bzw. unzugänglich gemacht werden.
O.Data_16	Für alle Cookies, auf die nicht mittels JavaScript zugegriffen wird, MUSS das HTTP-Only-Flag verwendet werden.
O.Data_17	Für alle Cookies, die sensible Daten enthalten, MUSS das Secure-Flag gesetzt sein.
O.Data_18	Für alle Formularfelder mit sensiblen Eingabedaten MUSS die Autocomplete-Funktion abgeschaltet sein.

O.Data_19 Im Browser persistierte Daten SOLLEN für weitere Hosts einer Domain unlesbar sein (d.h. Vermeidung von Domain-Cookies).

3.1.8 Prüfaspekt (8): Kostenpflichtige Ressourcen

O.Paid_1 Die Web-Anwendung MUSS für den Nutzer kenntlich machen, welche kostenpflichtigen Leistungen (z.B. Zusatzfunktionalitäten oder Premiumzugriffe) und welche kostenpflichtigen Ressourcen (z.B. SMS, Telefonate, mobile Daten) von der Anwendung angeboten oder verwendet werden.

O.Paid_2 Die Web-Anwendung MUSS vor der Verwendung kostenpflichtiger Leistungen das Einverständnis des Nutzers einholen.

O.Paid_3 Die Web-Anwendung MUSS vor einer Zugriffsanforderung auf kostenpflichtige Ressourcen oder kostenpflichtige Leistungen, das Einverständnis des Nutzers einholen.

O.Paid_4 Die Web-Anwendung KANN für den Zugriff auf häufig verwendete, kostenpflichtige Ressourcen oder kostenpflichtige Leistungen ein dauerhaftes Einverständnis des Nutzers einholen, soweit dies dem primären Zweck der Anwendung angemessen ist.

O.Paid_5 Die Web-Anwendung MUSS den Nutzer in die Lage versetzen, zuvor erteilte Einverständnisse zurückzuziehen.

O.Paid_6 Die Anwendung SOLL die Transaktionshistorie von kostenpflichtigen Leistungen im Hintergrundsystem ablegen. Die Transaktionshistorie, einschließlich der Metadaten, MUSS als sensibles Datum gemäß O.Purp_8 behandelt werden.

O.Paid_7 Falls die Web-Anwendung kostenpflichtige Funktionen anbietet, MUSS der Hersteller ein Konzept vorlegen, welches vorbeugt, dass Dritte die Zahlungsströme zur Nutzung von Anwendungsfunktionen zurückverfolgen können.

O.Paid_8 Die Web-Anwendung MUSS dem Nutzer eine Übersicht der entstandenen Kosten anbieten. Falls die Kosten aufgrund einzelner Zugriffe erfolgt sind, MUSS die Anwendung einen Überblick der Zugriffe aufführen.

O.Paid_9 Die Validierung von getätigten Bezahlvorgängen MUSS im Hintergrundsystem vorgenommen werden.

O.Paid_10 Zahlverfahren von Drittanbietern MÜSSEN die Anforderungen an Drittanbieter-Software erfüllen (vgl. Kapitel 3.1.4).

3.1.9 Prüfaspekt (9): Netzwerkkommunikation

O.Ntwk_1 Jegliche Netzwerkkommunikation der Web-Anwendung MUSS durchgängig mit gegenseitiger Authentisierung verschlüsselt werden.

O.Ntwk_2 Die Konfiguration der TLS-Verbindungen MUSS dem aktuellen Stand der Technik entsprechen (vgl. [TR02102-2]).

O.Ntwk_3 Die Web-Anwendung MUSS die Sicherheitsfunktionalität der jeweilig verwendeten Betriebssystem-Plattform und des Browsers verwenden, um sichere Kommunikationskanäle aufzubauen.

O.Ntwk_4 Die Web-Anwendung SOLL die Verwendung von Zertifikaten, deren Zertifikatskette dem Hersteller nicht vertrauenswürdig erscheint, unterbinden.

3.1.10 Prüfaspekt (10): Plattformspezifische Interaktionen

O.Plat_1 Für die Nutzung der Web-Anwendung SOLL das Endgerät über einen aktivierten Geräteschutz (Passwort, Mustersperre, o. ä.) verfügen. Im Fall eines nicht aktivierten

Geräteschutzes MUSS der Hersteller den Nutzer über die damit verbundenen Risiken aufklären.

O.Plat_2 Die Web-Anwendung DARF Berechtigungen, die für die Erfüllung ihres primären Zwecks nicht notwendig sind, NICHT einfordern.

O.Plat_3 Die Web-Anwendung MUSS den Nutzer auf den rechtmäßigen Zweck der anzufragenden Berechtigungen und auf die Auswirkungen hinweisen, die eintreten, falls der Nutzer diese nicht gewährt.

O.Plat_4 Die Web-Anwendung DARF KEINE sensiblen Daten in erweiterten Meldungen oder Benachrichtigungen, die nicht vom Nutzer explizit eingeschaltet wurden (siehe O.Plat_5), anzeigen.

O.Plat_5 Die Web-Anwendung KANN dem Nutzer die Optionen bieten, erweiterte Meldungen und Benachrichtigungen, ggf. auch mit sensiblen Inhalten, anzuzeigen. Bei Werkseinstellung MUSS diese deaktiviert sein.

O.Plat_6 Die Web-Anwendung MUSS das Nachladen von Inhalten auf Quellen beschränken, die unter der Kontrolle des Herstellers sind oder durch den Hersteller autorisiert wurden.

O.Plat_7 Die Web-Anwendung MUSS den Nutzer über das Risiko informieren, dass ggf. nach Beendigung der Web-Anwendung nutzerspezifischen Daten im Arbeitsspeicher verbleiben können.

O.Plat_8 Der Nutzer MUSS über Sicherheitsmaßnahmen informiert werden, sofern diese durch den Nutzer umsetzbar sind.

3.1.11 Prüfaspekt (11): Resilienz

O.Resi_1 Die Web-Anwendung MUSS dem Nutzer barrierearme Best-Practice-Empfehlungen zum sicheren Umgang mit der Anwendung und ihrer Konfiguration bereitstellen.

O.Resi_2 Die Web-Anwendung MUSS über die Nutzungsbedingungen dem Nutzer darstellen, welche Risiken für die Daten des Nutzers bei einer Benutzung von Geräten, deren Betriebssystem in keinem vom Betriebssystemhersteller vorgesehenen Betriebszustand ist, bestehen.

4 Prüfschritte einer Anwendung im Finanzwesen

4.1 Anforderungen an die Prüfung

Die TR-Prüfung von Anwendungen im Finanzwesen orientiert sich an den Prüfaspekten, die in Kapitel 3.1 aufgeführt sind. Kapitel 4.3 leitet aus den Prüfaspekten Testcharakteristika ab, welche die Anforderungen um eine Prüftiefe und Hinweise für TR-Prüfer erweitert. Unterstützend kann der Hersteller Aussagen tätigen, in denen er die betreffende Umsetzung skizziert und eine Referenz auf die jeweilige Implementierung angibt. Bei komplexen Testcharakteristika stellt der Hersteller eine umfassende Liste der Vorkommen zur Verfügung. Abhängig von der umgesetzten Prüftiefe unterstützen diese Herstelleraussagen die TR-Prüfung. Die untenstehende Tabelle legt dar, welche Prüfschritte mindestens für die jeweilige Prüftiefe gefordert sind.

Tabelle 2: Prüftiefen und Mindestanforderungen

Prüftiefe	Mindestanforderungen an die Prüfung
CHECK	Der Evaluator validiert (englisch „check“, analog zu Begriffsverwendung in der Common Criteria Evaluation Methodology) die vom Hersteller beschriebene Maßnahme im Hinblick auf ihre Wirksamkeit und räumt bestehende Zweifel (Plausibilitätsprüfung) aus, ob der Prüfaspekt und die damit verbundene Sicherheitsproblematik umfassend durch die beschriebenen Maßnahmen adressiert wird. Hierbei MUSS der Evaluator den aktuellen Stand der Technik für die jeweilige Plattform mitberücksichtigen. Die Validierung KANN weitergehende Schritte, wie z.B. eine Quelltextanalyse, umfassen, falls der Evaluator diese für eine umfassende Einschätzung benötigt.
EXAMINE	Der Evaluator untersucht (englisch „examine“, analog zu Begriffsverwendung in der Common Criteria Evaluation Methodology) die betreffende Testcharakteristik. Der Evaluator MUSS in seiner Prüfung über die Mindestanforderungen für „CHECK“ hinausgehen: In der Regel wird dies durch umfassende Quelltextanalyse der relevanten Implementierungsanteile und Penetrationstests geschehen. Die Unterstützung durch den Hersteller kann genutzt werden. „EXAMINE“ erfordert in jedem Fall eine eigenständige Beurteilung durch den Evaluator.

Aus den Prüftiefen folgt auch der Einsatz einer Quelltextanalyse bei der Begutachtung. Bei „CHECK“ wählt der TR-Prüfer aus, wie hoch die Abdeckung der Analyse für seine Einschätzung notwendig ist. Für „EXAMINE“ muss der TR-Prüfer erläutern, inwiefern sämtliche relevanten Codezeilen in Betracht gezogen wurden.

