



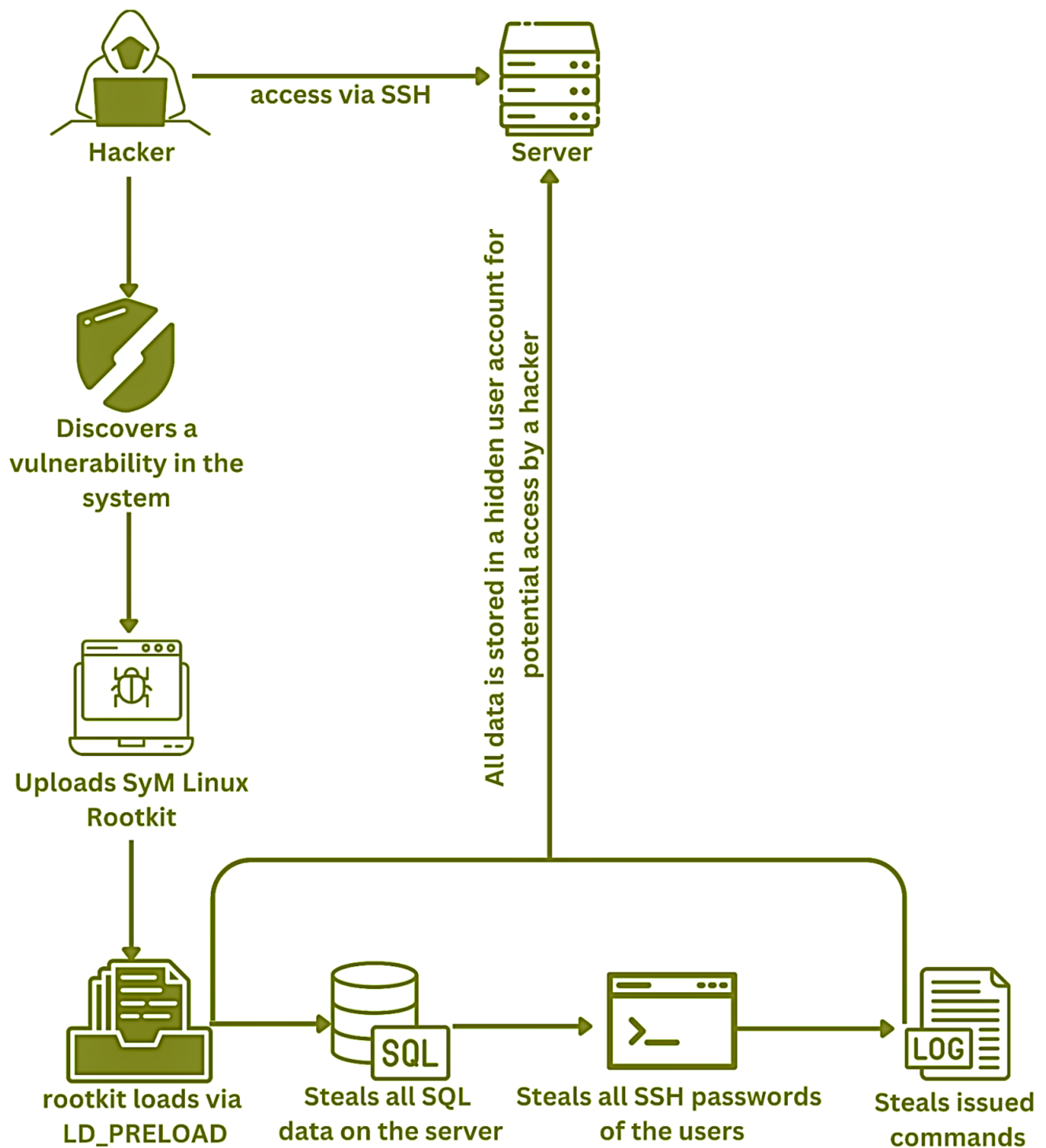
SyM Linux Rootkit

CTI Report

Contents


Contents.....	2
Attack Chain of SyM Linux Rootkit.....	3
About SyM Linux Rootkit.....	4
Features of SyM Linux Rootkit.....	4
SyM Linux Rootkit From The Eyes of Attackers.....	5
Basic Analysis of SyM Linux Rootkit.....	7
About Hack The Planet APT Group.....	11
Categorization of SyM Linux Rootkit.....	13
Malware Family.....	13
APT Group.....	13
Threat Category.....	13
IOCs.....	13
HASHs:.....	13

Attack Chain of SyM Linux Rootkit



About SyM Linux Rootkit

[SOURCE] Linux rootkit
by succumb - Saturday October 28, 2023 at 06:47 AM



succumb

uid=0

GOD

Posts: 19
Threads: 3
Joined: Jun 2023
Reputation: 99

10-28-2023, 06:47 AM (This post was last modified: 11-05-2023, 03:48 AM by succumb.)

