IP ADDRESS MANAGEMENT PRINCIPLES AND PRACTICE

Timothy Rooney

WILEY

IEEE
IEEE PRESS

IEEE Press
Series on
Network
Management
Thomas Plevyak and
Veli Sahin, *Series Editors*

IP ADDRESS MANAGEMENT

IEEE Press
445 Hoes Lane
Piscataway, NJ 08854

Technical Reviewers:

Greg Rabil
Paul Vixie

Books in the IEEE Press Series on Network Management

Telecommunications Network Management Into the 21st Century, edited by Thomas Plevyak and Salah Aidarous, 1994

Telecommunications Network Management: Technologies and Implementations, edited by Thomas Plevyak and Salah Aidarous, 1997

Fundamentals of Telecommunications Network Management, by Lakshmi Raman, 1999

Security for Telecommunications Management Network, by Moshe Rozenblit, 2000

Integrated Telecommunications Management Solutions, by Graham Chen and Quinzheng Kong, 2000

Managing IP Networks: Challenges and Opportunities, edited by Thomas Plevyak and the late Salah Aidarous, 2003

Next-Generation Telecommunications Networks, Services, and Management, edited by Thomas Plevyak and Veli Sahin, 2010

Introduction to IP Address Management, by Timothy Rooney, 2010

IP Address Management: Principles and Practices, by Timothy Rooney, 2011

IP ADDRESS MANAGEMENT

Principles and Practice

Timothy Rooney

IEEE Press Series on Network Management

Thomas Plevyak and Veli Sahin, *Series Editors*

IEEE PRESS

A JOHN WILEY & SONS, INC., PUBLICATION

Library of Congress Cataloging-in-Publication Data:

Rooney, Tim.
IP address management : principles and practice / Tim Rooney.
p. cm.
Includes bibliographical references and index.
ISBN 978-0-470-58587-0 (cloth : alk. paper)
1. Internet addresses. 2. Internet domain names. I. Title.
TK5105.8835.R66 2011
004'67'8–dc22

2010010791

Printed in Singapore

oBook ISBN: 978-0-470-88065-4
ePDF ISBN: 978-0-470-88064-7

10 9 8 7 6 5 4 3 2 1

In memory of my father, Patrick Rooney

CONTENTS

PREFACE

The practice of IP address management (IPAM) entails the application of network management disciplines to Internet Protocol (IP) address space and associated network services, namely Dynamic Host Configuration Protocol (DHCP) and Domain Name System (DNS). The linkages among an IP address plan and configurations of DHCP and DNS servers are inseparable. A change of an IP address affects DNS information and perhaps DHCP as well. These services provide the foundation for today's converged services IP networks, which offer ad hoc anytime, anyplace communications.

If end-user devices such as laptops or voice-over IP (VoIP) phones cannot obtain an IP address via DHCP, they will be rendered unproductive and users will call the help desk. Likewise, if DNS is improperly configured, application navigation by name, phone number, or web address will likewise impair productivity and induce help desk calls.

Effective IPAM practice is a key ingredient in an enterprise or service-provider IP network management strategy. As such, IPAM addresses configuration, change control, auditing, reporting, monitoring, trouble resolution, and related functions as applied to the three foundational IPAM technologies

1. **IP Address Subnetting and Tracking (IPv4/IPv6 Addressing):** Maintenance of a cohesive IP address plan that promotes route summarization, maintains accurate IP address inventory, and provides an automated individual IP address assignment and tracking mechanism. This tracking of individual IP address assignments on each subnet includes those assigned by hard-coding, for example, routers or servers, and others assigned dynamically, for example, laptops and VoIP phones.
2. **DHCP:** Automated IP address and parameter assignment relevant to location and device type. This requires tracking address assignments configured on devices and setting aside dynamically allocated address pools. These address pools can be configured on DHCP servers in order to enable devices to request an IP address, and receive a location-relevant address in reply.
3. **DNS:** Lookup or resolution of hostnames, for example, www entries to IP addresses. This third key aspect of IP address management deals with simplifying IP communications for humans through the use of names, not IP addresses, to establish IP communications. After all, the mapped IP addresses must be consistent with the IP address plan.

The technologies comprising these three core functions are discussed in the first three parts of this book. The practice of IPAM in the fourth part* explains their interrelationships and practices for managing them cohesively. Most IP networks are constantly changing, with the daily demands of the business new stores are opened, offices are closed or moved, companies are acquired, and new devices and device types need IP addresses. These and other changes impacting the IP network can have major repercussions on the existing IP address plan. As the number of users and IP addresses increases, along with the number of subnets or sites, the task of tracking and managing IP address allocations, individual assignments, and associated DNS and DHCP server configurations grows in complexity.

