

Sichere Webanwendungen

Lerneinheit 5: Common Vulnerability Scoring System

Prof. Dr. Christoph Karg

Studiengang Informatik
Hochschule Aalen



Sommersemester 2025

Gliederung

Diese Lerneinheit beschäftigt sich mit dem **Common Vulnerability Scoring System**.

Sie gliedert sich in folgende Abschnitte:

- Einleitung
- Aufgabenstellung
- CVSS Metriken
- Berechnung des CVSS Scores
- Ausblick auf CVSS 4.0
- Zusammenfassung

Common Vulnerability Scoring System (CVSS)

- Bewertungssystem für Schwachstellen
- Entwicklung des Forum of Incident Response and Security Teams (FIRST)
- An der Entwicklung beteiligt: Carnegie Mellon University, NIST, Mitre, Cisco, Microsoft, IBM, Qualys, Symantec, eBay, ...
- Einsatz in zahlreichen Sicherheitsprodukten
- Homepage: <http://www.first.org/cvss>
- Behandelte Version: 3.1 Revision 1 [1]
- Aktuelle Version: 4.0 [2]

Aufgabenstellung

Ziel: Bewertung der Gefahr, die von einer Schwachstelle ausgeht.

Kriterien:

- Art der Auswirkungen (z.B. Verlust der Vertraulichkeit)
- Aufwand zur Ausnutzung der Schwachstelle
- Gefährdungsgrad der IT-Systeme in einer konkreten Umgebung (z.B. in der Firma XY)
- Informationen zum Bekanntheitsgrad der Schwachstelle und zu bereits durchgeführten Gegenmaßnahmen

Vorteile von CVSS

- Standardisierte Art der Bewertung:
 - ▷ Einsatz einer einzigen Bewertungsmethode im gesamten Unternehmen
 - ▷ Einheitliche Priorisierung von Maßnahmen
- Offener Ansatz:
 - ▷ Die Formeln zur Bewertung sind frei zugänglich.
 - ▷ Eine Bewertung beinhaltet Informationen, mit denen das Zustandekommen nachvollziehbar ist.
- Kontextbezogene Bewertung:
 - ▷ Die Bewertung ist im Kontext eines Unternehmens durchführbar.
 - ▷ Für das Unternehmen nicht relevante Bedrohungen lassen sich aussortieren.

Bewertungsmethode hinter CVSS

- Bewertung einer Schwachstelle anhand von drei **Kategorien**:
 - ▷ Base Metric Group \rightsquigarrow Eigenschaften, die sich im Lauf der Zeit nicht ändern.
 - ▷ Temporal Metric Group \rightsquigarrow Eigenschaften, die sich im Lauf der Zeit ändern können.
 - ▷ Environmental Metric Group \rightsquigarrow Eigenschaften, die sich auf das Umfeld eines Benutzers beziehen.
- Eine **Bewertung** besteht aus:
 - ▷ Score (eine Zahl zwischen 0 und 10), die anhand der obigen Kategorien berechnet wird
 - ▷ Vektor mit textbasierten Informationen zu den Details der Bewertung

Durchführung der Bewertung

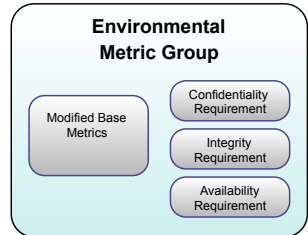
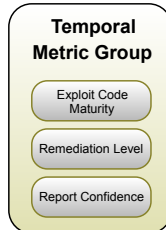
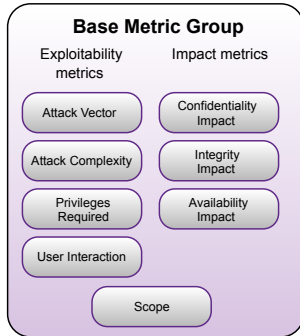
Base/Temporal Metric Group:

- Bewertung durch Sicherheitsexperten oder Herstellern von Sicherheitsprodukten
- Begründung:
 - ▷ Nur Experten besitzen das notwendige Fachwissen für eine solide Bewertung.
 - ▷ Die Bewertung hängt nicht vom Unternehmenskontext ab.

