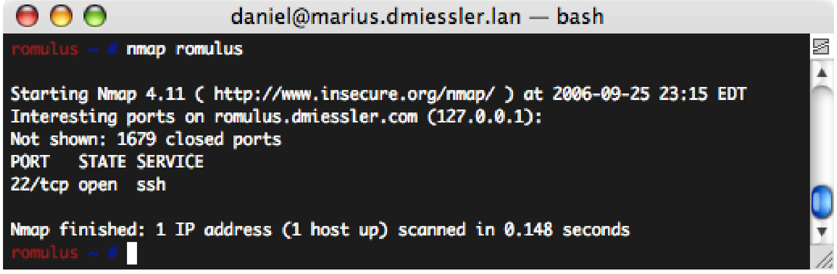
Security: How To Monitor Your Network Connections

Home » Blog » Information Security » Security: How To Monitor Your Network Connections

SEPTEMBER 25, 2006 | INFORMATION SECURITY



```
romulus ~ # nmap romulus

Starting Nmap 4.11 ( http://www.insecure.org/nmap/ ) at 2006-09-25 23:15 EDT
Interesting ports on romulus.dmiessler.com (127.0.0.1):
Not shown: 1679 closed ports
PORT   STATE SERVICE
22/tcp open  ssh

Nmap finished: 1 IP address (1 host up) scanned in 0.148 seconds
romulus ~ #
```

☞ Every Sunday I put out a curated list of the week's most interesting stories in infosec, technology, and humans. You can subscribe to it here.

One of the most important concepts in computer security is “knowing thy system”. This essentially means that in order to be able to protect something you need to have some idea of what it’s doing and/or how it works.

Your computer’s connections to the outside world is among the most important information you can have about your system. In addition to what connections are currently established, you also want to know what ports your computer is “listening” on, or in other words, what ways other systems are able to interact with your computer.

Below I’ll cover how to see who your Windows or Linux computer is currently talking to, and the ways your computer is *willing to talk* through open, listening ports.

Ports

There is often some confusion about what network ports are, and what it means for them to be “open”. Think of network ports as spring-loaded windows on a house. So if someone doesn’t actively hold the window open, it’ll shut automatically and remain closed until it’s opened again.

> If a port is open, it means there’s someone (an application) in the window waiting to speak with someone outside the house. Imagine that each open window has a midget in it, and each midget is waiting to have a certain type of conversation with an outsider. If it’s port 25 that’s open on your machine, then you’ve likely[1] got an email midget in the window waiting to process mail for you. If it’s port 445 that’s open, you’ve probably got a Windows Networkingmidget in there waiting to send and receive files, etc.

The important thing to remember is that when you see a port open on your system, it’s because *something opened it*. Remember, if there wasn’t a midget in the window it would just close by itself. The issue then becomes finding out what program opened the port, and whether or not it’s legitimate.

Windows

Windows has a built-in tool called `netstat` that can show a decent amount of information. If you just have a quick question about a certain port you can use it right from the command line and avoid using a third party application:

```
netstat -an | find "LISTENING"
```

```
TCP    0.0.0.0:135           0.0.0.0:0              LISTENING
TCP    0.0.0.0:445            0.0.0.0:0              LISTENING
TCP    0.0.0.0:1049           0.0.0.0:0              LISTENING
TCP    0.0.0.0:9000           0.0.0.0:0              LISTENING
TCP    0.0.0.0:33333          0.0.0.0:0              LISTENING
```

You want to take note of the red portions: those are the ports that your system is listening for connections on. You can do the same thing and search for established connections as well:

```
netstat -an | find "ESTABLISHED"
```

```
TCP    1.2.3.4:4095           66.102.7.99:80             ESTABLISHED
TCP    1.2.3.4:8324           209.73.177.115:25          ESTABLISHED
```

Here were seeing the systems we're currently connected to, and which the ports the connections are using. Notice that the colon ":" is used to show an ip / port pair. So this is showing that we (1.2.3.4) are connected to Google (66.102.7.99) on port 80 — which means we are browsing the Google website.

Tcpview

For those that want more information about their network connections and/or are graphically inclined, there's a free tool called Tcpview that's a must for any serious Windows user.

Tcpview allows you to view, in real time, the connections that are open on your system. Not only does it update constantly as connections spawn or die off, but *it also shows you what program is responsible for opening a given port on your system*. [For those bent on command line kung-fu, you can get similar functionality from `netstat -anb`]

Linux

Being a Linux/OS X guy myself I would deserve a good pummeling if I didn't show how to get similar information from a *nix system. The best way to do this is with the `lsof` command:

`lsof -i`

```
COMMAND  PID USER   FD   TYPE DEVICE SIZE NODE NAME
dhcpcd  6061 root    4u  IPv4   4510      UDP *:bootpc
sshd    7703 root    3u  IPv6   6499      TCP *:ssh (LISTEN)
sshd    7892 root    3u  IPv6   6757      TCP 10.10.1.5:ssh->
192.168.1.5:49901 (ESTABLISHED)
```

Using `lsof` you can ask to see only TCP or UDP connections, only connections to a certain host, only connections using a certain port, as well as a ton of other options. Here are a few examples:

lsof -iTCP // only TCP lsof -iUDP // only UDP lsof -i :22 // involving port 22 lsof -i :@attacker.com // connections with attacker.com lsof -i :1.2.3.4 // connections to 1.2.3.4 lsof -i :mail.com:25 // connections to mail.com on the SMTP port lsof -i | grep LISTEN // see what's listening lsof -i | grep ESTABLISHED // see what's established

Conclusion

Knowing who your system is talking to (and who it's *willing* to talk to) is crucial to your overall computer security. Using the short guide above you can now gather this information in both Windows and *nix environments.: