Universität Zürich[UZH]

Merkblatt 1: Generelle Sicherheitsanforderungen an Web Applikationen

Gegenstand:	Grundsätze und Anforderungen an Web Applikationen
Datum:	07.03.2017
Herausgeber:	ITSIBE – IT – Sicherheitsbeauftragter der UZH
Ersetzt:	-
Geltungsbereich:	Gilt für die gesamte UZH
Version:	1.0

1. GRUNDLAGEN

Das vorliegende Merkblatt beinhaltet Sicherheits - Anforderungen an Web Applikationen.

2. SICHERHEITS GRUNDSÄTZE

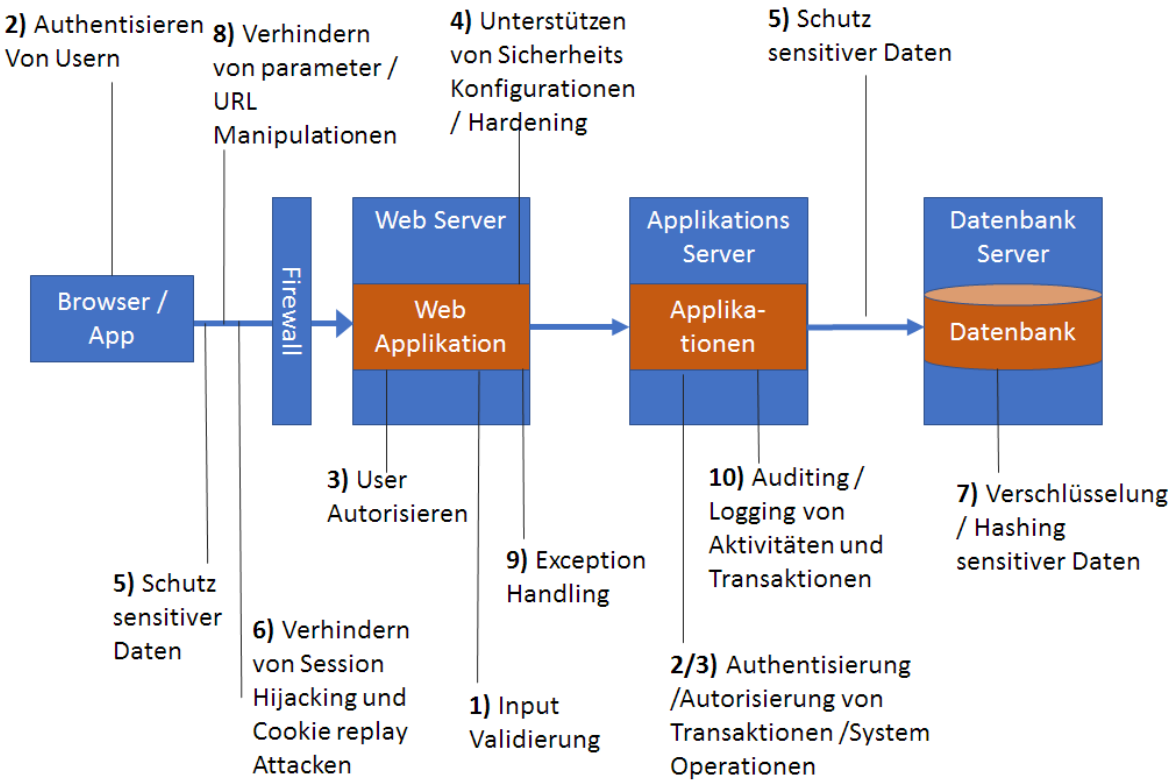
Folgende Sicherheitsgrundsätze sollen durch die Einhaltung der Anforderungen erreicht werden:

- Erstellung von Eingabe Validierungsstrategien in Web Applikationen
- Partitionierung von Websites in offene und eingeschränkte Bereiche
- Implementierung effektiver Account Management-Praktiken
- Entwicklung effektiver Authentifizierungs- und Berechtigungsstrategien
- Schutz sensibler Daten
- Schutz von Benutzersitzungen
- Vermeidung von Parametermanipulationen
- Sicherer Umgang mit Ausnahmen resp. Fehlermeldungen an Systemen / Applikationen
- Sichern der Konfigurationsverwaltungsfunktionen einer Anwendung / Server's
- Aufzählung von Audit- und Logging-Überlegungen

3. SICHERHEITSARCHITEKTUR

3.1 Überblick

Die nachfolgende Grafik zeigt auf, wo die geforderten Sicherheitsanforderungen greifen.



