Sicherheit von Webanwendungen

Lerneinheit 4: Die OWASP Top Ten 2025

Prof. Dr. Christoph Karg

Studiengang Informatik
Hochschule Aalen

Sommersemester 2026

11.5.2026

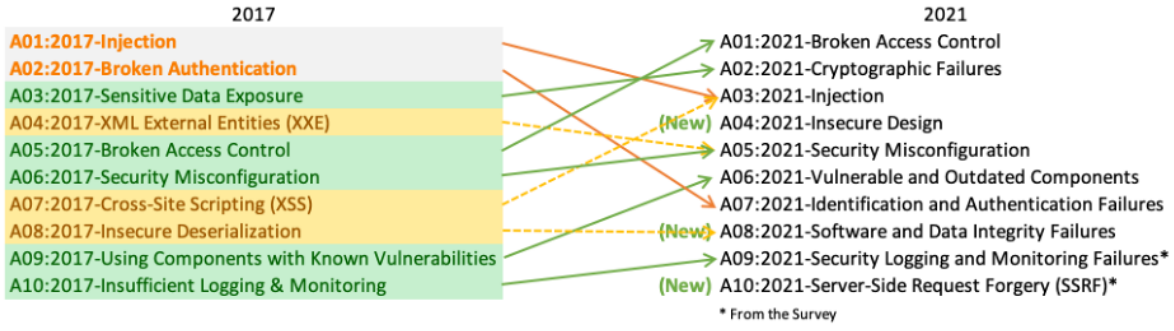
Einleitung

- Das Open Worldwide Application Security Project (OWASP) ist eine Non-Profit-Organisation, die sich der Sicherheit von Software widmet.
- Hinter OWASP steht eine weltweite Community von Firmen, Bildungseinrichtungen und Einzelpersonen.
- Die Arbeit von OWASP umfasst zahlreiche Projekte rund um das Thema Anwendungssicherheit.
- Projekte mit hohem Reifegrad werden als „Flagship Projects“ ausgewiesen.
- Ein Flagship Project ist die OWASP Top Ten, in der die häufigsten und gefährlichsten Schwachstellen in Webanwendungen gelistet werden.
- Für weitere Informationen siehe: `https://owasp.org`

Kategorien der OWASP Top Ten 2025

- A01:2025 – Broken Access Control
- A02:2025 – Security Misconfiguration
- A03:2025 – Software Supply Chain Failures
- A04:2025 – Cryptographic Failures
- A05:2025 – Injection
- A06:2025 – Insecure Design
- A07:2025 – Authentication Failures
- A08:2025 – Software and Data Integrity Failures
- A09:2025 – Security Logging and Monitoring Failures
- A10:2025 – Mishandling of Exceptional Conditions

Änderungen zur OWASP Top Ten 2017



Methodik

- Der Ansatz zur Ermittlung der zehn Kategorien ist stark, aber nicht ausschließlich datengestützt.
- Acht Kategorien wurden auf Basis von statistischen Daten ausgewählt.
- Zwei Kategorien (A6 und A10) stammen aus einer Umfrage, die bei Sicherheitsexperten durchgeführt wurde.
- Die National Vulnerability Database und das dort eingesetzten Common Vulnerability Scoring System (CVSS) spielen bei der Bewertung eine wichtige Rolle.
- Für jede Kategorie werden zahlreiche Querbezüge zur Common Weakness Enumeration (CWE) hergestellt.

Common Weakness Enumeration (CWE)

- Die Common Weakness Enumeration (CWE) ist eine Sammlung von Schwachstellen (engl. weakness) von Hard- und Software.
- Für jede dieser Schwachstellen lässt sich ein Schadenspotential (vulnerability) ableiten.
- Ziel: Schulung von Entwicklern und Hardware-Designern zur frühzeitigen Vermeidung der Schwachstellen.
- Für weitere Details siehe: `https://cwe.mitre.org`

Datenfaktoren

- CWEs Mapped ⇝ Anzahl der CWE-Einträge, die der Kategorie zugeordnet wurden
- Incident Rate ⇝ Anteil der Anwendungen in Prozent, die durch die CWEs verwundbar waren
- Avg Weighted Exploit ⇝ Durchschnittlicher Exploit-Subscore der CVSSv2/CVSSv3 der CVEs, die den CWEs zugeordnet wurden
- Avg Weighted Impact ⇝ Durchschnittlicher Impact-Subscore der CVSSv2/CVSSv3 der CVEs, die den CWEs zugeordnet wurden

Datenfaktoren (Forts.)

