A Heterogeneous Internetworking Model with Enhanced Management and Security Functions

Youlu Zheng
Computer Science Department
University of Montana

Yan Zhu
Sybase, Inc.

To demonstrate how different network components are integrated into a heterogeneous network, and how different network devices, protocols and software, such as routers, gateways, Simple Network Management Protocol (SNMP), etc. work, a limited number of computers are internetworked as three subnetworks. A software router and a gateway are installed to interconnect the three LANs.

1. Topology, Protocols, and Hardware Configuration

When designing a LAN, before connecting all computers and related resources together, it must first be determined how they are connected, what kind of network protocols and network operating system are used to control the network, and which networking services to be offered. The model network runs both TCP/IP and NetWare on its subnetworks.

NetWare is an easy choice for a typical client-server model. Novell's NetWare running IPX protocol is one of the most widely used local area network (LAN) software in the business world. It uses client-server architecture based on PC and Unix clients requesting services from a NetWare file/print server. NetWare is mature, robust and reasonably priced for the IBM PC compatible environment. It can also be conveniently connected to other popular internetworking protocols. For peer-to-peer connection, TCP/IP and related protocols that internetwork numerous remotely located LANs are the base of the real world information super highway. In addition, IPX and TCP/IP are the two major and most representative network protocols in the computer and network industry. Using these two protocols makes the network heterogeneous and creates a crucial problem of communicating in that environment, thereby presents the challenge of resolving communications conflicts.

The prototype heterogeneous network consists of three subnetworks: one includes two 486PC, another includes a Sun 3/110C workstation, a 386PC and a 486PC, and the third contains several 386 and 28C PCs.

A network's physical layer defines the physical link between computers and networks. This is primarily the network interface card (NIC) required in each connected computer and the cables needed to interconnect the NICs. The computer may then function as a file server, workstation, or gateway to a network or other communication device. In this network, four types of Ethernet NICs are used:

- A S-bus Ethernet card for the Sun 3-110 workstation.
- An SMC Elite 16 Ethernet card for a 486 running BSD4.3Net2 UNIX.
- NE2000 network cards for a 486 PC running LINUX and 286 PCs running DOS.
- Two NE2000 compatible cards on a 386 PC for the NetWare server/gateway.

Coaxial cable is mostly used in bus topology networks. A computer or other network device can be attached to the bus cable through a T-connector at the back of its NIC. One terminator is attached to each end

of the bus. This structure presents the real world annoying problems of locating trouble on the bus. Due to resource availability, coaxial cable is used for this model network. A better choice is 10 base-T cable using unshielded twisted pair (UTP) wiring and a simple hub. The network configuration (Figure 1) is as follows:

