CHAPMAN & HALL/CRC COMPUTER and INFORMATION SCIENCE SERIES

Series Editor: Sartaj Sahni

PUBLISHED TITLES

ADVERSARIAL REASONING: COMPUTATIONAL APPROACHES TO READING THE OPPONENT'S MIND
Alexander Kott and William M. McEneaney

DISTRIBUTED SENSOR NETWORKS
S. Sitharama Iyengar and Richard R. Brooks

DISTRIBUTED SYSTEMS: AN ALGORITHMIC APPROACH
Sukumar Ghosh

FUNDEMENTALS OF NATURAL COMPUTING: BASIC CONCEPTS, ALGORITHMS, AND APPLICATIONS
Leandro Nunes de Castro

HANDBOOK OF ALGORITHMS FOR WIRELESS NETWORKING AND MOBILE COMPUTING
Azzedine Boukerche

HANDBOOK OF APPROXIMATION ALGORITHMS AND METAHEURISTICS
Teofilo F. Gonzalez

HANDBOOK OF BIOINSPIRED ALGORITHMS AND APPLICATIONS
Stephan Olariu and Albert Y. Zomaya

HANDBOOK OF COMPUTATIONAL MOLECULAR BIOLOGY
Srinivas Aluru

HANDBOOK OF DATA STRUCTURES AND APPLICATIONS
Dinesh P. Mehta and Sartaj Sahni

HANDBOOK OF DYNAMIC SYSTEM MODELING
Paul A. Fishwick

HANDBOOK OF PARALLEL COMPUTING: MODELS, ALGORITHMS AND APPLICATIONS
Sanguthevar Rajasekaran and John Reif

HANDBOOK OF REAL-TIME AND EMBEDDED SYSTEMS
Insup Lee, Joseph Y-T. Leung, and Sang H. Son

HANDBOOK OF SCHEDULING: ALGORITHMS, MODELS, AND PERFORMANCE ANALYSIS
Joseph Y.-T. Leung

HIGH PERFORMANCE COMPUTING IN REMOTE SENSING
Antonio J. Plaza and Chein-I Chang

INTRODUCTION TO NETWORK SECURITY
Douglas Jacobson

PERFORMANCE ANALYSIS OF QUEUING AND COMPUTER NETWORKS
G. R. Dattatreya

THE PRACTICAL HANDBOOK OF INTERNET COMPUTING
Munindar P. Singh

SCALABLE AND SECURE INTERNET SERVICES AND ARCHITECTURE
Cheng-Zhong Xu

SPECULATIVE EXECUTION IN HIGH PERFORMANCE COMPUTER ARCHITECTURES
David Kaeli and Pen-Chung Yew

Introduction to Network Security

Douglas Jacobson

Iowa State University
Ames, Iowa, U.S.A.

CRC Press
Taylor & Francis Group
Boca Raton London New York

CRC Press is an imprint of the
Taylor & Francis Group, an **informa** business
A CHAPMAN & HALL BOOK

CRC Press
Taylor & Francis Group
6000 Broken Sound Parkway NW, Suite 300
Boca Raton, FL 33487-2742

Version Date: 20131121

International Standard Book Number-13: 978-1-4200-8587-7 (eBook - PDF)

Visit the Taylor & Francis Web site at
http://www.taylorandfrancis.com

and the CRC Press Web site at
http://www.crcpress.com

Contents

Preface

Approach

This book focuses on network security from the viewpoint of a network's vulnerabilities, protocols, and security solutions. Unlike other books that focus on security and security paradigms where networks are viewed as a mechanism for communication, this book focuses on the network as a source of both insecurity and security. The book will examine various network protocols looking at vulnerabilities, exploits, attacks, and methods to mitigate an attack.

Networks as communication systems have been around since the dawn of human history and rely on trust between communicating parties in order to function. Early communications systems relied on visual verification of the communicating parties involved and often used simple codes to protect the data. For example, couriers were known by both parties and messages were sealed with wax to help ensure privacy. As technology improved, methods used to transmit data also improved, and so did the methods to steal and protect data. However, even as late as the end of the twentieth century, data was still being transmitted directly between two parties with no concept of a network. These parties relied on additional knowledge to verify the authenticity of the data. The issues we face today are more complex than those of the past. Today we have interconnected computers using a network not controlled by any one entity or organization. Unlike data communications of the past, today's networks consist of numerous devices that handle the data as it passes from the sender to the receiver. These networks are designed to facilitate communication and are intended for a small group of trusted and knowledgeable individuals. Security is not part of the design process.

Organization

Part I of this book is a brief discussion of network architectures and the functions of layers in a typical network, along with a taxonomy of network-based vulnerabilities and attacks. This taxonomy is the framework for presenting the vulnerabilities and attacks at each layer of interest. The taxonomy divides the vulnerabilities and attack space into four categories:

Header-based vulnerabilities and attacks: The protocol headers have been modified or are not valid.
Protocol-based vulnerabilities and attacks: The packets are valid but are not used correctly.
Authentication-based vulnerabilities and attacks: The identity of the sender or receiver is modified.
Traffic-based vulnerabilities and attacks: The volume of traffic creates the attack.

The remainder of the book is divided into three parts. Part II covers the different layers of the network (physical, network, and transport), looking at the security for each. Using a bottom-up approach to network security allows the reader to understand the vulnerabilities and the security mechanisms provided by each layer of the network. For example, by understanding which vulnerabilities are introduced by the physical layer and what level of security can be provided, the reader can understand which vulnerabilities may exist in the network layer and which security mechanisms could be used to overcome the vulnerabilities. Part III looks at the security of several common network applications. On the Internet, applications treat the lower layers of the network as a simple pipe that sends data to another application, and it arrives without error. This book views vulnerabilities as network functions provided by the layer below, thus giving the reader insight into understanding the security needed to overcome the vulnerabilities. Part IV provides an overview of several network-based security solutions that are often deployed and relates them back to the taxonomy.

This book describes a define–attack–defend methodology for network security. The relevant protocols are briefly introduced, followed by detailed descriptions of known vulnerabilities and possible attack methods. The book then focuses on the attack methodology rather than on particular tools, though tools are introduced as possible homework problems and lab experiments. Once the reader understands the threats against the protocol, possible solutions will be presented. Each chapter has homework problems that are based on the concepts introduced in the chapter and will have lab experiments that will allow the reader to try some of the attacks and look at the effectiveness of the solutions. An appendix provides details to develop and deploy a low-cost lab environment that can be used to support the classroom or used as a small corporate test bed. Another appendix provides an overview to cryptology.

Target Audience

This book is targeted at two compatible audiences. The primary focus of the book is as a text for a senior or first-year graduate course in network security for students in computer science or computer engineering. The book can be used for a network security course that is part of a security curriculum or for a course that is part of a networking curriculum. The book is also intended as a reference for network and security professionals.

Differences between this book and other books include:

Network focused: This book looks at network security by exploring network protocols, their weaknesses, and countermeasures. Several books also have a network focus but primarily deal with a few application-level protocols (Kerberos, secure email, secure web, etc.) and are not concerned about the lower layers (physical, network, transport). Many of the difficult problems arise from the vulnerabilities in these layers.

Network view of security: This book looks at network security using the approaches found in most network books, by looking at the layers and what services and functions are provided. We will look at vulnerabilities and security as services and functions provided by the layer. By using a network view, the book could be used in either a networking curriculum to add security or in a security curriculum to add network security.

Lab experiments: This book contains lab experiments to support the material. The experiments will look at both attacks and defenses. The book also provides a low-cost lab configuration that can be used as a model.

Web site: A web site is provided to support the book (http://www.dougj.net/textbook/). The web site contains lecture materials, tutorials on UNIX, C, and socket programming, and detailed information to establish and maintain the test laboratory.

Practical view of network security: This book has a practical view of network security. We will look at actual protocols and provide readers with the details and information they need to understand the vulnerabilities and to develop appropriate countermeasures. This is reinforced through the lab experiments.

Attack-and-defend approach: This book looks at network security from an attack-and-defend approach. The book looks at the vulnerabilities in the current protocols and then looks at defense systems that could mitigate the attacks. While the book will not focus on attack tools, it will look at attack methods, and through the lab experiments, students will be able to study the effects of certain attacks on the network and the effectiveness of the security system.

Terms defined: So much of networking and security involves the use of terms, many of which are specific to the field. Thus, I feel that it is important after each section of a chapter to enumerate with a short definition any new terms that were defined in that section. Before we begin the text, there are a few terms that should be defined so readers have a common frame of reference.

Definitions

Application.
A computer program that allows a user to connect to the network and perform a task.

Attacker.
A person or persons that use the network to attack computer systems, networks, or other devices connected to the Internet.

Hacker.
Same as an attacker.

Host.
A term used to describe a computer connected to the Internet.

Internet.
A global collection of networks of interconnected network devices.

Network.
A group of interconnected devices that can communicate with each other.

Network device.
A device connected to the network. This is more generic than a host or computer in that it can be any network-enabled device.

Target.
The device, host, user, or object that the hacker is trying to attack.
User.
The individual using a computer application that utilizes the network, or a general computer user.

Acknowledgments

I thank my wife, Gwenna, and my children (Sarah, Jordan, and Jessica) for all of their support and patience. I also thank Sharon Sparks for her editing help.

The Author

Doug Jacobson is a university professor in the Department of Electrical and Computer Engineering at Iowa State University. He is currently director of the Iowa State University Information Assurance Center, which has been recognized by the National Security Agency as a charter Center of Academic Excellence for Information Assurance Education. Dr. Jacobson teaches network security and information warfare. He also works with local law enforcement and is a computer forensics analyst for the Iowa State University Police Department. Dr. Jacobson is the founder of Palisade Systems, Inc., an Ames-based company marketing Internet management and security devices. He has received two R&D 100 awards for his security technology and has two patents in the area of computer security.

Part I

Introduction to Network Concepts and Threats

This part provides an introduction to basic network concepts and the taxonomy for network-based vulnerabilities and attacks. Readers that have studied networking could skip the first three chapters of this part. Chapter 1 discusses the concepts behind the layered approach to networking and how the common network architecture provides insight into security. Chapter 2 provides an overview into network protocols and several key aspects of network protocols that relate to security. Chapter 3 focuses on key aspects of the Internet, such as routing and addressing, and how they relate to security. Chapter 4 introduces the taxonomy for network-based vulnerabilities and attacks and introduces a network threat model that is the basis for analyzing vulnerabilities, attacks, and countermeasures in the remaining chapters of this book.

Chapter 1

Network Architecture

Before discussing network concepts and security it would be helpful to review a brief history of networking [1–9], since we often discover that what was done in the past has an effect on the security of today. Figure 1.1 shows a timeline of the history of networking.

As can be seen from the figure, a lot has changed in the past 30 years. Both the size and complexity of networks have increased. The networks were designed to provide connectivity and not to support security. The first networks in the 1970s were between a small number of research organizations and universities [8, 9]. Everyone that was connected was trusted and security was not an issue. In 1988, the first major attack [10] against computers connected to the Internet was released, and to this day some of the same underlying methods used by that attack still work. What has driven innovation and growth in the network is ease of use and interconnection, not security. We will see this throughout the remainder of the book.

1.1 Layered Network Architecture

This section provides an overview into how networks are implemented and describes the functions provided by a network. A network is divided into different functional components called layers [11, 12]. Each of these layers has a different responsibility for providing the overall functionality of a modern network. The layers can be implemented in software or hardware, and not every layer is needed for every device on the network. For example, routers do not need to implement every layer since they are not responsible for the end-to-end transport of the data; they are only concerned with getting data to the next point on the network. This section starts with a description of the network's layered architecture and then describes the services and functions provided by the layers in the Internet.

The first examples of computer communication consisted of point-to-point connections between the two devices wishing to communicate. In this case, the

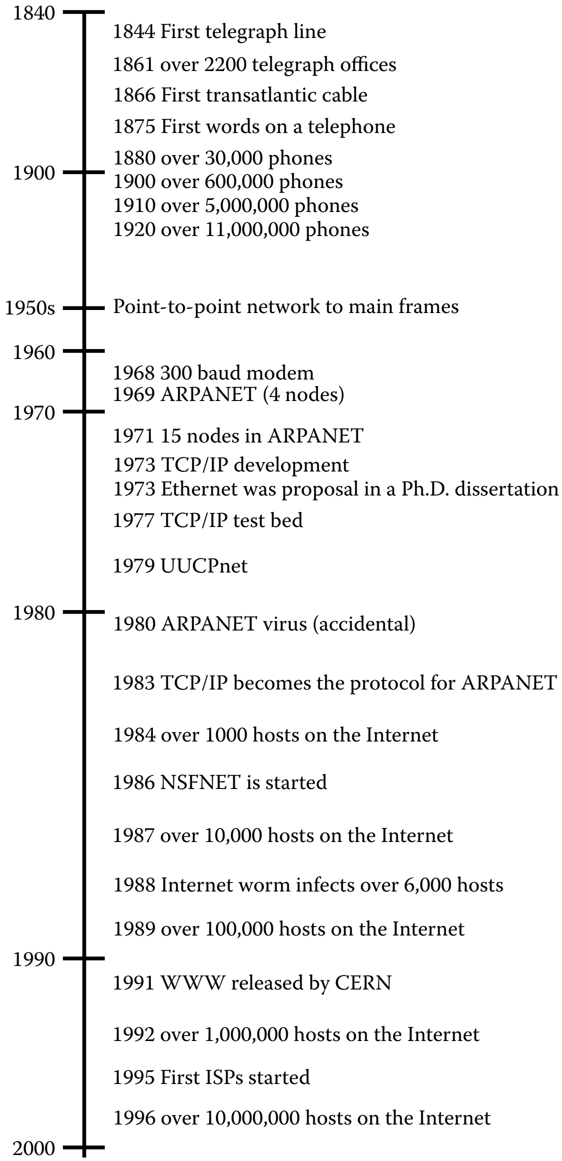
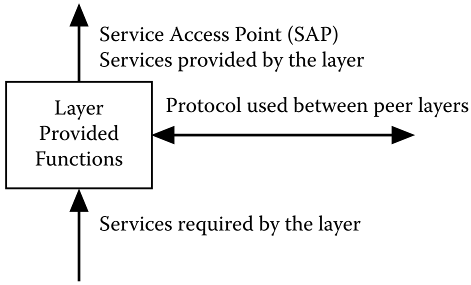


Figure 1.1: History of networking.

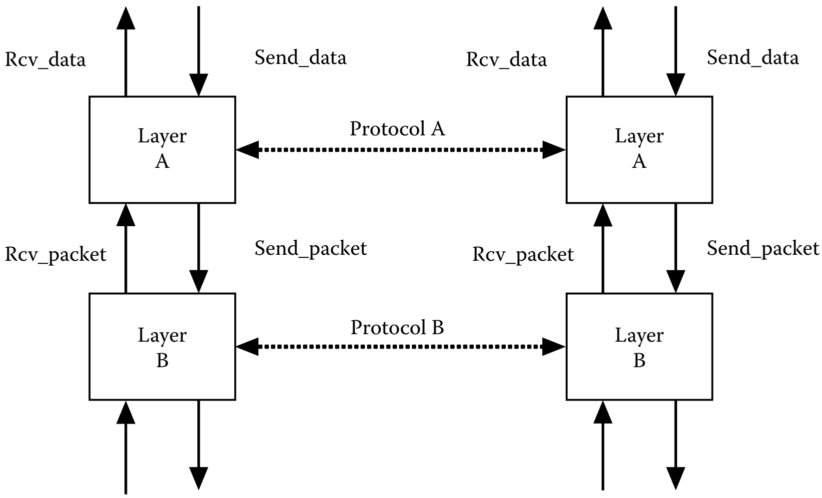
software required to communicate was completely self-contained and often was proprietary to the vendor. The physical connection was either direct using wires or over the telephone using a modem. The data rates were low compared to those of today's networks, and the applications often used simple text-based communications. These early applications were typically used for either simple file transfer or remote access. With these applications there was no need to have data relayed between computers. One of the first applications that used relaying of data between computers was email. Early email systems were designed to transport text messages between computers of like types. As with the early file transfer systems, they used proprietary software to enable communications, which made emailing between dissimilar computer systems difficult.

In the 1970s there was an effort started to develop standards [13] to allow different devices to communicate with each other over a network. The architects of the early standards decided that the problem should be divided into functional modules to enable the development of different methods for different computers to communicate with each other. Each of these modules, or layers, would perform a set of functions and provide a set of services to the layer above it using the services provided by the layer below. Figure 1.2 shows a diagram of the black box approach to defining a layer. Figure 1.2a shows that as with any black box design, the inputs and outputs are specified as a set of services along with the functions that need to be carried out. The services provided by a layer are called service access points (SAPs). Each layer carries out a set of functions specified in the standard. These functions are used to support the services and often involve communication between the corresponding layers on the two devices wishing to exchange data. This interlayer communication is called the protocol. The actual method to implement the layer is not specified as part of the standard. As we will see later, this can lead to some interesting security problems. This black box approach to defining each layer allowed different vendors to implement the same functions and services.

As we see in Figure 1.2b, layer A provides services to the layer above it and layer B provides services to layer A. These services are often specified as subroutine calls like we see in a program. For example, there might a service called send_data(destination, source, data, options, length) provided by layer A, which defines a service that is used to send a block of data to the corresponding layer A on another device specified by the destination address. The service has several parameters that can be used to instruct the layer on how to handle the service request, or may include information that is meant to be passed to the other peer layer. The parameter data in this example would contain the data that is to be passed from layer A to the corresponding layer A on the destination device. Each layer will use services provided by the layer below it to carry out the functions it provides. So as shown in Figure 1.2b, layer B might provide a service called send_packet(destination, source, data, options).



(a)



(b)

Figure 1.2: Network layers.

Notice that in this example layer B provides a send_packet routine that sends a fixed amount of data, but the upper layer A provided a service that could send a larger amount of data. This is where the functions provided by a layer come into play. In this example, layer A will need to provide a function that splits the data it receives from the upper layer into smaller packets and sends them into the lower layer. The corresponding layer A that receives the data will need to provide a function that puts the packets back together and presents a block of data to the upper layer. For a layer to communicate with its corresponding layer, it must send data to the layer below. For a layer to carry out its functions, it must also be able to communicate control information to the corresponding layer. Based on our example shown in Figure 1.2b, layer A will need to send control

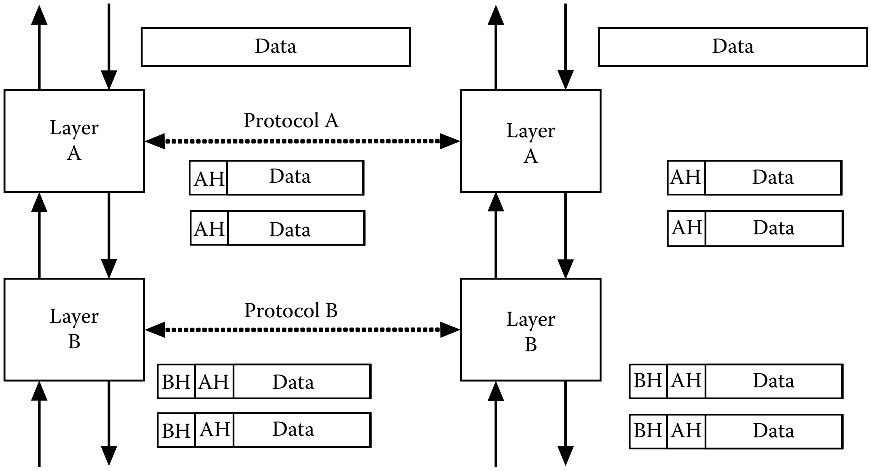


Figure 1.3: Control information encapsulation.

information that can be used by the receiving layer A to reassemble the packets. There are also rules that dictate the interaction between two corresponding layers, such as maximum packet size, format of the control information and data, timing and sequence of control messages, etc. These rules are called a protocol, and the control information is used to carry out the protocol. Every layer is defined as a combination of services, functions, and protocols. Figure 1.3 shows how the control information might be added (encapsulated) to the data as each of the layers processes the requests from the layer above it.

As we see in Figure 1.3, the data presented to layer A is divided into two packets by layer A. Each packet has control information added, which would include information on how to put the two packets back together when they are received by layer A on the destination device. The control information section of the packet is called the header. Layer A passes the two packets to layer B using the services provided by layer B. Layer B adds its own control information (header) to each packet it handles to enable it to communicate with layer B on the destination device. This continues as the packets flow down the network layers until the packets reach the physical transmission media. When the packets are received at the destination, each layer on the receiving device will use the control information to determine how to handle the packet. The layer will strip off the control information that is relevant to it and pass the rest of the packet up to the next-higher layer.

Figures 1.2 and 1.3 showed the interaction between layers as data was passed down the protocol stack and back up the receiving side. Another part of the layer specification is the protocol used between the corresponding layers. For example, in Figure 1.3, layer A on each device needs to understand how to handle packets of data. It needs to know the format of the control information. The protocol is used to provide the functions. For example, another function that could be provided by a layer would be to ask for packets to be resent if there is an error in a packet or a packet is missing. In order to implement this function, the layer would need to determine when a packet is corrupt or missing. This will require coordination between the layers using a protocol. A protocol defines how control information and data are exchanged between layers, and also defines the format of the information exchanged between the layers. The protocol is needed to implement the functions and services. Functions provided by a layer can be exploited by an attacker and will be detailed in subsequent chapters of this book. However, there are several basic functions provided by layers that are highlighted below:

1. **Segmentation and reassembly:** There are cases when a layer has a restriction on the amount of data it will allow from the layer above. This may be because of limits in the amount of buffer space, the protocol headers, or because of limits of the physical connection. For example, many physical local area networks (e.g., Ethernet) limit the packet size to a couple thousand bytes to ensure fair access to the physical network. As shown in Figure 1.3, if a layer receives more data from the upper layer than the layer below it can handle, the data must be divided into smaller packets (segmentation) and eventually put back together by the receiving layer (reassembly). The layer that does the segmentation is responsible for putting the reassembly instructions in its header, which is typically some type of packet number and data offset.

2. **Encapsulation:** Encapsulation is the addition of control information to the packet in the form of a header. This was shown in Figure 1.3. The headers typically contain the following information:

   Address: The address of the sender or receiver.

   Error detection code: Some sort of code is often included for error detection.

   Protocol control: Additional information needed to implement the protocol.

3. **Connection control:** A layer may use connectionless data transfer or connection-oriented data transfer. In connection-oriented data transfer, a logical association, or **connection**, is established between entities before any data is transferred. This is similar to the phone system, where a person dials the number and waits for the other side to pick up the phone before

the two sides can talk. In connection-oriented data transfer both sides have to be ready to talk at the same time. The connection is established using information in the headers of the packets, and in many cases the packets used to establish the connection contain no data. The three phases of **connection control** are the request/connect phase, the data transfer phase, and the termination phase. Many network-based attacks focus on the connection control exchanges. In a network that uses connectionless data transfer, each packet is independent of every other packet and can be delivered out of order and may not be delivered at all. This is analogous to the postal mail system. The sender can send a letter and it will arrive at some time, and each letter is independent of every other letter.

4. **Ordered delivery:** In some cases the service provided by the layer requires the packets to be delivered in order, but the packets may be delivered out of order by the layer below. This is true in the Internet, where the packets are transferred using a connectionless protocol, but the applications require the packets to be delivered in the same order they were transmitted. In order for a layer to provide this service, it will need to add control information to the header to be able to number the packets so they can be put back together by the receiving layer.

5. **Flow control:** Flow control is a technique for ensuring that the transmitting layer does not overwhelm a receiving layer. Flow control is typically implemented in several layers and is found in most connection-oriented protocols.

6. **Error control:** Errors can occur in the transmission of packets. Whether the packet is lost or corrupted, the layer may be responsible for detecting missing or damaged packets and retransmitting these packets. Not every layer is responsible for retransmission of packets, but most layers have some type of error detection (generally using a checksum) in the header. Attackers can sometimes use the error control protocols in an attack by sending corrupt packets to a device and causing the layer to react.

7. **Multiplexing:** Multiplexing is when packets from multiple upper layers share a lower layer. The best example of this is to consider a computer connected to a single physical network. If you think of all of the applications that are using the network at the same time (web, email, IM, etc.), each of them would send packets on the physical network. It makes sense to only have one layer that controls access to the physical network. Therefore, somewhere within the computer's multiple network layers there needs to

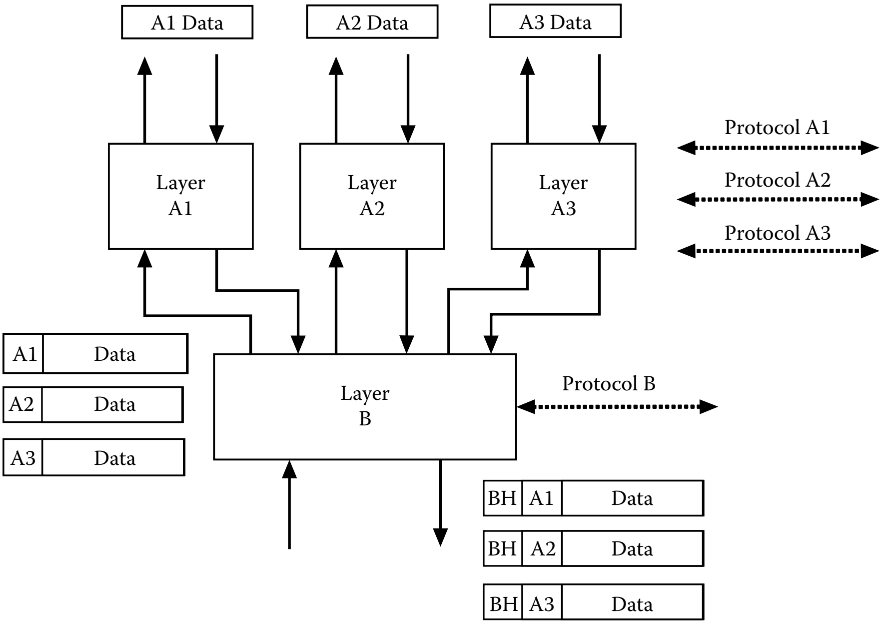


Figure 1.4: Layer multiplexing.

be one or more layers that can support multiple upper layers. Figure 1.4 shows an example of multiplexing. Notice in the example that several layers use the services provided by layer B. In order for the receiving layer B to know which layer A is to get the packets, the layer B header will need to include an address in the packet header to indicate the identity of each of the upper layers.

Definitions

Connectionless.
No connection is needed to transfer data.

Connection oriented.
Before data can be transferred, the two communicating parties must agree to communicate by establishing a connection.

Encapsulation.
Adding layer headers to the data to create a new packet.

Error control.
A function provided by a layer that will detect and try to correct packet loss and packet corruption.

Flow control.
A function provided by a layer that will slow the sender's packet transmission rate when the receiver starts to get behind.
Layered network functions.
A set of operations provided by a layer in coordination with its peer layer on another device in the network designed to provide network services. Functions enable the services provided by a layer to work and rely on the services provided by the lower layer.
Multiplexing.
When a layer provides service access points to multiple upper layers and in turn only uses service access points from one lower layer to send and receive the packets for the multiple upper layers.
Network layer.
A functional component of a network architecture that has a defined set of inputs and outputs and provides a set of functions that aid in the operation of the network.
Packet.
A block of data that is passed between layers.
Packet header.
The part of the packet that is added by a layer to enable the protocol to function.
Protocol.
A set of rules that govern the interaction between two peer layers in the network architecture. The protocol is used to carry out the functions of the layer.
Reassembly.
A function provided by a layer that combines packets that were segmented by a peer layer back into the original data element.
Router.
A network device that is responsible for moving data from one network to another network. A router understands the route the data needs to take to get from the sender to the receiver.
Segmentation.
A function provided by a layer that divides the data received from an upper layer into multiple smaller data elements.
Service access point.
The set of services provided by a network layer. SAPs are often defined as a series of subroutine calls.

1.2 Overview of a Protocol

Protocols are in use every day. For example, the telephone system can be viewed as having multiple layers, each with a protocol. There is a protocol used between the two people talking. Think of this as the upper layer in a network. The phone system is the lower layer that provides basic services and functions to the layer above. Figure 1.5a shows the protocol exchange between the devices in the phone system, and Figure 1.5b shows the protocol exchange between two users of the telephone system. The protocol exchange is often expressed as a protocol diagram, as shown in Figure 1.5, where the vertical lines represent the communicating layers and the horizontal lines indicate information exchange. The diagram also can show a temporal element since time progresses down the diagram. The slanted horizontal lines represent the time it takes for the information to flow from one side to the other. The gaps between the lines represent wait or processing time by the layer.

So, as we can see in Figure 1.5a, the caller on the left side of the diagram starts by picking up the receiver. The caller listens for a dial tone, which is part of the protocol, after hearing the dial tone the caller dials the number. If the called party's phone is not busy, then the caller gets a ring tone and the called party's phone rings. We can also see that the diagram shows error conditions like a busy signal. Not all possible error conditions may have been specified as part of the standard, and therefore would not be covered in the protocol definition. As we will see later, this can cause security problems. Once the called party picks up the phone, the connection between the lower layers is completed and the two people start a protocol, as shown in Figure 1.5b.

First, the person answering the telephone starts the interaction by saying something and the other person responds. The figure shows a possible protocol and also shows an attempt at authenticating the called party. The two people will continue to talk (send data) in a back-and-forth manner until one of them terminates the communication. This is often done by saying goodbye; however, the call can be terminated by just hanging up. This abrupt termination is often used when something has gone wrong between the two parties. The protocol between the two parties is not well defined, and therefore the protocol may fail. One part of the protocol is often identification of one or more parties. This is done through many different methods. We do have a method that is part of the phone system to identify the calling device (caller id). However, caller id identifies the phone number of the caller and not the person using the phone. There is no method to identify the actual calling or called party. We can imagine that this could lead to problems if a

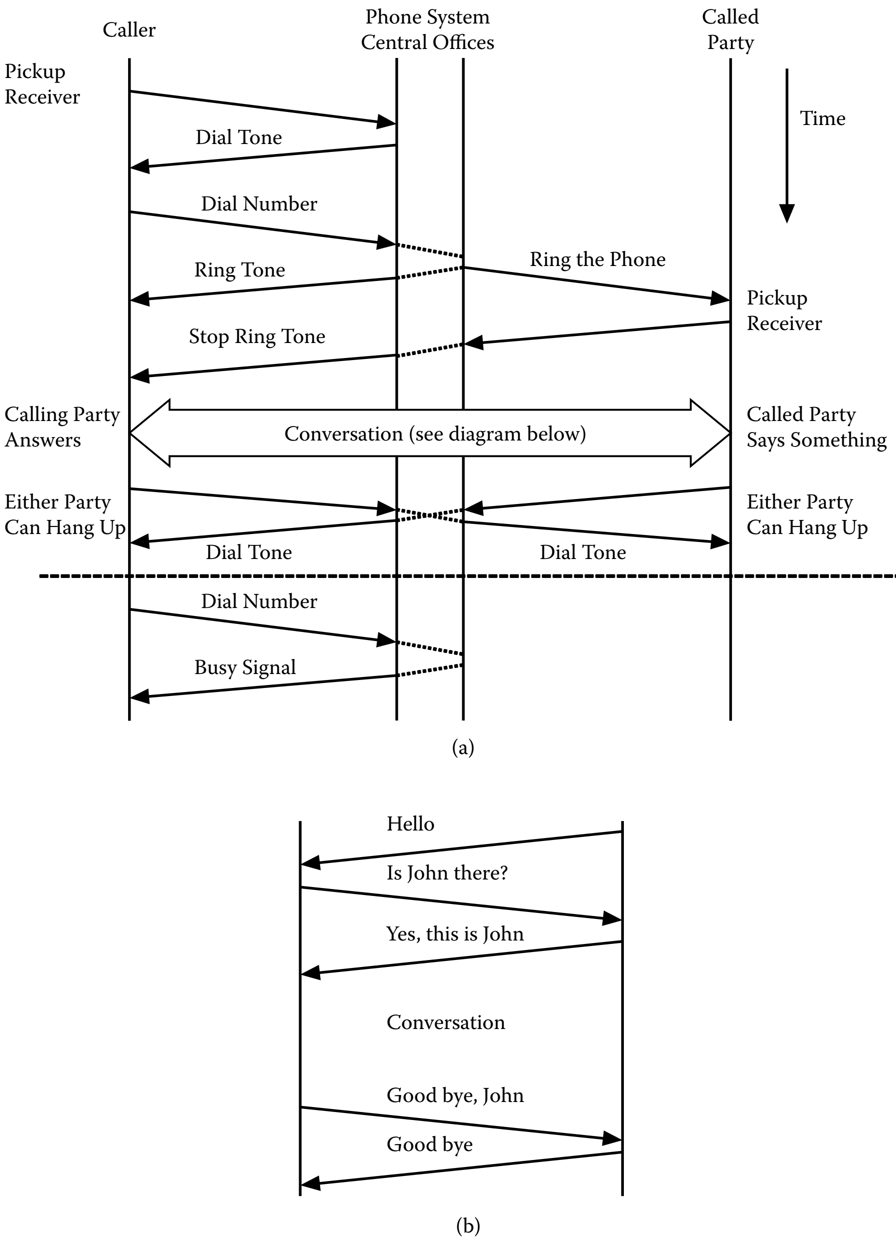


Figure 1.5: Phone system protocol diagram.

person wanted to use the phone for dishonest purposes. Even with caller id, only the phone is identified, even though it was primarily added to provide screening of incoming calls. The phone system was not originally designed to handle what we now consider to be a security problem. Throughout the book we will see many examples of protocols that were not designed with security in mind.

The phone system provides an example of what is called connection-oriented communications. This is where a protocol exchange is used to establish a connection between the two parties (dialing the phone, picking up the phone). Once the connection has been established, the data flows between the two parties and is received in the same order it is sent. There is another method that is used to transfer data between two parties referred to as connectionless. In connectionless communications the information is broken up into packets and each packet is handled separately as it is sent from one party to another. An example of a connectionless system is the post office. Each letter we send is handled independently and could follow a different route to get to the same destination. Each letter is self-contained and has its own address information. If we send multiple letters from the same place to the same destination, there is no guarantee they will all arrive at the same time and in the same order. While the connectionless method may seem to be less reliable than the connection-oriented method, that may not be the case. Let us look at the phone (FAX system) versus the postal mail system and compare sending a ten-page document. (For this analogy we will ignore the difference in data transfer times.) If we use the phone system, the connection must stay up the entire time we are sending the document. The phone system is very reliable; however, if the system were to fail during the transfer, it would need to start over again. If we took the document and divided it up into ten letters and mailed each one, the odds are that most, if not all, would make it. If one is lost, then we would only need to send the lost page again. Now we need a method to put the pages back together again, which can add overhead. This would be part of the protocol used by the sender and receiver of the letters. This would create a connection-oriented system on top of a connectionless service. Later in the book we will see some protocols within the Internet that are connectionless and others that are connection oriented.

Definition

Protocol diagram.

A diagram used to show the interaction between two entities using a protocol. The diagram shows the information flow and the timing between information exchanges.

1.3 Layered Network Model

As we discussed in the previous section, the network functions have been divided into multiple layers. As with many technologies, the standards often follow the first implementation and we can have competing standards. This is also true of networking. The first networks did not follow the layered architecture. In the early 1970s, the concept of packet switching [3, 7, 9] was proposed, and that gave way to the Transmission Control Protocol/Internet Protocol (TCP/IP). In 1984 the International Standards Organization (ISO) proposed a seven-layer network, the Open Systems Interconnection (OSI) model [14], and started to develop standards for each of the layers. The OSI model was heavily influenced by the telecommunications industry and its focus on circuit-switched (connection-oriented) technologies. So with two competing standards there were two competing forces at work trying to push their own agenda. At one point the federal government pushed for the adoption of the OSI standards, while at the same time the TCP/IP standards were being implemented at universities and research labs. As we know, the TCP/IP standards are used by the Internet, and with a few exceptions, the OSI standards have been abandoned. What has remained is the OSI model for describing network layers. Even though the standards are not used, any current standard is always mapped to the OSI model.

Figure 1.6 shows the layers of the TCP/IP model compared to the OSI model. A brief description of the functions provided by each layer in the OSI model is listed next, along with a description of the TCP/IP layers. As we see in Figure 1.6, some of the layers are implemented in hardware and some in software. We also see that in a typical implementation the lower layers are part of the operating system and the upper layers are part of the user space and often contained within the application. In addition, Figure 1.6 shows that not all devices need every layer, and how some protocols are between the end systems and some protocols are between intermediate devices like routers.

The following list highlights the functions [12] provided by each layer of the OSI and TCP/IP models.

1. Physical layer: The physical layer is responsible for the transparent transmission of bit streams across the physical interconnection of systems. The physical layer must provide the data link layer with a means to identify the endpoint (typically using source and destination addresses). The physical layer must deliver the bits in the same order in which they were offered for transmission by the data link layer.

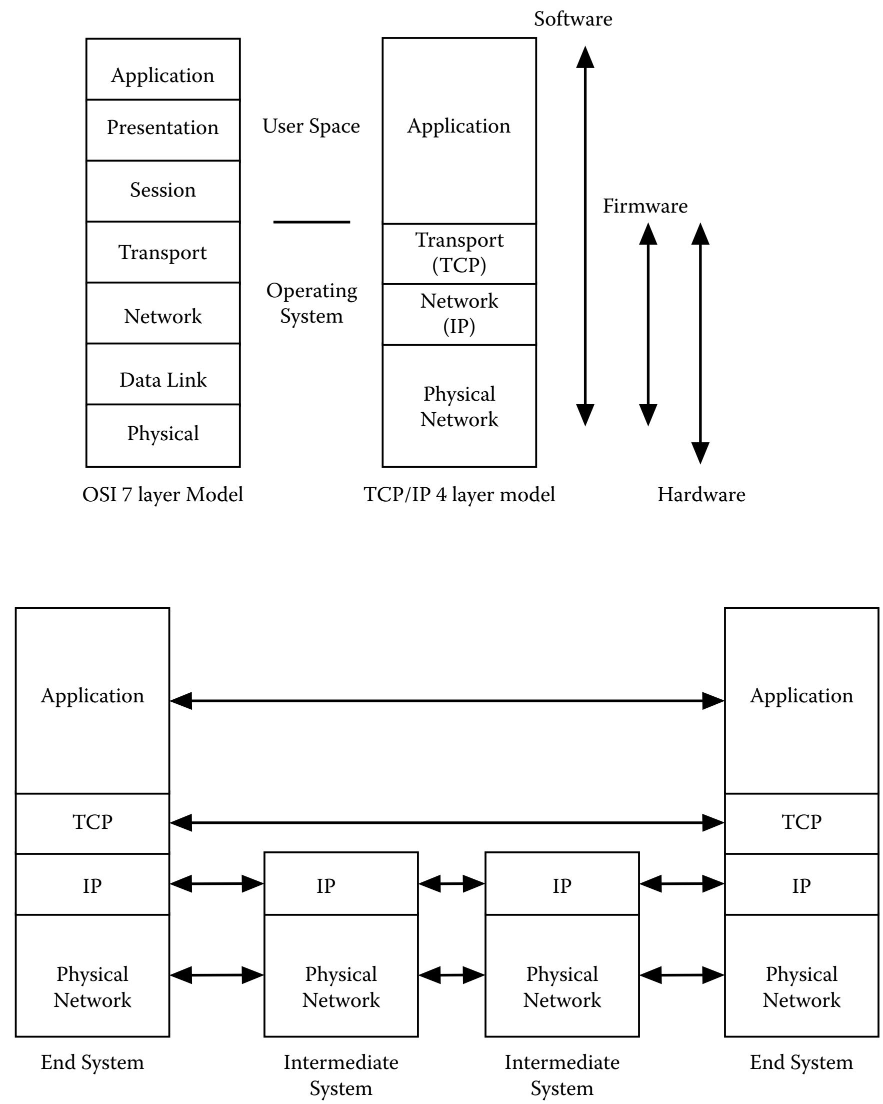


Figure 1.6: OSI and TCP/IP models.

2. Data link layer: The main task of the data link layer is to shield higher layers from the characteristics of the physical transmission medium. The data link layer should provide the higher layers with a reliable transmission that is basically **error-free**, although errors may occur in the transmission on the physical connection. Each data unit from the network layer is mapped to the data link protocol data unit along with the data link protocol information,

and is called a **frame**. The data link layer must provide a method of recognizing the start and end of the frame. Frames must be presented to the physical layer in the same order they are received. The data link layer can also implement **flow control** to prevent data overrun.

3. Network layer: The primary responsibility of the network layer is to provide the transparent transfer of all data submitted by the transport layer to any transport layer anywhere in the network. The network layer must handle the routing of data packets. The network layer can be the highest layer in a device, such as a gateway or router. In the OSI model the network layer was first designed to be connection oriented, and therefore the protocol was complex.

4. Transport layer: The transport layer is responsible for the **reliable** transparent data transfer between two session layer entities. The transport layer is only concerned with the transfer of data between session layers. It is not aware of the structure of the underlying layers or the topology. The transport layer will use the network layer to get data from one transport entity to another. Depending on the quality of the service provided by the network layer, the transport layer may have to perform additional functions, like ordered delivery, to offer the service. The transport layer provides flow and error control.