4.2 Protokollierung der Ergebnisse

Die Protokollierung der Testergebnisse ist so zu gestalten, dass unbeteiligte Dritte in die Lage versetzt werden, anhand der Angaben aus dem Prüfbericht die vorgenommen Prüfschritte zu wiederholen und dabei das gleiche Ergebnis zu erzielen. Hierzu ist es neben der Beschreibung der einzelnen Prüfschritte notwendig, dass die verwendeten Prüfwerkzeuge in den verwendeten Versionen in dem Prüfbericht ersichtlich sind. Die untenstehende Tabelle definiert die zulässigen Prüfergebnisse, welche sich aus der Prüfung einer Charakteristik ergeben können. Der TR-Prüfer begründet, wie er zu einem entsprechenden Ergebnis gekommen ist.

Tabelle 3: Mögliche Prüfergebnisse

Ergebnis	Notwendige Angaben
PASS	Der Prüfer erläutert sein Verständnis, warum die Hersteller-Implementierung das geforderte Sicherheitsziel erfüllt. Der Prüfbericht führt die durchgeführten Prüfschritte sowie das Prüfergebnis aus.
INCONCLUSIVE	Der Prüfbericht spezifiziert/referenziert die fehlenden oder inkonsistenten Informationen, damit der Hersteller die Nicht-Konformität zu dem betreffenden Aspekt des Sicherheitsziel bereinigen kann.
FAIL	Die geprüfte Anwendung verfehlt das betreffende Sicherheitsziel. Der Prüfer dokumentiert, inwiefern Angriffe durch Sicherheitsmaßnahmen in der Umgebung der Anwendung (z.B. operative Maßnahmen) verhindert werden können. Der Prüfer nimmt eine Evidenz der Verletzung der Prüfcharakteristik in die Protokollierung auf. Der Prüfer nimmt das durch die Verletzung der Testcharakteristik entstehende Risiko in die Risikoabschätzung mit auf.
NOT APPLICABLE (N/A)	Die geprüfte Anwendung verfügt über keinerlei Implementierung der durch den Prüfaspekt zu schützenden Funktionalitäten. Daher kann die betreffende Testcharakteristik nicht auf die zu prüfende Anwendung angewandt werden.

Die TR-Prüfer identifizieren bestehende Restrisiken beim Einsatz der Anwendung im Finanzwesen. Mit der Aufdeckung bestehender Restrisiken wird dem Umstand Rechnung getragen, dass der Verlust von Finanzdaten sofort zu einem Schaden für den Nutzer führt und ausreichende Schutzmaßnahmen zum Zeitpunkt der TR-Prüfung nicht identifiziert werden konnten. Die Risikobewertung muss mindestens folgende Aspekte umfassen:

- Identifikation von Risiken aus der unterlassenen oder unzureichenden Umsetzung von „SOLL"-Anforderungen in Sicherheitszielen.
- Implementierungsspezifische Risiken.
- Risiken durch Integration in der geplanten Betriebsumgebung.
- Die Eignung des Monitorings sowie im Produkt vorgesehene Reaktionsmöglichkeiten für den Betreiber in der Produktprüfung berücksichtigen.

4.3 Testcharakteristika

Die Testcharakteristiken erweitern die Prüfaspekte aus Kapitel 3 um ihre Prüftiefe und ergänzende Informationen für Evaluatoren. Der Evaluator soll über die einzelnen Prüfschritte hinaus sicherstellen, dass das betreffende Sicherheitsziel insgesamt erfüllt wird. Dies umfasst möglicherweise weitere, hier nicht aufgeführte Testcharakteristika.

4.3.1 Testcharakteristik zu Prüfaspekt (1): Anwendungszweck

Tabelle 4: Testcharakteristik: Anwendungszweck

Prüfaspekt	Kurzfassung des Prüfaspekts	Prüftiefe	Anmerkungen
O.Purp_1	Informationspflicht des Herstellers zum rechtmäßigen Zweck und Verarbeitung von personenbezogenen Daten.	CHECK	Der Evaluator prüft, ob eine Beschreibung vorhanden ist und diese den rechtmäßigen Zwecken der Anwendung entspricht. Dabei werden die vom Hersteller definierten rechtmäßige Zwecke als Grundlage genutzt. Eine juristische Prüfung der Rechtmäßigkeit ist nicht erforderlich.
O.Purp_2	Zweckgebundene Erhebung und Verarbeitung der Daten.	CHECK	Der Evaluator prüft, ob alle Daten die erhoben und den rechtmäßigen Zwecken der Anwendung entsprechen. Dabei werden die vom Hersteller definierten rechtmäßige Zwecke als Grundlage genutzt. Eine juristische Prüfung der Rechtmäßigkeit ist nicht erforderlich.
O.Purp_3	Einholung einer Einwilligungserklärung des Nutzers.	CHECK	Der Evaluator prüft, ob ohne Zustimmung des Nutzers personenbezogene Daten verarbeitet werden können.
O.Purp_4	Nutzung ausschließlich zugestimmter Daten.	CHECK	Der Evaluator gleicht die in O.Purp_2 gewonnen Erkenntnisse mit den erteilten Zustimmungen ab.
O.Purp_5	Entzug der Einwilligung ermöglichen.	CHECK	Der Evaluator prüft, ob dem Nutzer die Möglichkeit gegeben wird erteilte Einwilligungen wieder zu entziehen. Darüber hinaus validiert er, dass der Nutzer beim Entzug von Einwilligungen auf die daraus resultierenden Konsequenzen hingewiesen wird.
O.Purp_6	Führen eines Verzeichnisses der Nutzereinwilligungen.	CHECK	Der Evaluator prüft das Vorhandensein, die Aktualität und die Vollständigkeit des Verzeichnisses. Darüber hinaus prüft er, ob eine Historie dieses Verzeichnisses angefordert werden kann.

Prüfaspekt	Kurzfassung des Prüfaspekts	Prüftiefe	Anmerkungen
O.Purp_7	Nutzung nur erforderlicher Drittanbieter-Software von Dritten.	CHECK	Der Evaluator prüft die Abwägungen des Herstellers bei Funktionen, die nicht dem rechtmäßigen Zweck für die Anwendung dienen. So dürfte beispielsweise eine API für soziale Netzwerke nur verwendet werden, wenn dies mit dem rechtmäßigen Zweck der Anwendung vereinbar ist. Die Risikobewertung erfasst die Auswirkungen auf den Schutz der Finanzdaten, beispielsweise bei dem für Dritte erkennbaren Nutzungsverhalten in Logging Frameworks.
O.Purp_8	Weitergabe von sensiblen Daten nur für den primären oder rechtmäßigen Zweck.	CHECK	Der Evaluator prüft die Abwägungen des Herstellers, ob die Weitergabe von sensiblen Daten an Dritte dem primären oder rechtmäßigen Zweck für die Anwendung dient. Darüber hinaus prüft er, ob die Weitergabe immer explizit durch den Nutzer erlaubt werden muss (Opt-In). Die Weitergabe an Dienste, deren primärer Zweck die Verarbeitung von Daten für Werbezwecke ist, ist generell verboten. Die Risikobewertung berücksichtigt, wie die Weitergabe von Daten an Dritte im Verhältnis zum Schutzbedarf der weitergeleiteten Informationen (Daten) und der daraus resultierenden Gefahr der Preisgabe von Informationen steht. Die Nutzung durch Dritte könnte z.B. im Aufruf von Cookies, Bannern von Dritten oder die Umleitung von Formular-Input über Webseiten von unbeteiligten Parteien bestehen.

Prüfaspekt	Kurzfassung des Prüfaspekts	Prüftiefe	Anmerkungen
O.Purp_9	Nur zweckgebundene Darstellung sensibler Daten auf dem Bildschirm.	CHECK	Der Evaluator prüft die Abwägungen des Herstellers, ob die Darstellung von sensiblen Daten zum gegebenen Zeitpunkt für die Erfüllung des Zwecks der Anwendung erforderlich ist. Für die Risikobewertung ist zu berücksichtigen, wie die Anwendung den Nutzer davor schützt, sensible Daten anzuzeigen (vergleiche T.VisibleAsset). Bei Formularen muss die Web-Anwendung den Web-Browser anweisen, die Eingabe bei sensiblen Daten auszublenden (type="password") und Eingaben geeignet zu behandeln.

4.3.2 Testcharakteristik zu Prüfaspekt (2): Architektur

Tabelle 5: Testcharakteristik: Architektur

Prüfaspekt	Kurzfassung des Prüfaspekts	Prüftiefe	Anmerkungen
O.Arch_1	„Security“ ist Bestandteil des Softwareentwicklungs- und Lebenszyklus.	CHECK	Der Evaluator prüft, ob der Quelltext und die Design-Dokumente auf die Verwendung aktueller „Best-Practices“ bei der Entwicklung schließen lassen.
O.Arch_2	Berücksichtigung der Verarbeitung sensibler Daten in der Design-Phase.	CHECK	Der Evaluator prüft Design- und Architektur-Dokumente auf die Berücksichtigung der Verarbeitung sensibler Daten inkl. des Datenlebenszyklus.
O.Arch_3	Dokumentation des Lebenszyklus von kryptographischem Material.	CHECK	Der Evaluator bewertet die ausgearbeitete Richtlinie des Herstellers und deren Berücksichtigung in der Risikobewertung.
O.Arch_4	Keine unverschlüsselten sensiblen Daten in Backups oder im Browser-Cache persistiert.	EXAMINE	Der Evaluator prüft durch Quelltextanalyse und praktische Tests, ob sensible Daten gemäß dem aktuellen Stand der Technik in Backups und/oder im Browser-Cache verschlüsselt sind.