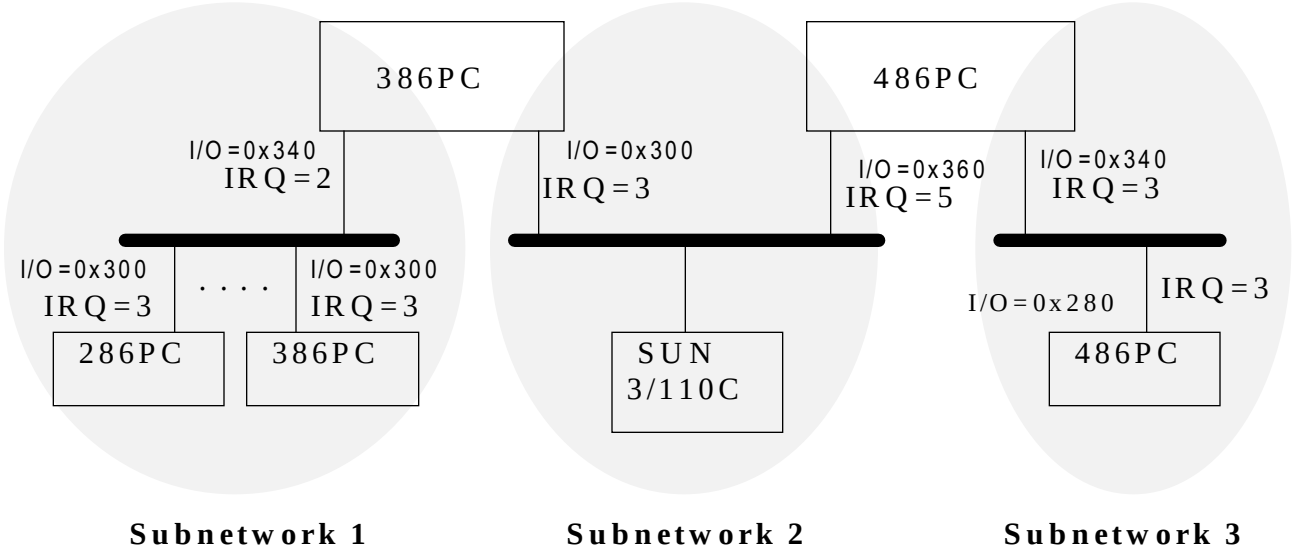

Figure 1: The hardware configuration of a heterogeneous network with three subnetworks.

Subnetwork 1 is a client/server NetWare network with a 386PC as the server. Subnetwork 2 includes three computers, and Subnetwork 3 includes only two computers. A gateway and router connect the three subnetworks. The Sun workstation is a single node. The network card comes with the workstation which uses the default I/O base address and the default interrupt. One 386PC installed with NetWare 3.12 works as the NetWare Server as well as a gateway between subnetwork 1 and subnetwork 2. It uses two NE2000 compatible cards. The first card uses I/O base address 0X340 and IRQ 2. The second NIC uses I/O base address 0X300 and IRQ 3. Several 386 and 286PC work as NetWare's clients. They use NE2000 cards with the default configuration: I/O base address 0X300 and IRQ 3.

Since IP addresses are long, they may be difficult to remember. Almost every system has a text name—host name—associated with each IP address. The host names are defined in the file `/etc/hosts`. In the model network, *Darkstar, Linux, Bsd, Fusion, Netware,* and *Pcware* are the host names for the nodes. The IP addresses along with the associated names of the nodes in the network are listed in Figure 2.

Linux is a 486 PC working as a router between subnetwork 2 and subnetwork 3. It has two NE2000 cards installed. The configuration of one card uses 0X340 I/O base address and interrupt IRQ 3. The other uses 0X360 and IRQ 5 for I/O base address and interrupt. The BSD computer is a 486PC that works as a node in Subnetwork 3. The SMC network card is installed with an I/O base address 0X280 and interrupt IRQ 3.

2. Software and Internet Services on the Heterogeneous Network

The heterogeneous network consists of two 486PC running LINUX and BSD4.3Net2 UNIX respectively, a Sun 3/110C workstation running SunOS4.1 (UNIX), a 386PC running NetWare 3.12, and several 286PC and 386PC running MSDOS or MSWindows as NetWare clients. LINUX proves to be a very valuable software package providing a complete set of utilities, an advanced graphical user interface, and complete network support. It is a true multiuser multitasking operating system. The complete operating system including the source code is free. In fact, all the BSD4.3Net2 and SunOS (UNIX) operating systems along with the Sun workstation can be replaced by 386/486 PCs and the LINUX operating system.

Subnetwork 1 is a client-server LAN, running NetWare 3.12 with both IPX protocol and TCP/IP protocol. NetWare's native IPX communication protocol is used to connect clients to the NetWare server. TCP/IP protocol on the NetWare server connects Subnetwork 1 to the other two LANs. Client applications, such as rconsole, login etc., are installed on the PCs running MSWindows and MSDOS.

In a UNIX environment, `ifconfig`, either explicitly or through some `.rc` initialization file, is used to assign the IP address, mask and broadcasting address to an interface card. In NetWare, the `bind` command is used to assign the IP address. Note that for a router and a gateway such as the Linux computer and the NetWare server in the model network, the computer has more than one IP address.