3.2 Beschreibung

1) Eingabe Validierung	Attacken durch Einbettung von bösartigen Strings in Abfragen, Web Form Felder, Cookies, URL und HTTP headers. Diese beinhalten auch die Ausführung von cross-site scripting (XSS), SQL injection und buffer overflow Attacken.
2) Authentisierung	Nachahmen von bestehenden Identitäten, password cracking, unberechtigtes Erlangen von höheren Systemrechten sowie Unautorisierter Zugriff auf Systeme.
3) Autorisierung	Zugriff auf vertrauliche Informationen oder über Zugriffsrechte eingeschränkter Zugriff auf Daten, Manipulation von Daten sowie Ausführung von unautorisierten Transaktionen oder Systemoperationen
4) Konfigurations Management	Unbefugter Zugriff auf Administrationsschnittstellen, Möglichkeit Konfigurationsdaten zu aktualisieren und unberechtigten Zugriff auf Benutzerkonten und Kontoprofile zu erlangen.

5) Sensitive Data	Offenlegung vertraulicher Informationen sowie Datenmanipulationen.
6) Session Management	Schwache oder keine Session Identifier führen zu session hijacking und Identitäts Nachahmung.
7) Verschlüsselung	Einschränken des Zugriffes auf vertrauliche Daten oder Kontoanmeldeinformationen oder beides
8) Parameter Manipulation	Pfad-Traversal-Attacken, Befehlsausführung und Umgehung von Zutrittskontrollmechanismen, was zur Offenlegung von Informationen, zur Erhöhung von Privilegien (Systemrechten) und zur Denial-of-Service führen kann.
9) Exception Management	Denial-of-Service und Offenlegung von sensiblen System-Details.
10) Auditing and Logging	Ignorieren von Anzeichen einer Intrusion, Unfähigkeit, die Handlungen eines Benutzers zu beweisen sowie Schwierigkeiten bei der Problemdiagnose.

4. ANFORDERUNGEN

Nachfolgend sind die wichtigsten Anforderungen bezüglich Entwicklung von Web Applikationen enthalten. Die nachfolgenden Anforderungen aus den OWASP Top Ten stellen nur die am meisten ausgenutzten Schwachstellen dar. Generell muss auch ein Leistungsanbieter / Auftragnehmer entsprechende Vorkehrungen in seinem Entwicklungsprozess analog ISO27001 / NIST treffen um eine, den heutigen Ansprüchen, entsprechend sichere Applikation entwickeln zu können.

4.1 Owasp Top 10 Kategorien – Anforderungen zur minmierung von Zugriffen durch nicht autorisierte / authentisierte Personen auf ein System

#	**Anforderung** (Schutz vor...)	**Beschreibung**	**Kategorie**	**Referenz**
1.	A1 – Injection	Injection-Schwachstellen, wie beispielsweise SQL-, OS- oder LDAP-Injection, treten auf wenn nicht vertrauenswürdige Daten als Teil eines Kommandos oder einer Abfrage von einem Interpreter verarbeitet werden. Ein Angreifer kann Eingabedaten dann so manipulieren, dass er nicht vorgesehene Kommandos ausführen oder unautorisiert auf Daten zugreifen kann.	Webapplikation/ Webserver	OWASP Top Ten - 2013
2.	A2 – Fehler in Authentifizierung und Session-Management	Anwendungsfunktionen, die die Authentifizierung und das Session-Management umsetzen, werden oft nicht korrekt implementiert. Dies erlaubt es Angreifern Passwörter oder Session-Token zu kompromittieren oder die Schwachstellen so auszunutzen, dass sie die Identität anderer Benutzer annehmen können.	Browser/ Webserver/ / Webapplikation	OWASP Top Ten - 2013
3.	A3 – Cross-Site Scripting (XSS)	XSS-Schwachstellen treten auf, wenn eine Anwendung nicht vertrauenswürdige Daten entgegennimmt und ohne entsprechende Validierung oder Umkodierung an einen Webbrowser sendet. XSS erlaubt es einem Angreifer Scriptcode im Browser eines Opfers auszuführen und somit Benutzersitzungen zu übernehmen, Seiteninhalte zu verändern oder den Benutzer	Browser/ Webapplikation	OWASP Top Ten - 2013