The most common method for performing IPAM functions today entails the use of spreadsheets to track IP addresses, and text editors or Microsoft Windows to configure DHCP and DNS services. As such, IPAM concepts will be demonstrated throughout the book using sample spreadsheet data and configuration file examples as applied to a fictitious organization called IPAM Worldwide, Inc. The intent is to link the technology and configuration details to a real-world example.

CONVENTIONS

This book is typeset in 10-point Times Roman font. *Times Italic* font is used for terms introduced for the first time or to provide emphasis.

To differentiate prose from example configuration information within a DHCP or DNS server, for example, the Courier font in the following manner:

`Courier plain` font: Used to denote keywords or literal text within a configuration file or screen.

`Courier italic` font: Used to denote a parameter name that in practice is substituted for a value reflecting the denoted data element or type.

ORGANIZATION

The book is organized into four parts. The first three parts of the book focuses on each of the three core IPAM aspects, respectively: IP addressing and management, DHCP, and DNS. Part IV then integrates these three core components, describing management techniques and practice.

Part I: IP Addressing. Part I provides a detailed overview of IPv4, IPv6, and IP allocation and subnetting techniques.

Chapter 1: The Internet Protocol. Chapter 1 covers IP (IPv4) from a review of the IP header to classful, classless, and private IP addressing and discusses evolution of Internet

* In actuality, several constituent IPAM practices are discussed in respective technology chapters, though they are summarized in the context of overall practices in Part IV.

Protocol and the development of network address translation and private addressing as key technologies in preserving global IP address space.

*Chapter 2: Internet Protocol Version 6 (IPv6).*Chapter 2 describes the IPv6 header and IPv6 addressing, including address notation, structure, and current IANA allocations. This includes a detailed discussion of each address allocation by type (i.e., reserved, global unicast, unique local unicast, link local, and multicast). Special use addresses, including the solicited node address and the node information query address are also described. The chapter continues with a discussion of the modified EUI-64 algorithm and address autoconfiguration, then concludes with a discussion of reserved subnet anycast addresses and addresses required of IPv6 hosts.

Chapter 3: IP Address Allocation. Chapter 3 discusses techniques for IP block allocation for IPv4 and IPv6 address spaces. This includes coverage of best-fit hierarchical address allocation logic and examples, as well as sparse and random allocation approaches for IPv6. This chapter also discusses unique local address space as well as the role of Internet Registries. Block allocation is an important function of IP address management and it lays the groundwork for configuration of DHCP and DNS services.

Part II: DHCP. Part II provides an overview of DHCP for IPv4 and IPv6 and covers applications that rely on DHCP, DHCP server deployment strategies and DHCP and relevant network access security.

Chapter 4: Dynamic Host Configuration Protocol. Chapter 4 describes the DHCP protocol, including a discussion of protocol states, message formats, options, and examples. A table of standard option parameters with descriptions of each is provided.

Chapter 5: DHCP for IPv6 (DHCPv6). Chapter 5 covers the DHCPv6 protocol, including a comparison with DHCP(v4), message formats, options, and examples. A table of DHCPv6 option parameters is provided.

Chapter 6: DHCP Applications. Building on the previous two technology-based chapters, Chapter 6 highlights the end-user utility of DHCP in describing key applications that rely on DHCP, including VoIP device provisioning, broadband access provisioning, PXE client initialization, and lease limiting.

Chapter 7: DHCP Server Deployment Strategies. DHCP server deployment considerations are covered in Chapter 7, in terms of trading off server sizing, quantities, and locations. DHCP deployment options regarding distributed versus centralized approaches will be discussed, as will redundant DHCP configurations.

Chapter 8: DHCP and Network Access Security. Chapter 8 covers DHCP security considerations as well as discussion of network access security, of which DHCP is a component. A DHCP captive portal configuration example is described as is a summary of related network access control (NAC) approaches, including DHCP-based approaches, switch-based, Cisco NAC, and Microsoft NAP approaches.

Part III: DNS. Part III describes the DNS protocol, DNS applications, deployment strategies and associated configurations, and security, including the security of DNS servers and configurations and DNSSEC.

Chapter 9: The Domain Name System (DNS) Protocol. The opening chapter of Part III, provides a DNS overview, including a discussion of DNS concepts, message details, and protocol extensions. Covered DNS concepts include the basic resolution

process, the domain tree for forward and reverse domains, root hints, local-host domains, and resolver configuration. Message details include the encoding of DNS messages, including the DNS header, label formatting, and an overview of International domain names. DNS Update message formatting is also discussed as is EDNS0.