Environmental Metric Group:

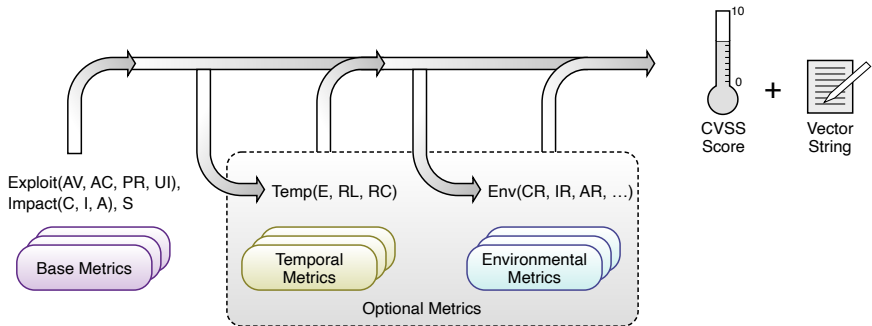
- Bewertung durch die Endnutzer
- Begründung: Bewertung hängt ausschließlich vom Unternehmenskontext ab.

Metric Groups



Quelle: [1]

Berechnung einer Bewertung



Quelle: [1]

Base Metrics

Die Base Metric Group unterteilt sich in zwei Untergruppen:

- **Exploitability Metrics** \rightsquigarrow Darstellung der Eigenschaften der verwundbaren Sache (Software/Hardware/Firmware)
- **Impact Metrics** \rightsquigarrow Darstellung der Auswirkungen eines erfolgreichen Angriffs auf die verwundbare Sache

Exploitability Metrics

Die **Exploitability Metrics** beinhalten folgende Metriken:

- Access Vector
- Access Complexity
- Priviledges Required
- User Interaction
- Scope

Zielsetzung: Bewertung des Aufwands zur Ausnutzung der Schwachstelle.

Access Vector (AV)

Ziel: Bewertung der Schwachstelle anhand des Standorts, von dem aus der Angriff durchführbar ist.

Ansatz: Je weiter der Angreifer vom Angriffsziel entfernt sein kann, desto höher ist die Bewertung.

Werte:

- Physical (P)
- Local (L)
- Adjacent (A)
- Network (N)

Access Vector — Bewertung

Physical (P):

- Der Angreifer benötigt physikalischen Zugriff auf das System.
- Beispiele: Einsatz eines manipulierten USB-Stick, Cold Boot Attacke.

Local (L):

- Der Angreifer muss lokalen Zugriff auf den Rechner haben, z.B. über ein Terminal oder eine Remote Shell.
- Beispiel: Local Privilege Escalations (sudo).

Access Vector — Bewertung (Forts.)

Adjacent (A):

- Der Angreifer muss Zugriff auf ein lokales Netzwerk haben.
- Beispiele: Lokales Ethernet-Segment, WLAN, Bluetooth.

Network (N):

- Der Angriff ist aus dem Internet durchführbar.
- Beispiel: Buffer Overflow eines Webdienstes.

Attack Complexity (AC)

Ziel: Bewertung der Bedingungen für den Zugriff, die außerhalb des Einflussbereichs des Angreifers liegen.

Ansatz: Je einfacher der Zugriff, desto höher die Gefährdung.

Werte:

- High (H)
- Low (L)

Attack Complexity — Bewertung

High (H): Die Zugriffsanforderungen für einen erfolgreichen Angriff hängen von Bedingungen ab, die außerhalb der Kontrolle des Angreifers liegen.

Beispiele:

- Vor dem eigentlichen Angriff muss der Angreifer bereits seine Privilegien erhöht oder andere Systeme getäuscht haben.
- Der Angriff erfordert den Einsatz von Social Engineering, um Details über die Konfiguration des anzugreifenden Systems zu ermitteln.
- Der Angriff erfordert eine Systemkonfiguration, die in der Praxis selten zu finden ist.
- Der Angriff hängt von einer Race Condition mit einem engen Zeitfenster ab.

Attack Complexity — Bewertung (Forts.)

Low (L): Es sind keine speziellen Anforderungen notwendig.

Beispiele:

- Das Angriffsziel ist für eine Vielzahl von Nutzern über das Internet erreichbar, z.B. Mail-Server, Web-Server.
- Der Angriff ist in der Standard-Konfiguration des Systems ausführbar.
- Der Angriff ist einfach durchführbar.
- Es existiert Software zur automatisierten Durchführung des Angriffs
- Die Durchführung des Angriffs ist nicht zeitkritisch.

