Von niedrig bis kritisch: Schwachstellenbewertung mit CVSS

Originalartikel

Backup

<html> <header class=„article-header“><h1 class=„articleheading“>Von niedrig bis kritisch: Schwachstellenbewertung mit CVSS</h1> Andreas Kurtz </header><figure class=„aufmacherbild“><figcaption class=„akwa-caption“>(Bild: The Viz / Shutterstock.com (nachbearbeitet))</figcaption></figure> Das Common Vulnerability Scoring System hilft bei der Bewertung von Schwachstellen. Wir erklären Funktionsweise und Grenzen des Systems. Beim Common Vulnerability Scoring System (CVSS) geht es hauptsächlich darum, die Gefahr zu bewerten, die von einer Sicherheitslücke ausgeht. In einigen Fällen geht das recht leicht. Bei der Ende 2019 als „Shitrix“ bekannt gewordenen Schwachstelle in Citrix-Produkten konnten unauthentifizierte Angreifer betroffene Systeme aus der Ferne angreifen und mit nur wenig Aufwand unter ihre Kontrolle bringen (CVE 2019-19781). Im Laufe des Jahres 2020 wurden über Shitrix zahlreiche Systeme infiziert und mit Backdoors versehen. Diese wurden anschließend vielfach zum Einschleusen von Ransomware verwendet. Im Falle der Universitätsklinik Düsseldorf etwa forderte die Ransomware „DoppelPaymer“ ein Lösegeld [1]. Es liegt auf der Hand, dass Shitrix eine „kritische“ Lücke ist. Die Sachlage ist jedoch nicht immer so eindeutig. Der 2014 bekannt gewordenen POODLE-Angriff auf HTTPS-Verschlüsselung [2], nutzte sehr trickreich Schwächen im SSL-Protokoll und sorgte für viel Aufsehen (CVE-2014-3566). Doch so genial der Angriff in der Theorie war, so schwierig war dann die Ausführung unter realen Bedingungen: Der Aufwand war hoch und der Schaden beschränkte sich auf das Byte-weise Auslesen von Daten aus den Browsern einzelner Nutzer. Gefühlt ist POODLE also deutlich weniger gefährlich Shitrix. Doch wie lässt sich dieses Bauchgefühl quantifizieren? <h3 class=„subheading“ id=„nav_risiko0“>Risiko = Eintrittswahrscheinlichkeit x Schaden</h3> Bei der systematischen Bewertung von Schwachstellen macht man sich das allgemeine Prinzip der Risikoanalyse zunutze. Dabei identifiziert man Schadensereignisse und schätzt ab, wie wahrscheinlich diese Ereignisse eintreten und wie hoch die daraus resultierenden Schäden sein könnten. In der Praxis lassen solche Einschätzungen jedoch viel Interpretationsspielraum zu. Systeme zur Schwachstellenbewertung helfen mit vordefinierten Faktoren, Wahrscheinlichkeit und Schadensausmaß möglichst objektiv zu beziffern. Ein solches System ist das Common Vulnerability Scoring System (CVSS), das sich international zunehmend als De-facto-Standard etabliert, um wesentliche Merkmale einer Schwachstelle zu beschreiben und deren Schweregrad zu bestimmen. Die Anfänge von CVSS reichen bis ins Jahr 2005 zurück, als das US