Prüfaspekt	Kurzfassung des Prüfaspekts	Prüftiefe	Anmerkungen
O.Arch_5	Sichere Nutzung der Funktionen von Drittanbieter-Software.	EXAMINE	Der Evaluator prüft, durch Quelltextanalyse und praktische Tests, dass Funktionalitäten sicher verwendet werden und ungenutzte Funktionalitäten nicht zugänglich sind. Darüber hinaus prüft er, ob der Nutzer ausreichend über die Verwendung von Drittanbieter-Software informiert wird.
O.Arch_6	Serverseitig lokalisierte Verarbeitungslogik der Web-Anwendung.	EXAMINE	Der Evaluator prüft, ob die Verarbeitungslogik der Web-Anwendung durch das Hintergrundsystem realisiert wird. Er stellt sicher, dass keine Java-Applets, ActiveX-Plugins oder vergleichbare komplexen Aktiven Inhalte eingesetzt werden. Beim Umgang mit aktiven Inhalten sind die „Sicherheitsmaßnahmen beim Einsatz aktiver Inhalte“ [BSI-CS-120] einzuhalten.
O.Arch_7	Barrierearme Möglichkeit zum Melden von Sicherheitsproblemen.	CHECK	Der Evaluator prüft, ob eine entsprechende Möglichkeit vorhanden ist. Falls kein verschlüsselter Kanal bereitgestellt wird, ist dies in der Risikobewertung zu berücksichtigen.
O.Arch_8	Prüfung auf Aktualität des genutzten Web-Browsers.	CHECK	Der Evaluator prüft, ob ein Zugriff auf sensible Daten mit einer veralteten Browserversion möglich ist. Das Hintergrundsystem kann im Rahmen der Verbindungsaufnahme durch den Web-Browser über Informationen im HTTP-Header prüfen, welche Version des Web-Browser genutzt wird. Allerdings ist das Hintergrundsystem auf die Kooperation des Web-Browsers angewiesen und kann diese Informationen nicht authentisch ermitteln. Es kann davon ausgegangen werden, dass der Nutzer seinerseits keinen unerlaubten Web-Browser mit einer gefälschten Browser-Kennung verwendet, da er sich an die Nutzungsbedingungen hält (vgl. OSP.User).

Prüfaspekt	Kurzfassung des Prüfaspekts	Prüftiefe	Anmerkungen
O.Arch_9	Verwendung von dem Stand der Technik entsprechenden HTTP-Server-Headern.	CHECK	Der Evaluator prüft, ob entsprechende HTTP-Server-Header verwendet werden. Falls keine dem Stand der Technik entsprechenden HTTP-Server-Header verwendet werden, ist dies in der Risikobewertung zu berücksichtigen.

4.3.3 Testcharakteristik zu Prüfaspekt (3): Quellcode

Tabelle 6: Testcharakteristik: Quellcode

Prüfaspekt	Kurzfassung des Prüfaspekts	Prüftiefe	Anmerkungen
O.Source_1	Prüfung von Eingaben vor Verwendung.	CHECK	Der Evaluator prüft, ob alle Eingaben vor ihrer Verarbeitung dem Stand der Technik entsprechend geprüft werden. Eingaben meinen jegliche Art von Daten, die in die Anwendung hineinfließen. Das sind zum Beispiel Nutzereingaben, Eingaben aus Drittanbieterkomponenten etc.
O.Source_2	Nutzung einer Escape-Syntax bei strukturierten Daten.	CHECK	Der Evaluator prüft, ob eine Escape-Syntax von strukturierten Daten für alle Eingaben vorhanden ist. Schadhafte Zeichen sind kontextabhängig zu betrachten. Im Datenbank-Kontext sind beispielsweise Hochkommata oder Prozentzeichen gegebenenfalls schadhaft, während im Web/HTML Kontext eher Tag-Klammern (<) schadhaft sind. Grundsätzlich muss die Input-Validierung daher kontextbezogen stattfinden. Wird eine potenziell schädliche Eingabe erkannt, muss sie entweder bereinigt/maskiert oder abgelehnt/verworfen werden. Das Verwerfen sollte dem Bereinigen vorgezogen werden. Sofern vorher maskierte oder bereinigte Eingaben weitergegeben werden, müssen diese so maskiert oder enkodiert werden, dass sie im Kontext der Weitergabe keine schädlichen Effekte haben.
O.Source_3	Keine sensiblen Daten in Meldungen.	CHECK	Der Evaluator prüft, ob sensible Daten über Fehlermeldungen oder Log-Dateien einsehbar werden.

Prüfaspekt	Kurzfassung des Prüfaspekts	Prüftiefe	Anmerkungen
O.Source_4	Kontrollierte Behandlung und Dokumentation von Ausnahmen (Exceptions).	EXAMINE	Der Evaluator prüft durch Quelltextanalyse und praktische Tests die kontrollierte Behandlung und Dokumentation von Exceptions.
O.Source_5	Abbruch des Zugriffs auf sensible Daten bei Exceptions.	EXAMINE	Der Evaluator prüft den Zugriff auf sensible Daten bei Ausnahmen im Programmablauf. Jeglicher identifizierte Zugriff muss in der Risikobewertung betrachtet werden.
O.Source_6	Deaktivierung von unterstützenden Entwicklungsoptionen in der Produktiv-Version.	EXAMINE	Der Evaluator überprüft die produktive Anwendung auf Rückstände von Optionen zur Unterstützung der Entwicklung.
O.Source_7	Verwendung moderner Sicherheitsmechanismen.	EXAMINE	Der Evaluator prüft, ob bei der im Verlauf der Entwicklung und Bereitstellung der Anwendung moderne Sicherheitsmechanismen verwendet wurden.
O.Source_8	Verwendung von Werkzeugen zur statischen Codeanalyse.	CHECK	Der Evaluator prüft durch Quelltextanalyse und Befragung des Herstellers, ob bei der Entwicklung Werkzeuge zur statischen Codeanalyse eingesetzt wurden. Wurden keine Werkzeuge zur statischen Codeanalyse verwendet, muss dies in der Risikobewertung betrachtet werden.

Prüfaspekt	Kurzfassung des Prüfaspekts	Prüftiefe	Anmerkungen
O.Source_9	Kontrollierte Verwendung von URL-Weiterleitungen.	EXAMINE	Der Evaluator stellt sicher, dass URL-Weiterleitungen kontrolliert verwendet werden. Keinesfalls darf der Link auf eine externe Seite hierbei die Session-ID enthalten. Ist die Liste der Weiterleitungs-URLs bekannt, darf auch nur auf diese weitergeleitet werden. Wenn möglich kann in diesen Fällen eine indizierte Liste auch serverseitig gehalten werden und die Weiterleitungs-URL dann über einen Index ermittelt werden. Eine Weiterleitung kann auch über eine dem Benutzer explizit angezeigte Webseite erfolgen, so dass der Benutzer selbst den Link vor dem aktiven Klick prüfen kann. Lokale Weiterleitungen sind daraufhin zu prüfen, dass die Ziel-URL nicht auf eine externe Seite führt. Wenn vermeidbar, sollte es dem Nutzer nicht selbst möglich sein, die Eingabe der Weiterleitungs-URL vorzunehmen. Ist es nicht vermeidbar, dass ein Nutzer selbst die Weiterleitungs-URL eingeben darf, muss diese umfassend daraufhin geprüft werden, dass sie gültig, zur Web-Anwendung passend und für den Nutzer zulässig ist. Wenn implementiert soll bevorzugt auf Whitelisting von erlaubten Adressen gesetzt werden und nicht auf Blacklisting.

Prüfaspekt	Kurzfassung des Prüfaspekts	Prüftiefe	Anmerkungen
O.Source_10	Vorbeugende Maßnahmen zum Schutz von Funktionalitäten außerhalb der eigenen Entwicklungshoheit.	EXAMINE	Der Evaluator stellt durch Quelltextanalyse und praktische Tests sicher, dass die Web-Anwendung dem Stand der Technik entsprechende Maßnahmen ergreift, um ein ausführen von injizierten Funktionalitäten zu verhindern. Beispielsweise kann die Web-Anwendung als Gegenmaßnahme gegen Cross-Site-Request-Forgery (CSRF) Angriffe einen SessionCode (auch Synchronizer Token Pattern (STP) oder CSRF-Token) in jede URL als weiteren zufälligen (d.h. nicht erratbaren) Parameter in einem HiddenField einfügen. Auf Basis des SessionCodes kann das Hintergrundsystem die Gültigkeit der Anfrage zusätzlich prüfen.
O.Source_11	Keine sensiblen Daten in der URL.	EXAMINE	Der Evaluator stellt durch Quelltextanalyse und praktische Tests sicher, dass sensible Daten ausschließlich über sichere, dem Stand der Technik entsprechende Methoden verarbeitet werden und zu keiner Zeit in der URL vorkommen.

