KCKB UPJS

UNIVERZITA PAVLA JOZEFA ŠAFÁRIKA V KOŠICIACH

Financované Európskou úniou NextGenerationEU

PLÁN OBNOVY

MINISTERSTVO INVESTÍCIÍ, REGIONÁLNEHO ROZVOJA A INFORMATIZÁCIE SLOVENSKEJ REPUBLIKY

7. Rozšírené techniky reverzného inžinierstva a analýzy malvéru v OS Linux

Kontext a evolúcia hrozieb v prostredí Linux

- **Dominancia Linuxu a rozširovanie "Attack Surface":**
  - Linux poháňa 90 % cloudovej infraštruktúry (AWS, Azure, Google Cloud), 100 % superpočítačov a väčšinu IoT zariadení.
  - Kompromitácia Linux servera má často dopad na tisíce koncových klientov (Supply Chain Attacks).
- **Zmena paradigmy a jazykov:**
  - Posun od skriptových útokov k sofistikovaným binárkam.
  - Nárast malvéru písaného v moderných jazykoch ako **Golang** a **Rust**. Tieto jazyky produkujú obrovské binárne súbory, majú neštandardné volacie konvencie a komplexný runtime, čo extrémne sťažuje reverzné inžinierstvo.
- **Technologický posun - APT skupiny:**
  - Adaptácia techník známych z Windows (Process Hollowing, DLL Injection) do prostredia ELF (SO Injection, Reflektívne načítanie).
  - Skupiny ako Winnti, Lazarus či Turla aktívne vyvíjajú Linux-native nástroje.
- **Hlavné výzvy:**
  - **Diverzita architektúr:** Jeden botnet musí bežať na x86, ARM, MIPS, PowerPC, čo vyžaduje cross-compilation a multi-arch payloady.
  - **Statické linkovanie:** Malvér často neobsahuje závislosti na systémových knižniciach (libc), ale má ich vkompilované ("všetko v jednom"). To zväčšuje veľkosť súboru a zahlcuje analytika štandardným kódom knižníc (FLIRT signatúry sú menej účinné).

Financované Európskou úniou NextGenerationEU
PLÁN OBNOVY
MINISTERSTVO INVESTÍCIÍ, REGIONÁLNEHO ROZVOJA A INFORMATIZÁCIE SLOVENSKEJ REPUBLIKY

Extrakcia ukrytého kódu: Manipulácia formátu ELF

- **ELF (Executable and Linkable Format) - Teória vs. Prax:**
  - Štandardný formát pre spustiteľné súbory, zdieľané knižnice a core dumpy.
- **Kritická diskrepancia v parsovaní:**
  - *Kernel View (Loader):* Zameriava sa na efektivitu a rýchlosť. Jadro číta `Elf64_Ehdr` (hlavičku) a iteruje cez `Elf64_Phdr` (Program Headers) typu `PT_LOAD`, aby namapovalo segmenty do virtuálnej pamäte. Všetko ostatné je pre spustenie irelevantné.
  - *Analyst View (Tools):* Nástroje ako `readelf`, `objdump` a GDB sa spoliehajú na `Section Header Table` (SHT) pre symboly, debug informácie a členenie kódu.
- **Technika útoku - Štrukturálna malformácia:**
  - **SHT Stripping/Corruption:** Útočník nastaví `e_shoff` (offset tabuľky sekcií) na 0 alebo na náhodnú hodnotu. Program beží bez problémov, ale analytické nástroje (readelf, objdump) stratia schopnosť zobraziť sekcie a symboly, čím sa sťaží statická analýza.
  - **Header Overlapping:** Prekrývanie hlavičiek (napr. PHT začína vo vnútri ELF hlavičky), aby sa ušetril priestor a zmiatol disassembler, ktorý očakáva štandardné zarovnanie.
  - **Falošné sekcie:** Vytvorenie sekcií s nereálnymi veľkosťami, ktoré spôsobia "buffer overflow" alebo pád analytického nástroja pri pokuse o načítanie.

