NETWORK AND CYBER SECURITY

Ajay Shriram Kushwaha
Dr. Ramkumar Krishnamoorthy

BOOKS ARCADE

Network and Cyber Security

Network and Cyber Security

Ajay Shriram Kushwaha
Dr. Ramkumar Krishnamoorthy

BOOKS ARCADE
KRISHNA NAGAR, DELHI

Network and Cyber Security

Ajay Shriram Kushwaha
Dr. Ramkumar Krishnamoorthy

BOOKS ARCADE

Regd. Office:
F-10/24, East Krishna Nagar, Near Vijay Chowk, Delhi-110051
Ph. No: +91-11-79669196, +91-9899073222
E-mail: info@booksarcade.co.in, booksarcade.pub@gmail.com
Website: www.booksarcade.co.in

Year of Publication 2023

International Standard Book Number-13: 978-81-19199-27-3

CONTENTS

CHAPTER 1

INTRODUCTION OF NETWORK AND CYBER SECURITY

Ajay Shriram Kushwaha, Associate Professor,
Department of Computer Science and Information Technology, Jain (Deemed to be University) Bangalore, Karnataka, India
Email Id- ks.ajay@jainuniversity.ac.in

Network and cyber security are the practices and technologies used to protect networks, devices, and sensitive information from unauthorized access, use, disclosure, disruption, modification, or destruction. This includes protecting against cyber-attacks, such as hacking and malware, as well as natural disasters and human error. Network security involves the use of firewalls, encryption, and security protocols to protect the integrity and confidentiality of data transmitted over a network. Cyber security, on the other hand, encompasses a broader range of threats, including those related to the Internet and the cloud. It includes practices such as incident response, risk management, and compliance.

Network security focuses on protecting the infrastructure of the network, such as routers, switches, and servers, from unauthorized access and attacks. It includes practices such as securing the network perimeter with firewalls, implementing secure protocols for data transmission, and using virtual private networks (VPNs) to encrypt communications. Network security also includes monitoring and analyzing network traffic for unusual activity and implementing intrusion detection and prevention systems. Cyber security, on the other hand, is a broader term that encompasses the protection of all types of devices, including computers, smartphones, and IOT devices, as well as the data and applications that run on them. It includes practices such as implementing strong authentication and access controls, regularly patching and updating software, and conducting regular security audits and risk assessments. Cyber security also includes incident response and disaster recovery planning to ensure that organizations can quickly and effectively respond to security breaches and minimize the impact of any disruption. Network and Cyber Security are essential for protecting an organization's sensitive information and maintaining the availability and integrity of its systems. Some additional practices and technologies that are often used in network and cyber security include:

Antivirus and anti-malware software: These tools are used to detect and remove malware, such as viruses, Trojans, and ransomware, from devices and networks.

Security information and event management (SIEM): These systems collect and analyze log data from various devices and systems to detect and alert potential security threats.

Endpoint protection: This includes the use of endpoint security software and mobile device management (MDM) to protect and secure devices such as laptops, smartphones, and tablets.

Content filtering: This is the use of software and hardware to block access to certain websites or types of content, such as sites that host malware or are known to be used for phishing attacks.

Cloud security: With more and more organizations moving their data and applications to the cloud, securing these cloud-based resources has become a critical aspect of cyber security. This includes the use of encryption and secure protocols, as well as implementing security controls within the cloud infrastructure itself.

Security awareness training: Employee education and training is a vital part of any security program, as it helps to ensure that everyone in the organization understands the risks and knows how to recognize and respond to potential security threats.

All of these practices and technologies are essential for protecting networks and devices from cyber threats. However, no single solution or technology can completely protect against all threats. Organizations need to adopt a comprehensive and layered approach to security that involves multiple layers of defense and continuous monitoring, assessment, and improvement.

Additional security practices and technologies that organizations can use to protect their networks and devices include:

Two-factor authentication (2FA): This adds a layer of security by requiring users to provide two forms of identification, such as a password and a fingerprint or a token before they can access sensitive data or systems.

DDoS Mitigation: Distributed Denial of Service (DDoS) is a type of attack that floods a network or website with traffic from multiple sources, making it difficult or impossible for legitimate users to access it. DDoS mitigation solutions can help detect and mitigate these types of attacks in real time.

Vulnerability management: This involves identifying and assessing vulnerabilities in systems and networks, and then taking steps to remediate them. Vulnerability management includes vulnerability scanning and penetration testing.

Identity and Access Management (IAM): IAM is the practice of managing and controlling access to systems and networks based on users' roles and responsibilities. This includes creating and managing user accounts, assigning permissions, and controlling access to sensitive data.

Data Loss Prevention (DLP): DLP is the practice of identifying, monitoring, and preventing the unauthorized transfer of sensitive data. This can include monitoring network traffic for sensitive data, encrypting sensitive data at rest and in transit, and implementing controls to prevent data leakage.

Compliance: Organizations may be subject to various regulatory requirements, such as HIPAA, SOC2, PCI-DSS, etc. Compliance with these regulations requires specific security controls and regular testing and audits.

As with any security measure, it's important for an organization to regularly review and update its security practices and technologies to keep up with the evolving threat landscape. Additionally, security should be integrated into all aspects of the organization, not just IT, it should be a part of the culture and should be an ongoing process.

History of Internet: The history of the internet can be traced back to the late 1950s when the United States Department of Defense began developing a communication network that would allow its various research projects to share resources and communicate with one another. This project, known as the ARPANET, was the precursor to the modern internet. The first successful transmission over the ARPANET occurred in October 1969, and by the end of the year, four universities were connected to the network. In the 1970s, the development of the internet was taken over by a group of researchers and engineers known as the Internetworking Working Group (IWG). They developed a set of protocols, known as TCP/IP that allowed different networks to communicate with one another. This allowed for the creation of a global network that connected government, academic, and commercial organizations.

The 1980s saw the emergence of new technologies such as the Domain Name System (DNS) and the Simple Mail Transfer Protocol (SMTP), which greatly improved the usability of the internet. In the late 1980s and early 1990s, the World Wide Web (WWW) was created, which allowed for the easy sharing of multimedia content and documents.

The commercialization of the internet began in the 1990s, with the emergence of companies such as AOL, CompuServe, and Prodigy, which offered dial-up internet access to consumers. The first web browsers, such as Mosaic and Netscape, also appeared during this time, which made it easier for people to access the web.

In the early 2000s, the introduction of high-speed internet access and the widespread use of mobile devices greatly increased the number of people using the internet. Today, the internet is a global network that connects billions of people, devices, and organizations and it has become an essential part of modern life, used for communication, commerce, entertainment, education, and much more.

In the late 2000s and early 2010s, the emergence of social media platforms such as Facebook, Twitter, and LinkedIn, changed the way people interact and share information online. These platforms have also played a significant role in the development of the sharing economy, which allows individuals to share resources such as cars and homes.

The rise of cloud computing has also been a major development in the internet's history. Cloud computing allows companies and individuals to store and access data and applications over the internet, without the need for expensive hardware and software. This has greatly increased the accessibility and scalability of technology for businesses of all sizes.

The 2010s also saw the explosion of mobile internet usage, driven by the rapid adoption of smartphones and tablets. Mobile devices have allowed people to access the internet from anywhere, at any time, and have played a major role in the development of new technologies such as mobile commerce and mobile banking. The internet of things (IoT) is another key development in the history of the internet. It refers to the growing network of everyday devices, such as appliances, vehicles, and medical equipment that are connected to the internet and can communicate with one another. The IOT has the potential to greatly increase efficiency and automation across a wide range of industries.

Overall, the internet has undergone a rapid evolution since its inception in the late 1950s and continues to evolve as new technologies and applications are developed. Today, the internet plays a vital role in many aspects of our lives and has opened up new opportunities for communication, commerce, and innovation.

Information Security Protocols: Given the susceptibility of computer networks and the risks both known and unknown that these systems confront from an unpredictably skewed user base. While it is difficult to be aware of every potential sort of assault on computer networks, we have attempted to describe and classify these attacks and how they impact the target computer network systems based on what is presently known. The security measures and best practices that may be employed to safeguard a corporate network will, however, be our main emphasis.

A strong firewall regime, strong cryptographic systems, authentication and authorization, intrusion detection, vigilant virus detection, legislation, regulation, self-regulation, moral and ethics education, and a number of other protocols and best practises are worth investing in when it comes to securing networks or cyberspace in general.

An effective security plan: A security policy, according to RFC 2196, is a written declaration of the guidelines that those who are granted access to an organization's technology and

information assets must follow. The specifics of a security policy define how strong a system's security is for an organisation. The security policy is the instrument that asserts no when it is necessary. The need to say no arises from the system administrator's need to restrict the usage of network computers, resources, and capabilities in order to maintain the system's security. Implementing a set of rules, regulations, and guidelines that explain to all staff members and business partners what constitutes acceptable and inappropriate usage of the company's computer system is one approach to do this equitably. The organization's security policy is made up of these rules, regulations, and policies.

The security policy specifies which resources must be secured as well as how the company may do so.

A security policy is a dynamic collection of rules and guidelines that affect and may place restrictions on the freedoms and, naturally, degrees of personal security accountability of all users. The security of an organisation depends on a structure like this. Yet, some individuals in the security sector are not big fans of security policies. We think security rules are crucial to a system's overall security strategy for a number of reasons, including:

Firewall setups: A firewall's rule base has to be built on a reliable security policy in order for it to be configured to work.

User control: While accessing a network, such as the Internet, via a firewall, all users inside an organisation are required to follow the security policy.

Without a strict security policy to which every employee must adhere, the business may lose data and employee productivity as a result of workers spending time, among other things, patching security holes and addressing vulnerabilities and retrieving lost or compromised data.

The security policy should be adaptable enough to let each employee the level of access they need to do their assigned activities; complete access should only be given to those whose jobs require it. As a general rule, both workers and employers should be informed completely about the access policy. There should be absolutely no misunderstanding.

A good security policy, according to Mani Subramanian, should: Identify what needs to be protected; decide which items need to be protected from authorized access, unauthorized or unintended disclosure of information, and denial of service; determine the likelihood of attack; Implement the most effective protection; and review the policy continuously and update it if weaknesses are found.

According to Merike Kaeo, a security policy should be able to be technically implemented, organizationally implemented, enforced with security tools when necessary, and enforced with sanctions when prevention is not technically possible. It should also clearly define the areas of responsibility for users, administrators, and management. Finally, it should be flexible and adaptable to changing environments.

A security policy addresses a broad range of subjects and fulfils a number of crucial tasks in the system security cycle. Building a security strategy has many diverse parts that must all work together, much like constructing a home. Each step in the development of the security policy adds value to the final product, making it distinctive to the firm. A security policy needs the support of the organization's senior management in order to be effective, and it also has to clearly state everyone's position and responsibility in the organization's security.

Decide which goods should be safeguarded and how much it will cost. Provide as a useful teaching tool for everyone in the company on security, the things that need to be safeguarded,

and the reasons for and methods for doing so. Establish guidelines for what conduct is acceptable and unacceptable in terms of the security and privacy of the organization's resources. Establish a clearinghouse and authority for security. Possess the flexibility to adjust to new developments.

Be applied uniformly throughout the whole company. Jasma advises doing the essential steps 4 to accomplish all of those: Identify the resources that need to be safeguarded and create a profile of each resource's characteristics. Physical, logical, network, and system assets should all be included among these resources. They should be organised in a table in order of priority. Decide who you must defend each resource from for each one that has been identified. Determine the kinds of possible risks and the probability of such threats for each resource that has been identified. Threats include information disclosure, information alteration, and illegal access. Create a table ranking the security risks for each threat in order of priority. Create a policy team with at least one representative from senior administration, legal, frontline staff, and the IT division. Recruit a writer or editor to assist with the policy's writing as well.

Identify the areas that need auditing. Audit systems, including security events on servers, firewalls, and specific network hosts, using tools like Tripwire. Auditable logs on servers, firewalls, and certain network hosts include logfiles and object accesses.Determine how to handle encryption, passwords, key production and distribution, and wireless devices that connect to the organization's network. Describe appropriate usage of system resources including email, news, and the Web. Provide remote access to accommodate mobile employees, individuals working from home, and potential VPN connections from business partners. Using all of this data, create two structures, one outlining user access rights to the identified resources and the other outlining user duties related to maintaining the security of a certain resource.

Finally, a sound security strategy has to include the following elements:

1. A matrix of access permissions for security policies.
2. Limitations on logical access to system resources.
3. Resource and site environment physical security.
4. Restrictions imposed by cryptography.
5. Rules and regulations.
6. Typical assaults and potential deterrents.
7. A skilled labour force.
8. Certification of equipment.
9. Audit trails and archival documentation.
10. Privacy issues.
11. Instruction on security awareness.
12. Handling of incidents.
13. Vulnerability Evaluation

A system's vulnerabilities are found, tracked, and managed via a regular procedure called vulnerability assessment. A vulnerability assessment performs a system check-up. It is a crucial security procedure and recommended practise for the system's health. Depending on the organisation, different objects are verified in this procedure.

All PCs, servers, routers, and firewalls could be included. System administrators may often get the following information from vulnerability assessment services:

1. Network mapping and system fingerprinting of all known vulnerabilities.
2. A thorough analysis of vulnerabilities with a rating of all exploitable flaws based on probability and possible effect for all services on each host.

3. A ranking of incorrect settings.
4. A final report is always generated at the conclusion of the process, including the findings and the recommended course of action for addressing such vulnerabilities. According to the organization's operating plan, this report also includes suggestions for further system reassessments at predetermined intervals or on a regular basis. The recommendations are arranged in priority order for reducing or removing deficiencies.

Vulnerability assessment has become a highly popular security technique due to its importance, and as a consequence, a flurry of software solutions have been developed to suit the demand. Due to the proliferation of security assessment companies, the practise has also resulted in a high degree of knowledge in the field. Nevertheless, trust is a problem since there are so many of these businesses. Therefore, it is suggested that a system administrator occasionally hire an outsider to acquire a more unbiased perspective. The perimeter and internal systems of a private computer network are often the focus of security assessment services, which also include application evaluation.

Velocities Scanning: Automated system and network vulnerability scanning involves sending network traffic to all or specific network computers with the expectation of receiving back traffic that will reveal if those machines have known vulnerabilities. Operating systems, application software, and protocols may all have flaws that make them vulnerable. The vulnerability scanning services aim to identify critical security vulnerabilities and gaps in the current system's security practises because vulnerability scanning is designed to give a system administrator a thorough security review of the system, including both the perimeter and system internals. Comprehensive system scanning often yields a lot of false positives and false negatives due to the precision that is required and sought after by these services. The system administrator's responsibility is to come up with strategies for handling these false positives and negatives. Each scan's final report includes prioritised suggestions and strategic guidance to make sure the most important gaps are filled first. Depending on the severity of the required scan, system users or service providers may schedule system scanning to run periodically and send frequent automated emails to a selected user. The scans may also be kept for later examination on a safe server.

Three versions of vulnerability scanning have been developed to date. The first generation needed compilation and execution for certain hardware or platforms of either code or script, which were often obtained from the Internet or completely disseminated. They constantly required upgrades to comply with requirements for newer technologies since their code and scripts were hardware- and platform-specific. Due to these drawbacks, the second generation emerged, which was more potent and sophisticated and offered more thorough and in-depth reports. Tools were able to scan a variety of hardware and operating systems and isolate tests for certain vulnerabilities. This was a significant advancement. They were, however, insufficiently detailed and often provided false positives and negatives.

The third generation included a double, and sometimes triple, scan of the same network resources in an effort to decrease erroneous reports. It conducted successive scans for vulnerabilities using information from the first scan. This was a huge improvement since those extra checks often turned up second-level vulnerabilities, which are datagram vulnerabilities that are more serious. If not identified and fixed in a timely manner, the second-level vulnerabilities are successfully exploited by hackers when information from less secure servers is utilised to target additional system servers, leading to cascading network flaws.

System screening for network vulnerabilities has advantages and disadvantages:

Both system burglars and system security directors may utilise it efficiently to create an electronic inventory of the network. The scanner swiftly locates security flaws in the network as it continually scans it, and it produces reports detailing the security holes' nature and locations. Both internal and external attackers may use the information in the electronic inventory to access the network, and the system security team may use it to close any gaps that have been found. As a result, vulnerability scanning offers the following advantages to the network security team:

It pinpoints network flaws, including their kinds and locations. The security team must close the found security gaps.

Network security administrators can quickly and thoroughly test operating system privileges and permissions, the main cause of network vulnerabilities, test compliance with company policies, the most likely source of network security intrusions, and finally set up a continuous monitoring system once they have the electronic network security inventory. When there are fewer and less significant security breaches, maintenance costs are reduced and the concern of data loss is lessened. After these steps are implemented, it may result in fewer security breaches, enhancing consumer trust.

Tools Used for Scanning: Today's market is flooded with scripts and tools for network security scanning. When applied appropriately, each of these tools will uncover various vulnerabilities. It is difficult for any one vulnerability tool or script to be helpful for a vast collection of system vulnerabilities as network technology develops, along with the changing terrain of assaults, the advancements in virus production, and other attack tools. As a result, the majority of security specialists combine various tools and scripts for maximum effectiveness. The most often used instruments typically include 140 options that are carefully utilised to alter the tool's sensitivity or target the tool to concentrate the scan.

We will examine the most recent tools and scripts for commercial vulnerability scanners. They are separated into two groups: host-based and network-based. The goal of network-based tools is to protect the whole network, and they do this by scanning the network for various vulnerabilities. In addition to servers, routers, firewalls, and locally located facilities, they check all Internet resources. Host-based scanning focuses on a single host that is thought to be susceptible since a significant portion of network security risk originates from inside sources, such as workers. To scan the machine's hardware and operating system, a host installation is necessary. The scanner looks for missing security checks, susceptible service setups, lax password rules, and weak or subpar passwords at the operating system level.

Nmap, a network port: Information Security Protocols and Best Practices 129 scanning tool for single hosts and small and large networks, is one of the most widely used scanners available today. Nmap supports many scanning techniques including Vanilla TCP connect, TCP SYN (half \sopen), TCP FIN, Xmas or NULL, TCP FTP proxy (bounce attack), SYN/ \sFIN, IP fragments, TCP ACK and Windows, UDP raw ICMP port unreachable, ICMP (ping-sweep), TCP ping, direct (non-portmapper) RPC, remote OS identification by TCP/IP fingerprinting, and reverse- identity scanning.

When completely set up, N-map may do decoy scans using any set of TCP addresses the user chooses. N-map can also mimic a coordinated scan to simultaneously target several networks in one or more nations. Moreover, it has the ability to cover its tracks by launching a barrage of what a user or system administrator would mistake for global assaults. It may disperse its assaults in order to remain undetected below a monitoring threshold that the system administrator or the system security team has established. N-map is particularly good at

detecting the kinds of machines operating in a targeted network and the services that could be susceptible on each one of them.

Penetration Testing and Vulnerability Assessment: Another crucial stage of system security vulnerability evaluation is penetration testing and vulnerability assessment. It should be carried out in a complete, thorough, and exhaustive manner. It aims to test both known and unknown vulnerabilities in the system. This step involves testing all currently available hacking methods and tools to simulate actual attack situations. At this stage of thorough real-world system testing, hidden vulnerabilities are sometimes discovered, attack methods are recognised, and the origins and severity of vulnerabilities are classified and prioritized according to the risks that the user has specified.

Application Evaluation: As the quantity of services offered by computer network systems explodes, there are correspondingly greater needs for system application software, as well as for application automation and new dynamism of these applications. A new security paradigm in system management has emerged as a result of the dynamism offered in application software. Many businesses are becoming more aware of these risks and are making significant strides in securing their systems against intrusions by web-based apps. As a result, evaluating the security of system applications is turning into a specialized competence required to safeguard key systems.

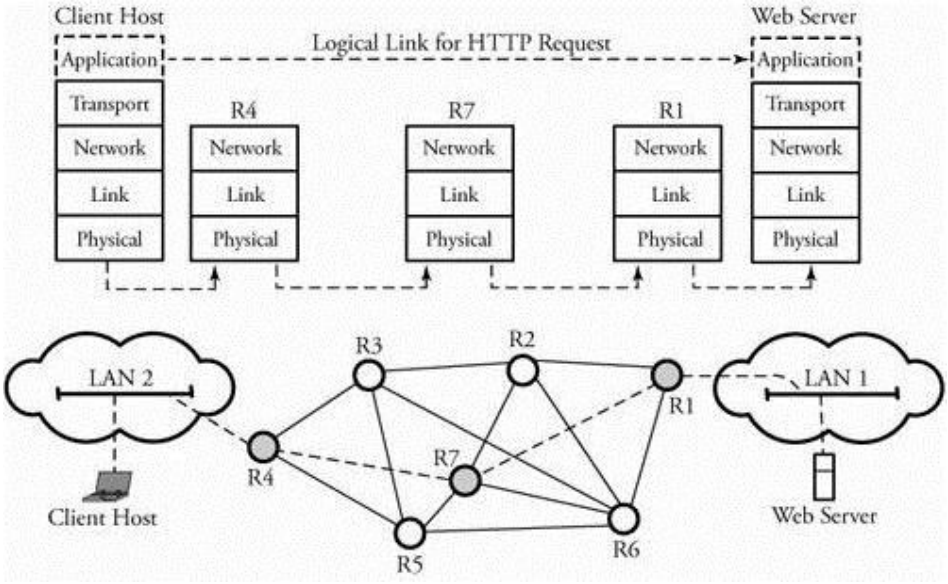


Figure 1.1: Online connection between two end systems.

The application layer gives the user ways to utilize an application to access data on the network. This layer serves as the user's primary interface and environment for interacting with the programme and, by extension, the network. The level of the TCP/IP protocol stack where the user-accessed network processes are found may be presumed to be the application layer. Some network operations that take place above the transport layer are the apps that operate at the application layer. All of the user-interactive procedures are included in this. As a result, the application layer offers the services necessary for user applications to communicate across a network, such as SMTP, FTP, Telnet, and Rlogin. We will talk about the different common services provided by the application layer in this section, including DNS, FTP, Telnet, and SMTP.

Application Layer: The transport layer serves as the foundation for the application layer, which offers network services to user applications. Electronic mail (e-mail), remote computer access, file transfers, newsgroups, the Web, streaming video, Internet radio and telephony, P2P file sharing, multiuser networked games, streaming stored video clips, and real-time video conferencing are all examples of applications that are defined and carried out by the application layer.

The application layer is dependent on certain pieces of software. In order to avoid having to rewrite an application for networking hardware like routers that operate at the network layer, software for new applications must be able to execute on numerous workstations. For instance, in a client/server architecture, a client end host asks a server host for services. A client host could be active sometimes or continually. An example of application-layer communication is shown in Figure 1.1.

Client and server model: A client/server computing paradigm offers various machines specialized computational services, such as partial-time consumption services. TCP and other dependable communication protocols enable interactive usage of distant servers as well. For instance, we might create a server that offered customers remote image processing services. A client and a server both need to be loaded with the application protocol in order to implement such a communication service. A client software that creates a TCP connection to a server must first be executed by the user in order to launch remote image processing. After that, the client starts sending bits of a raw picture to the server. The server processes the items that are requested and returns the processed data.

CHAPTER 2

DOMAIN NAME SYSTEM (DNS)

Kannagi Anbazhagan, Associate Professor,
Department of Computer Science and Information Technology, Jain (Deemed to be University) Bangalore, Karnataka, India
Email Id- a.kannagi@jainuniversity.ac.in

The Domain Name System (DNS) is the system that converts human-friendly domain names, such as www.example, into IP addresses that computers use to identify and communicate with one another. It is an essential component of the internet, as it allows users to easily remember and access websites and other resources without having to memorize the IP addresses. DNS is based on a hierarchical naming system, with the top level being the root domain. The root domain is managed by the Internet Corporation for Assigned Names and Numbers (ICANN), which is responsible for coordinating the assignment of top-level domains (TLDs) such as .com, .org, and Edu. Under each TLD, there can be multiple second-level domains.

DNS uses a distributed database system to map domain names to IP addresses. This database is spread across many servers, called name servers, that work together to resolve domain names. When a user types a domain name into their web browser, the browser contacts a name server to request the corresponding IP address . The name server then looks up the IP address in its local database, and if it doesn't have it, it contacts other name servers until it finds the correct IP address.

DNS is a critical component of the internet and is used by nearly every application and service that uses the internet. However, it is also a potential target for cyber-attacks, such as DNS spoofing and cache poisoning, which redirect users to malicious websites or interrupt can access to legitimate websites. To protect against these types of attacks, organizations can use security measures such as DNSSEC (DNS Security Extensions) and DNS Firewall. There are a few different types of DNS servers that can be used to resolve domain names.

Recursive name servers are the servers that users typically interact with. They are typically operated by internet service providers (ISPs) or other organizations that provide internet access. Recursive name servers receive requests from users and then query the appropriate authoritative name servers to resolve the domain name. Authoritative name servers are the servers that contain the actual mapping of domain names to IP addresses. They are typically operated by the organizations that own the domain names. Authoritative name servers respond to requests from recursive name servers and provide the IP address associated with the domain name.

Root name servers are the servers that provide the starting point for the resolution of domain names. They contain a list of the top-level domains and the IP addresses of the authoritative name servers for each TLD. DNSSEC (DNS Security Extensions) is an extension to the DNS protocol that provides authentication of DNS data, it helps prevent DNS spoofing and cache poisoning by digitally signing DNS records. This ensures that the information returned by a name server is legitimate and has not been tampered with.

DNS firewalls are security systems that are designed to protect against malicious domain name resolution requests. They can be used to block requests to known malicious domains or to rate-limit requests to prevent denial-of-service attacks.

DNS is a fundamental service that allows the internet to function by translating human-friendly domain names into IP addresses that computers can understand. The DNS system is composed of different types of servers and it's important to secure it to prevent malicious activities.

The Domain Name System (DNS) server is one of the most crucial elements of the application layer. DNS is a distributed, worldwide directory that uses a hierarchical structure to transform machine or domain names into IP addresses. Either UDP or TCP may be used for DNS. A DNS server performs a variety of information processing tasks, including: finding a host's address; assigning a subtree of server names to another server; identifying the beginning of a subtree that contains cache and configuration parameters and providing corresponding addresses; naming a host that handles incoming mail for a designated target; discovering the host type and operating system details; and discovering an alias for a host's real name.

Domain name system (DNS): Each host system on the Internet is given an IP address, as we already know. The numerical form of a 32-bit IP address looks like this: 202.12.32.22. Therefore, it might be exceedingly challenging for a user to recall an IP address for a particular computer. As a result, the IP addresses have been represented as a string of characters in English, such as yahoo.com for the IP address 68.142.226.32. The term "domain name" refers to this human readable name used in lieu of an IP address, such as "google.com". An agency named ICANN registers the domain names. The domain names may include alphabetic, numeric, or hyphenated letters and are not case-sensitive. When a person connects to the Internet, they enter the domain name rather than the IP address. Thus, a system that can change host domain names into IP addresses and vice versa is required. A mechanism called Domain Name System handles these conversions. Domain Name System (DNS) converts IP addresses into domain names as its primary purpose (human readable names).

Hierarchical Name Space: The IP addresses of different hosts on the Internet and various domains are stored in the DNS, a large database that is spread across several servers. In essence, the Domain Name System (DNS) is an Internet directory service that is disseminated. In this approach, the host must map the nearby DNS computer system that has the necessary data.

Some kind of naming system was necessary to give names to different devices. As a result, a hierarchical name space was created, with each domain name being separated into different branches of the tree, for example, the domain name "yahoo.com" has two branches. The domain name is first read from right to left, with yahoo denoting the name of the website and com designating a commercial website (higher level of the hierarchical structure) (lower level of hierarchical system) look at Figure 2.1.

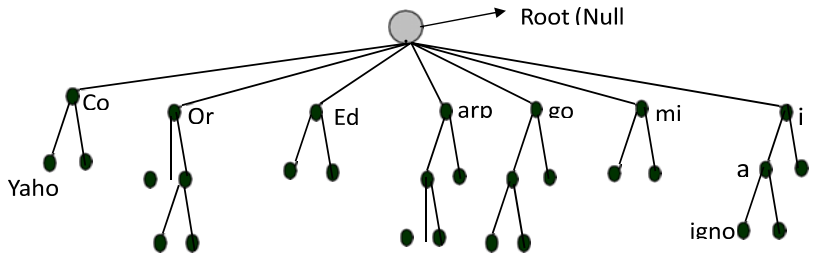


Figure 2.1: The DNS's Hierarchical Name Space

In DNS, a tree structure has been created such that its root connects the whole tree. The tree may have up to 128 levels, and each label can include a string of 63 characters. The period "." designates the root of the DNS hierarchy (tree). After that, the tree has a collection of top-level domains, including well-known ones like com, org, and edu as well as other country-level domains like in (India), etc. The domain names at this level, however, cannot be tied to a specific computer. The second-level registered domains, like hotmail.com, are located in the following levels of the DNS tree. It is possible to provide local domain names below the second level. The computer system must store the aforementioned tree structure. Nevertheless, storing this much data on a single computer system makes it susceptible to inefficient processing for the following reasons:

1. System accessibility is number one.
2. Reliability.
3. The system's ability to compute.
4. The network traffic volume on that specific transmission connection.

Domain Servers: These factors have led to the distribution of domain name information over a number of domain servers. The DNS database is separated into zones, each of which the server is in charge of or has jurisdiction over. Name servers are the servers in each zone that are in charge of responding to requests for that zone. A name resolver is client software that receives requests from a name server, which is a server programme that maintains a master or copy of a name-to-address mapping database or otherwise links to a server that does and responds to those requests. The naming hierarchy relates conceptually to a tree structure that is used to organise all Internet domain servers.

A zone is only a DNS subtree that is managed independently. For a zone, there are several name servers. One major name server and one or more subordinate name servers are typical configurations. More than one zone's authoritative name server is possible. The server whose zone contains the whole tree is the root server. In essence, a root server really delegated its own power to other servers rather than keeping track of domain names. There are now 13 root servers spread out over the globe that can handle the whole list of domain names. We have now covered the operation of domain name systems in general.

DNS Operate Online: The three categories that make up the Internet's domain name hierarchy are as follows:

1. Simple domain names
2. Domain names based on a country
3. Inverse domains

Resolution process refers to the idea of translating a domain name to an IP address and vice versa. Client-server technology fundamentally underpins the resolution process. The DNS calls a client application called resolver each time a user wants to map an address to a domain or vice versa. The resolver then makes a request to the nearby DNS server (name server). If the server has the required data, it responds with the outcomes. Alternatively, it proposes the resolver to other domain servers or requests the needed information from other servers. After receiving the findings, the resolver verifies the information and then sends the required information to the particular host process. The following are the actions taken in the resolution:

A system call such as get host by name is made by the user application. By giving the host name as an argument, this specific call is used to request the IP address of a host.

The resolver creates a name server query.

If the response is present in the name server's local authoritative database or cache, it is checked to determine whether it may return it to the client. If not, it will ping any additional name servers that are accessible, beginning at the DNS tree's root or as far up the tree as is feasible.

Finally, a matching IP address (or host name, depending on the query) or an error will be sent to the user application, or nothing will happen. These domain name request inquiries might be either recursive or iterative since the resolution procedure is carried out with the aid of queries. When a client requests a recursive inquiry, a flag bit in the domain name query indicates this, and a flag bit in the response indicates if the server allows recursive inquiries. A recursive inquiry asks the server to ask another query to get the information it needs to know and then deliver the whole result to the client. An iterative inquiry, on the other hand, requires the name server to deliver both the information it already has and a list of other servers the client might visit to finish the question.

DNS makes use of the caching idea as well. Once a name-server gets a client request, it immediately forwards the request to other servers. When the answer comes in later, it initially stores this data in its own cache memory before providing the requested data to the client. From this point forward, whenever any client requests the same information, the name-server may quickly answer the query by checking its cache memory. Such a reaction, however, is seen as non-authoritative. There are two categories of domain name replies provided by the name server: authoritative and non-authoritative.

DNS message use: There are two different message kinds in DNS since it adheres to the client-server paradigm: query and response. The messages in the DNS all adhere to the same format. Figures 2.2 and 2.3 depict the general formats of a query message and a response message, respectively.

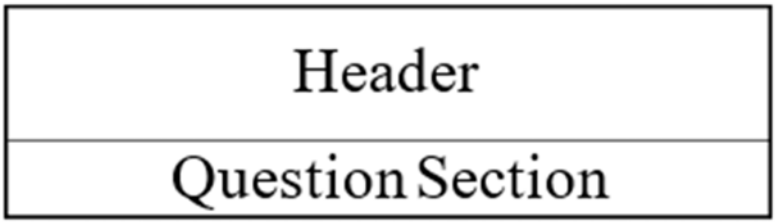


Figure 2.2: Query Message.

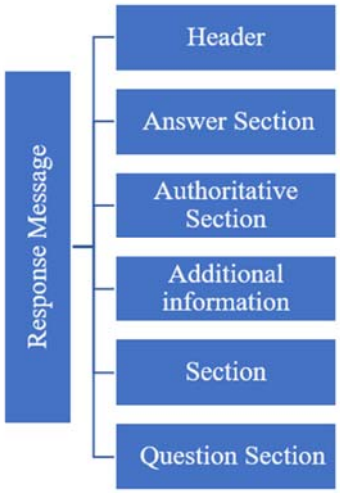


Figure 2.3: Response Message.

Table 1 illustrates the message's header's format. Always present and having a set length of 12 bytes is the header section (Table 2.1).

Table 2.1: Header format for Query/Response

Identification	Flags
Number of Question Records	Number of Answer Records
Number of Authoritative Records	Number of Additional Records

The means through which the sender of the message identifies themselves in order to match the query's answer.

1. Flags includes a number of attributes that specify the message type, such as recursive, iterative, authoritative, non-authoritative, etc.
2. The entire number of questions the resolver submitted in the Question Section is included in the number of Question Records.
3. The entire number of responses listed in the Answer Section is included in the Number of Answer Records.
4. The number of authoritative records in the authoritative section is shown under Number of Authoritative Records.
5. The entire number of additional records in the additional section are shown under • Number of Additional Records.
6. It should be noticed that the query message will have ZERO value for the Number of Answer Records, Number of Authoritative Records, and Number of Further Records sections.

Dynamic Domain Name System (DDNS):

The Dynamic Domain Name System (DDNS), a protocol that specifies extensions to the DNS, allows DNS servers to accept requests to dynamically add, update, and remove records from the DNS database. A DDNS server may simultaneously serve both static and dynamic domains since it provides a functional superset to current DNS servers. The secure variant of DDNS authenticates update requests from DDNS hosts using public key security and digital signatures as opposed to allowing any host to change its DNS entries.

Several DNS systems come with the following three tools for querying name servers:

1. The host acquires an IP address linked to a host name or an IP address linked to a host name.
2. Nslookup enables you to get information about network nodes, explore a name server database's contents, and determine if name servers are reachable.
3. Hig enables you to test name servers, compile massive amounts of domain name data, and do simple domain name searches. Domain Internet Groper is what DIG stands for.

The Domain Name System standard and the data stored in the system are described in the following RFCs:

1. Domain Administrator's Handbook, RFC 1032
2. Domain Administrator Operations Handbook (RFC 1033)

3. Domain Names - Concepts and Facilities, RFC 1034
4. Domain Names - Implementation and Specification (RFC 1035)
5. DNS Encoding of Network Names and Other Types, RFC 1101.

Electronic mail: The most used TCP/IP service is likely electronic mail, or email. It has integrated itself into most peoples' daily lives. For two end users, electronic mail offers a platform for information sharing. It is primarily used for sending and receiving emails and messages between end users, including text, audio, graphical, and video messaging. An overview of the TCP/IP application protocol for email is given in this section.

Simple Mail Transfer Protocol (SMTP): Simple Mail Transfer Protocol is the common electronic mail delivery method used on the Internet (SMTP). It enables message and mail transmission between TCP/IP hosts. End-to-end delivery is the foundation of SMTP; to send mail, an SMTP client speaks with the destination host's SMTP server directly. Once the message has been properly duplicated into the recipient's SMTP, the mail is kept on the destination host's SMTP server. As seen in Figure 2.4, the SMTP is a client-server communication service that utilises port number 25 on the server. The different parts of SMTP include:

1. Mail Transfer Agent (MTA)
2. User Agent (UA)

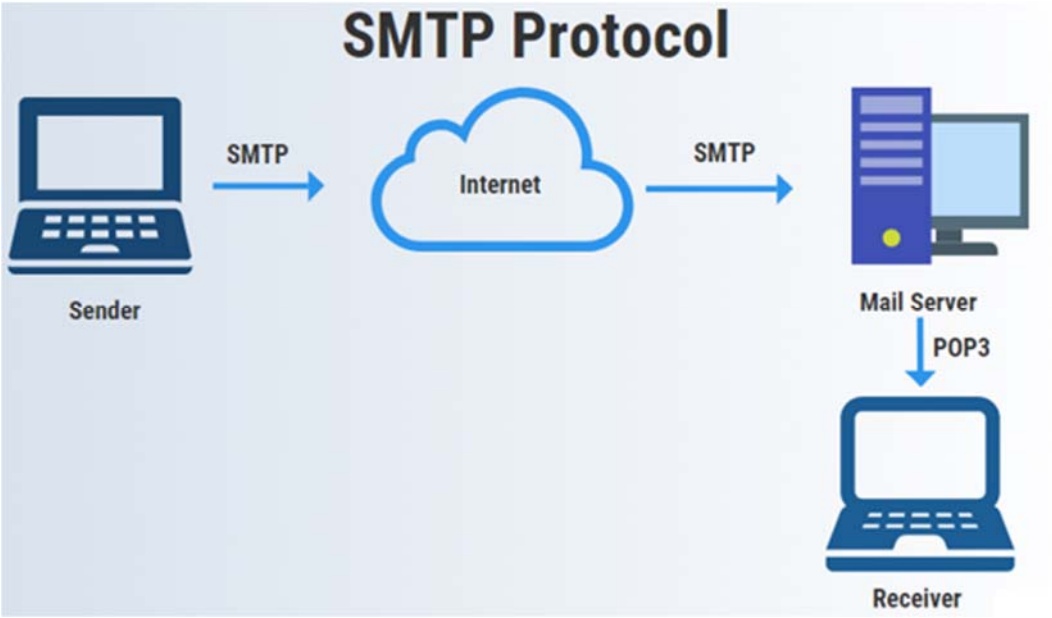


Figure 2.4: SMTP Service

Message Transmission Agent: The transmission of messages i.e., mails is delivered by an agent called as Message Transfer Agent (MTA) (MTA). MTAs aid a user in sending as well as receiving the messages. The MTA includes of two colours i.e., MTA client for sending the mail while MTA server for listening/receiving the mails as illustrated in Figure 2.4. The MTA is general and establishes the format for sending back and forth between instructions and answers. A prominent example of MTA under UNIX is known as send mail.

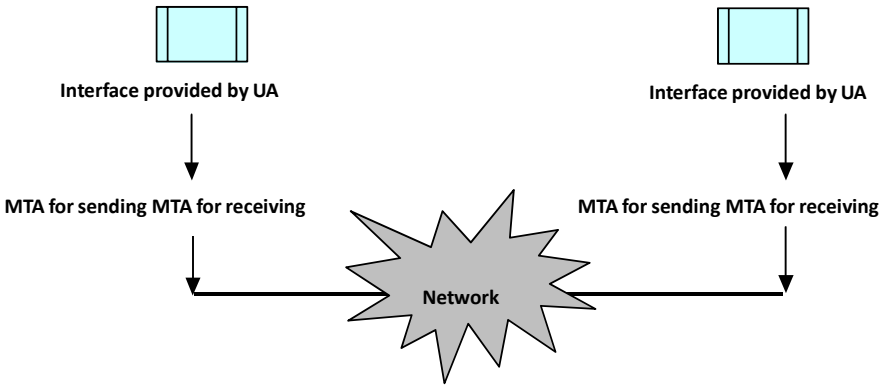


Figure 2.5: Represent the SMTP.

User Agent: The user agent primarily handles message composition. The user agent gives the user access to an interface in the manner shown in Figure 2.5 where they may compose messages and specify destination addresses (that is create an envelope). Hence, UA places the letter inside the envelope. The user agent provides the following range of services (Figure 2.6):

1. Reviewing the texts, you've received
2. Answering the messages, you've read
3. Writing the messages.
4. Sending the messages forward
5. Taking care of the different mailbox settings.

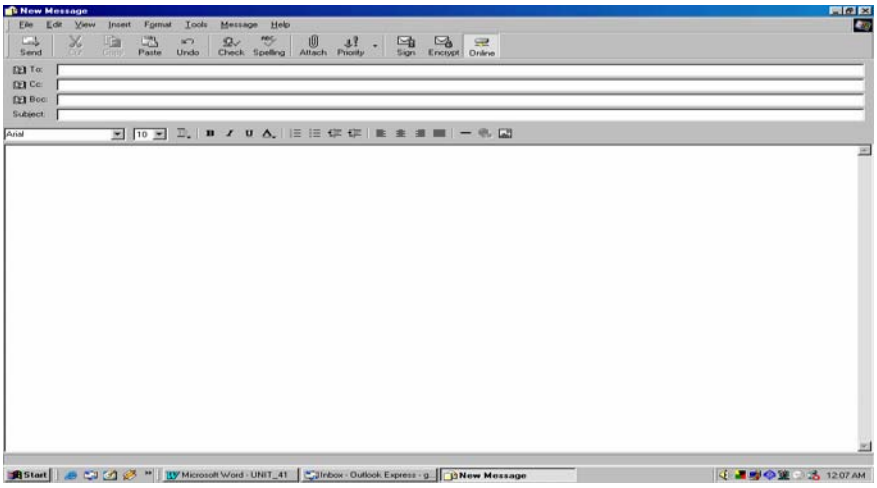


Figure 2.6: shows the interface offered by UA for creating messages.

The user name and the domain name are the two components that make up a user's address (e-mail address). As an example, the email address amitgoel@yahoo.com shows the user name "amitgoel" and the domain name "yahoo.com" separated by the @ symbol. As a result, the SMTP protocol uses the user name, @, and domain name to represent an email address. Please keep in mind that UA does not provide the ability to send and receive messages. The domain name used to identify a specific network and then a specific user on that network is included in the address of the destination host.

User agents may offer an interface that is either command-based or graphical interface-based. Pine, Mail, and other examples of user agents for command-based interfaces are available, whereas Netscape, Outlook Express, and other examples are available for GUIs.

Postal Protocol (POP): Due to the fact that SMTP is based on the TCP/IP protocol, a TCP connection must be created between the two endpoints. Users cannot expect their computers to stay online always, particularly a desktop computer used at home. As a result, it was necessary to create a system that would allow users to access their email even while their computer was turned off. As a result, the majority of businesses install an SMTP server that is constantly up and receives emails on behalf of each and every user connected to the company's network. The SMTP server essentially serves as a post office.

A mechanism known as Post Office Protocol (POP) has been developed to retrieve the users' emails stored in the SMTP server. As seen in Figure 2.7, it aids in downloading emails from the SMTP server.

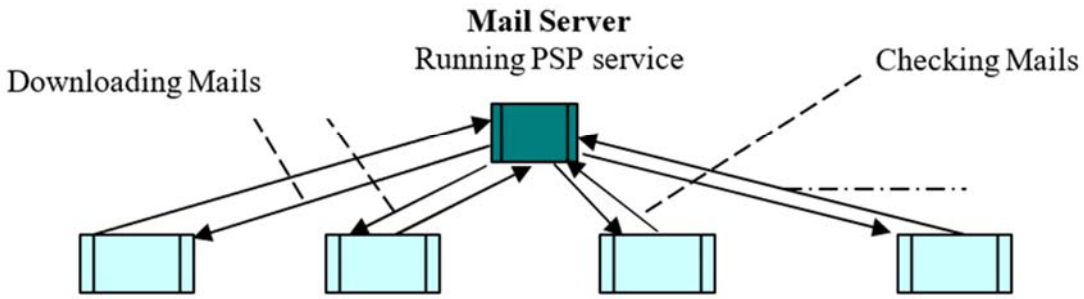


Figure 2.7: Illustration of POP

Internet Mail Access Protocol (IMAP): When a user accesses, or downloads, emails from a mail server using POP, the downloaded emails are immediately erased from the mail server. POP is thus inappropriate for users who access their mail from many places, such as a cybercafé, their house, a hotel, etc. POP does not provide the ability to organise emails on the mail server into folders or otherwise. Another protocol called Internet Mail Access Protocol (IMAP) has been created to prevent the deletion of emails from the mail server. Together with the features provided by POP, IMAP further offers the following services:

1. The mailbox on the mail server may be created, renamed, or deleted by the user.
2. Before to downloading the message, the user may examine the mail's header.

Internet Mail Extension with Many Uses (MIME): The Network Virtual Terminal seven-bit ASCII format is the sole one used by the SMTP protocol to transmit emails, or messages. In other words, it will only handle languages that can be encoded in seven-bit ASCII. As a result, messages composed in German, Russian, or French cannot be transmitted using SMTP. Also, binary data, audio files, and video files cannot be sent using SMTP. Thus, a system for enabling the transfer of non-compliant formats has to be created.

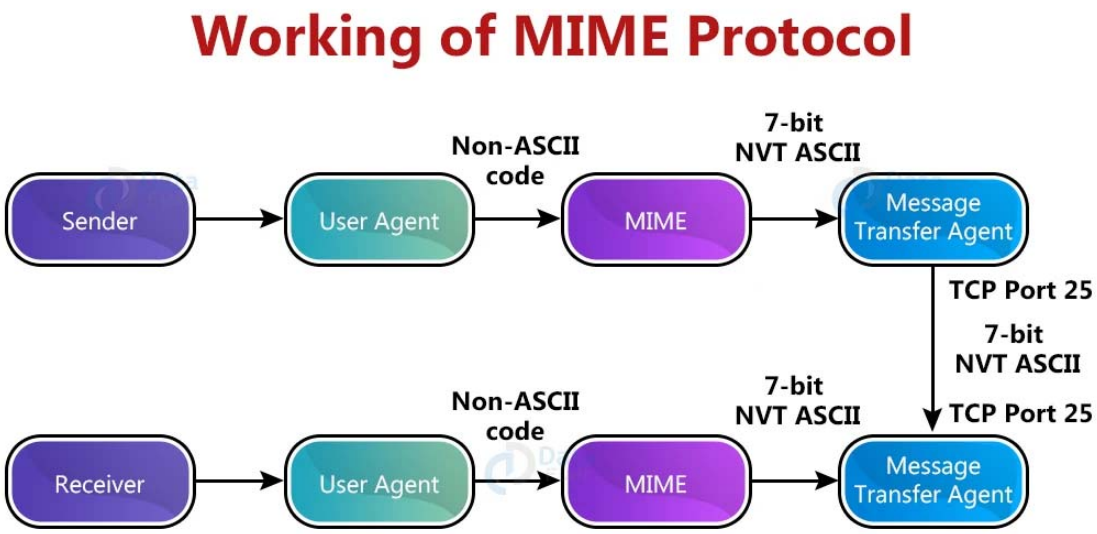


Figure 2.8: Multipurpose Internet Mail Extension.

Non-ASCII formats may be sent using SMTP thanks to the Multipurpose Internet Mail Extension (MIME) protocol. MIME primarily translates non-ASCII formats into ASCII format before sending the information to SMTP. As a result, the SMTP transfers the data to the target system in ASCII format. The destination machine's SMTP service sends ASCII data to MIME, which transforms it into non-ASCII format as seen in Figure 2.8. Keep in mind that MIME is not a mail protocol; it is only an extension of SMTP. The original SMTP header may be added to one of the five MIME-defined headers to specify the following parameters:

MIME Version: It details the specific MIME version that is being utilized. MIME is currently at version 1.1.

Content-Type: This specifies the kind of data that will be used in the message's body, such as text, images, video, audio, postscript, etc.

Content-Transfer-Encoding: This specifies how to convert a mail message into a series of 0s and 1s. The many encoding methods include the following: ASCII-7-bit, non-ASCII-8-bit, Non-ASCII-Binary, Non-ASCII-Base64, and Non-ASCII-Quoted-Printable are the different character sets.

Content-id: In a case of several messages, it identifies the whole message.

Content-Description: Indicates if the message's body is text, an image, a video, or any other kind of media.

TELNET: The capacity to do remote execution, or calling a programme on a distant terminal, is the most basic data transfer technique used on a network. A well-known application protocol that allows for remote execution is TELNET. A well-liked client-server application software is TELNET. TErminal NETwork is referred to by the acronym TELNET. A programme on one host, known as a TELNET client, may use an interface provided by the standard application protocol TELNET to access the resources of another computer, known as a TELNET server. Figure 2.10 illustrates how TELNET's environment enables the client to function as a local terminal linked to the server. TELNET is essentially a tool that allows users to access files and

programmes that are located on distant computers after initially logging into the remote workstation.

