

Identifying Open Ports/Services in Windows XP/2003

If you are running Windows XP/2003, the native tools bundled with the operating system allow for mapping of an open TCP/UDP port to a running service or application.

First, we want to use some specific parameters for the NETSTAT.EXE command:

```
C:\>netstat /?
```

Displays protocol statistics and current TCP/IP network connections.

```
NETSTAT [-a] [-b] [-e] [-n] [-o] [-p proto] [-r] [-s] [-v] [interval]
```

-a Displays all connections and listening ports.

-b Displays the executable involved in creating each connection or listening port. In some cases well-known executables host multiple independent components, and in these cases the sequence of components involved in creating the connection or listening port is displayed. In this case the executable name is in [] at the bottom, on top is the component it called, and so forth until TCP/IP was reached. Note that this option can be time-consuming and will fail unless you have sufficient permissions.

-e Displays Ethernet statistics. This may be combined with the **-s** option.

-n Displays addresses and port numbers in numerical form.

-o Displays the owning process ID associated with each connection.

-p proto Shows connections for the protocol specified by proto; proto may be any of: TCP, UDP, TCPv6, or UDPv6. If used with the **-s** option to display per-protocol statistics, proto may be any of: IP, IPv6, ICMP, ICMPv6, TCP, TCPv6, UDP, or UDPv6.

-r Displays the routing table.

-s Displays per-protocol statistics. By default, statistics are shown for IP, IPv6, ICMP, ICMPv6, TCP, TCPv6, UDP, and UDPv6; the **-p** option may be used to specify a subset of the default.

-v When used in conjunction with **-b**, will display sequence of components involved in creating the connection or listening port for all executables.

interval Redisplays selected statistics, pausing interval seconds between each display. Press CTRL+C to stop redisplaying statistics. If omitted, netstat will print the current configuration information once.

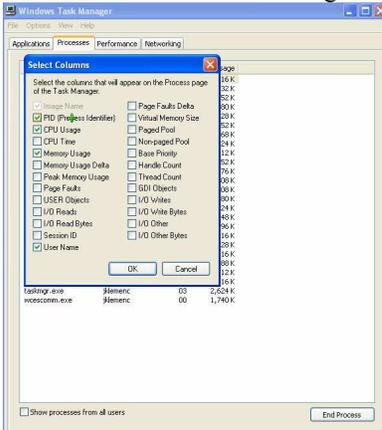
Next, we want to find all the open ports and Process ID's listening on the system by using the 'NETSTAT.EXE -ano' command:

```
C:\>netstat -ano
```

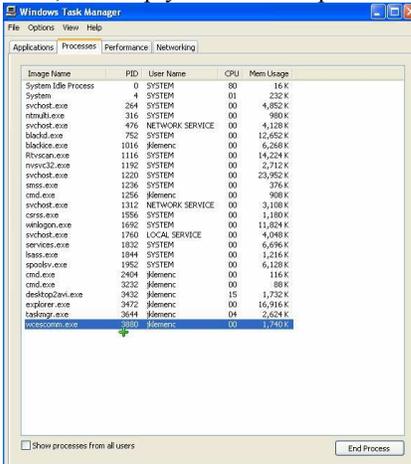
Active Connections

Proto	Local Address	Foreign Address	State	PID
TCP	0.0.0.0:135	0.0.0.0:0	LISTENING	476
TCP	0.0.0.0:445	0.0.0.0:0	LISTENING	4
TCP	0.0.0.0:3389	0.0.0.0:0	LISTENING	264
TCP	0.0.0.0:5679	0.0.0.0:0	LISTENING	3880
UDP	0.0.0.0:135	0.0.0.0:0	LISTENING	476
UDP	0.0.0.0:445	0.0.0.0:0	LISTENING	4

Port TCP/5679 looks suspicious, so lets investigate further. Since we have the Process ID from the output above, we can use the Task Manager to find the service which opened this port. First, we need to add the PID column to the Task Manager Processes list:



Next, we simply look for our process ID in the list:



Third party tools are also available to consolidate these steps into one simple process. There are many examples of such utilities on the Internet, but ActivePorts and FPORT are some of the more popular utilities:

C:\fport.exe
 FPort v2.0 - TCP/IP Process to Port Mapper
 Copyright 2000 by Foundstone, Inc.
<http://www.foundstone.com>

```
Pid Process Port Proto Path
476 -> 135 TCP
4 System -> 139 TCP
4 System -> 445 TCP
264 svchost -> 3389 TCP C:\WINDOWS\system32\svchost.exe
3880 wcescomm -> 5679 TCP C:\PDA\WCESCOMM.EXE
```