National Infrastructure Advisory Council (NIAC) eine erste Entwurfsfassung veröffentlichte. Die Verantwortung für CVSS ging seitdem an das Forum of Incident Response and Security Teams (FIRST) über, ein Zusammenschluss internationaler Sicherheits- und Incident-Response-Teams aus Regierungen, Industrie und Wissenschaft. Bei FIRST kümmert sich seitdem die CVSS Special Interest Group (SIG) um die Weiterentwicklung von CVSS. Die heute gültige Version 3.1 stammt aus dem Jahr 2019. <h3 class=„subheading“ id=„nav_metriken_zur1“>Metriken zur differenzierten Bewertung</h3> Die Bewertung von Schwachstellen erfolgt bei CVSS anhand verschiedener Kriterien, sogenannter Metriken. Für jede Metrik gibt es vordefinierte Wahlmöglichkeiten. Aus denen errechnet sich ein Schweregrad von 0.0 bis 10.0, wobei 10.0 dem höchsten Schweregrad entspricht. Diesen Zahlenwerten werden anschließend die auch aus Schwachstellen-Meldungen bekannten qualitativen Kategorien („None“, „Low“, „Medium“, „High“ und „Critical“) zugeordnet. Die Metriken zur Bestimmung des Schweregrads sind dabei in drei Gruppen unterteilt: Base Metrics, Temporal Metrics und Environmental Metrics. Basis-Metriken (Base Metrics) beschreiben die wesentlichen technischen und unveränderlichen Merkmale einer Schwachstelle. Aus ihnen lässt sich ein sogenannter „Base Score“ errechnen, der für den technischen Schweregrad einer Schwachstelle steht. Er kann später nachjustiert und an zeitliche Veränderungen (Temporal Metrics) oder die jeweilige Umgebung des betroffenen Systems (Environmental Metrics) angepasst werden. <figure class=„a-inline-image a-u-inline“> <figcaption class=„a-caption“>Die CVSS-Metriken gliedern sich, wie hier abgebildet, in drei Gruppen.</figcaption> (Bild: first.org [3]) </figure> Den Base Score einer Schwachstelle errechnet in der Regel deren Entdecker oder aber der Hersteller des betroffenen Produkts beziehungsweise ein CERT, das die Behebung der Schwachstelle koordiniert. Für Schwachstellen in öffentlichen Standard-Produkten wird meist auch eine CVE-ID als eindeutige Schwachstellenbezeichung im Format CVE-YYYY-NNNNN beantragt beziehungsweise vergeben. Viele Leser von Schwachstellen-Meldungen dürften das CVE-System, dem wir einen separaten Hintergrundartikel gewidmet haben [4], und CVSS bisher als untrennbare Einheit wahrgenommen haben. In Wirklichkeit sind CVE und CVSS jedoch zwei völlig voneinander unabhängige Systeme: Eine CVE-ID ist keine zwingende Voraussetzung zur CVSS-Verwendung. Während CVE das Ziel verfolgt, Schwachstellen eindeutige Bezeichner zuzuweisen und zu verwalten, möchte CVSS den Schwachstellen einen Schweregrad zuteilen. <h3 class=„subheading“ id=„nav_den_base_score2“>Den Base Score bestimmen</h3> Nun aber zurück zum Base Score: Die zur Berechnung verwendeten Basis-Metriken bewerten zum einen die Voraussetzungen für einen Angriff (Exploitability Metrics) und zum anderen auch die aus einer Ausnutzung resultierenden Konsequenzen (Impact Metrics). Bei den Voraussetzungen wird beispielsweise hinterfragt, ob ein Angriff über das Internet durchgeführt werden kann oder ob ein Angreifer physischen Zugriff auf ein Gerät benötigt (Attack Vector). Weiter wird abgeschätzt, wie komplex die Durchführung eines Angriffs ist (Attack Complexity) und ob ein Angriff unauthentifiziert durchgeführt werden kann oder ob ein Angreifer über ein gültiges Benutzerkonto mit bestimmten Privilegien verfügen muss (Privileges Required). Zudem fließt mit ein, ob für eine erfolgreiche Ausnutzung die Interaktion mit einem Benutzer erforderlich ist (User Interaction). Zur Bewertung der Konsequenzen eines erfolgreichen Angriffs ist entscheidend, inwieweit Daten von dem betroffenen System ausgelesen oder verändert werden können oder das System in seiner Verfügbarkeit eingeschränkt werden kann. Ermittelt wird also, wie stark durch eine erfolgreiche Ausnutzung

