The Linux Networking Overview HOWTO

Table of Contents

Table of Contents

The Linux Networking Overview HOWTO

Daniel Lopez Ridruejo, `ridruejo@rawbyte.com`

v0.32, 8 July 2000

The purpose of this document is to give an overview of the networking capabilities of the Linux Operating System and to provide pointers for further information and implementation details.

---

7. Remote execution of applications

8. Network Interconnection

9. Network Management

10. Enterprise Linux Networking

11. Sources of Information

12. Document history

13. Acknowledgements and disclaimer

---

1. Introduction

The purpose of this document is to give an overview of the networking capabilities of the Linux operating system. Although one of the strengths of Linux is that plenty of information exists for nearly every component of it, most of this information is focused on implementation. New Linux users, particularly those coming from a Windows environment, are often unaware of the networking possibilities of Linux. This document aims to show a general picture of such possibilities with a brief description of each one and pointers for further information. The information has been gathered from many sources: HOWTOs, faqs, projects' web pages and my own hands−on experience. Full credit is given to the authors of these other sources. Without them and their programs this document would have not been possible or necessary.

---

2. Linux.

2.1 What is Linux?

The primary author of Linux is Linus Torvalds. Since his original versions, it has been improved by countless numbers of people. It is a clone, written entirely from scratch, of the Unix operating system. One of the more interesting facts about Linux is that its development occurs simultaneously around the world.

Linux has been copyrighted under the terms of the GNU General Public License (GPL). This is a license written by the Free Software Foundation (FSF) that is designed to prevent people from restricting the distribution of software. In brief, it says that although money can be charged for a copy, the person who received the copy can not be prevented from giving it away for free. It also means that the source code must be available. This is useful for programmers. Anybody can modify Linux and even distribute his/her modifications, provided that they keep the code under the same copyright.

2.2 What makes Linux different?

Why work on Linux? Linux is generally cheaper (or at least no more expensive) than other operating systems and is frequently less problematic than many commercial systems. But what makes Linux different is not its price (after all, why would anyone want an OS − even a free one − if it is not good enough?) but its outstanding capabilities:

- Linux is a true 32−bit multitasking operating system, robust and capable enough to be used in organizations ranging from universities to large corporations.
- It runs on hardware ranging from low−end 386 boxes to massive ultra−parallel machines in research centres.
- Out−of−the−box versions are available for Intel, Sparc, and Alpha architectures, and experimental support exists for Power PC and embedded systems, among others such as SGI, Ultra Sparc, AP1000+, Strong ARM, and MIPS R3000/R4000.
- Finally, when it comes to networking, Linux is choice. Not only because networking is tightly integrated with the OS itself and a plethora of applications is freely available, but for the robustness under heavy loads that can only be achieved after years of debugging and testing in an Open Source project.

---

3. Networking protocols

Linux supports many different networking protocols:

3.1 TCP/IP

The Internet Protocol was originally developed two decades ago for the United States Department of Defense (DoD), mainly for the purpose of interconnecting different−brand computers. The TCP/IP suite of protocols allowed, through its layered structure, to insulate applications from networking hardware.

Although it is based on a layered model, it is focused more on delivering interconnectivity than on rigidly adhering to functional layers. This is one of the reasons why TCP/IP has become the de facto standard internetworking protocol as opposed to OSI.

TCP/IP networking has been present in Linux since its beginnings. It has been implemented from scratch. It is one of the most robust, fast and reliable implementations and is one of the key factors of the success of Linux.

Related HOWTO: http://metalab.unc.edu/mdw/HOWTO/NET3−4−HOWTO.html

3.2 TCP/IP version 6

IPv6, sometimes also referred to as IPng (IP Next Generation) is an upgrade to the IPv4 protocol in order to address many issues. These issues include: shortage of available IP addresses, lack of mechanisms to handle time−sensitive traffic, lack of network layer security, etc.

The larger name space will be accompanied by an improved addressing scheme, which will have a great impact on routing performance. A beta implementation exists for Linux, and a production version is expected for the 2.2.0 Linux kernel release.