4.3.4 Testcharakteristik zu Prüfaspekt (4): Drittanbieter-Software

Tabelle 7: Testcharakteristik: Drittanbieter-Software

Prüfaspekt	Kurzfassung des Prüfaspekts	Prüftiefe	Anmerkungen
O.TrdP_1	Abhängigkeiten durch Drittanbieter-Software.	CHECK	Der Hersteller stellt eine Liste der eingesetzten Drittanbieter-Software inkl. der verwendeten Versionen bereit. Der Evaluator prüft die bereitgestellte Liste auf Vollständigkeit.
O.TrdP_2	Verwendung der aktuellen Version bei Drittanbieter-Software.	CHECK	Der Evaluator prüft die in O.TrdP_1 bereitgestellte Liste auf Aktualität der verwendeten Drittanbieter-Software-Versionen. Diese Abwägungen zu den gewählten Versionen werden in der Risikobewertung berücksichtigt.

Prüfaspekt	Kurzfassung des Prüfaspekts	Prüftiefe	Anmerkungen
O.TrdP_3	Herstellerprüfung von Drittanbieter-Software auf Schwachstellen.	CHECK	Der Hersteller stellt eine Übersicht der letzten Schwachstellenanalyse der eingesetzten Drittanbieter-Software bereit. Diese wird vom Evaluator geprüft und in der Risikobewertung berücksichtigt. Zusätzlich prüft der Evaluator, ob der Hersteller bei Auftreten von Schwachstellen eine Mitigationsstrategie im Rahmen einer angemessenen Grace-Period bereitstellt.
O.TrdP_4	Sicherheitskonzept für zeitnahes Einspielen von Sicherheitsupdates für Drittanbieter-Software.	CHECK	Der Evaluator prüft das Vorhandensein eines solchen Konzeptes. Eine inhaltliche Prüfung ist im Rahmen der TR nicht erforderlich. Zusätzlich prüft der Evaluator, ob der Hersteller eine Mitigationsstrategie bereitstellt.
O.TrdP_5	Prüfung auf Vertrauenswürdigkeit der Quelle von Drittanbieter-Software.	CHECK	Der Evaluator prüft die Maßnahmen des Herstellers zur Verifikation der Vertrauenswürdigkeit von Drittanbietern.
O.TrdP_6	Keine Weitergabe von sensiblen Daten an Drittanbieter-Software.	EXAMINE	Der Evaluator prüft durch eine Quelltextanalyse und praktische Tests, dass keine Weitergabe von sensiblen Daten an Drittanbieter-Software vorgenommen wird. Eine Ausnahme hierzu bietet die Weitergabe von Daten, die für den primären oder rechtmäßigen Zweck der Anwendung erforderlich ist (beispielsweise Drittanbieter-Software zur Transportverschlüsselung). Risiken, die aus einer Nichteinhaltung resultieren, sind in der Risikobewertung zu berücksichtigen.
O.TrdP_7	Validierung eingehender Daten über Drittanbieter-Software.	CHECK	Der Evaluator prüft, ob eingehende Daten über Drittanbieter-Software gemäß O.Source_1 behandelt werden und Sicherheitsfunktionen vorhanden sind.

Prüfaspekt	Kurzfassung des Prüfaspekts	Prüftiefe	Anmerkungen
O.TrdP_8	Prüfung der Wartung von verwendeter Drittanbieter-Software.	CHECK	Der Evaluator prüft, ob die verwendete Drittanbieter-Software vom Hersteller aktiv gepflegt wird. Eine Software gilt als nicht mehr gewartet, sofern sicherheitskritische Verwundbarkeiten bekannt sind, jedoch nicht innerhalb einer angemessenen Frist repariert worden sind.

4.3.5 Testcharakteristik zu Prüfaspekt (5): Kryptographische Umsetzung

Tabelle 8: Testcharakteristik: Kryptographische Umsetzung

Prüfaspekt	Kurzfassung des Prüfaspekts	Prüftiefe	Anmerkungen
O.Cryp_1	Keine fest einprogrammierten Schlüssel oder anderweitige Geheimnisse.	EXAMINE	Der Evaluator prüft, ob fest einprogrammierte geheime, bzw. private Schlüssel eingesetzt werden.
O.Cryp_2	Nur bewährte Implementierungen bei kryptographischen Primitiven.	EXAMINE	Der Evaluator prüft die Liste der verwendeten Krypto-Implementierungen gegen den aktuellen Stand der Technik (vgl.[TR02102-2]).
O.Cryp_3	Passende Wahl der kryptographischen Primitive.	EXAMINE	Der Evaluator prüft die Abwägungen des Herstellers zur Wahl der kryptographischen Primitive und prüft, ob diese dem aktuellen Stand der Technik entsprechen (vgl. [TR02102-1])
O.Cryp_4	Zweckbindung kryptographischer Schlüssel.	EXAMINE	Der Evaluator prüft die verwendeten kryptographischen Schlüssel auf ihre Zweckgebundenheit. Es wird der Zweck nach Schutz durch Verschlüsselung und Authentisierung unterschieden.
O.Cryp_5	Nutzung von starken kryptographischen Schlüsseln.	EXAMINE	Der Evaluator prüft die Stärke der verwendeten Schlüssel gegen den aktuellen Stand der Technik (vgl. [TR02102-1]).

4.3.6 Testcharakteristik zu Prüfaspekt (6): Authentisierung und Authentifizierung

Tabelle 9: Testcharakteristik: Authentisierung, Authentifizierung und Autorisierung

Prüfaspekt	Kurzfassung des Prüfaspekts	Prüftiefe	Anmerkungen
O.Auth_1	Herstellerkonzept zur Authentisierung von Anwendungssitzungen.	CHECK	Der Evaluator prüft das vom Hersteller bereitgestellt Konzept zur Authentisierung, Autorisierung und Beenden der Anwendungssitzung. Er bewertet die Güte der eingesetzten Verfahren Anhand des aktuellen Standes der Technik.
O.Auth_2	Getrennte Realisierung von Authentisierungsmechanismen und Autorisierungsfunktionen.	EXAMINE	Der Evaluator prüft und bewertet die getroffenen Maßnahmen zur Trennung von Autorisierungs- und Authentisierungsmechanismen. Sollte keine Trennung der Mechanismen vorgenommen sein oder die getroffenen Maßnahmen nicht ausschließlich vom Hintergrundsystem durchgesetzt werden, sind die Abwägungen des Herstellers zu prüfen und in der Risikobewertung zu berücksichtigen.
O.Auth_3	Zwei-Faktor-Authentifizierung.	EXAMINE	Der Evaluator prüft durch Quelltextanalyse und praktische Tests das Vorhandensein der Zwei-Faktor-Authentifizierung. Insbesondere prüft er, ob die verwendeten Faktoren aus unterschiedlichen Kategorien stammen (Wissen, Besitz, Inhärenz).

Prüfaspekt	Kurzfassung des Prüfaspekts	Prüftiefe	Anmerkungen
O.Auth_4	Zusätzliche Informationen bei Bewertung des Authentisierungsvorgangs einbeziehen.	EXAMINE	Der Evaluator prüft das Vorhandensein und die Güte von zusätzlichen Informationen zur Bewertung eines Authentisierungsvorgangs. Solche Informationen können beispielsweise über die Invalidierung/Löschung von Schlüsseln bei Änderung von Merkmalen biometrischer Systeme oder eine Prüfung auf Änderung von biometrischen Metadaten umgesetzt werden. Eine Prüfung auf Konformität zum Datenschutz der erhobenen Informationen ist im Rahmen der TR nicht erforderlich, eine zusätzliche Prüfung ist daher empfehlenswert. Werden keine zusätzlichen Informationen zur Bewertung verwendet, prüft der Evaluator die Abwägungen des Herstellers. Diese sind in der Risikobewertung zu berücksichtigen.
O.Auth_5	Information des Benutzers über ungewöhnliche Anmeldeversuche.	CHECK	Der Evaluator prüft, ob dem Nutzer leicht zugänglich die Möglichkeit gegeben wird, Informationen zu Anmeldevorgängen nachzuvollziehen. Ist das nicht der Fall, sind die Abwägungen des Herstellers zu prüfen und in der Risikobewertung zu berücksichtigen.
O.Auth_6	Verhinderung des Ausprobierens von Login-Parametern.	CHECK	Der Evaluator validiert, dass ein Ausprobieren von Login-Parametern verhindert wird. Dies kann beispielsweise durch Verzögerung nachfolgender Login-Versuche oder den Einsatz von sogenannten Captchas erreicht werden.
O.Auth_7	Erneute Authentifizierung bei unterbrochener Anwendung.	CHECK	Der Evaluator validiert, dass nach einer der Anwendung angemessenen Zeit, in der sie in den Hintergrundmodus versetzt wurde, eine erneute Authentifizierung erfolgen muss. Die Güte der geforderten Authentifizierung muss dem Vertrauensniveau angemessen sein (vgl. O.Auth_3).