Chapter 10: DNS Applications and Resource Records. Chapter 10 builds on the material in Chapter 9 to describe key applications, which rely on DNS, including name resolution, services location, ENUM, antispam techniques via black/white listing, SPF, Sender ID, and DKIM. Discussion of applications support is presented in the context of associated resource records.

Chapter 11: DNS Server Deployment Strategies. DNS server deployment strategies and trade-offs are covered in Chapter 11. DNS server deployment scenarios include external DNS, Internet caching, hidden masters/slaves, multimaster, views, forwarding, internal roots, and anycast.

Chapter 12: Securing DNS (Part I). Chapter 12 is the first of two chapters on DNS security. This chapter covers a variety of topics related to DNS security, other than DNSSEC (DNS security extensions), which is covered in its own chapter. Known DNS vulnerabilities are presented first, followed by mitigation approaches for each.

Chapter 13: Securing DNS (Part II): DNSSEC—Chapter 13 covers DNSSEC in detail. The process of creating keys, signing zones, securely resolving names, and rolling keys is discussed, along with an example configuration.

Part IV: IPAM Integration. Part IV brings together the prior three parts, discussing techniques for cohesively managing IP address space, including impacts to DHCP and DNS.

Chapter 14: IP Address Management Practices. In Chapter 14, everyday IP address management functions are described, including IP address allocation and assignment, renumbering, moves, splits, joins, DHCP and DNS server configuration, inventory assurance, fault management, performance monitoring, and disaster recovery. This chapter is framed around the FCAPS network management model, emphasizing the necessity of a disciplined "network management" approach to IPAM.

Chapter 15: IPv6 Deployment and IPv4 Coexistence. The implementation of IPv6 within an IPv4 network will drive a lengthy coexistence of IPv4 and IPv6 protocols. Chapter 15 provides details on coexistence strategies, grouped into sections on dual stack, tunneling approaches, and translation techniques. Coverage includes 6to4, ISATAP, 6over4, Teredo, DSTM, and tunnel broker tunneling approaches and NAPT-PT, SOCKS, TRT, ALG, and bump-in-the-stack or API translation approaches. The chapter concludes with some basic migration scenarios.

Norristown, Pennsylvania TIMOTHY ROONEY
May 2010

ACKNOWLEDGMENTS

First, and foremost, I'd like to thank the following technical reviewers who provided extremely useful feedback, suggestions, and encouragement in the process: Greg Rabil (IPAM and DHCP engineer extraordinaire) and Paul Vixie (Internet guru and President of the Internet Systems Consortium).

I'd like to thank Janet Hurwitz, Alex Drescher, Brian Hart, and Michael Dooley who also provided input and feedback on this book.

I'd also like to thank the following individuals with whom I've had the pleasure to work and from whom I've learned tremendously about communications technologies and IPAM in particular: John Ramkawsky, Steve Thompson, Andy D'Ambrosio, Sean Fisher, Chris Scamuffa, David Cross, Scott Medrano, Marco Mecarelli, Frank Jennings, Jim Offut, Rob Woodruff, Stacie Doyle, Ralph Senseny, and those I've worked with at BT Diamond IP, INS, and Lucent. From my past life at Bell Laboratories, I thank John Marciszewski, Anthony Longhitano, Sampath Ramaswami, Maryclaire Brescia, Krishna Murti, Gaston Arredondo, Robert Schoenweisner, Tom Walker, Ray Pennotti, and especially my mentor, Thomas Chu.

Most of all, I'd also like to thank my family, my wife LeeAnn and my daughters Maeve and Tess, for putting up with my countless hours in writer's isolation and for supporting me throughout this process!

T. R.

PART I

IP ADDRESSING

Part I begins our discussion of the first IPAM cornerstone: IP addressing. This part covers IPv4 and IPv6 protocols as well as address block management techniques.

1

THE INTERNET PROTOCOL

1.1 HIGHLIGHTS OF INTERNET PROTOCOL HISTORY

The Internet Protocol (IP) has changed everything. In my early days at AT&T Bell Laboratories in the mid-1980s when we used dumb terminals to connect to a mainframe, the field of networking was just beginning to enable the distribution of intelligence from a centralized mainframe to networked servers, routers, and ultimately personal computers. Now that I've dated myself, a little later, many rival networking technologies were competing for enterprise deployments with no clear leader. Deployment of disparate networking protocols and technologies inhibited communications among organizations, until during the 1990s the Internet Protocol, thanks to the widespread embrace of the Internet, became the world's de facto networking protocol.