- Linux IPv6 HOWTO: http://www.wcug.wwu.edu/ipv6/faq/

3.3 IPX/SPX

IPX/SPX (Internet Packet Exchange/Sequenced Packet Exchange) is a proprietary protocol stack developed by Novell and based on the Xerox Network Systems (XNS) protocol. IPX/SPX became prominent during the early 1980s as an integral part of Novell, Inc.'s NetWare. NetWare became the de facto standard network operating system (NOS) of first generation LANs. Novell complemented its NOS with a business−oriented application suite and client−side connection utilities.

Linux has a very clean IPX/SPX implementation, allowing it to be configured as an:

- IPX router
- IPX bridge
- NCP client and/or NCP Server (for sharing files)
- Novell Print Client, Novell Print Server

And to:

- Enable PPP/IPX, allowing a Linux box to act as a PPP server/client
- Perform IPX tunnelling through IP, allowing the connection of two IPX networks through an IP only

link

Additionally, Caldera offers commercial support for Novell NetWare under Linux. Caldera provides a fully featured Novell NetWare client built on technology licensed from Novell Corporation. The client provides full client access to Novell 3.x and 4.x fileservers and includes features such as NetWare Directory Service (NDS) and RSA encryption.

- IPX HOWTO: http://metalab.unc.edu/mdw/HOWTO/IPX−HOWTO.html

3.4 AppleTalk Protocol Suite

Appletalk is the name of Apple's internetworking stack. It allows a peer−to−peer network model which provides basic functionality such as file and printer sharing. Each machine can simultaneously act as a client and a server, and the software and hardware necessary are included with every Apple computer.

Linux provides full Appletalk networking. Netatalk is a kernel−level implementation of the AppleTalk Protocol Suite, originally for BSD−derived systems. It includes support for routing AppleTalk, serving Unix and AFS filesystems over AFP (AppleShare), serving Unix printers and accessing AppleTalk printers over PAP.

See section 5.1 for more information.

3.5 WAN Networking: X.25, Frame−relay, etc...

Several third parties provide T−1, T−3, X.25 and Frame Relay products for Linux. Generally special hardware is required for these types of connections. Vendors that provide the hardware also provide the drivers with protocol support.

- WAN resources for Linux: http://www.secretagent.com/networking/wan.html

3.6 ISDN

The Linux kernel has built−in ISDN capabilies. Isdn4linux controls ISDN PC cards and can emulate a modem with the Hayes command set ("AT" commands). The possibilities range from simply using a terminal program to connections via HDLC (using included devices) to full connection to the Internet with PPP to audio applications.

- FAQ for isdn4linux: http://ww.isdn4linux.de/faq/

3.7 PPP, SLIP, PLIP

The Linux kernel has built−in support for PPP (Point−to−Point−Protocol), SLIP (Serial Line IP) and PLIP (Parallel Line IP). PPP is the most popular way individual users access their ISPs (Internet Service Providers). PLIP allows the cheap connection of two machines. It uses a parallel port and a special cable, achieving speeds of 10kBps to 20kBps.

- Linux PPP HOWTO
- PPP/SLIP emulator
- PLIP information can be found in The Network Administrator Guide

3.8 Amateur Radio

The Linux kernel has built−in support for amateur radio protocols.

Especially interesting is the AX.25 support. The AX.25 protocol offers both connected and connectionless modes of operation, and is used either by itself for point−point links, or to carry other protocols such as TCP/IP and NetRom.

It is similar to X.25 level 2 in structure, with some extensions to make it more useful in the amateur radio environment.

- Amateur radio on Linux web site

3.9 ATM

ATM support for Linux is currently in pre−alpha stage. There is an experimental release, which supports raw ATM connections (PVCs and SVCs), IP over ATM, LAN emulation...

- Linux ATM−Linux home page

---

4. Networking hardware supported

Linux supports a great variety of networking hardware, including some obsolete equipment.

Some interesting documents:

- Hardware HOWTO
- Ethernet HOWTO

---

5. File Sharing and Printing

The primary purpose of many PC based Local Area Networks is to provide file and printer sharing services to the users. Linux as a corporate file and print server turns out to be a great solution.

5.1 Apple environment

As outlined in previous sections, Linux supports the Appletalk family of protocols. Linux netatalk allows Macintosh clients to see Linux Systems as another Macintosh on the network, share files and use printers connected to Linux servers.