- Coverage ⇝ Anteil der Anwendungen, die von den teilnehmenden Organisationen bezüglich eines CWE-Eintrags untersucht wurden
- Total Occurrences ⇝ Gesamtzahl der Anwendungen, die von CWEs der Kategorie betroffen waren
- Total CVE ⇝ Gesamtzahl der CVEs in der NVD Datenbank, denen die CWEs der Kategorie zugeordnet waren

Darstellung der Informationen einer Kategorie

- Statistische Daten als Tabelle
- Ursachen für die Schwachstelle
- Tipps zur Vermeidung der Schwachstelle
- Beispiele
- Links zu weiterführenden Informationen
- Verweis auf die zugeordneten CWEs

A01:2025 – Broken Access Control

CWEs Mapped	Max Incident Rate	Avg Incident Rate	Avg Weighted Exploit	Avg Weighted Impact
40	**20.15%**	**3.74%**	**7.04**	**3.84**

Max Coverage	Avg Coverage	Total Occurrences	Total CVEs
100.00%	**42.93%**	**1.839.701**	**32.654**

Link: A01:2025

Beschreibung

- Zugriffskontrolle stellt sicher, dass die Benutzer einer Webanwendung nur im Rahmen ihrer zugewiesenen Berechtigungen agieren können.
- Fehlfunktionen führen zu:
  - Unautorisierter Offenlegung vertraulicher Informationen
  - Unautorisierte Änderung von Daten
  - Unautorisiertes Löschen von Daten
  - Unberechtigte Nutzung von Funktionen einer Webanwendung

CWE-Einträge (Auswahl)

- CWE-200: Exposure of Sensitive Information to an Unauthorized Actor ($\rightsquigarrow$ Link)
- CWE-201: Insertion of Sensitive Information Into Sent Data ($\rightsquigarrow$ Link)
- CWE-352: Cross-Site Request Forgery ($\rightsquigarrow$ Link)

Gängige Schwachstellen

- Verletzung des Prinzips der minimalen Rechte (Deny by Default)
- Umgehen von Zugriffskontrolle durch Manipulation der URL, des internen Zustands der Anwendung oder der HTML-Seite
- Zugriff auf den Account eines anderen Nutzers anhand dessen Unique Identifiers (UID)
- Zugriff auf die API ohne die erforderlichen Berechtigungen
- Manipulation von Metadaten
- Fehlkonfiguration von Cross-Origin Resource Sharing (CORS)
- Zugriff eines nicht authentifizierten Nutzers auf eine privilegierte Webseite

Gegenmaßnahmen

- Anwendung der „Deny by Default“ Richtlinie (außer bei öffentlich zugänglichen Informationen)
- Einmalige Implementierung von Mechanismen zur Zugriffskontrolle
- Deaktivieren des Zugriffs auf Verzeichnisse und Metadaten in der Konfiguration des Webservers
- Logging von abgelehnten Zugriffsversuchen und Benachrichtigung der Administratoren
- Begrenzung der Zugriffsrate auf API-Funktionen
- Löschen der Sitzungsinformationen nach dem Ausloggen des Benutzers

A02:2025 – Security Misconfiguration

CWEs Mapped	Max Incident Rate	Avg Incident Rate	Avg Weighted Exploit	Avg Weighted Impact
16	**27,70%**	**3,00%**	**7,96**	**3,97**

Max Coverage	Avg Coverage	Total Occurrences	Total CVEs
100,00%	**52,35%**	**719.084**	**1.375**

Link: A02:2025

Beschreibung

- Durch zunehmende Konfigurationsmöglichkeiten in der eingesetzten Software steigt das Risiko von Fehlkonfigurationen.
- Von den getesteten Anwendungen enthielten 4% verschiedenste Arten von Konfigurationsfehlern.
- Beispiele:
  - In einer Produktivumgebung werden bei einem Anwendungsserver die Beispielanwendungen nicht entfernt.
  - Bei einem Webserver wird die Darstellung der Verzeichnisse der Webanwendung nicht deaktiviert.
  - Der Anwendungsserver liefert detaillierte Fehlermeldungen inklusive Stack Traces aus.

