7. Übung zur VL Betriebs- und Kommunikationssysteme

Tutor: Thomas Tegethoff
Bearbeiter: Etienne Jentzsch, Carola Bothe

1a)

Directory Traversal

Trickst ein schwaches Programm aus, sodass es Zugang zu einer geschützten/inneren Datei gewährt indem es den Verzeichnisbaum hoch und dann runter zur Passwort Datei geht. Als Verteidigungsmaßnahme sichert/verschlüsselt man die Passwortdateien und erlaubt web Servern keinen vollen Zugriff auf Dateisysteme. (VL Folie 7.29, Mitschrift)

Buffer Overflow

Bezeichnet man, wenn ein Programm Daten über die Grenzen eines Puffers auf benachbarten Speicherplatz schreibt. Dadurch kann bösartiger Code auf diesen benachbarten Speicherplatz geschrieben und später ausgeführt werden. (https://en.wikipedia.org/wiki/Buffer_overflow)

Trapdoor/Backdoor

Ursprünglich als Tool zum debuggen und testen gedacht, erlaubt es mittels Hintertüren den Zugang zu einer Datei/einem Programm ohne den üblichen Sicherheitsprozess zu durchlaufen. Die Tür wird durch ein Stück Software implementiert (dazu werden im Negativfall z.B. Geräte beim Transport abgefangen) und kann vom Betriebssystem nur schwer kontrolliert werden. (VL Folie 7.35, Mitschrift)

Logic Bomb

Gehört zu den ältesten Schädlingen und ist ebenfalls Teil eines Programms. Sie 'explodiert' wie der Name sagt, wenn eine bestimmte (logisch) Bedingung erfüllt ist. Die Folgen der Explosion können veränderte oder gelöschte Daten, ein Abstürzen des Systems oder anderes sein. (VL Folie 7.36, https://de.wikipedia.org/wiki/Logikbombe)

Trojan Horse

Angeblich nützliches Programm, das die gewünschte Funktion ausführt aber zusätzlich eine bösartige Funktion hat. Beispiel wäre eine Taschenlampen Anwendung, die nebenbei Kontaktdaten übermittelt, was man hier zumindest mit der Ablehnung von Zugriffsrechten im Zaum halten kann. (VL Folie 7.37, Mitschrift)

Virus

Software, die nach einem Auslöser andere Software modifiziert und neben dem schädlichen Code auch Code zur Selbstreplikation enthält. Viren haben heutzutage lange Inkubationszeiten, damit sie auch auf etwaigen Backups vertreten sind, und die Fähigkeit zur Mutation, sodass Virenprogramme an ihre Grenzen stoßen. (VL Folie 7.40, Mitschrift)

Worm

Sind wie Zombies unabhängig vom Gastbetriebssystem und sobald sie einmal aktiviert wurden, replizieren sie sich wie Viren, wobei sie aktiv nach weiteren Verbreitungsmöglichkeiten auf andere System und Netzwerke suchen. In Gegensatz zum Virus findet die Verbreitung ohne die Infektion von fremden Dateien oder Bootsektoren statt.
(VL-Folie7.45,
https://web.archive.org/web/20101124010540/http://service1.symantec.com/SUPPORT/nav.nsf/pfdocs/1999041209131106)

Bot (aka Zombie, Drone)

Ein Programm, das heimlich andere am Internet hängende Geräte übernimmt, von denen dann die Attacke gestartet wird. Besonders im Zeitalter des Internets stellen sie daher eine Bedrohung dar und man kann die Bots selten zu ihrem Ersteller zurückverfolgen. (VL Folie 7.48, Mitschrift)