Netatalk faq and HOWTO:

- http://thehamptons.com/anders/netatalk/
- http://www.umich.edu/~rsug/netatalk/
- http://www.umich.edu/~rsug/netatalk/faq.html

5.2 Windows Environment

Samba is a suite of applications that allow most Unices (and in particular Linux) to integrate into a Microsoft network both as a client and a server. Acting as a server it allows Windows 95, Windows for Workgroups, DOS and Windows NT clients to access Linux files and printing services. It can completely replace Windows NT for file and printing services, including the automatic downloading of printer drivers to clients. Acting as a client allows the Linux workstation to mount locally exported windows file shares.

According to the SAMBA Meta−FAQ:

```
"Many users report that compared to other SMB implementations Samba is more stable,
faster, and compatible with more clients. Administrators of some large installations say
that Samba is the only SMB server available which will scale to many tens of thousands
of users without crashing"
```

- Samba project home page
- SMB HOWTO
- Printing HOWTO

5.3 Novell Environment

As stated in previous sections, Linux can be configured to act as an NCP client or server, thus allowing file and printing services over a Novell network for both Novell and Unix clients.

- IPX HOWTO

5.4 Unix Environment

The preferred way to share files in a Unix networking environment is through NFS. NFS stands for Network File Sharing and it is a protocol originally developed by Sun Microsystems. It is a way to share files between machines as if they were local. A client "mounts" a filesystem "exported" by an NFS server. The mounted filesystem will appear to the client machine as if it was part of the local filesystem.

It is possible to mount the root filesystem at startup time, thus allowing diskless clients to boot up and access all files from a server. In other words, it is possible to have a fully functional computer without a hard disk.

Coda is a network filesystem (like NFS) that supports disconnected operation, persistant caching, among other goodies. It's included in 2.2.x kernels. Really handy for slow or unreliable networks and laptops.

NFS−related documents:

- http://metalab.unc.edu/mdw/HOWTO/mini/NFS−Root.html
- http://metalab.unc.edu/mdw/HOWTO/Diskless−HOWTO.html
- http://metalab.unc.edu/mdw/HOWTO/mini/NFS−Root−Client−mini−HOWTO/index.html
- http://www.redhat.com/support/docs/rhl/NFS−Tips/NFS−Tips.html
- http://metalab.unc.edu/mdw/HOWTO/NFS−HOWTO.html

CODA can be found at:

6. Internet/Intranet

Linux is a great platform to act as an Intranet / Internet server. The term Intranet refers to the application of Internet technologies inside an organisation mainly for the purpose of distributing and making available information inside the company. Internet and Intranet services offered by Linux include mail, news, WWW servers and many more that will be outlined in the next sections.

6.1 Mail

Mail servers

Sendmail is the de facto standard mail server program (called an MTA, or Mail Transport Agent) for Unix platforms. It is robust, scalable, and properly configured and with the necessary hardware, can handle loads of thousands of users without blinking. Alternative mail servers, such as smail and qmail, are also available.

- Sendmail web site
- Smail faq
- Qmail web site

Mail HOWTOs:

- http://metalab.unc.edu/mdw/HOWTO/Mail−User−HOWTO.html
- http://metalab.unc.edu/mdw/HOWTO/mini/Qmail+MH.html
- http://metalab.unc.edu/mdw/HOWTO/mini/Sendmail+UUCP.html
- http://metalab.unc.edu/mdw/HOWTO/mini/Mail−Queue.html

Remote access to mail

In an organisation or ISP, users will likely access their mail remotely from their desktops. Several alternatives exist in Linux, including POP (Post Office Protocol) and IMAP (Internet Message Access Protocol) servers. The POP protocol is usually used to transfer messages from the server to the client. IMAP permits also manipulation of the messages in the server, remote creation and deletion of folders in the server, concurrent access to shared mail folders, etc.

- Brief comparison IMAP and POP

Mail related HOWTOs:

- http://metalab.unc.edu/mdw/HOWTO/Mail−User−HOWTO.html
- http://metalab.unc.edu/mdw/HOWTO/Cyrus−IMAP.html

Mail User Agents

There are a number of MUA (Mail User Agents) in Linux, both graphical and text mode. The most widely used ones include: pine, elm, mutt and Netscape.

- List of mail related software
- http://metalab.unc.edu/mdw/HOWTO/mini/TkRat.html

Mailing list software

There are many MLM (Mail List Management) programs available for Unix in general and for Linux in particular.

- A good comparison of existing MLMs may be found at: ftp://ftp.uu.net/usenet/news.answers/mail/list−admin/
- Listserv
- Majordomo home page

Fetchmail

One userful mail−related utility is fetchmail. Fetchmail is a free, full−featured, robust, well−documented remote−mail retrieval and forwarding utility intended to be used over on−demand TCP/IP links (such as SLIP or PPP connections). It supports every remote−mail protocol now in use on the Internet. It can even support IPv6 and IPSEC.

Fetchmail retrieves mail from remote mail servers and forwards it via SMTP, so it can then be be read by normal mail user agents such as mutt, elm or BSD Mail. It allows all the system MTA's filtering, forwarding, and aliasing facilities to work just as they would on normal mail.

Fetchmail can be used as a POP/IMAP−to−SMTP gateway for an entire DNS domain, collecting mail from a single drop box on an ISP and SMTP−forwarding it based on header addresses.

A small company may centralise its mail in a single mailbox, configure fetchmail to collect all outgoing mail, send it via a single mailbox at their ISP and retrieve all incoming mail from the same mailbox.

- Fetchmail home page

6.2 Web Servers

Most Linux distributions include Apache. Apache is the number one server on the internet according to http://www.netcraft.co.uk/survey/ . More than a half of all internet sites are running Apache or one of it derivatives. Apache's advantages include its modular design, stability and speed. Given the appropriate hardware and configuration it can support the highest loads: Yahoo, Altavista, GeoCities, and Hotmail are based on customized versions of this server.

Optional support for SSL (which enables secure transactions) is also available at:

- http://www.apache−ssl.org/
- http://raven.covalent.net/
- http://www.c2.net/

Related HOWTOs:

- http://metalab.unc.edu/mdw/HOWTO/WWW−HOWTO.html
- http://metalab.unc.edu/mdw/HOWTO/Virtual−Services−HOWTO.html
- http://metalab.unc.edu/mdw/HOWTO/Intranet−Server−HOWTO.html
- Web servers for Linux

6.3 Web Browsers

A number of web browsers exist for the Linux platform. Netscape Navigator has been one of the choices from the very beginning and the upcoming Mozilla (http://www.mozilla.org) will have a Linux version. Another popular text based web browser is lynx. It is fast and handy when no graphical environment is available.

* Browser software for Linux
* http://metalab.unc.edu/mdw/HOWTO/mini/Public−Web−Browser.html

6.4 FTP Servers and clients

FTP stands for File Transfer Protocol. An FTP server allows clients to connect to it and retrieve (download) files. Many ftp servers and clients exist for Linux and are included with most distributions. There are text−based clients as well as GUI based ones. FTP related software (servers and clients) for Linux may be found at: http://metalab.unc.edu/pub/Linux/system/network/file−transfer/

6.5 News service

Usenet (also known as news) is a big bulletin board system that covers all kinds of topics and it is organised hierarchically. A network of computers across the internet (Usenet) exchange articles through the NNTP protocol. Several implementations exist for Linux, either for heavily loaded sites or for small sites receiving only a few newsgroups.

* INN home page
* Linux news related software

6.6 Domain Name System

A DNS server has the job of translating names (readable by humans) to IP addresses. A DNS server does not know all the IP addresses in the world; rather, it is able to request other servers for the unknown addresses. The DNS server will either return the wanted IP address to the user or report that the name cannot be found in the tables.

Name serving on Unix (and on the vast majority of the Internet) is done by a program called named. This is a part of the bind package of The Internet Software Consortium.

* BIND
* DNS HOWTO

6.7 DHCP, bootp

DHCP and bootp are protocols that allow a client machine to obtain network information (such as their IP number) from a server. Many organisations are starting to use it because it eases network administration, especially in large networks or networks which have lots of mobile users.

Related documents:

* DHCP mini−HOWTO

6.8 NIS

The Network Information Service (NIS) provides a simple network lookup service consisting of databases and processes. Its purpose is to provide information that has to be known throughout the network to all machines on the network. For example, it enables an administrator to allow users access to any machine in a network running NIS without a password entry existing on each machine; only the main database needs to be maintained.