Privileges Required (PR)

Ziel: Beschreibung der Zugriffsberechtigungen, die ein Angreifer besitzen muss, bevor er die Attacke ausführen kann.

Ansatz:

- Die Bewertung ist am höchsten, wenn der Angriff unautorisiert durchführbar ist.

Werte:

- None (N)
- Low (L)
- High (H)

Privileges Required — Bewertung

None (N):

- Der Angreifer kann die Attacke ohne Authentisierung durchführen.

Low (L):

- Zum Ausführung des Attacke muss der Angreifer als normaler Benutzer angemeldet sein, d.h., es sind keine besonderen Berechtigungen erforderlich.

High (H):

- Der Angreifer benötigt für die Attacke administrative Rechte, die über die Berechtigungen eines normalen Benutzers hinausgehen.

User Interaction (UI)

Ziel: Bewertung der Anforderung, dass neben dem Angreifer ein weiterer Benutzer zur erfolgreichen Durchführung des Angriffs notwendig ist.

Ansatz: Die Bewertung ist am höchsten, wenn der Angreifer die Attacke alleine durchführen kann.

Werte:

- None (N)
- Required (R)

User Interaction — Bewertung

None (N):

- Der Angriff ist durch den Angreifer ohne Unterstützung eines anderen Benutzers durchführbar.

Required (R):

- Zur erfolgreichen Ausnutzung der Schwachstelle ist die Aktion eines anderen Benutzers erforderlich.
- Beispiel: der Administrator muss auf dem System eine bestimmte Anwendung installieren.

Scope (S)

Ziel: Bewertung der Möglichkeit, über die Schwachstelle auf Ressourcen zuzugreifen, die außerhalb des Autorisierungsbereichs der Anwendung liegen.

Beispiel für einen Scope-Wechsel:

- Ein Angreifer kann durch eine Schwachstelle in einer virtuellen Maschine aus der Virtualisierungssandbox ausbrechen und beliebige Dateien auf der Festplatte der Virtualisierungsplattform löschen.
- Es liegt ein Wechsel des Autorisierungsbereichs von der virtuellen Maschine hin zum Wirtssystem vor.

Werte:

- Unchanged (U)
- Changed (C)

Scope — Bewertung

Unchanged (U):

- Die Ausnutzung der Schwachstelle betrifft nur Ressourcen, die sich im Autorisierungsbereich der verwundbaren Komponente befinden.
- Die verwundbare Komponente ist identisch mit der vom Angriff betroffenen Komponente.

Changed (C):

- Die Ausnutzung der Schwachstelle ermöglicht den Zugriff auf Ressourcen außerhalb des Autorisierungsbereichs der verwundbaren Komponente.
- Die verwundbare Komponente ist nicht identisch mit der vom Angriff betroffenen Komponente.

Impact Metrics

Die **Impact Metrics** beinhalten folgende Metriken:

- Confidentiality Impact
- Integrity Impact
- Availability Impact

Ziel: Bewertung der Folgen eines erfolgreichen Angriffs hinsichtlich Vertraulichkeit, Datenintegrität und Verfügbarkeit.

Confidentiality Impact (C)

Ziel: Bewertung der Auswirkungen eines erfolgreichen Angriffs auf die Vertraulichkeit.

Ansatz: Die Vertraulichkeit ist beeinträchtigt, wenn ein unautorisierter Zugriff auf vertrauliche Daten erfolgt. Beispiele:

- Unzureichende Verschlüsselung oder Kompromittierung von Schlüsseln
- Fehlerhafte Zugriffsberechtigungen

Werte:

- High (H)
- Low (L)
- None (N)

Confidentiality Impact – Bewertung

High (H):

- Die Ausnutzung der Schwachstelle legt alle auf dem System gespeicherten vertraulichen Daten offen.
- Der Angreifer gelangt an vertrauliche Informationen (z.B. Administrator Passwort), die das System komplett kompromittieren und uneingeschränkten Zugang ermöglichen.

Low (L):

- Durch die Ausnutzung der Schwachstelle werden Teile der vertraulichen Daten offengelegt.
- Der Angreifer gelangt an Zugangsdaten, die vertrauliche Informationen teilweise offenlegen.