5. Session layer: The session layer is not concerned with the network. The session layer's goal is to coordinate the dialog between presentation layers. The session layer must provide the establishment of a session connection and the management of the dialog on that connection. The session layer in the OSI model was one of the last layers to be standardized and can be optional in that it can provide no functions and just pass data from the presentation layer to the transport layer. An example of a session layer would be an ATM, which maintains a constant connection with a bank (transport service). A session would start when the user starts a transaction.

6. Presentation layer: The presentation layer provides the application layer with services related to the presentation of information in a form that is meaningful to the application entities. The presentation layer provides the mechanism for the application layer to translate its data into a common format that can be translated by the peer application layer.

7. Application layer: The highest layer provides a means for application processes to access the OSI stack. The application layer provides the protocol to carry out the functions of the application. The application layer typically does not define the user interface or even the user-level commands to carry out the functions. A good example is the web; the application protocol (Hypertext Transfer Protocol [HTTP]) defines the functions and services needed to access web pages and transfer information to the web browsers, but does not specify how the browser will interact with the user.

Most of the functions provided in the OSI model are also provided in the TCP/IP [15] model. The biggest difference is that the application layer in the TCP/IP model encompasses the upper three layers in the OSI model. Many applications do not require all of the functions provided by the session and presentation layers, and even in the OSI model these functions were implemented as part of the application. The descriptions below set the stage for the remainder of the book. The service, functions, and security weaknesses of each of the TCP/IP layers will be discussed in subsequent chapters.

1. TCP/IP physical network layer: The TCP/IP physical network layer combines the functions of the OSI physical and data link layers. The services provided are simple and consist of sending and receiving packets. The TCP/IP protocols are designed to operate on any type of network, and therefore assume a minimal level of service.

2. Network (IP) layer: The network (IP) layer provides the routing of packets across the Internet and also is concerned with the global address space. The IP layer is connectionless, and the services provided consist of sending and receiving packets.

3. Transport (TCP) layer: The transport (TCP) layer, just like the OSI transport layer, is responsible for the reliable end-to-end transfer of data across the network. The TCP layer will use the send and receive packet functions provided by the network layer to communicate with its peer transport layer. The TCP layer will need to compensate for the IP layer's unreliable connectionless service.

4. TCP/IP application layer: The application layer provides the same types of services as the upper three layers in the OSI protocol model. Depending on the application, the functions of the session and presentation layer might be minimal or nonexistent.

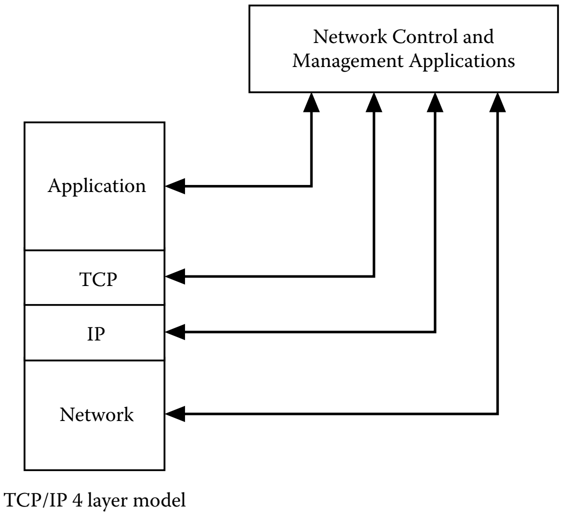


Figure 1.7: Nonlayered services.

When the layered architecture was designed, little thought was given to network management, network security, or network monitoring. These services were not considered important when networks were small and primarily controlled by a few organizations. As networks have grown in size and complexity, the need for these services has also grown. As we look at the requirements for these services, it quickly becomes obvious that the layered model does not map into the requirements of these services. These services need access to the inner workings of each layer, and often need to read or modify internal parameters within the layer. Network management, for example, often requires direct control over each layer. This led to a modified network architecture where several nonlayered services are introduced, as shown in Figure 1.7. This also has an impact on security since programs are given access to each layer. For example, a rogue program might be able to interject packets at a lower layer that violates the header format of the layer above.

Definitions

Frame.
The name used to describe the packet used by the data link layer in the OSI networking model.

Nonlayered services.
Used to describe network services that need access to one or more layers directly, without using other layers. Often used in network management.

OSI model.
A seven-layer model that describes the high-level functions that should be provided by each of the layers that make up a complete network implementation.
TCP/IP model.
A four-layer model that describes the high-level functions that are implemented to support the Internet.
User space.
Programs that run in user space have the same access rights as the user that is running them, which can limit the access the program has to system files.

Homework Problems and Lab Experiments

Homework Problems

1. From a design standpoint, provide three reasons why the layered network architecture is better than a nonlayered architecture?
2. Why would the network designers include fragmentation as a function instead of just requiring all packets to be a certain size?
3. Assume each layer adds 20 bytes of header information. Plot a curve that shows the percentage overhead versus the user payload size for both the seven-layer OSI model and the four-layer TCP/IP model. (Use data sizes from 1 to 1,400 bytes.)
4. Assume the four-layer TCP/IP network model, with each layer adding 20 bytes of header information and a maximum physical layer packet size of 1,500 bytes (the maximum size of the packets transmitted on the physical network). Create a table showing the number of packets and the total number of bytes transmitted given each of the following sizes for the user data.

   a. 1,000 bytes
   b. 10,000 bytes
   c. 100,000 bytes
   d. 1 million bytes

5. Compute the percentage overhead for each of the user data sizes in problem 4.

6. Describe a common action (like using an elevator) in the form of a protocol diagram.
7. Research the history of the OSI networking model versus the TCP/IP model showing a timeline of the two models and their adoption. Comment on the government's efforts to standardize on the OSI model and why that did not work.

Lab Experiments

1. Using resources found on the Internet, plot the growth of the following over the past 20 years:
   a. Estimated number of hosts on the Internet
   b. Estimated number of web sites on the Internet
   c. Estimated total web traffic volume
   d. Estimated total FTP traffic volume
   e. Estimated total Internet traffic volume
2. Using resources found on the Internet, look up the history of the Internet and reference it to other world events.
3. Using resources found on the Internet, research the history of network speed and compare it to the history of the Internet developed in lab experiment 2. Comment on what you discover. Do you think the growth of the Internet was driven by the growth of network speed, or that the growth of the Internet drives the need to faster networks?

References

[1] Casson, H. N. 1910. *The history of the telephone*. Manchester, NH: Ayer Company Publishers.

[2] Winston, B. 1998. *Media technology and society: A history: From the telegraph to the Internet*. London: Routledge.

[3] Poole, H., et al. 1999. *History of the Internet: A chronology*, *1843 to the present*. Santa Barbara, CA: ABC-CLIO, INC.

[4] Cerf, V. G. 2004. On the evolution of Internet technologies. *Proceedings of the IEEE* 92:1360–70.

[5] Leiner, B., et al. 1985. The DARPA Internet protocol suite. *IEEE Communications Magazine* 23:29–34.

[6] Baran, P. 1964. On distributed communications networks. *IEEE Transactions on Communications* 12:1–9.

[7] Cerf, V., and R. Kahn. 1974. A protocol for packet network intercommunication. *IEEE Transactions on Communications* 22:637–48.

[8] Abbate, J. 1994. *From ARPAnet to Internet: A history of ARPA-sponsored computer networks, 1966–1988*. Philadelphia: University of Pennsylvania.

[9] Hauben, M. 1994. *History of Arpanet*, 2000. New York: Columbia University.

[10] Spafford, E. H. 1989. The Internet worm program: An analysis. *ACM SIGCOMM Computer Communication Review* 19:17–57.

[11] Zimmermann, H. 1980. OSI reference model—The ISO model of architecture for open systems interconnection. *IEEE Transactions on Communications* 28:425–32.

[12] Halsall, F. 1995. *Data communications, computer networks and open systems*. Redwood City, CA: Addison Wesley Longman Publishing Co.

[13] Russell, A. L. 2006. Rough consensus and running code and the Internet—OSI standards war. *IEEE Annals of the History of Computing* 28:48–61.

[14] Day, J. D., and H. Zimmermann. 1983. The OSI reference model. *Proceedings of the IEEE* 71:1334–40.

[15] Forouzan, B. A., and S. C. Fegan. 1999. *TCP/IP protocol suite*. New York: McGraw-Hill Higher Education.

References

1 Chapter 1. Network Architecture

[1] Casson, H. N. 1910. The history of the telephone. Manchester, NH: Ayer Company Publishers.

[2] Winston, B. 1998. Media technology and society: A history: From the telegraph to the Internet. London: Routledge.

[3] Poole, H., et al. 1999. History of the Internet: A chronology, 1843 to the present. Santa Barbara, CA: ABC-CLIO, INC. of the IEEE 92:1360–70.

[5] Leiner, B., et al. 1985. The DARPA Internet protocol suite. IEEE Communications Magazine 23:29–34.

[6] Baran, P. 1964. On distributed communications networks. IEEE Transactions on Communications 12:1–9.

[7] Cerf, V., and R. Kahn. 1974. A protocol for packet network intercommunication. IEEE Transactions on Communications 22:637–48.

[8] Abbate, J. 1994. From ARPAnet to Internet: A history of ARPA-sponsored computer networks, 1966–1988. Philadelphia: University of Pennsylvania.

[9] Hauben, M. 1994. History of Arpanet, 2000. New York: Columbia University.

[10] Spafford, E. H. 1989. The Internet worm program: An analysis. ACM SIGCOMM Computer Communication Review 19:17–57.

[11] Zimmermann, H. 1980. OSI reference model—The ISO model of architecture for open systems interconnection. IEEE Transactions on Communications 28:425–32.

[12] Halsall, F. 1995. Data communications, computer networks and open systems. Redwood City, CA: Addison Wesley Longman Publishing Co.

[13] Russell, A. L. 2006. Rough consensus and running code and the Internet— OSI standards war. IEEE Annals of the History of Computing 28:48–61.

[14] Day, J. D., and H. Zimmermann. 1983. The OSI reference model. Proceedings of the IEEE 71:1334–40.

[15] Forouzan, B. A., and S. C. Fegan. 1999. TCP/IP protocol suite. New York: McGraw-Hill Higher Education.

[1] Alvestrand, H. T. 2004. A mission statement for the IETF. RFC 3935.

[2] Postel, J. 1981. Internet protocol. RFC 791. 1. protocols and architecture. Englewood Cliffs, NJ: Prentice Hall.

[4] Spurgeon, C. E. 2000. Ethernet: The definitive guide. Sebastopal, CA: O'Reilly Media.

[5] Reynolds, J. K., and J. Postel. 1990. Assigned numbers. RFC 1060.

[6] Mockapetris, P., and K. J. Dunlap. 1988. Development of the domain name system. SIGCOMM Computer Communication Review 18:123–33.

3 Chapter 3. The Internet

[1] Comer, D. E. 1995. Internetworking with TCP/IP. Vol. 1. Principles, protocols and architecture. Englewood Cliffs, NJ: Prentice Hall.

[2] Calvert, K. I., M. B. Doar, and E. W. Zegura. 1997. Modeling Internet topology. IEEE Communications Magazine 35:160–63.

[3] Subramanian, L., et al. 2002. Characterizing the Internet hierarchy from multiple vantage points. In INFOCOM 2002: Proceedings of the TwentyFirst Annual Joint Conference of the IEEE Computer and Communications Societies, 2. New York, NY. top-down approach featuring the Internet. Reading, MA: Addison-Wesley.

[5] Postel, J. 1981. Assigned numbers. RFC 790.

[6] Postel, J. 1981. Internet protocol. RFC 791.

[7] Heberlein, L. T., and M. Bishop. 1996. Attack class: Address spoofing. In Proceedings of the 19th National Information Systems Security Conference, Baltimore, MD: 371–77.

[8] Bellovin, S. M. 1989. Security problems in the TCP/IP protocol suite. ACM SIGCOMM Computer Communication Review 19:32–48.

[9] Mockapetris, P., and K. J. Dunlap. 1988. Development of the domain name system. SIGCOMM Computer Communication Review 18:123–33.

[10] Stevens, W. R., and T. Narten. 1990. Unix network programming. ACM SIGCOMM Computer Communication Review 20:8–9.

[11] Huitema, C. 1995. Routing in the Internet. Upper Saddle River, NJ: PrenticeHall.

[12] Halabi, B., S. Halabi, and D. McPherson. 2000. Internet routing architectures. Indianapolis, IN: Cisco Press.

4 Chapter 4. Taxonomy of Network-Based Vulnerabilities

[1] Chien, E., and P. Szo¨r. 2002. Blended attacks, exploits, vulnerabilities and buffer-overflow techniques in computer viruses. VIRUS 1.

[2] Whalen, S., M. Bishop, and S. Engle. 2005. Protocol vulnerability analysis. Technical Report CSE-2005-04, Department of Computer Science, University of California, Davis.

[3] Ramakrishnan, C. R., and R. Sekar. 2002. Model-based analysis of configuration vulnerabilities. Journal of Computer Security, 10:189–209.

[4] Schneier, B. 1998. Cryptographic design vulnerabilities. Computer 31:29– 33.

[5] Shuo, C., et al. 2003. A data-driven finite state machine model for analyzing security vulnerabilities. In Proceedings of 2003 International Conference on Dependable Systems and Networks. San Francisco, CA.

[6] Ritchey, R. W., and P. Ammann. 2000. Using model checking to analyze network vulnerabilities. In Proceedings of IEEE Symposium on Security and Privacy 2000, Oakland, CA: 156–65.

[7] Crandall, J. R., Z. Su, and S. F. Wu. 2005. On deriving unknown vulnerabilities from zero-day polymorphic and metamorphic worm exploits. cations Security, Alexandria, VA: 235–48.

[8] Zakon, H. R. 2006. Hobbes Internet timeline v8.2, www.zakon.org/robert/ internet/timeline/

[9] Gilliam, D., J. Kelly, and M. Bishop. 2000. Reducing software security risk through an integrated approach. In Proceedings of the Ninth IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, Gaithersburg, MD, June, 141–46.