Related HOWTO:

- NIS HOWTO

6.9 Authentication

There are also various ways of authenticating users in mixed networks.

- For Linux/Windows NT: http://www.mindware.com.au/ftp/smb−NT−verify.1.1.tar.gz
- The PAM (pluggable authentication module) which is a flexible method of Unix authentication: PAM library.
- Finally, LDAP in Linux

---

7. Remote execution of applications

One of the most amazing features of Unix (yet one of the most unknown to new users) is its great support for remote and distributed execution of applications.

7.1 Telnet

Telnet is a program that allows a person to use a remote computer as if that person were actually at the remote site. Telnet is one of the most powerful tools for Unix, allowing for true remote administration. It is also an interesting program from the point of view of users, because it allows remote access to all their files and programs from anywhere in the Internet. Combined with an X server, there is no difference (apart from the delay) between being at the console or on the other side of the planet. Telnet daemons and clients are available with most Linux distributions.

Encrypted remote shell sessions are available through SSH (http://www.ssh.fi/sshprotocols2/index.html) thus effectively allowing secure remote administration.

- Telnet related software

7.2 Remote commands

In Unix, and in particular in Linux, remote commands exist that allow for interaction with other computers from the shell prompt. Examples are: rlogin, which allows for login in a remote machine in a similar way to telnet, rcp, which allows for the remote transfer of files among machines, etc. Finally, the remote shell command rsh allows the execution of a command on a remote machine without actually logging onto that machine.

7.3 The X Window System

The X Window System was developed at MIT in the late 1980s, rapidly becoming the industry standard windowing system for Unix graphics workstations. The software is freely available, very versatile, and is suitable for a wide range of hardware platforms. Any X environment consists of two distinct parts, the X server and one or more X clients. It is important to realise the distinction between the server and the client. The server controls the display directly and is responsible for all input/output via the keyboard, mouse or display. The clients, on the other hand, do not access the screen directly – they communicate with the server, which handles all input and output. It is the clients which do the "real" computing work – running applications or whatever. The clients communicate with the server, causing the server to open one or more windows to handle input and output for that client.

In short, the X Window System allows a user to log in into a remote machine, execute a process (for example, open a web browser) and have the output displayed on his own machine. Because the process is actually being executed on the remote system, very little CPU power is needed in the local one. Indeed, computers exist whose primary purpose is to act as pure X servers. Such systems are called X terminals.

A free port of the X Window System exists for Linux and can be found at: Xfree. It is included in most Linux distributions.

Related HOWTO:

- Remote X Apps HOWTO

7.4 VNC

VNC stands for Virtual Network Computing. It is, in essence, a remote display system which allows one to view a computing 'desktop' environment not only on the machine where it is running, but from anywhere on the Internet and from a wide variety of machine architectures. Both clients and servers exist for Linux as well as for many other platforms. It is possible to execute MS−Word in a Windows NT or 95 machine and have the output displayed in a Linux machine. The opposite is also true; it is possible to execute an application in a Linux machine and have the output displayed in any other Linux or Windows machine. One of the available clients is a Java applet, allowing the remote display to be run inside a web browser. Another client is a port for Linux using the SVGAlib graphics library, allowing 386s with as little as 4 MB of RAM to become fully functional X−Terminals.

- VNC web site

---

8. Network Interconnection

Linux networking is rich in features. A Linux box can be configured so it can act as a router, bridge, etc... Some of the available options are described below.

8.1 Router

The Linux kernel has built−in support for routing functions. A Linux box can act either as an IP or IPX router for a fraction of the cost of a commercial router. Recent kernels include special options for machines acting primarily as routers:

- Multicasting: Allows the Linux machine to act as a router for IP packets that have several destination addresses. It is needed on the MBONE, a high bandwidth network on top of the Internet which carries audio and video broadcasts.
- IP policy routing: Normally a router decides what to do with a received packet based solely on the packet's final destination address, but routing can also take into account the originating address and the network device from which the packet reached it.