die Schutzziele Vertraulichkeit (Confidentiality), Integrität (Integrity) und Verfügbarkeit (Availability) beeinträchtigt werden. Eine gewisse Sonderstellung nimmt die Scope-Metrik ein, die mit CVSS v3.0 eingeführt wurde. Über sie lässt sich erfassen, wenn zwar eine bestimmte Komponente verwundbar ist (Vulnerable Component), sich die Ausnutzung einer Schwachstelle aber unmittelbar auf eine andere, physisch oder logisch abgetrennte Komponente auswirkt (Impacted Component). Ein sogenannter „Scope Change“ tritt beispielsweise auf, wenn eine Schwachstelle in einer virtuellen Maschine (Vulnerable Component) einem Angreifer ermöglicht, Dateien auf dem Host-Betriebssystem (Impacted Component) zu lesen oder zu verändern. Die Überwindung der logischen Sicherheitsbarriere verursacht einen Scope-Wechsel und hat für die Bewertung der Schwachstelle einen höheren Schweregrad zur Folge. Die ermittelten Basis-Metriken werden schließlich miteinander verrechnet und ergeben den Base Score. Im Internet veröffentlichte CVSS-Bewertungen nutzen zumeist diesen Score. So nennt etwa auch die National Vulnerability Database (NVD) [5] des National Institute of Standards and Technology (NIST) zu jeder bekannt gewordenen Schwachstelle den CVSS Base Score. <h3 class=„subheading“ id=„nav_nachträgliche3“>Nachträgliche Feinjustierung</h3> Der Base Score kann später nachjustiert und an zeitliche Veränderungen (Temporal Metrics) oder die jeweilige Umgebung des betroffenen Systems (Environmental Metrics) angepasst werden. Zeitliche Veränderungen liegen zum Beispiel dann vor, wenn nicht mehr nur vage Textbeschreibungen zu einer abstrakten Schwachstelle vorliegen, sondern ein voll funktionsfähiger Exploit in freier Wildbahn auftaucht (Exploit Code Maturity). Ähnliches gilt, wenn eine Schwachstelle, über die zunächst nur spekuliert wurde, beispielsweise durch den Hersteller bestätigt wurde (Report Confidence). Die Gefahr kann auch abnehmen, etwa wenn für die Schwachstelle ein Workaround oder ein offizieller Hersteller-Fix zur Verfügung steht (Remediation Level). Diese Änderung der Gefahrenlage bildet CVSS etwas gewöhnungsbedürftig ab: Temporal Metrics können nämlich den Base Score immer nur nachträglich absenken, ihn aber nicht anheben. CVSS geht zunächst also immer vom Worst-Case-Szenario aus – ein Sachverhalt, der im Rahmen der Kritik an CVSS weiter unten noch genauer diskutiert wird. Die Environmental Metrics wiederum ermöglichen, den Score unternehmensintern an die jeweils vorherrschende IT-Umgebung anzupassen. Je nachdem, wie wichtig oder unwichtig das von einer Schwachstelle betroffene System für ein Unternehmen ist, wird der Base Score auf- oder abgewertet. So ist für ein Unternehmen eine bestimmte Schwachstelle in der Speiseplan-App der Kantine sicherlich weniger schlimm, als wenn sie das unternehmenseigene Data Warehouse betrifft. Mit den Environmental Metrics werden Vertraulichkeits-, Verfügbarkeits- und Integritätsanforderungen an das konkrete System festgelegt. Auch können möglicherweise bereits vorhandene Gegenmaßnahmen innerhalb der Umgebung für die Bewertung berücksichtigt werden. <h3 class=„subheading“ id=„nav_gesamtschweregra4“>Gesamtschweregrad und CVSS-Vektor</h3><figure class=„a-u-inline-right a-inline-image a-u-inline“> <figcaption class=„a-caption“>CVSS Scores werden qualitative Schweregrade zugeordnet.</figcaption> (Bild: first.org) </figure> Am Ende werden Base Score, Temporal Score und Environmental Score zu einem Gesamt-Score verrechnet. Für jeden einzelnen Score (Base, Temporal, Environmental) sowie für den Gesamt-Score ergibt sich so ein Zahlenwert. Diese Zahlenwerte werden in qualitative Schweregrad-Kategorien von „None“ bis „Critical“ unterteilt. Diese qualitative Zuordnung ist optional und soll vorrangig Unternehmen bei ihrem internen Schwachstellen-Management unterstützen. Zusätzlich werden die Metriken in einer textuellen Kurzform zusammengefasst, dem sogenannten CVSS-Vektor. Dieser Vektor enthält alle Informationen über die vorangegangenen Einstufungen und wird stets mit dem Score veröffentlicht. <figure class=„a-u-inline-left a-inline-image a-u-inline“> <figcaption class=„a-caption“>Der CVSS-Calculator zeigt hier die Bewertung der

Shitrix-Schwachstelle.</figcaption> (Bild: first.org (Screenshot)) </figure> Einen kompakten Überblick über den Aufbau des CVSS-Vektors liefert die CVSS Calculator [6]-Anwendung [7]. In der Praxis erleichtert sie übrigens auch die Bewertung: Über eine Weboberfläche lassen sich dort Metriken einfach „zusammenklicken“ und Scores sowie Vektoren direkt ablesen. Auch bereits vorhandene Basis-Vektoren können eingelesen werden, um dann etwa Temporal oder Environmental Metrics anzupassen. Zurück zu den Schwachstellen-Beispielen vom Anfang: Die Shitrix-Schwachstelle kommt auf einen Base Score von 9.8, was einem kritischen Schweregrad entspricht. Die dem Poodle-Angriff zugrundeliegende Schwachstelle erhält einen Base Score von 3.1, also ein niedriger Schweregrad. Das drückt das eingangs formulierte Gefühl recht gut in Zahlen aus. <figure class=„a-inline-image a-u-inline“> <figcaption class=„a-caption“>Zwei ganz unterschiedliche Beispiele für CVSS-Vektoren liefern Shitrix und POODLE.</figcaption> (Bild: Andreas Kurtz) </figure><h3 class=„subheading“ id=„nav_kritikpunkte_an5“>Kritikpunkte an CVSS</h3> Wie jedes System zur Schwachstellenbewertung hat auch CVSS seine Tücken. Eine der Hauptkritikpunkte ist die intransparente Herleitung der Formeln zur Berechnung der Scores. Beispielsweise legt die Spezifikation fest, dass eine Low Attack Complexity (AC:L) immer genau mit dem Faktor 0.77 gewichtet werden soll. Wie diese Faktoren im Detail zustande kamen, bleibt genauso unklar wie wissenschaftliche Belege dafür fehlen, dass die Formeln empirisch oder theoretisch fundiert sind. Wie bereits erwähnt gibt es auch an den Temporal Metrics Kritik. So ist laut Spezifikation sinnvollerweise vorgesehen, dass eine Schwachstelle kritischer zu betrachten ist, sobald ein öffentlicher Exploit auftaucht. Über Temporal Metrics lässt sich der Base Score aber lediglich absenken und nicht erhöhen. Genauso diskussionswürdig ist, ob beziehungsweise wie das Vorhandensein eines Hersteller-Updates (Remediation Level: Official Fix) den Schweregrad einer Schwachstelle absenkt. Denn nur durch das Vorhandensein eines Patches ist er auf den betroffenen Systemen noch lange nicht eingespielt. Und wenn man bedenkt, dass Angreifer häufig Patches analysieren, um so die Schwachstellen besser zu verstehen (Patch Diffing), könnte man auch argumentieren, der Schweregrad müsste durch das Vorhandensein eines Patches eher zunehmen. Neben der Kritik an CVSS selbst gibt es auch Einwände zur Verwendung in der Praxis. Häufig wird der CVSS Score nämlich eins zu eins als Risiko interpretiert. Hier ist aber wichtig zu verstehen, dass der Base Score lediglich einen technischen Schweregrad abbildet und nicht etwa ein konkretes Risiko aufzeigen kann. Ein aus dem Kontext gegriffener technischer Schweregrad sagt nur wenig über das tatsächlich mit einer Schwachstelle einhergehende Risiko für das betroffene System, für die darauf abgebildeten Geschäftsprozesse und letztendlich für das Unternehmen aus. Auch bereits vorhandene Gegenmaßnahmen, die eine Ausnutzung möglicherweise erschweren oder verhindern, werden im Base Score nicht berücksichtigt. <h3 class=„subheading“ id=„nav_unterschiedliche6“>Unterschiedliche Interessen – unterschiedliche Bewertung?</h3> Zwar sollen Schwachstellen möglichst unvoreingenommen bewertet werden, dennoch dürften auch die Interessen der bewertenden Person eine große Rolle spielen: Ein Security Researcher verspricht sich im Zweifel mehr Hype um eine entdeckte Schwachstelle, wenn sie möglichst hoch bewertet wird. Hersteller hingegen argumentieren, über eine bessere Informationsgrundlage zu verfügen, um den tatsächlichen Schweregrad verlässlicher bewerten zu können. Andererseits kann es durchaus vorkommen, dass Hersteller versuchen, Schwachstellen möglichst kleinzureden, um schlechte Presse zu vermeiden. Diskussionen innerhalb der Security Community und auch unabhängige Bewertungen des NIST sorgen hier in der Regel allerdings für eine selbstkorrigierende Wirkung