SyM Linux Rootkit
SyM is a universal user-mode Linux rootkit that will sustainably hold root persistence across all Linux kernel versions, and will successfully bypass any EDR or rootkit detection software. SyM will also come with a plethora of features capable of stealing important files such as SQL database backups, .git, and other configuration files; And much more. Along with being the first of its kind SyM implements some API system call hooking that has never been seen before which makes it such a unique, and undetectable rootkit experience.

C&C / C2 / backdoor methods:

- **ICMP backdoor**
Use a unique magic identifier to open a reverse shell
- **accept () backdoor**
Use a unique magic identifier to open a listening TCP server
- **PAM backdoor**
Direct interactive SSH backdoor with custom hidden port, username, and password

Internal System Logging:

- **SSH Log**
Log all incoming and outgoing SSH authorizations in plaintext by hooking pam_vprompt, read, and write API calls
- **Execution Log**
Log all normal (including root) user command execution flow

Hiding Self / Rootkit

- Hide all files, processes, open ports, and all connections based on unique magic identifier
- Hide process map files, to prevent direct mapping of process and being able to identify rootkit
- Hide any file, or directory of choice
- All rootkit master created directories and files will be kept track of, so no need to manually add or edit anything to keep it hidden!
- Note: It is possible to forge or fake as any other installed software, service, or similar

EDR Bypass / Evasion

- Hooking API calls to hide it's self from / proc * / * maps as well as many other system locations
- Bypassing SELinux and GRSec
- Bypasses and hides from SentinelOne and other similar software

File Stealer

- By scanning and keeping tracking of a user made list of interesting files and directories the rootkit is capable of stealing anything on the fly and uploading it directly to an external server
- Stuff like SQL databases are stolen automatically by default!

SyM is a user-mode universal Linux rootkit that can successfully evade any rootkit detection program or EDR and maintain root persistence across all Linux kernel versions. Along with many other features, SyM will be able to steal crucial files like.git, SQL database backups, and other configuration files. What makes SyM such a special and undetectable rootkit experience is that it implements some API system call hooking that has never been seen before in addition to being the first of its type.

Features of SyM Linux Rootkit

C&C / C2 / backdoor methods:

- **ICMP backdoor**
Use a unique magic identifier to open a reverse shell
- **accept () backdoor**
Use a unique magic identifier to open a listening TCP server
- **PAM backdoor**
Direct interactive SSH backdoor with custom hidden port, username, and password

Internal System Logging:

- **SSH Log**
Log all incoming and outgoing SSH authorizations in plaintext by hooking pam_vprompt, read, and write API calls
- **Execution Log**
Log all normal (including root) user command execution flow

Hiding Self / Rootkit

- Hide all files, processes, open ports, and all connections based on unique magic identifier
- Hide process map files, to prevent direct mapping of process and being able to identify rootkit
- Hide any file, or directory of choice
- All rootkit master created directories and files will be kept track of, so no need to manually add or edit anything to keep it hidden!
- Note: It is possible to forge or fake as any other installed software, service, or similar