There are some related projects which include one aiming at building a complete, running Linux router on a floppy disk: Linux router project

8.2 Bridge

The Linux kernel has built−in support for acting as an Ethernet bridge, which means that the different Ethernet segments it is connected to will appear as one Ethernet to the participants. Several bridges can work together to create even larger networks of Ethernets using the IEEE802.1 spanning tree algorithm. As this is a standard, Linux bridges will interoperate properly with other third party bridge products. Additional packages allow filtering based on IP, IPX or MAC addresses.

Related HOWTOs:

- Bridge+Firewall
- Bridge

8.3 IP Masquerade

IP Masquerade is a developing networking function in Linux. If a Linux host is connected to the Internet with IP Masquerade enabled, then computers connecting to it (either on the same LAN or connected with modems) can reach the Internet as well, even though they have no officially assigned IP addresses. This allows for reduction of costs, since many people may be able to access the Internet using a single modem connection as well as contributes to increased security (in some way the machine is acting as a firewall, since unofficially assigned addresses cannot be accessed outside of that network).

IP masquerade related pages and documents:

- http://ipmasq.home.ml.org/
- http://www.indyramp.com/masq/links.pfhtml
- http://metalab.unc.edu/mdw/HOWTO/IP−Masquerade−HOWTO.html

8.4 IP Accounting

This option of the Linux kernel keeps track of IP network traffic, performs packet logging and produces some statistics. A series of rules may be defined so when a packet matches a given pattern, some action is performed: a counter is increased, it is accepted/rejected, etc.

8.5 IP aliasing

This feature of the Linux kernel provides the possibility of setting multiple network addresses on the same low−level network device driver (e.g two IP addresses in one Ethernet card). It is typically used for services that act differently based on the address they listen on (e.g. "multihosting" or "virtual domains" or "virtual hosting services".

Related HOWTO:

- IP Aliasing HOWTO

8.6 Traffic Shaping

The traffic shaper is a virtual network device that makes it possible to limit the rate of outgoing data flow over another network device. This is especially useful in scenarios such as ISPs, where it is desirable to control and enforce policies regarding how much bandwidth is used by each client. Another alternative (for web services only) may be certain Apache modules which restrict the number of IP connections by client or the bandwidth used.

- http://metalab.unc.edu/mdw/HOWTO/NET3−4−HOWTO−6.html#ss6.15

8.7 Firewall

A firewall is a device that protects a private network from the public part (the internet as a whole). It is designed to control the flow of packets based on the source, destination, port and packet type information contained in each packet.

Different firewall toolkits exist for Linux as well as built−in support in the kernel. Other firewalls are TIS and SOCKS. These firewall toolkits are very complete and combined with other tools allow blocking/redirection of all kinds of traffic and protocols. Different policies can be implemented via configuration files or GUI programs.

- TIS home page
- SOCKS
- Firewall HOWTO

8.8 Port forwarding

An increasing number of web sites are becoming interactive by having cgi−bins or Java applets that access some database or other service. Since this access may pose a security problem, the machine containing the database should not be directly connected to the Internet.

Port Forwarding can provide an almost ideal solution to this access problem. On the firewall, IP packets that come in to a specific port number can be re−written and forwarded to the internal server providing the actual service. The reply packets from the internal server are re−written to make it appear that they came from the firewall.

Port forwarding information may be found here

8.9 Load Balancing

Demand for load balancing usually arises in database/web access when many clients make simultaneous requests to a server. It would be desirable to have multiple identical servers and redirect requests to the less loaded server. This can be achieved through Network Address Translation techniques (NAT) of which IP masquerading is a subset. Network administrators can replace a single server providing Web services − or any other application − with a logical pool of servers sharing a common IP address. Incoming connections are directed to a particular server using one load−balancing algorithm. The virtual server rewrites incoming and outgoing packets to give clients the appearance that only one server exists.

Linux IP−NAT information may be found here

8.10 EQL

EQL is integrated into the Linux kernel. If two serial connections exist to some other computer (this usually requires two modems and two telephone lines) and SLIP or PPP (protocols for sending Internet traffic over telephone lines) are used on them, it is possible to make them behave like one double speed connection using this driver. Naturally, this has to be supported at the other end as well.