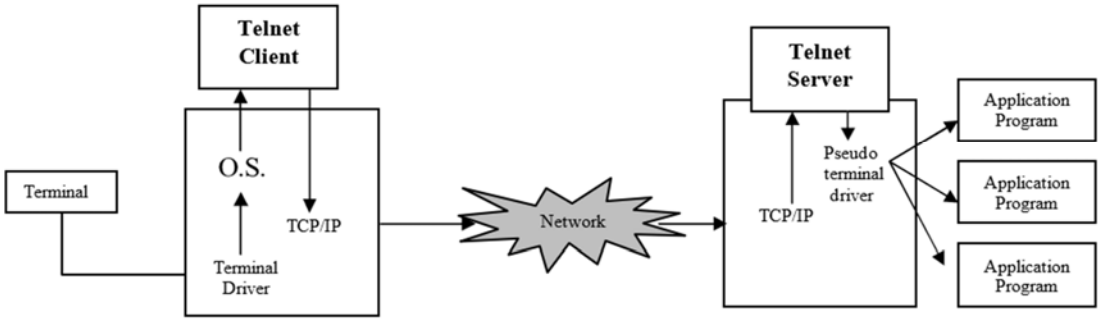


Figure 2.9: Represent the TELNET

The way TELNET works is as follows: The user types on the host computer's terminal, which is running the terminal driver driver (a module of the operating system). In essence, it accepts the user's keystrokes on the terminal and sends the operating system the relevant characters. These characters are then sent to the TELNET client by the operating system. The language and format that the terminal will accept may not be the same as what TELNET would accept, as was covered in SMTP. As a result, the Network Virtual Terminal (NVT) characters are used to represent the characters that the TELNET client receives from the terminal driver. The local machine's TCP/IP protocol stack receives the modified version of the characters (Figure 2.9).

TCP/IP is used by the characters as they move via the network until they arrive at the target operating system. The NVT characters are sent by the operating system to the TELNET server. The destination system can interpret the characters thanks to the TELNET server's transformation of them. With a unique kind of driver known as a pseudo terminal driver, the set of characters is sent to the operating system. The pseudo terminal driver assists the TELNET server in mimicking a terminal since the TELNET server cannot directly communicate with the operating system. As a result, these operating system characters are sent to the correct application software on the distant computer.

File transfer protocol (FTP): One of the most frequent network activities is the exchange of files between two devices. File Transfer Protocol is the common method for transferring files from one computer to another (FTP).

Even though it could seem to be fairly straightforward to transmit data between two sites, there are several problems that must be overcome. For example, two systems may use various file naming standards, data representations, directory topologies, etc. FTP has been used to assist overcome these problems. A client-server setup is used by FTP. With FTP, the user must first authenticate with the server before the server will enable them to view distant files. Keep in mind that FTP employs a connection-oriented service, which means that in order to start a file transfer, both hosts must be running TCP/IP. Data is sent between the client and server once the connection has been established.

TCP is a transport protocol that is used by FTP to provide dependable end-to-end connections. For data transfers (data connections), the FTP server waits for connections on the well-known port numbers 20 and 21, respectively (control connection). With the aid of logging in, the control connection does the authentication job and subsequently adheres to the TELNET

protocol. To access files and folders, the user needs a user name and password to log into the remote host.

As illustrated in Figure 2.10, the FTP programme is constructed using a user interface on top of the protocol, a Data Transfer Process (DTP), and a protocol interpreter (PI). The user interface on the client side interacts with the protocol interpreter, which manages the control connection. All control orders that are required to be sent to the remote system must be sent via this protocol interpreter (PI). In addition to responding to the TELNET protocol on the server side, the protocol interpreter also needs to start the data connection. DTPs control the data while the files are being sent. The server's PI must terminate the data connection when a user request has been fulfilled.

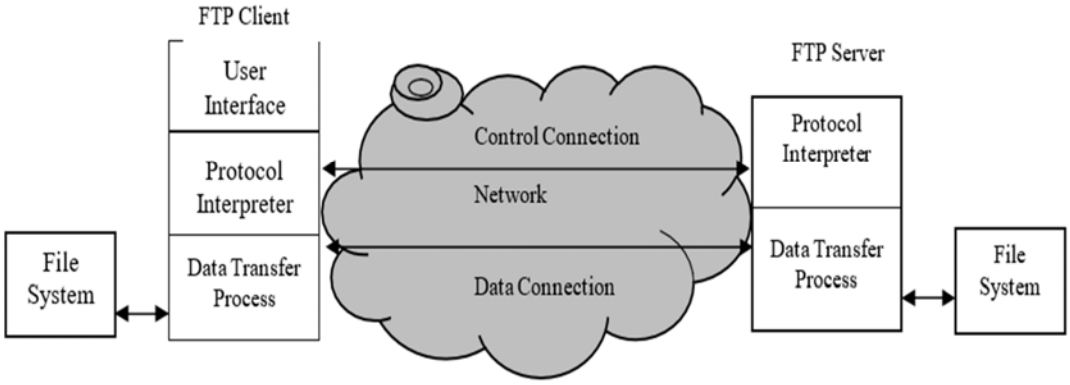


Figure 2.10: File Transfer Protocol (FTP)

The following are the fundamental actions that a client does during an FTP session:

Join the user to a distant host: With the use of a legitimate user name and password, the user must first verify their identity to the server system. For the purpose of establishing a connection, the following commands are necessary:

Type "ftp" at the command prompt.

Next, type the command "open" and the IP address of the server computer, for example, "open 202.12.141.81". The open command makes the remote computer seem to be in a login session.

Upon a successful connection to the remote host, the remote computer requests the user name and password associated with it.

Login and password for ftp

User and pass are the additional commands. Finding a remote user id is the goal of the user command. The pass command's main function is to verify the remote user id.

Decide on a directory: The client uses the "cd" (change directory) command to look for the proper directory once the connection has been established. Using the LCD (local change directory) command, the user may choose a local directory.

List the files that can be transferred: Depending on the operating system, commands like dir or ls are used to list the numerous files in a certain folder.

State the transfer mode: Data transformations into intelligible forms are necessary because data transmission might occur across systems with different architectures. The user must primarily choose between two data handling components: the technique by which the bits will be transferred from one location to another, and the various data representations. These commands are used to address the aforementioned problems:

Mode: This option determines whether the file should be treated as a record structure or as a byte stream.

Type: This indicates the character set, such as ASCII, Image, etc., that is being used to represent the data.

From the remote host: Files may be copied using the following commands between FTP clients and servers:

1. The get command transfers a file from a distant host to a local host.
2. Mget: This command transfers several files from a distant host to the local host.
3. The put command moves a file from the local host to the distant host.
4. The Mput command transfers several files from the local host to the distant host.

Disconnect the connection to the remote host: The following commands can be used to break a connection:

Finish: The connection to the remote host is cut off, but the FTP client is still open.

Quit/Bye: It ends FTP and disconnects from the remote host.

FTP anonymous: The first step in connecting to a remote computer is to have a working user name and password. There are, however, not many FTP servers that are open to the general public. Hence, a user does not need to have a valid user ID in order to provide access to such distant public servers. With these servers, the username is anonymous and the password is guest. Recall that under certain circumstances, user access is restricted.

This lesson gives a thorough understanding of TCP/application IP's layer. We have learned so far that there are several types of protocols connected to the application layer. Several of them, such as DNS, SMTP, TELNET, and FTP, have been covered in this unit. The client-server programme known as the domain name system links domain names to IP addresses. DNS keeps track of a hierarchical structure to store data on a massive number of domain names. The DNS servers are spread out over the name space. Each server has been given control over a certain zone.

There are 13 root servers spread over various geographical areas. The zone server is the principal server for each zone. The name resolution function, known as resolver, converts the domain name to the IP address and vice versa whenever a user requests an IP address.

The SMTP protocol offers a way to send messages across the network while also offering a user interface. There are two parts to it: UA and MTA. The interface for writing, reading, responding to, and forwarding mail messages is provided by the UA. The MTA handles the actual message transport across the network. MIME is an add-on for SMTP that enables the transmission of messages in non-ASCII formats, such as video and audio, using SMTP. POP and IMAP are SMTP extensions that enable messages to be saved in the mail server so that a user may reach the mail server in the future and view his or her mail.

Remote login is made possible through the client-server TELNET protocol. FTP is an application protocol that uses TCP/IP to move files from one host to another. Open, site, and

other fundamental instructions are needed for connection formation. The commands get, mget, put, and mput are used to transmit data. The commands quit and close are used to end a connection. The topics of network programming are covered in the next block of this course.

CHAPTER 3

COMPUTER NETWORKS: DATA LINK LAYER

Kamalraj R, Professor,
Department of Computer Science and Information Technology, Jain (Deemed to be University) Bangalore, Karnataka, India
Email Id- r.kamalraj@jainuniversity.ac.in

Data Link Layer: The second tier of the seven-layer OSI model reference model is known as the data link layer, or layer 2. In a computer communication context, this layer specifies the methods for gaining access to a shared communication channel and ensuring accurate data frame transfer. Framing, error detection and repair, acknowledgment, flow management, establishing a well-defined, dependable service interface to the network layer, encapsulating packets from the network layer to frames, etc. are some of the key duties of the data link layer. Links between two machine nodes may be made using a variety of link-level technologies. Data connection layer protocols include Ethernet, Token Ring, FDDI, and PPP as examples.

1. Functions of the Data Link Layer: The Data Link Layer is capable of performing a variety of distinct tasks.
2. One of these duties is to provide the network layer a clear service interface.
3. Framing.
4. Handling transmission mistakes.
5. Controlling the data flow to prevent rapid senders from overpowering slow receivers.

Services given to the network layer: It gives the layer 3 or network layer a clear and dependable service interface, which is also reliant on the effectiveness and error rate of the underlying physical layer. The following is how the data connection layer completes these tasks:

1. Unrecognized connectionless service.
2. Embraced connectionless service.
3. A connection-oriented service acknowledged.

Framing: The data connection layer gets a raw, potentially error-filled bit stream from the physical layer. The data link layer divides the bit stream into frames to provide a dependable delivery of bit streams to the network layer. The checksum is then calculated for each frame and sent along with the frame. When a frame is received, the destination host generates a new checksum from the data and compares it to the frame that was broadcast. This guarantees that the receiver's data connection layer will both identify and correctly interpret frames.

Error detection and correction: This is a group of coding-based techniques used to both find and fix faults in transmitted or stored data. A number of various techniques based on Single Error Correction and Double Error Correction (SECDEC). Several methods exist for identifying faults in transmitted data, including:

Parity Checks: The simplest kind of error detection tool, a parity check uses a single parity bit. Both even parity and odd parity approaches are included.

Checksum: This straightforward redundancy check is used to find errors. The binary values in a packet or other block of data are calculated using this approach, and the results are stored with the data.

Cyclic Redundancy Check: A CRC is applied to a packet frame at the data connection layer of the TCP/IP or OSI reference models. It is a technique for examining data that has been transferred via a communications network for mistakes.

Hamming code: This binary code is error-detecting and error-correcting.

Flow control: Managing the pace of data transfer between two sources and destination hosts is another crucial aspect of the architecture of the data connection layer. Packets will be dropped at the receiver end if there is a difference in data sending and receiving speeds between the sources and destination sites. Also, it makes the sender stop sending acknowledgment packets, which results in retransmission and reduces the effectiveness of the network. Various framing techniques:

1. Number of Characters
2. Flag bytes stuffed with bytes
3. Bit stuffing and starting and ending flags.
4. Infractions of the physical layer coding

Method 1: Character count: The number of characters in the frame is specified using this approach using a field in the header. The character count informs the data connection layer at the destination of how many characters are left and, therefore, the location of the end of frame. This method is shown in the following picture for four frames with character counts of 5, 5, 8, and 8 correspondingly (Figure 3.1).

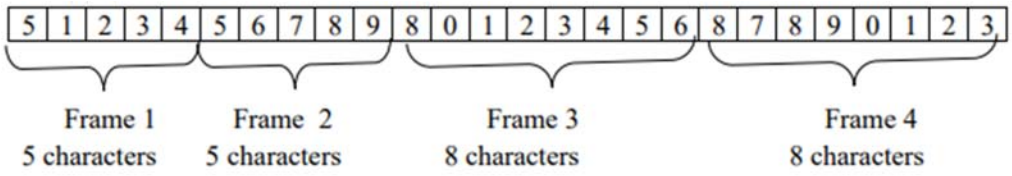


Figure 3.1: Character stream without errors

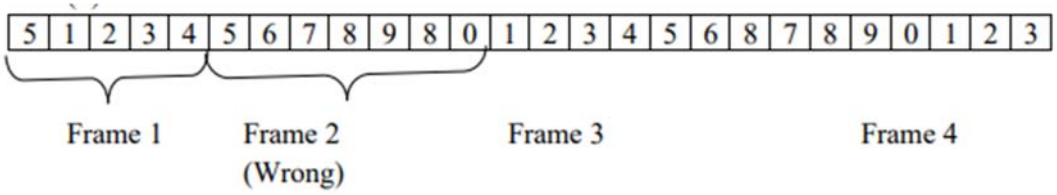


Figure 3.2: Character stream with one error

The issue with this approach is that a transmission fault might muddle the count. For instance, if the destination loses synchronisation and cannot find the beginning of the following frame if the character count of 5 in the second frame of figure 3.2 increases to 7. The destination has no means of knowing where the next frame begins, even if the checksum is off and it is obvious that the frame is faulty. The destination does not know how many characters to skip over to obtain the start of the retransmission, thus sending the source a frame requesting for a retransmission does not assist either.

1. Roles of the datalink layer
2. Support for the Network Layer Services
3. Framing
4. Error Management

Flow management

The data link layer takes the packets it receives from the network layer and wraps them into frames for transmission in order to achieve these objectives. A frame's payload field, frame trailer, and frame header are all included in each frame (Figure 3.3).

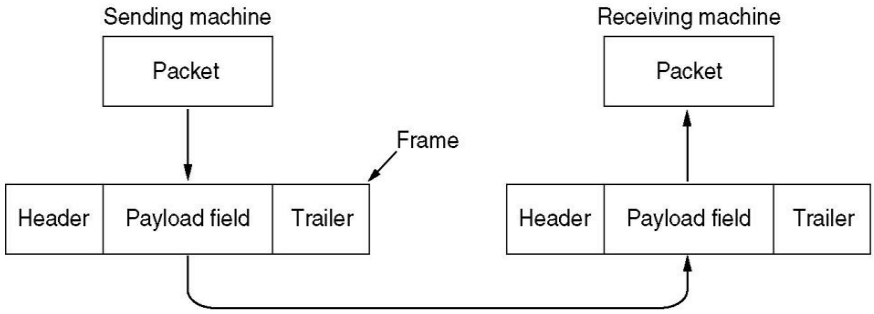


Figure 3.3: Relation between frames and packets:

Frame headers employ physical addresses to indicate source and destination.

Types of Services Provided To the Network Layer

1. Connectionless service not recognised
2. Connectionless service acknowledged
3. A service that is acknowledged to be connection-oriented
4. Connectionless service that is not recognised
5. No prior or subsequent logical relationship is made or relinquished.
6. No effort is taken in the data connection layer to identify or recover from a frame loss caused by noise on the line.
7. When the mistake rate is extremely low and recovery is left to higher levels, this type of service is suitable.

In real-time traffic, such as speech, when late data are worse than faulty data, it is also acceptable. On the data link layer, the majority of LANs provide unacknowledged connectionless service.

Connectionless service acknowledged

Without using any logical connections, the sending of each frame is uniquely recognised. Sender is aware of whether a frame arrived properly. It may be sent again if it hasn't arrived after a certain amount of time.

On unstable channels, like wireless networks, this service is helpful.

Connection-Oriented service that is acknowledged

There are three main stages that transfers go through when connection-oriented service is employed. The variables, buffers, and other resources required to maintain the connection are released when the connection has been created, one or more frames have actually been transferred, and the connection has been released.

Framing: The data link layer must use the service that the physical layer provides to it in order to deliver service to the network layer.

A raw bit stream is accepted by the physical layer, which makes an effort to transport it to the target. This bit stream's error-freeness cannot be guaranteed. The data connection layer is

responsible for error detection and correction. The typical method is to divide the data stream into discrete frames and calculate the checksum for each frame in order to identify and rectify problems. The checksum is updated when a frame reaches its destination. The data link layer responds to errors by taking action if the newly calculated checksum differs from the checksum that was included in the frame.

There are four common framing techniques:

1 Number of characters.

2. Byte stuffing in the flag bytes.

3. Bit stuffing and starting and ending flags.

Characters Used: The amount of characters in the frame are specified using a field in the header. The character count informs the data link layer at the destination of the number of characters remaining and, therefore, the location of the frame's end. This method is shown for four frames with character sizes of 5, 5, 8, and 8 correspondingly (Figure 3.4).

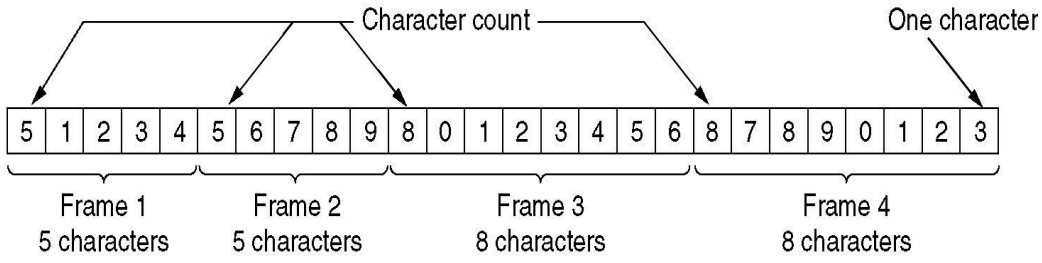


Figure 3.4: A character stream without errors.

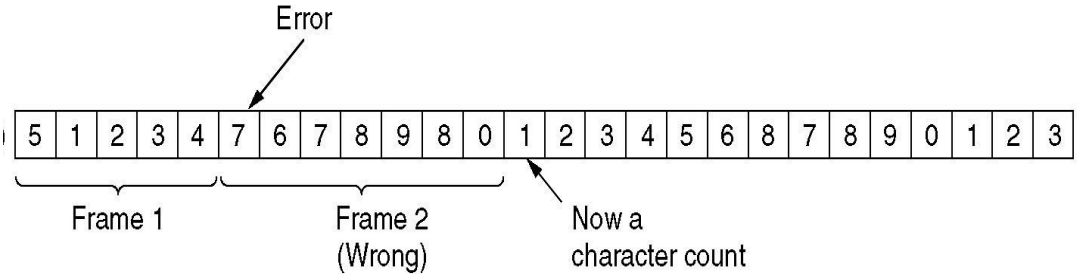


Figure 3.5: With one error with one error.

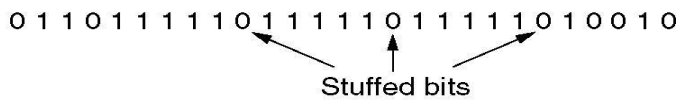
Problems: It is challenging to determine the start and finish of the frame if the count is destroyed. There is no way to determine where the next frame begins, even if the receiver has a checksum that indicates the frame is faulty. Retransmission requests are not feasible since it is unknown when the retransmitted frame will begin (Figure 3.5).

Using byte/character framing: The flag byte, which serves as both the beginning and ending delimiter for each frame, is a unique kind of byte. In this method, the receiver may simply look for the flag byte to determine the end of the current frame if it ever loses synchronization. The end of one frame and the beginning of the next are marked by two flag bytes in a row. This approach has a severe flaw in that the data may include the flag byte's bit pattern. The framing will often be hampered by this circumstance.

Solution: Before each "accidental" flag byte in the data, the sender's data connection layer inserts a special escape byte (ESC). Before sending the data to the network layer, the receiver's data link layer removes the escape byte. Byte stuffing or character stuffing is the name of this method. Fixed character size, 8-bit character size assumption, and inability to handle diverse environments are drawbacks.

Using bit stuffing to frame: A unique bit sequence, 01111110, is used to start and conclude each frame. The sender's data connection layer automatically inserts a 0 bit into the outgoing bit stream if it sees five consecutive 1s in the data. The receiver immediately deletes the 0 bit when it observes five consecutive incoming 1 bits followed by it. Using the new method, data frames may have any number of bits and character codes can have any number of bits per character.

0 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 1 0



0 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 1 0

The flag design may clearly identify the border between two frames when bit stuffing is present. Due to the fact that flag sequences may only ever appear at frame borders and never inside of data, if the receiver becomes disoriented all that is required of it is to scan the input for them.

Failure Control

Positive and Negative Recognition: The receiver sends back a special control frame with positive or negative acknowledgements about the incoming frames to ensure that all frames are ultimately delivered to the network layer at the intended location and in the correct sequence.

A frame has reached successfully if the sender gets a reply in the affirmative. A negative acknowledgment indicates a problem, and a new transmission of the frame is required.

Positive or negative acknowledgment will not reach the sender if there is hardware issue, and it will continue to linger indefinitely. Here, timers are applied.

Timers: Each time the sender sends a frame, a timer is also started. The timer is programmed to stop when there has been enough time for the frame to go to the intended location, be dealt with there, and for the acknowledgment to propagate back to the sender.

The timer will be terminated if the frame is successfully received and the acknowledgment is returned before the allotted time has passed. The timer will sound and notify the sender of the issue if either the frame or the acknowledgment are lost. The sender will then just transmit the frame once again.

Sequence numbers and duplicate frames: When a frame may be broadcast more than once, the receiver will accept it twice or more and send it more than once to the network layer. Each outgoing frame is given a sequence number to help the recipient differentiate between retransmissions and original frames in order to avoid this.

Speed Control: Creating communication across networks with different speeds is referred to as flow control. Feedback-based flow control and rate-based flow control are the two different forms of flow control.

Flow control based on feedback: The receiver informs the sender of its present data handling condition and grants the sender permission to increase or decrease data flow.

Rate-based flow regulation: The protocol includes a method to restrict the pace at which the transmitter may send data without receiving feedback from the recipient.

CHAPTER 4

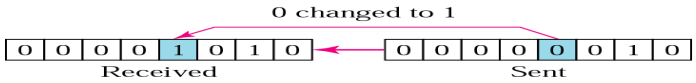
ERROR DETECTION AND CORRECTION

Nidhya M S, Associate Professor,
Department of Computer Science and Information Technology, Jain (Deemed to be University) Bangalore, Karnataka, India
Email Id- ms.nidhya@jainuniversity.ac.in

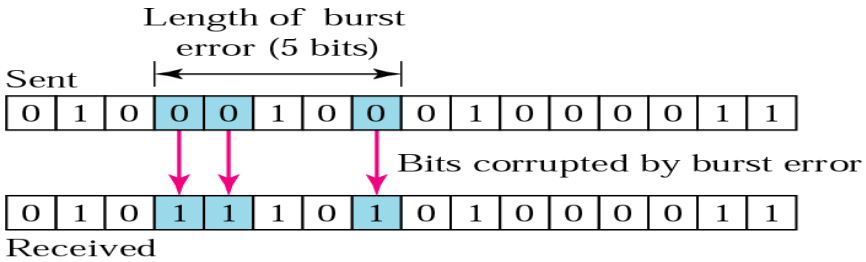
Data transmissions are susceptible to corruption. Errors must be found and fixed for communication to be reliable. Either the data connection layer or the transport layer of the OSI model include error detection and correction. Various Errors

1. One-bit mistake
2. Burst mistake

Single-bit mistake: Just one bit in the data unit has altered in a single-bit mistake.



When data is sent in parallel, single-bit errors occur.



Burst Error: When two or more bits in the data unit change, it is considered a burst error.

Burst mistakes don't always indicate that they happen in quick succession of bits. From the initial corrupted bit to the final corrupted bit, the burst's length is calculated. There may have been some uncorrupted parts in between. Serial transmissions may experience burst errors. The number of bits that are impacted depends on the data rate and noise duration. Example: A noise of 1/100 seconds may impact (1000/100) 10 bits of data delivered at 1 kbps. The same noise may damage 10,000 bits of data delivered at 1 mbps (1,000,000/100).

Error detection: Redundancy, which refers to include additional bits to detect faults at the destination, is a notion used in error detection. Four different kinds of redundancy tests (Figure 4.1).

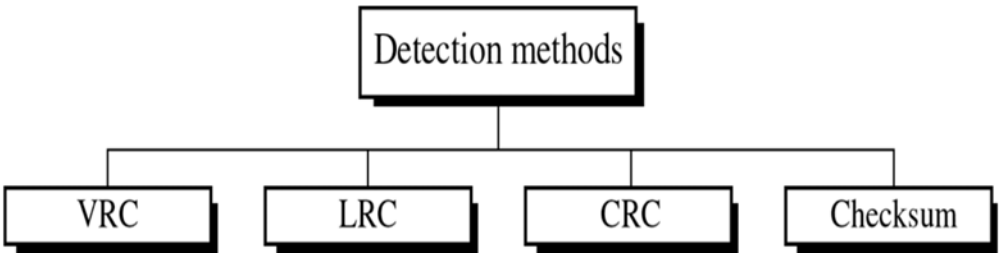


Figure 4.1: Four different kinds of redundancy tests.

1. Vertical Redundancy Check
2. Longitudinal Redundancy Check
3. Cyclic Redundancy Check (CRC)
4. Checksum

Check for Vertical Redundancy (VRC): VRC is another name for parity check. Every data unit is given a parity bit during parity check to ensure that there are an equal number of one's overall (or odd for odd-parity).

Let's say the sender is attempting to communicate the word world. The five characters are represented in ASCII by the codes

1110111 1101111 1110010 1101100 1100100.

The following displays the exact bits delivered, including any superfluous ones: 11101110, 11011110, 11100100, 11011000, and 11001001

The receiver counts the 1s in each character as it is received. Data are acceptable if the number of 1s in each character is even (6, 6, 4, 4, and 4).

Imagine if the word was tampered with during transmission.

11111110 11011110 11101100 11011000 11001001

The receiver calculates even and odd integers by counting the 1s in each letter (7, 6, 5, 4, and 4). The recipient discards the faulty data after realising it and requests a new transmission.

Any single-bit faults may be found with a simple parity check. Only if there are an odd number of faults overall in each data unit can it identify burst errors.

Check for Longitudinal Redundancy (LRC): A parity check in two dimensions. A block of bits is split into rows, rows of superfluous bits are added to the block, and a parity is established for each column (Figure 4.2).

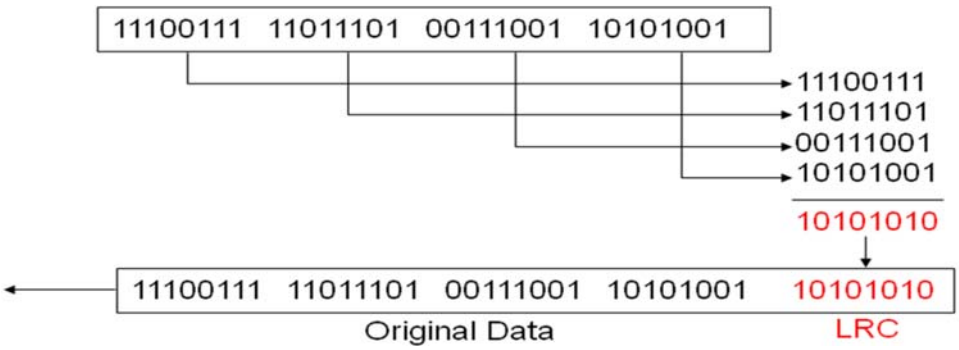


Figure 4.2: Longitudinal Redundancy Check (LRC)

LRC's performance: If two bits in one data unit are modified and two bits in precisely the same places in another data unit are similarly damaged, the LRC checker will not discover mistakes. Detects all burst errors up to length n (number of columns).

Example: Let's say the block sent is as follows:

10101001 00111001 11011101 11100111 10101010

Yet an 8-second long blast of noise hits it, corrupting part of the bits.

10100011 10001001 11011101 11100111 10101010

Certain bits do not meet the even-parity criterion when the receiver verifies the parity bits, and the whole block is thrown out as a result.

10100011 10001001 11011101 11100111 10101010

Cyclic Redundancy Check (CRC)

Most potent utilizes binary division (divisor and remainder). A group of redundant bits known as the CRC residual is added at the end of a data unit rather than adding bits to achieve the required parity (as in VRC or LRC).

CRC rest has two characteristics. Modulo-2 division is used to create the CRC residual.

It must have one bit less than the divisor precisely.

The data string must be precisely divided by the divisor when it is appended to the end (Figure 4.3).

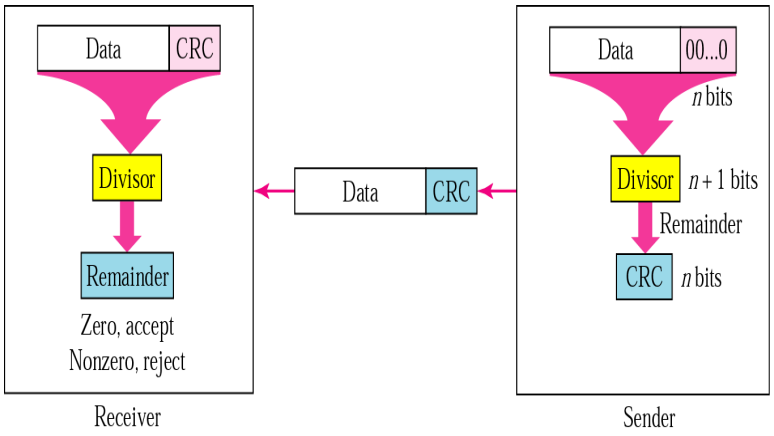


Figure 4.3: Cyclic Redundancy Check (CRC)

Basic actions

1. The data unit is appended with a string of n 0s.
2. n- One fewer than the specified divisor's predetermined number of bits, which is n+1 bits.
3. The divisor divides the newly larger data unit using binary division.
4. The n-bit CRC residual from step 2 is replaced at the end of the data unit. The CRC remainder might include just zeros.
5. The message is sent to the recipient along with the remaining information.
6. The receiver splits the whole string by the same factor that was used to calculate the CRC remainder.

If the remainder is 0, there was no transmission mistake, and the data is accepted without using the CRC's remainder. There was a communication issue if the remainder does not equal zero. Data is therefore discarded as having errors.

A polynomial-based CRC Generator (the divisor) was developed. The following characteristics are used to choose a polynomial. Both x and (x+1) should not be divisible by this number.

$$x^7 + x^5 + x^2 + x + 1$$

A polynomial's association with the equivalent binary form is shown in the Figure 4.4.

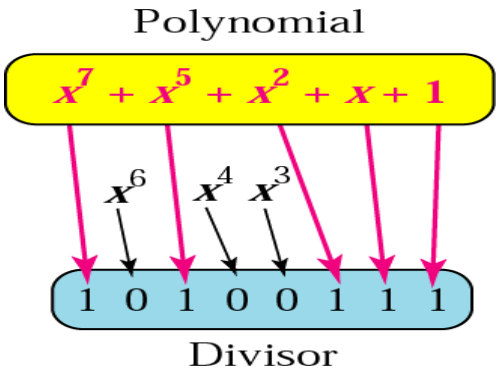


Figure 4.4: polynomial's association with the equivalent binary form.

The identical divisor established at the sender-side is used to divide data using CRC Remainder at the receiver-side. If the data is not altered, the remainder will unquestionably be zero.

CRC's performance:

Any burst mistakes that involve an odd number of bits may be found using CRC any burst faults with lengths less than or equal to the degree of the polynomial may be found using CRC (Figure 4.5).

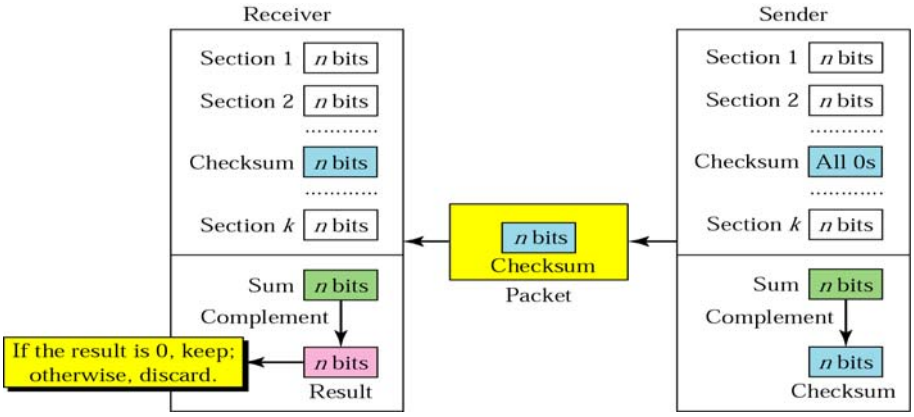


Figure 4.5: Represent the Checksum

1. Checksum: A higher layer's tool it recognizes both even-bit mistakes and errors involving an odd number of bits. It keeps all carry while adding. The sender performs the following actions: Splitting the data unit into k pieces, each with n bits.
2. The total is calculated by adding each part.
3. The checksum is created by complementing the sum.
4. The data is delivered together with the checksum.
5. The recipient takes the following actions:
6. The received data unit is split into k pieces, each with n bits.

7. The total is calculated by adding each part.
8. The total is completed.
9. The data are accepted if the result is zero; else, they are discarded.

There are two methods for handling error correction.

1. The receiver may request that the sender resend the complete data unit if an error is found.
2. An error-correcting code may be used by a receiver to automatically fix certain faults.

Protocols for elementary datalinks:

A. A Simplex Unrestricted Protocol
B. A simplex protocol for waiting
C. A Noisy Channel Simplex Protocol

Protocols for Sliding Windows: Having two distinct communications channels and using each one for one way is not a viable approach when there is a requirement for delivering data in both directions.

Using the same circuit for data in both ways is a superior concept. Piggybacking: Piggybacking is the process of momentarily postponing outgoing acknowledgements so they may be hooked onto the next outgoing data frame.

Better use of the channel bandwidth is an advantage.

Instead of sending a second control frame right away after a data frame arrives, the receiver waits until the network layer delivers it the subsequent packet. The outgoing data frame has the acknowledgment attached to it.

Waiting period: The acknowledgment is piggybacked onto the next packet if it comes fast; otherwise, if no new packet has arrived by the end of this time period, the data connection layer simply transmits a separate acknowledgement frame.

Window for Sender & Recipient

When sliding window protocols are used

The sender keeps a list of sequence numbers for the frames that it is allowed to transmit. Moreover, the receiver keeps a receiving window that corresponds to the frame set it is allowed to accept. The lower and upper bounds as well as the size of the sender's and receiver's windows are not had to be the same. They may increase or decrease over time when frames are broadcast and received in certain protocols, which have predetermined sizes, and in others.

Sent from: It includes transmitted but unrecognized frames. The top edge within the sender window is increased in response to new packets from the host. The bottom border within window is increased when the receiver acknowledges a frame. Every frame in the sender's window has to be kept for potential retransmission, and we need a timer for each frame.

The sender requires B buffers if the sender window's maximum size is B.

The protocol must turn off the host (the network layer) if the sender window fills up and exceeds its maximum window size. This is done until buffers are made available.

Receiver window: Sequence numbers outside the receiver window are not allowed in frames received. Typically, the receiver window size is constant. As "approved" frames are received, the list of permissible sequence numbers is cycled.

Go Back N: The protocol only takes frames in sequence since the receiver's window size is 1. Go Back N. is the name of this disadvantage. The frames that follow a faulty frame are repeated unnecessarily judicious repetition (also called selective reject)

Maximum window size is half the sequence number range. Sender only resends messages that were misdirected.

Advantage: improved performance in noisy channels

Disadvantage: Larger buffers, a more difficult algorithm, and more expenses

---

CHAPTER 5

TRANSPORT LAYER

Revathi Theerthagiri, Assistant Professor
Department of Computer Science and Information Technology, Jain (Deemed to be University) Bangalore, Karnataka, India
Email Id- revathigiri13@gmail.com

The full message must be sent from one process to another through the transport layer. A network application executing on a host is known as a process. Although the network layer controls the transportation of packets from source to destination, it is blind to any connections between the packets. The host application receives the whole message thanks to the transport layer.

Services & Duties at the Transport Layer:

Delivery from Process to Process

Congestion control, connection control (TCP or UDP), flow control, segmentation, reassembly, multiplexing and demultiplexing, simultaneous usage of multiple programmes, error control

UDP: User Datagram Protocol

It is known that the user datagram protocol (UDP) is an unstable, connectionless transport mechanism. The only enhancement to IP's services is the availability of process-to-process communication in place of host-to-host communication (Figure 5.1).

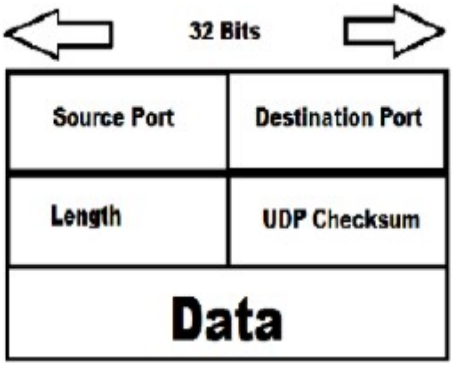


Figure 5.1: The following is the UDP packet structure

UDP Operations: UDP makes advantage of transport-layer-related ideas. Connectionless Services: Since UDP offers connectionless services, each user datagram delivered through the protocol is a separate datagram. Even if they originate from the same source process and are directed towards the same destination process, there is no connection between the various user datagrams. Each user datagram might follow a distinct route since there is no connection establishment or termination, no numbering of the user datagrams, and no connection setup. Data cannot be transmitted in a continuous stream; it must be broken up.

Control of flow and errors: UDP is a relatively straightforward, unstable transport protocol that doesn't provide flow control. The receiver can be flooded with messages. As UDP lacks any error-control mechanisms except checksums, the sender cannot determine if a message has

been lost or copied. The user datagram is secretly destroyed by the receiver when the checksum reveals an error.

Encapsulation and Decapsulation: The UDP protocol encapsulates and decapsulates messages in an IP datagram to deliver a message from one process to another.

Queuing: In UDP, queuing essentially means asking for a port number for client processes and utilising that port number to send messages from process to process.

UDP Uses and Characteristics:

UDP enables extremely straightforward data delivery without error checking. As a result, it may be utilised in networking applications like VOIP, streaming video, etc. where a little packet loss can still be accepted and the system will still perform as intended.

Unlike TCP, which lacks the multicasting functionality, UDP software is suited for multicasting. UDP is used by the Dynamic Host Configuration Protocol (DHCP) to assign IP addresses to computers in a dynamic manner. UDP is also used by various route update protocols, such as RIP.

UDP is the best protocol for network applications like gaming and audio and video communications, where latency is crucial. Transmission Control Protocol (TCP): Similar to UDP, TCP is a transport layer protocol for communication between processes. It is a trustworthy transport protocol that is connection-oriented. To transfer data, TCP establishes a fictitious connection between two TCP clients. At the transport level, TCP also makes use of flow and error control methods.

TCP Operations:

The following list of TCP services and operations is provided:

Process-to-Process Communication: Similar to UDP, TCP offers port-based process-to-process communication.

Stream Delivery Service: TCP is a protocol that is stream-oriented. It enables data delivery as a stream of bytes for the sending process and data acquisition as a stream of bytes for the receiving process. The two processes seem to have a separate connection that transfers their data over the internet thanks to TCP's environment-creating techniques. TCP employs sending and receiving buffers for flow control, which provide some storage for data packets in case of overflow. This synchronises the flow rate, preventing packet loss.

Full-Duplex Communication: TCP provides full-duplex services, allowing simultaneous data transmission in both directions. Segments then go in both ways and each TCP has a transmitting and receiving buffer.

Connection-Oriented Service: The following takes place when a process at site A wishes to communicate and receive data from another process at site B:

1. A connection is established between the two TCPs
2. Both directions exchange data
3. The connection is broken
4. Keep in mind that this is a virtual and not a physical connection

Service Reliability: TCP is a trustworthy transport protocol. It makes use of an acknowledgment mechanism to verify that data arrived safely. Because of effective error control techniques, this is achievable.

Attributes and traits of TCP: The following characteristics are necessary to support TCP's services:

Numbering System: TCP records the sequence numbers of segments sent or received. For numbering the bytes inside the segments and acknowledgements, respectively, there are two fields called the sequence number and the acknowledgement number.

Flow regulation: TCP offers a flow regulation technique. The quantity of data that must be delivered by the sender is determined by the recipient. To avoid overloading the receiver with data, this is done. TCP is able to apply byte-oriented flow control because of the numbering scheme.

Error Control: TCP uses an error control system to provide dependable service. Error control is byte-oriented even though the error detection system uses segments as the unit of data (lost or damaged segments).

Control of Congestion: TCP takes network congestion into consideration. In addition to being controlled by the receiver (flow control), the degree of network congestion also affects how much data a sender sends.

Shaking hands three times: TCP needs three phases: connection formation, data transmission, and connection termination since it is a connection-oriented service.

Three-way handshaking is the term used to describe the TCP connection setup. The handshaking gesture is seen in the Figure 5.2.

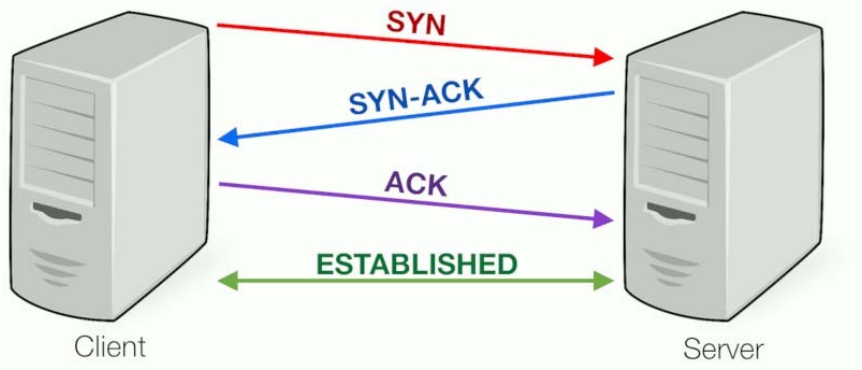


Figure 5.2: shows the handshaking process.

The termination of a connection may also be accomplished by three-way handshaking.

Important distinctions between TCP and UDP:

1. In contrast to UDP, which is a connectionless protocol, TCP is connection-oriented.
2. TCP is slower than UDP and vice versa in terms of speed.
3. TCP is dependable because it ensures data transmission, but UDP is unreliable.
4. TCP employs handshake protocols like SYN, SYN-ACK, and ACK, but UDP does not.
5. In contrast to UDP, which does error checking but discards incorrect packets, TCP performs error checking and also provides error recovery.
6. Although acknowledgement segments are present in TCP, they are absent in UDP.
7. UDP is lightweight but TCP is heavy-weight.
8. TCP ensures that data packets reach the recipient in the correct sequence; UDP does not.
9. Unlike UDP, TCP does not allow broadcasting.

10. HTTP, HTTPS, FTP, SMTP, TELNET, and SMTP all utilise TCP, while DNS, DHCP, SNMP, RIP, and VoIP all use UDP.

Relationship-Oriented Services: Before data can be exchanged between the linked terminals in a connection-oriented service, a connection between peers must be established. This process is often referred to as a "reliable" network service. Compared to connectionless protocols, this one manages real-time traffic more effectively since it arrives in the same order as it was delivered. Moreover, connection-oriented protocols are less prone to errors. Users of connection-oriented services must follow a certain order of operations. Which are:

1. A connection has been made.
2. Communications are sent.
3. The connection is cut off.

With connection-oriented services, a connection must be established before any communication may take place. As soon as the connection is made, we transmit the message or the information before cutting it off. The TCP (Transmission Control Protocol) protocol is an example of a connection-oriented protocol.

Virtual Circuits: A virtual circuit (VC) is a technique for sending data over a packet-switched computer network such that it seems as if the source and destination end systems of the data are connected by a dedicated physical layer connection. The network layer only offers either a host-to-host connection service or a host-to-host connectionless service in all current main computer network topologies (Internet, ATM, frame relay, etc.).

Virtual-circuit (VC) networks are computer networks that only provide connection-oriented services at the network layer; datagram networks are computer networks that only offer connectionless services at the network layer. Unlike many alternative network topologies, such as ATM and frame relay, which employ connections at the network layer, the Internet is a datagram network. Virtual circuits are the name for these connections at the network layer (VCs). A virtual circuit has three distinct stages that may be identified: Data Transmission, VC Teardown, and VC Setup (Figure 5.3).

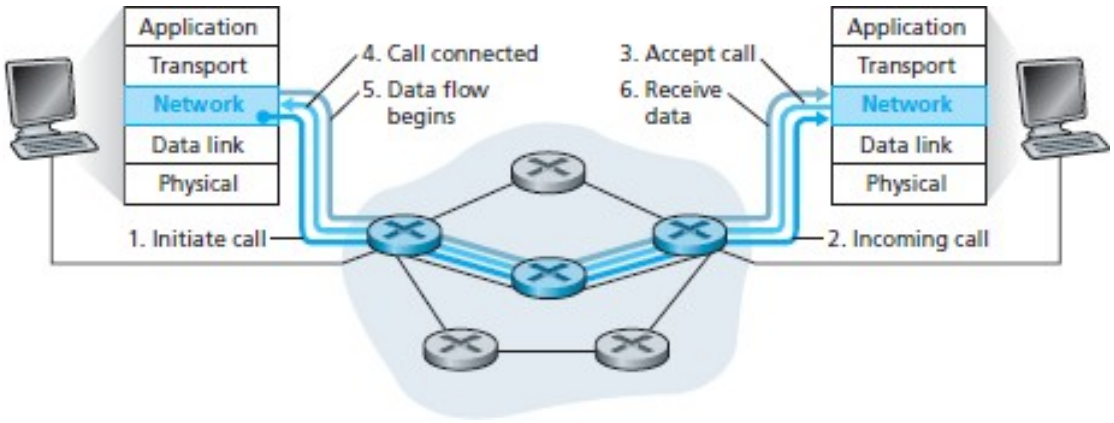


Figure 5.3: Virtual Circuit Setup

Services with fewer connections: A terminal or node may transfer data packets to a destination without first establishing a connection with it by using connectionless service. The sender just begins transmitting the data; a session connection between the sender and the receiver is not necessary. Without previous preparation, the message or datagram is transmitted, which is a less reliable but quicker transaction than a connection-oriented service. Since error handling

protocols allow for error repair methods like requesting retransmission, this works. Since it includes the whole address where the message (letter) is to be sent, it is comparable to postal services. Every communication travels on its own path from source to destination. It is possible for the sequence of messages transmitted and received to vary.

In reality, LANs are connectionless systems in which each computer is capable of sending data packets as soon as it has network access. The Internet is a sizable connectionless packet network where Internet service providers are in charge of all packet delivery. The UDP (User Datagram Protocol) protocol is an example of a connectionless service. Datagram networks are the connectionless services provided at the network layer. When an end system wishes to transmit a packet in a datagram network, it stamps the packet with the address of the target end system before releasing it into the network. Figure illustrates that there is no VC setup and that routers do not keep track of any VC state data (because there are no VCs) (Figure 5.4).

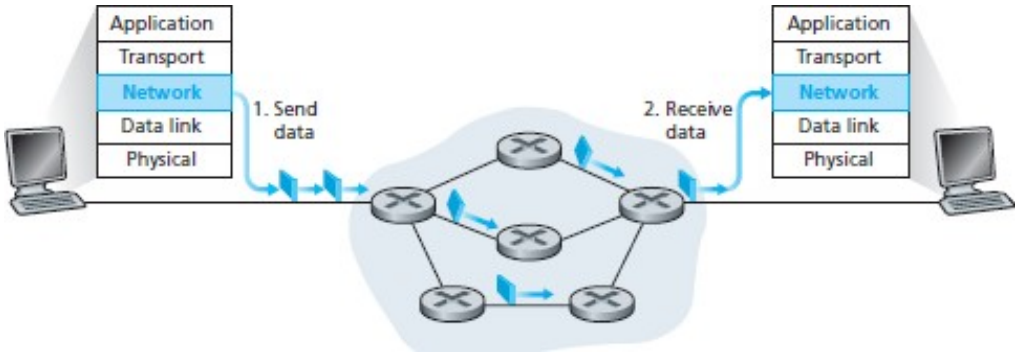


Figure 5.4: Datagram Network

Congestion management

When there are too many packets in (a section of) the network, performance suffers due to packet loss and delay. Congestion is the term for this circumstance. In other words, if a network's capacity is exceeded by the load on the network, or the number of packets transmitted to the network, congestion may result (the number of packets a network can handle).

Congestion control is a shared duty of the network and transport levels. Given that congestion happens inside the network, the network layer is the one who deals with it directly and is ultimately responsible for deciding what to do with the extra packets. Yet, lowering the load that the transport layer is putting on the network is the most efficient strategy to manage congestion. Congestion control describes the methods and procedures used to prevent congestion and maintain load levels below capacity (Figure 5.5).

1. Resulting from congestion
2. As delay grows, performance declines.
3. Retransmission happens if the delay grows, making the problem worse.

The term "congestion control" refers to methods and procedures that may either stop congestion in its tracks before it starts or relieve it after it has started.

The following are the general principles of congestion control:

Open Loop Theory: make an effort to avoid congestion; once the system is functioning, make no corrections

Closed-loop theory: keep an eye on the system for congestion, provide information to those who can take action; modify system behavior to address issues.