None (N):

- Die Ausnutzung der Schwachstelle hat keine Auswirkungen auf die auf dem System gespeicherten vertraulichen Daten.

Integrity Impact (I)

Ziel: Bewertung der Auswirkungen eines erfolgreichen Angriffs auf die Datenintegrität und die Autorisierung beim Zugriff auf Daten.

Ansatz: Die Integrität ist beeinträchtigt, wenn die Daten nicht mehr vertrauenswürdig oder verfälscht sind.

Werte:

- High (H)
- Low (L)
- None (N)

Integrity Impact – Bewertung

High (H):

- Die Integrität der auf dem System gespeicherten Daten ist komplett verloren.
- Der Angreifer kann Schutzmechanismen für die Datenintegrität komplett ausgehebeln.

Low (L):

- Der Angreifer kann Teile der auf dem System gespeicherten Daten verändern.
- Der Angreifer kann jedoch die änderbaren Daten nicht frei wählen.

None (N):

- Die Schwachstelle stellt keine Gefährdung für die Datenintegrität dar.

Availability Impact (A)

Ziel: Bewertung der Auswirkungen eines erfolgreichen Angriffs auf die Verfügbarkeit des Systems.

Ansatz: Die Verfügbarkeit ist eingeschränkt, wenn das System die zu erfüllenden Aufgaben nur noch eingeschränkt erledigen kann.

Werte:

- High (H)
- Low (L)
- None (N)

Availability Impact – Bewertung

High (H):

- Das System wird komplett lahmgelegt.
- Der Angreifer kann einen Dienst komplett ausschalten.
- Der Angreifer kann einen Dienst zwar nicht lahmlegen, aber Zugriffe darauf unterbinden.

Low (L):

- Die Leistungsfähigkeit des Systems ist beeinträchtigt oder es kommt zu Unterbrechungen beim Zugriff auf das System.

None (N):

- Die Schwachstelle stellt keine Gefährdung für die Verfügbarkeit dar.

Temporal Metrics

Die Temporal Metric Group beinhaltet folgende Metriken:

1. Exploit Code Maturity
2. Remediation Level
3. Report Confidence

Zielsetzung:

- Bewertung der Verlässlichkeit der Informationen über die Schwachstelle
- Bewertung der Verfügbarkeit von Gegenmaßnahmen, z.B. Sicherheitsupdates

Beachte: Die Bewertungen können sich im Laufe der Zeit ändern.

Exploit Code Maturity (E)

Ziel: Bewertung des aktuellen Status hinsichtlich der Ausnutzbarkeit der Schwachstelle.

Ansatz: Die Verfügbarkeit von einfach zu benutzenden Angriffswerkzeugen (z.B. Shell-Skripte oder Shell-Codes) erhöht den Wert.

Werte:

- Not Defined (X)
- High (H)
- Functional (F)
- Proof-Of-Concept (P)
- Unproven (U)

Exploitability – Bewertung

Not Defined (X):

- Es wurde kein Wert angegeben.

High (H):

- Es gibt einen automatisiert durchführbaren Exploit, z.B., Skript, Virus, Wurm, ...
- Die Schwachstelle ist ohne Exploit direkt ausnutzbar.

Exploitability – Bewertung (Forts.)

Functional (F):

- Es gibt einen funktionierenden Exploit, der in den meisten Fällen zu einem erfolgreichen Angriff führt.

Proof-of-Concept (P):

- Die Ausnutzung der Schwachstelle wurde als Proof-of-Concept nachgewiesen.
- Der Exploit ist nicht in der Breite nutzbar.

Unproven (U):

- Es gibt zur Zeit keinen Exploit-Code.
- Der Exploit ist rein theoretischer Natur.

Remediation Level (RL)

Ziel: Bewertung des aktuellen Status der Beseitigung der Schwachstelle.

Ansatz: Je höher die Qualität der Gegenmaßnahmen ist, desto niedriger ist der Wert dieser Metrik.

Werte:

- Not Defined (X)
- Unavailable (U)
- Workaround (W)
- Temporary Fix (T)
- Official Fix (O)

Remediation Level – Bewertung

Not Defined (X):

- Es wurde kein Wert angegeben.

