

# The Unified Kill Chain Applied to the Linux Malware Landscape

Thibault Debatty<sup>1</sup> Zacharia Mansouri<sup>1</sup>

<sup>1</sup>Royal Military Academy, Brussels, Belgium

## Linux Malware

For a long time, Linux-targeting malware flew under the radar, limited to proof-of-concept samples and largely dismissed by the antivirus industry. That changed in late 2014, when VirusTotal spotlighted its rising prevalence, prompting a shift in awareness across the security landscape. Since then, Linux malware has grown increasingly sophisticated, exploiting cross-platform flexibility, privilege escalation vulnerabilities and using tailored UPX packers to conceal its payloads. Linux malware has diversified, adopting exotic names like Auto-Color, Sword2033, Cl0p, SprySOCKS, Ebury, FASTCash, Springtail, etc.

## Unified Kill Chain

The Unified Kill Chain (UKC) is designed to detail the tactics used in modern cyber attacks by Advanced Persistent Threats (APTs) and ransomware groups. The UKC is divided into 18 phases to fully describe attacks. Here is a UKC-based analysis tailored to the evolving tactics of Linux-targeting malware.

### 1. Reconnaissance

Some Linux-specific (network) tools like ping, traceroute, netcat, nmap and nslookup are used for active reconnaissance. Passive reconnaissance includes OSINT tools (Google, theHarvester...), assessment of the target organization's operating environment and physical searches.

### 2. Resource Development

New Linux malware and adapted threats (e.g., SprySOCKS) target Linux with architecture-specific payloads. Persistence and evasion use Linux mechanisms like /etc/rc.local, /etc/init.d/, cron, symbolic links, udev rules, ELF header tampering, and modified utilities (ls, ps, netstat, ldd, crontab, top).

### 3. Delivery

Some phishing campaigns spread via phishing links that download archive files containing malicious shortcut (.lnk) files or ELF executables.

### 4. Social Engineering

Attackers use Linux shell commands and ClickFix tactics—like fake CAPTCHA or prompts—to trick users into copying malicious shell commands for manual execution, bypassing security controls.

### 5. Exploitation

Linux exploitation includes common kernel vulnerabilities and lists Linux-specific privilege escalation exploits such as Polkit (CVE-2021-4034, pwnkit) and Dirty COW (CVE-2016-5195), as well as the exploitation of Linux-specific system features like GRUB manipulation, initramfs modifications, PolicyKit (Polkit) abuse and D-Bus exploitation.

### 6. Persistence

Linux persistence techniques include GRUB bootloader modification, initramfs manipulation, modifying system initialization (rc) scripts or /etc/init.d/

files, using systemd or chkconfig, cron jobs, modifying user-level configuration files (~/.bashrc, ~/.bash\_profile), PolicyKit (Polkit) rules, D-Bus service files, NetworkManager dispatcher scripts, file infection of ELF files, rootkits (/etc/ld.preload), udev rules, shared library (.so files) hijacking and daemonization.

### 7. Defense Evasion

Many evasion techniques involve Linux-specific mechanisms such as removing symbols from ELF binaries, using ptrace system calls, examining /proc/self/status, using madvise() with MADV\_DONTDUMP to prevent specific memory ranges from being included in core dumps, checking Linux-specific sandbox artifacts (e.g., /proc/vz), disabling SELinux, modifying UFW or iptables firewall rules, and wrapping Linux system executables (who, netstat, ps, ldd, crontab, top).

### 8. Command & Control (C2)

C2 channels are disguised as kernel processes, use the BPF engine in covert channels and also hide their network activity by hooking file access functions for /proc/net/tcp and internal communication using Unix sockets and signals (e.g., kill syscall interception), and tampering with Linux utilities like crontab, lsof, ldd, and top.

### 9. Pivoting

Techniques include the use of iptables rules for Adversary-in-the-Middle (AitM) attacks and internal communication via Unix sockets, alongside the modification of Linux system binaries.

