Linux Ports

Introduction

Linux systems are often used as server computers, or at least they are connected to the internet more or less directly. On such systems, network security is particularly important, because incorrectly configured servers can provide miscreants with a way into your system to do whatever damage they like. One of the first lines of defence against such problems is limiting access by port. In this context a port is a numbered access point for your computer, much like a telephone extension number in a business phone system. Ports are related to sockets, which are programming abstractions of network connection endpoints. Typically you wont deal with sockets per se as a system administrator, though. You can protect access by port in three main ways:

- By configuring a Firewall
- By using restrictions built into super servers
- By disabling servers you are not actively using

First, though, you must know a bit about ports.

Common server ports

Many firewalls and other network security devices operate by blocking or enabling access to specific ports. For instance, a firewall might block outside access to the SSH ports but let through traffic to the SMTP (Simple Mail Transfer Protocol) mail server port. In order to configure a firewall in this way, of course, you must know the port numbers. Linux systems contain a file, /etc/services, that lists service names and the ports with which the are associated. Lines in this file look something like this:

- **ssh 22/tcp # SSH Remote Login Protocol**
- **ssh 22/udp # SSH Remote Login Protocol**
- **telnet 23/tcp**
- **# 24 - private**
- **smtp 25/tcp**

The fisrt coloumn contains a service name (ssh, telnet, or smtp in this example). The second coloumn contains the port number and protocol (such as 22/tcp, meaning TCP port 22). Anything following a bash mark (#) is a comment and is ignored. The etc/services file lists port numbers for both TCP (Transmission Control Protocol) and UDP (User Datagram Protocol) ports. Typically, a single service is assigned use of the same TCP and UDP port numbers (as in the ssh service in this example), although in practice most protocols use just one or the other. When configuring a firewall, its generally bets to block both TCP and UDP ports; this ensures that you wont accidently block the wrong port type.

The following summarizes the port numbers used by the most important protocols run on Linux systems. This list is, however, incomplete; it only hit some of the most common protocols. In fact, even /etc/services is incomplete and may need to be expanded for certain obscure servers. (Their documentation describes how to, if necessary.)

Port Numbers Used by Some Common Protocols

Port Number TCP/UDP Protocol
20 & 21 TCP FTP
22 TCP SSH
23 TCP Telnet
25 TCP SMTP
53 TCP & UDP DNS
67 UDP DHCP
69 UDP TFTP
80 TCP HTTP
88 TCP Kerberos
109 & 110 TCP POPv2 & POPv3
111 TCP & UDP Port Mapper
113 TCP auth/ident
119 TCP NNTP
123 UDP NTP
137 UDP NetBIOS Name Service
138 UDP NetBIOS Datagram
139 TCP NetBIOS Session
143 TCP IMAP 2
161 UDP SNMP
177 UDP XDMCP
220 TCP IMAP 3
389 TCP LDAP
443 TCP HTTPS
445 TCP Microsoft DS
514 UDP Syslog
515 TCP Spooler
636 TCP LDAPS
749 TCP Kerberos Admin
5800-5899 TCP VNC via HTTP
5900 TCP VNC
6000-6099 TCP X (X.org-X11, XFree86)

One key distinction between TCP/IP ports is that between privileged ports and unprivileged ports. The former have numbers below 1024. Unix and Linux systems restrict access to to privileged ports to root. The idea is that a client can connect to a privileged port and and be confident that the server running on that port was configured by the system administrator and can then be trusted. Well, today we don't trust eachother very much on the internet so that distinction isn't vey useful. Know your ports, believe me it's useful!!

Source: http://www.go4expert.com/articles/linux-ports-t8841/