Financované Európskou úniou NextGenerationEU
PLÁN OBNOVY
MINISTERSTVO INVESTÍCIÍ, REGIONÁLNEHO ROZVOJA A INFORMATIZÁCIE SLOVENSKEJ REPUBLIKY

Metodika extrakcie poškodených ELF súborov

- **Problém:** Statická oprava binárky na disku je časovo náročná a náchylná na chyby (guesswork).
- **Riešenie - Runtime Dump (Dynamická extrakcia):**
  - Využívame fakt, že jadro už úspešne parsovalo súbor.
- **Krok za krokom:**
  - **Zastavenie pred spustením:** Použitie `gdb` s príkazom `starti` (zastaví na prvej inštrukcii, často v dynamickom linkeri) alebo skriptovanie pomocou `ptrace`.
  - **Lokalizácia v pamäti:** Analýza súboru `/proc/<PID>/maps`. Hľadáme bázu, kde je namapovaný hlavný spustiteľný súbor (často s príznakmi `r-xp`).
  - **Memory Dump:** Použitie príkazu `gcore` alebo manuálny dump cez GDB: `dump memory dump.bin 0xStartAddr 0xEndAddr`.
  - **Rekonštrukcia (Fixing):** Surový dump z pamäte má iné zarovnanie ako súbor na disku (Memory Alignment vs Disk Alignment).
    - Je nutné upraviť ELF hlavičku v dumpe tak, aby `File Offset` sekcií zodpovedal ich pozícii v dumpe.
    - Nástroje na automatizáciu: LIEF (pre programatickú úpravu hlavičiek v Pythone), linux-injector/elf-dump nástroje, a pre RAM forenziu Volatility plugin linux.elfs alebo linux_dump_map.
- **Výsledok:** Získame validný ELF, kde sú rozbalené dáta a opravené skoky, pripravený pre import do Ghidry.

Financované Európskou úniou NextGenerationEU
PLÁN OBNOVY
MINISTERSTVO INVESTÍCIÍ, REGIONÁLNEHO ROZVOJA A INFORMATIZÁCIE SLOVENSKEJ REPUBLIKY

Advanced Packing a Anti-UPX techniky

- **Princíp “packovania”:** Kompresia alebo šifrovanie spustiteľného kódu, ktorý je za behu rozbalený malým "stubom".
- **UPX (Ultimate Packer for eXecutables):** Najrozšírenejší open-source packer.
- **Anti-UPX modifikácie (Ako rozbiť `upx -d`):**
  - **Magic Bytes:** Zmena reťazca `UPX!` na náhodné znaky bráni detekcii signatúrou.
  - **Overlay Data:** Pridanie "smetia" na koniec súboru, čo mätie výpočet veľkosti rozbaleného súboru.
  - **Štrukturálna sabotáž:** Prepísanie polí `l_info` (veľkosť komprimovaných dát) a `p_info` (veľkosť po dekompresii) vnútorne v UPX hlavičke, ktorú dekompresor potrebuje na alokáciu pamäte.
- **Detekcia špecializovaných packerov:**
  - **Entropia:** Shannonova entropia blízka hodnote 8.0 indikuje šifrované alebo komprimované dáta (kód vyzerá ako šum).
  - **RWX Sekcie:** Prítomnosť segmentov s oprávneniami Read-Write-Execute (`PROT_READ|PROT_WRITE|PROT_EXEC`). Packer musí do sekcie zapisovať (rozbaľovať) a následne ju spustiť. V modernom bezpečnom softvéri je RWX veľká anomália (W^X policy).
- **Metóda manuálneho rozbalenia:**
  - Pri 32-bit (x86): Klasický 'ESP trick' - Hardware breakpoint na zásobníku (sledovanie inštrukcií PUSHAD/POPAD).
  - Pri 64-bit (x64): Sledovanie systémových volaní mmap a mprotect. Packer musí pamäť najprv alokovať (mmap), rozbaliť do nej kód a následne jej dať právo na spustenie (mprotect s flagom PROT_EXEC). Breakpoint sa kladie na zavolanie mprotect alebo na skok na novo-alokovanú OEP (Original Entry Point).
  - Sledovanie inštrukcií dlhého skoku (`JMP far`, `RET`), ktoré vedú z kódu stubu do novo-alokovanej pamäte (OEP).

Financované Európskou úniou NextGenerationEU
PLÁN OBNOVY
MINISTERSTVO INVESTÍCIÍ, REGIONÁLNEHO ROZVOJA A INFORMATIZÁCIE SLOVENSKEJ REPUBLIKY

Control Flow Flattening (CFF)