[10] Hoo, K. J. S. 2000. How much is enough? A risk management approach to computer security. Stanford, CA: Stanford University.

[11] Stoneburner, G., A. Goguen, and A. Feringa. 2002. Risk management guide for information technology systems, 800–30. NIST Special Publication.

[12] Hinde, S. 2003. The law, cybercrime, risk assessment and cyber protection. Computers and Security 22:90–95.

[13] McDermott, J., and C. Fox. 1999. Using abuse case models for security requirements analysis. In Proceedings of 15th Annual Computer Security Applications Conference (ACSAC ’99) Scottsdale, AZ: 55.

[14] Arbaugh, W. A., W. L. Fithen, and J. McHugh. 2000. Windows of vulnerability: A case study analysis. Computer 33:52–59.

[15] Venter, H. S., and J. H. P. Eloff. 2003. A taxonomy for information security technologies. Computers and Security 22:299–307.

[16] Ali, A., S. Abdulmotaleb El, and M. Ali. 2006. A comprehensive approach to designing Internet security taxonomy. In Canadian Conference on Electrical and Computer Engineering (CCECE ’06). Ottawa, Canada, 1316–1319.

[17] Chakrabarti, A., and G. Manimaran. 2002. Internet infrastructure security: A taxonomy. IEEE Network 16:13–21.

[18] Irvine, C., and T. Levin. 1999. Toward a taxonomy and costing method for security services. In Proceedings of the 15th Annual Computer Security Applications Conference (ACSAC ’99). In IEEE Systems, Man and Cybernetics Society Information Assurance Workshop. Washington, DC: 76–83.

[20] Templeton, S. J., and K. Levitt. 2001. A requires/provides model for computer attacks. In Proceedings of the 2000 Workshop on New Security Paradigms, Ascona, Switzerland: 31–38.

[21] Garber, L. 2000. Denial-of-service attacks rip the Internet. Computer 33:12- 17.

5 Chapter 5. Physical Network Layer Overview

[1] Zimmermann, H. 1980. OSI reference model—The ISO model of architecture for open systems interconnection. IEEE Transactions on Communications 28:425–32.

[2] Comer, D. E. 1995. Internetworking with TCP/IP. Vol. 1. Principles, protocols and architecture. Englewood Cliffs, NJ: Prentice Hall.

[3] IEEE 802 standards. http://www.ieee802.org/.

[4] Simon, D., B. Aboba, and T. Moore. IEEE 802.11 security and 802.1 x, p. 802.11-00.

[5] Templeton, S. J., and K. E. Levitt. 2003. Detecting spoofed packets. Paper presented at Proceedings of DARPA Information Survivability Conference and Exposition. Washington, DC: 164–176.

[6] Medhi, D. 1999. Network reliability and fault tolerance. In Wiley Encyclopedia of Electrical and Electronics Engineering. New York: John Wiley & Sons.

[7] Shake, T. H., B. Hazzard, and D. Marquis. 1999. Assessing network infrastructure vulnerabilities to physical layer attacks. In 22nd National Information Systems Security Conference, Arlington, VA: 18–21.

[8] Held, G. 2003. Ethernet networks: Design, implementation, operation, management. New York: Wiley.

[9] Lundy, G. M., and R. E. Miller. 1993. Analyzing a CSMA/CO protocol through a systems of communicating machines specification. IEEE Transactions on Communications 41:447–49. April.

[11] Wagner, R. 2001. Address resolution protocol spoofing and man-in-themiddle attacks. www.sans.org

[12] Kwon, K., S. Ahn, and J. W. Chung. 2004. Network security management using ARP spoofing. Paper presented at Proceedings of ICCSA. Assis: Italy.

[13] Crow, B. P., et al. 1997. IEEE 802.11 wireless local area networks. IEEE Communications Magazine 35:116–26.

[14] O'Hara, B. 2004. The IEEE 802.11 handbook: A designer's companion. IEEE Standards Association.

Piscataway, NJ.

[15] Brenner, P. 1992. A technical tutorial on the IEEE 802.11 protocol. BreezeCom Wireless Communications. San Jose, CA.

[16] Ramanathan, R., J. Redi, and B. B. N. Technologies. 2002. A brief overview of ad hoc networks: Challenges and directions. IEEE Communications Magazine 40:20–22.

[17] Calı`, F., M. Conti, and E. Gregori. 2000. IEEE 802.11 protocol: Design and performance evaluation of an adaptive backoff mechanism. IEEE Journal on Selected Areas in Communications, 18(9).

[18] Carney, W., W. N. B. Unit, and Texas Instruments. 2002. IEEE 802.11 g new draft standard clarifies future of wireless LAN. Texas Instruments.

[19] Wardriving home page. http://www.wardriving.com/.

[20] Shipley, P. 2003. Open WLANs: The early results of wardriving. www.dis.org-filez-openlans.

[21] Kim, M., J. J. Fielding, and D. Kotz. 2006. Risks of using AP locations discovered through war driving. In Proceedings of the 4th International Conference on Pervasive Computing (Pervasive 2006), Dublin, Ireland: 67–82.

[22] Freeman, E. H. 2006. Wardriving: Unauthorized access to wi-fi networks. Information Systems Security 15:11–15. McGrawHill/Osborne.

[24] Beyah, R., et al. 2004. Rogue access point detection using temporal traffic characteristics. In IEEE Global Telecommunications Conference (GLOBECOM’04), Dallas, TX: 4.

[25] Welch, D., and S. Lathrop. 2003. Wireless security threat taxonomy. In IEEE Systems, Man and Cybernetics Society Information Assurance Workshop. Washington, DC: 76–83.

[26] Fleck, B., and J. Dimov. 2003. Wireless access points and ARP poisoning. Online document (accessed October 12, 2001). www.cigital.com

[27] Lim, Y. X., et al. 2003. Wireless intrusion detection and response. In IEEE Systems, Man and Cybernetics Society Information Assurance Workshop. Washington, DC: 68–75.

[28] Cam-Winget, N., et al. 2003. Security flaws in 802.11 data link protocols. Communications of the ACM 46:35–39.

[29] Miller, S. K. 2001. Facing the challenge of wireless security. Computer 34:6–18.

[30] Craiger, J. P. 2002. 802.11, 802.1 x, and wireless security. www.sans.org/ reading-room/whitepapers/wireless/171.php

[31] Arbaugh, W. A. 2003. Wireless security is different. Computer 36:99–101.

[32] Wong, S. 2003. The evolution of wireless security in 802.11 networks: WEP, WPA and 802.11 standards. 28:5. http://www.sans.org/rr/ whitepapers/wireless/1109.php.

[33] Edney, J., and W. A. Arbaugh. 2004. Real 802.11 security: Wi-fi protected access and 802.11 i. Reading, MA: Addison-Wesley Professional.

[34] Boland, H., and H. Mousavi. 2004. Security issues of the IEEE 802.11 b wireless LAN. In Canadian Conference on Electrical and Computer Engineering, 1. Sashatdon, Sashatchewan, Canada. key hash of WPA. ACM SIGMOBILE Mobile Computing and Communications Review Philadelphia, PA: 8:76–83.

[36] Bridges, V. IEEE p802. 1ap/d3. 0.

[37] Zhu, M., M. Moile, and B. Brahman. 2004. Design and implementation of application-based secure VLAN, 29th Annual IEEE Conference on Local Computer Networks CLCN ’04. Tampa, FL: 407.408.

[38] Shi, L., and P. Sjodin. 2007. A VLAN Ethernet backplane for distributed network systems. In Workshop on High Performance Switching and Routing (HPSR ’07). New York, NY: 1–4.

[39] Ferraiolo, D. F., D. R. Kuhn, and R. Chandramouli. 2003. Role-based access control. Boston, MA: Artech House.

[1] Zimmermann, H. 1980. OSI reference model—The ISO model of architecture for open systems interconnection. IEEE Transactions on Communications 28:425-32.

[2] Day, J. D., and H. Zimmermann. 1983. The OSI reference model. Proceedings of the IEEE 71:1334–40.

[3] Forouzan, B. A., and S. C. Fegan. 1999. TCP/IP protocol suite. New York: McGraw-Hill Higher Education.

[4] Comer, D. E. 1995. Internetworking with TCPIP. Vol. 1. Principles, protocols and architecture. Englewood Cliffs, NJ: Prentice Hall.

[5] Leiner, B., et al. 1985. The DARPA internet protocol suite. IEEE Communications Magazine 23:29–34.

[7] Reynolds, J. K., and J. Postel. 1990. Assigned numbers. RFC 1060.

[8] Fuller, V., et al. 1993. Classless inter-domain routing (CIDR): An address assignment and aggregation strategy. RFC 1519.

[9] Dall'Asta, L., et al. 2006. Exploring networks with traceroute-like probes: Theory and simulations. Theoretical Computer Science 355:6–24.

[10] Periakaruppan, R., and E. Nemeth. 1999. Gtrace—A graphical traceroute tool. Paper presented at Proceedings of the 13th Systems Administration Conference. Seattle, WA. (LISA 1999).

[11] Branigan, S., et al. 2001. What can you do with traceroute? IEEE Internet Computing 5(5).

[12] Altunbasak, H., et al. 2004. Addressing the weak link between layer 2 and layer 3 in the Internet architecture. In 29th Annual IEEE International Conference on Local Computer Networks, Tampa, FL: 417–18.

[13] Kumar, S. Impact of distributed denial of service (DDOS) attack due to ARP storm. Lecture Notes in Computer Science. Berlin Springer-Verlag, V. 3421/2005, 997–1002.

[14] de Vivo, M., O. Gabriela, and G. Isern. 1998. Internet security attacks at the basic levels. ACM SIGOPS Operating Systems Review 32:4–15.

[15] Lau, F., et al. 2000. Distributed denial of service attacks. In IEEE International Conference on Systems, Man, and Cybernetics, Nashville, TN: 3.

[16] Richards, K. 1999. Network based intrusion detection: A review of technologies. Computers and Security 18:671–82.

[17] Lippmann, R. P., et al. 1998. The 1998 DARPA/AFRL off-line intrusion detection evaluation. Paper presented at the First International Workshop on Recent Advances in Intrusion Detection (RAID). Louvain-La-Nueve, Belgium.

[18] Hariri, S., et al. 2003. Impact analysis of faults and attacks in large-scale networks. IEEE Security and Privacy Magazine 1:49–54.

[20] Harris, B., and R. Hunt. 1999. TCP/IP security threats and attack methods. Computer Communications 22:885–97.

[21] Hastings, N. E., and P. A. McLean. 1996. TCP/IP spoofing fundamentals. In Conference Proceedings of the IEEE Fifteenth Annual International Phoenix Conference on Computers and Communications, 218–24.

[22] de Vivo, M., et al. 1999. Internet vulnerabilities related to TCP/IP and T/TCP. ACM SIGCOMM Computer Communication Review 29:81–85.

[23] Moore, D., and C. Shannon. 2002. Code-red: A case study on the spread and victims of an Internet worm. In Proceedings of the Second ACM SIGCOMM Workshop on Internet Measurement, Marseille, France: 273–84.

[24] Berghel, H. 2001. The code red worm. Communications of the ACM 44: 15–19.

[25] Moore, D., et al. 2003. Inside the slammer worm. IEEE Security and Privacy Magazine 1:33–39.

[26] Droms, R. 1999. Automated configuration of TCP/IP with DHCP. IEEE Internet Computing 3:45–53.

[27] Perkins, C. E., and K. Luo. 1995. Using DHCP with computers that move. Wireless Networks 1:341–53.

[28] Schulzrinne, H. 2002. Dynamic host configuration protocol (DHCP-forIPv4) option for session initiation protocol (SIP) servers. RFC 3361.

[29] Deering, S., and R. Hinden. 1995. Internet protocol, version 6 (IPv6) specification. RFC 1883.

[30] Hinden, R., and S. Deering. 2003. Internet protocol version 6 (IPv6) addressing architecture. RFC 3513.

[31] Bound, C. J., M. Carney, and C. E. Perkins. 2000. Dynamic host configuration protocol for IPv6, DHCPv6. Internet draft, draft-ietfdhc-dhcpv6-15.txt. 2003. from distributed denial of service attacks using history-based IP filtering. In IEEE International Conference on Communications (ICC'03), Anchorage, AK: 1.

[33] Ferguson, P., and D. Senie. 1998. Network ingress filtering: Defeating denial of service attacks which employ IP source address spoofing. RFC 2267.

[34] McCanne, S., and V. Jacobson. 1993. The BSD packet filter: A new architecture for user-level packet capture. In Proceedings of Winter '93 USENIX Conference. San Diego, CA.

[35] Chapman, D. B. 1993. Network (in) security through IP packet filtering. In Proceedings of the Third UNIX Security Symposium, Baltimore, MD: 63–76.

[36] Tsirtsis, G., and P. Srisuresh. 2000. Network address translation–protocol translation (NAT-PT). RFC 2766.

[37] Srisuresh, P., and M. Holdrege. 1999. IP network address translator (NAT) terminology and considerations. RFC 2663.

[38] Srisuresh, P., and K. Egevang. 2001. Traditional IP network address translator (traditional NAT). RFC 3022.

[39] Senie, D. 2002. Network address translator (NAT)-friendly application design guidelines. RFC 3235.

[40] Braun, T., et al. 1999. Virtual private network architecture. In CATI— Charging and Accounting Technologies for the Internet, 1.

[41] Carugi, M., and J. De Clercq. 2004. Virtual private network services: Scenarios, requirements and architectural constructs from a standardization perspective. IEEE Communications Magazine 42:116–22.

[42] Guo, X., et al. 2003. A policy-based network management system for IP VPN. In International Conference

on Communication Technology Proceedings (ICCT 2003), Beijing, China: 2. at Workshop on Open Signaling for ATM, Internet and Mobile Networks (OPENSIG'98). Toronto, Canada.