Prüfaspekt	Kurzfassung des Prüfaspekts	Prüftiefe	Anmerkungen
O.Auth_8	Erneute Authentifizierung nach angemessenen Zeit in der sie nicht aktiv verwendet wurde.	CHECK	Der Evaluator validiert, dass nach einer der Anwendung angemessenen Zeit, in der sie nicht aktiv verwendet wurde, eine erneute Authentifizierung erfolgen muss. Die Güte der geforderten Authentifizierung muss dem Vertrauensniveau angemessen sein (vgl. O.Auth_3).
O.Auth_9	Erneute Authentifizierung nach angemessenen Zeit in der sie dauerhaft aktiv verwendet wurde.	CHECK	Der Evaluator validiert, dass nach einer der Anwendung angemessenen Zeit, in der sie dauerhaft aktiv verwendet wurde, eine erneute Authentifizierung erfolgen muss. Die Güte der geforderten Authentifizierung muss dem Vertrauensniveau angemessen sein (vgl. O.Auth_3).
O.Auth_10	Ausreichende Authentifizierung des Nutzers für Änderung der Authentisierungsdaten.	EXAMINE	Der Evaluator prüft, ob er ohne angemessene Authentifizierung die Authentisierungsdaten verändern kann. Dies betrifft auch einen Ablauf zum Passwort zurücksetzen. Beruht dieser Ablauf bspw. auf Sicherheitsabfragen, darf die Antwort nicht einfach zu erraten oder gar aus möglicherweise öffentlichen Informationen ermittelbar sein (z.B. Mädchenname der Mutter).
O.Auth_11	Authentifizierung an der Schnittstelle zwischen Anwendung und Hintergrundsystem.	CHECK	Der Evaluator prüft, ob die Anwendung eine Authentifizierung des Hintergrundsystems unterstützt.
O.Auth_12	Schutz von Authentisierungsdaten	CHECK	Der Evaluator prüft, ob Authentisierungsdaten als sensible Daten gemäß den Anforderungen der TR behandelt werden.
O.Auth_13	Invalidierung von Authentisierungsdaten durch den Anwender.	CHECK	Der Evaluator prüft, ob die Anwendung dem Nutzer ermöglicht, ein oder alle zuvor ausgestellten Authentisierungsdaten ungültig zu machen.
O.Auth_14	Benachrichtigung des Hintergrundsystems über beendete Anwendungssitzungen durch die Anwendung.	CHECK	Der Evaluator prüft, ob das Hintergrundsystem bei einer ordnungsgemäßen Beendigung der Anwendungssitzung durch die Anwendung informiert wird.

Prüfaspekt	Kurzfassung des Prüfaspekts	Prüftiefe	Anmerkungen
O.Auth_15	Information des Nutzers bei Änderung der Zugangsparameter.	CHECK	Der Evaluator prüft, ob der Nutzer über die zuletzt hinterlegten, gültigen Kontaktdaten bei einer Änderung der Zugangsparameter informiert wird. Darüber hinaus stellt er sicher, dass der Nutzer über diesen Weg die Möglichkeit bekommt die Änderung zu sperren und neue Zugangsparameter zu setzen. Erfolgte die Änderung der Zugangsparameter durch einen Angreifer, kann der tatsächlich berechtigte Nutzer über diesen Mechanismus eventuell einen größeren Schaden verhindern, in dem die neuen Zugangsparameter gesperrt werden und somit auch der Angreifer keinen Zugang besitzt. Daher ist ein besonderes Augenmerk auf die notwendige erneute Authentifizierung des berechtigten Benutzers zu legen.
O.Auth_16	Information des Nutzers über das Restrisiko der Speicherung der Login-Credentials im Web-Browser.	CHECK	Der Evaluator prüft, ob der Nutzer in leichter und verständlicher Form über das Restrisiko der Speicherung von Login-Credentials im Web-Browser oder einem anderen externen Programm informiert wird.
O.Pass_1	Durchsetzung starker Passwortrichtlinien.	CHECK	Der Evaluator prüft, ob Passwortrichtlinien, welche dem aktuellen Stand der Technik entsprechen, eingesetzt werden. Andernfalls sind die Abwägungen des Herstellers zu prüfen und in der Risikobewertung zu berücksichtigen.
O.Pass_2	Anzeige der Stärke des verwendeten Passworts.	EXAMINE	Der Evaluator prüft, ob dem Nutzer die Stärke des verwendeten Passworts angezeigt wird. Ist dies der Fall, prüft er durch Quelltextanalyse und praktische Tests, ob dadurch Informationen über das Password oder dessen Güte im Anwendungsspeicher verbleiben.

Prüfaspekt	Kurzfassung des Prüfaspekts	Prüftiefe	Anmerkungen
O.Pass_3	Möglichkeit zur Änderung des Passwortes.	CHECK	Der Evaluator prüft, ob der Nutzer die Möglichkeit hat, sein Passwort zu ändern und verifiziert, dass diese Funktionalität nicht zweckentfremdet werden kann.
O.Pass_4	Protokollierung und Information über Änderungen und Zurücksetzen von Passwörtern.	CHECK	Der Evaluator prüft das Vorhandensein und die Güte von zusätzlichen Informationen zur Protokollierung von Änderungen und dem Zurücksetzen von Passwörtern.
O.Pass_5	Verwendung von kryptographisch sicheren Hashing-Algorithmen und Salts zur Speicherung der Passwörter.	EXAMINE	Der Evaluator prüft, ob Passwörter in der Anwendung gespeichert werden. Er verifiziert, dass die verwendeten Schutzmechanismen dem aktuellen Stand der Technik und den Anforderungen an Hash-Funktionen, Anzahl an Iterationen und Salts genügen (vgl. [TR02102-1]). In der Risikobewertung werden Maßnahmen, die Brute-Force-Angriffe verlangsamen, berücksichtigt.

4.3.7 Testcharakteristik zu Prüfaspekt (7): Datensicherheit

Tabelle 10: Testcharakteristik: Datenspeicherung und Datenschutz

Prüfaspekt	Kurzfassung des Prüfaspekts	Prüftiefe	Anmerkungen
O.Data_1	Maximale Sicherheit bei Werkseinstellung.	CHECK	Der Evaluator prüft die Standardeinstellungen der Anwendung bei ihrer Installation. Das umfasst unter anderem die Berechtigungen des Betriebssystems, welche die Anwendung einfordert. Die Berechtigungen müssen dem Zweck der Anwendung dienen und dürfen erst angefragt werden, sobald sie Verwendung finden.

Prüfaspekt	Kurzfassung des Prüfaspekts	Prüftiefe	Anmerkungen
O.Data_2	Verschlüsselung aller sensiblen Daten in Exports.	CHECK	Der Evaluator prüft, ob es dem Nutzer möglich ist sensible Daten von der Anwendung unverschlüsselt zu exportieren. Ist dies der Fall prüft der Evaluator, ob der Nutzer angemessen auf die daraus resultierenden Risiken aufmerksam gemacht wird.
O.Data_3	Zugriff auf sensible Daten durch Dritte.	EXAMINE	Der Evaluator prüft, ob die Anwendung Ressourcen zur Verfügung stellt, über die Dritte Zugriff auf sensible Daten erhalten können. Dies umfasst Daten in geteilten Speicherbereichen, Diensten oder Interfaces über die sensible Daten bereitgestellt werden.
O.Data_4	Löschung aller erhobenen sensiblen Daten nach Abschluss der Verarbeitung durch die Anwendung.	CHECK	Der Evaluator prüft, ob Daten über den Zeitraum ihrer Verarbeitung hinaus in der Anwendung gehalten werden. Daten, die nicht mehr genutzt werden, müssen sicher gelöscht werden.
O.Data_5	Erhebung, Speicherung und Verarbeitung von ausschließlich für den Zweck der Anwendung notwendigen Daten.	CHECK	Der Evaluator prüft, welche Daten von der Anwendung erhoben, gespeichert und verarbeitet werden und stellt diese dem Zweck der Anwendung gegenüber.
O.Data_6	Speicherung und Verarbeitung von sensiblen Daten.	CHECK	Der Evaluator prüft, welche Daten die Web-Anwendung permanent speichert bzw. verarbeitet. Er ermittelt das Risiko, das durch eine solche Speicherung und Verarbeitung in der Anwendung entsteht und nimmt es in die Risikobewertung mit auf.

Prüfaspekt	Kurzfassung des Prüfaspekts	Prüftiefe	Anmerkungen
O.Data_7	Entfernung von Metadaten mit Datenschutzrelevanz.	CHECK	Der Evaluator prüft, ob die Anwendung Daten erheben kann, die Metadaten enthalten. In diesem Fall prüft der Evaluator, ob Metadaten mit Datenschutzrelevanz vor der Weiterverarbeitung, wie beispielsweise dem Transfer an das Hintergrundsystem, entfernt werden.
O.Data_8	Zugriffsbeschränkung bei der Erhebung von sensiblen Daten.	EXAMINE	Der Evaluator prüft, ob erhobene sensible Daten anderen Anwendungen auf dem Gerät verfügbar gemacht werden oder Daten in öffentlichen Verzeichnissen gespeichert werden.
O.Data_9	Zugriffsbeschränkung bei der Erhebung von sensiblen Daten.	EXAMINE	Der Evaluator prüft, ob erhobene sensible Daten anderen Anwendungen auf dem Gerät verfügbar gemacht werden oder Daten in öffentlichen Verzeichnissen gespeichert werden. Dies schließt insbesondere Caches, Autokorrektur- und Autovervollständigungs-verfahren mit ein. Ist dies der Fall, prüft der Evaluator die Abwägungen des Herstellers und berücksichtigt diese in der Risikobewertung.
O.Data_10	Kein Export sensibler Daten in die Zwischenablage.	CHECK	Falls die Anwendung einen Export sensibler Daten in die Zwischenablage des Betriebssystems erlaubt, muss der Abfluss dieser Daten in der Risikobewertung berücksichtigt werden.
O.Data_11	Kein Export von sensiblen Daten aus der Quelle.	EXAMINE	Der Evaluator prüft, ob sensible Daten, bei denen keine Notwendigkeit für einen Export besteht, trotzdem exportierbar sind. Dies umfasst unter anderem private kryptografische Schlüssel.