- **Koncept CFF (OLLVM obfuscator):**
  - Technika zameraná na zničenie čitateľnosti grafu toku riadenia (Control Flow Graph - CFG).
  - Pôvodné logické bloky (if-else, while) sú extrahované a umiestnené na rovnakú úroveň vnútri jedného obrovského prepínača (switch).
- **Anatómia CFF v Assembleri:**
  - **Prológ:** Inicializácia stavovej premennej (napr. `state = 0`).
  - **Dispečer (Dispatcher):** Slučka, ktorá porovnáva `state` s konštantami a skáče na príslušný blok.
  - **Funkčné bloky:** Vykonajú pôvodný kód a následne aktualizujú `state` na novú hodnotu.
  - **Návrat:** Skok späť na dispečera.
- **Vizuálny efekt:**
  - V disassembleri vzniká graf pripomínajúci hviezdicu alebo plochú štruktúru, kde všetko smeruje do jedného uzla. Nie je vidieť sekvenciu operácií.
- **Opaque Predicates (Nepriehľadné predikáty):**
  - Doplnková technika: Podmienky, ktoré sú vždy pravdivé alebo vždy nepravdivé (napr. `7y^2 - 1 != x^2`), ale statický analyzátor to nevie vyhodnotiť a musí analyzovať obe vetvy (aj "mŕtvy kód").
- **Deobfuskácia (Advanced):**
  - Iba statická analýza nestačí.
  - Využíva sa **symbolické vykonávanie (Triton, Angr)** a **Taint Analysis**. Cieľom je zistiť vzorec, ako sa mení stavová premenná, a následne skriptom prelinkovať bloky priamo, obchádzajúc dispečera.

Financované Európskou úniou NextGenerationEU
PLÁN OBNOVY
MINISTERSTVO INVESTÍCIÍ, REGIONÁLNEHO ROZVOJA A INFORMATIZÁCIE SLOVENSKEJ REPUBLIKY

Anti-Reverse Engineering: Anti-Debugging

- **Princíp:** Malvér aktívne skúma svoje prostredie, aby zistil, či je pozorovaný.
- **Technika `ptrace(PTRACE_TRACEME)`:**
  - V Linuxe môže byť jeden proces debugovaný (trasovaný) iba jedným rodičom v danom čase.
  - Malvér zavolá `ptrace(PTRACE_TRACEME, 0, 0, 0)` na začiatku main funkcie.
  - *Scenár 1:* Ak beží normálne, volanie vráti 0. Malvér vie, že je "čistý".
  - *Scenár 2:* Ak ho už drží GDB alebo strace, volanie zlyhá (return -1, errno EPERM). Malvér sa okamžite ukončí alebo spustí benignú vetvu kódu.
- **Pasívna detekcia (`/proc` filesystem):**
  - `/proc/self/status`: Kontrola poľa `TracerPid`. Ak je hodnota > 0, proces je sledovaný.
  - `/proc/self/cmdline`: Kontrola argumentov, či neobsahujú `--debug` alebo známe analytické flagy.
  - `/proc/self/wchan`: Indikuje, na akom kernel symbole proces spí (napr. `ptrace_stop`).
- **Detekcia softvérových breakpointov:**
  - Debugger (GDB) implementuje breakpoint tak, že prepíše inštrukciu bajtom `0xCC` (INT 3).
  - Malvér môže skenovať vlastný kód v pamäti (checksumming). Ak nájde `0xCC` (x86/x64) alebo sa zmení hash sekcie `.text`, vie, že kód bol modifikovaný.

Financované Európskou úniou NextGenerationEU
PLÁN OBNOVY
MINISTERSTVO INVESTÍCIÍ, REGIONÁLNEHO ROZVOJA A INFORMATIZÁCIE SLOVENSKEJ REPUBLIKY

Detekcia nástroja Frida a časové útoky

- **Frida - Moderná nočná mora malvéru:**
  - Frida je nástroj pre dynamickú inštrumentáciu (injectuje JS engine do procesu).
  - Umožňuje hookovať funkcie v reálnom čase bez rekompilácie.
- **Anti-Frida techniky:**
  - **Memory Scanning:** Prechádzanie `/proc/self/maps` a hľadanie knižníc `frida-agent.so` alebo reťazcov `gum-js-loop`.
  - **Port Scanning:** Frida Server komunikuje defaultne na porte 27047. Malvér sa pokúsi pripojiť na localhost:27047. Ak sa spojenie podarí, ukončí sa.
  - **TCP state checking:** Kontrola `/proc/net/tcp` na prítomnosť podozrivých spojení.
- **Timing Attacks (Časové útoky):**
  - Analýza, emulácia a debugovanie spomaľujú beh kódu o rády (z mikrosekúnd na milisekundy).
  - **Inštrukcia RDTSC (Read Time-Stamp Counter):** Vráti počet cyklov procesora od reštartu.
  - *Algoritmus:*
    - `t1 = RDTSC`
    - `vykonaj_kód()` (alebo len pár inštrukcií)
    - `t2 = RDTSC`
    - `if (t2 - t1 > THRESHOLD) { exit(); }`
  - Ak je medzi t1 a t2 breakpoint, rozdiel bude obrovský.
- **Riešenie:**
  - Použitie "Anti-Anti-Frida" skriptov.
  - Patchovanie inštrukcií RDTSC v binárke (nahradenie NOP).
  - Hypervisor-based tracing, ktorý je pre inštrukcie RDTSC transparentný (nemožné detegovať z user-mode).