Today, the Internet Protocol is the most widely deployed network layer* protocol worldwide. Emerging from a U.S. government sponsored networking project for the U.S. Department of Defense begun in the 1960s, the Transmission Control Protocol/Internet

* The network layer refers to layer 3 of the Open Systems Interconnect (OSI) seven-layer protocol model. IP is designed for use with Transmission Control Protocol (TCP) or User Datagram Protocol (UDP) at layer 4, the transport layer, hence the term *TCP/IP protocol suite*. The OSI model and IP networking in general are discussed in the book entitled *Introduction to IP Address Management*. (Ref 11)

IP Address Management: Principles and Practice, by Timothy Rooney

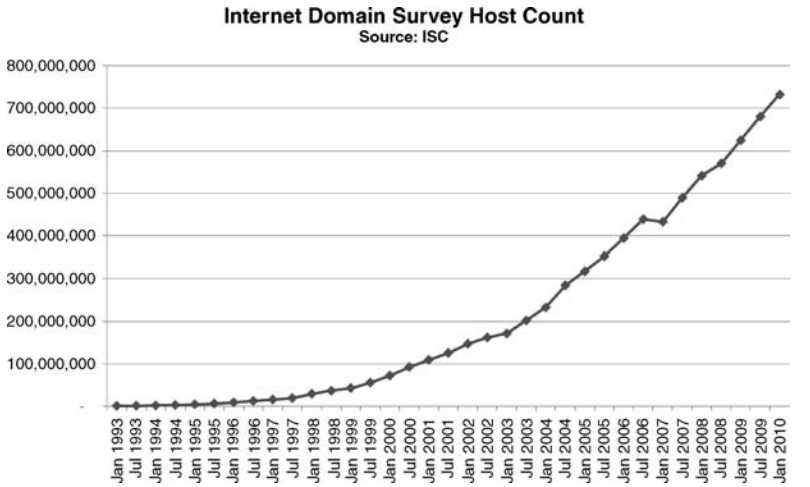

Figure 1.1. Growth of Internet hosts during 1993–2010 (3). *Source:* ISC.

Protocol (TCP/IP) suite has evolved and scaled to support networks from hundreds of computers to hundreds of millions today. In fact, according to Internet Systems Consortium (ISC) surveys, the number of devices or hosts[†] on the Internet exceeded 730 million as of early 2010 with average annual additions of over 75 million hosts *per year* over each of the past 6 years (see Figure 1.1). The fact that the Internet has scaled rather seamlessly from a research project to a network of over 730 million computers is a testament to the vision of its developers and robustness of their underlying technology design.

The Internet Protocol was "initially" defined in 1980 in Request for Comments (RFC[‡]) 760 (1) and 791 (2), edited by the venerable Jon Postel. We quote "initially" because as Mr. Postel pointed out in his preface, RFC 791 is based on six earlier editions of the ARPA (Advanced Research Projects Agency, a U.S. Department of Defense agency) Internet Protocol, though it is referred to in the RFC as version 4 (IPv4). RFC 791 states that the Internet Protocol performs two basic functions: addressing and fragmentation. While this may appear to trivialize the many additional functions and features of the Internet Protocol implemented then and since, it actually highlights the importance of these two major topics for any protocol designer. Fragmentation deals with splitting messages into a number of IP packets so that they can be transmitted over networks that have limited packet size constraints, and reassembly of packets at the destination in the proper order. Addressing is of course one of the key topics of this book, so assuring unique addressability of hosts requiring reachability is critical to basic protocol operation.

[†] The term *host* refers to an end node in the communications path, as opposed to a router or intermediate device. Hosts consist of computers, VoIP telephones, PDAs, and other such IP-addressable devices.

[‡] The Internet Protocol continues to evolve and its specifications are documented in the form of RFCs numbered sequentially. The Internet Engineering Task Force (IETF) is an open community organization with no formal membership and is responsible for publishing RFCs.

The Internet has become an indispensable tool for daily personal and business productivity with such applications as email, social networking, web browsing, wireless access, and voice communications. The Internet has indeed become a key element of modern society. And in case you're interested, the term "Internet" evolved from the lower case form of the term used by the early developers of Internet technology to refer to communications among interconnected networks or "internets."

Today, the capitalized "Internet," the global Internet that we use on a daily basis, has become a massive network of interconnected networks. Getting all of these networks and hosts on them to cooperate and exchange user communications efficiently requires adherence to a set of rules for such communications. This set of rules, this *protocol*, defines the method of identifying each host or endpoint and how to get information from point A to point B over a network. The Internet Protocol specifies such rules for communication using the vehicle of IP packets, each of which is prefixed with an IP header.

1.1.1 The IP Header

The IP layer within the TCP/IP protocol suite adds an IP header to the data it receives from the TCP or UDP transport layer. This IP header is analyzed by routers along the path to the final destination to ultimately deliver each IP packet to its final destination, identified by the destination IP address in the header. RFC 791 defined the IP address structure as consisting of 32 bits comprised of a network number followed by a local address. The address is conveyed in the header of every IP packet. Figure 1.2 illustrates the fields of the IP header. Every IP packet contains an IP header, followed by the data contents within the packet, including higher layer protocol control information.