- http://metalab.unc.edu/mdw/HOWTO/NET3−4−HOWTO−6.html#ss6.2

8.11 Proxy Server

The term proxy means "to do something on behalf of someone else." In networking terms, a proxy server computer can act on the behalf of several clients. An HTTP proxy is a machine that receives requests for web pages from another machine (Machine A). The proxy gets the page requested and returns the result to Machine A. The proxy may have a cache with the requested pages, so if another machine asks for the same page the copy in the cache will be returned instead. This allows efficient use of bandwidth resources and less response time. As a side effect, as client machines are not directly connected to the outside world this is a way of securing the internal network. A well−configured proxy can be as effective as a good firewall.

Several proxy servers exist for Linux. One popular solution is the Apache proxy module. A more complete and robust implementation of an HTTP proxy is SQUID.

- Apache
- Squid

8.12 Diald on demand

The purpose of dial on demand is to make it transparently appear that the users have a permanent connection to a remote site. Usually, there is a daemon who monitors the traffic of packets and where an interesting packet (interesting is defined usually by a set of rules/priorities/permissions) arrives it establishes a connection with the remote end. When the channel is idle for a certain period of time, it drops the connection.

- Diald HOWTO

8.13 Tunnelling, mobile IP and virtual private networks

The Linux kernel allows the tunnelling (encapsulation) of protocols. It can do IPX tunnelling through IP, allowing the connection of two IPX networks through an IP only link. It can also do IP−IP tunnelling, which it is essential for mobile IP support, multicast support and amateur radio. (see http://metalab.unc.edu/mdw/HOWTO/NET3−4−HOWTO−6.html#ss6.8)

Mobile IP specifies enhancements that allow transparent routing of IP datagrams to mobile nodes in the

Internet. Each mobile node is always identified by its home address, regardless of its current point of attachment to the Internet. While situated away from its home, a mobile node is also associated with a care−of address, which provides information about its current point of attachment to the Internet. The protocol provides for registering the care−of address with a home agent. The home agent sends datagrams destined for the mobile node through a tunnel to the care−of address. After arriving at the end of the tunnel, each datagram is then delivered to the mobile node.

Point−to−Point Tunneling Protocol (PPTP) is a networking technology that allows the use of the Internet as a secure virtual private network (VPN). PPTP is integrated with the Remote Access Services (RAS) server which is built into Windows NT Server. With PPTP, users can dial into a local ISP, or connect directly to the Internet, and access their network as if they were at their desks. PPTP is a closed protocol and its security has recently being compromised. It is highly recomendable to use other Linux based alternatives, since they rely on open standards which have been carefully examined and tested.

- A client implementation of the PPTP for Linux is available here
- More on Linux PPTP can be found here

Mobile IP:

- http://www.hpl.hp.com/personal/Jean_Tourrilhes/MobileIP/mip.html
- http://metalab.unc.edu/mdw/HOWTO/NET3−4−HOWTO−6.html#ss6.12

Virtual Private Networks related documents:

- http://metalab.unc.edu/mdw/HOWTO/mini/VPN.html
- http://sites.inka.de/sites/bigred/devel/cipe.html

---

9. Network Management

9.1 Network management applications

There is an impressive number of tools focused on network management and remote administration. Some interesting remote administration projects are linuxconf and webmin:

- Webmin
- Linuxconf

Other tools include network traffic analysis tools, network security tools, monitoring tools, configuration tools, etc. An archive of many of these tools may be found at Metalab

9.2 SNMP

The Simple Network Management Protocol is a protocol for Internet network management services. It allows for remote monitoring and configuration of routers, bridges, network cards, switches, etc... There is a large amount of libraries, clients, daemons and SNMP based monitoring programs available for Linux. A good page dealing with SNMP and Linux software may be found at : http://linas.org/linux/NMS.html

---

10. Enterprise Linux Networking

In certain situations it is necessary for the networking infrastructure to have proper mechanisms to guarantee network availability nearly 100% of the time. Some related techniques are described in the following sections. Most of the following material can be found at the excellent Linas website: http://linas.org/linux/index.html and in the Linux High−Availability HOWTO

10.1 High Availability

Redundancy is used to prevent the overall IT system from having single points of failure. A server with only one network card or a single SCSI disk has two single points of failure. The objective is to mask unplanned outages from users in a manner that lets users continue to work quickly. High availability software is a set of scripts and tools that automatically monitor and detect failures, taking the appropriate steps to restore normal operation and to notifying system administrators.