CWE-Einträge (Auswahl)

- CWE-16: Configuration (⇝ Link)
- CWE-260: Password in Configuration File (⇝ Link)
- CWE-526: Cleartext Storage of Sensitive Information in an Environment Variable (⇝ Link)
- CWE-611: Improper Restriction of XML External Entity Reference (⇝ Link)

Ursachen für Verwundbarkeiten

- Die Systeme sind unzureichend bezüglich Sicherheit gehärtet.
- Auf den Systemen sind nicht benötigte Komponenten installiert bzw. aktiviert.
- Default-Zugänge wurden nicht deaktiviert bzw. die zugehörigen Passwörter wurden nicht geändert.
- Das Error Handling liefert zu detaillierte Fehlermeldungen oder Stack Traces an die Nutzer aus.
- Beim Upgrade eines Systems sind aktuelle Sicherheitsmechanismen nicht aktiviert bzw. falsch konfiguriert.
- Die eingesetzte Software ist veraltet oder enthält bekannte Schwachstellen.

Gegenmaßnahmen

- Ein wiederholbarer Prozess für die Systemhärtung ermöglicht das Aufsetzen neuer System mit entsprechender Absicherung.
- Eine minimale Plattform ohne nicht benötigte Software-Pakete, Dokumentationen und Beispielanwendungen bildet die Grundlage für den Betrieb der Webanwendung.
- Eine Checkliste zur Aktualisierung und Überprüfung der Systeme sollte für alle Komponenten erstellt werden.
- Die Anwendung sollte segmentiert werden, um eine sichere Isolierung der eingesetzten Komponenten und die Mehrmandantenfähigkeit zu erreichen.

A03:2025 – Software Supply Chain Failures

CWEs Mapped	Max Incident Rate	Avg Incident Rate	Avg Weighted Exploit	Avg Weighted Impact
6	**9,56%**	**5,72%**	**8,17**	**5,23**

Max Coverage	Avg Coverage	Total Occurrences	Total CVEs
65,42%	**27,47%**	**215.248**	**11**

Link: A03:2025

Beschreibung

- Software Supply Chain Failures sind Störungen oder Kompromittierungen im Prozess der Erstellung, Verteilung oder Aktualisierung von Software.
- Sie werden häufig durch Schwachstellen oder böswillige Änderungen in Code, Tools oder anderen Abhängigkeiten von Drittanbietern verursacht.
- Trotz dieses erweiterten Umfangs sind Lieferkettenausfälle nach wie vor schwer zu identifizieren. Lediglich 11 CVEs sind den zugehörigen CWEs zugeordnet.
- Diese Kategorie belegte in der OWASP-Top-10-Community-Umfrage den ersten Platz mit 50%.

CWE-Einträge (Auswahl)

- CWE-447: Use of Obsolete Function (⇝ Link)
- CWE-1104: Use of Unmaintained Third Party Components (⇝ Link)
- CWE-1329: Reliance on Component That is Not Updateable (⇝ Link)
- CWE-1357: Reliance on Insufficiently Trustworthy Component (⇝ Link)
- CWE-1395: Dependency on Vulnerable Third-Party Component (⇝ Link)

Ursachen für Verwundbarkeiten

- Versionen aller verwendeten Komponenten nicht sorgfältig erfasst werden einschließlich direkter und transitiver (verschachtelter) Abhängigkeiten.
- Die eingesetzte Software ist veraltet, wird vom Hersteller nicht mehr unterstützt oder enthält Schwachstellen.
- Es existiert kein Change-Management-Prozess zur Nachverfolgung von Änderungen in der Lieferkette,
- Teile der Lieferkette sind nicht gehärtet, insbesondere bezüglich Zugriffskontrolle und minimaler Rechtevergabe.
- Plattformen, Frameworks und Abhängigkeiten werden nicht zeitnah und risikobasiert aktualisiert oder gepatcht.