#	**Anforderung** (Schutz vor...)	**Beschreibung**	**Kategorie**	**Referenz**
		auf bösartige Seiten umzuleiten.		
4.	A4 – Unsichere direkte Objektreferenzen	Unsichere direkte Objektreferenzen treten auf, wenn Entwickler Referenzen zu internen Implementierungsobjekten, wie Dateien, Ordner oder Datenbankschlüssel von außen zugänglich machen. Ohne Zugriffskontrolle oder anderen Schutz können Angreifer diese Referenzen manipulieren um unautorisiert Zugriff auf Daten zu erlangen.	Browser/ Applikation/ Datenbankserver	OWASP Top Ten - 2013
5.	A5 – Sicherheits-relevante Fehlkonfiguration	Sicherheit erfordert die Festlegung und Umsetzung einer sicheren Konfiguration für Anwendungen, Frameworks, Applikations-, Web- und Datenbankserver sowie deren Plattformen. Sicherheitseinstellungen müssen definiert, umgesetzt und gewartet werden, die Voreinstellungen sind oft unsicher. Des Weiteren umfasst dies auch die regelmäßige Aktualisierung aller Software.	Webserver/ Browser/ Datenbankserver	OWASP Top Ten - 2013
6.	A6 – Verlust der Vertraulichkeit sensibler Daten	Viele Anwendungen schützen sensible Daten, wie Kreditkartendaten oder Zugangsinformationen nicht ausreichend. Angreifer können solche nicht angemessen geschützten Daten auslesen oder modifizieren und mit ihnen weitere Straftaten, wie beispielsweise Kreditkartenbetrug, oder Identitätsdiebstahl begehen. Vertrauliche Daten benötigen zusätzlichen Schutz, wie z.B. Verschlüsselung während der Speicherung oder Übertragung sowie besondere Vorkehrungen beim Datenaustausch mit dem Browser.	Datenbankserver / Applikation	OWASP Top Ten - 2013
7.	A7 – Fehlerhafte Autorisierung auf Anwendungsebene	Die meisten betroffenen Anwendungen realisieren Zugriffsberechtigungen nur durch das Anzeigen oder Ausblenden von Funktionen in	Applikation/ Datenbank	OWASP Top Ten - 2013

#	**Anforderung** (Schutz vor...)	**Beschreibung**	**Kategorie**	**Referenz**
		der Benutzeroberfläche. Allerdings muss auch beim direkten Zugriff auf eine geschützte Funktion eine Prüfung der Zugriffsberechtigung auf dem Server stattfinden, ansonsten können Angreifer durch gezieltes Manipulieren von Anfragen ohne Autorisierung trotzdem auf diese zugreifen.		
8.	A8 – Cross-Site Request Forgery (CSRF)	Ein CSRF-Angriff bringt den Browser eines angemeldeten Benutzers dazu, einen manipulierten HTTP- Request an die verwundbare Anwendung zu senden. Session Cookies und andere Authentifi-zierungsinformationen werden dabei automatisch vom Browser mitgesendet. Dies erlaubt es dem Angreifer Aktionen innerhalb der betroffen Anwendungen im Namen und Kontext des angegriffen Benutzers auszuführen.	Webserver/ Webapplikation	OWASP Top Ten - 2013
9.	A9 – Nutzung von Komponenten mit bekannten Schwachstellen	Komponenten wie z.B. Bibliotheken, Frameworks oder andere Softwaremodule werden meistens mit vollen Berechtigungen ausgeführt. Wenn eine verwundbare Komponente ausgenutzt wird, kann ein solcher Angriff zu schwerwiegendem Datenverlust oder bis zu einer Serverübernahme führen. Applikationen, die Komponenten mit bekannten Schwachstellen einsetzen, können Schutz-maßnahmen unterlaufen und so zahlreiche Angriffe und Auswirkungen ermöglichen.	Alle involvierten Systeme	OWASP Top Ten - 2013
10.	A10 – Ungeprüfte Um- und Weiterleitungen	Viele Anwendungen leiten Benutzer auf andere Seiten oder Anwendungen um oder weiter. Dabei werden für die Bestimmung des Ziels oft nicht vertrauenswürdige Daten verwendet. Ohne eine entsprechende Prüfung können Angreifer ihre Opfer auf Phishing-Seiten oder	Webserver / Applikationsserver	OWASP Top Ten - 2013

#	**Anforderung** (Schutz vor...)	**Beschreibung**	**Kategorie**	**Referenz**
		Seiten mit Schadcode um- oder weiterleiten.		