### 10. Discovery

Attackers collecting system information by reading Linux-specific files like /proc/version, /etc/lsb-release, /etc/passwd, and inspecting /proc and sysfs virtual file systems. It also involves Linux commands such as ps aux, who, uname -a, df, ls -hal, arp -a, and crontab. Detection of virtualized environments also relies on Linux-specific paths like /proc/vz.

### 11. Privilege Escalation

Techniques in this phase include exploiting Polkit (pwnkit) and Dirty COW vulnerabilities, abusing PolicyKit rules, D-Bus exploitation, NetworkManager dispatcher scripts, GRUB bootloader and initramfs manipulation, using setuid()/setgid() functions, /etc/ld.preload manipulation, and cron job abuse.

### 12. Execution

Attackers abuse Linux utilities (bash, perl) and commands (wget, curl, chmod); use process masquerading (kdevtmpfs), OpenSSH injection, and Linux-specific syscalls (ptrace, dlopen); and persist via startup scripts (rc, /etc/init.d/, systemd) and libraries (/etc/ld.preload, libkeyutils.so.1).

### 13. Credential Access

This phase details techniques focused on SSH credential theft and manipulation, including modifying authorized\_keys, using ARP cache poisoning with iptables for SSH AitM attacks, capturing credentials via OpenSSH client hooks, and dumping Linux-specific files like /etc/passwd, /etc/shadow, and bash history. Hooking PAM authentication functions

(pam\_authenticate) is also Linux-specific.

### 14. Lateral Movement

This is highly OS-dependent, with prevalent methods including the reuse of stolen SSH credentials, installation of backdoor SSH accounts, creating reverse shells (often Bash), and abusing native Linux tools like curl, wget, chmod, chattr, ssh, base64, chroot, crontab, ps, and kill.

### 15. Collection

Attackers gather information extensively from Linux-specific file systems like /proc and sysfs, using Linux commands (uname -a, df, ls -hal, ps), and extracting data from Linux-specific log files (/var/run/utmp, /var/log/wtmp, /var/log/lastlog) and configuration files (/etc/resolv.conf, /etc/hosts).

### 16. Exfiltration

While the act of exfiltration is general, the methods described include abusing SSH service (often Linux), using tar to archive data, utilizing curl to send system information, and employing BPF engine in covert channels, which is Linux-specific. It also includes exfiltrating data from Linux-specific log files.

### 17. Impact

This phase includes commands specific to the ESXi hypervisor (e.g., esxcli vm process kill for virtual machines) and techniques like manipulating Linux system utilities (ps, iptables, who, netstat), disabling Linux security enforcement (SELinux, UFW, iptables), and using kernel-level rootkits.

### 18. Objectives

Strategic goals of modern Linux attacks are diverse, including financial gain (ransomware, cryptomining, money laundering), espionage (backdoors, rootkits, full APTs), sabotage (DDoS)...and are performed by groups such as Earth Lusca (China-suspected), APT36 (Pakistan-suspected), Equation (US-suspected), Lazarus (North Korea-sponsored), etc.

## References

- [1] E. Cozzi, M. Graziano, Y. Fratantonio & D. Balzarotti, Understanding Linux Malware, *2018 IEEE Symposium on Security and Privacy*
- [2] J. Baines, Programming linux anti-reversing techniques
- [3] Pangu Lab, Bvp47: Top-tier Backdoor of US NSA Equation Group
- [4] P. Pols, The Unified Kill Chain
- [5] Y. He, R. Guo, Y. Xing, X. Che, K. Sun, Z. Liu, K. Xu & Q. Li, Cross Container Attacks: The Bewildered eBPF on Clouds, *32nd USENIX Security Symposium*
- [6] <https://attack.mitre.org/groups/>
- [7] Selected blog sources: asec.ahnlab.com, blog.exatrack.com, blog.sekoia.io, dfir.ch, haxrob.net, hunt.io, research.checkpoint.com, unit42.paloaltonetworks.com, www.aon.com, www.bleepingcomputer.com, www.elastic.co, www.group-ib.com, www.security.com, www.securonix.com, www.sentinelone.com, www.seqrte.com, www.techtarget.com, www.trendmicro.com, www.uptycs.com, www.usna.edu, zw01f.github.io