EDR Bypass / Evasion

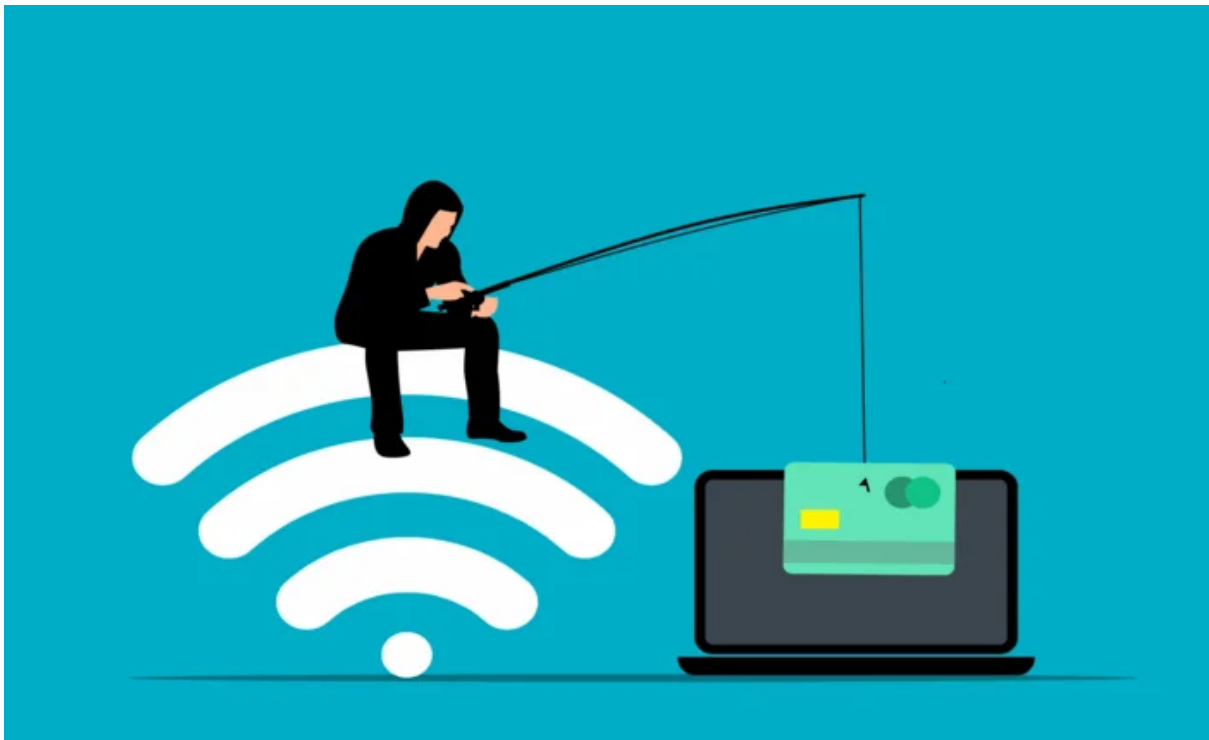
- Hooking API calls to hide it's self from / proc * / * maps as well as many other system locations
- Bypassing SELinux and GRSec
- Bypasses and hides from SentinelOne and other similar software

File Stealer

- By scanning and keeping tracking of a user made list of interesting files and directories the rootkit is capable of stealing anything on the fly and uploading it directly to an external server
- Stuff like SQL databases are stolen automatically by default!

The tool has many features within itself. C&C/C2 backdoor methods, Internal system logging, Hiding itself, EDR Bypass, File Stealer are the main contents of the features. All the features related with the main features are explained in detail by the seller.

SyM Linux Rootkit From The Eyes of Attackers



In the first phase, the hacker aims to send the SyM Linux Rootkit payload on a targeted system. The Hacker can achieve this by exploiting system vulnerabilities or using a phishing attack at the level where the payload will be executed through social engineering.

```
login as: adm1n
adm1n@██████████ password:
root@██████████:~#
```

After successfully infecting the system, the rootkit creates an account within the software that the hacker can access. The hacker can gain access to the system by providing a username and password through this account.

```
root@ [REDACTED] ~# ls
backup.sql  execlog  libdl.so  r  sshpass.txt  sshpass2.txt
root@ [REDACTED] :~#
```

I

After successfully infecting the system, the rootkit keeps its files hidden within the system.

The “backup.sql” file steals all the “sql” files on the infected system.

The “execlog” file steals the commands executed on the infected system.

The “libdl.so” file is the replacement shared object that rootkit loads via LD_PRELOAD. It is needed for the proper operation of the rootkit.

The “r” is the packed malicious file. The 'sshpas.txt' and 'sshpas2.txt' files are the ones that keep the stolen SSH passwords.

```
cpp                                libipq.so.0                        modules
discover                          libipq.so.0.0.0                  startpar
ifupdown                          libiptc.so.0                     systemd
init                              libiptc.so.0.0.0                terminfo
klibc-IpHGKKbZiB_yZ7GPagmQz2GwVAQ.so  libseconf                       udev
libip4tc.so.0                     libxtables.so.10                 x86_64-linux-gnu
libip4tc.so.0.1.0                 libxtables.so.10.0.0            xtables
libip6tc.so.0                     lsb
libip6tc.so.0.1.0                 modprobe.d
```