[44] Doraswamy, N., and D. Harkins. 1999. IPSEC: The new security standard for the Internet, intranets, and virtual private networks. Englewood Cliffs, NJ: Prentice Hall.

[45] Blaze, M., J. Ioannidis, and A. D. Keromytis. 2002. Trust management for IPSEC. ACM Transactions on Information and System Security 5:95–118.

[46] Elkeelany, O., et al. 2002. Performance analysis of IPSEC protocol: Encryption and authentication. IEEE International Conference on Communications (ICC 2002), New York, NY: 2.

[47] Keromytis, A. D., J. Ioannidis, and J. M. Smith. 1997. Implementing IPSEC. In IEEE Global Telecommunications Conference (GLOBECOM'97), Phoenix, AZ: 3.

7 Chapter 7. Transport Layer Protocols

[1] Zimmermann, H. 1980. OSI reference model—The ISO model of architecture for open systems interconnection. IEEE Transactions on Communications 28:425–32.

[2] Halsall, F. 1995. Data communications, computer networks and open systems. Redwood City, CA: Addison Wesley Longman Publishing Co. York: McGraw-Hill Higher Education.

[4] Comer, D. E. 1995. Internetworking with TCPIP. Vol. 1. Principles, protocols and architecture. Englewood Cliffs, NJ: Prentice Hall.

[5] Postel, J. 1981. Transmission control protocol (TCP). RFC 793.

[6] Schuba, C., et al. 1997. Analysis of a denial of service attack on TCP. In Proceedings of the 1997 IEEE Symposium on Security and Privacy, Oakland, CA: 223.

[7] Joncheray, L. 1995. A simple active attack against TCP. Paper presented at 5th USENIX Security Symposium. Salt Lake City, UT.

[8] Harris, B., and R. Hunt. 1999. TCP/IP security threats and attack methods. Computer Communications 22:885–97.

[9] Bellovin, S. M. 1989. Security problems in the TCP/IP protocol suite. ACM SIGCOMM Computer Communication Review 19:32–48.

[10] Wang, H., D. Zhang, and K. G. Shin. 2002. Detecting SYN flooding attacks. In Proceedings of the Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2002), New York, NY: 3.

[11] Schuba, C., et al. 1997. Analysis of a denial of service attack on TCP. In Proceedings of the 1997 IEEE Symposium on Security and Privacy, Oakland, CA: 223.

[12] Oliver, R. 2001. Countering SYN flood denial-of-service attacks. Invited presentation at the 10th Usenix Security Conference. Washington, DC.

[13] Ricciulli, L., P. Lincoln, and P. Kakkar. 1999. TCP SYN flooding defense. Paper presented at Proceedings of CNDS. San Francisco, CA.

[14] Mutaf, P. 1999. Defending against a denial-of-service

attack on TCP. Paper presented at Proceedings of the Recent Advances in Intrusion Detection Conference. West Lafayette, IN. attacks using QoS regulation. Texas A&M University Tech Report TAMU-ECE2001-06, 45–53.

[16] Arlitt, M., and C. Williamson. 2005. An analysis of TCP reset behaviour on the Internet. ACM SIGCOMM Computer Communication Review 35:37–44.

[17] Dittrich, D. 2000. The dos project's 'trinoo' distributed denial of service attack tool. Technical report, University of Washington. http://staff. washington.edu/dittrich/misc/trinoo.analysis.txt.

[18] Thomsen, D. 1995. IP Spoofing and session hijacking network security, Issue 3. Amsterdam: Elsevier, 6–11.

[19] Cowan, C., et al. 2000. The cracker patch choice: An analysis of post hoc security techniques. Paper presented at Proceedings of the 19th National Information Systems Security Conference (NISSC 2000). Baltimore, MD.

[20] Postel, J. 1980. User datagram protocol. STD 6, RFC 768.

[21] IETF. e. 2000. 164 number and DNS. RFC 2916. http://www.ietf.org/rfc/ rfc2916.txt.

[22] Mockapetris, P. V. 1987. Domain names—Implementation and specification. RFC 1035.

[23] Ateniese, G., and S. Mangard. 2001. A new approach to DNS security (DNSSEC). In Proceedings of the 8th ACM Conference on Computer and Communications Security, Philadelphia, PA: 86–95.

[24] Householder, A., K. Houle, and C. Dougherty. 2002. Computer attack trends challenge Internet security. Computer 35:5–7.

[25] Bellovin, S. M. 1995. Using the domain name system for system breakins. Paper presented at Proceedings of the Fifth Usenix UNIX Security Symposium, Salt Lake City, UT.

[26] Chakrabarti, A., and G. Manimaran. 2002. Internet infrastructure security: A taxonomy. IEEE Network 16:13–21. denial-of-service attacks: A tutorial. IEEE Communications Magazine 40: 42–51.

[28] Lewis, J. A., D.C.C.f. 2002. Assessing the risks of

cyber terrorism, cyber war and other cyber threats. Center for Strategic & International Studies.

[29] Brownlee, N., K. C. Claffy, and E. Nemeth. 2001. DNS measurements at a root server. In IEEE Global Telecommunications Conference (GLOBECOM'01), San Antonio, TX: 3.

[30] Dierks, T., and C. Allen. 1999. The TLS protocol version 1.0. RFC 2246.

[31] Persiano, P., and I. Visconti. 2000. User privacy issues regarding certificates and the TLS protocol: The design and implementation of the SPSL protocol. In Proceedings of the 7th ACM Conference on Computer and Communications Security, Athens, Greece: 53–62.

[32] Paulson, L. C. 1999. Inductive analysis of the internet protocol TLS. Paper presented at Proceedings of Security Protocols: 6th International Workshop, Cambridge, UK, April 15–17.

[33] Díaz, G., et al. 2004. Automatic verification of the TLS handshake protocol. In Proceedings of the 2004 ACM Symposium on Applied Computing, Nicosia, Cyprus: 789–94.

[1] Forouzan, B. A., and S. C. Fegan. 1999. TCP/IP protocol suite. New York: McGraw-Hill Higher Education.

[2] Comer, D. E. 1995. Internetworking with TCPIP. Vol. 1. Principles, protocols and architecture. Englewood Cliffs, NJ: Prentice Hall.

[3] Comer, D. E., and D. L. Stevens. 1996. Internetworking with TCP/IP. Vol. iii. Client-server programming and applications BSD socket version. Upper Saddle River, NJ: Prentice-Hall.

[4] Stevens, W. R., and T. Narten. 1990. UNIX network programming. ACM SIGCOMM Computer Communication Review 20:8–9.

[5] Toll, W. E. 1995. Socket programming in the data communications laboratory. In Proceedings of the Twenty-Sixth SIGCSE Technical Symposium on Computer Science Education, Nashville, TN, 39–43.

[6] Schmidt, D. C., and S. D. Huston. 2001. C++ network programming. Reading, MA: Addison-Wesley Professional.

[7] Stevens, W. R., B. Fenner, and A. M. Rudoff. 2004. UNIX network programming: The sockets networking API. Reading, MA: Addison-Wesley Professional.

9 Chapter 9. Email

[1] Leiner, B. M., et al. 1999. A brief history of the Internet. Arxiv preprint cs.NI/9901011.

[2] Segal, B. 1995. A short history of Internet protocols at CERN. http://www. cern.ch/ben/TCPHIST.html, accessed August 23, 2008.

[3] Mowery, D. C., and T. Simcoe. 2002. Is the Internet a US invention? An economic and technological history of computer networking. Research Policy 31:1369–87.

[4] Leiner, B. M., et al. 1997. The past and future history. Communications of the ACM 40:103.

[5] Hafiz, M. 2005. Security patterns and evolution of MTA architecture. In Conference on Object Oriented Programming Systems Languages and Applications, San Diego, CA: 142–43.

[6] Giencke, P. 1995. The future of email or when will grandma be on the net? In Electro/95 International. Professional Program Proceedings, Boston, MA: 61–67.

[7] Knowles, B., and N. Christenson. 2000. Design and implementation of highly scalable e-mail systems. Paper presented at Proceedings of the LISA Conference, New Orleans, December. (MIME) part one: Format of Internet message bodies. RFC 2045.

[9] Freed, N., and N. Borenstein. 1996. Multipurpose Internet mail extensions (MIME) part two: Media types. RFC 2046.

[10] Moore, K. 1996. MIME (multipurpose Internet mail extensions) part three: Message header extensions for non-ASCII text. RFC 2047.

[11] Freed, N., J. Klensin, and J. Postel. 1996. Multipurpose Internet mail extensions (mime) part four: Registration procedures. RFC 2048.

[12] Freed, N., and N. Borenstein. 1996. Multipurpose Internet mail extensions (MIME) part five: Conformance criteria and examples. RFC 2049.

[13] Postel, J. B. 1982. SMTP-simple mail transfer protocol. RFC 821. http:// www.ietf.org/rfc/rfc0821.txt, accessed August 23, 2008.

[14] Leiba, B., et al. 2005. SMTP path analysis. Paper presented at Proceedings of the Second Conference on E-mail and Anti-Spam (CEAS). Stanford, CA.

[15] Secure, S. 2002. Network working group p. Hoffman request for comments: 3207 Internet mail consortium obsoletes: 2487 February category: Standards track.

[16] Hoffman, P. 1999. SMTP service extension for secure SMTP over TLS. RFC 2487.

[17] Hoffman, P. 2002. SMTP service extension for secure SMTP over transport layer security. RFC 3207.

[18] Manabe, D., S. Kimura, and Y. Ebihara. 2006. A compression method designed for SMTP over TLS, 803. Lecture Notes in Computer Science 3961.

[19] Gray, T. 1993. Comparing two approaches to remote mailbox access: IMAP vs. POP, 1–4. http://www.imap.org/imap.vs.pop.brief.html, accessed August 23, 2008.

[20] Newman, C. 1999. Using TLS with IMAP, POP3 and ACAP. RFC 2595. RFC 2060. Sebastopol, CA.

[22] Garfinkel, S. 1995. PGP: Pretty good privacy. O'Reilly.

[23] Garfinkel, S. L., et al. 2005. How to make secure email easier to use. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Portland, OR: 701–10.

[24] Zhou, D., et al. 1999. Formal development of secure email. Paper presented at Proceedings of the 32nd Annual Hawaii International Conference on System Sciences. Maui, HI.

[25] Borisov, N., I. Goldberg, and E. Brewer. 2004. Off-the-record communication, or, why not to use PGP. In Proceedings of the 2004 ACM Workshop on Privacy in the Electronic Society, Washington, DC: 77–84.

[26] Hidalgo, J. M. G. 2002. Evaluating cost-sensitive unsolicited bulk email categorization. In Proceedings of the 2002 ACM Symposium on Applied Computing, Madrid, Spain, 615–20.

[27] Michelakis, E., et al. 2004. Filtron: A learning-based anti-spam filter. Paper presented at Proceedings of the

First Conference on Email and Anti-Spam (CEAS). Mountain View, CA.

[28] Bass, T., and G. Watt. 1997. A simple framework for filtering queued SMTP mail (cyberwar countermeasures). In MILCOM 97 Proceedings, Monterey, CA: 3.

[29] Cerf, V. G. 2005. Spam, spim, and spit. Communications of the ACM 48:39–43.

[30] Jung, J., and E. Sit. 2004. An empirical study of spam traffic and the use of DNS black lists. In Proceedings of the 4th ACM SIGCOMM Conference on Internet Measurement, Taormina, Sicily, Italy. 370–75.

[31] Golbeck, J., and J. Hendler. 2004. Reputation network analysis for email filtering. Paper presented at Conference on Email and Anti-Spam (CEAS). Mountain View, CA. malicious impostor emails. In Proceedings of the 2006 International Symposium on World of Wireless, Mobile and Multimedia Networks, Buffalo, NY: 657–61.

[33] Gansterer, W. N., A. G. K. Janecek, and P. Lechner. 2007. A reliable component-based architecture for e-mail filtering. In Proceedings of the Second International Conference on Availability, Reliability and Security, 43–52.

[34] Twining, R. D., et al. 2004. Email prioritization: Reducing delays on legitimate mail caused by junk mail. In Proceedings of Usenix Annual Technical Conference, Boston, MA: 45–58.

[35] Levine, J. R. 2005. Experiences with greylisting. Paper presented at Conference on Email and Anti-Spam. Stanford University, Stanford, CA.

[36] Wiehes, A. 2005. Comparing anti spam methods. Masters of Science in Information Security, Department of Computer Science and Media Technology, Gjøvik University College.

[37] Miszalska, I., W. Zabierowski, and A. Napieralski. 2007. Selected methods of spam filtering in email. In CADSM'07. 9th International Conference: The Experience of Designing and Applications, Chapel Hill, NC: 507–13.

[38] de Vel, O., et al. 2001. Mining e-mail content for author identification forensics. ACM SIGMOD Record 30:55–64. Santa Barbara, CA.

[1] Catledge, L. D., and J. E. Pitkow. 1995. Characterizing browsing strategies in the world-wide web. Computer Networks and ISDN Systems 27:1065-73.

[2] Lawrence, S., and C. L. Giles. 1998. Searching the World Wide Web. Science 280:98.

[3] Albert, R., H. Jeong, and A. L. Barabasi. 1999. The diameter of the World Wide Web. Arxiv preprint cond-mat/9907038.

[4] Vass, J., et al. 1998. The World Wide Web. IEEE Potentials 17:33–37. for the uniform identification of objects. Paper presented at Fourth Annual ACIS International Conference on Computer and Information Science. Jeju Island, South Korea: 100–104.

[6] Schatz, B. R., and J. B. Hardin. 1994. NCSA mosaic and the World Wide Web: Global hypermedia protocols for the internet. Science 265: 895–901.

[7] Berners-Lee, T., et al. 1992. World-wide web: The information universe. Internet Research 2:52–58.

[8] Berners-Lee, T. Hypertext transfer protocol. 1996. Work in progress of the HTTP working group of the IETF.< URL: ftp://nic.merit.edu/ documents/internet-drafts/draft-fielding-http-spec-00.txt.