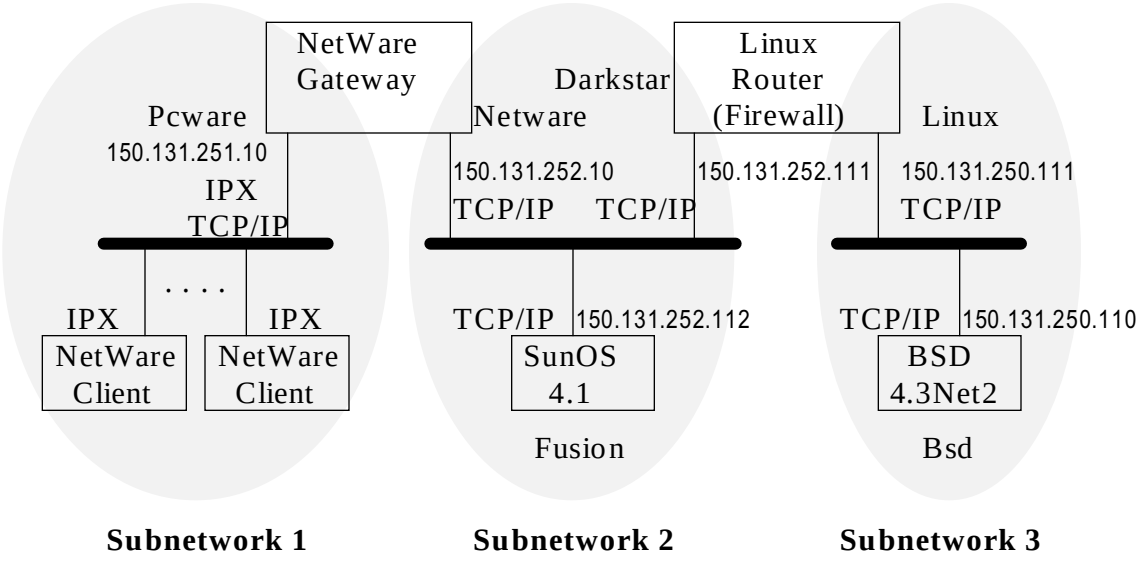

Figure 2: The network software configuration and IP addresses.

3. Software-based Routing in the Heterogeneous Network Environment

Both gateways and routers aim to provide all interconnections among physical networks. They pass packets between different sections of the network. Operating at the IP level, they are capable of making routing decisions based on information in routing tables. There are dedicated hardware routers with very high performance as well as very high prices. However a host computer with appropriate software can also perform routing functions.

For software based routers, routing is the process of directing a packet through the maze of networks that links the source to the destination. Routing information is stored in a table in the kernel. Each table entry has several parameters, including a reliability field that determines routes when the table contains conflicting information. To route a packet to a particular address, the kernel picks the most specific of the available rules. If there is no relevant route and no default, a "network unreachable'' error is returned to the sender.

Packets come in on one interface card and are either delivered locally or compared to the routing table to determine to where they should be forwarded. IP routing is determined by searching the routing table and deciding which interface card is used to send the packet out.

The maintenance of routing tables can be done statically, dynamically or with a combination of the two. A static route is entered explicitly using the command `route`. Static routes should stay in the routing table forever. This method is inflexible in respect to changes in the network configuration. Dynamic routing is performed once every 30 seconds by a daemon process that maintains and modifies the routing table. Routing daemons on different hosts communicate to discover the topology of the network and to determine the path to

distant destinations. In a UNIX environment, two daemon processes maintain and update routing information. They are `routed` and `gated,` based on the well known Routing Information Protocol (RIP) and Open Shortest Path First Protocol (OSPF). In addition, ICMP redirection can add a route entry into the routing table.

In the model network, `routed` runs on the SunOS, BSD4.3Net2 and the NetWare operating systems by setting the `routed` parameter to the "on" position. This allows `routed` to dynamically manage the routing table as well as establish static entries when machines are started. In the Linux computer, a static routing table is used and managed manually.

4. Internet Services and Network Management

The most important traditional TCP/IP services are supported by the appropriate protocols. They are:

- File Transfer Protocol (FTP).
- Network Terminal Protocol (TELNET).
- Simple Mail Transfer Protocol (SMTP).
- Simple Network Management Protocol (SNMP) and SNMPv2.

To use FTP, TCP port 21 is fixed as the command channel and TCP port 20 as the data channel. In a UNIX system, the protocol consists of `ftp` and the server daemon process `ftpd.` FTP is available in the model network for each node running UNIX software to transfer files between each other.