Unavailable (U):

- Es gibt keine Lösung zur Schließung der Schwachstelle.
- Es gibt eine Lösung, aber diese ist in der Praxis nicht anwendbar.

Remediation Level – Bewertung (Forts.)

Workaround (W):

- Es gibt einen Workaround aus einer inoffiziellen Quelle, z.B. von Sicherheitsexperten oder Benutzern.

Temporary Fix (T):

- Der Hersteller hat einen vorläufigen Patch oder einen Workaround bereitgestellt.

Official Fix (O):

- Der Hersteller hat einen Patch oder ein Update bereitgestellt, der die Schwachstelle komplett schließt.

Report Confidence (RC)

Ziel: Bewertung des Vertrauensgrad über die Existenz der Schwachstelle.

Ansatz: Die Bewertung ist umso höher, je detaillierter die Schwachstelle und deren Ausnutzung beschrieben wird.

Werte:

- Not Defined (X)
- Confirmed (C)
- Reasonable (R)
- Unknown (U)

Report Confidence – Bewertung

Not Defined (X):

- Es wurde kein Wert angegeben.

Confirmed (C):

- Es gibt detaillierte Berichte über die Schwachstelle.
- Es wurde Source Code veröffentlicht, anhand dem man die Korrektheit der Schwachstelle nachvollziehen kann.
- Der Hersteller hat die Existenz der Schwachstelle bestätigt.

Report Confidence – Bewertung (Forts.)

Reasonable (R):

- Es wurden mehrere Details über die Schwachstelle veröffentlicht.
- Es liegt noch kein Code vor, anhand dem man die Ausnutzung der Schwachstelle verifizieren kann.

Unknown (U):

- Es gibt Berichte über Vorfälle, welche die Existenz der Schwachstelle andeuten.
- Die Berichte sind ungenau und widersprüchlich.

Environmental Metrics

Die Environmental Metric Group besteht aus folgenden Metriken:

- Confidentiality Requirement (CR)
- Integrity Requirement (IR)
- Availability Requirement (AR)
- Modifizierte Base Metriken

Zielsetzung:

- Bewertung der durch eine Schwachstelle ausgehende Bedrohung für ein konkretes Umfeld
- Möglichkeit der Beeinflussung einer Bewertung durch den Benutzer

Security Requirements (CR, IR, AR)

Ziel: Anpassung einer CVSS-Bewertung hinsichtlich der Wichtigkeit von Vertraulichkeit, Datenintegrität und Verfügbarkeit innerhalb eines konkreten Umfeldes (z.B. Unternehmen, Institution).

Ansatz: Der Benutzer hat die Möglichkeit, die Bewertung von C, I und A für sein Umfeld anzupassen.

Werte für Confidentiality Requirement (CR), Integrity Requirement (IR) und Availability Requirement (AR):

- Low (L)
- Medium (M)
- High (H)
- Not Defined (X)

Security Requirements – Bewertung

Low (L):

- Der Verlust der Vertraulichkeit, Integrität und Verfügbarkeit wirkt sich im betrachteten Umfeld nur in geringem Maße aus.

Medium (M):

- Der Verlust der Vertraulichkeit, Integrität und Verfügbarkeit hat im betrachteten Umfeld einen nachteiligen Effekt.

High (H):

- Der Verlust der Vertraulichkeit, Integrität und Verfügbarkeit hat im betrachteten Umfeld katastrophale Auswirkungen.

Not Defined (X):

- Es wurde kein Wert angegeben.

Modifizierte Base Metriken

- Die modifizierten Base Metriken ermöglichen die Anpassung der Base Metriken auf die Anforderungen des Unternehmens.
- Der Environmental Score hängt vom entsprechenden Base Score ab.
- Die Modified Base Metriken haben dieselben Werte wie die Base Metriken, plus Not Defined (X).

Modifizierte Base Metriken (Forts.)

Bezeichnungen:

- Modified Attack Vector (MAV)
- Modified Attack Complexity (MAC)
- Modified Privileges Required (MPR)
- Modified User Interaction (MUI) x
- Modified Scope (MS)
- Modified Confidentiality (MC)
- Modified Integrity (MI)
- Modified Availability (MA)

Bemerkungen zu den Environmental Metriken

- Der Einsatz dieser Metriken ist optional.
- Jede Organisation muss selbst festlegen, welche Auswirkungen ein Kollateralschaden hat.
- Die Auswirkungen von Schwachstellen auf das Umfeld eines Unternehmens sollten von Experten untersucht und bewertet werden.
- Fehlerhafte Bewertungen können zu einer fehlerhaften Einschätzung der Bedrohungssituation führen.