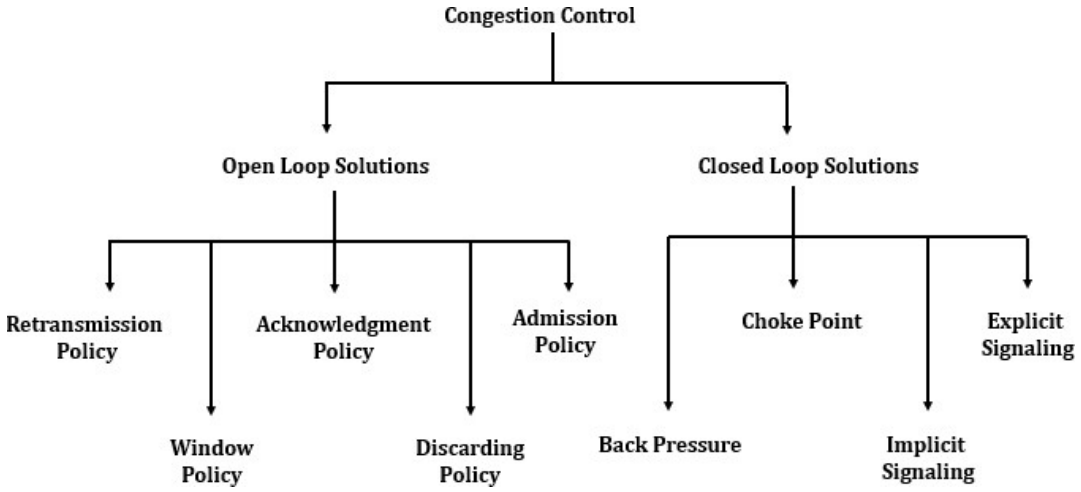


Figure 5.5: Congestion Control

Open Loop Congestion Control: With open-loop congestion control, regulations are put in place to stop congestion before it starts. With these techniques, either the source or the destination is in charge of congestion management.

The following are some of the regulations that may reduce traffic:

Retransmission Policy: This is the policy that governs how packet retransmissions are handled. A sent packet must be retransmitted if the sender believes it to be damaged or lost. The network congestion might become worse as a result of this communication. Retransmission times must be created with congestion prevention and efficiency optimization in mind.

Glass Policy: Congestion could also be impacted by the kind of window on the sender side. Despite the possibility of some packets being successfully received at the receiver side, many of the packets in the Go-back-n window are resent. This duplication might worsen the network's congestion by adding to it.

Consequently, because Selective Repeat Window transmits the precise packet that would have been lost, it should be used.

Congestion may be impacted by the acknowledgment policy that the receiver imposes since acknowledgements contribute to network load. There are many methods that may be used to avoid acknowledgment-related congestion.

Instead of sending acknowledgment for only one packet, the receiver should send acknowledgement for N packets. Only when a packet has to be sent or a timer needs to be reset should the receiver send an acknowledgement.

Discarding Guidelines: A good discarding strategy chosen by the routers is that the routers may preserve message quality while also preventing congestion and partly rejecting corrupted or less sensitive packages. In order to avoid congestion and preserve the audio file's quality during audio file transmission, routers might reject fewer sensitive packets.

Admission Procedure: Congestion control should be included in admissions policy. Before sending a network flow farther, switches in the flow should first determine if it has the necessary resources. The router shall refuse to make a virtual network connection if there is a

possibility of congestion or if there is already congestion in the network to avoid future congestion.

Congestion Control in Closed Loops: Since congestion has already occurred, closed-loop congestion management systems strive to lessen its impacts. Back Pressure: Backpressure is a mechanism where a crowded node prevents an upstream node from sending packets to it. As a result, receiving data from nodes above may be rejected and the upstream node or nodes may become overloaded.

Backpressure: Backpressure is a congestion management method that spreads from node to node in the reverse direction of data flow. The backpressure approach is only applicable to virtual circuits in which each node has knowledge of its upstream neighbouring node.

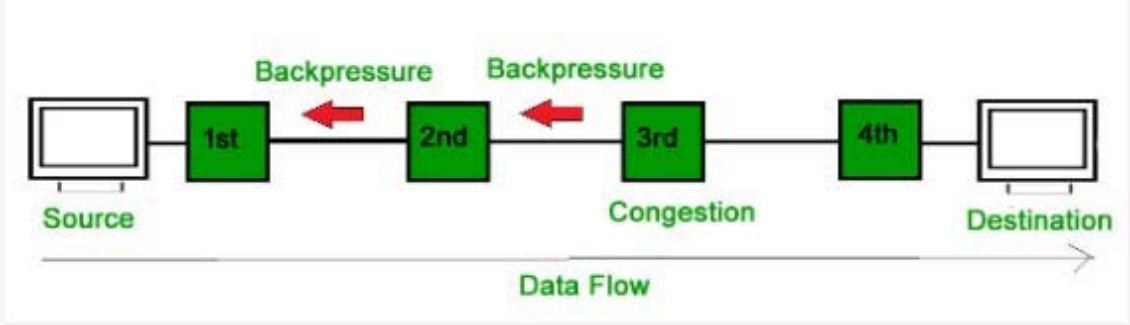


Figure 5.6: Represent the Backpressure

In the figure 5.6 above, the third node becomes overloaded and stops accepting packets, which may cause the second node to become overloaded as the output data flow slows. Similar to how the first node could get overloaded and tell the source to slow down.

Choke Packet: The choke packet approach may be used in datagram subnets as well as virtual networks. A choke packet is one that a node sends to the source to let it know there is congestion. Every router keeps an eye on the use of its resources and each of its output lines once the resource use goes over the cutoff point based on the administrator's settings, the router immediately sends a choke packet to the source, providing input to make the traffic less dense. There is no congestion notification for the intermediary nodes the packets passed through (Figure 5.7).

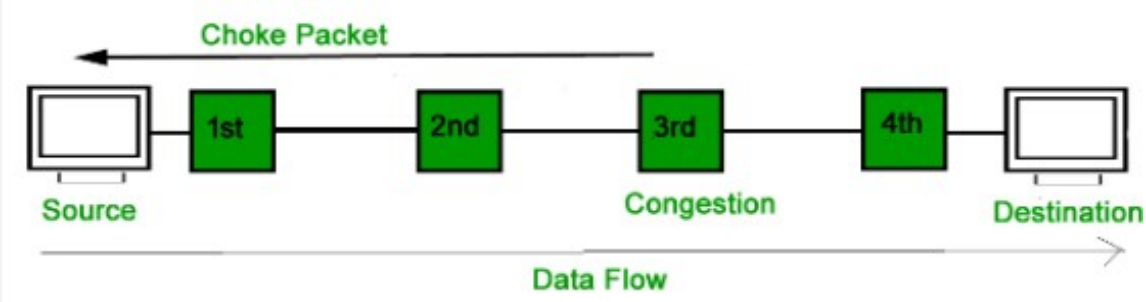


Figure 5.7: Illustration of Choke Packet

Implicit Signaling: This approach prevents communication between the source and the congested node or nodes. Based on other symptoms, the source surmises that there is network congestion someplace. When a source transmits multiple packets and waits a while before receiving an acknowledgement, for instance, one would assume that the network is busy and that the source should slow down.

Unambiguous Signaling: Using explicit signaling, a node may explicitly transmit a packet to the source or destination to advice about congestion if it encounters it. Contrary to the choke packet strategy, explicit signaling involves include the signal in the data packets itself rather than producing new packets specifically for the signal. Signaling that is explicit may go either forward or backward.

Forward Signaling: A signal is transmitted towards the direction of the congestion in forward signalling. Congestion is announced to the destination. In this scenario, the receiver initiates measures to stop future congestion.

Reverse Signaling: A signal is transmitted in the opposite direction of the congestion when backward signalling is used. The source is informed that it has to slow down due to traffic.

Controlling TCP Congestion: Congestion Control is a closed-loop-based architecture for connection-oriented services that may be used during connection setup in TCP (Virtual Circuit Subnet).

The fundamental idea is that we must ensure that congestion is avoided while setting up a virtual circuit. In order to manage congestion in TCP, the following technique is used:

Entrance Restrictions: No fresh virtual circuits may be created up after the congestion has been detected until the issue has been resolved. This method is often used in conventional telephone networks. No new calls are made while the exchange is overloaded.

An additional strategy:

To enable new virtual connections, carefully plan their routes to avoid the overloaded portion of the network and exclude any congested routers (or trouble areas) from the path. Another strategy is to have the host and the network negotiate distinct settings when the connection is made. The host provides the specifications for the traffic it would be providing to the network, including the volume and kind of the traffic, the quality of service, the maximum latency, and other factors.

Before the actual packet follows, the path's necessary resources are reserved once the host indicates its demand. Typically, one of the two crowded routers would be used as a minimum-hop route when router A establishes a connection to router B. Congested routers are removed from a temporary subnet in order to prevent congestion. Then, a virtual circuit may be created to prevent congestion.

Traffic management: It functions as a method to regulate the volume and speed of traffic transmitted to the network. Traffic may be shaped using the leaky bucket and token bucket strategies.

Bucket leakage algorithm:

1. Each host has an interface with a leaky bucket, or a limited internal queue, that connects it to the network.
2. A packet is dropped if it enters the queue at a time when it is already full.
3. It is really only a single server queuing system with a constant service time (Figure 5.8).

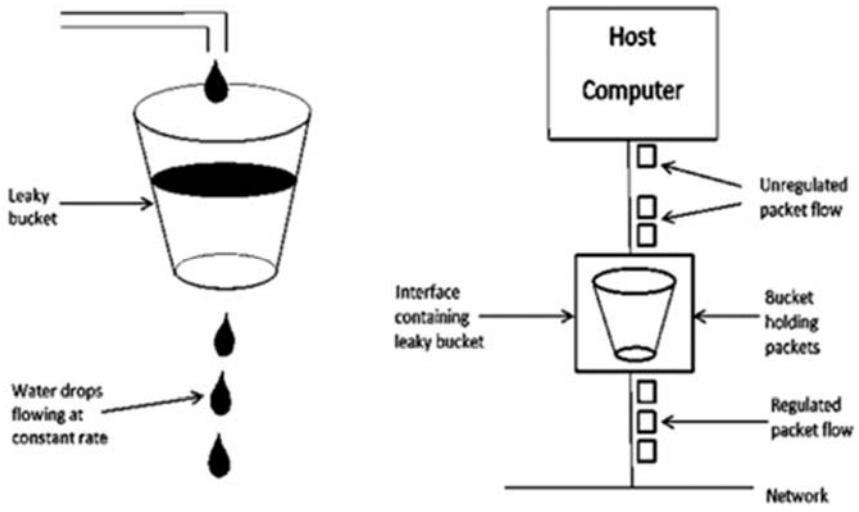


Figure 5.8: Bucket leakage algorithm

Bucket algorithm for tokens: The leaky bucket contains tokens that are produced by a clock one token per T seconds. A packet must capture and destroy one token in order to send it. The token bucket technique enables idle sites to accumulate authorization up to bucket size n maximum for surge traffic later (Figure 5.9).

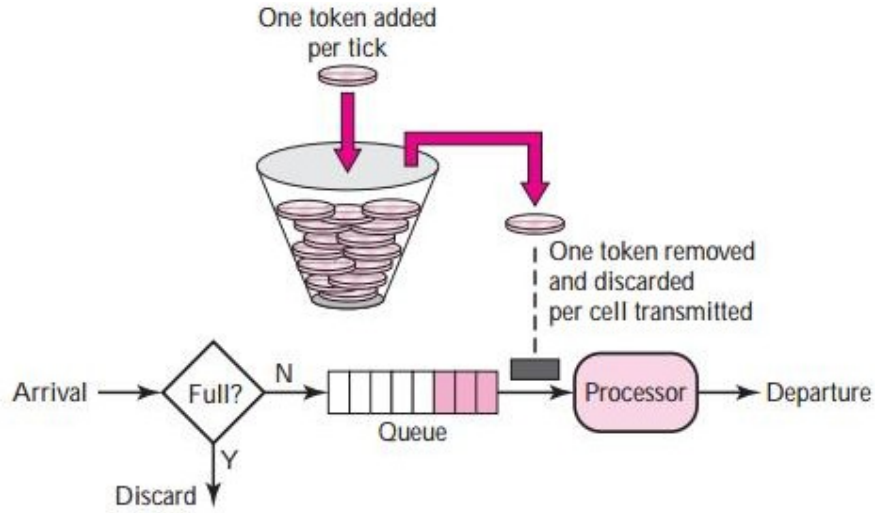


Figure 5.9: Token bucket algorithm

Scheduling Queuing Techniques: The scheduling of network traffic using QoS is known as QoS traffic scheduling (Quality of Service). A switch or router receives packets from various flows for processing. A smart scheduling method addresses the various flows equally and appropriately. Various scheduling strategies are intended to improve the level of service. There are three main queueing methods: FIFO, Priority, and Weight Fair.

FIFO Queueing: Packets wait in a buffer (queue) in first-in-first-out (FIFO) queuing until the node (router or switch) is prepared to handle them. The queue will become full and new packets will be deleted if the average arrival rate exceeds the average processing rate. Everyone who has had to wait at a bus stop for a bus knows what a FIFO line is like.

Prioritysing queues: Packets are initially given a priority class in priority queuing. Each type of priority has its own queue. The first packets processed are those in the queue with the greatest priority. The final packets to be processed are those in the lowest priority queue. The system continues to serve a queue until it is empty, it should be noted. A priority queue may provide greater quality of service (QoS) than a FIFO queue because higher priority traffic, such multimedia, can get to its destination faster. There might be a downside, however. The packets in the lower-priority queues will never get a chance to be processed if there is a constant flow in a high-priority queue. Starvation is the name for this situation.

Fair Weighted Queuing: Weighted fair queuing is a more effective scheduling technique. The packets are still allowed to various queues and allocated to various classes in this method. The queues are, however, weighted according to their priority; a greater priority corresponds to a larger weight. The quantity of packets processed from each queue is determined by the associated weight, and the system processes packets in each queue in a round-robin method. Three packets are processed from the first queue, two from the second queue, and one from the third queue, for instance, if the weights are 3, 2, and 1. All weights may be equal if the system does not give the classes any kind of priority. We have equitable queuing with priority in this manner.

Port and socket introduction (port addressing and socket addressing): A certain amount of data must go from a source to a destination host, and this requires both the IP address and the physical address. The goal of data transmission via the Internet is not always to reach the target site. A system is incomplete if it just transmits data between computers. Computers of today are machines that can execute many processes at once. A process interacting with another process is the goal of Internet communication.

For instance, TELNET may be used to connect computer A and computer C. Computer A and Computer B converse simultaneously using the File Transfer Protocol (FTP). We need a technique to identify the various processes in order for them to receive data concurrently.

They thus need addresses. The label given to a process in the TCP/IP architecture is referred to as a port address. In TCP/IP, a port address is 16 bits long. The IP packet contains source and destination addresses, which are part of the network layer. An IP packet is used to deliver a transport layer datagram or segment that employs port numbers. The network layer sends the packet over the network using the IP packet information (routing). Once reaching the destination host, the host's IP stack transfers the data to the application using the transport layer information (Table 5.1).

Table 5.1: Port Number with their Assignment

Port Number	Assignment
21	File Transfer Protocol (FTP)
23	TELNET remote login
25	SMTP
80	HTTP
53	DNS

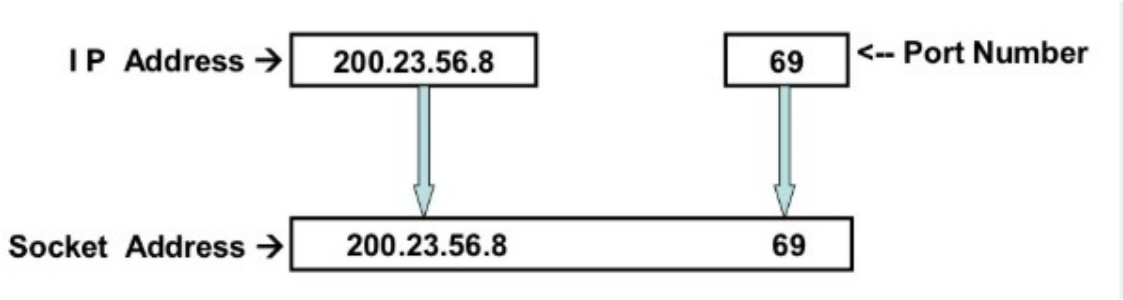


Figure 5.10: Socket Address

Socket Address, which refers to the IP address and Port Number combined, identifies the host and the networking programme operating on it (Figure 5.10).

One endpoint of a two-way communication channel between two network-running applications is a socket. In order for the TCP layer to recognise the application that data is intended to be transferred to, a socket is tied to a port number. A port number plus an IP address make up an endpoint.

Programming Sockets: A typical network application is made up of two separate end systems' worth of software a client programme and a server programme. A client process and a server process are established when these two programmes are run, and they interact with one another via reading from and writing to sockets. Socket programming, or writing the code for both the client and server applications, is the primary effort of a network application developer.

Network applications come in two different varieties. One kind is an implementation whose operation is detailed in a protocol standard, such an RFC or another standards document; this kind of application is sometimes referred to as "open" since the guidelines defining it are public knowledge. The client and server programmes for such an implementation must abide by the RFC's requirements.

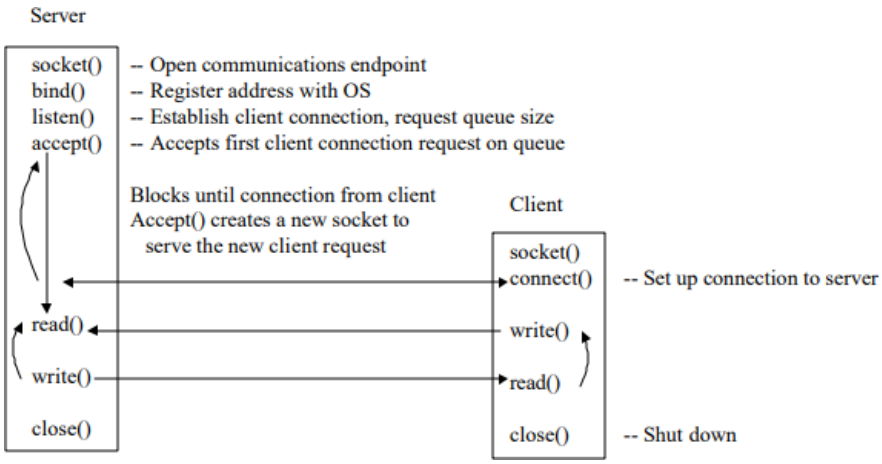


Figure 5.11: Representing the Rocket program.

A proprietary network application is the other kind of network application. The client and server programmes in this instance use an application-layer protocol that hasn't been publicly released in an RFC or anywhere else. Both the client and server applications are written by a single developer (or development team), who has total control over the code. However other independent developers won't be able to create code that interacts with the programme since the code does not use an open protocol.

The choice of whether to execute the programme via TCP or UDP is one of the first choices the developer must make throughout the development process. A socket application is immediately started to receive/send to the process when a web page is accessed. Using the corresponding source port and destination port numbers, the socket programme at the source computer connects with the socket software at the destination machine. The socket applications will be automatically stopped when a web page is closed (Figure 5.11).

When utilising socket programming for both TCP and UDP, the client-server application goes through the following series of events:

1. The client provides data to the server by reading a line of characters from its keyboard.
2. After receiving the data, the server changes the characters' case to uppercase.
3. The changed data is sent from the server to the client, who then gets it and displays the line on its screen.
4. The well-known socket programming API known as BSD Socket specifies a collection of common function calls that are accessible at the application level.

API-Applications Programming Interface, BSD-Berkeley Software Distribution

Programmers may include Internet communications capabilities in their products thanks to these features. Generally speaking, client/server architecture is used with BSD Sockets. One host monitors incoming connection requests for TCP communications. The server host will accept a request when it comes in, at which time information may be exchanged between the servers. Creating a connection via UDP is also permitted but not essential. Just sending or receiving data to or from a host is possible. Ports and sockets are the two methods used by the Sockets API to transmit data to the application level.

CHAPTER 6

INTER-DOMAIN ROUTING BASICS

Dr. Nagaraj S, Associate Professor
Department of Computer Science and Information Technology, Jain (Deemed to be University) Bangalore, Karnataka, India
Email Id- nagaraj.s@jainuniversity.ac.in

The Internet is a collection of autonomous systems that establish the governing principles and routing practises of many organisations. Interior Gateway Protocols (IGPs) like Routing Information Protocol (RIP), Enhanced Interior Gateway Routing Protocol (EIGRP), Open Shortest Path First (OSPF), and Intermediate System-to-Intermediate System (IS-IS) are used by routers in autonomous systems, while an Exterior Gateway Protocol is used for interconnection (EGP). The Border Gateway Protocol Version 4 (BGP-4), described in RFC 17711, is the de facto Internet standard EGP at the moment.

Summary of Routing with Routers

Devices called routers are used to direct traffic between hosts. They create routing tables that include information gathered on all the top routes to all the locations they are aware of. The following are the stages for basic routing:

Step 1 Routing protocols are programmes that routers use to send and receive route data to and from other routers in the network.

Step 2 Routers utilise this data to fill the routing tables linked to each specific routing protocol.

If more than one routing protocol is active, step 3 routers check the routing tables from each protocol and choose the optimal route(s) to each destination.

Step 4: Routers connect the local outgoing interface to be utilised for forwarding packets to the destination and the next-hop device's associated data link layer address to the destination. Remember that the target host or another router might be the next-hop device.

Step 5: The router's forwarding database is updated with the forwarding information for the next-hop device (data link layer address and outgoing interface).

Step 6 A router determines the destination address of a packet by looking at its header when it gets it.

Step 7: The router looks for the destination's outgoing interface and next-hop address in the forwarding table.

Step 8: The router completes any extra tasks that are necessary (such reducing the IP TTL or adjusting the IP TOS settings), and then it transmits the packet to the proper device.

Step 9 Repeat this up till you reach the target host. This behaviour reflects the hop-by-hop routing model that packet switching networks often use.

Since IGPs do not scale effectively in networks with thousands of nodes and hundreds of thousands of routes, which are networks that go beyond the corporate level, EGPs, like BGP, were established. IGPs were never meant to be used in this way. Concepts of routing protocols In general, link-state or distance vector distributed routing algorithms form the foundation of

the majority of routing protocols in use today. We'll go through the various characteristics of the distance vector and link-state routing algorithms in the sections that follow.

Protocols for distance vector routing: The Bellman-Ford moniker for distance vector protocols honours the creators of the technique used to determine the shortest paths2 and the individuals who first reported a distributed use of the algorithm3. The phrase "distance vector" comes from the fact that each destination prefix routing message in the protocol is accompanied by a vector (list) of distances (hop counts or other metrics).

The path to each destination prefix is calculated using a distributed calculation method via distance vector routing protocols like Routing Information Protocol (RIP). To put it another way, distance vector protocols demand that every node choose the optimum route (output link) to every destination prefix independently.

A router communicates distance vectors to its neighbours after deciding on the optimum route, informing them on the reachability of each destination prefix and the appropriate metrics related to the route it has chosen to reach the prefix. Its neighbours likewise choose the optimal route to each destination that is open in parallel, and they then inform their neighbours of the route they have chosen (along with any relevant metrics) to get there. The router may conclude that a better path is available through a different neighbour after receiving messages from neighbours describing the destination and related metrics that the neighbour has chosen. The router will once again inform its neighbours of the pathways it has chosen (along with any relevant metrics) to take to get to each destination. This cycle keeps on until every router has come to an agreement on the most effective routes to take to reach every target prefix.

RIP Version 1 (RIP-1) and other distance vector routing systems' initial specifications had a number of shortcomings. For instance, RIP-1 solely considered hop count while choosing a route. This placed a number of restrictions. Have a look at 4-1's RTA routing tables, for instance. These tables show the routing data that is taken into account while utilising RIP and OSPF, respectively. (The link-state routing protocol OSPF will be covered in greater depth in the sections that follow.)

To connect to network 192.10.5.0 while utilising RIP-1, RTA would choose the direct connection between RTA and RTB. The direct approach only takes one hop via the RTB path as opposed to two hops through the RTC-RTB path, which is why RTA favours this link. Yet, RTA is unaware that adopting the RTC-RTB way would provide a higher quality of service since the RTA-RTB link is really a very low-capacity, high-latency connection.

Nevertheless, when employing OSPF and metrics other than hop count alone for route selection, RTA will discover that the direct connection to RTB is actually less efficient than the way through RTC (cost: 60 + 60 = 120; 2 hops) (cost: 2000; 1 hop). The constraint on counting to infinity with hop counts is another problem. A route is deemed inaccessible once a certain number of hops—often 15, in the case of traditional distance vector protocols like RIP-1—have been reached. The propagation of routing changes would be constrained as a result, which would be problematic for big networks (those with more than 15 nodes in a given path). One drawback of distance vector protocols is their dependency on hop counts, while newer distance vector protocols, such RIP-2 and EIGRP, aren't restricted in this way.

The manner that routing information is communicated is another flaw. The foundation of conventional distance vector protocols is the idea that routers periodically exchange distance vector broadcasts, which are broadcasts that are delivered when a "refresh timer" connected to the message exchange expires. When a new piece of routing information is broadcast to your neighbours when the refresh timer ends, the timer is reset and no new information is provided

until the timer expires again. Now imagine what would happen if a link or route suddenly became inaccessible just after a refresh. Convergence would move very slowly since propagation of the route failure would be stopped until the refresh cycle ended.

Thankfully, triggered updates are introduced by more recent distance vector protocols like EIGRP and RIP-2. Failures are immediately propagated via triggered updates, greatly accelerating convergence.

As you may have guessed, periodic routing table exchanges between neighbours may get quite big and very difficult to maintain in large networks, or even small networks with a lot of destination prefixes, which can hinder convergence. Also, the periodic broadcast of routing information might require a significant amount of CPU and network overhead. Newer distance vector protocols also have the capacity to reliably transmit distance vectors between neighbours, doing away with the requirement to revert the whole routing table on a cyclical basis.

Convergence is the moment in time when the whole network is informed that a certain route has emerged, vanished, or modified. Previous distance vector protocols relied on hold-down timers and periodic updates to function: A route enters a hold-down status and is removed from the routing table if it is not received in a certain length of time. Prior to the whole network detecting that a route has vanished, the hold-down and ageing process results in minutes of convergence time. Temporary forwarding loops or black holes may form as a consequence of the time it takes for a route to go from being accessible to ageing out of the routing tables.

Another problem with certain distance vector protocols (such as RIP) is that the route is still placed in a hold-down state when an active route vanishes but then returns with a higher metric (likely coming from another router, signalling a potential "good" alternative path). As a result, it takes longer for the whole network to converge.

The classful nature of first-generation distance vector protocols and their lack of support for VLSM or CIDR are two further significant drawbacks. Certain distance vector protocols are unable to support these technologies because they do not share mask information in their routing updates. In RIP-1, a router will apply its locally defined subnet mask to any routing updates it gets on a specific interface. When a component of the transmitted network address does not match the local network address, IGRP performs the same function as RIP-1 but falls back to Class A, B, and C network masks. In the event that the interface is a part of a network with many subnets, this might cause confusion and result in a wrong interpretation of the routing update that was just received. Modern distance vector protocols like EIGRP and RIP Version 2 (RIP-2) fix the aforementioned issues.

There have been a number of changes made to classic distance vector routing protocol behaviour to address their shortcomings. For instance, VLSM and CIDR are supported by RIP-2 and EIGRP. Moreover, IGRP and EIGRP may create more optimum pathways than just a hop count alone since they have the capacity to take into account composite metrics that describe connection properties along a path (such as bandwidth, usage, latency, MTU, and so forth).

The popularity of distance vector protocols is a result of their maturity and simplicity. The main problem with conventional distance vector protocol implementation is sluggish convergence, which may cause forwarding loops and black holes in traffic during topological changes. In reality, however, more recent distance vector protocols—most notably, EIGRP—converge extremely effectively. Without explaining that BGP is a distance vector, this piece wouldn't be comprehensive. BGP uses a further method known as the path vector in addition to the usual distance vector attributes to get around the count to infinity issue that was previously covered.

In essence, the path vector is a list of the routing domains (AS numbers) that the route has passed through. A domain ignores a route if it comes in with its domain identifier already mentioned in the path. This route information offers a way for pruning routing loops. Moreover, domain-based policies may be utilised with it.

Protocol for link-state routing

Link-state routing methods, like Open Shortest Path First (OSPF) and Intermediate System-to-Intermediate System (IS-IS), are regarded as more complicated routing protocols since they make use of a replicated distributed database model. Link-state protocols are based on the idea that routers exchange data packets known as link states that include details about connections and nodes in the routing domain. As opposed to distance vector protocols, link-state protocols do not need routers to share routing tables. Instead, they incorporate metric data related to the connection and share information about nearby neighbours and networks.

Link-state routing protocols may be thought of as a jigsaw puzzle. Each router in the network creates a puzzle piece called a "link state" that details the router itself and the locations of its connections to other puzzle pieces. Also, a list of the metrics pertaining to the relationship with each component of the puzzle is provided. After every node in the domain has obtained a copy of the puzzle piece, the local router's piece is successfully propagated over the network, router by router, using a flooding method. Every router in the network has a copy of each jigsaw piece when distribution is finished, and the pieces are kept in a link-state database. The complete puzzle is then built independently by each router, resulting in an exact replica of the full problem on every router in the network.

Each router then determines a tree of the shortest routes to each destination, putting itself at the root, using the SPF (shortest path first) method, which is most often the Dijkstra Algorithm, to solve the problem.

The following are some advantages link-state protocols offer:

1. No maximum hop count: A route may take as many hops as it likes. Instead of using hop counts, link-state protocols operate on the basis of link metrics.
2. Bandwidth representation: When determining the shortest route to a certain location, link bandwidth and delays may be taken into account (manually or dynamically). As a result, load balancing is improved and is based on real connection cost rather than hop count.
3. Improved convergence: Link-state modifications rapidly flood the domain with changes to links and nodes. Instantaneously, all routers in the domain will update their routing tables (some similar to triggered updates).
4. Support for VLSM and CIDR: As part of the information components that are flooded into the domain, link-state protocols share mask information. Networks with variable-length subnet masks may therefore be recognised with ease.
5. Improved hierarchy: In contrast to flat networks like distance vector networks, link-state protocols provide methods for segmenting the domain into several levels or regions. This hierarchical method more accurately pinpoints network instability within regions.

Link-state algorithms should only be utilized for interior routing, despite the fact that they have historically offered higher routing scalability and may thus be employed in larger and more complicated topologies. Link-state protocols by themselves are unable to provide the kind of worldwide connection needed for Internet interdomain routing. Any one router will not be able

to manage link-state retransmission and recomputation in extremely large networks or in the situation of route oscillation brought on by link instabilities.

Interconnections, Second Edition: While a more thorough examination of IGPs is beyond the purview of this book, there are two great sources that go through the various link-state and distance vector routing protocols.

OSPF: Anatomy of an Internet Routing Protocol and Bridges, Routers, Switches and Internetworking Protocols6 by Radia Perlman. Due to their ability to quickly converge, link-state routing protocols are being used for intra-AS routing by the majority of big service providers. The two most often used protocols in this area are OSPF and IS-IS.

Several more recent service providers use OSPF or IS-IS as their IGP, whereas many older service providers have chosen IS-IS. Older networks seem to utilise IS-IS rather than OSPF at first glance since the U.S. Government needed networks to support ISO CLNP in order for the networks to be given government contracts. (Note that although OSPF may convey just IP information, IS-IS can carry both CLNP and IP Network layer information.) Internet mythology, however, states that the motivating cause was that, when early providers were choosing which routing protocol to deploy, IS-IS implementations were far more reliable than OSPF implementations. It is clear that the stability had a big influence on the choice of IGP service providers.

In ISP networks nowadays, both IS-IS and OSPF are frequently used. Because of IS-reliability IS's and scalability, it is still used in many big networks and is the IGP of choice for certain more modern networks.

Creating Autonomous Systems throughout the World: In order to limit the growth of routing tables and to provide a more organised view of the Internet, exterior routing protocols were developed. These protocols divide routing domains into separate administrations, known as autonomous systems (ASs), each of which has its own independent routing policies and distinctive IGPs.

An outside gateway protocol known as EGP8 (not to be confused with Exterior Gateway Protocols generally) was used in the early days of the Internet. Reachability data was sent between the regional networks and the backbone via EGP on the NSFNET.

While EGP was often used, its topological limitations and ineffectiveness in handling routing loops and establishing routing rules led to the need for a new and more reliable protocol. The Internet's de facto interdomain routing standard at the moment is BGP-4. The main distinction between intra-AS and inter-AS routing is that the former is typically tailored to meet the necessary technical requirements, whilst the latter often reflects the political and commercial ties between the networks and enterprises involved.

Routing types include static, default, and dynamic. Before presenting and examining the fundamental manner in which autonomous systems might connect to ISPs, it is necessary to define several fundamental terms and ideas related to routing:

Static routing is the practise of manually entering, or statically, as the name suggests, routes to destinations into a router. In this instance, network reachability is not reliant on the presence and functionality of the network. The routing database still contains static routes for each destination, and traffic is still sent there whether it is active or not.

A "last resort" outlet is referred to as default routing. The router receives traffic to unknown destinations at that default output. The simplest kind of routing for a domain linked to a single exit point is default routing.

Dynamic routing is the process of learning routes using an internal or external routing protocol. Reachability across a network depends on the network's existence and condition. If a destination is unavailable, the route is removed from the routing database and no traffic is sent there.

These three routing strategies are options for all the AS configurations that will be discussed in the next sections, although often one is best. Consequently, this explores whether static, dynamic, default, or any mix of these is appropriate in order to illustrate various autonomous systems. The appropriateness of inner vs external routing techniques. Due to the fact that they do not employ dynamic routing, many people believe they are not technologically advanced. It is a waste of bandwidth, time, and resources to try to enforce dynamic routing in circumstances that do not call for it.

Intelligent Systems: An autonomous system (AS) is a group of routers that share a common IGP, operate under a single technical administration, and have a single routing policy (the AS could also be a collection of IGPs working together to provide interior routing). The whole AS is seen as a single entity by the outside world. Each AS has a unique number that is allocated to it by either an Internet Registry or, in the case of private ASs, a service provider (Figure 6.1).

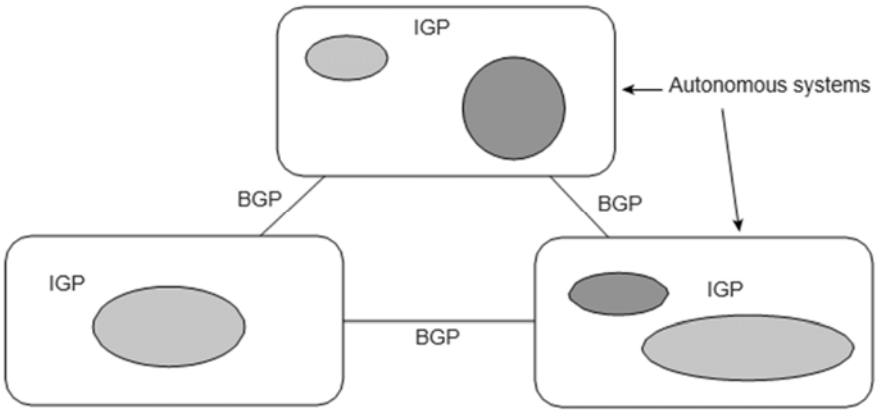


Figure 6.1: Routing Information Exchange between Autonomous Systems.

A stub: When an AS only has one exit point and connects to networks outside of its own domain, it is said to be stub. Regarding other providers, these ASs are also known to as single-homed. The single-homed or stub AS is shown in 6.2.

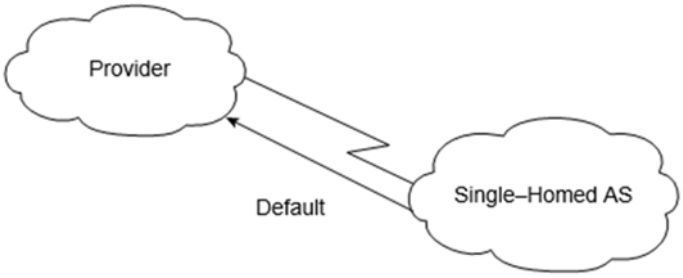


Figure 6.2: Single-Homed (Stub) AS.

It is unnecessary for a single-homed AS to learn Internet routes from its provider. There is just one exit, thus all traffic may go to the provider by default. While employing this configuration, the provider may inform other networks about the customer's routes via a variety of techniques.

One option is for the provider to add the customer's subnets to its router as static entries. After that, the provider would use BGP to promote these static entries to the Internet. If the customer's routes can be summed up into a limited number of aggregate routes, this strategy would scale extremely well. Static routes become ineffective when the client has a large number of noncontiguous subnets. As an alternative, the provider might use IGPs to promote the networks of the consumer. In order for the customer to promote its routes, an IGP might be employed between the client and provider. This offers all the advantages of dynamic routing, where updates and information about the network are delivered to the provider on demand. This is not typical, however, mainly because customer connection instability may lead to IGP instabilities, which means it doesn't scale well.

Using BGP between the consumer and the provider is the third way the ISP may discover and promote the routes of the customer. Since the customer's routing rules are an extension of the policies of a single provider, it is challenging to get a registered AS number from an IRR in the stub AS scenario. Alternatively, if the provider's routing rules have included support for utilizing private AS space with clients, as explained in RFC 227010, the provider may assign the customer an AS number from the private pool of ASs (65412–55535).

There are several possible protocol combinations that may be utilized between the ISP and the consumer. A few of the potential configurations are shown in Figure 6.3 using merely stub ASs as an example. (In the sections that follow, the definitions of EBGP and IBGP will be covered.) Both client routers and provider routers may be extended to the customer's network by the service provider. As indicated before, not all circumstances necessitate that a client run BGP with its supplier.

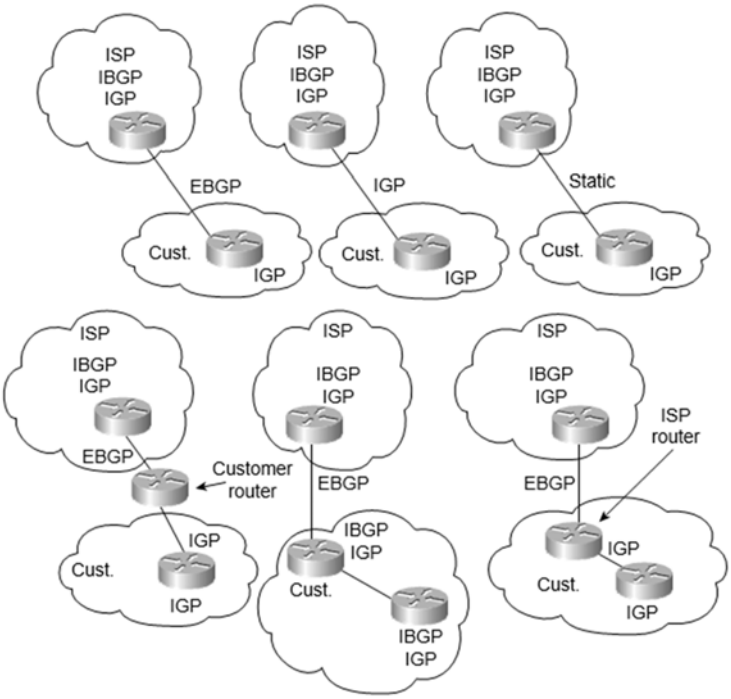


Figure 6.3: Example Protocol Implementation Modifications Stub AS.

Nontransit Multihomed AS: If an AS has more than one way out into the outside world, it is multihomed. A single provider or a number of providers may be multihomed to an AS. A nontransit AS forbids transit traffic from passing through it. All traffic that originates and ends outside the AS is considered transit traffic.

Multihomed Transit AS: A multihomed transit AS May still be utilised for transit traffic by other ASs while having several connections to the outside world. All traffic with an origin and destination outside the local AS is considered transit traffic (in the context of the multihomed AS).

Although though BGP-4 is an outside gateway protocol, BGP updates may still be exchanged using it within an AS as a pipe. Internal BGP (IBGP) refers to BGP connections between routers inside an independent system, whilst External BGP refers to BGP connections between routers in different autonomous systems (EBGP). When they transport transit traffic across the AS, IBGP-running routers are referred to as transit routers.

Routes that a transit AS learns from another AS would be advertised to that AS. In this manner, the transit AS would be exposed to outside traffic. For their connections to other ASs and to protect their internal nontransit routers from Internet routes, multihomed transit ASs are encouraged to employ BGP-4. Not every router in a domain has to be running BGP; internal nontransit routers may send default routing requests to the BGP routers, reducing the number of routes these routers need to carry. Nonetheless, all routers typically carry a complete set of BGP routes internally in the majority of big service provider networks.

The foundation for Internet routing designs has been established by the Border Gateway Protocol. The administrative and political boundaries between organisations have been rationally defined by the division of networks into autonomous systems. Interior Gateway Protocols may now function apart from one another, however BGP is still used by networks to link and provide global routing.

CHAPTER 7

INTER-DOMAIN INTERNET ROUTING

Dr. Ramkumar Krishnamoorthy, Assistant Professor
Department of Computer Science and Information Technology, Jain (Deemed to be University) Bangalore, Karnataka, India
Email Id- ramkumar.k@jainuniversity.ac.in

The aim of this chapter is to describe how Internet routing works between various administrative domains. We go through how Internet service providers (ISPs) trade routing information, packets, and—most importantly—cash with one another, as well as how the services they procure from and provide to one another and their clients affect routing. The Border Gateway Protocol, Version 4 (BGP4, sometimes known as BGP for short), the Internet's current interdomain routing mechanism, is covered in detail. Finally, we talk about a few noteworthy routing system flaws and failures. These notes give up a lot of specificity in favour of clarity and broad generality, concentrating primarily on the crucial components of interdomain routing.

Distributed Systems:

IP addresses 32-bit values in IPv4 are used to identify network attachment points for Internet hosts. The employment of topological addressing by the Internet's network layer is the primary factor in its scalability. A connected network interface's IP address relies on where it is in the network topology, in contrast to an Ethernet address, which is always unique and identifies a host network interface regardless of where it is linked in the network. Any IP address of the form 18* in the Internet is in MIT's network, so external routers only need to maintain a routing entry for 18* to correctly forward packets to MIT's network. Topological addressing also enables routes to be summarised and exchanged by the routers participating in the Internet's routing protocols. There would be no chance of growing the routing system to hundreds of millions of hosts linked across tens of millions of networks housed in tens of thousands of ISPs and organisations without aggressive aggregation.

In their routing messages, routers utilise address prefixes to indicate a contiguous set of IP addresses. As an example, an address prefix of the type 18.31* indicates the 216-bit IP address range 18.31.0.0 to 18.31.255.255. To go to address prefix P, you may utilise the link on which you heard me say this, together with information about the route, are included in each routing message sent by a router to a nearby router (the information depends on the routing protocol and could include the number of hops, cost of the route, other ISPs on the path, etc.).

Figure 7.1 depicts an abstract, highly idealised view of the Internet. End-hosts connect to routers, which connect to other routers to create a nice connected graph of essentially "peer" routers. These routers work well together by exchanging "shortest-path" or similar information and enabling global connectivity through routing protocols. According to the same theory, there is a lot of redundancy in the graph created by the routers and their connections, and the Internet's routing algorithms are built to quickly identify flaws and issues with the routing substrate and find a workaround. Load-sensitive routing techniques using sophisticated routing protocols dynamically shift load from crowded pathways onto less-loaded paths in addition to avoiding failures.

Sadly, although being straightforward, this abstraction is profoundly deceptive in terms of the wide-area Internet. The fact that a large number of for-profit businesses provide Internet service and are often in competition with one another is the true narrative behind the Internet routing architecture. The urge for financial success, which often comes at the cost of one's competitors—the same individuals with whom one must cooperate runs against to cooperation, which is necessary for global connectedness. The practical implementation of this "competitive collaboration" (albeit there is much room for improvement) and how we may make things better make for an intriguing case study of how sound technological research can be influenced and constrained by market realities.

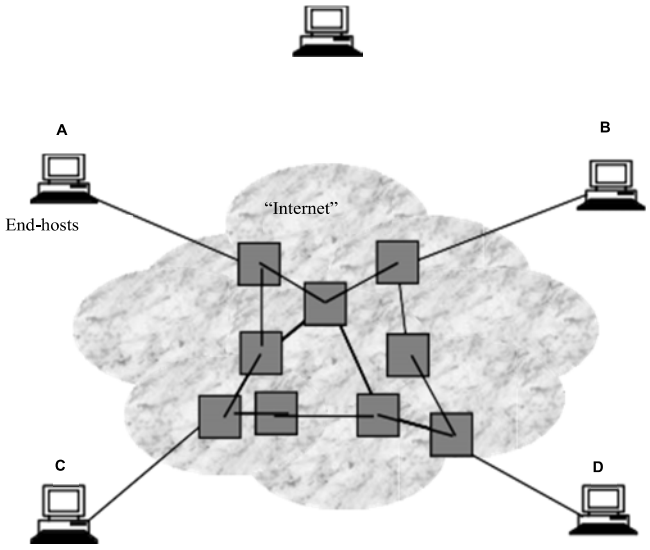


Figure 7.1: A rather misleading view of the Internet routing system.

A more accurate representation of the Internet infrastructure is shown in Figure 7.2. Here, Internet Service Providers (ISPs) work together to offer their various client networks with worldwide connectivity. The fact that ISPs don't all have the same internal structures and sizes is a crucial distinction to note. Some routing tables have global reachability, whereas others are larger and more "con- nected" than others. They come in three sizes: "little," "big," and "very gigantic," and these three sizes have names assigned to them. This is a simplified but helpful paradigm. Tier-2 ISPs typically have regional scope (e.g., state-wide, region-wide, or non-US country-wide), while Tier-1 ISPs, of which there are a small number (nine in early 2008), have global scope in the sense that their routing tables actually have explicit routes to all currently reachable Internet prefixes. Tier-3 ISPs have a relatively small number of typically localised (in geography) end-customers (i.e., they have no default routes). Figure depicts this company.

The phrase "route," which we describe as a mapping from an IP prefix (a set of addresses) to a link, was used in the preceding sentence. This mapping allows packets for any destination inside the IP prefix to be transmitted via the related link. The optimal path to reach each prefix is chosen by a router from among the route ads supplied by its nearby routers before adding such entries to its routing table.

Using a routing protocol, routers communicate with one another to exchange route ads. BGP (Border Gateway Protocol, Version 4) is the Internet's current wide-area routing system, which

connects routers at the ISP-to-ISP border. The wide-area routing architecture is more precisely separated into autonomous systems (ASes) that communicate reachability data. When selecting how to route packets to the rest of the Internet and how to export routes (its own, those of its clients, and other routes it may have learnt from other ASes) to other ASes, an AS is owned and run by a single commercial business. Each AS has its own own 16-bit number assigned to it.

Within each AS, a distinct routing protocol is used. Interior Gateway Protocols (IGPs) are a class of routing protocols that include RIP (Routing Information Protocol), OSPF (Open Shortest Paths First), IS-IS (Intermediate System-Intermediate System), and E-IGRP. BGP is an interdomain routing protocol, in contrast. Operationally, a major distinction between BGP and IGPs is that the former is focused on scalable reachability information exchange across ASes while enabling each AS to adopt independent routing rules, while IGPs are primarily focused on route metric optimization. IGPs often don't scale as well as BGP does in terms of the quantity of participants.

The remainder of this essay is divided into two sections: first, we'll examine inter-AS interactions (transit and peering); second, we'll examine some key characteristics of BGP. IGPs like RIP and OSPF won't be discussed; to understand more about them, read a normal networking textbook (e.g., Peterson & Davie or Kurose & Ross).

Associations between AS: Transit and Peering

There are many various kinds of ASes on the Internet, including corporate entities, regional ISPs, and national ISPs. ISPs are generally used by smaller ASes (such as colleges, businesses, etc.) to obtain Internet access. In turn, bigger regional ISPs with extensive "backbone" networks sell connection to smaller local ISPs.

It displays a few clients and an ISP, X, that is directly linked to a provider (from whom it purchases Internet access) (to whom it sells Internet service). The picture also displays two other ISPs that are directly linked to X and with whom X exchanges routing data via BGP (Figure 7.1).

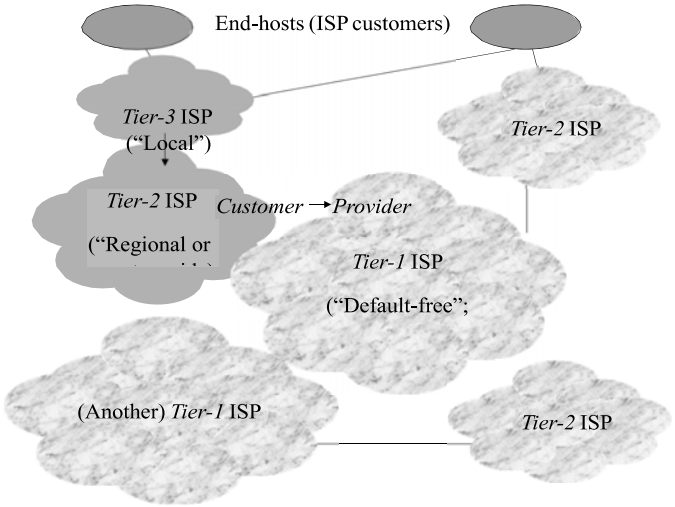


Figure 7.2: A more realistic representation of the wide-area Internet routing architecture, illustrating how different ISP types differ in terms of their geographic reach. Tier-1 ISPs have "default-free" routing tables (i.e., no default routes) and knowledge of the world's reachability.

