Ludwig-Maximilians-Universität München
Institut für Informatik
Lehr- und Forschungseinheit für Kommunikationssysteme und Systemprogrammierung
Prof. Dr. Helmut Reiser

MNM TEAM
Munich Network Management Team

IT-Sicherheit im Wintersemester 2021/2022
Übungsblatt 5

Besprechung: Di, 30.11.2021

Achtung: Zur Bearbeitung einiger Übungsaufgaben ist es notwendig sich über den Vorlesungsinhalt hinaus, durch Internet- und Literaturrecherche mit dem Thema zu beschäftigen.

Aufgabe 17: (T) Rootkits

Nachdem ein Angreifer erfolgreich Zugang zu einem IT-System, etwa durch das Ausnutzen einer dort vorhandenen Schwachstelle, erlangen konnte, wird dort meist ein Rootkit installiert.

a. Man unterscheidet grundsätzlich zwei Varianten von Rootkits: User-Mode- und Kernel-Mode-Rootkits. Erläutern Sie diese kurz.

b. Wie unterscheidet sich ein Rootkit von anderer Malware, z.B. Viren, Würmer und Trojanischen Pferden?

c. Rootkits verfügen im Allgemeinen über eine sogenannte *Dropper*-Komponente. Welchem Zweck dient diese Komponente. Was versteht man unter einem *Multistage Dropper*?

d. Charakteristisch für Rootkits sind sogenannte Anti-Forensik-Maßnahmen. Erläutern Sie folgende Maßnahmen

   - Data Destruction
   - Data Concealment
   - Data Fabrication

Aufgabe 18: (T) Security Information and Event Management

SIEM stellt eine Kombination aus dem *Security Information Management* (SIM) und dem *Security Event Management* (SEM) dar. Grundidee ist es hierbei viele/alle sicherheitsrelevanten Informationen an einer zentralen Stelle zu sammeln und in Echtzeit auszuwerten.

a. Beschreiben Sie Aufgaben und Funktionsprinzip eines SIEM.

b. Welche Herausforderungen stellen sich dabei?

c. Welche Informationsquellen lassen sich an das SIEM anbinden?

d. Lassen sich durch den Einsatz eines SIEM mehr Bedrohungen erkennen als ohne?

e. Welche Vor- und Nachteile ergeben sich beim Einsatz eines SIEMs?

f. Wie unterscheidet sich ein SIEM von einem *Intrusion Detection System* (IDS) oder *Intrusion Prevention System* (IPS)?

Aufgabe 19: (T) Tactics, Techniques and Procedures

TTP ist ein etabliertes Konzept zur Beschreibung von Cyberangriffen bzw. -angriffsmustern.

a. Beschreiben Sie die drei Bestandteile des TTP-Konzepts und instanziieren Sie es durch ein geeignetes Beispiel. Für welche Einsatzzwecke sind TTPs geeignet?

b. Der folgende Artikel berichtet von einem Cyberangriff.
`https://www.golem.de/news/trickbot-us-militaer-greift-botnetzwerk-an-2010-151452.html`
Beschreiben Sie diesen Angriff im TTP-Schema.

c. Was ist die *MITRE ATT&CK*-Matrix (`https://attack.mitre.org/`)? Wie ist sie aufgebaut? Wozu kann sie eingesetzt werden?

d. *Access Token Manipulation* ist eine Technik zur *Privilege Escalation*.

   (i) Wie funktionieren derartige Angriffe?

   (ii) Welche Formen sind bekannt?

   (iii) Wie können sie detektiert und verhindert werden?

e. Angenommen ein Angreifer möchte auf möglichst vielen Maschinen in einem fremden Netzwerk einen Krypto-Miner installieren. Welche Zwischenschritte muss der Angreifer verfolgen, um zu diesem Ziel zu gelangen? Welche Techniken könnte er dazu jeweils einsetzen? Nutzen Sie die *MITRE ATT&CK*-Matrix.