Gegenmaßnahmen

- Zentrale Erstellung und Verwaltung einer Software Bill of Materials (SBOM) für die gesamte Software
- Verfolgung nicht nur direkter, sondern auch transitiver Abhängigkeiten
- Kontinuierliche Überwachung von Quellen wie CVE, NVD und Open Source Vulnerabilities (OSV) auf Schwachstellen
- Komponenten ausschließlich von offiziellen (vertrauenswürdigen) Quellen über sichere Verbindungen beziehen
- CI/CD, IDEs und andere Entwickler-Tools regelmäßig aktualisieren
- Change-Management zur Erfassung von Änderungen an den eingesetzten Systemen
- Härtung von allen relevanten Systemen

A04:2025 – Cryptographic Failures

CWEs Mapped	Max Incident Rate	Avg Incident Rate	Avg Weighted Exploit	Avg Weighted Impact
32	**13,77%**	**3,80%**	**7,23**	**3,90**

Max Coverage	Avg Coverage	Total Occurrences	Total CVEs
100,00%	**47,74%**	**1.665.348**	**2.185**

Link: A04:2025

Beschreibung

- Diese Kategorie untersucht Schwachstellen, die durch fehlende oder fehlerhaft konfigurierte/implementierte kryptographische Mechanismen entstehen.
- Eine Konsequenz der Ausnutzung von Schwachstellen dieser Art ist die Offenlegung vertraulicher Informationen.
- Es ist zu prüfen, welche Daten während der Übertragung oder der Speicherung abgesichert werden müssen.
- Zu berücksichtigen sind gesetzliche Vorgaben (z.B. DSGVO) oder Regulierungen der Finanzbranche.

CWE-Einträge (Auswahl)

- CWE-259: Use of Hard-coded Password (⇝ Link)
- CWE-327: Broken or Risky Crypto Algorithm (⇝ Link)
- CWE-331: Insufficient Entropy (⇝ Link)

Zu klärende Fragen

- Gibt es Daten, die im Klartext übertragen werden?
- Sind veraltete oder schwache kryptographische Algorithmen im Einsatz?
- Wird einfache Verschlüsselung verwendet, obwohl ein Authenticated-Encryption-Verfahren besser geeignet wäre?
- Sind Default-Passwörter oder schwache Passwörter im Einsatz?
- Wurden Passwörter in ein Repository eingecheckt?
- Werden Serverzertifikate und die zugehörige Trust Chain korrekt validiert?

Zu klärende Fragen (Forts.)

- Ist bei der Verschlüsselung ein unsicherer Betriebsmodus wie beispielsweise ECB im Einsatz?
- Werden Passwörter direkt als kryptographische Schlüssel verwendet, anstatt eine Key Derivation Function einzusetzen?
- Ist der Zufallszahlengenerator für kryptographische Zwecke geeignet und wird er korrekt eingesetzt?
- Sind Fehlermeldungen oder Seitenkanal-Informationen für Angriffe ausnutzbar?

Gegenmaßnahmen

- Identifiziere, welche Daten gemäß gesetzlichen Vorgaben, regulatorischen Vorschriften oder vertraglichen Vereinbarungen zwecks vertraulich sensibel sind.
- Klassifiziere Data-In-Use, Data-At-Rest und Data-In-Transit.
- Speichere sensible Daten nur dann, wenn dies erforderlich ist (Datensparsamkeit).
- Stelle sicher, dass sensible Daten ausschließlich verschlüsselt gespeichert werden.
- Setze ausschließlich aktuelle und als sicher eingestufte kryptographische Verfahren ein.

Gegenmaßnahmen (Forts.)

- Speichere Passwörter unter Einsatz geeigneter Techniken ab.
- Nutze immer Authenticated-Encryption-Verfahren anstatt „normaler“ Verschlüsselung.
- Schlüssel sollten unter Einsatz eines für kryptographische Zwecke geeigneten Zufallszahlengenerators erzeugt werden und im Speicher als Byte-Arrays abgelegt werden.
- Deaktiviere Caching-Mechanismen für sensible Daten.
- Sichere zu übertragende Daten mit geeigneten Verfahren (z.B. TLS) ab.