10.2 RAID

RAID, short for Redundant Array of Inexpensive Disks, is a method whereby information is spread across several disks, using techniques such as disk striping (RAID Level 0) and disk mirroring (RAID level 1) to achieve redundancy, lower latency and/or higher bandwidth for reading and/or writing, and recoverability from hard−disk crashes. Over six different types of RAID configurations have been defined. There are three types of RAID solution options available to Linux users: software RAID, outboard DASD boxes, and RAID disk controllers.

- Software RAID: Pure software RAID implements the various RAID levels in the kernel disk (block device) code.
- Outboard DASD Solutions: DASD (Direct Access Storage Device) are separate boxes that come with their own power supply, provide a cabinet/chassis for holding the hard drives, and appear to Linux as just another SCSI device. In many ways, these offer the most robust RAID solution.
- RAID Disk Controllers: Disk Controllers are adapter cards that plug into the ISA/EISA/PCI bus. Just like regular disk controller cards, a cable attaches them to the disk drives. Unlike regular disk controllers, the RAID controllers will implement RAID on the card itself, performing all necessary operations to provide various RAID levels.

Related HOWTOs:

- http://metalab.unc.edu/mdw/HOWTO/mini/DPT−Hardware−RAID.html
- http://metalab.unc.edu/mdw/HOWTO/Root−RAID−HOWTO.html
- http://metalab.unc.edu/mdw/HOWTO/Software−RAID−HOWTO.html

RAID at linas.org:

- http://linas.org/linux/raid.html

10.3 Redundant networking

IP Address Takeover (IPAT). When a network adapter card fails, its IP address should be taken by a working network card in the same node or in another node. MAC Address Takeover: when an IP takeover occurs, it should be made sure that all the nodes in the network update their ARP caches (the mapping between IP and MAC addresses).

See the High−Availability HOWTO for more details:

http://metalab.unc.edu/pub/Linux/ALPHA/linux−ha/High−Availability−HOWTO.html

11. Sources of Information

If you have networking problems with Linux, please do not e−mail the questions to me. I just simply do not have the time to answer them. You have better chances to obtain help if you post a question in the comp.os.linux.networking newsgroup (which you can access through http://www.dejanews.com). Before posting there, make sure that you have read the relevant documentation. Then search the news archive, because chances are that somebody, sometime made the same question (and somebody answered). When posting, remember to explain all the steps you have followed and the error messages you got. Where to get further information:

- Linux: http://www.linux.org
- Linux Documentation Project: http://metalab.unc.edu/mdw/linux.html (check out the Linux Network Administrator Guide)
- Freshmeat: The latest releases of Linux Software. http://www.freshmeat.net
- Linux links: http://www.linuxlinks.com/Networking/

12. Document history

- 0.32 Updated many links that have changed. Special thanks go here to Kontiki for his careful review and detailed description of what needed to change. Many thanks also to Anne and Mathias who pointed out other links that were no longer valid.
- 0.31 (17 Sept 1999) Changed address for linux router project (thanks to John Ellis) and added another PPTP link (thanks to Benjamin Smith)
- 0.30 (6 April 1999) Included section on CODA (thanks to Brian Ristuccia
- 0.2−0.29 Bugfixes :−) (see acknowledgements, at the end of this document)
- 0.1 (5 june 1998)

13. Acknowledgements and disclaimer

This document is based on the work of many other people who have made it possible for Linux to be what it is now: one of the best network operating systems. All credit is theirs. A lot of effort has been put into this document to make it simple but accurate and complete but not excessively long. Nevertheless, no liability will be assumed by the author under any circumstance. Use the information contained here at your own risk. Please feel free to e−mail me suggestions, corrections or general comments about the document so I can improve it. Other topics that will probably be included in futures revisions of this document may include radius, web/ftp mirroring tools such as wget, traffic analyzers, CORBA... and many others that may be suggested and suitable. You can reach me at daniel@rawbyte.com.

Finally I would like to thank Finnbjorn av Teigum, Cesar Kant, Mathieu Arnold and specially Hisakuni Nogami and Phil Garcia for their careful reviews and comments on this HOWTO. Their help is greatly appreciated.

You can find a version of this document at http://www.rawbyte.com/lno/.