Aufgabe 20: (T) Common Vulnerability Scoring System 3 (CVSSv3)

Für diese Aufgabe soll die folgende Schwachstellenbeschreibung verwendet werden, die über die vier Teilaufgaben hinweg modifiziert wird. Änderungen in einer der Teilaufgaben gelten auch in den darauf folgenden Teilaufgaben. (d.h. Änderungen in Teilaufgabe b) gelten auch für Teilaufgaben c) und d)).

In einer weit verbreiteten Webanwendung, die in ihrem Unternehmen als Kundenportal zur Verwaltung von Softwarelizenzen verwendet wird und daher öffentlich im Internet zugänglich sein muss, existiert eine cross-site request forgery (CSRF) Schwachstelle. Durch diese Schwachstelle können Angreifer aus der Ferne Aktionen mit den Rechten des angegriffenen Benutzers ausführen, wenn der Benutzer eine aktive Session hat und dazu gebracht werden kann, einen schädlichen Link zu öffnen.

Hinweis: Geben Sie bei den Aufgaben, bei denen explizit CVSS-Berechnungen gefordert sind, nicht nur deren Ergebnisse an, sondern begründen Sie auch die von Ihnen gewählten Optionen.

a. Beschreiben Sie kurz wie ein Angriff per CSRF üblicherweise funktioniert.

b. Berechnen Sie mithilfe des unter `https://www.first.org/cvss/calculator/3.1` verfügbaren CVSSv3-Calculators für die beschriebene Schwachstelle den CVSSv3 Base-Score. Vergleichen Sie diesen mit dem über `https://nvd.nist.gov/cvss.cfm?calculator&version=2` berechneten CVSSv2 Base-Score.

c. Die beschriebene Schwachstelle wurde am selben Tag auch auf der Security-Mailingliste *Full-Disclosure* publiziert und deren Ausnutzbarkeit anhand eines Proof-of-Concept (POC) bewiesen. Der Hersteller der Webanwendung hat die Schwachstelle nun auch offiziell bestätigt, aber bislang nur einen Workaround veröffentlicht. Wie verändert sich dadurch der CVSSv3 Base- bzw. Temporal-Score?

d. Bereits am nächsten Tag tauchte in einschlägigen Foren ein Exploit für diese Schwachstelle auf. Dieser besitzt keine besonderen Voraussetzungen und ist somit in jeder Situation funktional. Wie verändert sich dadurch der CVSSv3 Base-/Temporal-Score aus Aufgabe c)?

Aufgabe 21: (H) XSS-Game

Erfolgreiches Cross-Site-Scripting ist erstaunlich einfach. Probieren Sie es selbst:
`https://xss-game.appspot.com/`

Aufgabe 22: (H) Web Vulnerability Scanning mit OWASP ZAP

Es gibt eine Vielzahl möglicher (bekannter) Schwachstellen, die Webapplikationen potentiell aufweisen könnten – definitiv zu viele, um WebApps händisch gegen all jene zu prüfen.
Zur Programmierzeit oder zu Pentesting-Zwecken kommen dazu (Web-)Vulnerability-Scanner zum Einsatz, die WebApps gegen Listen bekannter Schwachstellen oder -pattern abgleichen (z.B. gegen die der *Common Weakness Enumeration*, CWE: `https://cwe.mitre.org/`).

a. Wozu werden *Spiders* eingesetzt? Welche Rolle spielen sie? Welche Arten von Schwachstellen lassen sich mit ihnen entdecken?

b. Was versteht man in diesem Kontext unter *Fuzzing*? Welche Rolle spielt es bei der Eingabenvalidierung?

Mit dem Slogan „The world's most widely used web app scanner“ wirbt der **OWASP ZAP** (Zed Attack Proxy) für sich.

`https://www.zaproxy.org/`

Testen Sie ihn!
Natürlich nur um eigenentwickelte WebApps auf Schwachstellen zu überprüfen. Böswillige Missetaten anzurichten liegt uns fern! ;-)
Tipp: OWASP ZAP ist standardmäßig Teil von Kali.