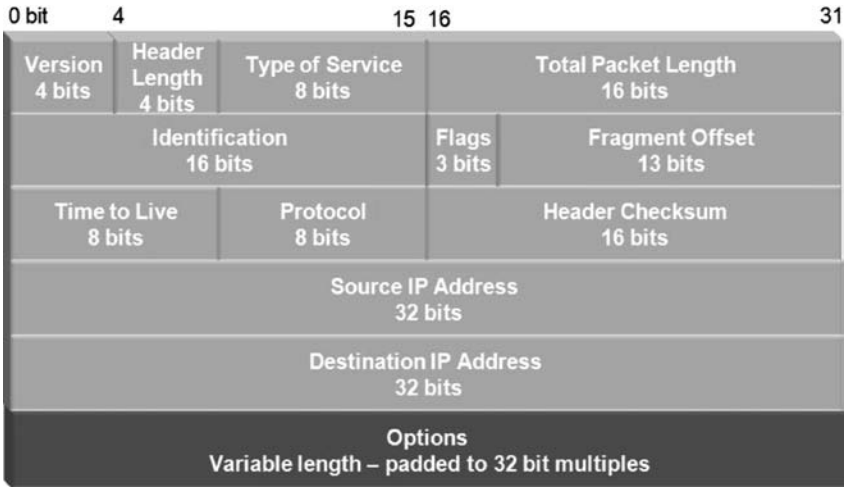

Figure 1.2. IPv4 header fields (1).

Version. The Internet Protocol version, 4 in this case.

Header Length (Internet Header Length, IHL). Length of the IP header in 32-bit units called "words." For example, the minimum header length is 5, highlighted in Figure 1.2 as the lightly shaded fields, which consists of 5 words $\times$ 32 bits/word = 160 bits.

Type of Service. Parameters related to the packet's quality of service (QoS). Initially defined as ToS (type of service), this field consisted of a 3-bit precedence field to enable specification of the relative importance of a particular packet, and another 3 bits to request low delay, high throughput, or high reliability, respectively.

The original ToS field has been redefined via RFC 2474, "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Header" (177). The DS field, or differentiated services field, provides a 6-bit code point (DSCP, differentiated services code point) field with the remaining 2 bits unused. The code point maps to a predefined service, which in turn is associated with a level of service provided by the network. As new code points are defined with respective services treatment by the Internet authorities, IP routers can apply the routing treatment corresponding to the defined code point to apply higher priority handling for latency-sensitive applications, for example.

Total Length. Length of the entire IP packet in bytes (octets).

Identification. Value given to each packet to facilitate reassembly of packet fragments at the receiving end.

Flags. This 3-bit field is defined as follows:

- Bit 0 is reserved and must be 0.
- Bit 1—Don't Fragment—indicates that this packet cannot be fragmented.
- Bit 2—More Fragments—indicates that this packet is a fragment, though this is not the last fragment.

Fragment Offset. Identifies the location of this fragment relative to the beginning of the original packet in units of 64-bit "double words."

Time to Live (TTL). A counter decremented upon each routing hop; once the TTL reaches zero, the packet is discarded. This parameter prevents packets from circulating on the Internet forever!

Protocol. The upper layer protocol that shall receive this packet after IP processing, for example, TCP or UDP.

Header Checksum. A checksum value calculated over the header bits only to verify that the header is not corrupted.

Source IP Address. The IP address of the sender of this packet.

Destination IP Address. The IP address of the intended recipient of this packet.

Options. Optional field containing zero or more optional parameters that enable routing control (source routing), diagnostics (trace route, maximum transmission unit (MTU) discovery), and more.

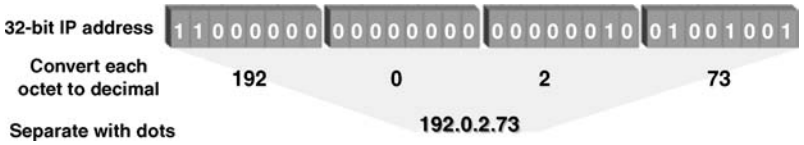

Figure 1.3. Binary to dotted decimal conversion.

It's ok if you find this IP header detail a bit droll. It's only to provide some context, but now let's focus our attention to the source and destination IP address fields and the IP addressing structure.