Prüfaspekt	Kurzfassung des Prüfaspekts	Prüftiefe	Anmerkungen
O.Data_12	Information des Nutzers über die Risiken bei Speicherung des Bildschirms.	CHECK	Der Evaluator prüft, ob der Nutzer angemessen über die Risiken von Bildschirmaufnahmen informiert wird.
O.Data_13	Information des Nutzers über die Risiken einer aktiven Verbindung mit dem Hintergrundsystem bei gesperrtem Endgerät.	CHECK	Der Evaluator prüft, ob der Nutzer angemessen über die Risiken einer aktiven Verbindung zum Hintergrundsystem bei gesperrtem Endgerät informiert wird.
O.Data_14	Enterfernen oder anderweitiges unzugänglich machen aller sensiblen Daten im Browser bei Beendigung der Web-Anwendung.	CHECK	Der Evaluator prüft, ob nach der Beendigung der Web-Anwendung Daten im Browser zurückbleiben. Ist dies der Fall, prüft der Evaluator weiterhin, ob diese Daten sensible Informationen beinhalten oder Rückschlüsse auf sensible Daten zulassen.
O.Data_15	Möglichkeit zum Löschen oder anderweitigen unzugänglich machen aller sensiblen Daten der Anwendung.	EXAMINE	Der Evaluator validiert, dass dem Nutzer die Möglichkeit gegeben wird alle sensiblen Daten vollständig zu löschen oder unzugänglich zu machen. Darüber hinaus prüft er die Wirksamkeit der getroffenen Maßnahmen durch praktische Tests.
O.Data_16	HTTP-Only-Flag bei Cookies.	CHECK	Der Evaluator stellt sicher, dass für alle Cookies, auf die nicht mittels JavaScript zugegriffen wird, das HTTP-Only-Flag gesetzt ist.
O.Data_17	Secure-Flag bei Cookies.	CHECK	Der Evaluator stellt sicher, dass für alle Cookies, die sensible Daten enthalten, das Secure-Flag gesetzt ist.
O.Data_18	Autovervollständigungsfunktion bei Formularfeldern.	CHECK	Der Evaluator stellt sicher, dass für alle Formularfelder, in die sensible Daten eingegeben werden, die Autovervollständigungsfunktion deaktiviert ist.

Prüfaspekt	Kurzfassung des Prüfaspekts	Prüftiefe	Anmerkungen
O.Data_19	Vermeidung von Domain-Cookies.	CHECK	Der Evaluator stellt sicher, dass persistierte Daten für weitere Hosts einer Domain unlesbar sind.

4.3.8 Testcharakteristik zu Prüfaspekt (8): Kostenpflichtige Ressourcen

Tabelle 11: Testcharakteristik: Kostenpflichtige Ressourcen

Prüfaspekt	Kurzfassung des Prüfaspekts	Prüftiefe	Anmerkungen
O.Paid_1	Anzeige kostenpflichtiger Leistungen und Ressourcen.	CHECK	Der Evaluator validiert, dass alle kostenpflichtigen Leistungen und Ressourcen eindeutig als solche erkennbar sind.
O.Paid_2	Einverständnis des Nutzers vor der Verwendung kostenpflichtiger Leistungen.	CHECK	Der Evaluator validiert, dass alle kostenpflichtigen Leistungen ausschließlich nach Bestätigung durch den Nutzer erbracht werden können.
O.Paid_3	Einverständnis des Nutzers vor einer Zugriffsanforderung auf kostenpflichtige Ressourcen oder kostenpflichtige Leistungen.	CHECK	Der Evaluator validiert, dass die Nutzung von Diensten, die zusätzliche Kosten für den Nutzer verursachen können (z.B. das Versenden von SMS), ausschließlich nach Abgabe einer Einverständniserklärung des Nutzers möglich ist.
O.Paid_4	Dauerhaftes Einverständnis des Nutzers auf häufig verwendete, kostenpflichtige Leistungen oder Ressourcen.	CHECK	Falls die Anwendung ein dauerhaftes Einverständnis des Nutzers für den Zugriff auf kostenpflichtige Ressourcen fordert, prüft der Evaluator, ob dies für den primären Zweck der Anwendung erforderlich ist (vgl. O.Purp_1).
O.Paid_5	Entzug des Einverständnisses ermöglichen.	CHECK	Der Evaluator prüft, ob die Anwendung eine Liste mit allen vom Nutzer gegebenen Einverständniserklärungen anzeigt und diese nachträglich geändert werden kann.

Prüfaspekt	Kurzfassung des Prüfaspekts	Prüftiefe	Anmerkungen
O.Paid_6	Ablage der sensiblen Transaktionshistorie im Hintergrundsystem.	EXAMINE	Der Evaluator prüft über praktische Tests und Quelltextanalyse, ob eine Transaktionshistorie in der Anwendung vorgehalten wird. Die Transaktionshistorie sollte im Hintergrundsystem sicher gespeichert werden und aus der Anwendung einsehbar sein. Wenn die Transaktionshistorie in der Anwendung selber gespeichert wird, ist in einer Risikobewertung darzustellen, inwieweit die Sicherheit der gespeicherten Daten gewährleistet werden kann.
O.Paid_7	Profilbildung durch Nachverfolgung der Zahlungsströme durch Dritte.	CHECK	Der Evaluator prüft, ob über die Nachverfolgung von Zahlungsströmen Rückschlüsse auf die Eigenschaften oder das Verhalten des Nutzers möglich sind. Die Abwägungen des Herstellers bei potenziellen Rückschlüssen sind in der Risikobewertung zu berücksichtigen.
O.Paid_8	Anzeige der Übersicht der entstandenen Kosten.	CHECK	Der Evaluator prüft, ob die Anwendung dem Nutzer eine Übersicht der entstandenen Kosten anbietet. Falls die Kosten aufgrund einzelner Zugriffe erfolgt sind, prüft der Evaluator, ob die Anwendung einen Überblick der Zugriffe aufführt.
O.Paid_9	Validierung von getätigten Bezahlvorgängen im Hintergrundsystem.	EXAMINE	Der Evaluator prüft durch Quelltextanalyse und praktische Tests, ob die Anwendung eigenständig Bezahlungen validiert und beispielsweise kostenpflichtige Funktionen freischalten kann.
O.Paid_10	Anforderungen bei Zahlverfahren von Drittanbietern.	CHECK	Der Evaluator prüft die Zahlverfahren durch Drittanbieter. Sowohl bei Drittanbieter-Software als auch bei Web-Diensten wird geprüft, dass keine sensiblen Nutzerdaten an den Zahlungsdienstleister abfließen (z.B., dass der Titel der gebuchten Leistung keine sensiblen Informationen enthält).

4.3.9 Testcharakteristik zu Prüfaspekt (9): Netzwerkkommunikation

Tabelle 12: Testcharakteristik: Netzwerkkommunikation

Prüfaspekt	Kurzfassung des Prüfaspekts	Prüftiefe	Anmerkungen
O.Ntwk_1	Netzwerkkommunikation durchgängig mit gegenseitiger Authentisierung verschlüsselt.	EXAMINE	Der Evaluator validiert, dass ausschließlich gegenseitig authentisierte, verschlüsselte Kommunikation zwischen der Anwendung und anderen Komponenten möglich ist.
O.Ntwk_2	Konfiguration der verschlüsselten Verbindung gemäß aktuellem Stand der Technik.	EXAMINE	Der Evaluator validiert, dass die in O.Ntwk_1 beschriebene Kommunikation dem Stand der Technik (siehe [TR02102-2]) entspricht. Dies umfasst auch Methoden, die jegliche Übertragung von Daten außerhalb des TLS-Kanals verhindern. Beispielsweise kann mittels HTTP Strict Transport Security (HSTS) sichergestellt werden, dass zwischen Browser und dem die Webanwendung anbietenden Hintergrundsystem ausschließlich HTTPS eingesetzt wird. Mittels Content Security Policy (CSP) kann ergänzend sichergestellt werden, dass Inhalte nur aus einer definierten Liste von Quellen geladen werden können.
O.Ntwk_3	Sichere Kommunikationskanäle nur mit Betriebssystem-Funktionen.	EXAMINE	Der Evaluator prüft, ob für den Aufbau sicherer Kommunikationskanäle auf Betriebssystem -Funktionen zurückgegriffen wird. Alternativ kann Drittanbieter-Software verwendet werden, die den in Kapitel 3.1.4 beschriebenen Anforderungen genügen. Eigene Implementierungen zum Aufbau sicherer Kommunikationskanäle sind nicht zulässig.