A05:2025 – Injection

CWEs Mapped	Max Incident Rate	Avg Incident Rate	Avg Weighted Exploit	Avg Weighted Impact
27	**13,77%**	**3,08%**	**7,15**	**3,97**

Max Coverage	Avg Coverage	Total Occurrences	Total CVEs
100,00%	**42,93%**	**1.404.249**	**62,445**

Link: A05:2025

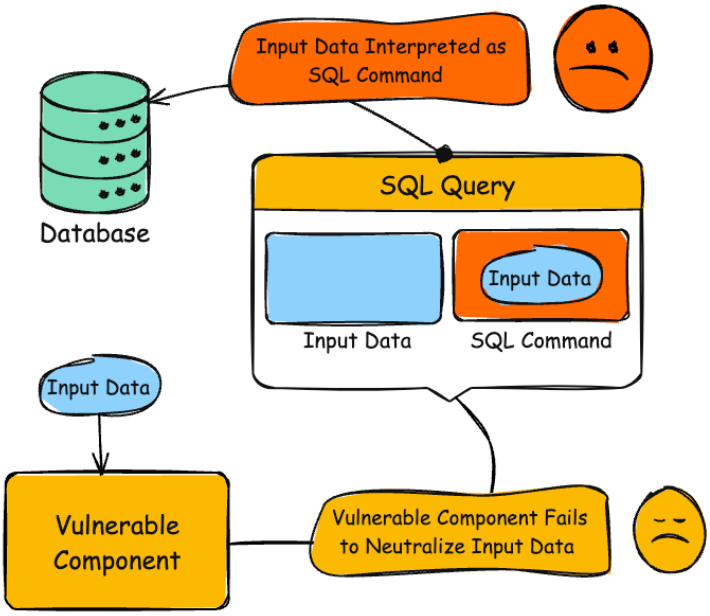
Beschreibung

- Beispiele für diese Art von Schwachstellen sind SQL, NoSQL, OS Command, Object Relational Mapping (ORM) und LDAP Injections.
- Eine Anwendung ist verwundbar, wenn Benutzereingaben nicht überprüft, gefiltert oder bereinigt werden.
- Die beste Methode, um Injection-Schwachstellen zu erkennen, sind Source Code Reviews.
- Automatisierte Tests wie z.B. Fuzzing sind empfehlenswert.

CWE-Einträge (Auswahl)

- CWE-20: Improper Input Validation (⇝ Link)
- CWE-79: Cross-site Scripting (⇝ Link)
- CWE-89: SQL Injection (⇝ Link)
- CWE-73: External Control of File Name or Path (⇝ Link)

Veranschaulichung SQL-Injection (CWE-89)



Quelle: [CWE-89]

Gegenmaßnahmen

- Um Injection Schwachstellen zu verhindern, ist die Trennung von Daten und Befehlen bzw. Anfragen erforderlich.
- Empfehlenswert ist der Einsatz einer sicheren API, die ein passendes Interface bereitstellt und den direkten Zugriff auf den Interpreter verhindert.
- Auf dem Server sollte eine positive Eingabevalidierung eingesetzt werden, z.B. unter Verwendung von regulären Ausdrücken.
- Bei SQL-Anfragen kann `LIMIT` zur Begrenzung der ausgegebenen Datensätze genutzt werden.
- Weitere Empfehlungen findet man in den entsprechenden Cheat-Sheets der OWASP.

A06:2025 – Insecure Design

CWEs Mapped	Max Incident Rate	Avg Incident Rate	Avg Weighted Exploit	Avg Weighted Impact
39	**22.18%**	**1,86%**	**6,96**	**4,05**

Max Coverage	Avg Coverage	Total Occurrences	Total CVEs
88,76%	**35,18%**	**729.882**	**7.647**

Link: A06:2025

Beschreibung

- Diese Kategorie adressiert Risiken, die durch Mängel im Design oder durch Fehler in der Architektur der Software entstehen.
- Sie zeigt den Bedarf an der Nutzung von Threat Modelling, Secure Design Patterns und Referenzarchitekturen auf.
- Unsicheres Design ist von unsicherer Implementierung zu unterscheiden.
- Per Definition kann ein unsicheres Design nicht durch eine perfekte Implementierung repariert werden.
- Eine der Ursachen für unsicheres Design ist die mangelnde Berücksichtigung/Analyse von betriebswirtschaftlichen Risiken.