1.2 IP ADDRESSING

The IP address field is comprised of 32 bits. The familiar dotted decimal notation for an IP address reflects the splitting of the 32-bit address into four 8-bit octets. We convert each of the four octets to decimal, and then separate them with decimal points or "dots." This is certainly easier than calculating these 32 bits as one huge number! Consider the 32-bit IP address in Figure 1.3. We simply split this into four octets, convert each octet to decimal, and then separate the decimal representation of each octet by "dots." Hence, the term "dotted decimal."

1.2.1 Class-Based Addressing*

RFC 791 (2) defines three classes of addresses: classes A, B, and C. These classes were identified by the initial bits of the 32-bit address as depicted in Figure 1.4. Each class corresponded to a particular fixed size for the network number and local address fields. The local address field could be assigned to individual hosts or further broken down into subnet and host fields, as we'll discuss later.

The division of address space into classes provided a means to easily define different sized networks for different users' needs. At the time, the Internet was comprised of certain U.S. government agencies, universities, and some research institutions. It had not yet blossomed into the de facto worldwide backbone network it is today, so address capacity was seemingly limitless. The other reason for dividing address space into classes on these octet boundaries was for easier implementation of network routing. Routers could identify the length of the network number field simply by examining the first few bits of the destination address. They would then simply look up the network number portion of the entire IP address in their routing table and route each packet accordingly. Computational horsepower in those days was rather limited, so minimizing processing requirements was another consideration. A side benefit of classful addressing was simple readability. Each dotted decimal number represents one octet in binary. As we'll see later when discussing classless addressing, this is not typically the case today.

* Much of the remainder of this chapter leverages material from Chapter 2 of Ref. 11.

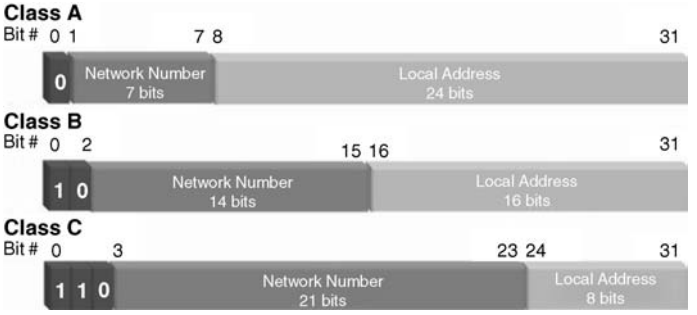

Figure 1.4. Class-based addressing.

Examining this class-based addressing structure, we can observe a few key points:

- Class A networks
  - Class A prefixes begin with binary 0 ($[0]_2$)[†] plus 7 additional bits or 8 network bits total.
  - The network address of all 0s is invalid.[‡]
  - The network address of $[01111111]_2 = 127$ is a reserved address. Address 127.0.0.1 is used for the "loopback address" on an interface.
  - This leaves us with a class A network prefix range of $[00000001]_2$ to $[01111110]_2 = 1$–126 as the first octet.
  - The local address field is 24 bits long. This equates to up to $2^{24} = 16{,}777{,}216$ possible local addresses per network address. Generally, the all 0s local address represents the "network" address and the all 1s is a network broadcast, so we typically subtract these two addresses from our local address capacity in general to arrive at 16,777,214 hosts per class A network. Thus, 10.0.0.0 is the network address of 10.0.0.0/8, and 10.255.255.255 is the broadcast address to all hosts on the 10.0.0.0/8 network.
- Class B networks
  - Class B networks begin with $[10]_2$ plus 14 additional bits or 16 network bits total.
  - The range of class B network prefixes in binary is $[10000000\ 00000000]_2$ to $[10111111\ 11111111]_2$ or networks in the range of 128.0.0.0 to 191.255.0.0, yielding 16,384 network addresses.
  - The local address field is 16 bits long for $65{,}536 - 2 = 65{,}534$ possible hosts per class B network.

[†] To differentiate a binary 0 (1 bit) from a decimal 0 (7–8 bits) in cases where it may be ambiguous, we subscript the number with the appropriate base. Don't worry; we're not digressing into chemistry with discussion of oxygen molecules with the 0_2 notation, simply "zero base 2."

[‡] Though some protocols such as DHCP use the all 0s address as a placeholder for "this" address.

