APPENDICES

CHAPTER 2

Appendix 2-1: ES Software Architecture and Client-Server Computing

ES software architecture conforms to a three-tier client-server architecture. ES applications are organized into three logical and computing tiers: a *presentation tier* that provides ES users with one or more UIs; an *application tier*, where the ES business logic is located and data is processed; and a *data tier*, where ES application data is stored and managed (see Figure 2-26). ES data is typically stored in a centralized database or a collection of several databases that appear to function as a single entity.

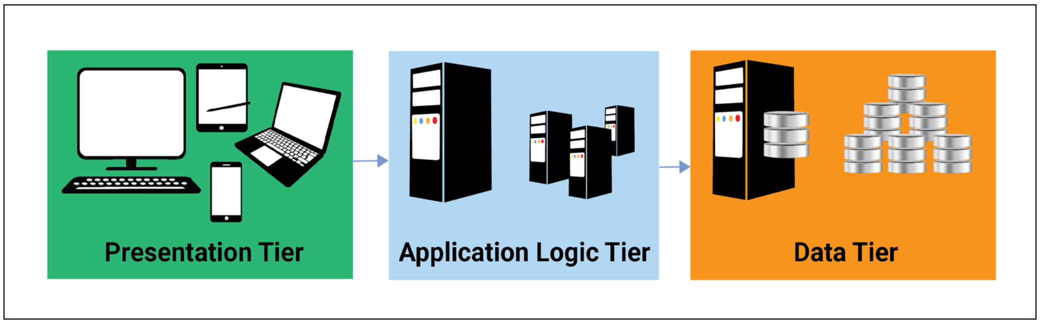


Figure 2-26: Conceptual depiction of the three-tier ES software architecture.

One benefit of this architecture is that each tier can run on its own infrastructure. Another benefit is that the three tiers can be colocated or be deployed at different locations. The independence of the tiers enables the technologies used at each tier to be updated or scaled without affecting the other tiers. For example, additional types of client devices, such as the Apple Watch or another wearable, can be added to the presentation tier without having to make changes to the application or data tier. Similarly, servers can be added to the application tier to accommodate more users and data traffic without having to make changes to the presentation or data tier. And, likewise, new database servers or storage technologies (such as SSD arrays) can be added to the data tier without making changes to the presentation or application tier.

Separate development teams can be created for each tier to ensure ongoing development and maintenance in response to technological advances or changing business needs. Independence among the three tiers also means that either the application tier or data tier (or both) could be extended or replaced by cloud services to provide additional and possibly less-expensive scaling and management.

Figure 2-27 illustrates how a three-tier ERP software architecture might be supported in a client-server network. Today, businesses have the option of deploying ERP application and database servers

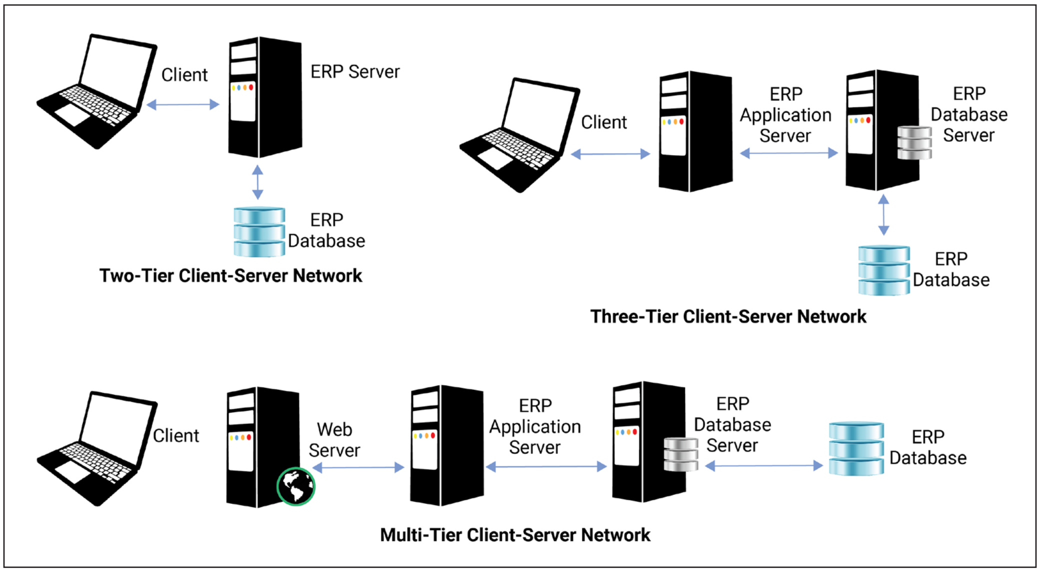


Figure 2-27: Examples of ERP deployment options in client-server networks.

in on-premises data centers or private clouds or accessing ERP application and supporting infrastructure in public clouds.