Qualitative Bewertungsskala

- Für manche Zwecke ist eine textuelle Bewertung besser als eine numerische.
- Ansatz: Konvertiere einen Score in eine textuelle Bewertung.
- Bewertungstabelle:

Bewertung	CVSS Score
None	0.0
Low	0.1 – 3.9
Medium	4.0 – 6.9
High	7.0 – 8.9
Critical	9.0 – 10.0

Bewertungsvektoren

Aufbau eines Bewertungsvektors:

- Base Vector :

AV: [N,L,A,P] / AC: [L,H] / PR: [N,L,H] / UI: [N,R] /
S: [U,C] / C: [H,L,N] / I: [H,L,N] / A: [H,L,N]

- Temporal Vector (optional):

E: [X,H,F,P,U] / RL: [X,U,W,T,O] / RC: [X,C,R,U]

- Environmental Vector (optional):

CR: [X,L,M,H] / IR: [X,L,M,H] / AR: [X,L,M,H]
MAV: [X,N,L,A,P] / MAC: [X,L,H] / MPR: [X,N,L,H] /
MUI: [X,N,R] / MS: [X,U,C] / MC: [X,H,L,N] /
MI: [X,H,L,N] / MA: [X,H,L,N]

Bewertung der Metriken

- Der Score einer Bedrohung wird anhand des Bewertungsvektors berechnet.
- Hierzu wird jedem Wert einer Metrik ein Zahlenwert zugewiesen.
- Die Werte werden anhand von Tabellen festgelegt.

Attack Vector/Modified Attack Vector

Attack Vector (AV)	
AV	$val(AV)$
Network	0.85
Adjacent	0.62
Local	0.55
Physical	0.2

Bemerkung: Analog für Modified Attack Vector (MAV)

Attack Complexity/Modified Attack Complexity

Attack Complexity (AC)	
AC	<i>val</i> (AC)
Low	0.77
High	0.44

Bemerkung: Analog für Modified Attack Complexity (MAC)

Privileges Required/Modified Privileges Required

Privileges Required (PR)	
PR	$val(PR)$
Falls Scope = U	
None	0.85
Low	0.62
High	0.27
Falls Scope = C	
None	0.85
Low	0.68
High	0.5

Bemerkung: Analog für Modified Privileges Required (MPR)

User Interaction/Modified User Interaction

User Interaction (UI)	
UI	$val(UI)$
None	0.85
Required	0.62

Bemerkung: Analog für Modified User Interaction (MUI)

Confidentiality/Modified Confidentiality

Confidentiality (C)	
C	$val(C)$
High	0.56
Low	0.22
None	0

Bemerkung: Analog für Modified Confidentiality (MC)

Integrity/Modified Integrity

Integrity (I)	
I	$val(I)$
High	0.56
Low	0.22
None	0

Bemerkung: Analog für Modified Integrity (IC)

Availability/Modified Availability

Availability (A)	
A	$val(A)$
High	0.56
Low	0.22
None	0

Bemerkung: Analog für Modified Availability (MA)

Exploit Code Maturity (E)

Exploit Code Maturity (E)	
E	$val(E)$
Not Defined	1
High	1
Functional	0.97
Proof of Concept	0.94
Unproven	0.91

Remediation Level (RL)

Remediation Level (R)	
RL	$val(RL)$
Not Defined	1
Unavailable	1
Workaround	0.97
Temporary Fix	0.96
Official Fix	0.95

Report Confidence (RC)

Report Confidence (RC)	
RC	$val(RC)$
Not Defined	1
Confirmed	1
Reasonable	0.96
Unknown	0.92

Confidentiality Requirement (CR)

Confidentiality Requirement (CR)	
CR	$val(CR)$
Not Defined	1
High	1.5
Medium	1
Low	0.5

Integrity Requirement (IR)

Integrity Requirement (IR)	
IR	$val(IR)$
Not Defined	1
High	1.5
Medium	1
Low	0.5

Availability Requirement (AR)

Availability Requirement (AR)	
AR	$val(AR)$
Not Defined	1
High	1.5
Medium	1
Low	0.5

Berechnung der Kennzahlen

Ansatz:

- Numerische Bewertung der den Metriken zugeordneten Werte
- Berechnung einer Kennzahl pro Gruppe.
- Jede Kennzahl ist ein Wert zwischen 0 und 10.