Prüfaspekt	Kurzfassung des Prüfaspekts	Prüftiefe	Anmerkungen
O.Ntwk_4	Prüfung der Zertifikatsketten.	EXAMINE	Der Evaluator prüft durch praktische Tests und Quelltextanalysen die Wirksamkeit der verwendeten Zertifikatsprüfung. Sollte der Hersteller keine Zertifikatsprüfung umgesetzt haben, prüft der Evaluator die Abwägungen des Herstellers zu den Auswirkungen einer Zertifikatsprüfung auf die Vertraulichkeit der Daten und die Verfügbarkeit der Anwendung. Die entstehenden Restrisiken sind in der Risikoanalyse festzuhalten.

4.3.10 Testcharakteristik zu Prüfaspekt (10): Plattformspezifische Interaktionen

Tabelle 13: Testcharakteristik: Plattformspezifische Interaktionen

Prüfaspekt	Kurzfassung des Prüfaspekts	Prüftiefe	Anmerkungen
O.Plat_1	Geräteschutz für die Nutzung der Anwendung erforderlich.	CHECK	Der Evaluator prüft, ob der Hersteller eine Nutzung ohne aktivierten Geräteschutz zulässt. Ist das der Fall, prüft der Evaluator, ob der Nutzer angemessen über die daraus resultierenden Risiken aufgeklärt wird, und er berücksichtigt die Abwägungen des Herstellers in der Risikobewertung.
O.Plat_2	Nur Anforderung der für den primären Zweck notwendigen Berechtigungen.	CHECK	Der Evaluator prüft die von der Anwendung geforderten Berechtigungen und bestätigt, dass diese für die Erfüllung des primären Zwecks der Anwendung (O.Purp_1) erforderlich sind.
O.Plat_3	Hinweis auf Zweck der Berechtigungen und Auswirkungen bei Nichterteilung.	CHECK	Der Evaluator prüft, ob die Anwendung auf den Zweck der geforderten Berechtigungen hinweist.

Prüfaspekt	Kurzfassung des Prüfaspekts	Prüftiefe	Anmerkungen
O.Plat_4	Keine sensiblen Daten in Meldungen oder Benachrichtigungen.	EXAMINE	Der Evaluator prüft anhand von Quelltextanalyse und generierten Log-Nachrichten, ob die Anwendung sensible Daten in diese Nachrichten schreibt. Sollten die geloggten Daten Rückschlüsse auf den Nutzer zulassen, muss dieser Datenabfluss in der Risikobewertung berücksichtigt werden.
O.Plat_5	Option zur Anzeige von Meldungen/Benachrichtigungen mit sensiblen Daten.	CHECK	Falls die Anwendung die Möglichkeit zur Anzeige von Meldungen mit sensiblen Daten bietet, prüft der Evaluator, ob diese standardmäßig deaktiviert ist. Weiterhin prüft er, ob der Nutzer bei Aktivierung dieser Option angemessen über die daraus resultierenden Risiken aufgeklärt wird. Die Abwägungen des Herstellers, solche Optionen anzubieten, sind in der Risikobewertung zu berücksichtigen.
O.Plat_6	Nachladen aktiver Inhalte.	CHECK	Der Evaluator prüft, ob die Komponenten das Nachladen aktiver Inhalte unterbinden oder auf Quellen unter der Kontrolle des Herstellers beschränken. Die Auswahl der zugelassenen Quellen wird in der Risikoanalyse berücksichtigt.
O.Plat_7	Information des Nutzers über den Verbleib von sensiblen Daten im Arbeitsspeicher.	CHECK	Der Evaluator prüft, ob der Nutzer angemessen über die Möglichkeit des Verbleibs von sensiblen Daten im Arbeitsspeicher informiert und über die daraus resultierenden Risiken aufgeklärt wird. Diese Informationspflicht schließt die Option aus O.Plat_8 mit ein.
O.Plat_8	Information des Nutzers über erforderliche Sicherheitsmaßnahmen zur Anwendung, Bibliotheken und Plattformen.	CHECK	Der Evaluator prüft, ob der Nutzer über selbst durchführbare Sicherheitsmaßnahmen informiert und ggf. angeleitet wird. Der Evaluator bewertet, ob die Maßnahmen ausreichend sind, um Restrisiken zu begrenzen.

4.3.11 Testcharakteristik zu Prüfaspekt (11): Resilienz

Tabelle 14: Testcharakteristik: Resilienz

Prüfaspekt	Kurzfassung des Prüfaspekts	Prüftiefe	Anmerkungen
O.Resi_1	Informationen zum sicheren Umgang mit der Anwendung.	CHECK	Der Evaluator prüft, ob die Anwendung „Best-Practices“ bereitstellt. Er bestätigt, dass vorhandene „Best-Practices“ dem aktuellen Stand der Technik entsprechen.
O.Resi_2	Information des Nutzers bei Verwendung von Geräten mit eingeschränkten Sicherheitsleistungen.	CHECK	Der Evaluator überprüft, ob der Nutzer angemessen über die Risiken einer Verwendung von Geräten, deren Betriebssystem in keinem vom Betriebssystemhersteller vorgesehenen Betriebszustand ist, informiert wird.

5 Sicherheitsstufen und Risikoanalyse

Grundlage für das Prüfurteil soll ein dokumentiertes Risikomanagementverfahren sein. Als allgemeine Referenz werden BSI Standard 200-3 [BSI200-3], ISO 27005 [ISO27005] und Anhang B der Common Criteria Evaluation Methodology [CEM] genannt. Das Prüflabor darf nach Abstimmung ein vergleichbares, auf eine IT-Sicherheitsanwendung ausgerichtetes Risikomanagementverfahren einsetzen.

Die TR-Prüfer führen eine methodische Risikoanalyse durch, die mindestens folgende Schritte umfassen muss:

1. Sicherheitsproblem vollständig aufarbeiten – Ausgangspunkt der Risikoanalyse sind die Bedrohungen, Annahmen und Policies der Anwendung (Kapitel 2). Der TR-Prüfer etabliert eine vollständige Liste aller sensiblen Daten, die in der Anwendung erfasst, erzeugt oder genutzt werden. Sensible Daten, die alleine im Hintergrundsystem verarbeitet werden, sind durch die Annahme A.Backend abgedeckt.
2. Schutzbedarf feststellen – die IT-Sicherheit betrachtet generell den Schutzbedarf hinsichtlich Vertraulichkeit, Integrität, Authentizität und Verfügbarkeit. Der TR-Prüfer klassifiziert den jeweiligen Schutzbedarf aller verarbeiteten Daten, vgl. Anhang B in [ISO27005]. Die Daten im Rahmen der TR werden anhand ihrer Kritikalität unterschieden (vgl. Tabelle 15).
3. Risikoszenarien bewerten – Der TR-Prüfer führt unter Berücksichtigung der etablierten Gegenmaßnahmen eine Bewertung von Risikoszenarien durch. Es muss dafür ein dokumentierter, ganzheitlicher Ansatz auf die sensiblen Daten der Anwendung durchgeführt werden, beispielsweise [ISO27005], Abschnitt 8.3 und Anhänge C/D/E.

In die Bewertung durch den TR-Prüfer geht ein, welche Schutzmaßnahmen im Produkt realisiert sind, und deren Effektivität (Beispielsweise Maßnahmen gegen Brute-Force Angriffe auf Login Credentials). Ebenfalls werden Vorgaben für die sichere Nutzung berücksichtigt, sofern diese dem Nutzer ausreichend dargelegt sind. Ob das Sicherheitsproblem angemessen behandelt wird, schätzen die TR-Prüfer anhand der Schwierigkeit ermittelter Angriffspfade ein. (Die Schwierigkeit eines Angriffs ersetzt dabei die in der ISO 27005 referenzierte Eintrittswahrscheinlichkeit eines Risikos.)

Häufig werden für die Angriffsbewertung auf mobile Anwendungen folgende Bewertungsprinzipien eingesetzt:

1. Zeitbasierter Ansatz – Der Prüfer schätzt den zeitlichen Aufwand eines Angreifers die bestehenden Gegenmaßnahmen auszuhebeln. Der Hersteller versichert, dass vor Ablauf dieser Zeit eine neue Produktversion mit neuem Schlüsselmaterial bereitgestellt wird (z.B. spätestens monatlich) und die Anwendung ist so beschaffen, dass Angriffe nur an der aktuellsten Produktversion ausgeführt werden können. In diesem Szenario wird eine Ausnutzung des Angriffspfads durch ein rechtzeitiges Update unterbunden.
2. Reaktiver Ansatz – Hier analysiert der Prüfer die effektive Bekämpfung der Risikoszenarien mittels proaktiven Monitorings / Reaktion. Beispielsweise werden Betriebsparameter erfasst und der Zugriff auf sensible Daten wird abgewehrt, sofern diese auf absichtliche Modifikationen hindeuten. Extern vom Hersteller selbst-realisierte Schutzmechanismen müssen im Rahmen der TR-Prüfung mitbetrachtet werden.

Der Prüfer muss aufgrund der ermittelten Restrisiken ein Urteil abgeben, inwiefern das von der TR adressierte Sicherheitsproblem adäquat erfüllt wird. Tabelle 15 zeigt die Anforderungen je Datum. Eine Zertifizierung kann nur erteilt werden, falls die TR-Prüfung ergibt, dass die Anforderungen für alle Daten erfüllt werden.