und Hersteller haben daher – nicht zuletzt auch aus Haftungsgründen – ein Interesse an seriösen Bewertungen. Inwieweit CVSS dahingehend verlässlich ist, dass unterschiedliche Personen zu einheitlichen Bewertungen kommen, untersucht derzeit eine Forschungsgruppe der Universität Erlangen-Nürnberg in einer Online-Umfrage [8]. Die Umfrage richtet sich an Experten, die regelmäßig CVSS verwenden. In der Umfrage werden Hintergrundkenntnisse zu CVSS abgefragt, bevor dann vier vorgegebene Schwachstellen mittels CVSS bewertet werden sollen. Die Umfrage läuft noch bis Mitte Februar, das Ausfüllen dauert etwa 20 bis 30 Minuten. Über die Ergebnisse werden wir hier auf heise Security berichten. <h3 class=„subheading“ id=„navinline7“>Lesen Sie auch<figure class=„a-article-teaserimage-container“> </figure> </h3><header><h1 class=„a-article-teasertitle a-u-mb-1“>Schubladen für Schwachstellen: Das CVE-System im Überblick</h1></header>[9]<h3 class=„subheading“ id=„nav_fazit_und8“>Fazit und Ausblick</h3> Trotz der genannten Kritikpunkte ist CVSS nach bald zwei Jahrzehnten aus der Welt des Schwachstellen-Managements – nicht nur mangels Alternativen – heute nicht mehr wegzudenken. Es ersetzt zwar keine individuelle Risikoanalyse, bei der in der Regel weit mehr Faktoren als nur der Schweregrad einer Schwachstelle betrachtet werden. Insbesondere erweitert um den Environmental Score kann der CVSS Base Score aber als wichtiges Merkmal mit in eine solche Analyse einfließen. Wie ein Blick in die Liste der möglichen Verbesserungen für CVSS v4.0 [10] zeigt, arbeiten die Autoren der CVSS SIG bereits mit Hochdruck an der nächsten Version. Darin sollen auch einige der genannten Kritikpunkte in Angriff genommen werden. Die Temporal Metrics sollen nachgebessert werden und es wird auch diskutiert, Nutzern ein Best-Practice-Dokument an die Hand zu geben, wie der CVSS-Score als Input in einer Gesamtrisikobewertung verwendet werden kann. Wer tiefer in das Thema CVSS einsteigen möchte, dem sei die CVSS-v3-Spezifikation [11] ans Herz gelegt. Hier werden sämtliche Metriken, die vordefinierten Wahlmöglichkeiten und die Berechnungsformeln detailliert erläutert. Ein User Guide gibt zusätzliche Hinweise zur Verwendung von CVSS [12]. Ein erstes Gefühl für das Zustandekommen von CVSS-Werten bekommt man schon mit wenigen Klicks im CVSS Calculator [13]. Diskutieren Sie mit uns im Expertenforum von heise Security Pro, wie sich CVSS in Ihrer Praxis bewährt hat: <figure class=„a-inline-image a-u-inline“> <a class=„akwa-inline-imglink“

href=„https://www.talque.com/go/org/l09hv0jMm0eOPa9AeiTj/post/bpHIHgyatZ7lSXBxNrfv/view“ rel=„external“ target=„_blank“ title=„heise Security Pro Expertenforum“>[14] </figure> () <hr /> URL dieses Artikels:
<small>

```
https://www.heise.de/-5031983
```

</small> Links in diesem Artikel:
<small>

```
<strong>[1]</strong> https://www.heise.de/news/Uniklinik-Duesseldorf-Ransomware-DoppelPaymer-soll-hinter-dem-Angriff-stecken-4908608.html
```

</small>
<small>

```
<strong>[2]</strong> https://www.heise.de/security/artikel/Poodle-So-funktioniert-der-Angriff-auf-die-Verschluesselung-2425250.html
```

</small>
<small>

```
<strong>[3]</strong> https://www.first.org/cvss/v3-1/media/MetricGroups.svg
```

</small>
<small>

```
<strong>[4]</strong> https://www.heise.de/hintergrund/Schubladen-fuer-Schwachstellen-Das-CVE-System-im-Ueberblick-4940478.html
```

</small>
<small>

```
<strong>[5]</strong> https://nvd.nist.gov/
```

</small>
<small>

```
<strong>[6]</strong> https://www.first.org/cvss/calculator/3.1
```

</small>
<small>

```
<strong>[7]</strong> https://www.first.org/cvss/calculator/3.1
```

</small>
<small>

```
<strong>[8]</strong> https://user-surveys.cs.fau.de/index.php?r=survey%2Findex&amp;sid=248857
```

</small>
<small>

```
<strong>[9]</strong> https://www.heise.de/hintergrund/Schubladen-fuer-Schwachstellen-Das-CVE-System-im-Ueberblick-4940478.html
```

</small>
<small>

[10] https://docs.google.com/document/d/1qmmk9TQulW9d1cuipu_ziXDX0pUswbZ1WSQyynHbvKU/edit

</small>
<small>

[11] https://www.first.org/cvss/specification-document

</small>
<small>

[12] https://www.first.org/cvss/user-guide

</small>
<small>

[13] https://www.first.org/cvss/calculator/3.1

</small>
<small>

[14] https://www.talque.com/go/org/l09hv0jMm0eOPa9AeiTj/post/bpHIHgyatZ7lSXBxNrfv/view

</small>
<small>

[15] mailto:ovw@heise.de

</small>
 Copyright © 2021 Heise Medien </html>