Gleichungen:

- Base Equation
- Temporal Equation
- Environmental Equation

Die folgenden Formeln basieren auf CVSS v3.1 Revision 1.

Hilfsfunktionen

$rnd_1(x)$ = Wert von x , auf eine Nachkommastelle aufgerundet

$min(x, y)$ = Minimum von x und y

Impact Sub-Score (ISS)

ISS(C, I, A):

1 $c \leftarrow 1 - val(C)$

2 $i \leftarrow 1 - val(I)$

3 $a \leftarrow 1 - val(A)$

4 **return** $1 - c \cdot i \cdot a$

Impact & Exploitability

Impact(ISS, S):

```
1  if S = U then  
2      return 6.42 · ISS  
3  else if S = C then  
4      return 7.52 · (ISS − 0.029) − 3.25 · (ISS − 0.02)15
```

Exploitability(AV, AC, PR, UI):

```
1  return 8.22 · val(AV) · val(AC) · val(PR) · val(UI)
```

Base Score

BaseScore(Impact, Exploitability, S):

```
1 if Impact  $\leq$  0 then  
2   return 0  
3 else  
4    $sum \leftarrow$  Impact + Exploitability  
5   if S = U then  
6     return  $rnd_1(\min(sum, 10))$   
7   else if S = C then  
8     return  $rnd_1(\min(1.08 \cdot sum, 10))$ 
```

Temporal Score Roundup

TemporalScoreRoundup(BaseScore, E, RL, RC):

1 return BaseScore · *val*(E) · *val*(RL) · *val*(RC)

Modified Impact Sub-Score (MISS)

MISS(CR, MC, IR, MI, AR, MA):

- 1 $c \leftarrow 1 - val(CR) \cdot val(MC)$
- 2 $i \leftarrow 1 - val(IR) \cdot val(MI)$
- 3 $a \leftarrow 1 - val(AR) \cdot val(MA)$
- 4 $v \leftarrow 1 - c \cdot i \cdot a$
- 5 return $min(v, 0.915)$

Modified Impact & Modified Exploitability

ModifiedImpact(MISS, MS):

- 1 **if** $MS = U$ **then**
- 2 **return** $6.42 \cdot \text{MISS}$
- 3 **else if** $MS = C$ **then**
- 4 **return** $7.52 \cdot (\text{MISS} - 0.029) - 3.25 \cdot (\text{MISS} \cdot 0.9731 - 0.02)^{13}$

ModifiedExploitability(MAV, MAC, MPR, MUI):

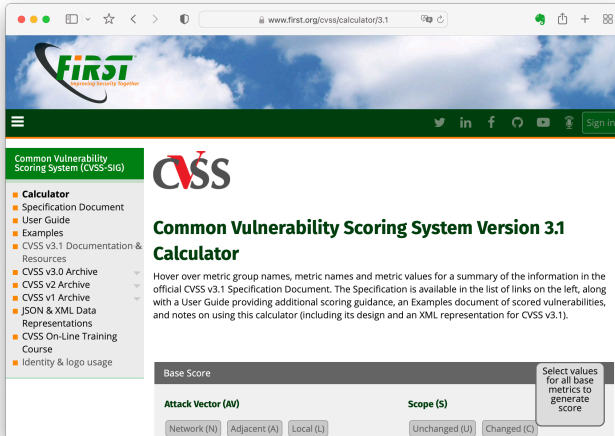
- 1 **return** $8.22 \cdot \text{val}(\text{MAV}) \cdot \text{val}(\text{MAC}) \cdot \text{val}(\text{MPR}) \cdot \text{val}(\text{MUI})$

Environmental Score

EnvironmentalScore(ModifiedImpact, ModifiedExploitability,
MS, E, RL, RC):