Sicheres Design

- Sicheres Design steht für eine Methodik, die
  - fortlaufend Bedrohungen analysiert und
  - sicherstellt, dass der entwickelte Code robust gegen aktuelle Angriffstechniken ist.
- Threat Modeling sollte ein fester Bestandteil des Entwicklungsprozesses sein.
- Änderungen im Datenfluss, in der Zugriffskontrolle und anderen Sicherheitskontrollen müssen fortlaufend beobachtet werden.
- Sicheres Design ist nicht als Add-On zur Software-Entwicklung zu verstehen.

CWE-Einträge (Auswahl)

- CWE-256: Unprotected Storage of Credentials ($\rightsquigarrow$ Link)
- CWE-286: Incorrect User Management ($\rightsquigarrow$ Link)
- CWE-501: Trust Boundary Violation ($\rightsquigarrow$ Link)
- CWE-522: Insufficiently Protected Credentials ($\rightsquigarrow$ Link)
- CWE-657: Violation of Secure Design Principles ($\rightsquigarrow$ Link)

Gegenmaßnahmen

- Einrichtung und Nutzung eines Secure Development Lifecycles unter Einbindung von AppSec Experten
- Einrichtung und Nutzung von Secure Design Patterns
- Einsatz von Threat Modeling für Authentifizierung, Zugriffskontrolle, Geschäftslogik und Datenflüsse
- Erstellung von Unit Tests zur Prüfung, ob alle kritischen Prozesse resistent gegenüber dem Threat Model sind
- Isolierung der Tiers und Netzwerkschichten gemäß den Sicherheitsanforderungen
- Isolierung der Mandanten durch ein robustes Design
- Begrenzung der Systemressourcen für Nutzer und Dienste

A06:2021 – Authentication Failures

CWEs Mapped	Max Incident Rate	Avg Incident Rate	Avg Weighted Exploit	Avg Weighted Impact
36	**15,80%**	**2,92%**	**7,69**	**4,44**

Max Coverage	Avg Coverage	Total Occurrences	Total CVEs
100,00%	**37,14%**	**1.120.673**	**7.147**

Link: A07:2025

Beschreibung

- Diese Kategorie geht aus der Top 7 der Top Ten 2021 hervor.
- Der Focus liegt nun ausschließlich auf Schwachstellen und Konfigurationsfehler in der Authentisierung.

CWE-Einträge (Auswahl)

- CWE-259: Use of Hard-coded Password ($\rightsquigarrow$ Link)
- CWE-287: Improper Authentication ($\rightsquigarrow$ Link)
- CWE-297: Improper Validation of Certificate with Host Mismatch ($\rightsquigarrow$ Link)
- CWE-384: Session Fixation ($\rightsquigarrow$ Link)

Ursachen für Verwundbarkeiten

- Die Anwendung erlaubt automatisierte Angriffe wie z.B. Brute Force Attacken oder Credential Stuffing.
- Die Anwendung erlaubt die Nutzung schwacher Passwörter.
- Es kommen schwache oder ineffiziente Methoden zur Wiederherstellung des Accounts zum Einsatz.
- Das Verfahren zur Speicherung der Passwörter erfüllt nicht die gängigen Standards.
- Es kommt keine oder wirkungslose Mehr-Faktor-Authentisierung zum Einsatz.
- Die URL enthält Session-Informationen, die für einen Angriff nutzbar sind.
- Session-IDs werden nach erfolgreichem Logout oder einer Inaktivitätsphase nicht korrekt gelöscht.

Gegenmaßnahmen

- Falls möglich, sollte Mehr-Faktor-Authentisierung genutzt werden.
- Auf die Installation von Default-Zugängen und/oder Default-Passwörter sollte verzichtet werden.
- Es sollten Verfahren zur Erkennung schwacher Passwörter eingesetzt werden.
- Die Anforderung an die Qualität von Passwörtern sollte sich an gängigen Empfehlungen orientieren.
- Bei mehreren fehlgeschlagenen Login-Versuchen sollte der nächste Login-Versuch verzögert werden.
- Serverseitig sollte ein Session-Manager eingesetzt werden, der gängige Anforderungen erfüllt.