4.2 Weitere funktionale Anforderungen

#	Anforderung	Beschreibung
11.	Rollenbasierendes Zugriffskonzept	Die Applikation muss ein Zugriffskonzept implementieren /umsetzen können welches nach dem need to know Prinzip Zugriffe auf Funktionen / Informationen einschränken kann. Das System muss ebenfalls eine Gruppenverwaltung sowie Rollenkonzepte unterstützen um nebst der Authentisierung auch die Autorisierung steuern zu können.
12.	Autorisierung	Die Autorisierung auf die Applikation muss durch die Applikaiton erzwungen werden
13.	Authentisierung	Die Applikation muss die Möglichkeit bieten weitere Authentisierungsmechnismen einzubinden(AAI) ohne die Autorisierungsvorgaben zu unterwandern.
14.	Nicht erfolgreiche Authentisierungsversuche	• Die Applikation unterstützt eine Limitierung der Anzahl nicht erfolgreicher Login Versuche • Die Applikation unterstützt das Locken eines Accounts nach X fehlerhaften Login Versuchen. ➔ Gelockte Accounts können via User Administration freigeschalten werden.
15.	Fehlerinformationen der Applikation	Fehlerinformationen der Applikation werden in einer try / Catch Funktion abgefangen. In der Catch Funktion muss eine neutrale Fehlerinformation auf der Benutzeroberfläche angezeigt werden. →Keine Standardfehler wie bspw. von Datenbanken / Webserver etc. welche Version und Produkt oder das Aufzeigen von fehlerhaften SQL Queries sollen dabei auf der Benutzeroberfläche angezeigt werden.
16.	Letztes erfolgreiches Login	Das System zeigt nach einem Login an, wann sich der benutzte Account zum letzten Mal erfolgreiche authentisiert hat.
17.	Keine gleichzeigten / parallelen Sessions des gleichen Accounts	Das System verhindert gleichzeitige Sessions von unterschiedlichen Browserinstanzen.
18.	Session Time out / Session Management / Session Authentizität	➔ Die Applikation unterstützt ein Session Time out den man konfigurieren kann (Standardwerte 180 Min ohne Aktivität, danach erfolgt ein Log out) ➔ Sessions welche unterbrochen werden (bspw. Ausfall Clientseitiger Internet Zugang) müssen geschlossen werden. Keine hängende Sessions / re – etablierte Sessions. ➔ Session sind pro Login über einen eindeutigen Identifyer referenziert. Keine Doubletten möglich.

#	Anforderung	Beschreibung
19.	Logout	Die Applikation stellt beim erfolgreichen Logout sicher, dass alle clientseitigen und schützenswerten Informationen (u.a. Browser Cache / Cookies) gelöscht werden. ➔ Bspw. ein erneutes Login nach erfolgreichem abmelden via der Backfunktion des Browsers darf nicht zu einem automatischem Re – Login führen.
20.	Logging / Auditierung des Systemes	Die Applikation schreibt Logfiles welche sicherheitsrelevanten Inhalt haben in ein Logfile welches an ein SIEM weitergeleitet werden kann (Syslog Format)
21.	Business Contingency	Das System unterstützt eine redundante Auslegung **bei Bedarf** sowie ein failover von einer aktiven Instanz auf eine Hot by Instanz ohne Datenverlust.
22.	Verschlüsselung	Die Lösung unterstützt eine Verschlüsselung von sensitiven Inhalten (bspw. DB Inhalt) bei Bedarf (vorbereitet).
23.	Applikations Partitionierung	Die Applikation separiert User Funktionen von Applikationsmanagement Funktionen
24.	Isolierung von sicherheitsrelevanten Funktionen	Die Applikation trennt sicherheitsrelevante Funktionen von nicht sicherheitrelevanten Funktionen.
25.	Logging	Die Applikation zeichnet wichtige User / System Transaktionen auf und hinterlegt diese in einem Logfile (Syslog).
26.	WAF – Webapplication Firewall	Die Applikation kann mit Webproxies und Webapplication Firewalls umgehen.

5. KONTAKT

Für Fragen, Ergänzungen, Präzisierungen und andere Anregungen bitte direkt den IT – Sicherheitsbeautragten informieren.

Zürich, 07.03.2017

Sacha Schweizer

security@uzh.ch

IT – Sicherheitsbeauftragter der Universität