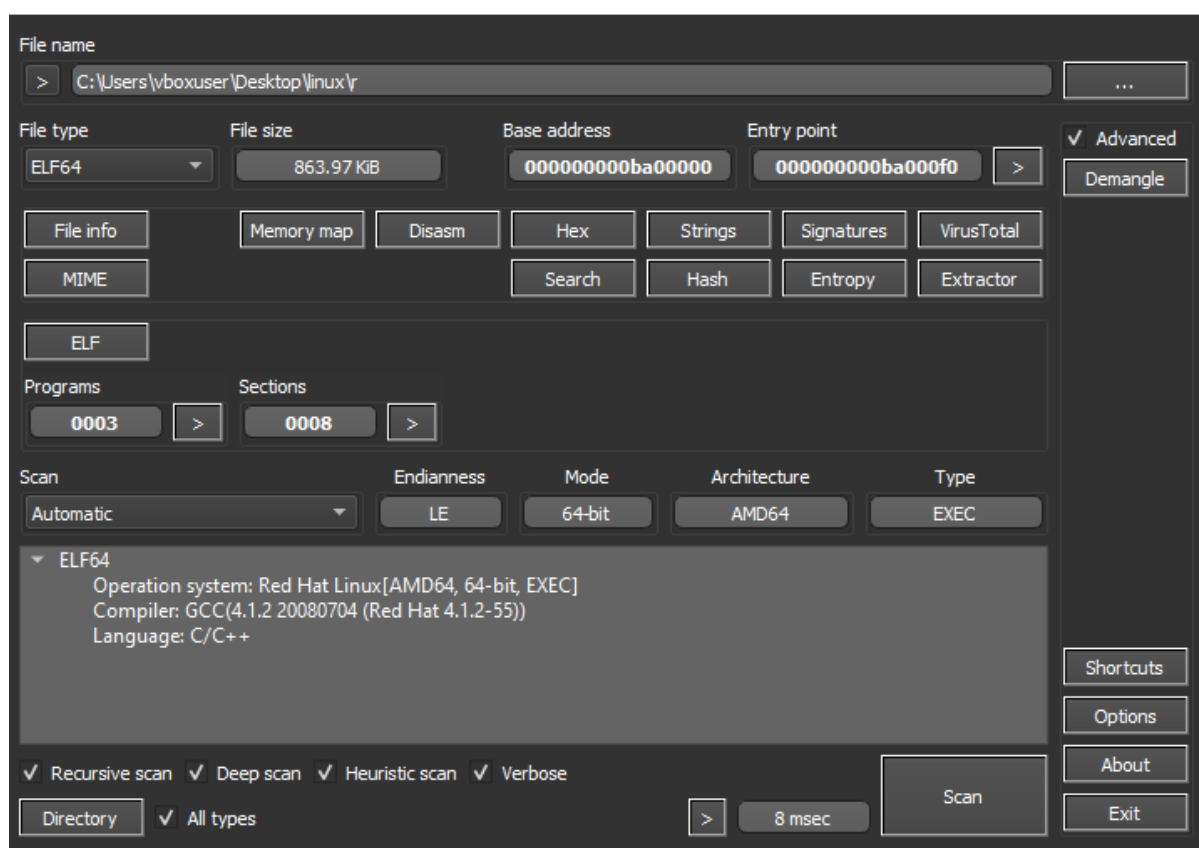
The rootkit directory can be seen in the user that's been created by the rootkit.

```
[REDACTED]@ [REDACTED] :~$ ls /lib
cpp                                libip6tc.so.0.1.0               modprobe.d
discover                          libipq.so.0                     modules
ifupdown                          libipq.so.0.0.0                  startpar
init                              libiptc.so.0                     systemd
klibc-IpHGKKbZiB_yZ7GPagmQz2GwVAQ.so  libiptc.so.0.0.0                terminfo
libip4tc.so.0                     libxtables.so.10                 udev
libip4tc.so.0.1.0                 libxtables.so.10.0.0            x86_64-linux-gnu
libip6tc.so.0                     lsb                              xtables

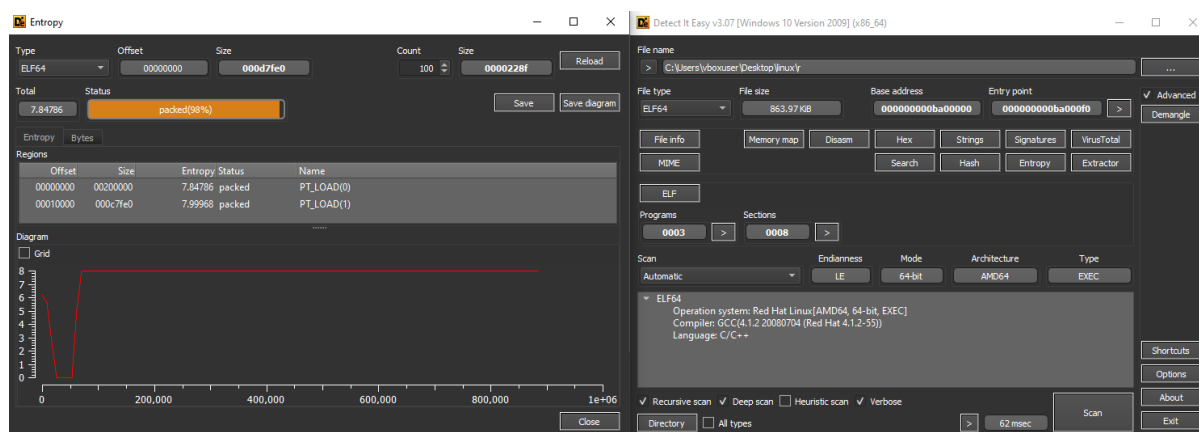
[REDACTED]@ [REDACTED] :~$ cd /lib/libseconf
-bash: cd: /lib/libseconf: No such file or directory
[REDACTED]@ [REDACTED] :~$ ls /lib/libseconf
ls: cannot access /lib/libseconf: No such file or directory
```

But it is hidden and cannot be seen on other accounts. This way, the victim remains unaware of the rootkit injection.