Rootkit

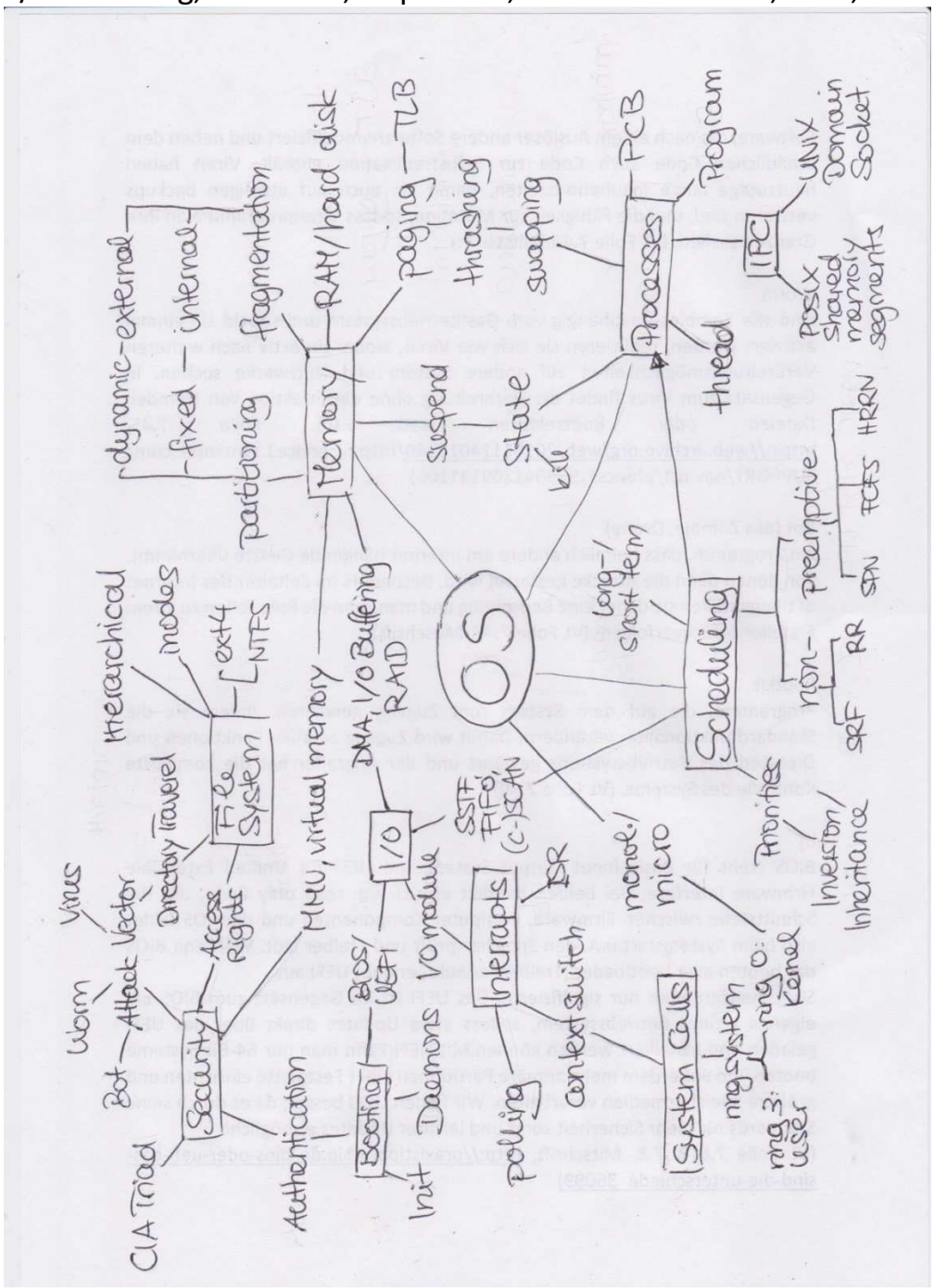
Sind Programme, die auf dem System Root Zugang gewähren, indem sie die Standardfunktionalität verändern. Damit wird Zugang zu allen Funktionen und Diensten des Betriebssystems gewährt und der Angreifer hat die komplette Kontrolle des Systems. (VL Folie 7.49)

b)

BIOS steht für Basic Input Output System und UEFI für Unified Extensible Firmware Interface. Bei beiden handelt es sich um read only Code (ist also nicht beschreibbar), der die Schnittstelle zwischen Firmware, Computer Komponenten und dem OS bildet also beim Systemstart u. A. den Speicher prüft und Treiber lädt. Während BIOS das booten aller bootloader (Treiber) erlaubt, erlaubt UEFI aus Sicherheitsgründen nur signifizierte. Das UEFI ist im Gegensatz zum BIOS ein eigenes kleines Betriebssystem, sodass etwa Updates direkt über das UEFI geladen und installiert werden können. Mit UEFI kann man nur 64-Bit Systeme booten und außerdem mehr primäre Partitionen einer Festplatte einrichten und größere Speichermedien verarbeiten. Wir finden UEFI

besser, da es durch seine Standards für mehr Sicherheit sorgt und leichter Updates ermöglicht und dazu eine benutzerfreundliche Oberfläche bietet.
(VL Folie 7.6 & 7.8, Mitschrift, http://praxistipps.chip.de/bios-oder-uefi-das-sind-die-unterschiede_36099)

c)
10 gut verstandene Begriffe:
Rings, scheduling algorithms, Process, swapping, fragmentation, partitioning, virtual memory, hierarchical file system, priorety inheritance, preemptive
Weniger gut verstandene Begriffe:
I/O Buffering, Daemons, dispatcher, countermeasures, Peer, NAP



2a) +Vorteil / - Nachteil

(Named) Pipe oder auch FIFO-Pipes:

Unter einer Pipe versteht man eine gepufferte Datenverbindung zwischen zwei Prozessen nach FIFO Prinzip.