A08:2025 – Software and Data Integrity Failures

CWEs Mapped	Max Incident Rate	Avg Incident Rate	Avg Weighted Exploit	Avg Weighted Impact
14	**8,98%**	**2,75%**	**7,11**	**4,79**

Max Coverage	Avg Coverage	Total Occurrences	Total CVEs
78,52%	**45,49%**	**501.327**	**3.331**

Link: A08:2025

Beschreibung

- Diese Kategorie in in der OWASP Top Ten neu hinzugekommen.
- Der Fokus liegt auf getroffenen Annahmen im Zusammenhang mit Software Updates, wichtigen Daten und CI/CD Pipelines ohne die entsprechende Datenintegrität zu prüfen.
- Beispiele:
  - In der Anwendung wird Software auf nicht-vertrauenswürdigen Quellen eingesetzt.
  - Objekte oder Daten werden in einer Struktur kodiert oder serialisiert, die ein Angreifer einsehen oder modifizieren kann (Insecure Deserialization).

CWE-Einträge (Auswahl)

- CWE-829: Inclusion of Functionality from Untrusted Control Sphere (⇝ Link)
- CWE-494: Download of Code Without Integrity Check (⇝ Link)
- CWE-502: Deserialization of Untrusted Data (⇝ Link)

Gegenmaßnahmen

- Einsatz von digitalen Signaturen oder ähnlichen Verfahren zur Überprüfung des Ursprungs von Daten oder Software
- Installation von Software ausschließlich über vertrauenswürdige Quellen
- Einrichtung eines Prozesses zur Überprüfung von Änderungen am Code oder der Konfiguration der Anwendung
- Überprüfung, ob die CI/CD-Pipeline eine fehlerfreie Aufgabentrennung, Konfiguration und Zugriffskontrolle besitzt
- Implementierung von Datenserialisierung, die gängige Sicherheitsanforderungen erfüllt

A09:2025 – Security Logging and Monitoring Failures

CWEs Mapped	Max Incident Rate	Avg Incident Rate	Avg Weighted Exploit	Avg Weighted Impact
5	**11,33%**	**3,91%**	**7,19**	**2,65**

Max Coverage	Avg Coverage	Total Occurrences	Total CVEs
85,96%	**46,48%**	**260.288**	**723**

Link: A09:2025

Beschreibung

- Die Analyse von Logging oder Monitoring Informationen kann herausfordernd sein.
- Oft ist menschliche Expertise für die Bewertung erforderlich.
- Diese Kategorie wurde aufgrund einer Community-Umfrage in die Top Ten 2021 aufgenommen.
- Für diese Kategorie gibt es kaum CVE/CVSS-Daten.

CWE-Einträge (Auswahl)

- CWE-778: Insufficient Logging (⇝ Link)
- CWE-117: Improper Output Neutralization for Logs (⇝ Link)
- CWE-223: Omission of Security-relevant Information (⇝ Link)
- CWE-532: Insertion of Sensitive Information into Log File (⇝ Link)

Ursachen

- Wichtige Events wie z.B. Logins oder fehlgeschlagene Logins werden nicht protokolliert.
- Warnungen und Fehlermeldungen von Software-Komponenten erzeugen keine oder unverständliche Log-Einträge.
- Log-Daten werden nicht auf auffällige Einträge überprüft.
- Log-Daten werden nur lokal auf dem Server gespeichert.
- Es gibt keine oder nur ineffektive Prozesse zur Behandlung von Fehlermeldungen.
- Automatisierte Penetration-Test-Tools liefern keine Alarme.
- Die Anwendung kann Angriffe nicht zeitnah erkennen.

Gegenmaßnahmen

- Alle Fehler bei Logins, Zugriffen und serverseitiger Eingabe-Validierung müssen protokolliert werden.
- Die Log-Daten sollten in einem Format gespeichert werden, welches von gängiger Analyse-Software verarbeitbar ist.
- Log-Daten müssen auf eine Art und Weise enkodiert werden, dass Angriffe auf die Analyse-Software verhindert werden.
- Es muss sichergestellt sein, dass Finanz-Transaktionen ab einer bestimmten Höhe fälschungssicher protokolliert werden.
- Es wird eine Einrichtung eines Incident Response And Recovery Plans empfohlen.