- Class C networks
  - Class C networks begin with $[110]_2$ plus 21 additional bits or 24 network bits total.
  - The range of class C network prefixes is $[11000000\ 00000000\ 00000000]_2$ to $[11011111\ 11111111\ 11111111]_2$ or networks in the range 192.0.0.0 to 223.255.255.0, yielding 2,097,152 networks.
  - The local address field is 8 bits long for $256 - 2 = 254$ possible hosts per class C network.
- Class D networks (not illustrated in Figure 1.4)
  - Class D networks were defined after RFC 791 and denote multicast addresses, which begin with $[1110]_2$. Multicast is used for streaming applications where multiple users or subscribers receive a set of IP packets from a common source. In other words, multiple hosts having a common multicast address would receive all IP traffic sent to the multicast group or address. There is no network and host portion of the multicast network as members of a multicast group may reside on many different physical networks.
  - The range of class D networks is from $[11100000\ 00000000\ 00000000\ 00000000]_2$ to $[11101111\ 11111111\ 11111111\ 11111111]_2$ or the 224.0.0.0 to 239.255.255.255 range, yielding 268,435,456 multicast addresses.
- Class E networks (not illustrated in Figure 1.4)
  - Networks beginning with $[1111]_2$ (class E) are reserved.

1.2.2 Internet Growing Pains

With seemingly limitless IP address capacity, at least as it seemed through the 1980s, class A and B networks were generally allocated to whomever asked. Recipient organizations would then subdivide or subnet* their class A or B networks along octet boundaries within their organizations. Keep in mind that every "network," even within a corporation, needed to have a unique network number or prefix to maintain address uniqueness and maintain route integrity.

Subnetting provides routing boundaries for communications and routing protocol updates. Each network over which IP packets traverse requires its own IP network number (network address). As more and more companies sought to participate in the Internet by requesting IP address space, Internet Registries, the organizations responsible for allocating IP address space, were forced to throttle address allocations. Those requesting IP address space from Internet Registries soon faced increasingly stringent application requirements and were granted a fraction of the address space requested. In having to make do with smaller network block allocations, many organizations were forced to subnet on nonoctet boundaries.

Whether on octet boundaries or not, subnetting is facilitated by specifying a *network mask* along with the network address. The network mask is an integer number

*The term *subnet* is frequently used as a verb as in this context, to mean the act of creating a subnet.

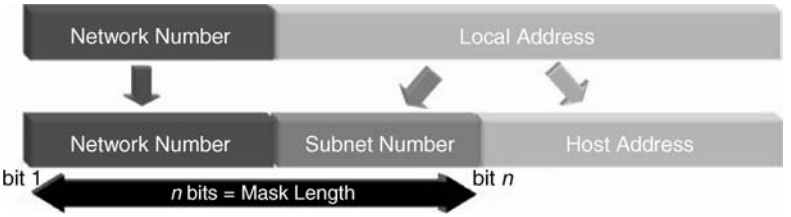

Figure 1.5. Subnetting provides more "networks" with fewer hosts per network.

representing the length in bits of the network prefix. This is sometimes also referred to as the mask length. For example, a class A network has a mask length of 8, a class B of 16, and C of 24. By essentially extending the length of the network number that routers need to examine in each packet, a larger number of networks can be supported, and address space can be allocated more flexibly. This is illustrated in Figure 1.5.

Routers need to be configured with this mask length for each subnet that they serve. This allows them to "mask" the IP address, for example, to expose only the indicated network and subnet bits within the 32-bit IP address to enable efficient routing without relying on address class. Based on this extended network number, the router can route the packet accordingly.

The network address and mask length were originally denoted by specifying the 32-bit mask in dotted decimal notation. This notation is derived by denoting the first n bits of a 32-bit number as 1s and the remaining $32 - n$ bits as 0s, and then converting this to dotted decimal.

For example, to denote a network mask length of 19 bits, you would

- create the 32-bit number with 19 1s and 13 0s: 11111111111111111100000000000000
- separate into octets: 11111111.11111111.11100000.00000000
- convert to dotted decimal: 255.255.224.0

For example, the notation for network 172.16.168.0 with this 19-bit mask is 172.16.168.0/255.255.224.0.

Thankfully, this approach was superseded by a simpler notation: the mask is now denoted with the network address as <network address>/<mask length>. While the notation is easier to read, it does not save us from the equivalent binary exercise! For example, the 172.16.0.0 class B network would be represented as 172.16.0.0/16. The "slash 16" indicates that the first 16 bits, in this case the first two octets, represent the network prefix.

Here's the binary representation of this network:

Network Address	*Network Prefix*	Local Address
`172.16.0.0/16`	*`10101100 00010000`*	`00000000 00000000`

Let's subnet this network using a 19-bit mask. Expanding this out into binary notation:

Network Address	*Network Prefix*	***Subnet*** Local Address
172.16.0.0/19	*10101100 00010000*	***000***00000 00000000
172.16.32.0/19	*10101100 00010000*	***001***00000 00000000
172.16.64.0/19	*10101100 00010000*	***010***00000 00000000
172.16.96.0/19	*10101100 00010000*	***011***00000 00000000
172.16.128.0/19	*10101100 00010000*	***100***00000 00000000
172.16.160.0/19	*10101100 00010000*	***101***00000 00000000
172.16.192.0/19	*10101100 00010000*	***110***00000 00000000
172.16.224.0/19	*10101100 00010000*	***111***00000 00000000