+ Im Gegensatz zu unbenannten Pipes können benannte Pipes auch zur Kommunikation zwischen Prozessen benutzt werden, die nicht miteinander verwandt sind. Einer Datei wird also ein gewisser Name zugeteilt und jeder Prozess, der diesen Namen kennt, kann darüber die Verbindung zur Pipe und damit zu anderen Prozessen herstellen.

- Für benannte Pipes müssen Zugriffsrechte verteilt werden.

http://faq.prosoft.de/post/was-ist-ein-named-pipe/

Memory Mapped File:

Ein Memory-Mapped-File (Speicher-abgebildete Datei) enthält den Inhalt/ein Teil des Inhalts einer Datei im virtuellen Speicher. Den arauf zeigenden Pointer kann man dann einfach zum lesen und schreiben von daten auf der Datei nutzen.

+ Datei kann durch Lesen und Schreiben direkt im Speicher geändert werden
+ Dateien eignen sich für die Arbeit mit extrem großen Quelldateien
+ Zugriff auf Speicher-abgebildete Dateien ist schneller als die Verwendung von direkten Lese- und Schreiboperationen
– concurrency Problemen, kann man jedoch durch Regulieren der Zugriffsrechte mit file lock oder semaphores umgehen
– mögliche Verschwendung von Leerzeichen bei kleinen Dateien, da Speicherkarten immer auf die Seitengröße (meist 4kB) ausgerichtet sind
– Versuch, den gesamten Inhalt einer Datei zu laden, die signifikant größer ist als der verfügbare Speicherplatz, kann zu schweren Thrashs führen, da das Betriebssystem von der Festplatte in den Speicher liest und gleichzeitig die Seiten vom Speicher zurück auf die Festplatte schreibt.

POSIX Shared Memory Segments:

Auf einen geteilten Speicher können mehrere Prozesse zugreifen, um eine Kommunikation zwischen ihnen bereitzustellen oder um redundante Kopien zu vermeiden.

+ Shared Memory ist ein einfaches und effizientes Mittel, um Daten zwischen Programmen zu übergeben
+ benötigen nicht zwingend syscalls und sind daher schneller
+ Programme können je nach Kontext auf einem einzelnen Prozessor oder auf mehreren separaten Prozessoren laufen.

- da mehrere Prozesse auf den gleichen Speicherplatz zugreifen, kann es zu sogenannten „concurrency“ Problemen kommen, die jedoch mit Semaphoren größtenteils vermieden werden können

Unix Domain Socket:
Sind Sockets die zwischen Prozessen auf einem Unix System für bi-/multidirektionale Kommunikation genutzt werden können. Die Sockets sind wie FIFO-Pipes, nur dass die Kommunikation in beide/alle Richtungen erfolgen kann. Die Kommunikation findet jedoch nicht über die Datei, sondern über die socket Schnittstelle statt.
+ keine concurrency Probleme
+ einfache Umwandlung zu network sockets, also leichter Übergang von Kommunikation auf einem System zu Kommunikation zwischen Systemen
– für die Kommunikation mehrerer Prozesse braucht man mehrere sockets, da eine socket immer nur eins-zu-eins Kommunikation kann
– Ressourcenintensiv, da jede Nachricht über das OS läuft

Gibt es weitere Kommunikationsmöglichkeiten?
Kommunikation über Dateien:
Ein sogenannter Austausch von Dateien, wobei ein Prozess die Daten in Dateien schreiben kann, die ein anderer Prozess ebenfalls lesen (oder beschreiben) kann.

Message Queue(Nachrichtenschlange):
Hier werden Nachrichten von einem Prozess an eine Message Queue geschickt, die in Form einer verketteten Liste verwaltet wird. Dort kann die Nachricht von einem anderen Prozess abgeholt werden. Jede Message Queue hat eine eindeutige Kennung, wodurch sie auch von einem anderen Prozess identifiziert werden kann
Sie erlauben mehreren Prozessen in die Queue zu schreiben, ohne dass sie direkt verknüpft sein müssen.

Für alle: http://beej.us/guide/bgipc/output/html/singlepage/bgipc.html
https://de.wikipedia.org/wiki/Interprozesskommunikation