The TELNET service is attached to TCP port 23. UNIX processors currently incorporate the command rlogin, which offers almost the same functionality as TELNET but provides better support for the UNIX environment. Both TELNET and RLOGIN run smoothly on the model network. FTP and TELNET are configured in the file `/etc/services` and `/etc/passwd`.

In a UNIX system, SMTP (TCP port 25) is implemented by the program `sendmail`. It is possible for sendmail to function not only as an SMTP server, but also as an SMTP client. However, users rarely use sendmail directly, but instead, use *pine* or *mail*, which control and simplify the processing of a message. Sendmail is only activated to forward the message.

A configuration file, `sendmail.cf` is used to control sendmail, which makes the program very adaptable. In addition to the definition of the local-mail-forwarding program and many other uses, this file contains the commands for converting the address of the connected mail system. File aliases, which may be used to create distribution lists and for forwarding requests, are edited and then converted into an indexed database using the command `newaliases`. The prototype network provides complete email services. The customized `sendmail.cf` file for the model network is provided in the attached CD.

For network management, the Simple Network Management Protocol (SNMP) on the NetWare server is automatically loaded when `TCP.NLM` is started.

5. Software Packet-Filter Firewall

The LINUX operating system supports the TCP/IP communication protocol and also allows a computer running LINUX to be configured as a router, thus it is feasible to implement a router-based packet filter as a firewall. To take advantage of the LINUX routing software, and to make the implementation more structured and easy to use, the packet filter is separated into two parts: One is the modification of the LINUX IP software, which manages the packet routing and controls the packets' forwarding based on a set of rules; The other is a utility that allows users to specify the packet routing rules, that determine how every arriving packet is processed.