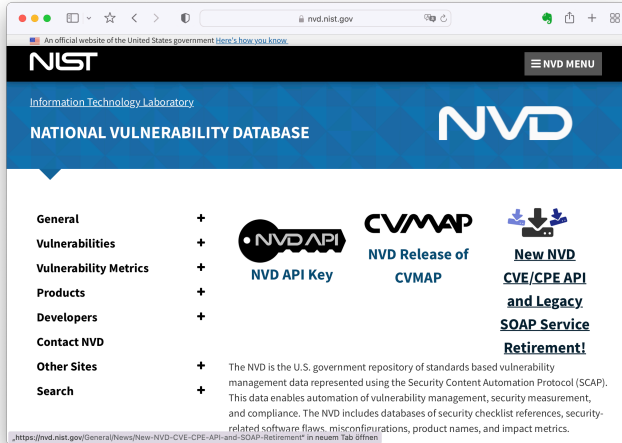
```
1 if ModifiedImpact  $\leq$  0 then
2   return 0
3 else
4    $s \leftarrow$  ModifiedImpact + ModifiedExploitability
5    $p \leftarrow val(E) \cdot val(RL) \cdot val(RC)$ 
6   if MS = U then
7     return  $rnd_1(rnd_1(min(s, 10))) \cdot p$ 
8   else if MS = C then
9     return  $rnd_1(rnd_1(min(1.08 \cdot s, 10))) \cdot p$ 
```

Einsatz eines CVSS-Rechners



Link: <https://www.first.org/cvss/calculator/3.1>

National Vulnerability Database

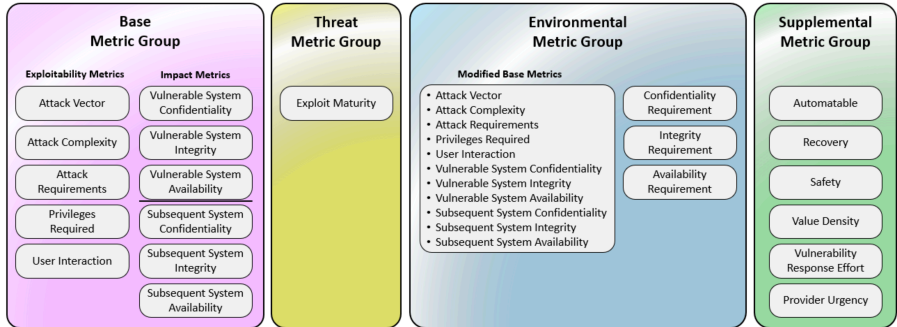


Link: `https://nvd.nist.gov`

Ausblick: CVSS 4.0

- Im November 2023 wurde mit Version 4.0 eine Aktualisierung von CVSS vorgestellt [2].
- Änderungen in den Metrikgruppen:
 - ▷ In der Base Metric Group werden Auswirkungen auf nachfolgende Systeme aufgenommen.
 - ▷ Die Temporal Metric Group wird durch die Threat Metric Group ersetzt.
 - ▷ Die Supplemental Metric Group kommt als neue optionale Gruppe hinzu.
- Die Berechnung des Scores basiert nun auf sogenannten MacroVectors.

CVSS 4.0 Metric Groups



Zusammenfassung

- Das Common Vulnerability Scoring System (CVSS) ist ein anerkanntes System zur Bewertung von Schwachstellen.
- Die in der Praxis am verbreitetste Version ist CVSS 3.1 Revision 1.
- Die aktuelle Version ist CVSS 4.0.
- Das System wird ständig weiter entwickelt.
- CVSS wird in diversen Schwachstellendatenbanken eingesetzt.
- Im Internet findet man Webseiten, die Rechner zur CVSS-Bewertung bereitstellen.

Literatur I

- [1] FIRST, Hrsg. *Common Vulnerability Scoring System Version 3.1. Specification Document. Version v3.1 Revision 1.* Forum of Incident Response und Security Teams (FIRST). 2019. URL: <https://www.first.org/cvss/v3.1/specification-document> (besucht am 12.05.2024).
- [2] FIRST, Hrsg. *Common Vulnerability Scoring System Version 4.0. Specification Document. Version v4.0.* Forum of Incident Response und Security Teams (FIRST). 2023. URL: <https://www.first.org/cvss/v4.0/specification-document> (besucht am 12.05.2024).