A10:2025 – Mishandling of Exceptional Conditions

CWEs Mapped	Max Incident Rate	Avg Incident Rate	Avg Weighted Exploit	Avg Weighted Impact
24	**20,67%**	**2,95%**	**7,11**	**3,81**

Max Coverage	Avg Coverage	Total Occurrences	Total CVEs
100,00%	**37,95%**	**769.581**	**3.416**

Link: A10:2025

Beschreibung

- Diese Art von Schwachstellen entsteht dadurch, dass in einer Anwendung ungewöhnliche oder unvorhersehbare Situationen nicht korrekt behandelt werden.
- Angreifer nutzen gezielt fehlerhaftes Fehlerhandling aus, um Schwachstellen wie Logikfehler, Überläufe, Race Conditions oder betrügerische Transaktionen hervorzurufen.
- Ein erfolgreicher Angriff kann die Vertraulichkeit, Verfügbarkeit und Integrität eines Systems sowie seiner Daten beeinträchtigen.
- Fehler dieser Kategorie sind oft schwer zu finden und können daher eine Gefährdung über einen langen Zeitraum darstellen.

CWE-Einträge (Auswahl)

- CWE-209: Generation of Error Message Containing Sensitive Information (⇝ Link)
- CWE-369: Divide By Zero (⇝ Link)
- CWE-636: Not Failing Securely ('Failing Open') (⇝ Link)
- CWE-703: Improper Check or Handling of Exceptional Conditions (⇝ Link)
- CWE-756: Missing Custom Error Page (⇝ Link)

Ursachen

- Fehlende, mangelhafte oder unvollständige Eingabevalidierung
- Zu späte oder zu allgemeine Behandlung von Fehlern bzw. Exceptions
- Unerwartete Systemzustände, z.B. Speicherüberläufe, fehlerhafte Berechtigungen oder Netzwerkverbindungen
- Inkonsistente oder gar keine Behandlung von Exceptions

Gegenmaßnahmen

- Jeden möglichen Fehler direkt an der Stelle abfangen und behandeln, wo er auftritt, inklusive verständlicher Fehlermeldung an den Nutzer, Protokollierung des Ereignisses und ggf. Auslösung eines Alerts.
- Strenge und zentrale Eingabevalidierung implementieren sowie Fehlerhandling, Logging, Monitoring und Alerting einheitlich an einer Stelle in der Anwendung bündeln.
- Einen globalen Exception-Handler als Sicherheitsnetz einsetzen, der alle nicht explizit behandelten Ausnahmen auffängt.
- Bei fehlgeschlagenen Transaktionen die gesamte Transaktion vollständig zurückrollen („Failing Closed") (also keine „Reparatur").

Gegenmaßnahmen (Forts.)

- Rate Limiting, Ressourcenquoten und Throttling einsetzen, um Ausnahmezustände von vornherein zu verhindern und Denial-of-Service sowie Brute-Force-Angriffe abzuwehren.
- Monitoring- und Observability-Werkzeuge einsetzen, die wiederholte Fehler oder Angriffsmuster erkennen und automatisch reagieren oder blockieren können.
- Sicherheitsanforderungen für das Fehlerhandling bereits in der Entwurfsphase festlegen und durch Bedrohungsmodellierung, Code-Reviews, statische Analyse sowie Stress- und Penetrationstests absichern.

Zusammenfassung

- Die OWASP Top Ten 2025 ist eine Liste der häufigsten in Webanwendungen zu findenden Schwachstellen.
- Viele der Schwachstellen treten auf, obwohl sie einfach zu verhindern wären.
- Neben den in der Top Ten 2025 gelisteten Schwachstellen gibt es noch diverse andere.
- Schwachstellen gibt es nicht nur in Webanwendungen, sondern auch in anderen Komponenten von IT-Systemen.
- Eine weiteres erwähnenswertes OWASP Projekt ist die Top 10 for LLM Applications ($\rightsquigarrow$ Link)