Many sorts of commercial interactions between ASes result from the various types of ASes, and these relationships in turn result in various exchange and route selection rules. There are two types of AS-AS connectivity that are widely used. First is provider-customer transit (also known as "transit"), in which all (or the majority) of the destinations listed in an ISP's routing tables are accessible via that ISP. When there is a financial settlement involved in an inter-AS transaction, transit is almost always significant; the provider bills its users for Internet access in exchange for forwarding packets on their behalf to destinations (and in the opposite direction in many cases).

Peering is the second common kind of inter-AS interaction. In this case, two ASes (usually ISPs) exchange access to a portion of their routing tables. Own transportation clients are the subgroup that is of importance here (and the ISPs own internal addresses). Peering is a commercial transaction, much like transportation, except it may not include payment. While paid peering is widespread in various regions of the globe, it is often the result of reciprocal agreements. There is often no financial settlement as long as the traffic ratio between the involved ASs is not significantly asymmetric. Peering transactions are often secret and covered by non-disclosure agreements.

Transit vs. Peering: The fact that peering interactions sometimes include business rivals is important to keep in mind. Peer connections exist for two basic reasons (Figure 7.3).

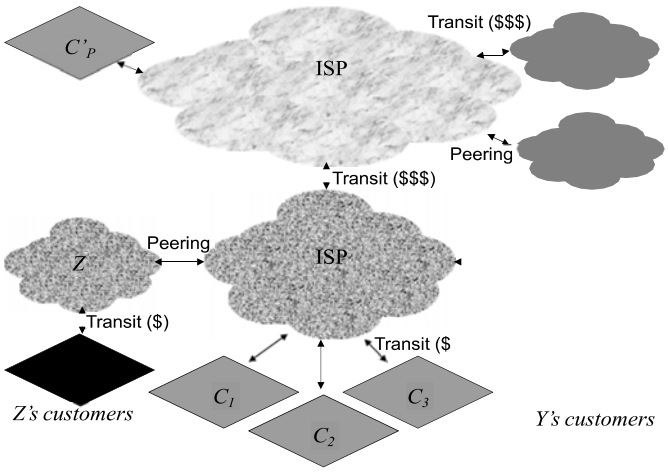


Figure 7.3: Common inter-AS relationships: transit and peering.

Tier-1 peering: Since cycles in the directed graph of ASes are illogical, an Internet with just provider-customer transit interactions would need a single (big) Tier-1 at the root (a cycle would require that the money paid by each AS to its provider would flow from an AS back to itself). Due to its control over all Internet traffic, that one Tier-1 would effectively have a monopoly (before the commercialization of the Internet, the NSFNET was in fact such a backbone). Many sizable Tier-1 ISPs were created as a result of commercial rivalry (nine today, up from five a few years ago). Since one ISP must be a specific size and manage a given amount of traffic in order to establish a peering arrangement with another Tier-1 provider, these ISPs essentially operate as a cartel. They have explicit default-free paths to all Internet destinations thanks to peering between Tier-1 ISPs (prefixes).

Financial savings: Peering isn't only for huge Tier-1s, however. Each AS has an incentive to avoid paying transit charges to its respective providers if a non-trivial fraction of the packets emerging from it or its customers are intended for another AS or its customers. The wisest

course of action for each AS would obviously be to wean away their respective client bases, but that may be challenging. The next best approach would be to avoid paying transit expenses to their separate providers and instead establish up a transit-free connection between them to forward packets for their direct consumers. This would be in both of their best interests. The benefit of this strategy is that it would provide its clients with superior end-to-end performance in terms of latency, packet loss rate, and throughput.

Peering is resisted by various factors that counteract these potential advantages. Although peering connections often don't, transit partnerships do. Asymmetric traffic ratios must be handled with caution in a fashion that is consistent with peering arrangements, which often need to be renegotiated frequently mutually agreeable. Above all, these connections often exist between rival businesses who are seeking for the same clientele.

We have already exploited an essential characteristic of existing intradomain routing in our discussion: An agreement by B that it would forward packets submitted through A and headed to any destination in the prefix is known as a route advertising from B to A for a destination prefix. This agreement (implicitly) suggests that one approach to think about Internet economics is to see ISPs as charging clients for entries in their routing tables. Of course, the interconnection's data rate is equally important and a key factor in determining an ISP's pricing strategy.

A note about price is necessary. ISPs impose one of two fees for Internet connection. The first is a set cost for a predetermined rate of access (fixed pricing is a common way to charge for home or small business access in the US). The second monitors traffic and assesses fees in accordance with the bandwidth used. A popular strategy is to gauge consumption throughout the course of the price period in 5-minute intervals (typically one month). These averages every five minutes represent samples that make up a distribution. Using the criteria outlined in the contract, the ISP determines the 95th (or 90th) percentile of this distribution and assesses a fee based on this number. Moreover, ISPs provide discounts for outages that are encountered; additionally, some ISPs now have delay assurances (through their network) built into the contract for certain types of traffic. It's unclear how far consumer networks may and do go in making sure that providers adhere to every clause included in their contract.

The basic lesson to be learned from this is that providers profit more from a client if they send more traffic on their behalf. This straightforward insight serves as the foundation for many complicated routing strategies that are used in real-world situations. We go through a few typical cases of routing policy in the next paragraphs of this section. Ways to Earn or Save Money by Exporting: With BGP, each AS (ISP) must decide which routes to export to its adjacent ISPs. Export regulations are crucial because no ISP wants to serve as a transit point for packets on which it is not in some way profitable. An AS should carefully advertise routes to neighbours since packets travel in the opposite direction from the (optimal) route advertising for each destination.

Routes used by transit users: Customer routes are perhaps the most significant to an ISP since they provide its clients the impression that they may be reached by any possible sender on the Internet. It is in the ISP's best interest to promote routes to as many other linked ASes as it can to its transit customers. The "fatter" the pipe that a client would need, indicating more income for the ISP, would be required to carry more traffic on their behalf. So, an ISP should favour the advertising from a customer above all other options if a destination were promoted from numerous neighbours (in particular, over peers and transit providers).

Will an ISP provide transit to the routes that its supplier exports to it? Very certainly not, since the ISP loses money by offering these transit services. Figure 7.3 illustrates this scenario, in

which CJP is a client of P and P has exported a route to CJP to X. Advertising this route to everyone, such as other ISPs with whom X has a peering arrangement, is not in X's best interest. Of course, a significant exception to this is provided by X's transportation customers, who pay X for the service that X offers to its clients. Ci's is that they may access any Internet location via. It is sensible for X to export as many routes to X as it can since X is X.

Peer pathways: The majority of the time, it makes sense for an ISP to merely export a subset of its routing tables to other peering ISPs. It makes logical to export routes to all of your transit clients, of course. Exporting routes to addresses within an ISP is also logical. To avoid a peering ISP using the advertising ISP to access a location promoted by a transit provider, it is not advisable to export an ISP's transit provider routes to other peering ISPs. Although doing so would use up ISP resources, no money would be made.

The same issue holds true for routes discovered via other peering connections. Take into account ISP Z, which has its own transit clients. As X doesn't profit from Y utilising X to deliver packets to Z's clients, it makes no sense for X to offer routes to Z's clients to another peering ISP (Y).

The arguments presented here demonstrate that the majority of ISPs end up offering selective transit: typically, full transit capabilities for their own transit customers in both directions, some transit (between mutual customers) in a peering relationship, and transit only for one's transit customers (and ISP-internal addresses) to one's providers.

The explanation thus far could give the impression that BGP is the sole protocol available for exchanging reachability data between an ISP and its clients or between two ASes. But, in reality, only a small percentage of end users—typically those that don't provide a lot of additional transit and/or aren't ISPs—run BGP sessions with their service providers. The reason is that BGP is difficult to set up, monitor, and administer and isn't very helpful if the customer's network's set of addresses is mostly deterministic. Via static routes, these clients communicate with their service providers. Often, these routes need manual configuration. Of course, in order to achieve global reachability to the client premises, a provider would often communicate data about customer address blocks with other ASes (ISPs) via BGP.

Choosing which routes to export is a crucial policy choice for the millions of networks that use BGP. Route filters, which are rules that determine which routes an AS's routers should filter to routers of adjacent ASes, are used to convey such judgements.

Making or Saving Money via Importing Routes: When a router hears many potential routes to a destination network, it must choose which route to import into its forwarding tables in addition to selecting how to filter them when exporting them. Ranking the routes to each destination prefix is the solution to the issue.

With BGP, selecting which routes to import involves a pretty sophisticated procedure that takes into account a number of the advertised routes' characteristics. For the time being, we focus on only one of the numerous factors an AS must take into account, but it's the most crucial one:

Normally, a router (such as X in Figure 3-3) has to make sure that packets going to its transit customers don't needlessly pass through other ASes when it receives ads about those customers from other ASes (for example, because the customer is multi-homed). The fundamental reason is that an AS wants to boost its perceived value among its direct customers and doesn't want to waste money paying its suppliers for traffic that is headed in that direction. According to this criteria, customer routes often take precedence over routes to the same network promoted by peers or suppliers. Second, because the goal of peering was to share reachability data about

shared transit customers, peer routes are probably better to provider routes. These two findings suggest that routes are frequently imported in the following priority order:

A consumer, a peer, and a supplier: Using a unique characteristic called the LOCAL PREF attribute, which is locally kept by routers in an AS, this rule (and many more like it) may be implemented in BGP. The LOCAL PREF characteristic should be used to rank routes when choosing one to use with BGP, and the route with the greatest value should be chosen. The ranking process only takes into account other properties of a route when this attribute is not specified for it.

Despite this, the majority of routes are rarely chosen in reality using the LOCAL PREF feature; instead, other properties, such as the length of the AS path, are more often used. While discussing the core concept of BGP, also known as the decision process, we go into depth about these additional route characteristics.

Routing Policy: Filtering + Ranking

While network operators express a vast variety of routing rules, the most of them may be reduced to ranking judgements and export filters, at least roughly speaking.

CHAPTER 8

BORDER GATEWAY PROTOCOL

Swati Sah, Associate Professor
Department of Computer Science and Information Technology, Jain (Deemed to be University) Bangalore, Karnataka, India
Email Id- iswatisah19@gmail.com

We will now examine how reachability data is sent via BGP and how routing rules similar to those described in the previous section may be stated and implemented. We begin by outlining the primary design objectives of BGP before summarising the protocol. Wide-area routing is most complicated when BGP routers are set up to apply policy and when routes that have been learnt from other ASes are distributed inside an AS. The remainder of the paragraph goes through these subjects.

The backbone routers of the previous NSFNET exchanged routing data using a tree topology and a routing protocol known as EGP (Exterior Gateway Protocol). The routing protocol was rather straightforward since the fundamental routing information was sent across a tree. The NSFNET EGP became outdated when the Internet transitioned from a solely managed backbone to its present commercial structure, necessitating a more advanced protocol. Three significant requirements drove the development of BGP:

Scalability. Although the NSFNet was then in charge of running the Internet's backbone, the Internet was split up into ASes with autonomous management. A variety of ISPs offering various sizes arose when the NSFNet was "switched off" in the early 1990s and the US Internet routing infrastructure became free to competition. Up to this point, the quantity of networks (and hosts) has increased. Routers need to be able to manage an increase in traffic to support this expansion.

BGP must guarantee that the volume of advertising traffic scales effectively with network "churn" (areas of the network falling down and coming back up), and it must quickly converge on loop-free pathways after any change. These objectives are difficult to achieve.

Policy. An essential design objective was to allow each AS to develop and enforce different types of routing policies. This led to the development of the BGP attribute structure for route announcements, which enabled route filtering and allowed each AS to arbitrarily rank its available routes.

Working together when there is rivalry. BGP was primarily created to deal with the change from the NSFNet to a scenario where the "backbone" Internet infrastructure was no longer controlled by a single administrative agency. This structure suggests that the routing protocol should let ASes to choose the most appropriate packet routing option from any collection of options. Moreover, BGP was created to let each AS maintain it’s filtering and ranking practises a secret from other ASes.

There was a compelling need to establish a workable routing system before the security narrative was completely developed, even though it was widely acknowledged that ensuring the authenticity and integrity of communications was a worthwhile aim. BGP security initiatives, most notably S-BGP, have been developed and entail the use of external registries, infrastructure, and public keys for ASes in order to preserve the mappings between prefixes and the ASes that possess them. These methods have not been implemented on the Internet for

a number of reasons, including the fact that users don't have a lot of faith in current routing registries since they often include mistakes and omissions.

Since routing to diverse destinations sometimes becomes messed up, misconfigurations and maliciousness may create connection failures. The next section provides some instances of previous routing issues that have made headlines; these instances serve as an excellent illustration of the adage that complex systems fail for complicated reasons.

Protocol Information: As far as protocols go, BGP is not very difficult. The protocol standard (RFC 4271, which replaces RFC defines the fundamental workings of BGP, including the protocol state machine, the structure of routing messages, and the propagation of routing changes. BGP use TCP on a widely used port. After establishing a TCP connection with another router on the BGP port, a router sends an OPEN message to begin participating in a BGP session with that router. Both routers share their tables of all active routes when the OPEN is complete (of course, applying all applicable route filtering rules). The information collected from each router's neighbor is subsequently included into its routing table. The whole procedure might take several seconds to a few minutes, particularly during sessions with a lot of active routes.

There are two primary sorts of messages on the BGP connection after this startup. Route UPDATE messages are first issued on the session by BGP routers. Just the routing entries that have changed since the previous update or the transmission of all active are sent in these updates.

There are two different types of updates: announcements, which include new or modified routes, and withdrawals, which include messages informing the recipient that the identified routes are no longer in operation. When a previously declared route can no longer be utilised, a withdrawal often occurs (e.g., because of a failure or a change in policy). There is no need to frequently publish routes since BGP employs TCP, which offers dependable and orderly delivery, unless they change. Yet, how can a router know if the neighbour at the other end of a session is still operating effectively in the absence of routine routing updates? BGP running over a transport protocol that implements its own "is the peer alive" message protocol may be one answer. These messages are also known as "keepalive" messages. Nevertheless, TCP does not (for good reason) include a transport-layer "keepalive," thus BGP utilises its own. The router commits to making at least one attempt to transmit a BGP message during each BGP session's defined keepalive duration. The router transmits KEEPALIVE messages as the session's second kind of communication if there are no UPDATE messages. The router ends a BGP session if it doesn't receive enough BGP KEEPALIVE signals to maintain it. The hold timer, which may be configured, affects how many messages are missed. The specification suggests that the hold timer be at least as long as the keepalive timer duration agreed upon for the session.

You may read more about the BGP state machine in BGP does not simply optimise any metrics, such as shortest-paths or latency, unlike many IGPs. Its announcements do not only publish some statistic like hop-count since its objectives are to give reachability information and enable routing policies. Instead, they are formatted as follows:

IP fix before: characteristics where one or more attributes are additionally broadcast together with each announced IP prefix (in the "A/m" format). BGP has many defined properties, some of which we'll go over in greater depth below.

We have previously discussed the LOCAL PREF BGP characteristic. This characteristic is utilised locally when choosing a route to a destination but is not distributed with route

announcements. The receiving BGP router reviews its configuration when a route is announced from a nearby AS and may establish a LOCAL PREF for this route.

eBGP and iBGP: Two categories of BGP sessions exist: Whereas iBGP sessions are between BGP routers in the same AS, eBGP sessions are between BGP-speaking routers in separate ASes. They utilise the same protocol but have distinct functions.

As BGP was created to allow multiple ASes on the Internet to communicate network routing data, eBGP is the "standard" way in which BGP is used. Figure 3-4 depicts an example of an eBGP session, during which BGP routers communicate a portion of their routes with routers in other ASes while also implementing route filtering rules. Typically, these sessions use a one-hop IP route (i.e., over directly connected IP links).

Each AS will often have more than one router that engages in eBGP sessions with nearby ASes. Each router will learn about a portion of all the prefixes that the whole AS is aware of throughout this procedure. All of the other routers in the AS must get routes to the external prefix from each of these eBGP routers. Care must be taken in this dissemination in order to achieve two key objectives:

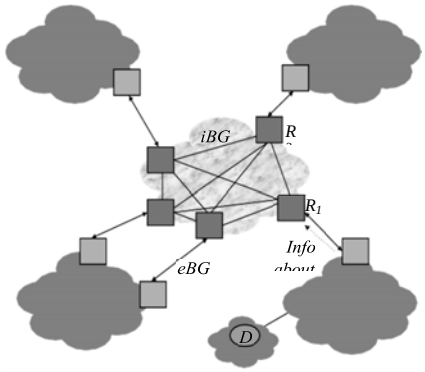


Figure 8.1: Shows the eBGP and iBGP.

Forwarding without loops. The resultant routes (and the following forwarding pathways of packets delivered along those routes) selected by all routers after the propagation of eBGP learnt routes should be devoid of deflections and forwarding loop.

Total discernibility. Allowing each AS to be viewed as a single monolithic unit is one of the objectives of BGP. For each eBGP-speaking route in the AS to have a comprehensive picture of all external routes, external route information must be exchanged. Take prefix D and Figure 8.1 as examples. In order to forward packets to D, router R2 has to hear a direct announcement from D, but R2 hasn't received one on any of its eBGP sessions. 2 By "full visibility," we mean that every router chooses the same route as it would have chosen if it had been shown the top routes from every eBGP router in the AS for every external destination.

Via internal BGP (iBGP) sessions that are active in each AS, routes that have been learnt externally are distributed to routers inside that AS. The topology across which iBGP sessions should be performed is a crucial topic. One option is to "flood" updates of external routes to all BGP routers in an AS using an arbitrarily linked graph. Of course, a flooding-based strategy would need additional methods to prevent routing loops. This issue was easily resolved by the original BGP definition by simply establishing a complete mesh of iBGP sessions (see Figure 8.2), in which each eBGP router maintains an iBGP connection with each and every other BGP router in the AS. It is now simple to flood updates; an eBGP router just has to send UPDATE

messages to its iBGP neighbours. Since it does not have any eBGP sessions with a router in another AS, an iBGP router is not required to send any UPDATE messages.

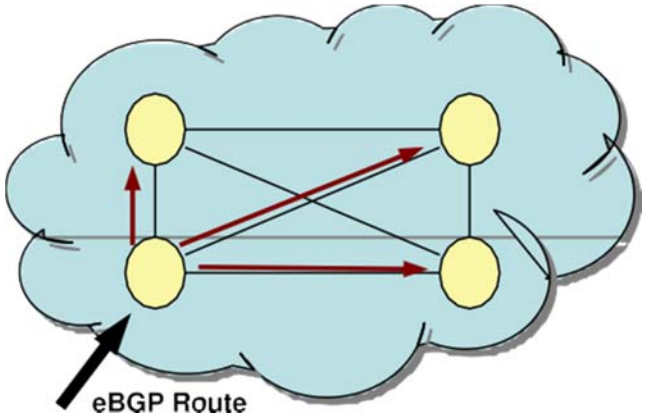


Figure 8.2: Tiny ASes create a "complete mesh" of iBGP sessions. A router is represented by each circle in an AS. Across iBGP sessions, only eBGP-learned routes are promoted again.

Note that iBGP is not an IGP like RIP or OSPF and cannot be used to configure routing state that enables packets to be successfully routed between internal nodes in an AS. Instead, iBGP sessions, which operate via TCP, provide routers within an AS a means to utilise BGP to communicate about external routes. In actuality, whatever IGP is in use in the AS is utilised to route iBGP sessions and messages between the BGP routers in the AS.

One may ask why iBGP is required and why one can't just transmit BGP updates using the IGP that is already in use in the AS. While it is feasible to utilise that way, there are various reasons why introducing eBGP routes into an IGP is cumbersome. The first problem is that most IGPs depend on periodic routing announcements rather than incremental updates and don't scale as well as BGP does in terms of the number of routes being broadcast (i.e., their state machines are different). Second, the extensive collection of properties contained in BGP are often not implemented by IGPs. It is preferable to perform BGP sessions inside an AS as well, in order to retain all the route-related information gained during eBGP sessions.

Scalability is constrained by the need for a full-mesh setup, which calls for e(e 1)/2 + ei iBGP sessions for a network with e eBGP routers and additional interior routers. Although a small AS with a few routers won't have an issue with this quadratic scalability, big backbone networks generally contain hundreds or even thousands of routers, necessitating tens of thousands of iBGP sessions. In certain circumstances, the quadratic scaling does not perform well. How to demonstrate the scalability of iBGP sessions is covered in the part that follows.

iBGP Scalability: The two most often used ways to increase iBGP scalability are presently. Both need that routers be manually configured into some kind of hierarchy. The first approach makes use of route reflectors, while the second involves establishing BGP router confederations. Here, we provide a short summary of route reflection's key concepts. For more information on BGP confederations, interested readers are referred to RFC 3065.

A BGP router called a route reflector may be set up to have client BGP routers. Every destination prefix is served by just one optimal route, which a route reflector chooses and publishes to all of its customers. The following guidelines are followed by an AS with a route reflector setup while updating its routes:

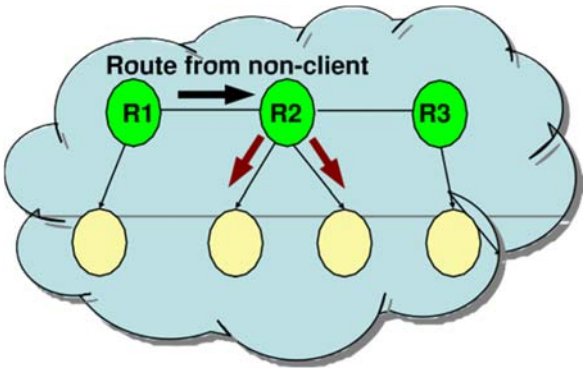


Figure 8.3: Routes learned from non-clients are re- advertised to clients only.

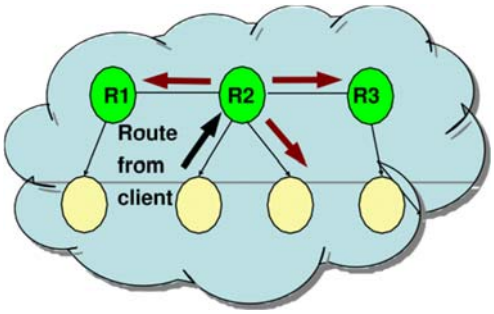


Figure 8.4: Routes learned from clients are re-advertised over all iBGP sessions.

Figure 8.4 as previously shown, larger ASes often deploy route reflectors to advertise certain iBGP-learned routes. Routers' directed edges signify iBGP sessions from route reflectors to clients (e.g., router R2 is a route reflector with two clients). All routers re-advertise eBGP-learned routes across all iBGP sessions, as shown in Figure 8.3.

A route reflector re-advertises a route it has learned through eBGP or iBGP from one of its clients to all of its clients during that session. A route reflector re-advertises a route to its client routers but not across any other iBGP sessions if it receives an iBGP route learning from a router that is not one of its clients.

A distinct scalability issue arises when an AS only has one route reflector because of the potential need to handle several client sessions. More crucially, if there are several egress connections from the AS to a destination prefix, a single route-reflector configuration may not effectively employ them all as all clients would inherit the route reflector's single decision. Many networks employ numerous route reflectors and arrange them hierarchically to address this issue. A sample route reflector hierarchy and the way routes spread from different iBGP sessions are shown in Figure 8.4.

Whether a BGP route update is propagating through an eBGP session or an iBGP session will affect how the update propagates. The IP addresses of the routers on each end of an eBGP connection are directly connected to one another and are often on the same local area network, making it a point-to-point session. While there are few exceptions (such as "multi-hop eBGP"), directly linked eBGP sessions are the norm. Since an eBGP session is point-to-point, both the opposite end of the point-to-point connection and the next-hop attribute for the BGP route are guaranteed to be accessible. Whether a route was discovered via eBGP or iBGP initially, a router will advertise it throughout an eBGP session.

The next-hop IP address for a route discovered via iBGP, however, may be more than one IP-level hop away if an iBGP session is established between two routers that are not physically linked. In reality, since the route's next-hop IP address is often one of the AS's border routers, this next hop could not even match to that router, but rather one that is a few iBGP hops distant. To establish communication between the two ends of the BGP session and to establish the route to the next-hop IP address specified in the route attribute, the routers in iBGP depend on the internal routing protocol of the AS.

It takes skill to appropriately configure an iBGP topology for loop-free forwarding and full visibility. Many sorts of erroneous behaviour, such as persistent forwarding loops and oscillations, may be produced by wrong iBGP topology setting. Since not all route reflector topologies meet visibility, route reflection has concerns with accuracy.

Important BGP Features: We can now comprehend the structure of a BGP route and how route announcements and withdrawals enable a router to create a forwarding table using all available routing information. To convey a packet intended for a prefix, this forwarding table normally contains one selected path in the form of the router's egress interface (port), which corresponds to the next nearby IP address. Remember that each router matches the longest prefix on the destination IP address of each packet. Reachability Trade: next hop Attribute: Each announced prefix contains a set of qualities that are part of a BGP route announcement. One of these is the NEXT HOP characteristic, which provides the router's IP address to which the packet should be sent. The NEXT HOP field is modified when the announcement travels over an AS boundary; normally, it is changed to the IP address of the AS's border router.

For eBGP speakers, act as described above. For speakers of iBGP, the first router that introduces the route into iBGP and assigns its so-called loopback address as the next hop (the address that all other routers within the AS can use to reach the first router). The other iBGP routers in the AS maintain this configuration and divert any packets going that way (in the opposite direction of the announcement) towards the next hop IP address using the AS's IGP. Prefix packets often move in the opposite direction from the prefix's route announcements.

AS Path Length: aspath Attribute: The AS- PATH property, a vector that identifies all the ASes that this route announcement has passed through (in reverse order), is another feature that varies when a route announcement travels several ASes. The first router prepends the distinctive identity of its own AS and broadcasts the notice when an AS border is crossed (subject to its route filtering rules). BGP is categorised as a path vector protocol because it uses a "path vector"—a list of ASes for each route.

A path vector has two functions. The first is avoiding loops. The router determines if its own AS identifier is present in the vector before crossing an AS border. If so, the route announcement is discarded since importing the route will only result in a routing loop when packets are forwarded. The route vector's second function is to assist in selecting an appropriate path from a variety of options. The aspath length is utilised to choose the route if a local PREF is absent for a given route. We favour shorter aspath lengths than longer ones. While traditional route vector protocols would choose the shortest vectors, it's vital to keep in mind that BGP isn't strictly a shortest-aspath protocol since it considers routing rules. Priority is always given to the local pref property over aspath. Yet, in practise, a lot of routes are ultimately chosen based on shortest-aspath.

Multi-Exit Discriminator (MED): The two most significant BGP properties that we have seen thus far are LOCAL PREF and ASPATH. There are additional characteristics that are used in practise, such as the multi-exit discriminator (MED). When two ASes are linked at numerous points, the MED characteristic may be used to define preferred routes. The LOCAL PREF

characteristic is useless when two ASes are connected at more than one location and one of them favours one transit point over another for certain (or all) prefixes. MEDs were developed to address this issue.

Using an example may help you understand MED the best. A provider-customer relationship with a nationwide reach for both the provider P and the consumer C. The client wants the supplier to pay for cross-country transit for the customer's packets since cross-country bandwidth is a significantly more costly resource than local bandwidth. Let's say we wish to send DSF-bound packets from the east coast (Boston) over P's network rather than C's. Since doing so would require C in Boston to spend its own resources and contradict the point of having P as its Internet provider, we wish to stop P from transiting the packet to C in Boston (Figure 8.5).

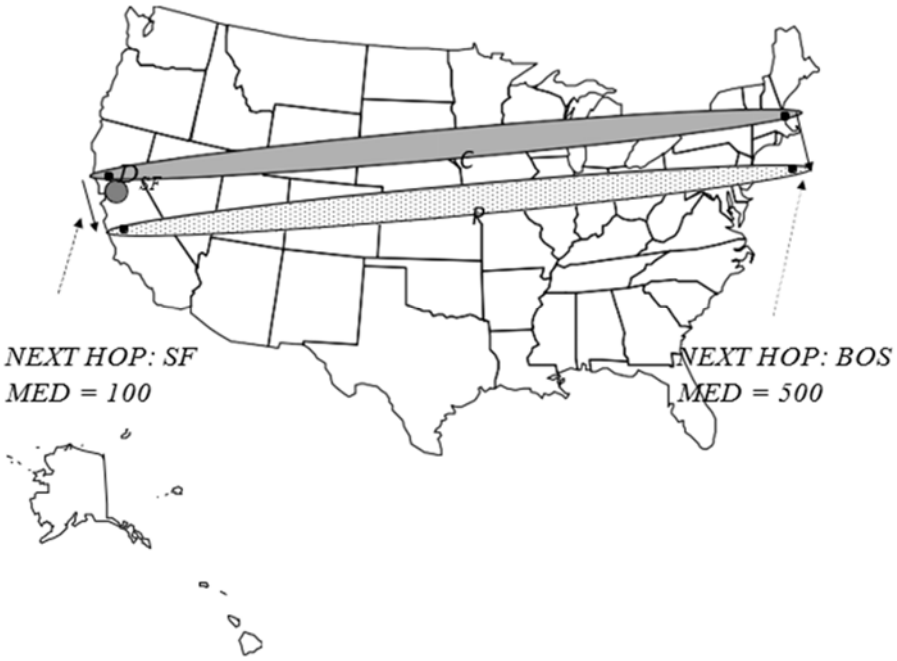


Figure 8.5: MEDs are helpful in a variety of circumstances, such as when P is C's transit provider and C's wide-area network is not the preferred route for cross-country packets to use to get to C. Yet, if C and P are peers, MED may—and often does—go unnoticed. The MED for DS F in this illustration is set to 100 at the SF exchange point and 500 in Boston, allowing P to act morally if it so chooses.

An AS, in this example C, may instruct another (P), using a MED attribute, on how to choose one NEXT HOP from a variety for a prefix DSF. The smallest MED among many options originating from the same neighbour AS will be chosen by each router. While there are no semantics attached to how MED values are chosen, it is evident that they must be chosen and reported consistently throughout the eBGP routers in an AS. In our case, the desired result is achieved with a MED of 100 for the SF next hop for prefix DSF and a MED of 500 for the BOS next hop for the same prefix.

The fact that MEDs are often disregarded in AS-AS relationships without a financial settlement is crucial to understand (or explicit arrangement, in the absence of money). In instance, the majority of peering agreements disregard MED. The wide-area Internet experiences a

significant degree of asymmetric routing as a result. As shown in Figure 3-7, for instance, if P and C were in a peering connection, cross-country packets from P to C would go via C's wide-area network and vice versa. Hot-potato routing is a kind of routing that is often used to describe how eager P and C would be to remove the packet from their respective networks. The enforcement of "cold-potato routing" would be made possible by a financial arrangement, which would provide an incentive to respect MEDs.

A great example of cold-potato routing in action is when major content hosts peer with tier-1 ISPs. An ISP could, for example, peer with a content-hosting provider in order to get direct access to the latter's clients (popular content-hosting sites), but it does not want the hosting provider to take use of its backbone. The ISP may demand that its MEDs be fulfilled in order to comply with this obligation.

Bringing It All Together: BGP Route Selection: We can now talk about the set of guidelines that BGP routers in an AS use to choose a route from a list of options.

BGP in Nature: The interdomain routing system's vulnerability sometimes shows up as disruptions in connection or other oddities. Usually, sluggish convergence, malicious intent, or misconfigurations are to blame for these issues. In order to improve load balancing and fault tolerance, BGP is also widely utilised to enable consumer networks to connect to numerous different providers. However, as we will see, BGP fails to adequately support this objective.

Accidental or Intentional Hijacking of Routes: Lack of origin authentication, or the inability to determine which AS is the owner of a particular prefix, is the cause of one set of issues. As a consequence, every AS (or BGP-speaking node) has the ability to start a route for any prefix and may as a result receive traffic from other networks sent to any destination in the prefix. Here are two intriguing instances of this conduct:

Pakistani-directed YouTube: The majority of Internet users lost access to the popular video-sharing website YouTube on February 24, 2008. A few details are helpful in understanding what occurred:

The longest prefix match (LPM) is used by Internet routers to transmit packets when the destination IP address matches more than one prefix item in the Use the routing table item that matches the destination address and prefix the longest.

YouTube access was being blocked by order of the Pakistani government to all ISPs. A user may be presented with a web page that reads, "We're sorry, but your friendly government has decided that YouTube isn't good for your mental well-being," if one chooses to redirect all traffic going to the IP address to a different location. There are generally two ways to block traffic from an IP address. The latter is probably a better customer experience since it informs consumers of what's happening, keeping them in the know and preventing them from calling customer service or flooding the website with demands. (Such distraction is extremely widespread; for instance, it is used in many public WiFi hotspots that need sign-on.)

A /24 routing table entry was added by Pakistan Telecom for the set of addresses that www.youtube.com resolves to. Everything is OK thus far. Regrettably, rather than out of malice, Pakistan Telecom's routers disclosed this /24 routing advertising to one of its ISPs due to a misconfiguration (likely brought on by a stressed or negligent engineer) (PCCW in Hong Kong). Usually, if PCCW had known (as it should have known) the legal set of IP prefixes that Pakistan Telecom possessed, this leak would not have been a problem. PCCW didn't disregard this route, however, and likely gave it priority over all the other routes it already knew to the appropriate addresses—unfortunately, maybe due to another mistake or oversight (recall that

the typical rule is for customer routes to be prioritised over peer and provider routes). At this point, all communication intended for YouTube would have been diverted to Pakistan Telecom and "blackholed" from machines within PCCW and its customers' networks.

Since almost the whole Internet was unable to access YouTube, the issue was substantially worse. This is as a result of the LPM approach that was used to determine the optimum path to a destination. In most cases, YouTube's ISPs do not market /24 on its behalf; instead, such routes are included in advertising that span a (far) larger range of IP addresses. As a result, when PCCW's neighbours saw PCCW advertise a more precise route, they followed the rules and imported those routes into their routing tables, readvertising them to their own neighbours, and so on, until the entire Internet (aside, presumably, from a few places like YouTube's internal network itself) had this poisoned routing table entry for the IP addresses in question.

This talk emphasises a crucial point regarding huge systems: their failures have complex causes. In this instance, all of the following things happened:

1. Concerned that access to a certain site may spark unrest, the Pakistani government decided to restrict it.
2. Pakistan Telecom made the mistaken decision to use a /24 to reroute traffic.
3. PCCW failed to disregard the leaked advertising despite the fact that it should have.
4. Had the initial proper commercials also used the /24 prefix, the issue could not have been as prevalent. The original correct adverts used less-specific prefixes.

YouTube is a highly famous website that receives a lot of requests, therefore Pakistan Telecom unintentionally orchestrated a major traffic assault on itself (and on PCCW). The volume of traffic probably made diagnosis difficult since packets from tools like traceroute may not have advanced beyond congested areas, which may have been upstream of Pakistan Telecom.

It seems that the first clue as to what could have really occurred came from a review of the routing announcement records that were made accessible to different ISPs as well as the general public. They demonstrated that a route to a prefix that had previously always been originated by another AS had began to be originated by a new AS.

This finding shows that it would be helpful to combine public "warning systems" that maintain such information and highlight irregularities (though there are many legitimate reasons why routes often change origin ASes too). Moreover, it mandates the upkeep of an accurate registry that contains prefix to owner AS mappings; investigations have shown that the present registries, sadly, include a lot of mistakes and omissions.

Such issues (black holes and hijacks) sometimes occur and are by no means a unique incidence. Every year, there are generally a few significant occurrences of this kind, albeit it is normally easier to generate news when a major website is selectively taken down. On a weekly basis, there are also a number of smaller-scale occurrences and abnormalities.

Spam with spoofed prefixes Sending difficult-to-trace email spam is an intriguing "application" of routing hijacks using the same ideas as those explained above. On the internet, it is simple for a source to pretend to transmit packets from an IP address that it hasn't really been given and easily spoof a source IP address. Email, however, is a bi-directional protocol that operates on top of TCP and employs a feedback channel for acknowledgments. Since the spoofer also has to be able to receive packets at the spoof IP address, untraceable source spoofing is a little more difficult.

A clever technique to solve this issue is for the bad guy to persuade one or more upstream ISPs to pay attention to BGP routing announcements. These ISPs may be unethical themselves or

may simply choose to overlook suspicious activity because they are paid to do so. By issuing notifications about that prefix, the bad guy temporarily hijacks a section of the IP address space, usually an unassigned one (but it isn't need to be). Routers on the Internet are able to access the relevant addresses after that notification has spread across the network. The bad guy opens a tonne of email connections, sends plenty of spam, and then, after 45 or 60 minutes, just closes the advertised route and vanishes. Afterwards, when one attempts to follow a trail back to the spam's malicious originating IP addresses, nothing is left! According to a recent research, 10% of spam sent to "spam traps" (domains set up to accept spam and collect data) originated from IP addresses that were associated with such route hijacks.

Naturally, one might determine where messages transmitted from hijacked routes originated if all BGP announcements were meticulously documented and kept, but often, these logs aren't adequately maintained at a fine enough temporal granularity. When such records are eventually kept at various BGP vantage points around the Internet, individuals looking to deliver untraceable trash may need to devise a different strategy.

Convergence issues: Using BGP, it might take several minutes for routes to stabilise to a consistent state once a problem has been detected. A router notifies its neighbours to withdraw when a problem is found. Each router implements a route flap damping scheme by ignoring frequently changing advertisements from a router for a prefix in order to prevent routing advertisements from spreading throughout the entire network and causing routing table calculations for failures or routing changes that may only be temporary. Many people think that damping enhances scalability, however it also prolongs convergence.

Multi-homing: By enabling numerous connections (and eBGP sessions) between two ASes, as well as an AS connecting to several providers, BGP enables an AS to be multi-homed. While common routing rules limit the topologies one may see in reality, BGP itself does not place any restrictions on the AS topology. Multihoming is used to distribute load and tolerate errors (Figure 8.6).

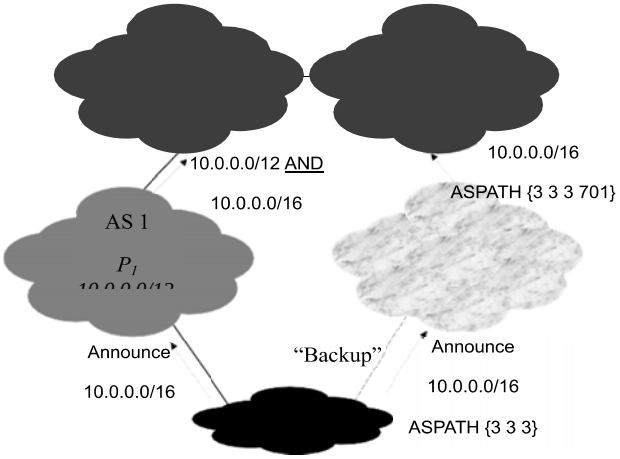


Figure 8.6: which displays the topology and address blocks of the relevant parties, provides an illustration. Since transit providers may combine address blocks from several customers into one or a few route announcements to their respective providers, this example employs provider-based addressing for the customer, which enables the routing state in the Internet backbones to scale better.

Scalable and effective multi-homing using BGP is still an unsolved problem. As there are more multi-homed customer networks, the interdomain routing system will be under more strain in terms of routing churn, convergence time, and routing table state. Moreover, the complicated

interactions between LPM, failover and load balancing objectives, as well as tricks like the AS path padding method, can have unforeseen implications.

The "competitive collaboration" environment in which the Internet routing system functions requires many independently functioning networks to collaborate in order to offer connection while competing with one another. It must also scale effectively to manage a vast and growing number of component networks and support a variety of routing strategies, some of which we covered in depth (transit and peering).

The interdomain routing protocol BGP is technically rather basic, yet it operates in a very complicated way in real life. Its complexity is caused by configuration flexibility, which enables the interchange of a wide range of characteristics in route announcements. In the field of wide-area routing, there are a variety of unsolved and intriguing research issues related to failover, scalability, configuration, accuracy, load balancing (traffic engineering), security, and policy design. Interdomain routing is still difficult to comprehend, describe, and create despite a lot of work and tremendous advancement over the previous several years.

CHAPTER 9

FIREWALLS AND IDS

Ganesh D, Professor
Department of Computer Science and Information Technology, Jain (Deemed to be University) Bangalore, Karnataka, India
Email Id- d.ganesh@jainuniversity.ac.in

A firewall is a security system that controls and monitors incoming and outgoing network traffic based on a set of predefined security rules. Its main purpose is to prevent unauthorized access to or from a private network. Firewalls are typically implemented as software or hardware and can be used to protect a single device, a group of devices, or an entire network. Firewalls can be classified into different types based on their functionality and deployment methods. Some common types include:

Network firewall: This type of firewall is typically a hardware device that sits at the edge of a network and controls traffic between the internal network and the internet.

Host-based firewall: This type of firewall is installed on individual host computers and monitors traffic to and from those specific machines.

Stateful firewall: This type of firewall tracks the state of network connections and only allows traffic that is part of an established connection.

Application-level firewall: This type of firewall inspects the contents of network traffic at the application layer and only allows traffic that conforms to a set of rules and policies.

In addition to these types, firewalls can also be classified based on their deployment methods such as software firewall, hardware firewall, or virtual firewall. They can also be used in conjunction with other security technologies like intrusion detection and prevention systems (IDPS) and virtual private networks (VPNs) to provide an additional layer of security .

A firewall examines each incoming and outgoing packet and compares it to a set of rules to determine whether it should be allowed or blocked. Firewalls can be configured to allow or block traffic based on several different criteria, including the source and destination IP addresses, the protocol being used, and the port number.

Firewalls are an essential part of a comprehensive security strategy, along with antivirus software, intrusion detection systems, and other security measures. They can be an effective way to prevent unauthorized access to a private network and block malicious traffic, but they are not a complete solution. It's important to keep them up to date and configure them properly to ensure maximum protection .

A firewall is a network security system that monitors and controls the incoming and outgoing network traffic based on predetermined security rules and policies. Firewalls can be implemented as hardware, software, or a combination of both. They are used to prevent unauthorized access to or from a private network and can be used to provide an additional layer of security for devices connected to the internet.

Classification of Firewall:

To classify firewalls based on their architecture:

Packet-filtering firewall: This type of firewall examines each packet that passes through the network and filters them based on a set of predefined rules. This is the simplest and most common type of firewall and is often used in small networks.

Circuit-level gateway: This type of firewall examines the connection information of each packet, such as the source and destination IP addresses and ports, but does not inspect the contents of the packets themselves. This type of firewall is more effective at blocking unauthorized access and is often used in larger networks.

Application-level gateway: This type of firewall examines the contents of each packet and can filter traffic based on specific application protocols, such as HTTP or FTP. This type of firewall is more advanced and can provide a higher level of security than the other types.

Another way to classify firewalls is based on their rules, for example:

Whitelist-based firewall: only allows traffic that is explicitly allowed by the administrator, all other traffic is blocked.

Blacklist-based firewall: blocks traffic that is explicitly disallowed by the administrator, all other traffic is allowed.

Firewalls are an important component of a comprehensive network security strategy and can help protect against a wide range of cyber threats, including viruses, worms, and other malicious software . They are also essential tools for compliance with various industry and government regulations.

Another way to classify firewalls is based on their location in the network:

Perimeter firewall: This type of firewall is placed at the edge of a network and is the first line of defense against incoming traffic. It is typically a combination of a network firewall and a DMZ (Demilitarized Zone).

Internal firewall: This type of firewall is placed within a network to protect internal resources and segment internal network traffic. It can be used to protect sensitive data and limit the spread of malware.

Virtual firewall: This type of firewall is implemented as a software component running on a virtual machine, allowing the deployment of firewall capabilities in cloud-based environments.

Firewalls are not only used to block incoming malicious traffic but also to control and monitor outgoing traffic from the network . This is known as "egress filtering" and can help prevent sensitive data from leaving the network, as well as prevent malware from communicating with command-and-control servers. Another important feature of firewalls is their ability to support multiple network protocols and ports. Most modern firewalls support a wide range of protocols such as TCP/IP, DNS, DHCP, and others. They also support different ports and services such as HTTP, FTP, SSH, and many others.

Firewalls are an important network security tool that can provide multiple layers of protection for your network and devices. They can be implemented in different ways, based on different criteria such as functionality, deployment methods, architecture, location, and more. It's important to understand the different types of firewalls and how they can be used to protect your network, so you can make an informed decision when selecting and configuring a firewall solution.

A firewall is a device that controls traffic between an internal, or "protected," network and an external, or "less trustworthy," network. A firewall is essentially a piece of executable code that

runs on a certain machine. Non-firewall tasks are not performed on the machine that is operating the firewall since all traffic should flow via it; moreover, because non-firewall code does not reside on the computer, it is difficult for an attacker to utilise any weakness to compromise the firewall.

Design concept: Firewalls execute a security strategy that focuses on what undesirable events should not take place in a "protected environment."Security regulations that specify what is permitted: A "default-deny" ruleset for firewalls is required by best security practises, which implies that the only network connections permitted are those that have been specifically indicated to be authorised. Security regulations that specify what is not permitted: Users and the business community prefer a "default-allow" ruleset, in which all traffic is permitted unless it has been expressly restricted, since they lack the precise knowledge necessary to specify what should be let in. Due to ignorance and newly developed apps, this configuration is more often utilised even though it is substantially more prone to accidental network connections and system compromise.

Firewalls: All firewalls do not necessarily require the same functionality. Based only on the security rules that each firewall is set with, one cannot compare the "quality" of two firewalls. Threats that an installation (network) needs to prevent from occurring are the main determinant in the choice of a security strategy for a firewall. A packet filtering firewall restricts access to packets based on the packet address (source or destination) or a specified transport protocol type (such as HTTP, Telnet, etc)

Egress filtering: Only packets belonging to certain networks or transport layer protocols would be sent out (or not sent out). Ingress filtering: Packets from (or not from) just particular source networks and/or transport layer protocols may be permitted entry. Having the packet filter set up to not allow in packets with a source address that matches to the internal network is a standard method of preventing IP spoofing attacks. In other words, the attacker has changed the source IP address to seem to be a device on the network within the firewall's protection. Given that we wish to restrict traffic from certain networks, IP addresses, and transport layer protocols, the packet filter code will get more complex.

Using Packet Filter Firewalls to Prevent Attacks:

Packet filter firewalls may be set up to prevent source routing and small fragmentation attacks in addition to IP spoofing attacks.

Source routing attacks: whenever source specifies the path a packet should travel to get around security measures, all source routed packets should be discarded.

Small fragment attacks: An attacker may impose a minimum fragment size requirement to contain the whole TCP header information by using the IP fragmentation option to produce very tiny fragments and push the information into fewer individual pieces (Figure 9.1 and Figure 9.2).

Firewalls with Stateful Inspection: Packet filter-based firewalls handle each packet individually and don't keep track of the status of the action that was done on a previous packet they've processed. Stateful firewalls, also known as circuit firewalls, assess each packet's contents in relation to where it falls in the sequence of packets for a particular connection. Stateful firewalls can tell whether a packet is the beginning of a new connection or a piece of an ongoing connection by keeping track of all connections that pass through the firewall. When an intruder delivers several TCP segments with various sequence numbers, stateful firewalls may remember the sequence numbers that are anticipated on both sides as part of a TCP session

and can prevent attempts to hijack the connection (trial-and-error). Certain firewall rules will be triggered based on the condition of a connection. Data packets for a connection cannot enter before the connection is fully created or leave after a connection is fully torn down, for instance. Limit the amount of TCP connections that may be established concurrently per IP address. Limit the quantity of data that may be transported from the internal network to any external IP address each day.

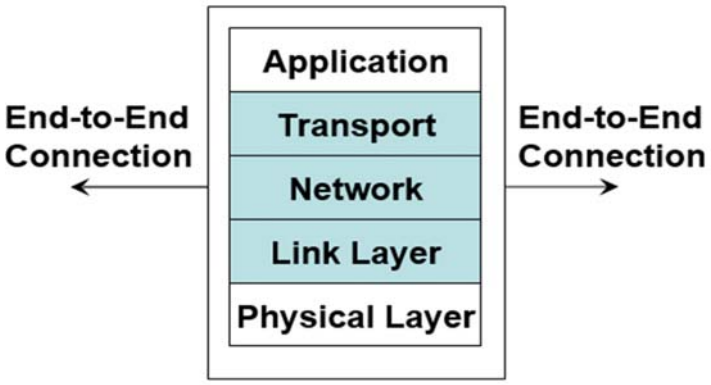


Figure 9.1: Layers supported by Packet Filter and Stateful Firewalls

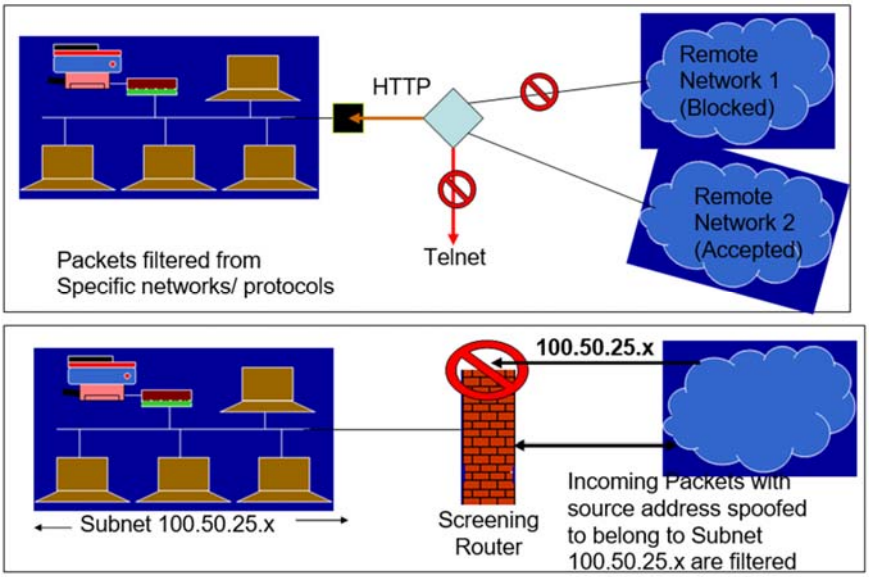


Figure 9.2: Representing the Packet Filters.