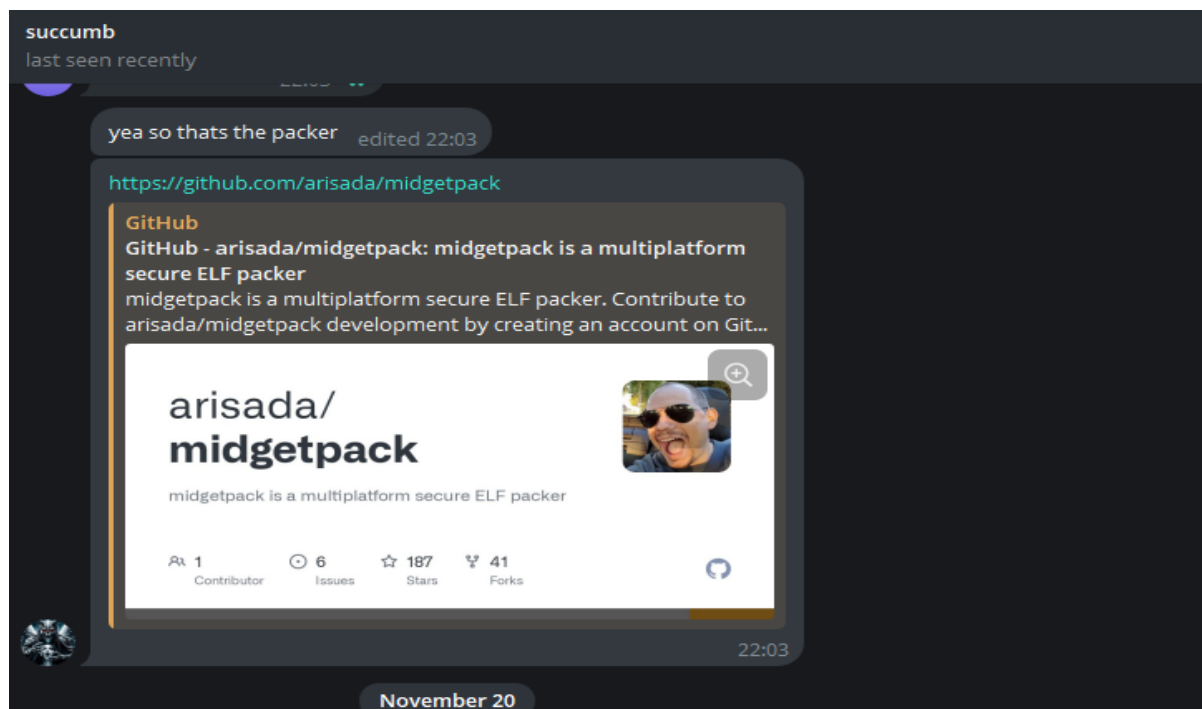
Basic Analysis of SyM Linux Rootkit



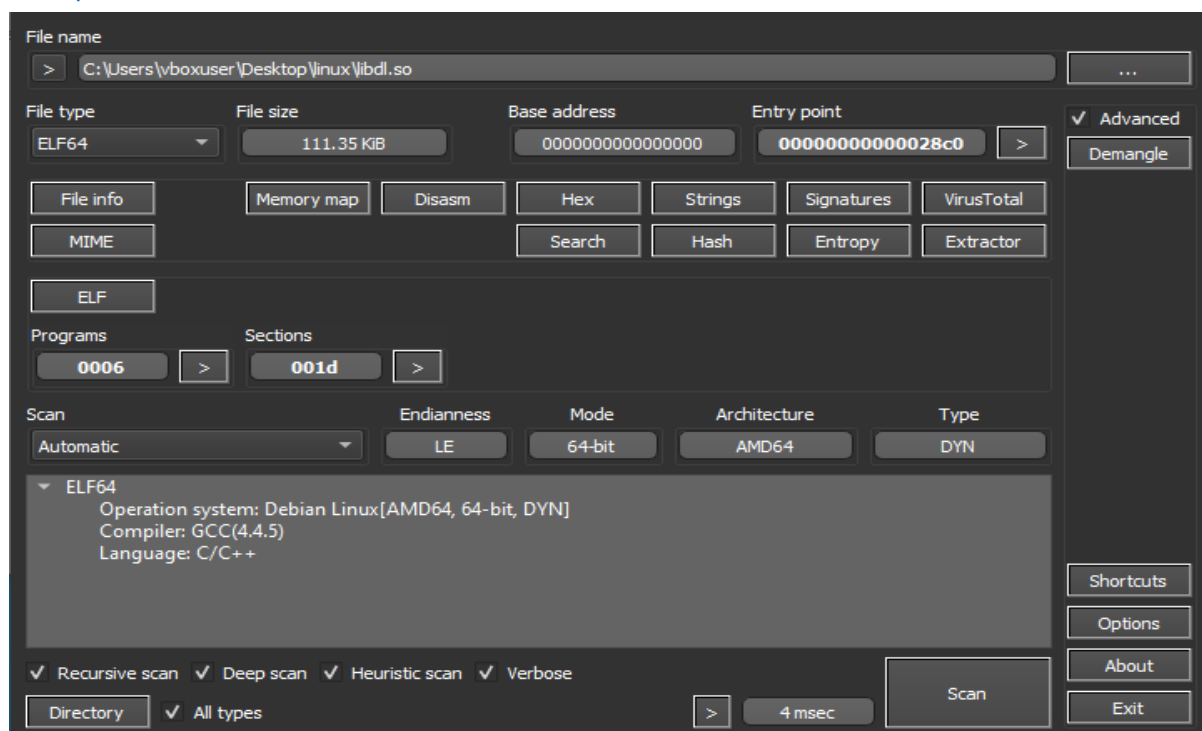
The stub is an ELF sample. It is written in C and has a file size of 863.97KB. It only runs on linux systems.



The software uses an old but still functional packer program to remain FUD (Fully Undetected). Even if the vendor sells the source code of the software, it is still packed after being compiled.



According to the seller, a packer named “midgetpack” is being used for packing the “SyM Linux Rootkit” tool, [midgetpack is a multiplatform secure ELF packer](https://github.com/arisada/midgetpack).



The “libdll.so” file is also written in the same language as the main stub. It is also an ELF file. It has a size of 111.35KB. It is the replacement shard object.


```

7      000183cb      0b A      /tmp/.orbit
8      000183d7      0d A      ld.so.nohwcap
9      000183e8      4f A      access("/etc/ld.so.preload", R_OK) = -1 ENOENT (No such file or directory)
0      00018457      0e A      %m-%d %H:%M:%S
1      00018469      17 A      [%s] [%s] [BLOCKED] %s
2      00018481      0d A      [%s] [%s] %s
3      00018491      0c A      /usr/bin/ssh
4      0001849e      0c A      /usr/bin/scp
  
```

The “libdll.so” file loads via LD_PRELOAD. LD_PRELOAD is a feature required to load shared objects during an ELF's initialization phase.

Scan result:	This file was detected by [0 / 40] engine(s)
File name:	r
File size:	884704 bytes
Analysis date:	2023-11-22 09:05:26
CRC32:	5dd1b071
MD5:	311f32fa3e1a638324343e17ce9d1c12
SHA-1:	43022e9929ea6fb8680203fea73e5f4f6fcd4339
SHA-2:	695c2484c7d506230be9c707991bc81b4f84b16479707e818c38bfa568765b a8
SSDEEP:	12288:8xuuLqmYeHMwZnhZ0pTGDP++Xk62QI5J9bTYL89laNN6s3E+hDJLRa aGYay0sVA4:8TnYHCZiGDWZJsoL8+NNXXRaajxbp

The main stub is fully undetectable and bypasses EDR systems. It is successfully evading detection by security software.