Notice that the class B network bits are depicted under the Network Prefix column in italic font, and we highlighted the subnet bits in larger bold italic font in the Subnet column. Using this 3-bit subnet mask, we effectively extended the network number from 16 bits to 19. By incrementing the binary values of these 3 bits from $[000]_2$ to $[111]_2$ as per the highlighted subnet bits above, we can derive $2^3 = 8$ subnets with this 3-bit subnet mask extension. Routers would then be configured to route using the first 19 bits to identify the network portion of the address by configuring the router serving such a subnet with the corresponding mask length, for example, 172.16.128.0/19, and then having the router communicate reachability to this network via routing protocols. This technique, called variable length subnet masking (VLSM), became increasingly more prevalent in helping to squeeze as much IP address capacity as possible out of the address space assigned within an organization.

The two-layer network/subnet model worked well during the first decades of IP's existence. However, in the early 1990s, demand for IP addresses continued to increase dramatically, with more and more companies desiring IP address space to publish web sites. At the then current rate of usage, the address space was expected to exhaust before the turn of the century! The guiding body of the Internet, the Internet Engineering Task Force, cleverly implemented two key policies to extend the usable life of the IP address space, namely, support of private address space [ultimate RFC 1918 (7)] and classless interdomain routing [CIDR, RFCs 1517–1519 (Ref. 4–6)]. The IETF also began work on a new version of IP with enormous address space during this time, IP version 6, which we'll discuss in the next chapter.

1.2.3 Private Address Space

Recall our statement that every "network" within an organization needs to have a unique network number or prefix to maintain address uniqueness and route integrity. As more and more organizations connected to the Internet, the Internet became a potential vehicle for hackers to infiltrate organizations' networks. Many organizations implemented firewalls to filter out IP packets based on specified criteria regarding IP header values, such as source or destination addresses, UDP versus TCP, and others. This guarded

partitioning of IP address space between "internal" and "external" address spaces dovetailed nicely with address conservation efforts within the IETF.

The IETF issued a couple of RFC revisions, resulting in RFC 1918 becoming the standard document that defined the following sets of networks as "private":

- 10.0.0.0—10.255.255.255 (10/8 network)—equivalent to 1 class A.
- 172.16.0.0—172.31.255.255 (172.16/12 network)—equivalent to 16 class B's.
- 192.168.0.0—192.168.255.255 (192.168/16 network)—equivalent to 1 class B or 256 class C's.

The term *private* means that these addresses are not routable on the Internet. However, within an organization, they may be used to route IP traffic on internal networks. Thus, my laptop is assigned a private IP address and I can send emails to my fellow associates, who also have private addresses. My organization in essence has defined a private Internet, sometimes referred to as an intranet. Routers within my organization are configured to route among allocated private IP networks, and the IP traffic among these networks never traverses the Internet.*

Since I'm using a private IP address, someone external to the organization, outside the firewall, cannot reach me directly. Anyone externally sending packets with my private address as the destination address in the IP header will not be able to reach me as these packets will not be routed by Internet routers. But what if I wanted to initiate a connection externally to check on how much money I'm losing in the stock market via the Internet? For employees requiring access to the Internet, firewalls employing network address translation (NAT) functionality are commonly employed to convert an enterprise user's private IP address into a public or routable IP address from the corporation's public address space.

Typical NAT devices provide address pooling features to pool a relatively small number of publicly routable (nonprivate) IP addresses for use on a dynamic basis by a larger number of employees who sporadically access the Internet. The NAT device bridges two IP connections together: the internal-to-NAT device communications utilize private address space, while the NAT device-to-Internet communications use public IP addresses. The NAT device is responsible for keeping track of mapping the internal employee address to the public address used externally.

This is illustrated in Figure 1.6, with the internal network utilizing the 10/8 address space and external or public addressing utilizing the 192.0.2.0/24 space. As per the figure, if my laptop has the IP address 10.1.0.1, I can communicate to my colleague on IP address 10.2.0.2 via the internal IP network. When I access the Internet, my packets need to be routed via the firewall/NAT device in order to map my private 10.1.0.1 address to a public address, for example, 192.0.2.108. The mapping state is maintained in the NAT device and it modifies the IP header to swap out 10.1.0.1 for 192.0.2.108 for outbound packets and the converse for inbound packets.

* Technically, with the use of virtual private networks (VPNs) or tunnels over the Internet, privately addressed traffic may traverse the Internet, but the tunnel endpoints accessing the Internet on both ends do utilize public IP addresses.