Financované Európskou úniou NextGenerationEU
PLÁN OBNOVY
MINISTERSTVO INVESTÍCIÍ, REGIONÁLNEHO ROZVOJA A INFORMATIZÁCIE SLOVENSKEJ REPUBLIKY

Fileless Malware & Process Injection

- **Filozofia Fileless:** "Žiť mimo krajiny" (Living off the Land). Minimalizovať stopu na disku, aby sa vyhlo AV skenerom.
- **Technika `memfd_create()` + `fexecve()`:**
  - Toto je "zlatý štandard" moderného Linux malvéru.
  - `memfd_create("name", 0)`: Systémové volanie, ktoré vytvorí súborový objekt *iba v RAM* (backed by RAM, nie diskom). Správa sa ako súbor, ale nemá cestu na disku.
  - Malvér stiahne payload zo siete, zapíše ho do tohto deskriptora.
  - `fexecve(fd, ...)`: Spustí binárku priamo z deskriptora.
  - **Artefakt:** V `ls -l /proc/<PID>/exe` vidíme linku smerujúcu na `/memfd:nazov (deleted)`.
- **Reflective ELF Loading (Userland Exec):**
  - Malvér implementuje vlastný "loader" v užívateľskom priestore.
  - Sám si alokuje pamäť (`mmap`), rozparsuje ELF hlavičky payloadu, vyrieši relocations a importy a skočí na Entry Point.
  - Výhoda: Nevolá sa `execve`, takže monitorovacie nástroje (auditd, execsnoop) nič nezaznamenajú. Proces sa navonok nezmení.
- **`LD_PRELOAD` Injection (dynamicky linkované):**
  - Environmentálna premenná, ktorá núti dynamický linker načítať špecifickú zdieľanú knižnicu (.so) pred všetkými ostatnými.
  - Slúži ako **Userland Rootkit**: Hookovanie funkcií `readdir` (na skrytie súborov), `open` (na ochranu configov) alebo `SSL_write` (na odpočúvanie HTTPS komunikácie).

Financované Európskou úniou
NextGenerationEU
PLÁN OBNOVY
MINISTERSTVO INVESTÍCIÍ, REGIONÁLNEHO ROZVOJA A INFORMATIZÁCIE SLOVENSKEJ REPUBLIKY

Stealth & Persistence: Kernel Rootkity

- **Ring 0 - Absolútna moc:**
  - Loadable Kernel Modules (LKM) bežia v jadre s najvyššími privilégiami. Môžu meniť správanie OS pre všetky procesy.
- **Syscall Hooking (Klasická metóda - historická, ťažko použiteľné vo verziách 5 a 6, vzhľadom na RO nastavenie tabuľky volaní):**
  - Jadro má tabuľku `sys_call_table` (pole ukazovateľov na funkcie).
  - Rootkit prepíše ukazovateľ pre `sys_read` alebo `sys_getdents` (get directory entries) na svoju škodlivú funkciu.
  - Škodlivá funkcia zavolá originálnu, vyfiltruje výstup (vymaže riadok s názvom malvéru) a vráti upravený výsledok užívateľovi.
- **Moderná metóda (Ftrace Hooking)**: Dnešné rootkity (napr. Reptile) využívajú legitímny framework jadra ftrace (Function Tracer). Ten umožňuje dynamicky 'zavesiť" (hook) vlastnú funkciu na takmer akúkoľvek inú funkciu v jadre (napr. sys_read) bez potreby prepisovať chránené pamäťové štruktúry."
- **DKOM (Direct Kernel Object Manipulation) - Pokročilá metóda:**
  - Namiesto zmeny kódu (hooking) meníme dáta.
  - Každý proces je v jadre reprezentovaný štruktúrou `task_struct` a je súčasťou prepojeného zoznamu (linked list).
  - **Hide Process:** Rootkit upraví ukazovatele `next` a `prev` v zozname úloh tak, aby škodlivý proces "preskočili".
  - Proces stále beží (scheduler o ňom vie cez iné štruktúry), ale nástroje ako `ps`, `top` alebo `/proc` iterujúce cez hlavný zoznam ho nevidia.
- **Case Study: Syslogk:**
  - Sofistikovaný rootkit, ktorý využíva Netfilter hooky (sieťová úroveň jadra).
  - Sleduje prichádzajúce TCP pakety. Ak nájde paket s konkrétnou sekvenciou dát ("Magic Packet"), aktivuje reverzný shell.
  - Neviditeľný pre netstat (lebo neotvára socket, hookuje priamo spracovanie paketu).