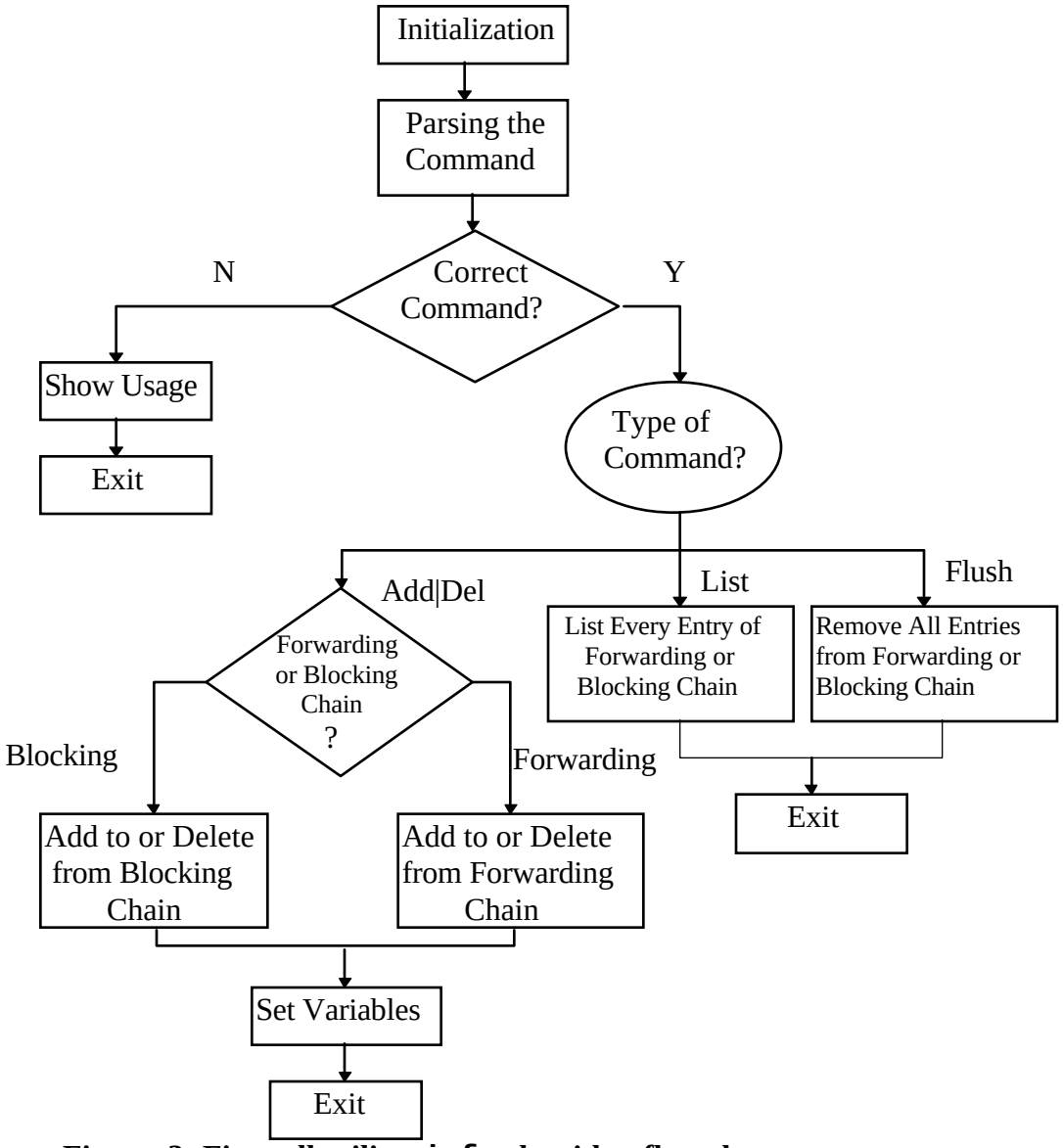

Figure 3: Firewall utility `ipfw` algorithm flow chart.

Modifications are made to the existing LINUX routing software. New functions are added at the IP level to examine arriving packets before they are routed. There are other functions and utilities that deal with packet receiving, sending, forwarding, header-building and sum-checking. They are the only interfaces between the IP layer and all other layers. The major new functions `ip_fw_chk` and `ip_fw_ctl` along with supporting functions `port_match`, `free_fw_chain`, `add_to_chain`, and `del_from_chain` are added to these functions for extra checking and deciding whether a packet should be discarded or forwarded.

There are three basic steps to setting up a packet filter:

1. Decide the security policy;
2. Express the policy in a format that the computer can understand;
3. Make the policy work.

An independent software package `ipfw` is developed as a user interface to the packet-filter firewall software. It allows users to specify what they want to do with each and every packet (datagram) coming to the router. `ipfw` maintains two chains—*forwarding* and *blocking*. The two chains are used to tell the firewall when to block and when to forward a packet. *Add, delete* and *list* are the three used for blocking or forwarding. After `ipfw` parses a command, it sets the corresponding socket options and variables.

Figure 3 is a flow chart of how `ipfw` works.

6. Conclusion

This project was supported, in part, by the National Science Foundation through the *Course Curriculum Development (CCD)* program. The lab engages the student actively in the learning process and provide opportunities for critical thinking, problem solving, and creativity in a controlled real-world environment.

Considering the fact that in a service-oriented network environment, all network management and security functions require supervisor privilege, the cost-effective prototype network introduced in this paper not only demonstrates how different network protocols and components are integrated into a heterogeneous network, it also provides an ideal experimental environment for network management and security problem investigation.

Many undergraduate and graduate students in the Department of Computer Science, University of Montana have worked in and used the lab in various networking projects. The authors are indebted for the support from the National Science Foundation.

The project and lab material are part of the textbook *Networks for Computer Scientists and Software Engineers* to be published in the near future.

References:

1. Comer, Douglas E. *Internetworking with TCP/IP, Vol I, II, III*, Prentice Hall, 1992 - 1995.

2. Cheswick, William, R. et al. Firewalls and Internet Security Repelling the Wily Hacker, Addison-Wesley, 1994

About the Authors:

YOULU ZHENG is a tenured professor in the Department of Computer Science, University of Montana. He is also consultant and adviser for several computer and network related companies. His email address is zheng@cs.umt.edu.

YAN ZHU is currently a software engineer in the Global Products Group, Sybase, Inc. She can be reached via email at yanz@sybase.com.