Firewall for application proxies: Packet filters only examine the headers of packets; they do not examine the contents contained inside them. An application layer firewall (also known as a proxy or bastion) replicates a program's intended effects so that it only accepts requests that will cause it to behave appropriately. A proxy gateway has two faces: from the inside, it seems to be the destination connection, and from the outside, it answers as if it were from the inside.

Firewall for application proxies: A proxy server and a proxy client are required for each application proxy in the firewall. Instead of enabling users to directly contact with servers on the Internet, all internal user communication with the Internet is routed via a proxy server. A request to connect to an external service is sent by an internal user (client). The request passes

via the Application Proxy Firewall, which operates a proxy server for the specific service being sought for. The proxy server assesses the request and chooses whether to approve or reject it in accordance with a set of rules that are controlled for the specific network service.

Proxy servers only let packets that adhere to the application protocol's services. Proxy servers are helpful for gathering audit logs of session data. The proxy server sends the request to the proxy client if it authorises it. After making communication with the actual server on the real client's behalf, the proxy client starts relaying requests from the proxy server to the real server and receiving replies back from the real server. Between the proxy client and the actual client, the proxy server distributes requests and answers.

Note: The description above makes the assumption that the server is in the external network and the client is on the internal network. The opposite case also bears the same discussion: The proxy server receives a request from the actual client (from the outside network), assesses it, and then transmits it to the proxy client, who then contacts the real server (running in the internal network). The proxy client transmits the actual server's answer to the proxy server, which then transmits it to the real client (in the outside network).

Using Proxy Firewall Examples

Case 1: A business wishes to provide dial-in access for its staff members without subjecting its resources to login attempts from distant non-staff members. Imagine that several operating system types are present on the internal network, none of which are capable of providing robust authentication using a challenge-response scheme.

Solution: The need might be met by a proxy that is particularly designed to demand robust authentication, such as a challenge-response, in addition to a legitimate login and associated password. The internal host's operating system requires that just the username and password be sent, and the proxy checks the challenge-response itself.

Case 2: A business wants to create an online pricing list so that customers may see the available goods and prices. It wants to make sure that (a) no external party can alter the pricing or product list and (b) external parties can only view the price list and not any of the more sensitive information kept within.

Solution: To fulfil the need, a specially designed proxy that keeps track of file transfer protocol data and verifies that just the price list file was accessed and that it could only be read, not changed, may be used.

A proxy firewall might act more as a guard, keeping an eye on the quantity and calibre of data transferred. It could monitor the volume of data sent from the internal network per user and block access if it went beyond a certain threshold. In order to screen all incoming files for viruses and, if necessary, exiting files as well, a proxy firewall may additionally run a virus scanner.

Proxy Servers: A proxy server is a server that serves as a middleman for requests from clients looking for resources and/or services from other servers. A proxy server may be a computer system or an application software. Typically, the proxy server assesses the request in accordance with its filtering criteria (such as by IP address or port number) and takes action as a result. A proxy server has a wide range of possible uses, including: - To maintain the anonymity of computers behind it (primarily for security) - To expedite resource access (using caching) Internet proxy servers

To implement network services or content access policies (to block visiting undesired sites) - content-filtering web proxy - proxy firewall - to log/ audit use - proxy firewall - to check transmitted material for malware before delivery. An intercepting proxy that actively intercepts all requests without needing client-side setup may be deployed inside a network environment in place of requiring all client computers to configure their browsers to utilise the proxy.

A gateway or, less often, a tunneling proxy is a kind of proxy server that transmits requests and responses unchanged. Reverse Proxy Servers a reverse proxy server is an Internet-facing proxy that is used as a front-end to manage and secure access to a server or servers on a private network as well as to carry out operations like load-balancing, authentication, and other similar processes.

Internet clients see a reverse proxy server as a regular server. Internally, it could do nothing more than pass on the client requests to the original internal servers. The answer would seem to have originated from the proxy server when it was returned.

The use of reverse proxy servers has various benefits, including:

SSL acceleration and encryption: By acting as a single "SSL proxy" to offer SSL encryption for any number of hosts, a reverse proxy server may speed up communication sessions by eliminating the requirement for a unique SSL Server Certificate for every host.

Load balancing: The reverse proxy server has the ability to flexibly split the workload across numerous web servers.

Security: The reverse proxy server may act as an extra line of security, guarding against threats that are particular to the operating system and web server. Home users, lone employees, and small enterprises utilise cable modems or DSL connections with unrestricted, always-on access as their driving force.

Personal Firewalls: These folks need a firewall, but it may appear too complicated and costly to use a separate firewall machine to safeguard a single workstation. A workstation may be susceptible to malicious active agents (ActiveX controls or Java applets), malicious malware, the leaking of personally identifiable information stored on the workstation, and vulnerability scans (like nmap) to find possible flaws.

A personal firewall is an application software that runs on a workstation to monitor traffic inside the workstation and prevent unauthorised traffic from leaving or entering the network to which the workstation is connected. A user might set up their own firewall to create records of prior activity, allow traffic only from particular websites, and reject access from other websites. A virus scanner that is integrated into a personal firewall might be set up to automatically scan any incoming data to the workstation. As compared to computers that are not protected by a personal firewall, static machines are a susceptible target for attack by the hacker community.

Personal Firewall Examples

1. Software-based firewalls based on UNIX.
2. TCP Wrappers: controlled by two text files named hosts.allow and hosts.deny, this feature restricts incoming network connections depending on port number, domain, or IP address.
3. An inbound connection request, for instance, is permitted if it originates from a trusted IP address and is headed for a port that is open for connections.
4. IPchains is a rule-based software firewall that can handle network traffic using three adjustable "chains" (sets of rules): input chain (for incoming traffic to the local system);

output chain (for data leaving the local system); and forward chain (for traffic received by the local system; but, not destined for the local system).

During processing, each packet travels through all three chains.

IP tables: Each packet is only handled at the proper chain when using IPtables, which employs the same three chains as IPchains for managing traffic and policy rules. This improves speed and enables more precise management over network traffic.

Firewalls' Capabilities and Limitations: Firewalls cannot safeguard an environment on their own. A firewall only defends the environment's perimeter against intrusions by outsiders attempting to run code or access data on the computers in the protected environment. Firewalls are unable to thwart internal attacks (through disgruntled employees). Malware that is imported through a laptop, PDA, or other portable storage device that was infected outside of the network before being joined and utilised within is not protected by firewalls. If firewalls are the sole way to govern the whole network perimeter, they may be held accountable for any security lapse. If a host on the inside network connects to the outside network via a modem, the host and the modem expose the whole inside network to the outside network. No assault can be attributed to a firewall.

Firewalls are unable to safeguard data that has already left them. A firewall often serves as a network's single point of failure. A more sophisticated strategy that includes a screening router, a proxy firewall, and a personal firewall may be more beneficial. Firewalls need to be regularly set and updated to reflect changes in the internal and external environment as well as based on an analysis of the firewall activity reports that may indicate intrusion attempts. In order to lessen the likelihood of an attack, the computer hosting the firewall code won't contain any other programmes like editors or compilers.

Networks in the Demilitarized Zone (DMZ): A DMZ network, also known as a perimeter network, is a subnet that houses services provided by an enterprise that are accessible from a broader untrusted network (like the Internet). In other words, hosts that provide services to users outside of internal LANs, such as email, web, and DNS servers, make up the DMZ (Figure 9.3).

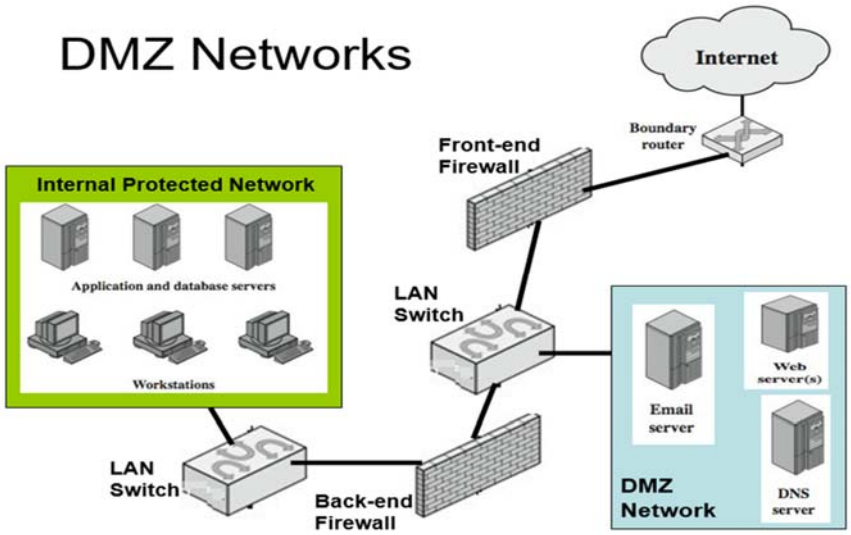


Figure 9.3: Illustration of DMZ Network.

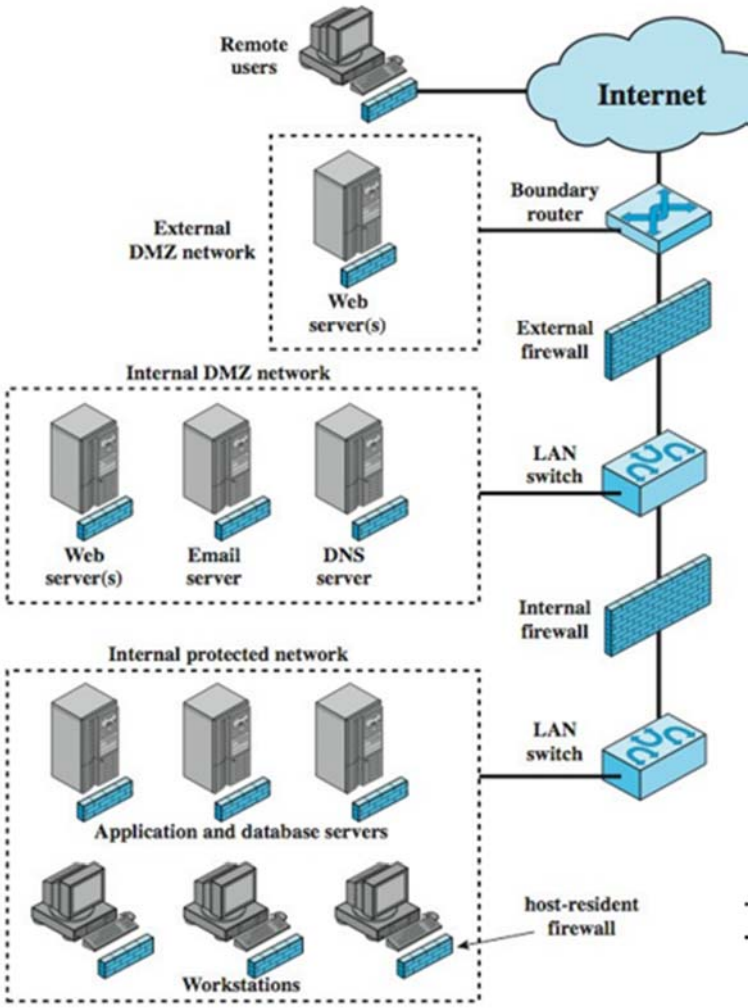


Figure 9.4: Distributed Firewalls

Due to the greater likelihood that these hosts would get hacked, they are put into their own sub-network to safeguard the remainder of the network in the event that an attacker were to be successful in assaulting them. Since an external attacker may only access the hosts in the DMZ and not any other internal networks, a DMZ network offers an extra layer of protection to a company's LAN. A front-end ("front-end") firewall monitors traffic between the DMZ network and the external Internet, while a back-end ("back-end") firewall monitors traffic between the DMZ hosts and the internal network clients. Hosts in the DMZ provide services to both the internal and external networks (Figure 9.4).

Intrusion Detection Systems (IDS): In the networking world, an IDS is comparable to a burglar alarm in the real world. An IDS's primary functions include identifying potentially harmful activity, noting conduct that deviates from norms, cataloguing and categorising the activity, and, if practical, responding to the action (Figure).

Host-based IDS (HIDS): It just monitors activity on a single system and isn't concerned with network or other systems.

Network-based IDS (NIDS): Rather than focusing on specific systems, it analyses activities (traffic) across the network it is watching (Figure 9.5).

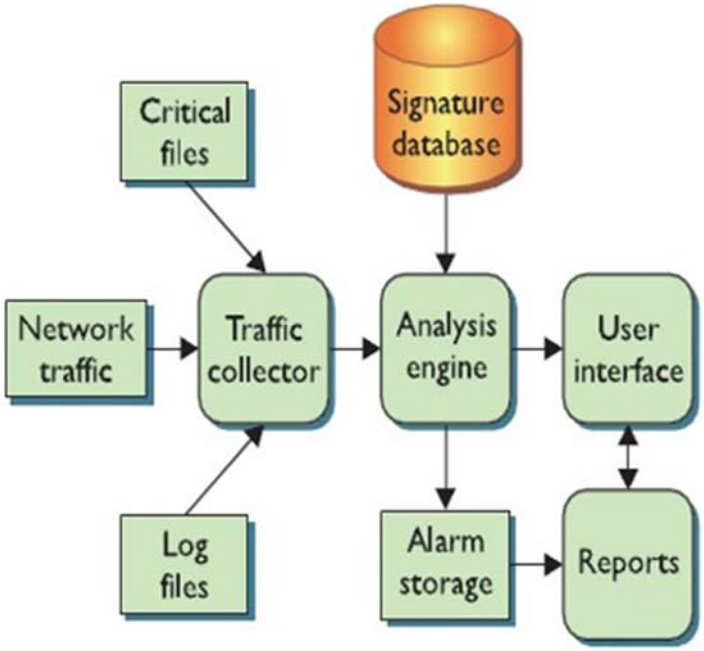


Figure 9.5: Logical Depiction of IDS Components

IDS's logical components

An intrusion detection system (IDS) is a tool that monitors activity to spot malicious or suspicious occurrences. It is often a separate computer.

An IDS often comprises of a number of specialised components that operate together on the device on which it is placed. These components are frequently logical and software-based rather than physical.

Traffic Collector - Gathers activities and events for IDS analysis. This might be log files, audit logs, or traffic entering or leaving a particular system for a HIDS. This may be network traffic that has been sniffed and stored by an NIDS.

Analytical Engine: Analyzes the network traffic that has been gathered and contrasts it with known patterns of dubious or malicious behaviour that are kept in the signature database. It is often referred to as the IDS's "brain".

A database of patterns and descriptions of recognised suspicious or harmful behaviour.

User Interface and Reporting: This section interacts with the human aspect by sending alarms when necessary and offering the user a way to utilise the IDS.

IDS based on signatures and anomalies: IDS may be divided into Signature-based and Anomaly-based IDS depending on the method used to identify suspicious or malicious traffic.

Signature-based IDS: This kind of IDS largely relies on pre-defined signatures, which are patterns of attack and traffic.

A signature-based IDS (like anti-virus software) can only compare against known patterns; if a new attack is launched that the signature-based IDS has never seen before, it will not be able to recognise it as suspicious or harmful. This is the main drawback of signature-based IDS.

Anomaly-based IDS: This kind of IDS keeps track of actions and makes an effort to categorise them as either "normal" or "anomalous" (suspicious and unidentified) based on self-developed rule sets.

An anomaly-based IDS creates and improves internal rule sets while using heuristic approaches to categorise and classify traffic.

One benefit of anomaly-based IDS is that it may be able to spot new attacks or modified versions of established ones.

One disadvantage of anomaly-based IDS is that while the system is figuring out what "normal" is, it may produce a large number of false positives. Thus, an IDS should be developed to respond to changes on the fly.

Identity-based IDS

Examples of attacks that a signature-based IDS may identify include:

1. Using a port search to find the TCP SYN flood assault.
2. The initial SYN (for example, to port 80) and the second SYN (from the same source address) to port 25 are probably not odd to an IDS, but if additional ports (particularly closed ports) start to receive SYN packets, the pattern may indicate a previous port scan.
3. One issue with signature-based IDS is the signature itself: An attacker will attempt to change a fundamental attack in a manner that does not fit the assault's recognised pattern.
4. An attacker may, for instance, change characters to their ASCII counterparts or convert lowercase to uppercase.
5. To detect attacks with various patterns, an IDS must learn new signature patterns.
6. Nowadays, statistical analysis is utilised to identify assaults whose patterns match the recorded signatures within a certain margin of error.

Heuristic-based/Anomaly-based IDS:

Heuristic-based IDS is constrained by the quantity of data the system has seen (to categorise activities into the appropriate category) and how well the current actions fit into one of the categories, similar to signature-based IDS.

There are three types of activities that may be categorised: Good/Benign, Suspicious, and Unknown. Depending on whether the IDS eventually learned that a certain activity is acceptable or not, some types of behaviour may eventually transfer from one category to another.

Model-based IDS: Create standardised models for certain tasks. Set up an alert if the actions were against the model.

As an example, an employee's typical morning activities at work might include reading emails, creating several documents using a word processor, periodically backing up data, etc. An issue could arise if an employee uses system-sensitive management tools right after after logging in.

State-based IDS: Keep track of the system's various state transitions. The system may sound an alert if the pace of state change exceeds a certain threshold, if it enters a previously unknown state, or if it enters a hazardous mode.

Misuse-based IDS: Look for actions that might be quickly misapplied

Apart for a few tools like login, password update, and create user, every attempt to access a password file is suspicious.

Positive and negative false alarms: A false positive is when an IDS triggers an alert for harmless traffic that is not a danger because it matches an activity to a specified pattern.

Theoretically, the IDS is operating properly by matching the pattern; it is unable to ascertain the motivation behind the activity, but from a human perspective, the analyst did not need to view this information since it does not pose a danger and does not call for action.

False negatives are hostile behaviour that does not match an IDS signature and goes unnoticed.

Keep in mind that an IDS can only match behaviour for which it has patterns recorded due to the limitations of its signature set.

Network-based IDS (NIDS): NIDS are installed next to the firewall on the network perimeter and monitor traffic passing by while analysing it for protocols, source, destination, content, traffic that has previously been noticed, etc. A NIDS generally monitors traffic for patterns that indicate malicious behaviour or abuse, such as denial-of-service attacks, port scans or sweeps, malicious information in the data payload of a packet or packets, vulnerability scanning, Trojan horses, viruses, worms, tunnelling, and brute-force assaults.

A NIDS's traffic controller: logically connects to a Network Interface Card (NIC), which is running in promiscuous mode (stealth mode) and sniffing the traffic (Figure 9.6).

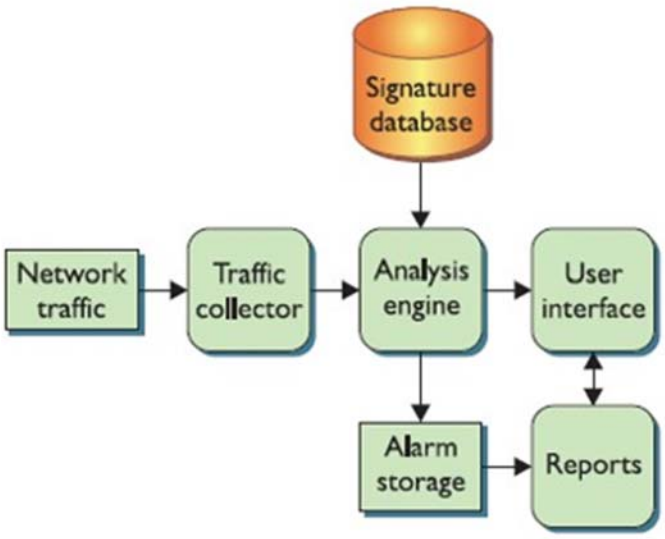


Figure 9.6: Network-based IDS (NIDS)

Benefits and Drawbacks of NIDS

Benefits of a NIDS: Reduced Overhead: A few strategically located NIDSs may be used to monitor all network traffic entering and leaving an organisation. Moreover, it is often less

expensive to upgrade and maintain a small number of NIDSs than it is to upgrade and maintain hundreds of host-based IDSs.

Big Picture: A small group of NIDSs can monitor all network traffic and correlate assaults across many systems, regardless of their size or concentration, unorganizedness or organisation.

A NIDS is useless when communication is encrypted, among other NIDS drawbacks: Non-crossing traffic is invisible to NIDS - A NIDS may overlook traffic moving through the internal network if it is only installed at the perimeter. With the availability of networks with more bandwidth, an NIDS must be able to handle significant amounts of traffic (even 1-Gbps is typical nowadays). A NIDS does not know about activity on the hosts themselves.

NIDS: Active vs. Passive

A passive NIDS only observes the traffic, analyses it, and produces alerts. It makes no adjustments to the system's defensive stance to respond to the traffic or engage in any interaction with the traffic itself. An essential addition to the passive NIDS is the active NIDS's ability to respond to the traffic it is monitoring. Active NIDS: An active NIDS has all the same parts and capabilities as the passive NIDS. An active NIDS's responses may be as basic as sending a TCP reset message to halt a prospective attack and cancel a connection, or as complicated as dynamically changing firewall rules to block all traffic from certain source IP addresses for the next few hours or days. Also known as intrusion prevention systems, active NIDS (IPSs). IPSs would be able to decode the SSH connection setup messages between a client and server and extract the session keys that would be utilised during the whole session when configured with the private keys of the servers in the internal network. This offers the IDS/IPS an extra edge when handling encrypted traffic.

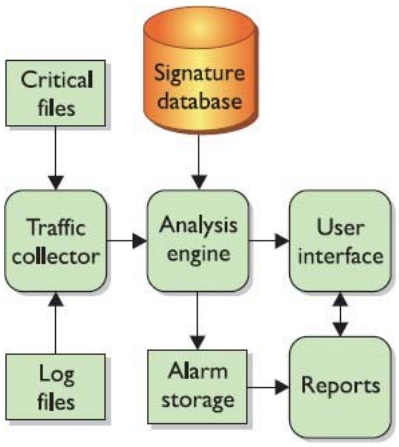


Figure 9.7: Logical Layout of a HIDS.

Host-based IDS (HIDS): A host-based IDS (HIDS) looks at network traffic entering or leaving a particular host as well as audit trails and log files, all of which are produced by the local operating system. On UNIX systems, the syslog, kernel, and error logs are studied logs; on Windows systems, the Application, System, and Security event logs are examined logs. The operation or general functioning of the system depends on the existence of critical files. These might be scripts to start or halt system processes or programme (or binary) files storing user accounts and passwords. Any unusual changes to the essential files (which, for instance, may be discovered using a checksum) could indicate that the system has been infiltrated or altered by an attacker. The HIDS can alert users of possibly harmful behaviour by keeping an eye on

certain important files. The HIDS searches the log files for specific behaviours that are indicative of malicious behaviour or misuse, such as the following: - Unusual login times, Login authentication errors, Creation of new user accounts, Modification or access to vital system files, Modification or removal of binary files (executables), Privilege escalation (Figure 9.7).

Benefits of HIDS: A HIDS might have more particular signatures that are more operating system-specific. False-positive rates may be decreased using HIDS. Administrators may construct more precise, comprehensive signatures instead of general alerts to detect fraudulent traffic much more effectively. Data may be examined by HIDS once it has been encrypted. When developed and deployed properly, a HIDS might be used to inspect encrypted communication that is opaque to NIDS. A HIDS may have extremely specialised application needs. At the host level, a HIDS may be created, altered, or customised to perform exceptionally well on a small number of apps, eliminating the need to examine or even save signatures for other programmes that are not now running on the system. A HIDS can evaluate if an alert could have an effect on that particular system. A HIDS may check things like patch levels, the existence of certain important files, and system condition while analysing traffic since it is installed on the machine. A HIDS may more precisely assess if a behaviour or pattern can be potentially detrimental to the system by being aware of all these aspects. This may drastically lower the amount of alerts that are produced.

The drawbacks of HIDS: For a system to be protected, the HIDS must be installed. The cost of ownership and maintenance for HIDS may be expensive. There will be a large number of processes to manage, programmes to update, and settings to fine-tune with a HIDS, even with a central console.

The HIDS consumes system resources on the local level. The resources (such as CPU cycles and memory) consumed by a HIDS prevent the host system from carrying out its other tasks. The HIDS has a very narrow field of vision and is unable to relate to activities outside of it; it can only detect an assault on the system it is operating on. The HIDS may be hacked or shut off if local logs were kept.

An attacker who successfully compromises the system may be able to edit or remove such alarms if the HIDS keeps its produced alarm traffic on the local system. Even if the existence of an empty log file would suggest that the system was attacked, no post-incident analysis could be carried out.

Solution: It would be a better security procedure to copy or store the log data, at least the security-related data, on a different machine.

Honeypot: A honeypot is a trap used to catch, divert, or otherwise thwart attempts at illegal access to information systems. A honeypot is often a computer, but it may also be data or an unoccupied IP address space that looks to be a part of a network but is really isolated, unprotected, and under constant observation and that appears to provide information or a resource that might be useful to attackers. As honeypots don't produce anything, they shouldn't experience any normal traffic or activity. Any information they get may be inferred to be unlawful or harmful. A honeypot network is known as a honeynet. A honeynet is used to keep an eye on a bigger, more diversified network when a single honeypot may not be enough. A honeypot or honeynet is more of a prophylactic method of spotting possible Internet attackers who may soon target the organization's network. Honeypots may be used to pose as open relays to draw spam emails and identify the spammers' source and destination email addresses.

Every Internet user may send email over an open relay, which is a mail server. Spammers keep sending the identical email to an open relay when they discover one in the hopes that it will propagate the spam. Keep in mind that a honeypot will not get any regular e-mail. It could classify everything it gets as spam. A packet sniffer, also known as a protocol analyzer, is a piece of computer hardware or software that is configured to intercept and record data packets travelling across a local area network (LAN).

Both good and bad intent may be employed with a packet sniffer:

Examine network issues and keep tabs on network use - Compile and present network statistics - Remove questionable material from network traffic - Snoop on other network users and get private data, such as password. Identify attempts at network intrusion - Reverse engineer (study using the structure of various packet headers) the protocols used across the network - Collect data for implementing a network incursion. The Network Interface Card (NIC) on the IDS housing the packet sniffer should operate in promiscuous mode and examine each packet that crosses the wire in order to collect all network traffic. The SPAN (Switched Port Analyzer) port, which is a mirrored port that can observe all traffic travelling through the switch or via certain virtual LANs, is included with the majority of switches. On a switch's SPAN port, packet sniffers may be operated.

CHAPTER 10

AN OVERVIEW ON CYBER CRIME

Ajay Shriram Kushwaha, Associate Professor,
Department of Computer Science and Information Technology, Jain (Deemed to be University) Bangalore, Karnataka, India
Email Id- ks.ajay@jainuniversity.ac.in

Cybercrime refers to criminal activities that are committed using the internet or other forms of digital communication. These activities can include a wide range of malicious actions, such as:

Hacking: unauthorized access to computer systems, networks, or devices.

Phishing: attempting to trick individuals into giving away sensitive information, such as login credentials or financial information.

Malware: the use of malware, such as viruses, Trojans, and ransomware, to damage or gain unauthorized access to computer systems.

Identity theft: stealing personal information, such as Social Security numbers or credit card numbers, and using it to commit fraud or other crimes.

Denial of Service (DOS) and Distributed Denial of Service (DDoS) attacks: overwhelming a website or network with a large amount of traffic to make it unavailable to legitimate users.

Intellectual property theft: unauthorized access to, use, or distribution of copyrighted material, trade secrets, or other proprietary information.

Online fraud: using the internet to commit financial crimes, such as Ponzi schemes, auction fraud, and investment fraud.

Cyberstalking and cyberbullying: using the internet to harass, intimidate, or otherwise harm individuals.

Cybercrime can be difficult to detect and prosecute, as criminals often use sophisticated technologies and techniques to conceal their activities. Additionally, cybercrime can cross national borders, making it difficult for law enforcement to pursue suspects . To protect against cybercrime, individuals and organizations can use security technologies such as firewalls, antivirus software, and intrusion detection systems. They can also use best practices such as keeping software up-to-date, using strong passwords and being cautious when clicking on links or providing personal information online . Cybercrime can have a wide range of consequences, both for individuals and for organizations. For individuals, it can result in financial losses, identity theft, and damage to their reputation. For organizations, it can result in financial losses, loss of sensitive information, and damage to their reputation.

In addition to the technical measures that organizations can take to protect against cybercrime, there are also legal and regulatory measures that have been put in place to combat cybercrime. For example, many countries have laws that criminalize certain types of cybercrime, such as hacking, identity theft, and online fraud. Additionally, there are international agreements, such as the Council of Europe's Cybercrime Convention, that have been put in place to help countries cooperate in the investigation and prosecution of cybercrime .

However, cybercrime is a constantly evolving field, and new types of cybercrime are emerging all the time, such as Advanced Persistent Threats (APTs) and Crypto jacking. APTs are long-term, highly targeted cyber-attacks, often by state-sponsored actors, on specific organizations or individuals. Crypto-jacking is the unauthorized use of someone else's computer to mine cryptocurrency. To combat these new types of cybercrime, organizations need to be vigilant and stay up to date with the latest threat intelligence and best practices for cybersecurity. This includes regularly reviewing and updating their security systems, as well as training employees on how to recognize and respond to cyber threats. Additionally, organizations should consider working with security experts or incident response teams to help them quickly detect, respond to, and recover from cyber incidents.

Cybercrime is a serious threat that affects individuals and organizations of all sizes. It is essential to take steps to protect against cybercrime, including implementing strong security measures, staying up-to-date with the latest threat intelligence, and raising awareness among employees and users. Another important aspect of protecting against cybercrime is incident response planning. Incident response planning is the process of preparing for and responding to cyber security incidents. It includes identifying potential incidents, determining the roles and responsibilities of various stakeholders, and establishing procedures and guidelines for responding to incidents. This can include procedures for containing and mitigating an incident, as well as procedures for reporting the incident to law enforcement and other relevant authorities .

Incident response planning also includes regular testing and exercises to help organizations identify and address any weaknesses in their incident response capabilities. This can include tabletop exercises, where participants discuss and simulate different incident scenarios, and live exercises, where participants carry out incident response procedures in a simulated environment. The process of collecting, preserving, analyzing, and reporting on evidence related to a cyber-incident. This can include collecting logs from devices and systems, analyzing network traffic, and conducting interviews with witnesses and suspects. The goal of incident forensics is to identify the cause of the incident, determine the extent of the damage, and gather evidence that can be used in legal proceedings.

Organizations can also consider purchasing cyber insurance, which can provide financial protection against losses caused by cybercrime. Cyber insurance policies can provide coverage for a wide range of cyber-related incidents, such as data breaches, network failures, and business interruptions. Cybercrime is a serious and constantly evolving threat that requires a comprehensive approach to protect against. This includes implementing strong security measures, incident response planning, incident forensics, and keeping up-to-date with the latest threat intelligence and best practices. Organizations should also consider purchasing cyber insurance to provide financial protection against losses caused by cybercrime.

Malware: Malware, short for malicious software, is any software designed to harm or exploit computer systems. There are several different types of malwares, each with their specific characteristics and effects:

Malware Types:

Viruses: These are programs that replicate themselves by attaching themselves to other files or programs on a computer. They can cause damage to files, slow down computer performance, and even render a computer inoperable.

Worms: These are self-replicating programs that can spread rapidly through networks, causing network congestion and slowing down or crashing systems.

Trojan horses: These are programs that appear to be legitimate but contain hidden functionality that can cause damage or steal sensitive information.

Ransomware: These are malware that encrypts the victim's files, rendering them inaccessible, and demands a ransom payment in exchange for the decryption key.

Adware: These are programs that display unwanted ads on the user's computer or browser, often in the form of pop-ups or banners.

Spyware: These are programs that collect sensitive information such as keystrokes, passwords, and browsing history without the user's knowledge or consent.

Rootkits: These are programs that can hide the presence of other malware or the attacker's activities by modifying the system's kernel, making it difficult to detect or remove.

Malware can be delivered to a computer in a number of ways, including email attachments, infected software downloads, and drive-by downloads. To protect against malware, it's important to keep software and operating systems up to date use reputable antivirus and anti-malware software, be cautious when clicking on links or downloading files from unknown sources, and use strong, unique passwords for all accounts .

In addition, it's important to be mindful of phishing attempts, which are designed to trick people into providing personal information or clicking on malicious links. This can include emails or text messages that appear to be from a legitimate source, such as a bank or government agency, but are actually from a cybercriminal .

It's also a good practice to regularly backup important files, so that in case of a ransomware attack, you can restore your files from the backup. Organizations should also consider implementing security measures such as firewalls, intrusion detection systems, and endpoint protection to protect their networks and devices from malware. They should also have incident response plans in place to respond to malware incidents quickly and effectively.

In addition, employees should be trained on how to recognize and respond to potential malware threats, and to follow safe practices such as not clicking on links or downloading files from unknown sources.

It's important to note that, even with the best security measures in place, it's impossible to completely prevent malware infections. That's why it's important to have a plan in place for detecting, responding, and recovering from malware incidents. Another important aspect of protecting against malware is to be vigilant about software updates and patches. Software vendors often release updates and patches to fix known vulnerabilities in their products, and it's important to install these updates as soon as they are available. This includes updates for operating systems, web browsers, and third-party software .

Segmentation strategy is the process of dividing a network into smaller, isolated segments, so that if malware does penetrate the network, it will be confined to a specific segment and will be less likely to spread to the rest of the network. This can include using virtual LANs (VLANs) to segment the network, as well as using firewalls and other security devices to control network traffic. Sandboxing is the process of running a program in a virtualized environment, where it can be observed and analyzed without affecting the rest of the system. This allows organizations to test and analyze unknown files and programs, to detect any malicious behavior. It's important for organizations to have incident response plans in place and to regularly test and exercise those plans. This will help them to quickly and effectively detect, respond, and recover from malware incidents.

Protecting against malware requires a multi-layered approach that includes using security software, being vigilant about software updates and patches, using network segmentation, sandboxing, incident response planning and regular testing. Additionally, training employees on safe computing practices and promoting a culture of security can help to minimize the risk of malware infections. Another important aspect of protecting against malware is to have a robust incident response program in place. This includes having incident response teams who are trained to detect, respond and recover from malware incidents, and having a clear incident response plan that outlines the steps to be taken in the event of a malware incident .

Organizations should have incident response teams in place to respond to a malware incident. These teams should be composed of individuals with expertise in areas such as network security, incident response, and forensic analysis. The incident response team should be responsible for coordinating the response to a malware incident, and should have the necessary tools and resources to detect, contain, and eradicate malware from the organization's systems. It's also important to have a clear incident response plan in place, which should outline the steps to be taken in the event of a malware incident. This plan should include procedures for identifying, containing and eradicating malware, as well as procedures for reporting the incident to law enforcement and other relevant authorities.

Incident forensics is the process of collecting, preserving, analyzing, and reporting on evidence related to a malware incident. This can include collecting logs from devices and systems, analyzing network traffic, and conducting interviews with witnesses and suspects. The goal of incident forensics is to identify the cause of the incident, determine the extent of the damage, and gather evidence that can be used in legal proceedings.

Organizations should communicate to their employees and other stakeholders, during and after an incident, to keep them informed of the situation, and to provide guidance on how to stay safe and protect their data. Protecting against malware requires a comprehensive approach that includes using security software, being vigilant about software updates and patches, using network segmentation, sandboxing, incident response planning, incident forensics, and regular testing. Additionally, having an incident response program and keeping stakeholders informed can help organizations respond effectively to malware incidents.

Cyber Stalking: Cyberstalking is the use of the internet, digital communication tools, or other forms of technology to stalk, harass, or otherwise harm an individual. This can include actions such as sending threatening or obscene messages, posting private information online, or using technology to track an individual's movements . Cyberstalking can take many forms and can be committed by individuals or groups. It can be directed at individuals, families, or even entire organizations. Some common forms of cyberstalking include:

Harassment: sending threatening or obscene messages, making unwanted phone calls, or sending unwanted emails

Reputation damage: spreading false or defamatory information about the victim online

Privacy invasion: posting private or personal information about the victim online

Identity theft: stealing the victim's personal information and using it to commit fraud or other crimes

Cyberbullying: using technology to bully, intimidate or harass an individual or group.

Cyberstalking can have serious consequences for the victim, including emotional distress, fear, and anxiety. In some cases, it can also lead to physical harm or even death. It is important for

victims of cyberstalking to take steps to protect themselves, such as blocking the stalker's contact information, changing passwords and email addresses, and being cautious about sharing personal information online.

It's also important to report cyberstalking to the authorities, as it is a criminal offense in many countries, and the police can help to investigate and prosecute the perpetrator. To prevent cyberstalking, organizations can implement security measures such as firewalls, intrusion detection systems, and endpoint protection to protect their networks and devices. They can also implement policies and procedures to prevent and respond to cyberstalking, and train employees on how to recognize and respond to cyberstalking .

Victims of cyberstalking may also seek help from legal and other professional services. A restraining order is a court order that can prohibit the stalker from contacting the victim. This may include in-person contact, phone calls, text messages, and other forms of communication. In some cases, victims of cyberstalking may also seek help from online safety organizations or support groups, which can provide counseling, advice, and resources on how to cope with cyberstalking.

It is also important for organizations to take cyberstalking seriously and to have policies and procedures in place to address cyberstalking incidents. This includes providing employees with clear guidance on how to report cyberstalking, and ensuring that the incident is investigated and handled in a timely and appropriate manner.

Organizations can also take steps to protect their employees from cyberstalking by providing them with training on how to recognize and respond to cyber stalking, as well as providing them with tools and resources to help them stay safe online. It is also important for society as a whole to address cyberstalking by raising awareness about the issue, and by promoting a culture of respect and responsibility online. This can include promoting education and awareness campaigns and encouraging the development of technologies and tools that can help to prevent and address cyberstalking .

Cyberstalking is a serious crime that can have serious consequences for the victims. It's important for victims to report cyberstalking to the authorities, seek legal and professional help, and use protective measures to protect themselves. Take steps to prevent and address cyberstalking, by implementing security measures, having policies and procedures in place, providing training and education, and promoting a culture of respect and responsibility online. It is important to note that some people may not be aware that their behavior constitutes cyberstalking and may not intend to harm the victim, however, it is important to take action to stop this behavior.

It is also important for organizations to have a clear and comprehensive cyberstalking policy in place that outlines the procedures for dealing with cyberstalking incidents, and the consequences for those who engage in cyberstalking. This policy should also guide how to support victims of cyberstalking and how to report cyberstalking incidents. It is also important for society to have a zero-tolerance approach towards cyberstalking to raise awareness of the issue and educate people on how to recognize and report cyberstalking. This can include working with law enforcement and other organizations to promote education and awareness campaigns and encouraging the development of technologies and tools that can help to prevent and address cyberstalking .

Cyberstalking is a serious crime that can have serious consequences for victims. It's important for victims to report cyberstalking to the authorities, seek legal and professional help, and use protective measures to protect themselves. Organizations and society as a whole should also

take steps to prevent and address cyberstalking, by implementing security measures, having policies and procedures in place, providing training and education, and promoting a culture of respect and responsibility online.

Another important aspect of addressing cyberstalking is to work with online platforms and service providers. Many cyberstalking incidents occur on social media and other online platforms, and these companies need to have policies and procedures in place to respond to cyberstalking reports and remove harmful content. Many online platforms have reporting mechanisms in place for users to report cyberstalking, and they also have teams that review and investigate reported content. Additionally, organizations and individuals can work with law enforcement agencies to bring cyberstalks to justice.

It's also important for victims of cyberstalking to document the cyberstalking incidents. This can include saving messages and other forms of communication from the cyberstalked, as well as keeping a log of the incidents, including dates, times, and any other relevant information. This can be used as evidence to support a criminal investigation or a restraining order.

Computer Hacking: Computer hacking is the unauthorized access or manipulation of a computer system or network. Hackers use a variety of techniques, including exploiting vulnerabilities in software, using stolen credentials, and social engineering to gain access to a computer system or network. Once they have gained access, hackers can use a variety of tools and techniques to extract information, disrupt operations, or take control of the system. Some common types of computer hacking include:

Network hacking: unauthorized access to a computer network to extract information or disrupt operations

Website hacking: unauthorized access to or manipulation of a website's content or functionality

Password cracking: guessing or using specialized tools to determine a user's password

Phishing: using social engineering techniques to trick users into providing their login credentials or other sensitive information

Malware: using malicious software to gain access to or control over a computer system

Advanced persistent threats (APT): long-term, targeted attacks on specific organizations or individuals, often by state-sponsored actors.

Hacking can have serious consequences for individuals and organizations, including financial losses, loss of sensitive information, and damage to reputation. To protect against hacking, it's important to keep software and operating systems up to date, use strong and unique passwords, and be cautious about clicking on links or downloading files from unknown sources .

Organizations should also implement security measures such as firewalls, intrusion detection systems, and endpoint protection to protect their networks and devices from hacking. Additionally, employee awareness and education play an important role in preventing hacking. Employees should be trained on how to recognize and respond to potential hacking threats, and to follow safe computing practices such as not clicking on links or downloading files from unknown sources.

Another important aspect of protection against hacking is to have a robust incident response program in place. This includes having incident response teams who are trained to detect,

respond and recover from hacking incidents, and having a clear incident response plan that outlines the steps to be taken in the event of a hacking incident .

Regular vulnerability assessments and penetration testing can also help organizations identify and address vulnerabilities in their systems before they can be exploited by hackers. This can include using automated tools to scan for vulnerabilities, as well as manually testing systems and networks to identify potential weaknesses.

Organizations should also have a data backup and recovery plan in place, in case of a successful hacking attempt that results in data loss. This should include regularly backing up important files and data, and having a plan in place to quickly restore systems and data in the event of a breach.

Implementing multi-factor authentication (MFA) can also help protect against hacking, as it adds a layer of security to login processes by requiring users to provide multiple forms of proof of identity.

In addition, organizations should also have an incident management and reporting process in place, which includes procedures for reporting and escalating incidents, and for communicating with employees and other stakeholders during and after an incident. Organizations should also stay up to date with the latest threat intelligence and best practices to stay aware of new and evolving hacking techniques and to adapt their security measures accordingly.

Protecting against hacking requires a multi-layered approach that includes keeping the software and operating systems up to date, using strong and unique passwords, being cautious about clicking on links or downloading files from unknown sources, incident response planning, data backup and recovery, vulnerability assessments and penetration testing, multi-factor authentication, incident management, and reporting, and staying up to date with the latest threat intelligence and best practices .

Encryption is the process of converting plain text into unreadable cipher text, which helps to protect sensitive information from being intercepted or accessed by unauthorized parties. Organizations can use encryption to protect data both in transit and at rest. This includes using SSL/TLS to encrypt data transmitted over the internet and using disk encryption to encrypt data stored on computers and servers.

Another important aspect of protection against hacking is to use access controls and authentication mechanisms. This includes using strong and unique passwords and using multi-factor authentication (MFA) to add a layer of security to login processes. Organizations can also use role-based access controls to restrict access to systems and data based on an individual's role and responsibilities within the organization .

Network security technologies such as firewalls, intrusion detection and prevention systems (IDPS), and virtual private networks (VPNs). These technologies can help to protect networks and systems from unauthorized access, and to detect and respond to hacking attempts in real time. Organizations should also have a comprehensive incident response plan in place, which includes procedures for detecting, responding, and recovering from hacking incidents. This plan should also include incident reporting procedures, which should be followed in the event of a security incident.

Protecting against hacking requires a comprehensive approach that includes using encryption, access controls and authentication mechanisms, network security technologies, incident response planning, as well as staying up to date with the latest threat intelligence and best

practices. Additionally, it's important for employees to be trained on safe computing practices and to be aware of the risks.

Encryption: Encryption is the process of converting plain text into unreadable cipher text using a mathematical algorithm, known as a cipher. The purpose of encryption is to protect sensitive information from being intercepted or accessed by unauthorized parties. There are two main types of encryption: symmetric and asymmetric . Symmetric encryption uses a single secret key to encrypt and decrypt the data. The same key is used for both encryption and decryption, and it must be kept secret to maintain the security of the data. Examples of symmetric encryption algorithms include AES and Blowfish.

Asymmetric encryption, also known as public-key encryption, uses a pair of keys: a public key and a private key. The public key is used to encrypt the data, and the private key is used to decrypt it. The public key can be freely distributed, while the private key must be kept secret. Examples of asymmetric encryption algorithms include RSA and Elliptic Curve Cryptography (ECC).

Encryption can be used to protect data at rest, such as data stored on a computer or server, and data in transit, such as data transmitted over a network. In data at rest, encryption can be done by disk encryption, which encrypts the entire disk or specific files or folders. This is useful for laptops or mobile devices which are at high risk of being lost or stolen. In data in transit, encryption can be done by using Transport Layer Security (TLS) or Secure Sockets Layer (SSL) protocol, which encrypts data transmitted over the internet. HTTPS is an example of this and is commonly used for online transactions and communications.

Encryption is the process of converting plain text into unreadable cipher text using a mathematical algorithm, known as a cipher. There are two main types of encryption: symmetric and asymmetric. Encryption can be used to protect data at rest, such as data stored on a computer or server, and data in transit, such as data transmitted over a network . Digital certificates are used to establish the identity of a website or an individual and to secure communications between them. They are issued by a trusted third party, known as a certificate authority (CA), and can be used to encrypt and sign electronic communications, such as email or web traffic.

Encryption can also be used in virtual private network (VPN) technology, which creates a secure, encrypted tunnel between a device and a network, allowing for secure remote access to a network. Email encryption protects the confidentiality of email messages. This can be done by encrypting the entire message or specific parts of the message, such as attachments. Another use case is the encryption of backups, which is important to protect sensitive information in case of data loss or disaster. Encryption key management is also an important aspect of encryption. This includes the generation, distribution, storage, and destruction of encryption keys. It's essential to have a robust key management system in place to ensure that encryption keys are kept secure and that access to them is controlled and auditable.

Encryption is a critical component of cyber security and it helps to protect sensitive information from being intercepted or accessed by unauthorized parties. Encryption can be used to protect data at rest, such as data stored on a computer or server, and data in transit, such as data transmitted over a network. Additionally, it's important to use digital certificates, use encryption in virtual private network (VPN) technology, email encryption, encryption of backups, and have a robust encryption key management system in place .

E2EE is a method of encrypting data so that only the sender and the intended recipient can read it, even if it passes through intermediaries such as servers or cloud providers. This ensures that

the data is only accessible to the parties involved in the communication and not to any third parties. Some examples of services that use E2EE are WhatsApp, I Message, and Signal.

Another important aspect of encryption is the use of encryption in cloud computing. Many organizations are moving their data and applications to cloud-based services, which can expose them to new security risks. To protect data in the cloud, organizations should use encryption to protect data both at rest and in transit, as well as use secure protocols such as HTTPS and SSH. Additionally, organizations should also use cloud access security brokers (CASBs) to monitor and control access to cloud-based services. In addition, organizations should also be aware of the legal and regulatory requirements related to encryption. Different countries have different laws and regulations regarding the use of encryption, and organizations should be aware of these requirements to ensure they comply.

Encryption is a critical component of cyber security, and it helps to protect sensitive information from being intercepted or accessed by unauthorized parties. Organizations should use encryption to protect data both at rest and in transit, use digital certificates, use encryption in virtual private network (VPN) technology, email encryption, encryption of backups, and have a robust encryption key management system in place. Additionally, organizations should be aware of the legal and regulatory requirements related to encryption, use end-to-end encryption (E2EE), and encryption in cloud computing, and use cloud access security brokers (CASBs) to monitor and control access to cloud-based services . Homomorphic encryption is a type of encryption that allows computations to be performed on cipher text, without the need to decrypt the data first. This can be useful in scenarios where sensitive data needs to be processed by third parties, but where the data cannot be decrypted due to regulatory or privacy concerns.

Post-quantum encryption with the advent of quantum computing, traditional encryption methods such as RSA and ECC may become vulnerable to attacks. Post-quantum encryption methods, such as lattice-based cryptography and code-based cryptography, are designed to be resistant to attacks from quantum computers. Hardware security modules (HSMs) are specialized devices that are designed to securely store encryption keys and perform cryptographic operations. They can be used to protect encryption keys from theft or attack, and can also be used to perform encryption and decryption operations in a secure environment.