Financované Európskou úniou
NextGenerationEU
PLÁN OBNOVY
MINISTERSTVO INVESTÍCIÍ, REGIONÁLNEHO ROZVOJA A INFORMATIZÁCIE SLOVENSKEJ REPUBLIKY

eBPF: Nová hranica stealth technológií

- **Čo je eBPF (Extended Berkeley Packet Filter)?**
  - Revolučná technológia v Linux jadre. Umožňuje nahrávať a spúšťať mini-programy (bytecode) v jadre bez potreby písať LKM alebo rekompilovať jadro.
  - Používa sa na performance monitoring, sieťovú filtráciu a bezpečnosť.
- **Zneužitie na "Malware Next-Gen":**
  - eBPF programy môžu byť pinnuté do BPF filesystému a prežiť zánik procesu, ktorý ich vytvoril, ale štandardne neprežívajú reboot systému.
  - eBPF programy sú extrémne ťažko detegovateľné, pretože nie sú súbormi na disku v tradičnom zmysle.
  - **XDP (eXpress Data Path):** Umožňuje malvéru manipulovať s paketmi priamo na sieťovej karte (NIC driver), skôr než ich vidí OS (iptables, tcpdump).
- **Príklady hrozieb:**
  - **BPFDoor:**
    - Pasívny backdoor. Pripojí BPF filter na raw socket.
    - Číta všetku prevádzku, aj keď ju lokálny firewall blokuje.
    - Reaguje na príkazy v paketoch, pričom pre systém sa javí ako legitímny proces.
  - **Symbiote:**
    - "Impossible-to-detect" rootkit. Hookuje sa do všetkých bežiacich procesov.
    - Používa BPF na skrytie vlastnej sieťovej prevádzky. Ak administrátor spustí `tcpdump`, Symbiote injektuje BPF filter, ktorý z výstupu vymaže jeho pakety.
- **Detekcia:**
  - Tradičné nástroje sú “slepé”.
  - Nutnosť použiť `bpftool prog show` na výpis nahratých BPF programov.

Financované Európskou úniou
NextGenerationEU
PLÁN OBNOVY
MINISTERSTVO INVESTÍCIÍ, REGIONÁLNEHO ROZVOJA A INFORMATIZÁCIE SLOVENSKEJ REPUBLIKY

Forenzná rekonštrukcia a záver

- **Význam Pamäťovej forenzie (Memory Forensics):**
  - V ére fileless malvéru a rootkitov je disk nedôveryhodný. Dôležité informácie sú prevažne je v RAM.
  - Získanie obrazu RAM: `LiME` (Linux Memory Extractor) alebo dump z virtualizačnej vrstvy (napr. VMware snapshot).
- **Nástroje a Postupy:**
  - **Volatility 3:** Framework na analýzu RAM. Plugin `linux.pslist` vs `linux.psscan` (rozdiel odhalí DKOM skryté procesy).
  - **LIEF (Library to Instrument Executable Formats):** Python knižnica na programatickú úpravu ELF. Umožňuje automatizovane opraviť poškodené hlavičky dumpnutých binárok.
- **Zhrnutie prednášky:**
  - Čistá statická analýza je pre pokročilý malvér nedostatočná; hybridný prístup (statická + dynamická) je nevyhnutný.
  - **"Dirty Hands" prístup:** Bezpečnostný expert musí rozumieť interným štruktúram jadra, formátu ELF a assembleru. Klikanie v GUI nástrojoch nestačí.
  - eBPF a Fileless techniky definujú novú éru Linux malvéru, kde hranica medzi legitímnym nástrojom a nebezpečenstvom mizne.

Financované Európskou úniou NextGenerationEU
PLÁN OBNOVY
MINISTERSTVO INVESTÍCIÍ, REGIONÁLNEHO ROZVOJA A INFORMATIZÁCIE SLOVENSKEJ REPUBLIKY

KCKB UPJS

UNIVERZITA
PAVLA JOZEFA ŠAFÁRIKA
V KOŠICIACH

Financované
Európskou úniou
NextGenerationEU

PLÁN OBNOVY

MINISTERSTVO
INVESTÍCIÍ, REGIONÁLNEHO ROZVOJA
A INFORMATIZÁCIE
SLOVENSKEJ REPUBLIKY

Ďakujem za pozornosť.