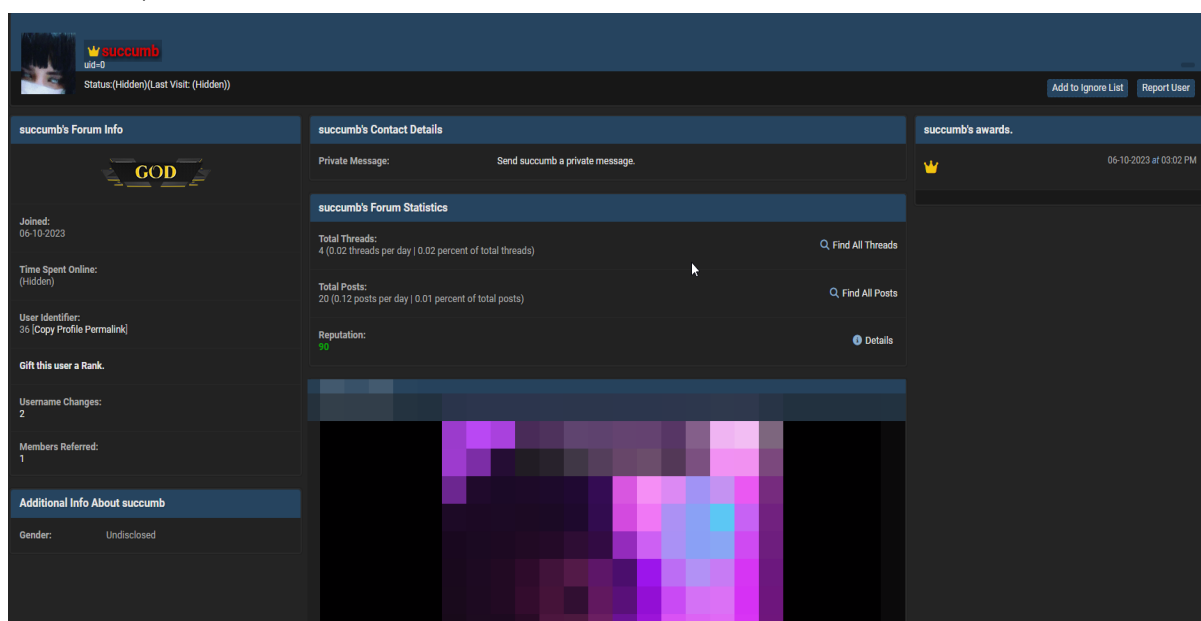
Scan result:	This file was detected by [3 / 40] engine(s)
File name:	libdll.so
File size:	114027 bytes
Analysis date:	2023-11-22 09:10:40
CRC32:	9a71fb67
MD5:	4e152dacab201c5bf5c22c93e31e9475
SHA-1:	914ff4116a5c55c37a157b912a198b23a79d1a70
SHA-2:	296d28eb7b66aa2cbea7d9c2e7dc1ad6ce6f97d44d34139760c38817aec083e 7
SSDEEP:	3072:vPXsx5tclmBrena5yavmTmng8TI1h2TExCxDOl5w:vPXsx5tQB15yavmT mng8TI1h2TExCh

The library file also has a low detection rate. It has been detected by 3 out of 40 antivirus software. It's detected by; Zone Alarm, NOD32 and Microsoft Defender.

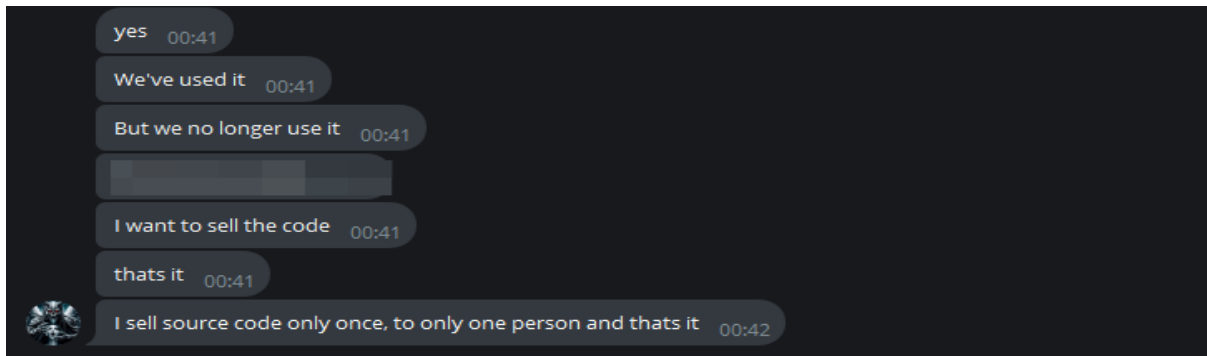
About Hack The Planet APT Group



Hack the planet, or abbreviated as HTP5, is a hacking group. The group was very active in the year 2013 and played a role in many exploits, malicious software, and attacks.



The person who is selling the SyM Linux Rootkit software used to be a member of HTP5. However, the group has recently disbanded. Currently, they are not an active threat actor.



The SyM Linux Rootkit software has been used in various attacks by the HTP5 group. They specifically targeted Bangladesh ISP servers and carried out successful attacks. However, since the group has disbanded, the software is no longer used by the HTP5 group, and its source code has been put up for sale by the developer.