Encryption is a critical component of cyber security, and it helps to protect sensitive information from being intercepted or accessed by unauthorized parties. Organizations should use encryption to protect data both at rest and in transit, use digital certificates, use encryption in virtual private network (VPN) technology, email encryption, encryption of backups, and have a robust encryption key management system in place. Additionally, organizations should be aware of the legal and regulatory requirements related to encryption, use end-to-end encryption (E2EE), encryption in cloud computing, and use cloud access security brokers (CASBs) to monitor and control access to cloud-based services, use homomorphic encryption, post-quantum encryption and hardware security modules (HSMs) and also be aware of the risks associated with using encryption.

Introduction to Digital Signature: A digital signature is a technique used to ensure the authenticity and integrity of digital documents and messages. It is similar to a physical signature on a paper document, but it uses cryptographic techniques to ensure that the signature is unique and cannot be forged. Digital signatures are created by using a combination of a private key and a hashing algorithm. The sender uses their private key to create a unique digital signature, which is then appended to the document or message. The recipient can then use the sender's public key to verify the signature and confirm that it was indeed created by the sender.

The digital signature serves multiple purposes:

Authentication: It confirms the identity of the sender of the document or message.

Integrity: It ensures that the document or message has not been tampered with during transit.

Non-repudiation: It ensures that the sender cannot deny having sent the document or message.

Digital signatures rely on Public Key Infrastructure (PKI) to establish trust in the digital identity of the signer. PKI uses digital certificates that are issued by a trusted third party (Certificate Authority) to verify the identity of the signer and ensure that their public key is valid. Digital signatures are commonly used in various applications such as electronic transactions, email communications, and software distribution. They are also used in various industries such as finance, healthcare, and government to ensure the authenticity and integrity of sensitive information .

A digital signature is a technique used to ensure the authenticity and integrity of digital documents and messages. It uses cryptographic techniques to ensure that the signature is unique and cannot be forged, and relies on Public Key Infrastructure (PKI) to establish trust in the digital identity of the signer. Digital signatures are commonly used in various applications and industries. Another important aspect of digital signatures is the use of digital signature standards, such as the Digital Signature Algorithm (DSA) and the RSA algorithm. These standards provide a set of guidelines for creating and verifying digital signatures and ensure that digital signatures are created and verified consistently and securely .

Time stamping is the process of adding a timestamp to a digital signature, which confirms when the signature was created. This can be useful in situations where the authenticity of a document or message needs to be proven at a later date. Qualified digital certificates are digital certificates that meet certain legal requirements and are issued by a qualified trust service provider (QTSP). They are commonly used in electronic transactions that are subject to legal regulations, such as e-invoicing and e-procurement.

It's important to note that digital signatures are only as secure as the underlying encryption and PKI infrastructure. Organizations should ensure that they are using secure encryption methods and that their PKI infrastructure is robust and well-managed. Additionally, organizations should be aware of the risks associated with digital signatures, such as the risk of certificate fraud, and should have incident response plans in place in case of a security incident .

Digital signatures are an important technique used to ensure the authenticity and integrity of digital documents and messages. They rely on Public Key Infrastructure (PKI) to establish trust in the digital identity of the signer and use digital signature standards, timestamping, and qualified digital certificates to ensure consistency and security. Additionally, organizations should ensure the underlying encryption and PKI infrastructure is secure, be aware of the risks associated with digital signatures, and have incident response plans in place. Biometric authentication uses unique physical or behavioral characteristics, such as fingerprints, facial recognition, or voice recognition, to confirm the identity of the signer. This can provide an additional layer of security and make it more difficult for an attacker to forge a digital signature.

Block chain is a distributed ledger technology that can be used to create tamper-proof digital signatures. It allows for a decentralized and transparent way to verify the authenticity of a signature and can be used in a variety of applications such as supply chain management, voting systems, and identity verification. Zero-knowledge proofs (ZKPs) are cryptographic method that allows one party to prove to another party that a certain statement is true, without revealing any additional information. This can be useful in situations where the signer wants to prove

their identity without revealing sensitive information, such as in anonymous voting systems or privacy-sensitive applications.

Digital Signatures rely on Public Key Infrastructure (PKI) to establish trust in the digital identity of the signer and use digital signature standards, timestamping, qualified digital certificates, biometric authentication, block chain technology, and zero-knowledge proofs (ZKPs) to ensure consistency, security, and privacy. Organizations should ensure the underlying encryption and PKI infrastructure is secure, be aware of the risks associated with digital signatures, and have incident response plans in place. Digital signature software is used to create, manage and verify digital signatures. It can be integrated with other applications such as document management systems, electronic signature platforms, and workflow management systems. Some examples of digital signature software include Adobe Sign, DocuSign, and Hello Sign .

Signature pads are specialized input devices that allow users to create digital signatures by signing on a digital surface with a pen or stylus. This allows users to sign paperless documents as if they were signing a physical document. In Mobile devices, many digital signature software providers offer mobile apps that allow users to create, manage and verify digital signatures on mobile devices. This can be useful for users who need to sign documents while on the go or in remote locations.

Antivirus: An antivirus is a type of software that is designed to prevent, detect, and remove malware, such as computer viruses, worms, Trojan horses, and other malicious software. Antivirus programs typically use a combination of techniques to identify malware, including signature-based detection, which looks for known patterns of data that are associated with known malware, and heuristic-based detection, which uses algorithms to identify suspicious behavior that may indicate the presence of malware. Some antivirus programs also include additional features such as firewalls, intrusion detection, and anti-phishing protection .

Antivirus software runs on a computer or mobile device and is designed to detect and remove malware that may already be present on the device or to prevent malware from being downloaded or installed in the first place. Antivirus programs use a variety of techniques to detect malware, including signature-based detection, which compares files on the device to a database of known malware signatures, and heuristic-based detection, which uses algorithms to identify and flag suspicious behavior that may indicate the presence of malware .

It's important to keep in mind that while antivirus software can be an effective tool in the fight against malware, it's not foolproof. New malware is being created all the time, and malware can evade detection by antivirus programs. Additionally, some malware is designed specifically to disable or bypass antivirus software. To stay protected, it's important to keep antivirus software up to date, as well as practice safe browsing habits and be cautious about opening email attachments or clicking on links from unknown sources.

There are many antivirus software available in the market, some popular ones are Norton, McAfee, Kaspersky, AVG, Avast, etc. Some of this software are paid and some are free. It's important to choose reputable software and keep it updated regularly to ensure maximum protection.

In addition to signature-based detection and heuristic-based detection, many antivirus programs also use other techniques to detect and remove malware. Some examples include:

Behavioral-based detection: This method monitors the behavior of programs and processes running on a device, looking for any suspicious or malicious behavior that may indicate the presence of malware.

Cloud-based detection: Some antivirus programs use cloud-based databases to identify and block malware in real time. This can be particularly effective against new or unknown malware that has not yet been added to the program's local signature database.

Sandboxing: This is a method where the antivirus software runs a potentially suspicious program in an isolated environment so that if it is malware, it will not be able to harm the computer.

It's also important to note that antivirus software alone is not enough to protect your device from all types of cyber threats. Other security measures, such as firewalls, intrusion detection systems, and anti-phishing software, can provide additional layers of protection.

It's also important to keep your operating system and other software up to date to address vulnerabilities that malware could potentially exploit. Additionally, maintaining strong passwords and avoiding clicking on links or opening attachments from unknown sources can also help to reduce the risk of infection.

Another important aspect of antivirus software is the ability to regularly update the program's malware signature database. As new malware is created and existing malware evolves, the signature database must be updated to include the latest information about known malware. Without regular updates, an antivirus program may not be able to detect and remove the latest threats .

It's also essential to note that antivirus software is not just for personal computers or laptops, but it's also available for mobile devices like smartphones and tablets. Mobile malware is becoming increasingly common, and it's important to protect these devices as well. Mobile antivirus apps typically include many of the same features as their desktop counterparts, such as signature-based detection, heuristic-based detection, and real-time scanning. It's also important to note that some antivirus software may have limitations, such as not being able to detect or remove certain types of malware, or not being able to protect against all types of cyber-attacks. Therefore, it's crucial to do some research before choosing antivirus software, read reviews, and compare features to find the one that best suits your needs.

Antivirus software can play a critical role in protecting your device and personal information from malware, but it's not a complete solution. Regular updates, safe browsing habits, and other security measures are also important to maintain the overall security of your device. Another important aspect of antivirus software is the ability to perform regular full-system scans. These scans check the entire computer or mobile device for any signs of malware and remove any malware that is found. It's important to schedule regular scans, as malware can be hidden and may not be detected by real-time scanning alone.

The ability of antivirus software to block malicious websites checks the websites you visit against a list of known malicious websites and blocks them if they are found to be harmful. This can help protect against phishing attacks and other types of cyber threats that can be delivered via a web browser . Additionally, some antivirus software also includes features such as parental controls and device control, which can help protect children from inappropriate content or limit access to certain devices or files on your computer.

It's also important to note that the effectiveness of antivirus software can vary depending on the vendor and the specific product. It's a good idea to check for independent test results from

reputable organizations, such as AV-Test or AV-Comparatives, to see how well a particular product performs in detecting and removing malware. Antivirus software is an important tool in the fight against malware, but it's not a one-stop solution. It's important to keep the software up to date, regularly scan your device, practice safe browsing habits and use other security measures to maintain the overall security of your device.

Ransomware is a type of malware that encrypts files on a computer or mobile device and demands payment in exchange for the decryption key. Ransomware protection features can help prevent this type of attack by monitoring for suspicious activity and blocking any attempts to encrypt files. Some antivirus software also includes the ability to perform a boot-time scan. This scan is performed before the operating system starts and can help detect and remove malware that may be hidden in the boot process.

Steganography: Steganography is the practice of hiding information within other information. It is used to conceal the existence of a message or data within an image, audio, video, or other types of digital media. The goal of steganography is to make the hidden message difficult or impossible to detect so that it can be transmitted without being detected by an eavesdropper.

There are several methods of steganography, such as:

Least Significant Bit (LSB) insertion: This method involves replacing the least significant bit of the pixel values in an image with the bits of the hidden message.

Masking and filtering: This method involves applying a mask or filter to an image to conceal the hidden message within the image.

Algorithms: This method involves using an algorithm to embed the hidden message in a cover image or audio/video file.

Steganography can be used for both legitimate and malicious purposes. Legitimate uses include protecting sensitive information during transmission and verifying the authenticity of digital media. However, it can also be used by malicious actors to conceal malware or exfiltration of sensitive information without detection. Detection of steganography is a difficult task and requires specialized software and techniques. Steganalysis is the process of detecting and extracting hidden information from stego-objects. It can be used to detect the presence of hidden information, identify the steganography method used, and extract the hidden data.

Steganography is the practice of hiding information within other information. It can be used for both legitimate and malicious purposes. Detection of steganography is difficult and requires specialized software and techniques. It's important to be aware of the risks of steganography and take steps to protect your network and devices from this type of threat. Another method used in steganography is called "syntactic steganography" which is the process of hiding data within the structure of a file or protocol. This can include hiding data within fields that are not typically used or using fields in a way that is not typical. For example, a malicious actor could use the User-Agent field in an HTTP header to send a message that would not be noticed by someone inspecting the header .

Another method is called "semantic steganography" which is the process of hiding information within the meaning of a message. This can include hiding information within the text of a document, within the audio of a song, or within the pixels of an image. For example, a message could be hidden within the least significant bits of an image, but the image would still appear normal to the human eye. While steganography can be used for malicious purposes, it can also be used for legitimate purposes such as digital watermarking, which is the process of

embedding a hidden message or code within an image, audio, or video file to identify the ownership or authenticity of the file.

It's important to note that steganography is not only limited to digital media, but also can be applied to physical media as well, for example, by writing a message on the back of a photograph, or using invisible ink to write a message. Steganography is a technique that can be used to conceal the existence of a message or data within an image, audio, video, or other types of digital media, and can also be applied to physical media. There are different methods to use steganography such as Least Significant Bit (LSB) insertion, Masking, and filtering, Algorithms, syntactic steganography, semantic steganography, etc. It's important to be aware of the risks of steganography and take steps to protect your network and devices from this type of threat .

Another related concept is "covert channels" which is a method of transmitting information by exploiting the way a system or network is designed or configured. Covert channels can be used to exfiltration data from a network, for example by using the network's normal traffic to send a message, or by using the timing of network traffic to send a message. It's different from steganography in that it does not hide data within other data but it uses the underlying architecture or design of a system to transmit data in a way that is not intended. It's important to note that the detection of steganography and covert channels can be challenging, as the methods used can be very subtle and difficult to detect.

Prevention and detection methods include:

1. Regularly monitoring network traffic and looking for unusual patterns or anomalies.
2. Using intrusion detection and prevention systems (IDPS) to detect and block malicious traffic.
3. Implementing strict security policies and access controls to limit the ability of malicious actors to exfiltrate data or introduce malware.
4. Educating employees and users about the risks of steganography and covert channels and how to recognize and report suspicious activity.
5. Regularly updating and patching software and systems to address known vulnerabilities that could be exploited by malicious actors.

Steganography and covert channels are two related but distinct methods for hiding or transmitting information in a way that is not intended. Steganography is the practice of hiding information within other information, while covert channels are a method of transmitting information by exploiting the way a system or network is designed or configured. Both can be challenging to detect and requires a multi-layered security approach to protect against these types of threats.

The impact of steganography and covert channels on incident response and forensics. The use of these techniques makes it more difficult to detect and investigate an incident, as the hidden data or the covert channel may not be immediately apparent. Additionally, the use of these techniques can make it more difficult to attribute an incident to a specific actor, as the hidden data or the covert channel may not be directly linked to the attacker.

Computer Forensics: Computer forensics is the process of collecting, analyzing, and preserving digital evidence in a legally admissible manner. It is used to investigate and uncover evidence of cybercrime, such as hacking, identity theft, and other types of cyber-attacks, as well as other types of digital misconduct, such as embezzlement and corporate espionage.

The process of computer forensics typically involves several steps, such as:

Seizing and preserving digital evidence: This step involves collecting and preserving digital evidence from the crime scene in a manner that ensures the integrity of the evidence. This can include making an exact copy of the hard drive, creating a forensic image, or collecting specific files or data.

Analyzing digital evidence: This step involves analyzing the collected evidence to uncover relevant information. This can include analyzing network traffic logs, system images, and other types of data.

Presenting digital evidence: This step involves presenting the evidence in a legally admissible manner. This can include preparing reports, creating timelines, and other types of documentation.

Computer forensics is a complex and technical field that requires a combination of technical skills, analytical skills, and a thorough understanding of legal procedures and regulations .

Computer forensics specialists use various tools and techniques to collect and analyze digital evidence, such as:

Forensic software tools: These tools are used to create forensic images, analyze disk and file systems, recover deleted files, and perform other types of forensic analysis.

Live response tools: These tools are used to collect information from a running system, such as memory, network connections, and running processes.

Network forensics tools: These tools are used to analyze network traffic, such as packet captures and log files, to uncover evidence of cybercrime.

Computer forensics is the process of collecting, analyzing, and preserving digital evidence in a legally admissible manner. It is used to investigate and uncover evidence of cybercrime, and other types of digital misconduct. The process of computer forensics typically involves several steps such as seizing and preserving digital evidence, analyzing digital evidence, and presenting digital evidence . Computer forensics specialists use various tools and techniques to collect and analyze digital evidence. It requires a combination of technical skills, analytical skills, and a thorough understanding of legal procedures and regulations.

Computer forensics also plays an important role in incident response, which is the process of identifying, responding to, and mitigating the effects of security incidents. Incidents can include cyber-attacks, data breaches, and other types of security-related events. Computer forensics is often used during the incident response process to identify the cause of the incident, determine the scope of the damage, and collect evidence that can be used to hold the attackers accountable .

Another important aspect of computer forensics is the use of a chain of custody and proper handling of evidence. Chain of custody is the chronological documentation or paper trail that records the sequence of custody, control, transfer, analysis, and disposition of physical or electronic evidence. It is crucial to maintain the integrity of the evidence and be able to demonstrate that the evidence has not been tampered with or altered.

Computer forensics is also used in civil and criminal litigation. Digital evidence can be used to prove or disprove a claim in a legal case. Computer forensics experts can be called upon to provide expert witness testimony and to authenticate and present digital evidence in court.

It's important to note that computer forensics is a continually evolving field, as technology and cyber threats are constantly changing. Computer forensics specialists must stay current with the latest tools, techniques, and developments in the field to effectively collect and analyze digital evidence .

Computer forensics plays an important role in incident response, a chain of custody and proper handling of evidence is crucial for maintaining the integrity of the evidence. It's also used in civil and criminal litigation. As technology and cyber threats are constantly changing, computer forensics specialists need to stay current with the latest tools, techniques, and developments in the field to effectively collect and analyze digital evidence.

Another important aspect of computer forensics is the use of industry standards and best practices to ensure the integrity and reliability of digital evidence. Some of the most widely recognized standards and best practices for computer forensics include:

ISO/IEC 27037:2012: Guidelines for identification, collection, acquisition, and preservation of digital evidence.

The National Institute of Standards and Technology (NIST) Digital Forensics Process Reference Model (NIST SP 500-292): Provides a comprehensive overview of the digital forensics process, from the initial identification of a crime or incident to the final presentation of evidence in court.

The Scientific Working Group on Digital Evidence (SWGDE) guidelines: Provides best practices for the collection, examination, and preservation of digital evidence.

Another important aspect of computer forensics is the use of specialized training and certifications. Many organizations and government agencies require that computer forensics specialists have specialized training and certifications, such as the Certified Forensic Computer Examiner (CFCE) or the Certified Information Systems Security Professional (CISSP) to demonstrate their knowledge and expertise in the field.

Computer forensics is a continually evolving field that requires specialized training and certifications, adherence to industry standards and best practices, and staying current with the latest tools, techniques, and developments in the field. This ensures the integrity and reliability of digital evidence and can help to demonstrate the expert’s knowledge and expertise.

CHAPTER 11

BASICS OF CRYPTOGRAPHY

Dr. Prerna Mahajan, Professor,
Department of Computer Science and Information Technology, Jain (Deemed to be University) Bangalore, Karnataka, India
Email Id- prerna.m@jainuniversity.ac.in

Cryptography, according to the Concise Oxford Dictionary from 2006, is the practise of creating or deciphering codes. While historically correct, this definition does not accurately reflect contemporary cryptography. It begins by concentrating just on the issue of covert communication. This is shown by the word "codes," which is elsewhere described as "a system of pre-arranged signals, notably intended to maintain secrecy in transferring communications," being specifically included in the definition. The term also describes cryptography as an artistic endeavour. Cryptography was, in fact, an art form up until the 20th century (and maybe even until the latter half of that century). Good code construction or code breaking need individual ability and inventiveness. There was not even a clear definition of what makes a good code, and there was very little theory that could be depended upon.

This understanding of cryptography drastically evolved in the late 20th century. A thorough theory was developed, allowing for the scientifically rigorous study of cryptography. Apart from secret communication, the topic of cryptography today incorporates a lot more, such as message authentication, digital signatures, authentication protocols, protocols for transferring secret keys, electronic auctions and elections, and digital money. In fact, it may be claimed that current cryptography is concerned with issues that could occur in any distributed computation that is vulnerable to internal or external assault. Modern cryptography is the scientific study of methods for protecting digital data, transactions, and distributed computations, without trying to provide a definitive definition.

The usage of current encryption differs significantly from classical cryptography, which was popular until the 1980s. Traditionally, military and intelligence agencies were the biggest users of cryptography. But, cryptography is widely used nowadays! Cryptographic security measures are a standard feature of practically all computer systems. Every time a user visits a protected website, cryptography is used—often without the user's knowledge. In multi-user operating systems, access control is enforced via cryptographic techniques, and trade secrets are not extracted from stolen laptops by criminals. To prevent copying, software protection techniques include encryption, authentication, and other measures. To put it briefly, cryptography has evolved from an art that dealt with covert military communication to a science that helps protect systems for regular folks anywhere in the world. Also, it suggests that computer science's focus on cryptography is expanding.

Private-Key Encryption Settings: As was said before, the goal of cryptography in the past was to facilitate secret communication. In particular, cryptography was concerned with creating cyphers (today referred to as encryption methods) enabling secret communication between two parties who had already shared certain information. The private-key (or symmetric-key) arrangement is now referred to as the situation in which the communication parties disclose some secret information beforehand. We first explain the private-key configuration and encryption in broad terms before detailing several historical cyphers.

In a private-key setup, two people exchange a hidden piece of information called a key and use it to communicate covertly. The key is used by the party sending the communication to encrypt (or "scramble") the message before it is transmitted, and by the party receiving the message to decrypt (or "unscramble") the message and retrieve it after receipt. The information that is "scrambled" and actually sent from the sender to the receiver is referred to as the ciphertext; see Figure 1.1. The message itself is sometimes referred to as the plaintext. The communication (which is expected to occur over a public channel) between the parties is protected by the shared key from any third parties that may be listening in.

We emphasise that in this situation, the plaintext is converted into a ciphertext and back again using the same key. Since both sides possess the same key, which is used for both encryption and decryption, this explains why this configuration is also known as the symmetric-key configuration. Asymmetric encryption, which was first present, operates in a context where the sender and receiver do not exchange any secrets and unique keys are used for encryption and decoding.

Each system implementing private-key encryption implicitly assumes that the communication parties have some method of first exchanging a key in a confidential manner. (Take note that an eavesdropper may also get the key if one side just communicates it to the other over a public channel.) Since communication parties may physically meet in a secure place to agree on a key, this is not a major issue in military circumstances. Yet, parties cannot schedule any such physical encounter in many contemporary contexts. This is a major worry and really restricts the use of cryptographic systems that only use private-key techniques.

Despite this, private-key techniques are nevertheless often used in a variety of situations. One such situation is disc encryption, where the same user (at various times) uses a fixed secret key to both write to and read from the disc. Private-key encryption is also often used in combination with asymmetric techniques.

The encryption syntax. We now formalise the discussion from before. Three algorithms make up a private-key encryption scheme, or cypher: a technique for generating keys in step one, an encryption step in step two, and a decryption step in step three. The following features of these algorithms:

1. The algorithm for generating keys a key k is produced using the probabilistic method Gen according to a distribution selected by the scheme.
2. The ciphertext c is produced by the encryption algorithm Enc from an input key k and a plaintext m. We indicate the plaintext m's encryption with Enck's key k. (m).
3. The decryption method Dec produces a plaintext m from an input key k and ciphertext c. We indicate that Deck used the key k to decode the ciphertext c. (c).
4. A key space K (i.e., the set of all possible keys) is defined by the key generation process, while a plaintext space M (also known as a message space) is established by the encryption process.

As every ciphertext is created by encrypting some plaintext with a certain key, K and M define a collection of all ciphertexts that may exist, which we will refer to as C. Be aware that defining the three algorithms (Gen, Enc, and Dec) as well as the plaintext space M completes the definition of an encryption scheme.

Every encryption technique must hold that Deck(Enck(m)) = m for every key k produced by Gen and every plaintext message m M in order to be considered fundamentally valid.

An encryption system must, in other words, have the characteristic that ciphertext decryption gives the original message that was encrypted when the right key is used. Resuming our previous explanation, two parties who want to communicate as follows would utilise an encryption technique. To start, Gen is used to get the shared key k between the parties. When one party has to communicate the other a plaintext m, they would calculate c:= Enck(m), and then send the ciphertext c that results across the open channel. The second side calculates m:= Deck(c) to get the original plaintext after receiving c.

Principle of Keys and Kerckhoffs. According to the preceding formulation, if an adversary who is listening in knows both the key k shared by the two communicating parties and the algorithm Dec, they will be able to decode every communication between them. Because of this, the communication parties are required to disclose the key k in secret and keep it a secret from everyone else. So maybe they need to also keep Dec a secret? In addition, maybe it would be best if Gen and Enc's individual encryption methods remained a secret. In a document he wrote defining crucial design guidelines for military cyphers in the late 19th century, Auguste Kerckhoffs expressed his viewpoint on this subject. The need for the encryption technique to remain secret must not be a need, and it must be possible for the adversary to get it without suffering any consequences. This is considered to be one of the most crucial of these ideas and is now known simply as the Kerckhoffs' principle.

In other words, the encryption method itself shouldn't be kept a secret; instead, the people talking should only exchange the key as the secret information.

As long as the opponent is unaware of the key being used, an encryption scheme should be created with the goal of being safe even if all of its component methods are known to an adversary. To put it another way, the Kerckhoffs' principle dictates that security must only depend on the confidentiality of the key. There are two main justifications for Kerckhoff's premise. The first is that keeping the secret of a short key is far simpler for the parties to do than keeping the secrecy of an algorithm. Sharing and securely storing a little (let's say, 100-bit) string is simpler than sharing and doing the same for a programme that is thousands of times bigger. Additionally, when the secret information is in the form of a randomly generated string, it is improbable that it will be revealed (perhaps by an insider) or that it will be discovered by reverse engineering.

Another defence is that, in the event that the key is discovered, it is considerably simpler for the truthful parties to change the key than it is to alter the algorithm in use. In reality, it is best security practise to update a key often even when it has not been compromised, and replacing the software in use would be far more difficult. Finally, it will be much simpler for all parties to use the same algorithm, but different keys, than it will be for everyone to use a different programme (which would additionally depend on the party with whom they are communicating). This is true if many pairs of people (within a company, say) need to encrypt their communication.

The Kerckhoffs principle is now understood to require that the algorithms employed be made public in addition to recommending that security should not depend on algorithm secrecy. The theory of "security by obscurity," which holds that more security may be obtained by keeping a cryptographic method secret (or hidden from public view), contrasts sharply with this. The following are some benefits of "open cryptography design," in which the algorithm specifications are made available to the public:

1. Designs that have been published are subject to public review and are thus probably stronger. Experience over many years has shown that developing effective cryptographic systems is

quite challenging. So, when a scheme has been thoroughly examined and has survived several attacks, our trust in its security is much stronger.

2. It is preferable for security holes to be discovered and made public by "ethical hackers" rather than simply being known to harmful actors.

3. Reverse engineering of the code (or leakage via industrial espionage) presents a major danger to security if the system's security depends on the algorithm's secret. In contrast, the secret key cannot be reverse engineered since it is separate from the code.

4. Public design allows for the creation of standards.

The Kerckhoffs' concept, or the idea of open cryptographic design, is routinely disregarded, sometimes to fatal results, despite how straightforward and clear it may seem. We emphasise that only publicly tried and proven algorithms should be used, and that it is very risky to utilise a proprietary algorithm (i.e., a non-standardized algorithm that was created in secret by some corporation). Nowadays, there is absolutely no need to employ any other algorithms because to the abundance of decent ones that are both standardised and unpatentable.

We point out that Kerckhoffs also provided a list of additional guiding principles, one of which stipulates that a system must be practically, if not mathematically, incomprehensible. Modern cryptography is founded on this paradigm, as we shall learn later in this book, and all current cryptographic methods may theoretically be cracked given enough time, with the exception of completely secret encryption. These techniques can thus be deciphered theoretically but not practically.

Attacking situations. Our broad study of encryption comes to an end with a quick look at some fundamental forms of attacks against encryption techniques.

They are listed in order of severity:

1. The simplest kind of attack, a ciphertext-only assault, involves the adversary just seeing a ciphertext and trying to decipher the plaintext that was encrypted.
2. Known-plaintext attack: In this technique, the adversary discovers one or more pairs of plaintexts and ciphertexts that are encrypted with the same key. The adversary's goal is to then decipher the plaintext that was encrypted to produce a different ciphertext (for which it is unaware of the matching plaintext).
3. Chosen-plaintext assault: In this kind of attack, the antagonist may gain the encryption of any plaintext or plaintexts of its choosing. Then it makes an effort to decipher the plaintext that was encrypted in order to produce another ciphertext.
4. Attack using chosen ciphertext: The third and final sort of assault gives the enemy the power to decode any ciphertext(s) of their choosing. Once again, the adversary's goal is to decipher the plaintext that was encrypted to produce another ciphertext (whose decryption the adversary is unable to obtain directly).

The first two forms of assaults, it should be noted, are passive in that the attacker just gets some ciphertexts (and probably some related plaintexts as well) before launching its assault. The adversary may adaptively request whatever encryptions and/or decryptions it wants in the final two forms of assaults, in contrast.

Clearly realistic assaults include the first two categories mentioned above. The simplest attack to execute in reality is a ciphertext-only one; all the adversary has to do is eavesdrop on the public communication channel being used to send encrypted messages. It is presumed in a known-plaintext assault that some of the ciphertexts it examined included encrypted plaintext,

which the adversary also managed to access. This is often accurate since not All encrypted communications are private, at least temporarily. In a simple example, anytime two persons start a conversation, they may always encrypt a "hello" message. Using encryption to conceal quarterly financial figures until their publication date is a more complicated scenario. Anybody listening in on this communication and receiving the ciphertext will thereafter be able to decipher the plaintext. Hence, any good encryption method must continue to be safe even in the presence of a known-plaintext assault.

The two most recent ongoing assaults could seem odd and need for explanation. When do parties encrypt and decode at the whim of a foe? With regard to chosen-plaintext attacks and chosen-ciphertext attacks, we postpone a more in-depth examination of these attacks until the sections of the text where protection against them is properly established.

We end by pointing out that various setups could need resistance to various attack types. Since it may be less effective than an encryption method safe against "weaker" assaults, it is not necessarily necessary to utilise an encryption scheme that is secure against the "strongest" form of attack. Instead, the latter may be used if it is sufficient for the application at hand.

The Fundamentals of Modern Cryptography

The major ideas and paradigms that set contemporary cryptography apart from the classical cryptography we covered in the previous section are outlined in this section, along with an emphasis on the scientific aspect of current encryption. Name three fundamental ideas:

Principle 1: Definitions Must Be Accurate

The insight that formal definitions of security are necessary precondition for the creation, use, or study of any cryptographic primitive or protocol has been one of the major intellectual achievements of contemporary cryptography. Let's go through each of them one at a time:

Significance for design: Let's assume that we want to build a safe encryption system. How can we possible know if (or when) we have done it if we do not have a clear knowledge of what it is we are trying to accomplish? A definition in mind makes it possible to assess the calibre of what we create and directs us towards creating the appropriate item. In instance, it is far preferable to identify what is required before starting the design process than to articulate what has been accomplished once the design is complete. The latter strategy runs the danger of having the design process finish before the objective has been reached (rather than when it has), or it may lead to a building that accomplishes more than is required and is thus less efficient than a superior option.

Usefulness: Let's say we wish to apply an encryption method within a bigger system. How do we decide the encryption method to employ?

How can we determine if an encryption method, if provided, is enough for our application? Having a specific description of the security produced by a particular scheme (combined with a security demonstration relative to a formally-stated assumption as outlined in principles 2 and 3) enables us to address these questions. To be more precise, we may specify the level of security we need for our system and then determine if an encryption method meets that need. Instead, we may define the criteria that the encryption scheme has to meet and then search for an encryption scheme that does.

The value of studying: How can two encryption techniques be compared? Efficiency is the sole criteria for comparison when there is no definition of security, however efficiency alone is a weak criterion since a very efficient method that is wholly unsecure is useless. Another

point of comparison is the precise definition of the degree of security that a method achieves. Two equally efficient systems are preferred over one another if the first one meets a stricter criterion of security than the second.

Conversely, there could be a compromise between security and effectiveness (see the previous two arguments), but at least with clear definitions, we can know what this compromise includes.

Dependence on Exact Assumptions is a second principle.

The majority of contemporary cryptographic structures cannot be categorically demonstrated to be safe. This is because they depend on computational complexity theory concerns that now appear to be a long way from being resolved. This regrettable situation has the effect of making assumptions a common part of security. Modern cryptography's second rule is that presumptions must be clearly stated.

This is because of two key factors:

Verification of the supposition Assumptions are by definition unproven claims that are instead conjectured to be true.

It is important to research the assumption in order to support this supposition. The fundamental idea is that the more often an assumption is examined without being contradicted, the more certain we are that it is correct. Studying an assumption may also demonstrate that it is indicated by another commonly held assumption, which is positive proof of its validity.

The assumption cannot be examined and (perhaps) disproved if it is not clearly expressed and presented. Consequently, having a clear description of what is assumed is a prerequisite to increasing our trust in it comparison of plans in cryptography, it happens often that we are faced with two schemes, each of which may be shown to meet a certain definition with regard to a different premise. Which plan should be chosen, if both are equally effective? The first scheme is to be preferred if the first assumption is stronger than the second assumption on which the second scheme is predicated (i.e., the second assumption implies the first assumption). This is because it may turn out that the first assumption is true while the second assumption is false. The general rule is to favour the scheme that is based on the more thoroughly researched assumption where the assumptions employed by the two schemes are incomparable (for the reasons outlined in the preceding paragraphs).

Facilitating a security proof: Modern cryptographic structures are offered together with proofs of security, as we have indicated and will go into more detail about in concept 3. A mathematical demonstration that "the construction is safe if the assumption is true" can only be given if there is a clear description of the assumption if the scheme's security cannot be established unconditionally and must depend on some assumption.

Principle 3: Strict Security Proofs: The previous two ideas are naturally related to this one.

Contemporary cryptography emphasises the need of thorough security justifications for proposed systems. Such a demonstration of security is feasible because specific definitions and presumptions are employed. So why is a proof required? The primary issue is that a construction's or protocol's security cannot be verified in the same way that software is usually verified. For instance, just because encryption and decryption "work" and the ciphertext seems jumbled does not guarantee that a knowledgeable opponent cannot defeat the technique. We must depend on our intuition that this is the case as we lack a proof that no opponent of the required power can defeat the system. Of know, intuition is generally a pretty difficult thing.

In reality, experience has shown that using intuition when it comes to computer security and encryption is catastrophic. Many instances of dubious ideas that were abandoned exist (sometimes right away, sometimes years after being presented or even put into use).

The potential harm that might be caused by using an unsafe system is another factor in why proofs of security are crucial. Even while software errors may sometimes be highly expensive, a bank might suffer a great deal of harm if its authentication or encryption systems were compromised. Lastly, it should be noted that despite the fact that software often has defects, most users do not intentionally aim to make their programme malfunction. In contrast, attackers assault security measures with the explicit goal of breaching them using very elaborate and intricate methods (using particular aspects of the structure). So, even if they are always desired in computer science, proofs of correctness are vitally necessary in the fields of cryptography and computer security. We emphasise that the conclusions drawn from the aforementioned findings are not only theoretical; rather, they are the result of years of practical research and experience that have taught us not to trust our intuition in this area.

CHAPTER 12

NETWORK LAYER

Dr. Solomon Jebaraj, Assistant Professor,
Department of Computer Science and Information Technology, Jain (Deemed to be University) Bangalore, Karnataka, India
Email Id- solomon.j@jainuniversity.ac.in

An IPv4 address is a 32-bit number that specifically and globally identifies how a host or router connects to the Internet. As each IPv4 address designates a single connection to the Internet, they are distinctive. Each host that wishes to connect to the Internet must accept the IPv4 addressing scheme, making IPv4 addresses ubiquitous. Address Space: The entire number of addresses utilised by the protocol is known as the address space. Since each bit may have either a value of 0 or 1, the address space is 2b if a protocol employs b bits to specify an address. The address space for IPv4 is 232 bits, or 4,294,967,296 (more than four billion), since it employs 32-bit addresses. More than 4 billion devices could be linked to the Internet if there were no limitations.

Notation

The three most popular ways to display an IPv4 address are in binary (base 2), dotted decimal (base 256), and hexadecimal (base 16), as shown in Figure 12.1.

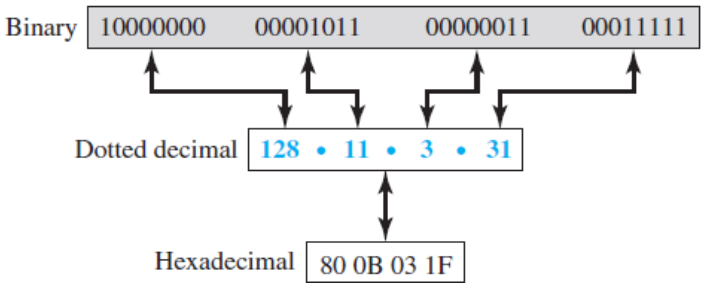


Figure 12.1: Three different notations in IPv4 addressing

Hierarchy in Addressing: While it is just separated into two parts, a 32-bit IPv4 address is also hierarchical. The prefix, or first part, of the address designates the network, while the suffix, or second part, and designates the node (connection of a device to the Internet). The suffix length is (32-n) bits, whereas the prefix length is n bits (Figure 12.2).

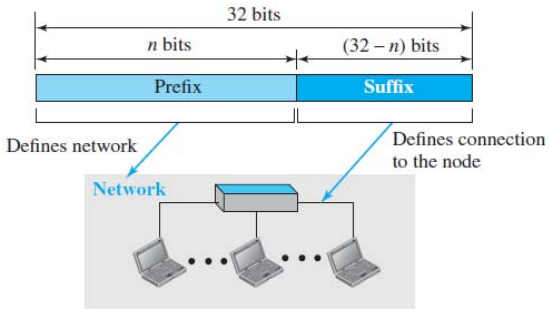
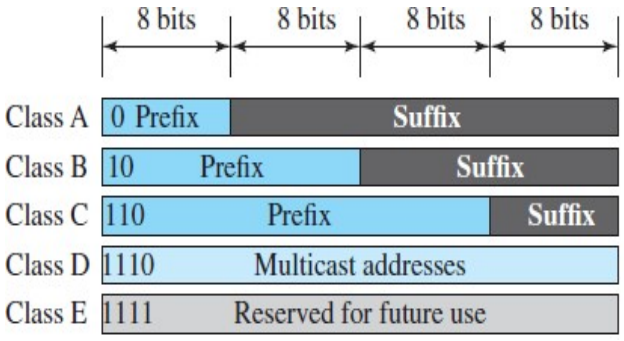


Figure 12.2: Hierarchy in Addressing

Classful Addressing:



Class	Prefixes	First byte
A	$n = 8$ bits	0 to 127
B	$n = 16$ bits	128 to 191
C	$n = 24$ bits	192 to 223
D	Not applicable	224 to 239
E	Not applicable	240 to 255

Figure 12.3: Classful Addressing:

Address Depletion: Address depletion is the cause of classful addressing's obsolescence. Consider class A in order to comprehend the issue. Just 128 organisations may be given this class in the whole globe, but they must each have a network with 16,777,216 nodes. Class B addresses were created for medium-sized businesses, however many of these addresses were never utilised. Class C addresses suffer from a very different design issue. As each network only allows for 256 addresses, most businesses did not feel safe employing a block in this address class (Figure 12.3).

Supernetwork and Subnetwork: A class A or class B block is partitioned into multiple subnets during subnetting. The prefix length of each subnet is longer than that of the original network. A network in class A, for instance, may be partitioned into four subnets, each of which would have the prefix nsub = 10. At the same time, subnetting enables the division of addresses among many companies if all of the addresses in a network are not in use.

Supernetting was developed to combine numerous class C blocks into a bigger block, while subnetting was developed to split a large block into smaller ones beyond the 256 addresses that may be found in a class C block. This concept also failed because it made packet routing more challenging.

Benefits of Classy Addressing: With an address, we can quickly determine the address' class and, as the prefix length for each class is constant, we can also quickly get the prefix length. In other words, no further information is required to extract the prefix and the suffix in classful addressing since the prefix length is intrinsic in the address.

Classless Addressing: The Internet's administrators unveiled a brand-new architecture in 1996 under the name classless addressing. Variable-length blocks that do not belong to any classes are utilised in classless addressing. A block of addresses may consist of 1, 2, 4, 128 addresses, and so on. The whole address space is partitioned into chunks of varying lengths with classless addressing. An address's prefix designates the block (network); its suffix designates the node (device). Theoretically, we are capable of having a block of 232 addresses (Figure 12.4).

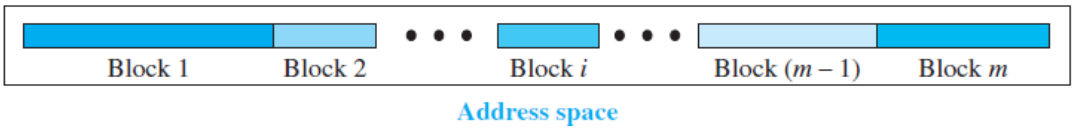


Figure 12.4: Variable-length blocks in classless addressing.

Classless addressing uses a flexible prefix length in contrast to classful addressing. Prefix lengths might be anything between 0 and 32. The prefix length has an inverse relationship with network size. A smaller network is indicated by a tiny prefix, and vice versa for big prefixes. Classful addressing may be simply adapted from classless addressing. A class A address may be compared to a classless address with a prefix length of 8. Consider an address in class B as a classless address with the prefix 16, and so on. Otherwise said, classless addressing is a specific case of classful addressing.

Prefix Length: Slash Notation: In this instance, the address is added after the prefix length, n, which is denoted by a slash. Slash notation is the colloquial name for the notation, while classless interdomain routing, or CIDR (pronounced cider) method, is the official name (Figure 12.5).

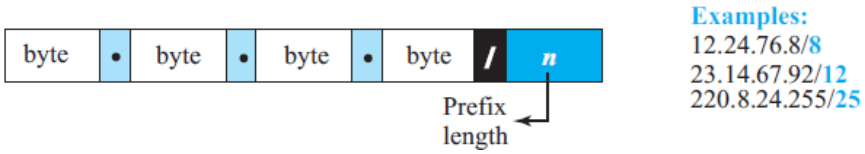


Figure 12.5: Slash Notations (CIRD)

Information Extraction from an Address

The number of addresses, the first address in the block, and the final address are the three pieces of information that describe the block to which the address belongs.

1. N = 232n is used to calculate the number of addresses in the block.

2. We maintain the n leftmost bits and set the (32 n) rightmost bits to all 0s in order to get the initial address.

3. We maintain the n leftmost bits and set the (32 n) rightmost bits to all 1s in order to determine the final address.

Subnetting: When given a set of addresses, a company (or ISP) may split the set into several subranges and allocate each subrange to a different subnetwork (or subnet). Notice that the organisation is free to add further levels at any time. Several sub-subnetworks may be created from a single subnetwork. One or more sub-subnetworks may be created inside a sub-subnetwork, and so on.

Creating Subnets: We suppose that the organisation has been awarded a total of N addresses, that the prefix length is n, that each subnetwork has been allocated a total of Nsub addresses, and that the prefix length for each subnetwork is nsub. To ensure the effective functioning of the subnetworks, carefully follow the instructions that come next.

1. Each subnetwork should include a power of two addresses.
2. For each subnetwork, the prefix length should be calculated using the following formula:

3. first address = (prefix in decimal) × 2^{32-n} = (prefix in decimal) × N.

$$n_{sub} = 32 - \log_2 N_{sub}$$

Each subnetwork's beginning address must be divisible by the subnetwork's total number of addresses. If we first allocate addresses to bigger subnetworks. Address Translation for Networks (NAT): Network Address Translation is a system that can enable virtual private networks and offer a mapping between private and universal addresses (NAT). The system enables a site to employ a set of private addresses for internal communication and a set of global Internet addresses (at least one) for external communication (Figure 12.6).

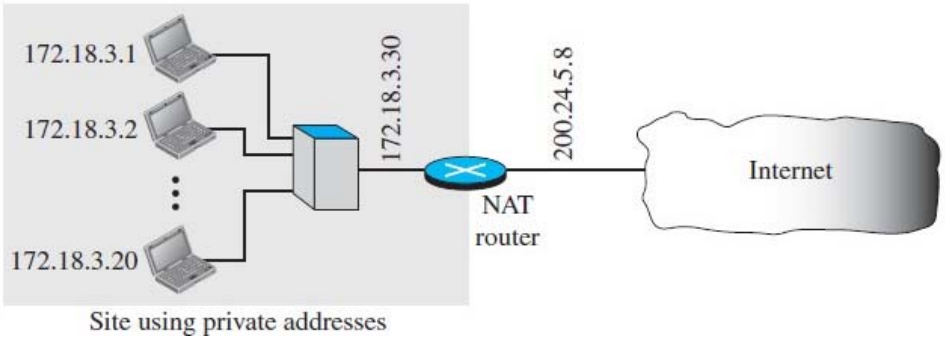


Figure 12.6: Address Translation for Networks.

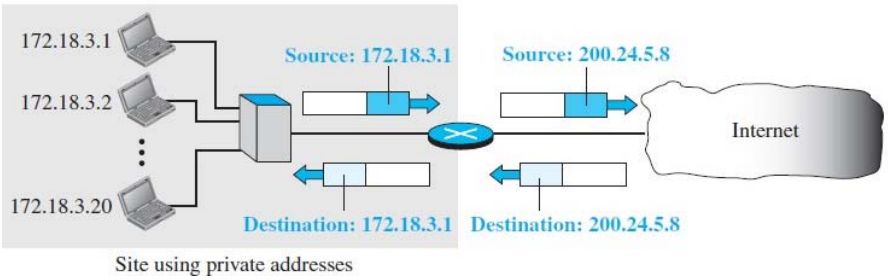


Figure 12.7: Address translation.

IPv6 Addressing: The limited capacity of the IPv4 address space is a major factor in the shift from IPv4 to IPv6. The length of an IPv6 address is 128 bits, or 16 bytes (octets), which is four times that of an IPv4 address (Figure 12.7).

Mapping of Addresses: A local address is one that is tangible. Since it is often (but not always) implemented in hardware, it is known as a physical address. The 48-bit MAC address used in the Ethernet protocol, which is imprinted on the NIC placed in the host or router, is an illustration of a physical address. There are two separate identifiers: the logical address and the physical address.

Logical Address to Physical Address Mapping: ARP

A packet has to be sent from the system on the left (A) to the system (B) with the IP address 141.23.56.23. System A does not know the recipient's physical address, thus it must transfer the packet to its data connection layer for real delivery. By instructing the ARP protocol to

issue a broadcast ARP request packet to inquire about the physical address of a system with an IP address of 141.23.56.23, it makes advantage of the services of ARP.

Every system on the physical network receives this packet, but only system B will respond, as illustrated in Figure 12.8. The physical address of System B is included in the ARP reply packet that it transmits. System A may now use the physical address it was given to transmit all of the packets it has for this destination.

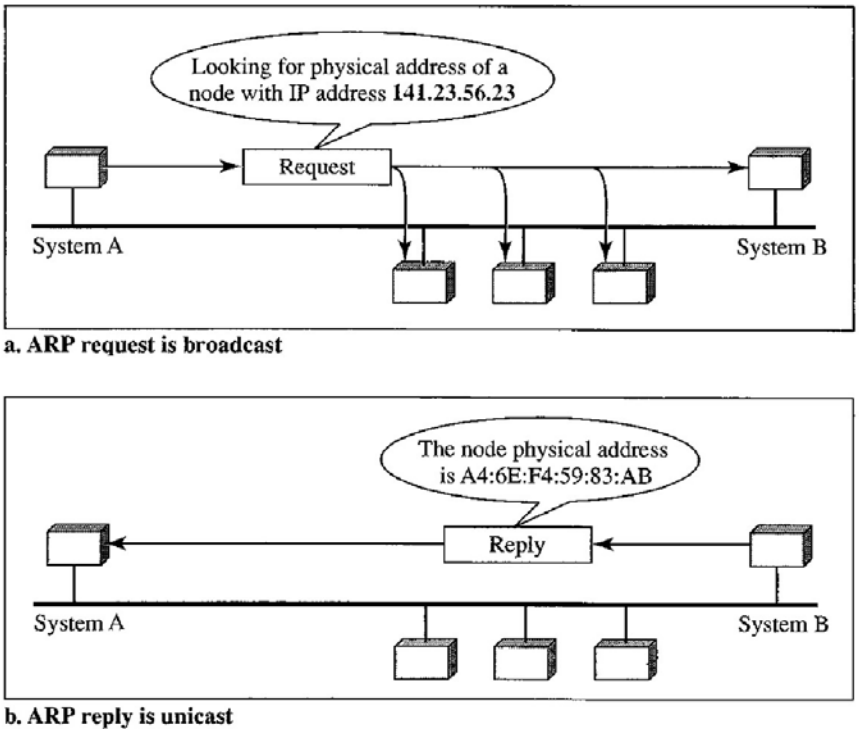


Figure 12.8: Representing the ARP operation.

The following are four instances in which ARP may be used in each of the four categories:

1. The sender wishes to transmit a packet to another host on the same network and is a host. The destination IP address in the datagram header is the logical address in this situation that has to be translated into a physical address.
2. A packet is being sent from a host to another host on a different network. The host in this scenario searches its routing database for the IP address of the next hop (router) for this destination. If it lacks a routing table, it searches for the default router's IP address. The logical address that has to be translated to a physical address is the router's IP address.
3. A router acting as the transmitter of a datagram received from a host on another network. It looks for the IP address of the subsequent router in its routing database. The logical address that has to be translated to a physical address is the IP address of the following router.
4. A router acting as the transmitter of a datagram that was sent to a host on the same network. The logical address that has to be translated to a physical address is the datagram's destination IP address. ARP responses may either be broadcast or unicast.

RARP, BOOTP, and DHCP: Physical to Logical Address Mapping

There are times when a host needs to know its logical address even while it knows its physical address. Two scenarios might result in this:

1. A diskless computer has just started up. The station does not know its IP address, but it may determine its physical address by looking at its interface.