[9] Fielding, R., et al. 1999. Hypertext transfer protocol—http/1.1. RFC 2616.

[10] Fielding, R., et al. 1997. Hypertext transfer protocol—http/1.1. RFC 2068.

[11] Touch, J., J. Heidemann, and K. Obraczka. 1998. Analysis of HTTP performance. ISI Research Report ISI/RR-98-463, USC/Information Sciences Institute. http://www.Isi.edu/touch/pubs/http-perf96.

[12] Raggett, D., A. Le Hors, and I. Jacobs. 1999. HTML 4.01 specification. Paper presented at W3C Recommendation REC-html401-19991224, World Wide Web Consortium (W3C). Cambridge, MA.

[13] Berners-Lee, T., J. Hendler, and O. Lassila. 2001. The semantic web. Scientific American 284:28–37.

[14] Lemay, L. 1994. Teach yourself web publishing with HTML in a week. Indianapolis: Sam's Publishing.

[15] Niederst, J. 2003. Learning web design: A beginner's guide to HTML, graphics, and beyond. O'Reilly Media.

[16] Hendler, J. 2003. Communication: Enhanced: Science and the semantic web. Science 299:520–21. Wiley.

[18] Kirda, E., et al. 2006. Noxes: A client-side solution for mitigating cross-site scripting attacks. In Proceedings of the 2006 ACM Symposium on Applied Computing, Dijon, France: 330–37.

[19] Jiang, S., S. Smith, and K. Minami. 2001. Securing web servers against insider attack. In Proceedings of the 17th Annual Computer Security Applications Conference (ACSAC 2001), New Orleans: 265–76.

[20] Thiemann, P. 2005. An embedded domain-specific language for type-safe server-side web scripting. ACM Transactions on Internet Technology (TOIT) 5:1–46.

[21] Xie, Y., and A. Aiken. 2006. Static detection of security vulnerabilities in scripting languages. In Proceedings of the 15th USENIX Security Symposium, Vancouver, B.C., Canada: 179–92.

[22] Minamide, Y. 2005. Static approximation of dynamically generated web pages. In Proceedings of the 14th International Conference on World Wide Web, Chiba, Japan: 432–41.

[23] Jim, T., N. Swamy, and M. Hicks. 2007. Defeating script injection attacks with browser-enforced embedded policies. In Proceedings of the 16th International Conference on World Wide Web, Banff, Alberta, Canada: 601–10.

[24] Yu, D., et al. 2007. Javascript instrumentation for browser security. In Proceedings of the 34th Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, Nice, France: 237–49.

[25] Erlingsson, U., B. Livshits, and Y. Xie. 2007. End-to-end web application security. Paper presented at Proceedings of the Workshop on Hot Topics in Operating Systems (HotOS XI). San Diego, CA.

[26] Huseby, S. H. 2005. Common security problems in the

code of dynamic web applications. Paper presented at Web Application Security Consortium. http://www.webappsec.org/projects/articles/062105.TX? Web security: The next battleground. In Enhancing Computer Security with Smart Technology. Boca Raton, FL: CRC Press, 41–72.

[28] Marchesini, J., S. W. Smith, and M. Zhao. 2005. Keyjacking: The surprising insecurity of client-side SSL. Computers and Security 24:109–23.

[29] Jovanovic, N., C. Kruegel, and E. Kirda. 2006. Pixy: A static analysis tool for detecting web application vulnerabilities. Paper presented at IEEE Symposium on Security and Privacy. Oakland, CA.

[30] Jackson, C., et al. 2006. Protecting browser state from web privacy attacks. In Proceedings of the 15th International Conference on World Wide Web, 737–44.

[31] Kirda, E., and C. Kruegel. 2005. Protecting users against phishing attacks with antiphish. Paper presented at Proceedings of 29th COMPSAC. Edinburgh, Scotland.

[32] Raffetseder, T., E. Kirda, and C. Kruegel. 2007. Building anti-phishing browser plug-ins: An experience report. Paper presented at Proceedings of the Third International Workshop on Software Engineering for Secure Systems. Minneapolis, MN.

[33] Jakobsson, M., and S. Stamm. 2006. Invasive browser sniffing and countermeasures. In Proceedings of the 15th International Conference on World Wide Web, 523–32.

[34] Reynaud-Plantey, D. 2005. New threats of Java viruses. Journal in Computer Virology 1:32–43.

[35] Fu, S., and C. Z. Xu. 2006. Mobile code and security. In Handbook of information security. Hoboken, NJ: John Wiley & Sons, V. III, Chapter 144.

[36] Tilevich, E., Y. Smaragdakis, and M. Handte. 2005. Appletizing: Running legacy Java code remotely from a web browser. Paper presented at IEEE International Conference on Software Maintenance (ICSM). Budapest, Hungary. Visual of SSL protected web sites and effective countermeasures. Paper presented at Information Security Practice and Experience Conference. Singapore.

[38] Herzog, A., and N. Shahmehri. 2005. An evaluation of

Java application containers according to security requirements. In Proceedings of the 14th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprise (WETICE’05), Linko¨ping, Sweden. 178–83.

[39] Kendall, K. E., and J. E. Kendall. 2002. Systems analysis and design. Upper Saddle River, NJ: Prentice-Hall.

[40] Bergmark, D. 2002. Collection synthesis. In Proceedings of the 2nd ACM/IEEE-CS Joint Conference on Digital Libraries, 253–62.

[41] Kim, J., K. Chung, and K. Choi. 2007. Spam filtering with dynamically updated URL statistics. IEEE Security and Privacy, 33–39.

[42] Lee, P. Y., S. C. Hui, and A. C. M. Fong. 2002. Neural networks for web content filtering. IEEE Intelligent Systems 17:48–57.

[43] Hammami, M., Y. Chahir, and L. Chen. 2003. Webguard: Web based adult content detection and filtering system. In Proceedings of IEEE/WIC International Conference on Web Intelligence (WI 2003), Beijing, China: 574–78.

[44] Lee, P. Y., S. C. Hui, and A. C. M. Fong. 2003. A structural and contentbased analysis for web filtering. Internet Research: Electronic Networking Applications and Policy 13:27–37.

[45] Zittrain, J., and B. Edelman. 2003. Internet filtering in China. IEEE Internet Computing, 70–77.

[46] Sugiyama, K., K. Hatano, and M. Yoshikawa. 2004. Adaptive web search based on user profile constructed without any effort from users. In Proceedings of the 13th International Conference on World Wide Web, New York, NY: 675–84.

11 Chapter 11. Remote Access Security

[1] Page, J. 1986. Kermit: A file-transfer protocol. Accounting Review 61: 368–69.

[2] Walters, W. 1987. Implementing a campus-wide computer-based curriculum. In Proceedings of the 15th Annual ACM SIGUCCS Conference on User Services, Kansas City, MO: 465–68.

[3] Banks, M. A. 2000. The modem reference. Medford, NJ: Cyberage Books. IEEE Internet Computing 2:88–91.

[5] Borman, D. 1994. Telnet environment option interoperability issues. RFC 1571.

[6] Altman, J., and T. Ts’o. 2000. Telnet authentication option. RFC 2941.

[7] Murphy, Jr., T., P. Rieth, and J. Stevens. 2000. 5250 Telnet enhancements. RFC 2877.

[8] Hedrick, C. L. 1988. Telnet remote flow control option. RFC 1080.

[9] Postel, J., and J. K. Reynolds. 1983. Telnet protocol specification. RFC 0854.

[10] Leiner, B., et al. 1985. The DARPA Internet protocol suite. IEEE Communications Magazine 23:29–34.

[11] Tam, C. M. 1999. Use of the Internet to enhance construction communication: Total information transfer system. International Journal of Project Management 17:107–11.

[12] Day, J. 1980. Terminal protocols. IEEE Transactions on Communications 28:585–93.

[13] Cohen, D., and J. B. Postel. 1979. On protocol multiplexing. In Proceedings of the Sixth Symposium on Data Communications, Pacific Grove, CA: 75–81.

[14] Kantor, B. 1991. BSD rlogin. RFC 1282.

[15] Kantor, B. 1991. BSD rlogin. RFC 1258.

[16] Bahneman, L. 1994. The term protocol. Linux Journal 1994(8es).

[17] Stevens, W. R. 1994. TCP/IP illustrated. Reading, MA: Addison-Wesley Professional.

[18] Rogers, L. R. 1998. Rlogin (1): The untold story. NASA. based terminal protocol. In Proceedings of the 14th Conference on Local Computer Networks, Minneapolis, MN: 85–97.

[20] Stevens, W. R. 1995. TCP/IP illustrated. Vol. I, 223–27. Upper Saddle River, NJ: Addision Wesley Publishing Company.

[21] Scheifler, R. W., and J. Gettys. 1986. The X window system. ACM Transactions on Graphics (TOG) 5:79–109.

[22] Richardson, T., et al. 1994. Teleporting in an X window system environment. IEEE Personal Communications Magazine 1:6–12.

[23] Quercia, V., and T. O'Reilly. 1993. X window system user's guide. Sebastopol, CA: O'Reilly.

[24] Nye, A. 1995. X protocol reference manual. Sebastopol, CA: O'Reilly.

[25] McCormack, J., and P. Asente. 1988. An overview of the X toolkit. In Proceedings of the 1st Annual ACM SIGGRAPH Symposium on User Interface Software, 46–55.

[26] Scheifler, R. W., et al. 1990. X-window system: The complete reference to XLIB, X protocol, ICCCM, XLFD: X version 11, release.

[27] Postel, J., and J. Reynolds. 1985. File transfer protocol (FTP). STD 9, RFC 959.

[28] Horowitz, M., and S. Lunt. 1997. FTP security extensions. RFC 2228.

[29] Bellovin, S. 1994. Firewall-friendly FTP. RFC 1579.

[30] Bhushan, A. K. 1973. FTP comments and response to RFC 430. RFC 0463.

[31] Bhushan, A., et al. 1971. The file transfer protocol. RFC 0172.

[32] Neigus, N. 1973. File transfer protocol. RFC 0542.

[33] Sollins, K. 1992. The TFTP protocol (revision 2). RFC

1350.

[34] Emberson, A. 1997. TFTP multicast option. RFC 2090.

[36] Aslam, T., I. Krsul, and E. Spafford. 1996. Use of a taxonomy of security faults. Paper presented at the 19th National Information Systems Security Conference Proceedings, Baltimore.

[37] Stevens, W. R., and T. Narten. 1990. Unix network programming. ACM SIGCOMM Computer Communication Review 20:8–9.

[38] Stevens, W. R. 1994. TCP/IP illustrated. Vol. 1. The protocols, chap. 15. Reading, MA: Addison Wesley.

[39] Golle, P., K. Leyton-Brown, and I. Mironov. 2001. Incentives for sharing in peer-to-peer networks. Electronic Commerce 14:264–67.

[40] Tran, D. A., K. A. Hua, and T. T. Do. 2004. A peer-to-peer architecture for media streaming. IEEE Journal on Selected Areas in Communications 22:121–33.

[41] Androutsellis-Theotokis, S., and D. Spinellis. 2004. A survey of peer-topeer content distribution technologies. ACM Computing Surveys (CSUR) 36:335–71.

[42] Androutsellis-Theotokis, S. 2002. A survey of peer-to-peer file sharing technologies. Athens University of Economics and Business.

[43] Ramaswamy, L., and L. Liu. 2003. Free riding: A new challenge to peerto-peer file sharing systems. Paper presented at Proceedings of the Hawaii International Conference on Systems Science. Big Island, HI.

[44] Lui, S. M., and S. H. Kwok. 2002. Interoperability of peer-to-peer file sharing protocols. ACM SIGecom Exchanges 3:25–33.

[45] Gummadi, P. K., S. Saroiu, and S. D. Gribble. 2002. A measurement study of Napster and Gnutella as examples of peer-to-peer file sharing systems. ACM SIGCOMM Computer Communication Review 32:82.

[46] Daswani, N., H. Garcia-Molina, and B. Yang. 2003. Open problems in data-sharing peer-to-peer systems. In Proceedings of the 9th International Conference on Database Theory, Sienna, Italy: 1–15. pollution and poisoning in

file sharing peer-to-peer networks. In Proceedings of the 6th ACM Conference on Electronic Commerce, San Diego, CA: 68–77.

[48] Yang, B., and H. Garcia-Molina. 2002. Improving search in peer-to-peer networks. In Proceedings of the 22nd International Conference on Distributed Computing Systems, Vienna, Austria: 5–14.

[49] Kant, K. 2003. An analytic model for peer to peer file sharing networks. In IEEE International Conference on Communications (ICC'03), Anchorage, AK: 3.

[50] Saroiu, S., K. P. Gummadi, and S. D. Gribble. 2003. Measuring and analyzing the characteristics of Napster and Gnutella hosts. Multimedia Systems 9:170–84.

[51] Scarlata, V., B. N. Levine, and C. Shields. 2001. Responder anonymity and anonymous peer-to-peer file sharing. In Ninth International Conference on Network Protocols, Riverside, CA: 272–80.

[52] Aberer, K., and M. Hauswirth. 2002. An overview on peer-to-peer information systems. Paper presented at Workshop on Distributed Data and Structures (WDAS-2002). Paris, France.

[53] Moro, G., A. M. Ouksel, and C. Sartori. 2002. Agents and peer-to-peer computing: A promising combination of paradigms. In Proceedings of the 1st International Workshop of Agents and Peer-to-Peer Computing (AP2PC2002), Bologna, Italy. 1–14.

[54] Howe, A. J. 2000. Napster and Gnutella: A comparison of two popular peerto-peer protocols. Department of Computer Science, University of Victoria, British Columbia, Canada.

[55] Braione, P. 2002. A semantical and implementative comparison of file sharing peer-to-peer applications. In Proceedings of the Second International Conference on Peer-to-Peer Computing (P2P 2002), Linko¨ping, Sweden: 165–66.