☐ Verified
 ☐ Has App

Show 15

Search:

Date	D	A	V	Title	Type	Platform	Author
2013-05-13				Kloxo 6.1.6 - Local Privilege Escalation	Local	Linux	HTP
2013-05-08				ColdFusion 9-10 - Credential Disclosure	WebApps	Multiple	HTP
2013-05-08				MoinMoin - Arbitrary Command Execution	WebApps	PHP	HTP

Showing 1 to 3 of 3 entries
 FIRST
 PREVIOUS
 1
 NEXT
 LAST

During the period when the HTP5 group was engaged in active attacks, they openly shared exploits with the public, in addition to using their proprietary software for their own operations.

https://www.exploit-db.com/search?e_author=htp5



The most well-known attack by the HTP5 group is the hacking of Linode servers. The group blackmailed Linode, and Linode was compelled by the FBI to disregard the extortion. Subsequently, with the involvement of the FBI in the matter, HTP5 decided to refrain from releasing customer information.

Categorization of SyM Linux Rootkit

Malware Family	APT Group	Threat Category
Orbit	Hack The Planet	Trojan

IOCs

HASHs:

IOC Type	IOC
SHA256	296d28eb7b66aa2cbea7d9c2e7dc1ad6ce6f97d44d34139760c38817aec083e7
SHA256	e0fb1906d7fb9c11b7c9efb47617031d470df4c948554fef52dd6d124d9bb543
SHA256	695c2484c7d506230be9c707991bc81b4f84b16479707e818c38bfa568765ba8
SHA256	548ed9ce697b3d645256d19ad67ff7fdde52d197a6f548e5d6388d0127b7061a



All the **services** you need to keep your **business** secure

Secure your business effectively against
cyber threats and attacks

In **InfinitumIT** we provide
Risk and Threat Analysis
Penetration Testing
Managed Security
Digital Forensics
Consultancy





Services at a glance



consultancy

- Continuous Cyber Security Consultancy
- Continuous Vulnerability Analysis Service
- Managed Detection and Response (MDR) Service
- SOC (Security Operations Center) Service



Managed Security

- Managed Detection and Response (MDR) Service
- SOC (Security Operations Center) Service
- Cyber Incident Response (SOME) Service
- SIEM / LOG Correlation Services



Risk & Threat Analysis

- Cyber Risk and Threat Analysis Service
- Ransomware Risk Analysis Service
- APT Detection & Cyber Hygiene Analysis Service
- Purple Teaming Service



Penetration Testing

- Penetration Testing
- Red Teaming Service
- Source Code Analysis Service



Forensics

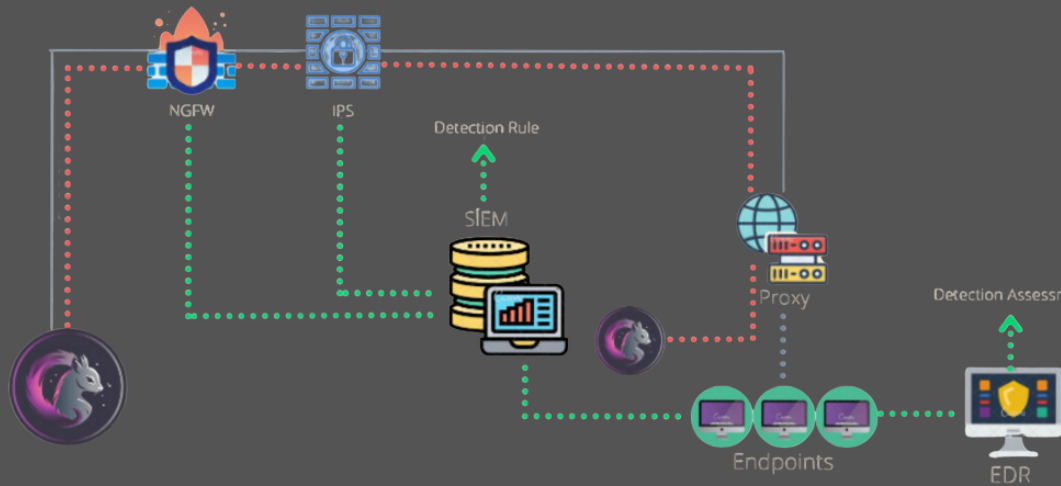
- Network Forensic Service
- Digital Forensic Service
- Mobile Forensic Service





Threatblade

Attack Simulation platform ThreatBlade simulates cyber attacks against your organization's network and systems.



Endpoint Risk Assessment

- Evaluate the security posture of individual endpoints, identify vulnerabilities, and mitigate risks by conducting endpoint-specific scenarios.



Network Risk Assessment

- Continuously monitor the network security posture using network specific attack scenarios, produce trend reports, and improve network security posture.



Identify Weaknesses

- Identify potential weaknesses in an organization's cybersecurity infrastructure and provide actionable insights for improvement purposes.





“Power of Integrated Security”

Your Business's Weaknesses Do you know?

Contact us now to find out



Check Your MDR Healthcheck For Free



@infinitemitlabs



@infinitemitlabs



@infinitemitlab1