2. An organisation needs to issue IP addresses on demand since it does not have enough IP addresses to assign one to each station. The station might request a brief lease and submit its actual address.

For a computer that only knows its physical address, the RARP Reverse Address Resolution Protocol (RARP) determines the logical address. The device may learn its physical address, which is specific to the area (by reading its NIC, for instance). The RARP protocol may then be used to get the logical address using the physical address. On the local network, a RARP request is generated and broadcast. A computer on the local network that is aware of every IP address will react to the RARP request with a RARP reply. A RARP client programme must be running on the asking system, and a RARP server application must be running on the replying machine.

There is a significant issue with RARP: The data connection layer is where broadcasting takes place. The physical broadcast address, which in the case of Ethernet is all 1s, does not cross network borders. As a result, an administrator who manages several networks or subnets must designate a RARP server for each network or subnet. This is the main cause of RARP's impending obsolescence. DHCP and BOOTP are taking the place of RARP.

BOOTP: A client/server protocol called the Bootstrap Protocol (BOOTP) was created to enable physical address to logical address mapping. An application layer protocol is BOOTP. The server and client may be on the same network or on separate networks, depending on the administrator (Figure 12.9).

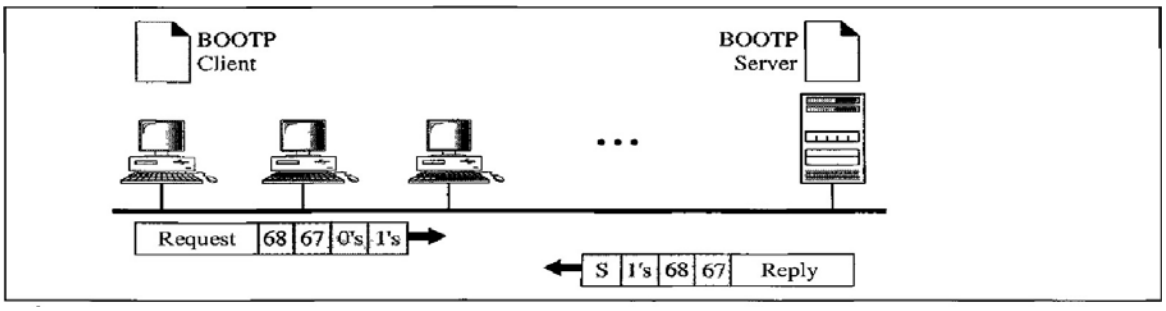


Figure 12.9: Client server on the same network.

The fact that the client and server are application-layer processes is one of the benefits of BOOTP over RARP. Similar to other application-layer activities, a client and server might be in distinct networks that are connected by a number of additional networks. However there is one issue that has to be resolved. Since the client is unaware of the server's IP address, the BOOTP request is broadcast. Any router cannot pass a broadcast IP datagram. A mediator is required to address the issue.

It is possible to utilise one of the hosts as a relay (or a router that may be set up to work at the application layer). In this scenario, the host is known as a relay agent. The unicast address of a BOOTP server is known to the relay agent. It delivers the request to the BOOTP server after enclosing the message in a unicast datagram when it gets this kind of packet. Every router may transport the packet with a unicast destination address until it reaches the BOOTP server. When the IP address of the relay agent is included in one of the fields in the request message, the

BOOTP server is aware that the message originated from a relay agent. After receiving the response, the relay agent transfers it to the BOOTP client. A dynamic configuration protocol is not DHCP BOOTP. The BOOTP server examines a table to find a match between a client's physical address and its IP address when a client requests its IP address. This suggests that there is already a binding between the client's IP address and physical address. There is a preset bound. So what happens if a host switches between physical networks? What happens when a host requests a temporary IP address? Since the binding between the physical and IP addresses is static and fixed in a table until altered by the administrator, BOOTP is unable to handle these circumstances. A static configuration protocol is BOOTP.

Static and dynamic address allocation that is manual or automated is made possible via the Dynamic Host Configuration Protocol (DHCP). Static and dynamic addresses may be assigned via DHCP, either manually or automatically. Static Address Assignment In this sense, DHCP functions similarly to BOOTP. Since it is backward compatible with BOOTP, a host running the BOOTP client may ask a DHCP server for a static address. Physical addresses and IP addresses are statically bound in a database on a DHCP server.

Dynamic Address Allocation: A pool of IP addresses is accessible in a second database that DHCP uses. DHCP is dynamic thanks to the second database. The DHCP server selects an IP address for a negotiated amount of time when a DHCP client requests a temporary IP address from the pool of available (unused) IP addresses.

CHAPTER 13

SECURITY AND PRIVACY IN ONLINE SOCIAL NETWORKS

Geetha G, Director,
School of Computer Science and Engineering, Jain (Deemed to be University) Bangalore, Karnataka, India
Email Id- solomon.j@jainuniversity.ac.in

A social network is a social mesh or structure made up of "nodes," which are people or groups. A particular sort of interdependency, such as friendship, kinship, a shared interest, likes and dislikes, common connections, shared opinions, shared knowledge, or prestige, is then used to link the nodes. Social networking as an idea is not new. For decades, sociologists and psychologists have studied and dealt with social networks. In reality, social networks have existed since the dawn of time. For a variety of reasons, including security, food availability, and social welfare, prehistoric man established social networks.

Online social networks (OSNs) are social networks with underlying digital or analogue communication infrastructure linkages that allow for the linking of network node interdependencies. The two categories of online social networks: The conventional OSNs, including Facebook and MySpace. Even without the capacity to handle mobile content, many of these may be viewed through mobile devices. The Mobile OSNs (mOSNs), which are more recent OSNs that can be accessed via mobile devices and are capable of coping with the new mobile environment. The OSNs' interdependence of nodes supports the use of social network services by individuals who are nodes in the network. The interdependencies that exist between the individuals using the network services determine the kind of OSNs.

Online Social Networks by Types

With the advent of digital communication, OSNs have expanded and undergone several sorts of evolution. Let's examine the most common categories utilising a historical timeline. The digital conversing based in a chat room gave rise to the chat network. The chat room was and is a virtual space where individuals "congregate" just for conversation. Most chat rooms have open access rules, so anybody interested in conversing or just reading other people's discussions is welcome to join.

During the conversations, users may "in" and "leave" whenever they want. There may be multiple public chat threads active at once. On their communication device, each chat participant in the room is provided a tiny window where they may type a few lines of text that will contribute to one or more of the threads of conversation. As this conversation takes place in real time, everyone's contributions to the chat room are visible to everyone else there.

A participant in a chat room may also invite another person who is presently in a public chat room into a private chat room where the two can continue their conversation with some degree of "privacy." You must establish a user name in order to join the chat room, and other users in the chat room will recognise you by it. Regular chatters will often get to know one another based on user names. Users may construct and upload profiles on certain chat room software so that other users can learn more about you through your profile. While chat rooms are open to everyone and public by nature, some are inspected for particular compliance depending on characteristics like the subjects being discussed. With the introduction of more graphical internet services, chat rooms are losing favour, particularly among young people. The blog

network is an additional online social network. Blogs are just online diaries maintained by individuals. Blog enthusiasts record their everyday activities in diaries. These diaries may sometimes focus on a single subject of interest to the blogger or be a collection of haphazard event records from a single activity. Some blogs provide comments on certain subjects. Depending on the topics, some bloggers have a loyal following.

The Instant Messaging Network (IMN) enables two-person or multi-person real-time communication. Each user in the IMN must have a user name, much like in chat rooms. One has to know a person's login or screen name in order to IM them. A tiny window is given for the IM sender to enter their message, and a similar window is supplied for the receiver to respond. The exchange's transcript is continuously moving up the screens of both individuals. These brief messaging exchanges, however, are private in contrast to the chat room. Several IMNs let members maintain profiles of themselves, similar to Chat Networks.

Online social networks (OSNs) are a mashup of all the network kinds we've covered so far, as well as additional very sophisticated online capabilities and cutting-edge aesthetics. These social networks include Facebook, Twitter, Myspace, Friendster, YouTube, Flickr, and LinkedIn, to name just a few. Many of the traits of these networks resemble those of the networks we have already explored since they are offshoots of the ones we have already seen. For instance, individuals on these networks may create profiles that they then upload to their network accounts along with their images and other elements. A login or screen name is required. Moreover, if real-time communication is necessary, it may be done through chat or instant messaging. These networks provide users delayed and archival functions in addition to real-time functionality so they may save and search for information, network managers have battled with the challenges of user security and privacy as a result of these increased archiving and search capabilities. Profiles may be changed to a private setting to prevent unauthorised users' access to private information and keep user data secure.

OSN Service Types

Because of the increasing popularity of the services that these networks are providing to their users, OSNs have been gaining popularity with the expansion of the Internet. Among these services, a few of the longest-lasting and most well-liked are:

Creating and gaining access to user profiles: In general, a profile is a view of an item that is delineated. An informal biography or a sketch of a person's life and personality is exactly what a personal profile is. User profiles have been essential to social networks ever since they first started to take off. There are several ways to publish and access user profiles on social networks.

Search in the social graph: Social networks' search functions enable users to tag user profiles and other user-provided information so that search engines may sift through the social graph in the network to gather metadata and linkages to other profiles.

Updates: Users may regularly update their profiles and contribute new data to the social network by using the update function. Users can use this to monitor the status of other users. An online social network's service offerings aid in the formation of "tribes" of users who have same interests. The following areas of interest lead to the formation of significant tribes: Social, Business, Religion, Ethnicity, and Profession are all factors.

As long as there are members to form it, a social network may have as many tribes as it wants. And although certain social networks are more well-known to particular tribes than others, tribes are not limited to particular social networks. Entities may have one or more connections

with one another within a tribe. The closer a couple develops, the stronger the corroboration between them, and the more private information and resources they exchange, the more relationships they have and more often these links are maintained. Tribes that work well together become cohesive. A strong feeling of belonging begins to emerge among the pairings and within the tribes as interactions and partnerships between groups of entities and within them increase. These sentiments of community and belonging among groups and tribes may cause individuals to be more committed to one another or the group, which may modify their behaviour as they get more used to being around other tribe members.

Privacy and security

Four rights make up the human value of privacy. These rights include the ability to be alone without interruption, anonymity, the right to conceal one's identity from the public, intimacy, the right to privacy, and reserve, the ability to manage one's personal information, including how it is shared. These four rights are highly valued by us as humans. These rights really become a part of our moral and ethical frameworks. Since the value of information has increased since the invention of the Internet, privacy has become even more valuable. The importance of privacy arises from the protection of the person's autonomy and sense of self.

Humans need a sense of control over their lives, which is why autonomy is crucial. The more a person's autonomy, particularly when making decisions, the less personal information others may know about them. Nonetheless, a person's autonomy may be questioned by others based on the volume, quality, and worth of the information they know about them. Individuals often prefer to form alliances and partnerships with people and organisations that respect their personal autonomy, particularly when making decisions.

It is crucial for people to protect their personal identification as information becomes increasingly significant and valuable. Having a unique identity is vital. Regrettably, it has been more and harder to secure one's identity since technology, particularly computer technology, has advanced so quickly.

OSN Privacy Concerns: Everywhere, even online social network groups, privacy may be compromised by infiltration, information abuse, information interception, and information matching. Online communities define intrusion as the unauthorised access, seizure, or acquisition of information or data belonging to other users of the social network. Information misuse is all too simple. We unavoidably divulge our personal information when online to anybody who requests it in order to get services. When it is allowed and being utilised for a justifiable purpose, gathering personal information is not wrong. Yet, it often happens that the information gathered from online community members is not utilised for the stated purpose. It is often utilised for inappropriate reasons, which constitutes a privacy violation.

There will certainly be fierce rivalry for personal information gathered online for commercial reasons as online commercial activity grows. Internet service providers may look for new clients by purchasing consumer information legally or illegally via eavesdropping, infiltration, and surveillance. Companies that manage these online communities must discover methods to improve the security of personal data online in order to combat this. The privacy and security of users when online, as well as the security of users' data while offline, have become major concerns as the quantity and membership of online social networks have surged. The challenges with online social networking are evident in the large and continuing numbers of young people, in particular, who pay little to no attention to privacy issues for either their own or others. There is news about and rising concern about privacy violations brought on by social networking platforms every day. Many people are increasingly concerned that internet service providers are abusing their personal data. These privacy concerns may all be summed up as follows:

Disclosure of private data to all OSN users:

Percentage of Network users share much too much personal information without thinking about who may misuse it. Teens' information on social networks is often used by sexual predators. Several OSNs are now collaborating with law enforcement to attempt to stop similar events. 9 In online, details like street address, phone number, and instant messaging aliases are often made public to an unidentified populace.

Accessibility to OSNs. Right now, anybody may easily create an account on any of these networks without having to provide any special identity. This may result in impersonation or identity theft. 10 ° a danger to privacy arises when too much personal data is given to big businesses or the government, enabling a profile of a person's behaviour to be created and used to make bad judgements. 11° Adding recent activity to profiles offers a serious concern, for instance, updating your profile to let others know where you are may help protect your privacy and security in online social networks. The OSNs' lack of specific guidelines about who should utilise which data.

Disclosure of private information to other parties: On many of these networks, information that a user has changed or deleted may actually be kept and/or disclosed to outside parties. Interconnections inside OSNs. The highly interconnected world of social networking sites may undermine user privacy in three different ways, according to Monica Chew, Dirk Balfanz, and Ben Laurie of Google, Inc. in their study titled "(Under) mining Privacy in Social Networks".

Inability to manage activity streams: The authors define an activity stream as a set of actions taken by a single user, such as editing one's personal page, adding or using a specific application on a social networking site, sharing news, or corresponding with friends. A user's privacy may be compromised by activity streams in two ways: A user may not be aware of all the events that are fed into such streams, in which case they have no control over them.

It's possible for a user to be unaware of the viewers of their activity streams, in which case they have no control over who could see the broadcast. Unwelcome linkage: When connections on the Internet expose details about a person that they had not meant to, this is known as unwelcome linking. Whenever hyperlink graphs on the World Wide Web are automatically generated to mimic relationships between individuals in the real world, unwanted linking may happen. With OSNs, it's crucial to maintain the division of personal activities and personas.

User deanonymization via the fusion of social networks: OSN sites have a tendency to collect a lot of information from users that might be used to identify them, such as their address and date of birth. Although if the data is only partly obscured in each OSN, it is still feasible to de-anonymize people using this information by comparing it across social networking sites. Unabated expansion in online social networks is adding a new player to the mix that is complicating issues that already exist. Mobile technologies, like cell phones, are the newcomer and are becoming more popular. In addition to being compact and portable, these new gadgets are becoming smarter all the time, offering extra features like voice communication, music and video playback, WiFi Internet access, and their own private communication networks. Not surprisingly, mobile devices are increasingly used to access OSNs in an ever-increasing range of 168 Computer Network Security and Cyber Ethics ways. Together with the privacy concerns highlighted above in relation to conventional OSNs, additional concerns brought on by these new technologies include:

The user's presence. The majority of mobile OSN (mOSN) now enable users to announce their presence through a "check-in" system, where a user identifies their position at a certain time. This is in contrast to the most conventional OSNs, where users were not automatically made

aware of the presence of their friends. The indicator of presence, according to Krishnamurthy and Wills16, enables their friends to anticipate prompt responses and may result in meeting new individuals who are also members of the same mOSN. While the ability to automatically find oneself is gaining popularity, it facilitates the leaking of personal information along two tracks: the potential sender and the potential recipient. Location in space. The mobile environment is characterised by the prevalence of this trait. Users should be informed that sharing their location with friends, friends who are online right now on this mOSN, friends in other mOSNs, and others may result in the disclosure of personal information to third parties.

Possibility of interaction between regular OSNs and mOSNs. Such links, in the opinion of Krishnamurthy and Wills, are advantageous to users who engage with mOSNs with the expectation that some of their activities will be visible to their friends on conventional OSNs. However both the conventional OSNs and the mOSNs may leak a lot of their private data to unauthorised users.

The user behaviours that automatically create events for respective activity streams should be made clear in both OSN and mOSN apps.

Events should be added to users' activity streams at their discretion, and users should be able to take them out of the streams once they have been added by an application. Users should be able to choose who will see their activity streams and should be aware of who that audience is. Activity stream events should be produced by OSN and mOSN apps that are in line with user expectations.

Additional ideas that might be helpful in this endeavour include:

Creating strong passwords

User knowledge of the terms of service and privacy policies for their OSNs and mOSNs.

In order to offer users with some privacy protection when using their networks, OSN and mOSN providers should create regulations and uphold current legislation.

CHAPTER 14

SECURITY IN MOBILE SYSTEMS

Krishnan Batri, Deputy Director,
School of Computer Science and Engineering, Jain (Deemed to be University) Bangalore, Karnataka, India
Email Id- krishnan.batri@jainuniversity.ac.in

Mobile telephones, which are broadly construed here to include devices based on Code Division Multiple Access (CDMA), Time Division Multiple Access (TDMA), Global System for Mobile Communications (GSM), and Wireless Personal Digital Assistants (WPDA) digital technologies and follow-ons, are among the devices that make up a mobile communication system. These devices run specially developed software to sustain a wireless communication link between them for a period of time. The world is being revolutionised by mobile communication technologies, which are reducing it to two or more tiny portable mobile devices. Rapid advancements in communication technology, ground-breaking developments in software, and the development of vast, robust communication network technologies have all made communication easier and more accessible over wide swathes of the world. An increasing number of underwater cables and less expensive satellite technologies are providing Internet connectivity to almost all of the world's rural poor quicker than many had predicted. These developments are being driven by the intense rivalry between mobile communications companies.

One must first comprehend the function operating systems play in the ecology and architecture of mobile systems in order to properly comprehend how they operate. An operating system created expressly to operate on mobile devices including mobile phones, smartphones, PDAs, tablet computers, and other portable devices is known as the mobile operating system, or simply mOS. An ecosystem of additional programmes, referred to as application programmes, may operate on mobile devices on top of the mobile operating system. The mOS operates similarly to its larger brother, which powers laptops and Desktops. Nevertheless, there are disparities in how much memory a typical and contemporary operating system would use to carry out those tasks. In the case of mOS, we're referring about tiny sizes across the board. Modern mOSs must combine the necessary components of a personal computer with touchscreen, cellular, Bluetooth, WiFi, GPS navigation, cameras, video cameras, speech recognition, voice recorders, music players, near field communication, personal digital assistants (PDA), and numerous other still-in-development features in addition to operating in a limited environment.

Operating systems for mobile devices are just as essential and vital to their functionality and security as they are in larger, less portable devices like PCs and laptops. The mobile device's operating system determines how safe it is in terms of security-related concerns. Thus, every mobile device incorporates as much security as its operating systems can handle without compromising speed, usability, or other features demanded by customers. The majority of mobile operating systems share many characteristics with their more mature counterparts, the operating systems found in PCs and laptops, which have had and still experience escalating security issues including backdoors, spyware, worms, Trojan horses, and others. The best method to safeguard these devices with mOS is to prepare for potential attacks rather than waiting for them to happen, like we did with laptops and PCs. This kind of prompt preventative procedures might definitely help secure the mobile device much more quickly.

The most popular mobile operating systems (mOSs) as of the writing are Android, Symbian, iOS, BlackBerry OS, Bada, and Windows Phone. Of course, they are many others. Android: The Open Handset Alliance, made up of major hardware and software manufacturers including Intel, HTC, ARM, Samsung, Motorola, and eBay, as well as a tiny start-up firm called Google, initially created Android, a Linux-based operating system, with their support. As of this writing, Android supports the following networking technologies: GSM/EDGE, IDEN, CDMA, EV-DO, UMTS, Bluetooth, Wi-Fi, LTE, NFC, and WiMAX. Multitasking. "Zoom-to-fill" screen compatibility mode.

Multiple touch input is supported, there is a notification bar, and the home screen and keyboard may both be customised.

IOS: Originally created for the iPhone, Apple's mobile operating system, or iOS, has subsequently been expanded to accommodate more Apple products including the iPod touch, iPad, and Apple TV. Installing iOS on hardware from other parties is not authorised. Swipe, tap, pinch, and reverse pinch are just a few of the gestures that may be used to interact with the OS; each has a particular meaning in relation to the iOS operating system and its multi-touch user interface. At the time of writing, the main features of iOS are: —Multitasking —A dock at the bottom of the screen where users may pin their most-used applications.

Center for Notifications:

iMessage (which functions much like a chat service but is only available between users of the iPod touch, iPhone, and iPad), Location-based reminders (get an alert as soon as you enter a certain place or region).

Microsoft's mobile operating system, Windows Phone, has received a significant software upgrade called Windows Phone 7.5. At the time of writing, the main characteristics of Windows Phone OS are: Multitasking

Support for Facebook Locations check-ins, Windows Live Messenger and Facebook Chat integration, an all-in-one thread view that combines SMS, MMS, IMs, and Facebook Chat, and support for threaded email discussions

The Korean term "bada" means both "ocean" and "seashore." The Samsung S8500 Wave was the first device to run the Bada operating system, which was initially shown at Mobile World Congress 2010 in Barcelona in February 2010.

The operating system offers multitasking and supports specific input devices, such the trackwheel, trackball, trackpad, and touchscreen that RIM has embraced for use in its portable devices. The BlackBerry platform is perhaps best known for its native support for business email via MIDP 1.0 and, more recently, a subset of MIDP 2.0. When used with BlackBerry Enterprise Server, these protocols enable full wireless activation and synchronisation with Microsoft Exchange, Lotus Domino, or Novell GroupWise email, calendar, tasks, notes, and contacts. The following are the main characteristics of the BlackBerry OS as of this writing:

Multitasking

NFC (near field communication)

Support for HTML5 and Ajax: Notifications preview on the home screen, Multi-touch capability (for touch screens), Geotagging, Liquid Graphics technology, which OS 7 uses to provide higher quality displays, smoother visuals, and a more responsive touchscreen.

Integrated Facebook Application with BlackBerry Messenger. The online surfing experience in OS 7 is 40% quicker than in OS 6, and 100% faster than in OS, more phones and smartphones worldwide than any other mobile OS use Symbian Symbian mOS. The longevity, broad adoption, and advanced state of Symbian as an operating system are its advantages. In its most recent edition, Symbian has put more of a focus on expanded e-mail capability, improved tools for third-party developers, and more security features.

Protection in Mobile Ecosystems

Using mobile devices is becoming riskier as they grow more commonplace. They are keeping and storing more and more sensitive information, both personal and commercial, while also using public networks with weak security and unreliable cryptographic algorithms. In reality, the security risks posed to these devices are comparable to and perhaps greater than those that PCs and laptops faced during their heyday.

Due to their ability to run continuously without human intervention and constant network connectivity, mobile devices face security risks that are at least as great as those posed by servers. Also, there is a greater zone of assault plagued by geographical, legal, and moral variations since these gadgets have the capability to roam on several networks. Service providers are eager to consolidate networks and standardise communication protocols in response to the high demand for global connectivity, particularly in developing nations. This will make it simpler for these devices to roam in expansive areas and networks, which will provide fertile ground for attackers. These smart mobile device penetration trends are not only occurring in remote rural areas; rather, what is more concerning is their quick infiltration into business IT environments, where device security is of utmost importance. The popularity of smart gadgets, which are steadily replacing business laptops as the primary business mobile device, has led to their expansion into company IT environments. Enterprise management is thus beginning to concentrate more on security-related concerns. While most high level operating systems have security best practises in place and anti-virus client apps accessible, this is not the case with tiny mobile devices. According to Byron Acohido's paper, "New Security Flaws Detected in Mobile Devices," Cryptography Research has just conducted two tests. Cryptography Research shown how it is feasible to eavesdrop on any smartphone or tablet when it is being used to make a purchase, perform online banking, or access a company's virtual private network in one research. Moreover, McAfee, an anti-virus software provider and a subsidiary of Intel, demonstrated how to remotely hack into Apple iOS, acquire secret keys and passwords, and collect private information including call logs, emails, and text messages. The alleged fact that the device being attacked would not in any way indicate that an assault is underway is more concerning. Almost all mobile system users, security professionals, and law enforcement authorities anticipate that as consumers and businesses start to depend more heavily on mobile devices for shopping, banking, and employment, cybergangs will step up their assaults. Thus, there is a pressing need for the community to adopt a wider range of security-related behaviours in order to help provide the maximum degree of safety for all users.

In its "2011 Mobile Threat Report," the smartphone security firm Lookout Mobile Security addresses security risks to mobile devices in four key areas: application, web-based access, network, and physical surroundings. Mobile devices deal with serious risks on a regular basis.

Risks Linked to Applications: The most alluring aspect of every mobile device is its capacity to run countless programmes (apps), which may be used to carry out a wide range of functions. These programmes were created by unidentified authors who had little to no loyalty to anybody and did not follow orders from them. Are the most serious security risks for every mobile device that can download software come from downloaded applications? Application-based

threats in downloadable applications include malware, spyware, functionality features, and vulnerable applications, which are all software that may have security flaws. Malware is software that is intended to perform malicious actions on a device, spyware is software that is intended to collect or use data without the user's knowledge or consent, and vulnerable applications are software that may have security flaws.

Online Threats

Once turned on, mobile devices continually roam on open networks with few security and cryptographic mechanisms to keep them safe. They often maintain a continual connection to the Internet in order to use standard web-based services. In such situations, they are exposed to a number of web-based threats, such as phishing scams, where intruders use web-based services to launch attacks on those devices connected to the web to obtain information such as usernames, passwords, credit card details, and other private data of the device owner; drive-by downloads, which are popups created by scammers to automatically download malware onto a user's computer; and other threats.

Network dangers: As previously said, as soon as a mobile device is turned on, it begins hunting for networks to connect to, whether it be cellular networks or the internet. They are vulnerable to network exploits once linked.

Physical Dangers: Physical dangers depend on the size and environment of the mobile device owner, as opposed to hazards depending on the nature and capability of the mobile device. Due mostly to the miniaturization of mobile devices, such dangers include lost or stolen gadgets.

Risks Specific to Operating Systems

A mobile device's operating system determines how secure it is. It's important to remember that the majority of operating system risks are brand-specific. Let's concentrate on several dangers that are known to be operating system-based:

K Data Atruct: This is a vulnerability in the Windows Mobile (WM) operating system bug. Microsoft combined all essential system operations into a single coredll.dll file to eliminate the need for programmers to incorporate the necessary code.

All the APIs it uses are simply called at their coredll addresses in the allotted memory. In doing so, an address to the list of modules is given, allowing for the identification of the coredll's address. From here, one may do a memory search to get the virtual address. What the API desired. This might make the gadget vulnerable to abuse. The WinCE malware takes use of this weakness.

Pocket IE, the default web browser for WM OSs and yet another Windows flaw discovered in the diminutive Internet Explorer. All of the flaws in the regular IE for desktop computers and laptops are present in the PIE. Check out the "General Mobile Devices Attack Types" section below to see all of these flaws.

Jailbreaking: This is a method by which a user may change the phone's operating system to get root access and enable programmes that haven't been officially approved by Apple's review guidelines. For instance, the non-malicious website JailbreakMe 3.0 for iOS devices uses two flaws to jailbreak a smartphone.

DroidDream is an Android malware that uses the Exploid and RageAgainstTheCage vulnerabilities to bypass the operating system's security sandbox, take root access, and automatically install apps.

Update Attacks: Using programme updates as an attack strategy in the Android Market is a developing issue. A virus author initially makes a genuine programme available that is malware-free. The virus author upgrades the programme with a harmful version once they have a sizable user base.

Malvertising is when an attacker uses deceptive advertising to trick consumers into installing malware, particularly on the Android Market. They depend on the fact that customers are used to installing applications via adverts since developers often employ in-app advertisements to increase user numbers. Additional dangers come from the IoS's flowing shell concept, root account, static addressing, static systems, and repetition of code.

Forms of Mobile Device Attacks: In addition to the attacks against individual operating systems mentioned above, there have also been significant generic mobile system assaults on certain mobile devices, operating systems, or apps. Here are a few of them, largely holdovers from the laptop and PC era:

Denial-of-service (DDoS): This method aims to disrupt the system such that the device, service, or network it uses to function cannot perform the task at hand that involves the device.

Phone theft: By accessing a mobile phone's voicemail or text messages without the owner's knowledge or permission, this approach may be used to intercept phone calls or voicemail messages. You may remember the UK's News of The World phone hacking incidents.

Mobile Malware/Virus: Software that specifically targets mobile phones or PDAs with wireless capabilities is known as mobile malware or a mobile virus.

Spyware: Spyware is a sort of malware that installs itself automatically or, in some situations, is actively placed on computers so that it may secretly and continually gather data about a number of events, users, or applications.

Exploit: An exploit is a piece of software code that takes advantage of a flaw, error, or vulnerability in order to affect electrical devices, computer software, or hardware in an undesired or unexpected way.

Everything Blue: Below are a few examples of spyware and malware that makes use of Bluetooth technology. Here are a few of them:

Bluejacking, which is similar to spamming in that the offender sends the victim's device unwanted messages, allows for communication between the linked devices. As a result, the attacker may be able to access the victim's device.

Bluesnarfing is a kind of Bluetooth hacking that enables an attacker to access the contact list, text messages, emails, and other crucial data on the victim's device. Even if the device is undetectable, the hacker may conduct a brute force attack to determine the victim's MAC address.

Bluebugging a Trojan Horse-style assault in which the hacker utilises advanced attack methods to take control of the victim's mobile device. • Bluetoothing—this is social engineering, where a hacker may utilise conventional social engineering techniques to pose as the genuine user of the mobile device. Once in control, the attacker can do anything with the device.

BlueBumping: Two mobile devices are used in this assault. In order to target other services, the attacking device first convinces the victim to accept a connection for a little data exchange, such exchanging a photograph. The attacker asks for a link key regeneration while the

connection is still active and uses it afterwards to get access to the victim's device, giving it complete control over all of the services on it.

BlueChopping: A Bluetooth piconet, or ad hoc Bluetooth network connecting other Bluetooth devices—is the target of this assault. It enables one master device to communicate with several other active slave devices, allowing for piconet disruption by impersonating a participating slave and confusing the master's internal state.

BlueDumping is the practise of causing a Bluetooth victim's mobile device to dump its stored connection key in order to sniff a Bluetooth device's key-exchange. The attacker must be aware of the BDADDR of a group of coupled devices prior to doing the sniff. The attacker connects to the other device using a fake address for one of the devices in order to get this. The attacker's device will answer to the target device's request for authentication with a "HCI Link_ Key_ Request_ Negative_ Reply" since it has no link key. In certain instances, this may force the target device to destroy its own link key and enter pairing mode. 5

BlueSmucking: A Bluetooth DoS attack that instantaneously disables certain Bluetooth-enabled devices. The original "Ping of Death" is used, although it has been modified to operate with Bluetooth.

One may confirm connection, request an echo from a different Bluetooth peer, and calculate the round-trip time on an established link using the L2CAP (echo request) layer. This is achievable with Bluetooth since the l2ping function in the BlueZ utils lets the user choose the size of the packet to be delivered to each peer. The -s num> option is used to do this. The Bluetooth equivalent of war driving is called BlueSniffing.

Phishing: Phishing on Bluetooth devices uses the same techniques as on PCs and laptops in that the attacker poses as a reliable friend in an electronic communication like an email or text message in order to obtain information such as user names, passwords, credit card numbers, and other private data of the device owner.

Smishing: SMiShing is a sort of social engineering crime similar to phishing in that it lures victims into disclosing confidential and sometimes personal information by using their mobile devices and messages as bait.

Vishing: similar to the previous two social engineering crimes, phishing is also a criminal activity. It mostly employs Voice over IP (VoIP)-enabled mobile phone capabilities to get private financial and personal information from the general public in order to reap financial rewards. 'Voice phishing' and 'voice' are combined to form the phrase.

Attacks on mobile devices are reduced: The threat of unmanaged, personal devices accessing sensitive enterprise resources and connecting to third-party services outside of the enterprise security controls is increasing due to the growing use of mobile devices and the trend of employers allowing employees to bring their own devices (BYOD) to work. This might disclose important company information to prospective attackers. Small and Medium Companies (SMBs) do not believe they have the tools to identify and manage these threats, and the security teams in these organisations are starting to feel vulnerable to mobile device security risk.

Yet, there are a number of security guidelines and best practises that might be helpful in this kind of circumstance. The minimal security standards for any mobile security management must include these three security elements. These elements include passcode policy setting, remote wiping, and hardware encryption:

Smartphone Encryption: Application and hardware encryption are the two methods for encrypting mobile devices.

Application Encryption: When mobile devices are secured via apps, encryption shields the target device from assaults on the host device as well as from end-to-end network connections. For this form of encryption, there are several vendor options.

Hardware Cryptography: Hardware encryption is an encryption system that is built into the hardware by the original maker of mobile devices, such as BlackBerry maker Research in Motion (RIM). In order to offer a robust security posture for business BlackBerrys, RIM combines strong Advanced Encryption Standard (AES) and Triple Data Encryption Standard (Triple DES) encryption with a powerful mobile device management platform. Similar embedded encryptions may be found in various mobile device manufacturers' operating systems, embedded SIM cards, or portable encryption SIM cards. These manufacturers include Apple, Google, Microsoft, and others.

Remote Mobile Wiping: The ability to remotely delete data from a lost mobile device is provided by mobile remote wiping to security IT administrators. Both the mobile device maker and a third party created the remote wipe and other management functions. Like Google's Apps Premier and School Edition, which are compatible with iPhones, Nokia E series devices, and Windows Mobile handsets, many of them are cross-platform.

Mobile Passcode Policy: The best solution to handle the expanding variety of devices running various operating systems or versions of an operating system is a security policy requiring a passcode tag for devices. A full mobile security package should contain:

A firewall to protect the system from intrusions and dangerous software. An authentication method to make sure that unauthorized users cannot access the device if it is lost or stolen; A VPN to offer customizable ways to guarantee secure connections for any wireless data traffic. On-device data encryption to prevent information theft both physically and electronically; Antivirus software to guard against viruses and malware.

CHAPTER15

SECURITY IN THE CLOUD

N Sengottaiyan, Deputy Director,
School of Computer Science and Engineering, Jain (Deemed to be University) Bangalore, Karnataka, India
Email Id- sengottaiyan.n@jainuniversity.ac.in

In contrast to todays cloud services models, traditional data centre computing models have relied heavily on a three-tier architectural design that includes access, distribution, and core switches. These models link relatively few customers and cater to a narrow range of client demands. In most situations, each server was devoted to either a single or restricted applications and had IP addresses and media access control addresses.

The static nature of the application environment was advantageous since it allowed for manual server deployment or redeployment procedures to operate properly. A spanning tree protocol was predominantly employed, according to Jim Metzler and Steve Taylor of Network World (2011), to prevent loops. Nevertheless, the staid character of the data centre has been significantly altered by recent spectacular advancements in virtualization technologies, distributed computing, quick enhancements, and access to high-speed Internet. The modern data centre, which offers cloud services, is anything from boring since it is rife with activities and offerings that set it apart from its more conventional cousin. As an example, its services are now available on demand, by the minute or the hour; elastic, meaning customers may have as much or as little of a service as they want at any given moment; and entirely controlled by the provider, meaning all a customer needs is a computer and Internet connectivity. The following discussion goes through these qualities.

Widespread Network Access: The availability of high speed internet, the adoption of virtualization technologies, and other factors have all contributed to changing the way that consumers obtain the computer services they need and expanding the range of options available to them. More options brought higher specialised and quality services that a client could choose from.

Determined Service: Since cloud services may be elastic, adaptable, and on demand, it's crucial that they be metered. Customers may acquire what they want in the necessary quantities at the time they need the service thanks to the notion of metered services. In order to give transparency for both the supplier and the customer, cloud systems include metering services that automatically manage and optimise resource consumption depending on the kind of service, such as storage, processing, bandwidth, and active user accounts.

Demand-Based Self-Service: The conventional and all other forms of media are no longer relevant due to the fast and unprecedented usage of virtualization technology, as well as the availability and access to high-speed internet.

The requirement for redundancy and outsourcing of services, as well as models of procurement of computer services that required permanent ownership of software or computing gear and lengthy contracts with workers who helped to utilise the service, all reduced and gave way to a more flexible model. The restrictive old models of ownership, outsourcing, or bundled services were no longer the only options available to users of computer services. A customer

may now choose when and how long to utilise the offered services in addition to having any computing resources and capabilities automatically provided as required.

Extreme Elasticity: Elasticity in computing services refers to the capacity to dynamically scale and resize the available virtualized resources, including as servers, processors, operating systems, and others, to satisfy the demands of the client. In order to guarantee that end users' demands are consistently and effectively handled, the provider makes sure that there are resources available that match elastic capabilities. Web service interfaces that make it easy for the client to get and configure capacity include IBM ASC and Amazon's EC2.

Pooling of resources: Higher and more diverse service expectations from clients are often brought on by increased flexibility, accessibility, and simplicity of use. Typically, service providers provide a range of system resources and services in response to these increased needs. The computing resources of the provider are pooled to serve multiple consumers using a multitenant model, as mentioned by Peter Mell and Timothy Grance in the NIST report (2011), with different physical and virtual resources being dynamically assigned and reassigned in accordance with consumer demand.

1. Beside the five mentioned, cloud computing also has additional traits. One of these is:
2. Huge scale: The cloud provides resources at a vast scale as needed, and virtualization is the foundational element of cloud computing.
3. The virtualization of the core operations of the physical computer makes the cloud conceivable.
4. Free or almost free software from the cloud, when required.
5. Autonomic computing, which refers to the ability to dynamically scale computer resources as needed.

Multi-tenancy: Cloud computing supports a huge number of concurrent users due to its enormous size and simple access to its resources.

Models for Cloud Computing Services

Technology as a Service (IaaS): Infrastructure as a Service (IaaS) is the process of giving the customer the ability and capability to manage and control system resources such as starting, stopping, accessing, and configuring the virtual servers, operating systems, applications, storage, processing, and other basic computing resources via a web-based virtual server instance API (IaaS). Nevertheless, despite performing all of this, the user has no access to or control over the underlying physical cloud infrastructure.

A collection of software and product development tools known as Platform as a Service (PaaS) are housed on the provider's infrastructure and made available to customers via a web-based virtual server instance API. In this case, the client may use the Internet to develop apps for the provider's platform. Since the client cannot manage or control the underlying physical cloud infrastructure, including network, servers, operating systems, or storage, accessing the platform via the web-based virtual instance API safeguards the resources.

Service-based software (SaaS): The price of software has always been a factor in determining how much software is purchased. Software has undergone a number of models as a consequence of efforts to manage cost. The original approach was user-created software, in which users created their own software based on their requirements. They were in charge of managing and updating everything since they owned it. The second approach, known as the conventional software model, was based on packaged software, in which the client purchased a more all-purpose programme from the vendor in exchange for a licence from the vendor.

Although the consumer was in charge of managing it, the supplier was in charge of the updates. Nonetheless, software developers may provide extra support services, or "premium support," often in exchange for a price. The third paradigm was the Open Source approach, which was promoted by the free software movement that emerged in the late 1980s. By the late 1980s, the Open Source Project had transformed free software into open source (OSI). Some for-profit "free software" began changing the strategy from totally free software to some sort of payment to fund the upgrades of the programme under the guise of the "open source" ideology. The open source software paradigm significantly reduced the cost of software. The fourth model included software outsourcing.

The outsourcing business model was developed in reaction to the rising price of software and software management. Slowly but surely, the cost of software administration began to outpace that of all other software components, such as licencing and updates. Model Four, however, maintains the perpetual licencing of software from the software provider; maintenance payments are still paid, but the software creator also manages the software.

Model five was Software as a Service (SaaS). The one-time licencing charge is dropped while using SaaS. The customer has access to all software applications from the provider through a variety of client devices using either a thin client interface, such as a web browser, a web portal, or a virtual server instance API. All software applications are held by the provider. The customer does not manage or control the underlying cloud infrastructure, including the network, servers, operating systems, storage, or even specific application capabilities, with the possible exception of a small number of user-specific application configuration settings. This is also true of previous cloud services.

Characteristics of SAAS Applications

The following characteristics of software as a service are particular:

1. Scalability: The ability to gracefully accommodate increasing workloads.
2. Multi-tenancy: One application instance may be used to serve hundreds of different businesses. This is distinct from the client-server paradigm from which the cloud computing model developed, in which each customer is provided with a server on which one instance of software is running.

The possibility for consumers to customise their applications using information

1. Models for Cloud Computing Deployment
2. There are three distinct cloud kinds, or cloud deployment methods.
3. These three models are public, private, and hybrid.

Open Clouds: Public clouds provide open Internet access to computing resources, enabling users to self-provision resources often via a web service interface on a pay-as-you-go basis. One advantage of public clouds is their ability to provide enormous pools of scalable resources on a temporary basis without requiring the customer to make an infrastructure investment.

Personal Cloud: Private clouds, in contrast to public clouds, provide customers rapid access to computing resources housed within the infrastructure and premises of an organisation.

Similar to how users pick and scale resources from a public cloud, users who normally have some kind of connection with the private cloud provider often do so using a web service interface. Also, the private cloud is set up within the company, utilises its resources, and is constantly protected by the firewall and other physical, technological, and procedural security measures. Private clouds provide a better level of protection in this situation.

Blended Cloud: Public and private clouds' computing capacities are combined in a hybrid cloud.

Computing in the Cloud and Virtualization

Virtualization in computing is the technique of producing computer resources that are virtual (i.e., created in effect and performance but not in actuality). Virtual resources in computing may be either software or hardware. In the past, operating systems have utilised software virtualization to generate a variety of virtual operating systems, including clones of themselves and even other operating systems, to operate on the underlying computer and carry out tasks at a better performance level. Virtualization has been used in hardware to build additional resources like servers, storage, and other things. Through the division of the underlying physical computing resources into numerous, equally powerful virtual machines, virtualization has the potential to significantly increase the performance of computing systems, such as hardware and software, scaling up the performance and creating elasticity of many computing systems. Scaling up or down processing and storage is simple using virtualization.

Security of Computer Networks and Cyber Ethics

In order to provide isolation and security, virtualization, which enables programmes from various clients to operate on various virtual machines, is a crucial component of cloud computing.

The advantages of cloud computing: The cloud computing paradigm is quite intriguing and offers the computer community several advantages. As you understand it, it is not only thrilling, but it also offers a variety of advantages, such as huge scale leveraging, homogeneity, virtualization, affordable software, service orientation, and cutting-edge security technologies.

Cost reductions are the primary advantage of cloud computing for businesses

Whether a manufacturing firm is small, medium-sized, or big, employing a cloud model for the majority of its computer requirements has significant financial advantages. The major problem is that, apart from a few devices required for using a web portal to access cloud services, cloud computing is run remotely outside of corporate boundaries. As a result, employees of the company can use fewer computers to complete the same amount of work, which saves money by avoiding the need to house data centers on the property, hire staff to manage the data center, and pay for other expenses that would typically be required to run a data center on the premises. Because there aren't many computers in the building, there are also power consumption reductions. In many businesses today, servers are only utilized at 15% of their potential, and 81% of corporate software costs go towards installing and maintaining software. The use of cloud apps may cut these expenditures by 50% to 90%. Updates 2 Automatic

Software must be updated often for efficiency and profitability as well as to stay up with evolving company capabilities since software powers the majority of enterprises and personal interactions. The price of managing and updating software has constantly increased, often outpacing the price of purchasing new software.

Companies must often update and modify their software in order to remain competitive and, in many circumstances, afloat. Resources used by the corporation are heavily depleted by the administration, licensing, and commercialization of software upgrades.

So, having automated upgrades and maintenance provided by the cloud provider may be a huge comfort to any firm. Nevertheless, updates are not only for software.

Without having to worry about hardware upgrades is also economical for businesses. Cloud Computing's Environmental Advantages: There has been a heated debate about the energy consumption of cloud computing, with some arguing that it is consuming resources because large cloud and social networking sites require daily megawatts of power to meet insatiable computing needs, while others argue that the computing model is actually saving power from millions of servers left idle daily and consuming more power. More on this will be covered in the parts that follow. For the time being, we believe that cloud computing does really reduce power usage.

Remote entry: Employees of the firm could be able to work when they are in the office, at home, or on the road with access to a web link to the cloud. The business will greatly profit from this since it will prevent downtime caused by employee absences.

Emergency relief: Because they have important corporate data housed on premises, many businesses live in continual dread of calamities. Nobody wants to be a victim of a major disaster like a storm, an earthquake, a fire, or a terrorist strike. Even if there is only little physical damage, such disasters may wreak havoc on a company's critical data and interrupt operations. Smaller catastrophes like computer failures and power outages may potentially seriously damage a company's critical data. While it's conceivable, many businesses, particularly small ones, may not even have a disaster recovery plan, and those that do might not be able to carry it out well. Cloud computing investments may help us get over this phobia. The clouds secure data centres may securely house a company's critical backup data instead of the server room.

Provisioning via self-service: With the help of cloud computing, users can quickly and easily deploy their own virtual sets of computing resources, such as servers, networks, and storage, as needed, without the usual delays, expertise requirements, or difficulties associated with purchasing, setting up, and managing physical resources. Regardless of where they are physically located, cloud owners can not only provide all the computing resources 190 Computer Network Security and Cyber Ethics that an organisation requires, but they also have the tools necessary to manage and respond to the infrastructure, software, and platform needs of the organisation on a daily and hourly basis.

Scalability: Cloud computing provides the greatest infrastructure, platform, and software scalability that cannot be matched in any owned computing facility since it can monitor an organization's computing demands on a minute-by-minute basis and adjust the needed resources as demand rises or falls.

Continuity and Tolerance for Errors: Cloud computing provides a high degree of dependability and fault tolerance since the cloud provider, with skilled people and expertise, monitors the computing needs of a client organisation and can readily grow to demand.

Effortless Usage: Cloud service providers need to simplify the user interface for clients to grow into the cloud with the least amount of work if they want to draw in additional consumers.

Knowledge and Competence: Professionalism and a broad range of abilities for clients are some of the most desired qualities in a cloud provider. Businesses, particularly small ones, would pay a significant amount to hire someone with the expertise, efficiency, and knowledge seen in cloud centre workers.

Reaction Time: Cloud computing services often have speed since the given computer resources are cutting-edge and potent enough to handle a big number of users, depending on the bandwidth at the firm web portal.

Mobility: Cloud computing, which enables users to access their applications from anywhere, is basically a mobile computing platform thanks to the web portal interface to the Cloud.

Additional Storage: The primary purpose of cloud computing is storage. As a result, it is affordable and easily adaptable to necessity.

Additional Advantages: Providing a high quality of service (QoS), offering a high quality, well defined, and stable industry standard API, and making computer resources available on demand depending on "at hand" cost constraints are further advantages.

Security: The separate virtual computers produced for each usage in cloud computing, which we'll talk about further in the section after this, have a built-in security feature. A strong authentication regime at the browser interface gateway, a security mechanism that is individually and quickly set up and torn down as needed, and a strong validation and verification scheme that is expensive to deploy at a single client-server model are all additional features of the cloud model in addition to these built-in provisions made possible by virtualization.

Problems with Security, Reliability, Availability, and Compliance in Cloud Computing, the current iteration of the cloud computing concept did not emerge overnight. It has taken years for the process to progress through seven different software models, starting with internal software, licenced software (also known as the traditional model), open source software, outsourcing, hybrid software, software as a service, and finally the Internet model, the last two of which are a part of the cloud computing model. One cannot fail to observe the backward compatibilities or the continuation of many of the characteristics that defined software across all the models when closely examining the cloud servicing model. Even while this delivers the advantages of each of those software models, the cloud computing model has inherited many, if not all, of the software complexity and security problems from those models. Our initial intention was to utilise these models as a lens through which to explore the security concerns with the cloud computing approach. While it is tempting, we will take a different route and keep the reader grounded in the various software models. One of the biggest problems with the cloud computing concept is and will always be security. Performance, compliance, and availability are the other three associated challenges. All four will be covered in this part, but security will be the first topic as it is the most pressing one.

To begin the conversation on cloud computing security, we'll paraphrase According to Sun Microsystems CTO Greg Papadopoulos, users of the cloud often "trust" cloud service providers with their data in the same way that they "trust" banks with their money. This implies that they anticipate the three concerns of security, availability and performance to be of little concern to them as they are with their banks. The actors and their responsibilities in the process you are interested in safeguarding, as well as the application or data in play, are two key components of any discussion of security. The application or data is considered in light of the condition it is experiencing at any given moment. For instance, the states of the data and the application may either be in motion between the remote hosts and the hypervisors and servers of the service provider, or they may be in the static state when they are stored at remote hosts, typically on the premises of the customer or in the servers of the service provider. In each of these two stages, a distinct level of security is required.

Customers and Cloud Providers:

Their responsibilities and roles: The key participants in the cloud computing paradigm are the cloud providers, customers who are data owners and employ the cloud provider's services, and users who could be the owners of the data stored in the cloud. The first two players have

assigned duties to everyone who works for them. To fully comprehend the delegated responsibilities given to each of these, it is necessary to first examine the minor security issues brought on by peripheral system access control, which always leads to the simplest security breach for any system, typically through the compromise of user accounts via weak passwords. This issue is significant and impacts both domestic and external cloud solutions. Companies that sell and use cloud solutions must create an access control regime to address this and any other administrative and security issues. Every user, whether local or distant, must be subject to these access restrictions, including ancillary ones like the creation and storage of user passwords. Because of how crucial access control management is, cloud providers must invest time and money in creating a robust access control regime.