[56] Fellows, G. 2004. Peer-to-peer networking issues—An overview. Digital Investigation 1:3–6. A peer-topeer system for music information retrieval. Computer Music Journal 28:24–33.

[58] Lam, C. K. M., and B. C. Y. Tan. 2001. The Internet is

changing the music industry. Communications of the ACM 44:62–68.

[59] Leibowitz, N., M. Ripeanu, and A. Wierzbicki. 2003. Deconstructing the KaZaA network. In Proceedings of the Third IEEE Workshop on Internet Applications (WIAPP 2003), San Jose, CA: 112–20.

[60] Liang, J., R. Kumar, and K. W. Ross. 2005. The KaZaA overlay: A measurement study. Computer Networks Journal 49(6).

[61] Good, N. S., and A. Krekelberg. 2003. Usability and privacy: A study of KaZaA P2P file-sharing. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Fort Lauderdale, FL: 137–44.

[62] Bleul, H., and E. P. Rathgeb. 2005. A simple, efficient and flexible approach to measure multi-protocol peer-to-peer traffic. Paper presented at IEEE International Conference on Networking (ICN'05). Reunion Island.

[63] Lowth, C. 2003. Securing your network against KaZaA. Linux Journal 2003(114).

[64] Shin, S., J. Jung, and H. Balakrishnan. 2006. Malware prevalence in the KaZaA file-sharing network. In Proceedings of the 6th ACM SIGCOMM on Internet Measurement, Rio De Janeiro, Brazil: 333–38.

[65] Sen, S., and J. Wang. 2004. Analyzing peer-to-peer traffic across large networks. IEEE/ACM Transactions on Networking (TON) 12:219–32.

[66] Balakrishnan, H., et al. 2003. Looking up data in P2P systems. Communications of the ACM 46:43–48.

[67] Liang, J., et al. 2005. Pollution in P2P file sharing systems. In Proceedings of the 24th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2005), Miami, FL: 2. identification of P2P traffic. In Proceedings of the 4th ACM SIGCOMM Conference on Internet Measurement, Taormina, Italy: 121–34.

[69] Spognardi, A., A. Lucarelli, and R. Di Pietro. 2005. A methodology for P2P file-sharing traffic detection. In Second International Workshop on Hot Topics in Peer-to-Peer Systems (HOT-P2P 2005), San Diego, CA: 52–61.

[70] Liang, J., N. Naoumov, and K. W. Ross. 2006. The index

poisoning attack in P2P file-sharing systems. Paper presented at Infocom 2006. Barcelona, Spain.

[71] Ripeanu, M. 2001. Peer-to-peer architecture case study: Gnutella network. In Proceedings of International Conference on Peer-to-Peer Computing, Linko¨ping, Sweden: 101.

[72] Zeinalipour-Yazti, D. 2002. Exploiting the security weaknesses of the Gnutella protocol. http://www.cs.ucr.edu/ncsyiazti/courses/cs260-2/ project/gnutella.pdf, accessed August 23, 2008.

[73] Saroiu, S., P. K. Gummadi, and S. D. Gribble. 2002. A measurement study of peer-to-peer file sharing systems. Paper presented at Proceedings of Multimedia Computing and Networking. San Jose, CA.

[74] Kwok, S. H., and K. Y. Chan. 2004. An enhanced Gnutella P2P protocol: A search perspective. In 18th International Conference on Advanced Information Networking and Applications (AINA 2004), Fukuoha, Japan: 1.

[75] Aggarwal, V., et al. 2004. Methodology for estimating network distances of Gnutella neighbors. Paper presented at GI Informatik—Workshop on P2P Systems. Ulm, Germany.

[76] Karagiannis, T., et al. 2004. Is P2P dying or just hiding? Paper presented at IEEE Globecom. Dallas, TX.

[77] Klingberg, T., and R. Manfredi. 2002. The Gnutella protocol specification v0. 6. Technical specification.

[78] Matei, R., A. Iamnitchi, and P. Foster. 2002. Mapping the Gnutella network. IEEE Internet Computing 6:50–57. the peer-topeer file-sharing system. Technical Report PDS-2004-007, Delft University of Technology Parallel and Distributed Systems Report Series.

[80] Pouwelse, J. A., et al. 2005. The BitTorrent P2P file-sharing system: Measurements and analysis. Paper presented at International Workshop on Peerto-Peer Systems (IPTPS). Ithaca, NY.

[81] Yang, W., and N. Abu-Ghazaleh. 2005. GPS: A general peer-to-peer simulator and its use for modeling BitTorrent. In Proceedings of the International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS), Atlanta, GA: 425–32.

[82] Bharambe, A. R., C. Herley, and V. N. Padmanabhan. 2006. Analyzing and improving a BitTorrent network's performance mechanisms. Paper presented at Proceedings of IEEE INFOCOM. Barcelona.

[83] Guo, L., et al. 2005. Measurements, analysis, and modeling of BitTorrentlike systems. In Internet Measurement Conference, Berkeley, CA: 19–21.

[84] Davis, B. C., and T. Ylonen. 1997. Working group report on Internet/intranet security. In Proceedings of the Sixth IEEE Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises, Cambridge, MA: 305–8.

[85] Barrett, D. J., R. E. Silverman, and R. G. Byrnes. 2005. SSH, the secure shell: The definitive guide. Sebastopol, CA: O'Reilly Media.

[86] Miltchev, S., S. Ioannidis, and A. D. Keromytis. 2002. A study of the relative costs of network security protocols. In Proceedings of the USENIX Annual Technical Conference, Freenix Track, Monterey, CA: 41–48.

[87] Poll, E., and A. Schubert. 2007. Verifying an implementation of SSH. Paper presented at Workshop on Issues in the Theory of Security (WITS'07). Braga, Portugal.

[88] Song, D. X., D. Wagner, and X. Tian. 2001. Timing analysis of keystrokes and timing attacks on SSH. In Proceedings of the 10th Conference on USENIX Security Symposium, Vol. 10, Washington, DC: 25 implementations. In Proceedings of the 21st IEEE International Conference on Software Maintenance (ICSM '05), Budapest, Hungary: 643–46.

[90] Vaudenay, S. 2005. A classical introduction to cryptography: Applications for communications security. New York, NY: Springer.

[91] Longzheng, C., Y. Shengsheng, and Z. Jing-li. 2004. Research and implementation of remote desktop protocol service over SSL VPN. In Proceedings of the IEEE International Conference on Services Computing (SCC 2004), Shanghai, China: 502–5.

[92] Tsai, P. L., C. L. Lei, and W. Y. Wang. 2004. A remote control scheme for ubiquitous personal computing. In 2004 IEEE International Conference on Networking, Sensing and Control, Taipei, Taiwan: 2.

[93] Miller, K., and M. Pegah. 2007. Virtualization: Virtually at the desktop. In Proceedings of the 35th Annual ACM SIGUCCS Conference on User Services, Portland, OR: 255–60.

12 Chapter 12. Common Network Security Devices

[1] Lucas, M., A. Singh, and C. Cantrell. 2006. Firewall policies and VPN configurations. Rockland, MA: Syngress Media.

[2] Rowan, T. 2007. Application firewalls: Filling the void. Network Security 2007:4–7.

[3] Gouda, M. G., and A. X. Liu. 2007. Structured firewall design. Computer Networks 51:1106–20.

[4] Loh, Y. S., et al. 2006. Design and implementation of an XML firewall. In 2006 International Conference on Computational Intelligence and Security, Guangzhou, China: 2.

[5] Jia, Z., S. Liu, and G. Wang. 2006. Research and design of NIDS based on Linux firewall. In 2006 1st International Symposium on Pervasive Computing and Applications, Xinjiang, China: 556–60. a flexible text search-based spam-stopping firewall. Paper presented at Proceedings of the Twenty-Third National Radio Science Conference, Menout, Egypt. (NRSC 2006).

[7] Goldman, J. E. 2006. Firewall architectures. In Handbook of information security, Vol. III, Chapter 170.

[8] Goldman, J. E. 2006. Firewall Basics. In Handbook of information security, Vol. III, Chapter 169.

[9] Byrne, P. 2006. Application firewalls in a defense-in-depth design. Network Security 2006:9–11.

[10] Hamed, H., and E. Al-Shaer. 2006. Dynamic rule-ordering optimization for high-speed firewall filtering. In Proceedings of the 2006 ACM Symposium on Information, Computer and Communications Security, Taipei, Taiwan: 332–42.

[11] Zhou, C., Z. Dai, and L. Jiang. 2007. Research and implementation of complex firewall based on netfilter. Jisuanji Celiang yu Kongzhi/Computer Measurement and Control 15:790–91.

[12] Zhang, C. C., M. Winslett, and C. A. Gunter. 2007. On the safety and efficiency of firewall policy deployment. In IEEE Symposium on Security and Privacy, Oakland, CA: 33–50.

[13] Firewall, B. I. M. 2006. Product roundup. Infosecurity Today 3:12.

[14] Biermann, E., E. Cloete, and L. M. Venter. 2001. A comparison of intrusion detection systems. Computers and Security 20:676–83.

[15] Hegazy, I. M., et al. 2005. Evaluating how well agent-based IDS perform. IEEE Potentials 24:27–30.

[16] Bace, R., and P. Mell. 2001. NIST special publication on intrusion detection systems. intrusion detection systems. In Proceedings of the 2004 International Symposium on Applications and the Internet, Tokyo, Japan: 208–15.

[18] Jansen, W. A. 2002. Intrusion detection with mobile agents. Computer Communications 25:1392–401.

[19] Cavusoglu, H., B. Mishra, and S. Raghunathan. 2005. The value of intrusion detection systems in information technology security architecture. Information Systems Research 16:28–46.

[20] Markatos, E. P., et al. 2002. Exclusion-based signature matching for intrusion detection. In Proceedings of the IASTED International Conference on Communications and Computer Networks (CCN), Cambridge, MA: 146–52.

[21] Undercoffer, J., A. Joshi, and J. Pinkston. 2003. Modeling computer attacks: An ontology for intrusion detection. Paper presented at 6th International Symposium on Recent Advances in Intrusion Detection. Pittsburg, PA.

[22] Mell, P., D. Marks, and M. McLarnon. 2000. A denial-of-service resistant intrusion detection architecture. Computer Networks 34:641–58.

[23] Pillai, M. M., J. H. P. Eloff, and H. S. Venter. 2004. An approach to implement a network intrusion detection system using genetic algorithms. In Proceedings of the 2004 Annual Research Conference of the South African Institute of Computer Scientists and Information Technologists on IT Research in Developing Countries, Maputo, Mozambigue: 221.

[24] Charitakis, I., K. Anagnostakis, and E. Markatos. 2003. An active traffic splitter architecture for intrusion detection. In 11th IEEE/ACM International Symposium on Modeling, Analysis and Simulation of Computer Telecommunications Systems (MASCOTS 2003), Orlando, FL: 238–41.

[25] Axelsson, S. 1999. The base-rate fallacy and its implications for the difficulty of intrusion detection. In Proceedings of the 6th ACM Conference on Computer and Communications Security, Singapore: 1–7. and analysis in network intrusion detection. In Proceedings of the 42nd IEEE Conference on Decision and Control, Maui, HI: 3.

[27] Sequeira, D. 2003. Intrusion prevention systems: Security's silver bullet? Business Communications Reviews 33:36–41.

[28] Rash, M., and A. Orebaugh. 2005. Intrusion prevention and active response: Deploying network and host IPs. Syngress. Rockland, MA: Media.

[29] Mattsson, U. 2004. A practical implementation of a real-time intrusion prevention system for commercial enterprise databases. Data Mining V: Data Mining, Text Mining and Their Business Applications, 263–72.

[30] Zhang, X., C. Li, and W. Zheng. 2004. Intrusion prevention system design. In Fourth International Conference on Computer and Information Technology (CIT '04), Wuhan, China: 386–90.

[31] Wilander, J., and M. Kamkar. 2002. A comparison of publicly available tools for static intrusion prevention. In Proceedings of the 7th Nordic Workshop on Secure IT Systems, Karlstad, Sweden: 68–84.

[32] Janakiraman, R. W., and M. Q. Zhang. 2003. Indra: A peer-to-peer approach to network intrusion detection and prevention. In Proceedings of the Twelfth IEEE International Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises (WET ICE 2003), Linz, Austria: 226–31.

[33] Ierace, N., C. Urrutia, and R. Bassett. 2005. Intrusion prevention systems. Ubiquity 6:2.

[34] Fuchsberger, A. 2005. Intrusion detection systems and intrusion prevention systems. Information Security Technical Report 10:134–39.

[35] Schultz, E. 2004. Intrusion prevention. Computers and Security 23:265–66.

Appendix A: Cryptology

[1] Stallings, W. 2006. Cryptography and network security: Principles and practice. Englewood Cliffs, NJ: Prentice Hall.

[2] Ferguson, N., and B. Schneier. 2003. Practical cryptography. New York: John Wiley & Sons.

[3] Enge, A. 1999. Elliptic curves and their applications to cryptography: An introduction. Norwell, MA: Kluwer Academic.

[4] Mollin, R. A. 2001. An introduction to cryptography. Boca Raton, FL: CRC Press.

[5] Cohen, H., G. Frey, and R. Avanzi. 2006. Handbook of elliptic and hyperelliptic curve cryptography. Boca Raton, FL: CRC Press. to and standards. Boston: Artech House.

[7] Wayner, P. 2002. Disappearing cryptography: Information hiding: Steganography and watermarking. San Francisco, CA: Morgan Kaufmann.

[8] Oppliger, R. 2005. Contemporary cryptography. Boston: Artech House.

[9] van Tilborg, H. 2005. Encyclopedia of cryptography and security. New York, NY: Springer.

[10] Mollin, R. A. 2003. RSA and public-key cryptography. London: Chapman & Hall/CRC.

[11] Boneh, D. 2003. Advances in cryptology-crypto 2003. Paper presented at Proceedings of the 23rd Annual International Cryptology Conference, Santa Barbara, CA, August 17–21.