Diese TR dient primär der Bewertung von Anwendungen, wie sie in Kapitel 1.3.1 definiert sind. Bei solchen Anwendungen ist der Schaden beim Verlust von Finanzdaten oft nicht zu beziffern, unter anderem weil eine einmal stattgefundene Offenbarung nicht mehr rückgängig gemacht werden kann. Anwendungen, die nach dieser TR evaluiert werden, können allerdings auch andere sensible Daten enthalten, die gegen

Offenbarung geschützt werden müssen. Das Sicherheitsniveau dieser Daten kann ggf. unter dem der Finanzdaten liegen (vgl. Tabelle 15). Die Klassifizierung der Sicherheitsstufen für die einzelnen Daten ist mit dem BSI im Einzelfall abzustimmen. Hierbei kann auf Risikoabschätzungen basierend auf etablierten Standards zurückgegriffen werden.

Tabelle 15: Anforderung anhand der Daten-Kritikalität

Kritikalität	Beschreibung	Anforderung
Sehr hoch	Eine Verletzung des Schutzbedarfs führt zu einem nicht zu beziffernden oder potenziell schwerwiegenden Schaden für den Dateninhaber.	Die realisierten Maßnahmen werden als wirksam erachtet, sämtliche Risikoszenarien ohne Restrisiken auszuräumen.
Hoch	Eine Verletzung des Schutzbedarfs führt zu einem hohen oder mittleren Schaden für den Dateninhaber.	Die realisierten Maßnahmen reduzieren die Risikoszenarien erheblich. Der TR-Prüfer muss die Durchführung verbleibender Angriffe bewerten und deren Auswirkungen dokumentieren. Im Einzelfall ist das Restrisiko darzustellen und kann Auflagen in der Nutzung der Zertifizierung verursachen.
Normal	Es kann höchstens ein geringer Schaden eintreten.	Die realisierten Maßnahmen reduzieren die Risikoszenarien. Der TR-Prüfer muss die Durchführung verbleibender Angriffe bewerten und Restrisiken offenlegen.

Anhang A: Schutzbedarf sensibler Datenelemente

Abhängig von der realisierten Anwendung und der Kritikalität der jeweils verarbeiteten Datenelemente kann ein unterschiedlicher Schutzbedarf notwendig sein. Personenbezogene Daten unterliegen dem Datenschutz und dürfen nur bei Zweckbindung und nach Einverständnis verarbeitet werden, vgl. Abschnitt 3.1.1. Die Sensibilität verarbeiteter Datenelemente wird in der folgenden Tabelle bestimmt.

Tabelle 16: Schutzbedarf sensibler Datenelemente

Information	Sensibel	Übertragung an Hintergrundsystem erlaubt	Ablage außerhalb einer sicheren Umgebung erlaubt	Bemerkungen
Anwendungsdaten	Ja	Ja	Ja	-
Eingabedaten (von extern, dritter Partei, einer Drittanbieter-Software, über Tastatur oder von Gerätesensoren)	Nein, falls nicht in anderer Kategorie speziell erfasst	Ja	Ja	Vorbehandlung, u.a. Größenprüfung, Escaping-Sequenzen (je nach Weiterverarbeitung)
Zugangsdaten	Ja	Ja	Ja	z.B. salted hashing. Zulässig sind Bibliotheken für Authentifizierung.
Kryptographische Schlüssel der Anwendung	Ja	Nein	Nein[6]	Zulässig ist die Nutzung in Bibliotheken zur Kryptographie und Session-Handling.

[6] Ausgenommen sind Public Keys oder kryptografische Schlüssel von Drittanbietersoftware, sofern diese nicht der Kontrolle des Anwendungs-Entwicklers unterliegen und mobile Endgeräte, die nicht über eine sichere Umgebung (z. B. embedded Secure Element/Secure Enclave/Trusted Execution Environment) verfügen.

Information	Sensibel	Übertragung an Hintergrundsystem erlaubt	Ablage außerhalb einer sicheren Umgebung erlaubt	Bemerkungen
Aggregierte Anwendungsdaten z.B. Therapiebericht als PDF	Ja	Ja	Ja	Eine Anzeige darf nur mit integriertem Viewer erfolgen. Die Implementierung soll eine Speicherung auf dem Gerät vermeiden. Eine Speicherung ist ausschließlich verschlüsselt erlaubt. Die für die Therapieform erforderliche Ausleitung soll über einen sicheren Kanal erfolgen.
Öffentliche Zertifikate	Nein	Ja	Ja	-

Abkürzungsverzeichnis

Tabelle 17: Abkürzungsverzeichnis

App	Applikation
A.*	Assumption (Annahme)
BSI	Bundesamt für Sicherheit in der Informationstechnik
CSP	Content Security Policy
CSRF	Cross-Site-Request-Forgery
EU	Europäische Union
GPS	Global Positioning System
HSTS	HTTP Strict Transport Security
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
IKT	Informations- und Kommunikationstechnologien
iOS	Betriebssystem des Unternehmens Apple für mobile Geräte
IoT	Internet of Things
IP	Internet Protokoll
IT	Informationstechnik
O.*	Objective (Prüfaspekt)
OSP.*	Organizational Security Policies (Organisatorische Sicherheitspolitiken)
PDF	Portable Document Format
SDK	Software Development Kit
SMS	Short Message Service

SPD	Security Problem Definition
T.*	Threat (Bedrohung)
TR	Technische Richtlinie
TLS	Transport Layer Security
URL	Uniform Resource Locator
WLAN	Wireless Local Area Network

Literaturverzeichnis

[ASVS]

The OWASP Foundation, „Application Security Verification Standard“, Version 4.0, verfügbar unter https://www.owasp.org/images/d/d4/OWASP_Application_Security_Verification_Standard_4.0-en.pdf

[BSI-CS-120]

Bundesamt für Sicherheit in der Informationstechnik, „Sicherheitsmaßnahmen beim Einsatz aktiver Inhalte - Verwendung aktiver Inhalte durch Anbieter von Webanwendungen“, Version 3.0, verfügbar unter https://www.allianz-fuer-cybersicherheit.de/SharedDocs/Downloads/Webs/ACS/DE/BSI-CS/BSI-CS_120.html

[CEM]

„Common Methodology for Information Technology Security Evaluation – Evaluation methodology”, April 2017, Version 3.1, Revision 5, verfügbar unter https://www.commoncriteriaportal.org/files/ccfiles/CEMV3.1R5.pdf

[GDR18]

we are social, „Global Digital Report 2018“, Version Januar 2018, verfügbar unter https://wearesocial.com/de/blog/2018/01/global-digital-report-2018

[gemSpec_IDP_Sek]

gematik, „Spezifikation Sektoraler Identity Provider“, verfügbar unter https://gemspec.gematik.de/docs/gemSpec/gemSpec_IDP_Sek/latest/

[ISO27005]

BS ISO/IEC 27005:2011, Information technology - Security techniques – Information security risk management

[KCC-C5]

Bundesamt für Sicherheit in der Informationstechnik, „Kriterienkatalog Cloud Computing“, Version 2020, verfügbar unter https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/CloudComputing/Anforderungskatalog/2020/C5_2020.pdf?__blob=publicationFile&v=2

[NIST80057]

National Institute of Standards and Technology, „Recommendation for Key Management“, Revision 5, verfügbar unter https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-57pt1r5.pdf

[RFC7469]

C. Evans, C. Palmer, R. Sleevi, Google Inc., „Public Key Pinning Extension for HTTP“, Version April 2015, verfügbar unter https://tools.ietf.org/html/rfc7469

[TR03107-1]

Bundesamt für Sicherheit in der Informationstechnik, „Elektronische Identitäten und Vertrauensdienste im E-Government Teil 1“, Version 1.1.1, verfügbar unter https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03107/TR-03107-1.html

[TR02102-1]

Bundesamt für Sicherheit in der Informationstechnik, „ Kryptographische Verfahren: Empfehlungen und Schlüssellängen”, Version 2023-01, verfügbar unter

https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR02102/BSI-TR-02102.pdf?__blob=publicationFile&v=10

[TR02102-2]

Bundesamt für Sicherheit in der Informationstechnik, „Kryptographische Verfahren: Empfehlungen und Schlüssellängen Teil 2 – Verwendung von Transport Layer Security (TLS)", Version 2023-01, verfügbar unter https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR02102/BSI-TR-02102-2.pdf?__blob=publicationFile&v=2

[TR03174-1]

Bundesamt für Sicherheit in der Informationstechnik, „Anforderungen an Anwendungen im Finanzwesen Teil 1: Mobile Anwendungen", Version 2.0, verfügbar unter https://www.bsi.bund.de/dok/TR-03161-1

[TR03174-3]

Bundesamt für Sicherheit in der Informationstechnik, „Anforderungen an Anwendungen im Finanzwesen Teil 3: Hintergrundsysteme", Version 1.0, verfügbar unter https://www.bsi.bund.de/dok/TR-03174-3

[WSTG]

The OWASP Foundation, „Web Security Testing Guide", Version 2021, verfügbar unter https://owasp.org/www-project-web-security-testing-guide/stable/