Data and application security in the cloud: We must first concentrate on the security and function of the hypervisor before turning our attention to the servers that host user services in order to comprehend and appreciate the security of data and applications in the cloud. A hardware virtualization approach known as a hypervisor, sometimes known as a virtual machine management (VMM), enables several operating systems, referred to as guests, to operate simultaneously on a host computer. The kernel software that runs on the main physical computer that serves as the physical server is piggybacked by the hypervisor. The hypervisor controls the execution of the guest operating systems and provides the guest operating systems with a virtual operating platform. The physical resources that have been virtualized may be shared by several instances of different operating systems.

On server hardware, hypervisors are often deployed with the purpose of running guest operating systems that behave as servers. The physical server, the different virtual operating systems, and their anchoring virtual machines are all included in the security of the hypervisor, together with the kernel software and the underlying physical machine.

Compromising the Hypervisor: Neil MacDonald, vice president of Gartner Research and a Gartner Fellow3, notes the following about a hypervisor and the risks connected with it in his blog post "Yes, Hypervisors Are Vulnerable":

Since the virtualization platform (hypervisor/VMM) is software created by people, it will have flaws. All of them, including Microsoft, VMware, Citrix, and others, will inevitably have vulnerabilities. Some of these flaws will cause the isolation that the virtualization platform was designed to ensure to fall down. Attacks from the opposition will target this stratum. While there have been a few publicly revealed assaults, it is just a matter of time until a significant organizational breach is linked to a hypervisor vulnerability. The advantages of compromising this layer are simply too large.

Load Balancer Security: Each hypervisor has a load balancer, which is used to direct traffic to various virtual machines in order to distribute it more equally across the available computers. Particularly during periods of heavy traffic, a load balancer in a hypervisor is essential for guaranteeing a fair allocation of available load to all virtual machines and maximizing the use of the cloud infrastructure. The role of an elastic load balancer in the cloud architecture can be summarized as follows:

It distributes incoming traffic among the cloud infrastructure by listening for all traffic headed for the internal network. It adjusts its capacity for processing requests automatically in response to incoming application traffic. It offers extra networking and security choices if and when necessary, as well as managing the security groups connected to each instance. It is able to determine the state of the virtual machines, and if it identifies a load-balanced virtual machine that is not in good condition, it diverts traffic away from it and distributes the load among the healthy virtual machines that are still running.

It allows user sessions to be permanently attached to certain virtual machines.

It provides SSL termination at the load balancer, including SSL decryption offloading from application virtual machines, centralized administration of SSL certificates, and encryption to backend virtual machines with optional public key authentication.

It is compatible with both IPv4 and IPv6 (Internet Protocol versions 4 and 6).

The load balancer is a prominent target for attackers because of its capacity to listen for and handle all traffic sent at the internal network of the cloud. An attacker may listen to traffic and compromise secure traffic going outside the network if a load balancer was hacked. Moreover, traffic may be sent to an insecure internal server where additional attacks are conducted if the load balancer and a virtual machine are both exploited. The load balancer is very susceptible to denial-of-service assaults since it is a single point in the cloud architecture. Compromise may result in the interruption of cloud activities.

Hence, how can the load balancer be protected from threats? The typical methods for securing a load balancer include correct setup and log monitoring. This is accomplished by setting the load balancer to only allow administrative access via a designated administrative network, which restricts access to the load balancer's administration. Connecting this administrative network to the administrative only network is recommended. The number of people who have access to the load balancer is significantly reduced by restricting access over the administrator network.

Security for Virtual Operating Systems: The virtualization system also hosts virtual servers that each run either a guest operating system or another hypervisor, in addition to the hypervisor and load balancer. Moreover, hosts and consoles are in the virtual machine system's periphery. The virtual machine system may be exposed to security flaws via each of these resources.

Data Security in Transition: Recommended Practices for Cloud Security: Given the risks in the cloud that have been mentioned, there are a number of strategies to safeguard cloud users. The main areas of concern for cloud users are first and foremost unauthorized access to customer data and other resources stored or implemented in the cloud, the strength of the encryption used by the cloud provider to protect customer data, secure access to and use of cloud applications, and secure cloud management. Including all of them into the service level agreements (SLAs).

Service Level Contracts (SLAs): Between a service provider and a customer, a service-level agreement (SLA) specifies the degree of service that is anticipated in terms of security, availability, and performance. The efficacy of these contracts relies on how effectively these services are optimized and matched to the unique demands of each customer. SLAs are a set of service contracts between cloud providers and clients that specify the level(s) of service depending on the kinds of services required by the client.

Encryption of Data: The data's encryption is also crucial. Data is kept in a shared environment called the cloud as soon as it leaves the end-point web-cloud access point at the user's location. In the open or with others, data may be intercepted and compromised by outsiders both within and outside of the cloud, as well as by man-in-the-middle cryptanalysts during transmission. Strong authentication and encryption protocols are required to stop these types of intrusions. Strong access control and authentication to all web-based cloud resource interfaces, encryption of all administrative access to the cloud hypervisor, and encryption of all access to applications and data are all necessary for encryption to protect against all kind of data breaches.

Internet Access Points Security: Most examples of cloud access are web-based. The majority of security lapses involving stored data were caused by Web apps. Strong security measures are therefore required in the cloud APIs.

Compliance: Cloud providers are required to adhere to a number of compliance regulations, including FISMA, HIPAA, SOX, and SAS 70 II for clouds based in the United States and the Data Protection Directive for clouds based in the EU. These regulations are in place because most clouds are either public, community, or hybrid and because clients using these clouds typically work for companies that deal with personal data. Moreover, PCI DSS compliance is required for service providers that take credit card payments.

CHAPTER 16

IDENTITY VERIFICATION AND AUTHORIZATION

Merin Thomas, Associate Professor,
School of Computer Science and Engineering, Jain (Deemed to be University) Bangalore, Karnataka, India
Email Id- merin.thomas@jainuniversity.ac.in

Authentication in Cyber Security and its types: Authentication in cyber security refers to the process of verifying the identity of a user or device attempting to access a computer system or network. This is an important aspect of cyber security, as it helps to prevent unauthorized access and protect sensitive information.

Several different types of authentication methods can be used, including:

1. Something you know: this includes passwords, PINs, or security questions.
2. Something you have: this includes security tokens, smart cards, or mobile devices.
3. Something you are: this includes biometric authentication, such as fingerprints or facial recognition.

A commonly used method is the combination of two or more factors, known as multi-factor authentication (MFA). Multi-factor authentication can provide a higher level of security than single-factor authentication, as it requires multiple forms of proof of identity before access is granted .

Single-factor authentication is considered less secure and more vulnerable to attacks such as phishing, spear phishing, and social engineering. It is usually recommended to use multi-factor authentication, which can include a combination of something you know, something you have, and something you are. Organizations can also use role-based access controls (RBAC) to restrict access to systems and data based on an individual's role and responsibilities within the organization. This can help to prevent unauthorized access and protect sensitive information.

Authentication is a critical component of cyber security, and it helps to prevent unauthorized access and protect sensitive information. Organizations should use multi-factor authentication, and use role-based access controls to restrict access to systems and data based on an individual's role and responsibilities. Additionally, organizations should regularly review and update their authentication policies and procedures to ensure they stay current with the latest security threats and best practices .

Cyber security is the use of password management Strong and unique passwords are a critical component of authentication, and organizations should encourage the use of long, complex passwords, and prohibit the use of easily guessable passwords. Organizations can also use password managers, which can help users to generate and store strong and unique passwords and can also help to detect and prevent password-related attacks such as brute force and dictionary attacks.

Organizations should also have a process in place to manage and secure authentication credentials, including the use of secure storage and encryption to protect passwords and other sensitive information.

Another important aspect of authentication in cyber security is to monitor and detect suspicious login attempts. This can include using tools such as intrusion detection systems and security

information and event management (SIEM) systems to detect and respond to suspicious login attempts in real-time. Organizations should also regularly review and update their authentication policies and procedures to ensure they stay current with the latest security threats and best practices. This can include regular testing and evaluating the effectiveness of their authentication methods and technologies, and making changes as necessary to improve security.

Authentication is a critical component of cyber security, and it helps to prevent unauthorized access and protect sensitive information. Organizations should use multi-factor authentication, password management, secure storage and encryption for authentication credentials, monitoring and detection of suspicious login attempts, and regularly review and update their authentication policies and procedures to ensure they stay current with the latest security threats and best practices.

In Cyber security the use of Single Sign-On (SSO) technology. SSO allows users to access multiple applications and systems with a single set of login credentials, rather than having to remember and enter different login information for each system. This can help to improve user productivity and reduce the risk of password-related security incidents such as phishing and brute force attacks. Organizations should also consider implementing adaptive authentication. An adaptive authentication is a dynamic approach to authentication that adjusts the level of security based on the user's context. For example, if a user is trying to log in from an unknown device or location, the system may require additional authentication factors such as a one-time code sent to a registered mobile phone.

Another important aspect of authentication in cyber security is the use of security tokens and smart cards. Security tokens are physical devices that generate one-time passwords, which are used in addition to a user's password to provide an additional layer of security. Smart cards are similar to security tokens, but they store the user's authentication credentials on the card itself. Organizations should also be aware of the risks associated with using third-party authentication services, such as social media logins. While these services can be convenient for users, they also introduce additional risks, such as the potential for account takeover attacks and the possibility of sharing sensitive information with third-party providers.

Authentication is a critical component of cyber security, and it helps to prevent unauthorized access and protect sensitive information. Organizations should use multi-factor authentication, password management, secure storage and encryption for authentication credentials, monitoring and detection of suspicious login attempts, regularly review and update their authentication policies and procedures, Single Sign-On (SSO), adaptive authentication, security tokens, and smart cards, and be aware of the risks associated with using third-party authentication services.

Verifying someone's or an information source's identity is the process of authentication. To ascertain if someone or something is, in fact, who or what it is claimed to be, the procedure employs information given to the authenticator. In computers, the authentication procedure often entails the user or another party providing a password issued by the system administrator to log in. The password is intended to serve as a verification of the user's identity. It denotes that the system administrator granted the user's request for a self-selected password at some earlier point and registered or assigned it to the user.

In order to establish one's identity to an authenticating agent, authentication often calls for the production of credentials or valuables. The objects of worth or the credentials are based on a number of distinctive criteria that demonstrate what you do, own, or know:

The phrase "something you know" refers to mental capacity. This might be a secret phrase or password that only the user and authenticator know. While this approach is inexpensive to administer, system administrators must make sure that password files are preserved securely since users often forget their passwords. The user has the option of using the same password for all system logins or changing it on a regular basis, which is advised. This factor includes things like passwords, passphrases, and PINs (Personal Identification Numbers).

Any kind of issued or obtained self-identification, such as a SecurID, CryptoCard, Activcard, or SafeWord, counts as something you own. Since it is difficult to misuse specific bodily identifications, this form is a little safer than anything you are familiar with. For instance, losing the card itself is harder than forgetting the card's number.

A bodily trait or characteristic of you, such as your voice, fingerprint, iris pattern, or other biometric. Although it is possible to lose what you have and forget something you know, you cannot lose yourself or what you are. Thus far, it seems that this is the most secure method of ensuring an individual's identity. Due to this, biometric identification is currently a fairly common method of identifying. Despite the fact that using biometrics is fairly simple, biometric readers are still highly costly.

Three methods of authentication implementation are used in daily life:

A server that manages a user file containing passwords, usernames, or any other practical piece of authenticating information is required for basic authentication. Before granting authorisation, this information is always analysed. While this is the most typical method of user authentication used by computer network systems, it has a number of flaws, including the possibility of users forgetting or losing passwords.

In challenge-response authentication, a challenge is generated by the server or another authenticating system and sent to the host along with an expectation of a response.

Centralized authentication occurs when all network users are identified, authorised, and audited by a single server. The client requesting authentication will subsequently be granted permission to utilise the required system resources if the authentication procedure is successful; if it is unsuccessful, permission will not be granted.

Authentication Methods: Non-repudiable and repudiable authentication are the two forms of authentication currently in use.

False-Proof Authentication: Physical traits that define what you are cannot be disputed, making authentication based on them impossible. This authentication is nonrepudiable. Since biometric traits cannot be misplaced, lost, stolen, guessed, or altered by an intruder, they may be used to definitely validate a person's identification. As a result, they provide a highly trustworthy method of access control and authorisation. It's also crucial to remember that modern biometric authorisation apps are automated, eliminating the possibility of human mistake during verification. Newer, more sensitive, and precise biometrics are being created as technology advances and our knowledge of the human body deepens.

Digital signatures are irrefutable and more reliable than biometrics in terms of being non-repudiable authentication objects. These signatures, created by Chaum and van Antwerpen, cannot be validated without the assistance of the signer and cannot be refuted by the signer with any reasonable likelihood. A confirmation or denial process is used to determine the veracity of the signer. As many undeniable digital signatures are built on the RSA architecture and technology, they have demonstrable security, making it just as difficult to forge undeniable signatures as it is to forge regular RSA signatures.

A sort of irrefutable signature known as a confirmer signature13 allows the signer to choose a third party, known as the confirmer, to confirm the validity of the signature.

Last but not least, there are chameleon signatures, a sort of unmistakable signatures in which the veracity of the content depends on the signer's dedication to the information included in the signed document. The signer's permission is also required before the receiver of the signature may reveal the details of the signed documents to anybody else.

Untrustworthy Authentication: Any information that may be inferred from "what you know" and "what you have" might create issues if it is given to the authenticator since it may not be accurate. Such information may be unreliable due to a number of well-known issues, including knowledge that is forgotten, assets that are shared or stolen, and items that are readily forgeried, lost, or quickly copied. This makes authentication based on this information simple to refute.

Identification Procedures: There are several authentication techniques in use today. Password authentication, public-key authentication, remote authentication, and anonymous authentication are the most used types.

Credential Authentication: The oldest, most reliable, and most popular option we will cover is password authentication. On a lot of systems, it is configured by default.

Using the more recent keyboard-interactive authentication, it may sometimes be interactive. Reusable passwords, one-time passwords, challenge-response passwords, and mixed approach authentication are some of the varieties of password authentication.

User authentication and client authentication are the two categories of reusable passwords. The most frequent and well-known kind of authentication, user authentication, always starts with the user sending a request to the server for authentication and authorisation to utilise a particular system resource. The server requests a username and password from the user after receiving the request. The server searches its database for matches after receiving these submissions. The match determines if authorisation is given. Contrarily, with client authentication, the user first asks the server for authentication before being granted permission to utilise a system or a specific set of system resources. It's possible that an authorised user won't be able to access every system resource they want. By this authentication, user authorisation is established to utilise the requested resources up to the given limit.

Once used, one-time passwords, commonly referred to as session authentication, are discarded after each session. Strong random number generators are used to create passwords at random, which lowers the likelihood that someone would guess them. One-time passwords come in a variety of forms. S/Key and token are the two most popular. An S/Key password is a one-time password generating system based on the MD4 and MD5 encryption algorithms and is documented in RFC 1760. It was created to defend against replay attacks, such as when a hacker eavesdrops on a network login session and steals the password and user ID of the authorised user. The client begins the S/Key exchange by sending the initial packet, and the server replies with an ACK and a sequence number. This protocol is built on a client-server concept. A token password is a method of creating passwords that needs the usage of a unique card, such as a smart card. The challenge response and time synchronised techniques are the foundation of the system.

Challenge-response passwords employ a handshake-based authentication method in which the authenticator challenges the user who is requesting authentication. To be authorised, a user must provide an accurate answer. Depending on the system, the challenge might take many different shapes. The challenge may appear as a notification indicating "unauthorised access"

and requiring a password in certain systems. In some systems, it may just be a straightforward request for a password, a number, a digest, or a once a server-specified data string that might be randomly created each time a server returns a 401 server error. The system challenge must be answered by the individual requesting authentication. Password tokens, also known as asynchronous tokens, are now used for one-way answers. The server verifies the password's accuracy after receiving the user answer. If so, the user has been verified. If not, or if the network does not wish to accept the password for any other reason, the request is rejected.

For increased security, combined-approach authentication employs several combined authentication approaches. Using a random challenge-response exchange with digital signatures is one of the most secure authentication techniques. The authentication system, which might be a server or firewall, sends back a random string as a challenge whenever the user tries to establish a connection. The user's private key is used to sign the random text before it is delivered back as a response. The user's public key may then be used by the authenticating server or firewall to confirm that the user is in fact the owner of the corresponding private key.

Authentication with public keys: Each user of the scheme must first produce a pair of keys and store each one in a file. This is necessary for the process of public-key authentication. Typically, keys range in length from 1,024 to 2,048 bits. A key generation tool is often used to produce public-private key pairs. The pair will be made up of a user's public and private key pair. The user's public key is known by the server since it is generally available. The private key, however, is only accessible by the user.

Authentication systems often employ public key systems to increase system security. Public key system authentication is handled by the centralised authentication server, sometimes called the access control server (ACS). When a user wants to access an ACS, it searches for the user's public key and uses it to issue a challenge to the user. The user must utilise his or her private key in order to respond to the challenge, as expected by the server. The user is verified as valid if he or she signs the reply using their private key. The private key never leaves the user's computer, increasing the security of the public key. As a result, it cannot be lost or stolen like a password. Moreover, even if the private key is taken, the attacker still has to guess the passphrase in order to obtain access since the private key is linked to a passphrase.

Secure Sockets Layer, Kerberos, and MD5 authentication are three types of public-key authentication: In Secure Sockets Layer (SSL), public key infrastructure is used to enable authentication, authentication, encryption, and data integrity (PKI). A public/private key pair is produced before SSL authentication can start since it is cryptographically based. A certificate authority (CA), a dependable third party between any two communicating elements like network servers, issues verification certificates to communicating elements as proof that the other two or more entities involved in the intercommunication, such as individual users, databases, administrators, clients, and servers, are who they claim to be. These certificates are signed by computing a checksum over the certificate and using the private key of a signing certificate to encrypt the checksum and associated data. A signing certificate that may be used in the SSL protocol for authentication reasons can issue and sign user certificates.

Kerberos authentication is a network authentication protocol that uses PKI technology to provide robust authentication for client/server applications. When a user on a network tries to access a network service and the service needs confirmation that the user is who he claims to be, Kerberos is often employed. To do this, the kerberos authentication server issues a ticket to the kerberos user (AS). The service then looks through the ticket to confirm the user's identity. If everything is in order, the user receives an access ticket.

One of the common encryption techniques used nowadays for authentication is MD5 encryption. With MD5, the authentication procedure is fairly straightforward. Every user has access to a file that contains a set of keys that are fed into an MD5 hash. These keys are used to generate the MD5 checksum of the data being sent to the authenticating server, such as passwords, which is then sent together with the MD5 hash result to the authenticating server. The authenticating server then retrieves user identifying data, such as a password, and user keys from a key file. Finally, it computes the MD5 hash value. Authentication is successful if both parties agree.

Distance Authentication: Not every user has a direct connection to the networks they wish to access. In reality, a large number of employees utilise corporate resources remotely when travelling. For many system administrators, remote authentication is thus crucial. Users who call in to the ACS from a distant host must first authenticate themselves using remote authentication. There are numerous methods to do this, including dial-up, remote authentication, and secure remote procedure calls dialin user services authentication:

Clients that do not need to identify themselves to the server and servers that do not need any client identification employ Secure Remote Procedure Call (RPC) authentication. RPC authentication offers the higher level of security that some services, such the Network File System (NFS), need compared to the other services. As the RPC authentication subsystem package is open ended, RPC may employ a variety of authentication methods, including Diffie-Hellman encryption, UNIX authentication, data encryption standard (DES) authentication, and NULL authentication.

Dial-up authentication establishes the identity of a distant user, often one connected over ISDN or a serial connection. The Point-to-Point Protocol is the most used dial-up connection type (PPP). The peer device, not the device's user, is authenticated using dial-up authentication services. Many dial-up authentication methods exist. PPP authenti8—Information Security Protocols and Best Practices, for instance the Extensible Authentication Protocol (EAP), the Challenge Handshake Protocol (CHAP), and the Password Authentication Protocol (PAP) (EAP).

A remote authentication system. RADIUS is a widely used user protocol that enables users to call in to the ACS, which handles user authentication. RADIUS is seen as being open to assaults and insecure since all data from the remote host is sent in the open.

CHAPTER 17

DETECTION OF INTRUSIONS

Sindhu Madhuri G, Assistant Professor,
School of Computer Science and Engineering, Jain (Deemed to be University) Bangalore, Karnataka, India
Email Id- g.sindhumadhuri@jainuniversity.ac.in

A new tool called intrusion detection (ID) recognises the distinctive characteristics of the software employed in cyberattacks. The signatures are used by the detecting programme to ascertain the kind of assaults. There is a separate method for acquiring, analysing, and reporting network traffic information for each level of network investigative activity. Network traffic that is arriving or already present in the network is used for intrusion detection. ID tool creators think that traffic abnormalities will make it possible to tell attackers from authorised network users.

The abnormalities discovered by the ID analysis are, in fact, significant departures from prior use trends. ID systems are designed to recognise three types of users: approved users, authorised users engaging in prohibited activities, and, of course, invaders who have obtained the necessary identity and authentication unlawfully. ID sensors are often and readily positioned outside of the organization's firewalls on the edge of a private network. This often acts as the last line of defence for the organization's network and the final barrier to an outside network, most often the Internet. Also, having sensors on the same system as the firewall is becoming more and more typical. The sensors are better protected with this method, making them less susceptible to planned assaults. This site is an excellent initial line of defence since all potential assaults entering the organisation network travel through it, even if some attacks are invisible to certain sensors. To keep an eye on internal activity, ID sensors may also be placed within the network on network hosts and network subnets.

Newer ID tools with embedded extended rule bases that enable them to learn are being developed, and over time they will be able to make better analyses and, therefore, decisions. This is because more research is being done in ID and because links between ID and artificial intelligence are being established. What sort of rule basis should be used in the ID tools is the topic of discussion, not what kind. The rule bases that educate the ID tools what patterns to search for in the traffic signature and how to learn those patterns are now used. For instance, a port number for an application should never be active if the server does not support it. The new trend, however, is distinct from the conventional embedded rule bases. The current emphasis is to teach these ID tools what Marcus J. Tanum refers to as "manufactured ignorance," or a rule basis that instructs them on what to search for and what not to seek for. Following this line of reasoning, many think the product will be more efficient and the rule base will be simpler.

Another change in ID systems' use is their scope. For some time, management and a large portion of the networking industry have incorrectly believed that ID systems shield network systems from outside intrusions. Yet, research has revealed that insiders are really responsible for the bulk of system invasions. Thus, this problem is the main focus of more recent ID tools. New ID tools are being created to combat system intrusions, and new attack patterns are being designed to take unexpected human behaviour into consideration. Moreover, the human mind is the most complex and unpredictable machine ever. ID systems must be updated often to stay up with all these changes.

In light of all these changes, the primary focus of ID systems is on the network as a whole, where network packet data is gathered by monitoring network packet traffic and then analysed based on network protocol pattern norms, typical network traffic signatures, and abnormal network traffic built into the rule base. The ID systems scan for three things: patterns of abuse, anonymous conduct, and signatures of well-known assaults.

One of three prevalent types is often included in the signatures of known attacks.

Text: These signatures are used to keep an eye on text strings that could point to an impending attack.

Port: These signatures are used to keep an eye out for programmes that try to connect to a port. Often, well-known and frequently attacked ports are used for the monitoring. TCP port 20, FTP port 21, and telnet port 23 are the most often attacked ports.

Header: These signatures look for unusual combinations of many well-known signatures, such as the IP address and sequence number signatures, in packet headers.

When the ID tools match observed activity to rule-based profiles for notable deviations, anonymous behaviours are discovered. The profiles are often for specific individuals, teams of users, resource usages of the system, and a variety of other things, as will be detailed below:

A user's frequent behaviours, with minimal deviance from the anticipated pattern, are included in their particular profile. This may apply to certain user occurrences such use lasting longer than normal, recent adjustments to user work habits, and big or irregular user demands.

A group profile includes information on a group of users' historical behaviours, resource demands, and work habits (146 Computer Network Security and Cyber Ethics). Every user within the group is expected to adhere to the collective activity patterns.

Monitoring the use patterns of system resources including applications, accounts, storage media, protocols, communications ports, and a long list of other items the system management may choose to include are all included in a resource profile. Depending on the rule-based profile, it is anticipated that frequent usage won't considerably stray from these guidelines.

The utilisation of system resources by executable applications is tracked by other profiles, such as executable profiles. For instance, if an executable software has a Trojan worm or a trapdoor virus encoded in it, this may be used to detect odd departures from the norm. There are also the following profiles in addition to executable profiles: Work profile, which includes port monitoring; static profile, which monitors other profiles and updates them periodically to prevent those profiles from gradually expanding to covertly introduce intruder behaviour; and adaptive profile, which monitors work profiles and updates them to reflect recent usage spikes. Finally, there is the adoptive rule-based profile, which keeps track of the historical use patterns of the previous profiles and modifies the rule base accordingly.

ID tools may effectively concentrate on usage patterns, or patterns of known misuse of system resources. Once identified, these patterns are matched against the rule base's descriptions of "bad" or "unwanted" resource utilisation. A knowledge database and a rule engine must be created in tandem to do this. Expert systems, model-based reasoning, or neural networks are the finest tools for analysing misuse patterns. We won't go into further detail about how each one operates. But, as networks grow and traffic volumes increase, it is growing more and harder for the ID system to "see" every data on a switched network such an Ethernet. This has prompted a new strategy of paying greater attention to the host. Hence, host-based and network-based ID systems are the two main kinds of ID systems.

Methods for detecting intrusions on hosts

In order to monitor individual user and application traffic handled by the network server, host-based intrusion detection systems (HIDS) approaches are used. Actually, it is monitoring log files and auditing traffic entering and leaving this particular system. Along with monitoring incoming and outgoing data, HIDS also checks the consistency of system files and keeps an eye on all of the machine's processes to look for unusual process activity. In fact, host-based ID systems are either sensor agents or personal firewalls. Wrappers, sometimes known as personal firewalls, are set up to monitor all network traffic, login attempts, connection attempts, and non-network interactions. Agents are set up to watch on changes to user rights as well as access to and modification of important system files. Host-based ID technologies, such as personal firewalls or agents, are effective in keeping an eye out for insider penetration on a network system.

Benefits of HIDS: The idea of HIDS is very new. As research revealed that a significant portion of illegal and unethical acts in organisational networks really started with the workers, they were widely used in the early and middle 1980s. The HIDS technology developed alongside other technological advancements throughout the next years. The advantages of HIDS for their overall security are becoming more and clearer to enterprises. In addition to being quicker than their network-based intrusion detection system (NIDS) counterparts and being able to handle less traffic, HIDS also provide the following benefits25:

The fast verification of an attack's success or failure. HIDS contain information that is more accurate and less prone to false positives than their relatives the NIDS since they record ongoing events that have really happened. With the use of this information, it is possible to determine quickly and correctly whether an assault was successful or not and to launch a timely defence. As a verification system, HIDS serves to supplement NIDS in this capacity.

Low-level observation. HIDS can "see" low-level local activities like file accesses, changes to file permissions, attempts to install new executables, attempts to access privileged services, changes to important system files and executables, attempts to overwrite crucial system files, or attempts to install Trojan horses or backdoors because they monitor at a local host. These low-level behaviours may be immediately identified, and timely reporting allows the administrator to take necessary action. These low-level assaults may sometimes be so subtle that no NIDS can pick them up. Response and detection in almost real time. A pace close to real-time may be reached by HIDS in reporting minute activity at the target hosts to the administrator. As a result, an intrusion may be discovered and prevented before significant harm is done. This is made possible by the operating system's ability to notice the event before any IDS can.

The capacity to operate in switched and encrypted contexts. It is common practise to switch slice large networks into several smaller network parts. Next, an NIDS is attached to each of these smaller networks. It might be challenging to decide where to implement a network-based IDS in a widely switched network to provide enough network coverage. Traffic mirroring and administration ports on switches may be used to remedy this issue, although they are less efficient. By being present on as many crucial hosts as necessary, HIDS offers the increased insight into these switched environments that is required. Moreover, HIDS that monitor the operating systems can deal with these encryptions better than NIDS, which may not even deal with them at all, since the operating systems observe incoming data after encryption has already been decoded.

Economical: The firm may save a lot of money since HIDS installation requires no new hardware. This contrasts positively with the high expenses of NIDS installation, which calls

for pricey and specialised servers. In reality, this expense may pile up in huge networks that are switch sliced and need a lot of NIDS per segment.

The drawbacks of HIDS: While HIDS have numerous benefits, they are still limited in what they can do. The following are some of these restrictions: HIDS have a narrow perspective. Although HIDS are placed near to users, they are more vulnerable to unauthorised manipulation since they have a highly constrained view of the network due to their deployment at a host.

Systems for Network-Based Intrusion Detection: Network sensors known as NIDS are set up to monitor all network activity, including activity on communication media, network servers, and firewalls. To detect intrusions, they keep an eye on network traffic. They are in charge of finding unusual, improper, or other material that may be regarded as damaging and illegal when it appears on a network and firewalls function quite differently, therefore may or may not run with firewalls. According to a set of rules, firewalls may be set up to either permit or prevent access to a certain service or host. Regardless of the content of the packet, traffic is only allowed to go forward when it conforms to a valid pattern. NIDS acts solely by issuing an alert if the packet signature, based on the contents of the packet, is not among the authorised signatures, even though it also gathers and inspects every packet that is intended for the network regardless of whether it is permitted or not.

An NIDS sensor may be installed and used in a variety of ways. It may be set up and operated as a standalone computer that monitors all external traffic entering the network, just monitors traffic entering a certain subnet, or simply acts as the target machine and monitors its own traffic. In this mode, for instance, it may keep an eye on itself to see whether somebody is trying to do a SYN flood or a TCP port search.

Although while NIDS, when properly deployed, may be quite successful at collecting all incoming network traffic, it is possible for an attacker to avoid detection by taking advantage of ambiguities in the data stream as seen by the NIDS. The causes of these exploitable ambiguities are listed by Mark Handley, Vern Paxson, and Christian Kreibich as follows27:

A lot of NIDS lack the ability to examine the whole spectrum of activity that may be revealed by the user and permitted by a certain protocol. Even if the NIDS analyses the protocol, the attacker may still avoid detection. Since NIDS are separated from specific hosts, they lack complete knowledge of how each host implements their protocol. Since various implementations interpret the same stream of packets in different ways, this information is crucial for the NIDS to be able to understand how the host may handle a certain series of packets. Once again, NIDS may be unable to identify whether a certain packet will ever be viewed by the hosts because they lack a complete understanding of the network topology between them and the hosts.

Benefits of NIDS NIDS seem to have several benefits over their relatives, firewalls and host-based IDS, as we shall soon learn. These advantages stem from their emphasis, location, operating, and needs. Since NIDS monitor network traffic at the Transport Layer, they can identify assaults that a host-based system might overlook. NIDS are able to look at both the packet addresses and the packet port numbers from the packet headers at the 150 Computer Network Security and Cyber Ethics level. Certain forms of attacks may go undetected by HIDS because they monitor traffic at a lower Link Layer.

Difficulty getting rid of evidence. In contrast to HIDS, which are located close to or at the attacker's workstation, NIDS are on specialised computers that are frequently safeguarded, making it more difficult for an attacker to delete evidence. It is particularly difficult for an

attacker to delete evidence since NIDS utilise real network traffic, which is what is caught by NIDS when there is an attack.

Instantaneous detection and reaction. NIDS can identify foreign intrusions into the network in real-time and report as fast as possible to the administrator for a prompt and suitable reaction since they are located at the most strategic and advantageous entrance points in the network. A speedy and suitable reaction is made possible by real-time notification, which is currently found in many NIDS. The administrators may even be able to give the intruder additional time if they conduct more focused monitoring.

The capacity to recognise failed assaults and nefarious intent.

As the HIDS are within the secured internal network, they are never exposed to many different sorts of attacks since they are often blocked by the external firewall. The assaults that evade the initial firewall are encountered by NIDS, particularly those in the DMZ, and are subsequently denied by the inner firewall and those aimed at the DMZ services that have been let in by the outer firewall. In addition to displaying these assaults, NIDS may also keep track of their frequency.

Negative aspects of NIDS

1. NIDS have limitations, even though they are extremely well suited to monitoring all traffic entering the network or subnet.
2. Blind spots: When placed at a network's edge, NIDS are blind to the whole interior network. Due to the placement of sensors in certain locations, particularly in switched networks, NIDS can have whole network segments they are blind to.
3. Encrypted data: This is one of the main areas where NIDS fall short. They lack the tools necessary to decipher encrypted data. Only portions of the packets that are not encrypted, such headers, may be scanned.
4. While ID technology has advanced significantly and has a bright future as its marriage with artificial intelligence matures, it still confronts several difficulties. ID technology is currently constrained by many issues.
5. The issue of false alarms is one. While the tools have advanced considerably and are gradually gaining acceptability as a result of their broad usage, they continue to generate a significant amount of false positives and false negatives.
6. The fact that technology is still insufficient to handle a massive attack is a second issue. This is due to ID's inherent requirement to thoroughly scan each and every packet, contact point, and traffic pattern in the network. Larger networks cannot depend on technology to continue operating with sufficient quality and grace amid a large-scale assault. Current technology is unable to effectively manage extremely quick and high volumes of traffic, barring a breakthrough today.
7. The largest obstacle is probably how ID's capabilities are seen, and often inflated. While the technology is excellent, it is not a panacea for all computer network woes as some have claimed. Similar to any other reliable security tool, it is.

Antivirus Software: A software programme that monitors or checks a system, including its data and programme files, for the presence of viruses is a virus detection programme, sometimes known as an antivirus application. It is crucial to get rid of any virus that has already infected a system, whether it is active or dormant.

Antivirus software employs a variety of ways to find viruses at any stage of development. The use of methods like file length, checksum, and viral signature detection is one example. A virus may be identified using its signature, which is a particular and distinctive collection of bytes

exhibiting distinctive viral traits. The instructions for the virus include the most prevalent of these traits. Every virus has unique traits that are unique to it. To strengthen protection against potential viruses, the known traits are exploited. While new viruses are developed and spread virtually daily, virus and e-attack reporting organisations and centres say that the same old viruses continue to be the most prevalent ones in use. So, it makes sense to develop defences against future e-attacks using previous information on viruses and their traits. This method is used by the majority of antivirus scanners nowadays to find infections in systems. Only known viruses may be identified via signature detection, which is one drawback. To create a signature archive, virus scanners must be updated often.

Due to the fact that viruses attach themselves to software as their surrogates, file duration is a valuable detection criterion. The surrogate software's length often grows as a result. When a file or piece of software is utilised, antivirus software compares the lengths of the original and used versions. If the two lengths are different, a virus may be present. It should be noted that this approach simply identifies the existence of a virus; it does not identify the kind of virus in the file or data.

When determining if data has been changed by a virus without lengthening the file, a checksum is a value computed in the file. Antivirus checkers may employ a checksum in two different methods. One method is to calculate and save the total amount of bytes in the file. The antivirus programme calculates a new checksum for the file each time it is used and compares it to the original checksum. The antivirus programme alerts users to the presence of a virus if the new value deviates from the stored original. The second method computes the checksum as the total of all the binary words in a file, more likely for tiny files. This approach works better to find viruses that just change a file's content rather than lengthen it in any manner. The checksum for transmission data is generated both before and after the data is transferred. The checksum will show whether a virus was added between the source and destination. A checksum should only be used if it is certain that the file was virus-free the first time it was calculated; otherwise, it will never catch a virus that was present when the file was used for the first time.

If a virus is identified in a file or piece of software, its symptoms will be present. The symptoms of a virus vary depending on its kind. Recall that not all viruses have the same symptoms. Several viruses might present with similar symptoms. Here are a few of the most typical signs:

1. Unexpected or frequent PC restarts.
2. Unexpected growth in software and data size.
3. The loss of data files.
4. The inability to save open files.
5. Memory deficit.
6. The existence of odd noises or writing.

Legislation As e-commerce and Internet usage grow, citizens of all countries who advocate for issues like environmental protection, media-reported violence, pornography, gambling, free speech, intellectual property rights, privacy, and security are making significant contributions to legislation. 8—Information Security Protocols and Best Practices 153 exert pressure on their country legislatures and other legislative bodies to pass legislation that restrict Internet use in the manner that these organisations believe will best suit their interests.

Countries like the United States, the United Kingdom, Germany, France, China, and Singapore are already experiencing this. Every day that goes by, the list lengthens. Several restrictive regulations, some excellent, are being passed in all of these nations to impose restrictions on online behaviour. Recent increases in unlawful Internet activity, including well known distributed denial of service assaults and headline-grabbing e-mail attacks, have prompted

demands for legislative action to put an end to these activities. Yet, it is unlikely that such measures would be effective in slowing or stopping the growing number of unlawful activity in cyberspace. The hodgepodge of laws will not, in the foreseeable future, meaningfully put a halt to these criminal actions. If anything, until and until long-term strategies are in place, such actions are likely to continue without ceasing. An education in ethics should be the initial component of such initiatives and goals.

Regulation: Governments all around the globe are being compelled to review, alter, and adopt new laws, charters, and acts as the battle between proponents of free speech and those who fight for the protection of minors becomes more heated. This has been one of the most well-liked and, for politicians, the most visible methods of Internet regulation, as we shall see in more detail in the next section. Law enforcement and judicial systems support legislative endeavours. A number of laws are in effect and enforceable in the United States. Several out-of-date acts have been examined and revised in recent years.

In addition to more open legislative procedures, there are also private efforts that collaborate with public judicial and law enforcement institutions or via workplace forces. There are several instances of huge corporations banding together to lobby their national governments to pass legislation protecting their interests, particularly high technology corporations like software, telecommunications, and Internet service providers. These businesses are also joining forces or creating partnerships to develop and use private control strategies.

Self-Regulation: Self-regulation as a method of Internet management is attractive to a sizable portion of people worldwide for a number of reasons. While regulation and enforcement may significantly contribute to the reduction of cybercrime, these measures alone cannot completely eliminate it. They need to be paired with other coordinated initiatives. Giving users adequate liberty to manage themselves is probably one of the most successful preventative strategies. Each user will take on the duty to the degree and level of control and regulation that best matches his or her requirements and surroundings. There are two ways to implement this self-regulatory cyberspace control: either hardware or software.

Hardware-Based Regulatory Self: Individual users may specify a variety of hardware security restrictions, including password protection, firewalls, memory access controls, file access controls, and authentication procedures for file access. Six areas are the focus of hardware controls: Prevention uses technologies that allow only authorised persons to the defined regions to limit access to information on system resources such discs on network hosts and network servers. Examples of such technology include firewalls.

Protection regularly determines, assesses, and updates system security needs to make them appropriate, thorough, and efficient. Detection sets up a monitoring system for early detection of planned and ongoing security breaches. Response evaluates all potential security gaps and develops pertinent corrective actions for a better security system based on observed failures. Limitation reduces the damages sustained in situations of failed security. Recovery refreshes contingent recovery plans and restores what has been lost as rapidly and effectively as feasible.

Computerized Self-Regulation: The software-based approach is less intimidating, making it more approachable and user-centered. This implies that it may be installed on a network server by a network system administrator or by the user on the user's PC. The settings for the required degree of control may be specified by the user if they install the software. Controls are imposed at the network level using a firewall or a specialised software programme and are decided upon by the majority of users. Ratings programmes and filtering programmes are the two types of software controls. Similar to how the film business rates movies for violence, profanity, and sexual content, rating systems assess Internet material.

Software rating labels allow Internet content suppliers to voluntarily affix labels to their goods in accordance with a predetermined set of standards. Due to the fact that they are offered by many rating organisations, such as CyberPatrol, CYBERsitter, Net Nanny, and Surf Watch, these labels are not standard throughout the industry. They all promise to offer a simple, but effective rating system for Web sites to safeguard minors and free expression for everyone who publishes on \sthe World Wide Web. The filtering application on the user's PC or server then makes use of these labels. Every Web document's header is always inspected by the filtering algorithms in search of a label.

Filtering software prevents access to documents and Web pages that include anything marked on a filter list, often offensive words and URLs. Filters may either be server-based, where they are centrally hosted and maintained, or client-based, where they are installed on the user's computer. Since they are difficult to tamper with, server-based filters provide superior security. While filtering software, both server-based and client-based, has lately gained a lot of popularity, it still has significant issues and downsides, including inaccurate categorization, limitations on unrated content, and the simple intentional exclusion of certain Web sites by a person or people. There are various causes of inaccuracy. Since they are close to a file that contains some adult material, certain websites are restricted. For instance, the Web site hosting the file without adult content can be restricted if other items are in the same directory as the adult-content file.

Websites may sometimes be restricted because they include terms that are judged offensive. These terms may sometimes be foreign words with entirely distinct meanings. Also, the user's choice to either ban or unblock unrated content may restrict their access to pertinent data. Blocking software only functions well when all online content is graded. Yet, as we are all aware, it is currently impossible to review everything that is available on the Internet due to the hundreds of thousands of new Web sites that are uploaded daily.

Broad moral and ethical instruction: Our opinion is that widespread moral and ethical education is one of the most effective methods for preventing and reducing criminal online activity. the need of instilling moral values and computer ethics in all computer users. In these value of having a solid moral and ethical foundation and how this helps to become a strong, honourable person. We are well aware that character education is challenging and that have been having trouble addressing this problem in schools around the United States. But in our opinion, character education shouldn't be restricted to the classroom. The home is where character education should begin. Character education must heavily involve the family. Character education has little value without this essential element, and the debate we now have over it will persist.

While we support a robust moral and ethical education for all people at home and in school, we also recognise the challenges that come with our varied society. Yet the sooner we address these issues head-on, the better, since there is no going back thanks to contemporary technology and the forces of globalisation. Society all across the globe are moving quickly towards diversity.

Many individuals do not believe that character education alone can solve the problem. To them, we say, let's create a solid plan that includes us all—the parents, the instructors, and you and I—to educate our kids about this new technology, the best ways to use it, and its risks. This is the moment to act if anything has to be done.

From kindergarten through college, formal character education should focus on the whole educational range. The level chosen will, however, affect the concentration and contact. For instance, it is suitable to teach children about the risks of information abuse and basic computer

ethics in elementary school, and both the material and the way it is delivered are evaluated for that level. The material and delivery method are more concentrated and aggressive in high school since the kids are more independent and inquisitive. College requires a different approach since students are more major-focused, and the intended teaching should match this.

The foundation of occasional or continuous education is the premise that teaching computer ethics in particular and ethical use of information in general is a lifetime endeavour. Professionals should be given this duty, as they often are. Professions may impose this education in a number of ways on its members. This is accomplished for many conventional professions via the creation and application of professional codes, standards, and canons. Some professions add refresher courses and in-service training requirements to their rules of conduct. Several professions need licensure as a way to ensure that its members are able to continue their education.

These facilities' main goal is to gather all pertinent data about cyber 8—Information Security Protocols and Best Practices 157 assaults and make it accessible to the public. When someone suspects or has proof of an electronic attack, the centres serve as the first point of contact. The centres also serve as informational resources for anyone who want to know more about the steps to take to stop, identify, and recover from assaults.

The NIST Computer Security Resource Clearinghouse, the Federal Computer Incident Response Capacity, the Center for Education and Research in Information Assurance and Security, the Carnegie-Mellon Emergency Response Team, the FedCIRC, and the National Infrastructure Protection Center are just a few of the reporting centres in the United States that are supported by the government and funded by private sources. These facilities may be divided into two groups:

(i) Organizations outside of law enforcement that compile, organise, and inform the public on all facets of cyber assaults, including defence, detection, and survival.

(ii) (ii) Law enforcement agencies that serve as national clearinghouses for cybercrime, collaborating directly with other national and international computer emergency response teams to keep an eye on and evaluate emerging dangers. In collaboration with business and foreign law enforcement organisations, law enforcement centres may also teach local law enforcement personnel. These facilities encompass all crimes committed via the wires, including those using telephones, and they not only concentrate on government break-ins but also those in the commercial sector.

Advisories

Due to the increase in e-attacks, government and private sector organisations have joined forces to educate the public about the risks associated with e-attacks and how to reduce their likelihood by removing vulnerabilities. When their goods are hacked, both large software and hardware manufacturers are highly active and quick to publish, transmit, and broadly disseminate alerts, vulnerability patches, and antivirus software.

A significant maker of Internet infrastructure network devices, Cisco has started contacting and emailing its clients, mostly Internet service providers (ISPs), to alert them to the possibility of cyberattacks that target Cisco's technology. Moreover, it alerts users to software updates that may be applied to stop or stop such assaults. Via its websites, it has also contributed to the public's access to crucial information about such assaults and how to avoid them and recover from them. Computer Network Security and Cyber Ethics on the software front

Microsoft, the most impacted target in the software industry, has also been active, providing crucial and important information on how to avoid and recover from assaults against its products by publishing, contacting, and emailing its customers.

Public reporting centres have also been active, issuing alerts of potential assaults and instructions on how to recover from attacks.

Application Service Providers' Function

Companies who decide to set up computer networks are soon in for a startling shock: the cost of the related equipment is only the tip of the financial iceberg. The biggest expense of owning a network is the personnel required for proactive and reactive maintenance, which keeps the network functioning properly. The difficulty of attempting to keep up with computer networks has given birth to a new business called application service providers (ASPs) in recent years. They move the majority of network operations, including security, outside of the company to a single location, where dependable computers can handle tasks like data backup, virus scanning, data storage, application software, and a variety of other tasks that would typically be handled by the company's hardware and staff. Small firms in particular lack the means and ability to achieve this. There are always teams of network experts accessible in all areas of network functionality.

Patching: It happens rather often for businesses to publish software only to later discover bugs and security weaknesses that allow attackers to access the resources the programme is operating on. Companies provide updates to close these gaps and mistakes as soon as they are made aware of them. The most recent fixes from software suppliers and manufacturers should be sought after by security directors, system administrators, and people.

When newly installed operating systems are initially used, all of the computer system's networking facilities are enabled, allowing hackers an opportunity to investigate system weaknesses. All users should take responsibility for their actions by shutting down any unnecessary network services before installing a new operating system.

Questionnaires

1. What is the difference between a domain and an autonomous system?
2. Briefly explain the functions of data link layer?
3. Briefly explain the functions of data link layer.
4. Briefly explain any one framing method.
5. Define flow control.
6. Define framing.
7. What is Network and Cyber Security?
8. What is the history about internet?
9. What is the malware and discuss its types?
10. What is Cyber Stalking?
11. What is Computer Hacking?
12. What is authentication and its type?
13. What is digital signature?
14. What is antivirus?
15. What is firewall?
16. What is Steganography?
17. What are Computer Forensics?

Reference Books for Further Reading

1. Brian Hatch, James B. Lee, George Kurtz, " Hacking Linux Exposed," McGraw-Hill, March 2001.
2. Joel Scambray, Stuart McClure, " Hacking Exposed Windows 2000," McGraw-Hill, August 2001.
3. Russ Housley, Tim Polk, " Planning for PKI: Best Practices Guide for Deploying Public Key Infrastructure," Wiley, March 2001.
4. Daniel J. Barrett, Richard Silverman, " SSH, The Secure Shell: The Definitive Guide," O'Reilly & Associates, February 2001.
5. Juanita Ellis, Tim Speed, William P. Crowell, " The Internet Security Guidebook: From Planning to Deployment," Academic Press, February 2001.
6. John E. Canavan, " The Fundamentals of Network Security," Artech House, February 2001.
7. Jon C. Graff, " Cryptography and E-Commerce: A Wiley Tech Brief," Wiley, December 2000.
8. Philip Cox, Tom Sheldon, Phillip Cox, " Windows 2000 Security Handbook," McGraw-Hill, November 2000.
9. Joel Scambray, Stuart McClure, George Kurtz, " Hacking Exposed," McGraw-Hill, October 2000.
10. Eric Rescorla, " SSL and TLS: Designing and Building Secure Systems," Addison-Wesley, October 2000.
11. Stephen Northcutt, Donald McLachlan, Judy Novak, " Network Intrusion Detection: An Analyst's Handbook (2nd Edition)," New Riders Publishing, September 2000.
12. Jan Killmeyer Tudor, " Information Security Architecture: An Integrated Approach to Security in the Organization," CRC Press, September 2000.
13. E. Eugene Schultz, " Windows NT/2000 Network Security," New Riders Publishing, August 2000.
14. Michael Howard, " Designing Secure Web-Based Applications for Microsoft Windows(r) 2000," Microsoft Press, July 2000.
15. Mark Cooper, et al, " Intrusion Signatures and Analysis," New Riders Publishing, January 2001.
16. Mike Wenstrom, " Managing Cisco Network Security," Cisco Press, January 2001.

17. Thomas R. Peltier, " Information Security Risk Analysis," Auerbach Publications, January 2001.
18. Stefan Norberg, Deborah Russell, " Securing Windows NT/2000 Servers for the Internet," O'Reilly & Associates, November 2000.
19. Richard Mansfield, " Hacker Attack," Sybex, September 2000.
20. Jeff Crume, " Inside Internet Security: What Hackers Don't Want You To Know," Addison-Wesley, August 2000.