Early ERP adopters (in the 1980s and 1990s) typically deployed their systems as two- or three-tier client-server computing systems in on-premises server rooms or data centers. In two-tier deployments, the ERP UI is displayed on client devices at the presentation tier, and the ERP application software and database are located on the second tier.

In three-tier client-server deployments, the ERP software's application logic runs on application servers that interact with database servers to retrieve or update data in the ERP database. When ERP users interact with the ERP system via a browser interface (such as SAP NetWeaver), the deployment resembles the N-tier client-server network illustrated in Figure 2-27. Client-server deployments with three or more tiers are also called *multi-tier* client-server networks.

The migration from two-tier to multi-tier client-server network deployments reflects the growing importance of online access to ES software. With three layers, the ES software architecture facilitates business adoption of hosted cloud-based ES products. ES software vendors are encouraging their customers to migrate to cloud-based solutions, and industry experts expect them to be the most common means to access ES software in the years ahead.

While ES adopters may have the option of fully implementing an ES system on premises, it is increasingly common for some or all of the supporting infrastructure to be cloud based. Because scalable cloud storage is inexpensive, the ES database is often the first ES component to be deployed in or supplemented by the cloud. Some ES adopters deploy both the application and database tiers in the cloud, and some ES products (such as the Salesforce CRM product) are fully cloud based with no on-premises deployment option.

Fully hosted ES systems are available from software as a service (SaaS) providers, and their users interact with the ES software via apps or browser interfaces. This option enables organizations to minimize on-premises infrastructure and associated costs. In such deployments, SaaS providers perform ES software updates and upgrades; this is attractive to smaller businesses and start-ups that want ES capabilities for low cost and with minimal technical expertise requirements.

Infrastructure-as-a-service (IaaS) is another cloud deployment option for ES adopters. IaaS provides subscribers with the servers and storage technologies needed to support their ES software and data, but the subscribers are responsible for installing, managing, and maintaining the infrastructure they rent from the provider. The subscriber is also responsible for installing ES software updates and upgrades. Ongoing costs for this deployment option are typically less than those for fully on-premises infrastructures but are usually higher than those for fully hosted ES systems.

Tight Coupling in ES Client-Server Applications

In traditional client-server computing models, a client device connects to a server that provides a specific resource or service. This may be an application server that performs application logic, a database server that performs data logic, or a web server that provides a browser-like interface as a front end for the application. When the client connects to the server, there is an implicit understanding of the communication protocol that will be used to communicate and exchange data. Usually, the connection involves "tight coupling" between the client and server, and in the absence of redundancy and failover mechanisms, the client-server application can be disrupted by hardware or network failures.

Logging into an ES server often exemplifies coupling in client-server applications. SAP ERP users, for example, are usually required to provide login credentials to log into a specific application server or server daemon. The SAP GUI is often provided by a browser interface (e.g., SAP NetWeaver), which prompts users to select a server from a list of available SAP servers. They are subsequently prompted to enter a client number (which identifies a specific ERP configuration), a user ID, and a password. The user's login credentials are authenticated against those stored on the server. The login process must be repeated after each logoff; usually, there is no "remember me" or "remember this device" option.

Such a login process makes sense from a business perspective. Like other ESs, ERP systems have important transactional roles in business computing infrastructures, and because of their importance, it makes sense to have centralized control over system access, use, and administration. It also makes sense for ES users to be fully aware that they are logging into an important business system and to require the same login process for all types of client devices (desktop or laptop computers, tablets, smartphones, etc.) because of the tight coupling required when the application is being used.

While there is some abstraction in traditional client-server ES applications, with servers hiding behind the ones that clients connect to, one might guess that there are other servers behind the scenes that assist in processing the application. In cloud deployments, however, users have little, if any, inkling of the server-side technologies associated with their apps.

In cloud deployments, hardware, network topology, storage technology, server type, server location, and the number of servers involved in processing an application are all abstracted to a single endpoint—the cloud—that provides users with an application's functionality, often without the need for a regimented and centrally controlled/administered login process. Better informed users know that they are interacting with machines in one or more data centers when using an app, but they are typically clueless about data center location(s), network topology, server type, server hardware, and other application deployment details. Less informed users have little or no awareness of cloud infrastructure and often are uninterested in learning the details.

Appendix 2-2: Middleware

Enterprises benefit from middleware when it is used to ensure that an application is not restricted to specific client devices (e.g., a Windows desktop computer or laptop) or server platforms (e.g., UNIX servers). It provides implementation flexibility as well as potential cost savings. Middleware

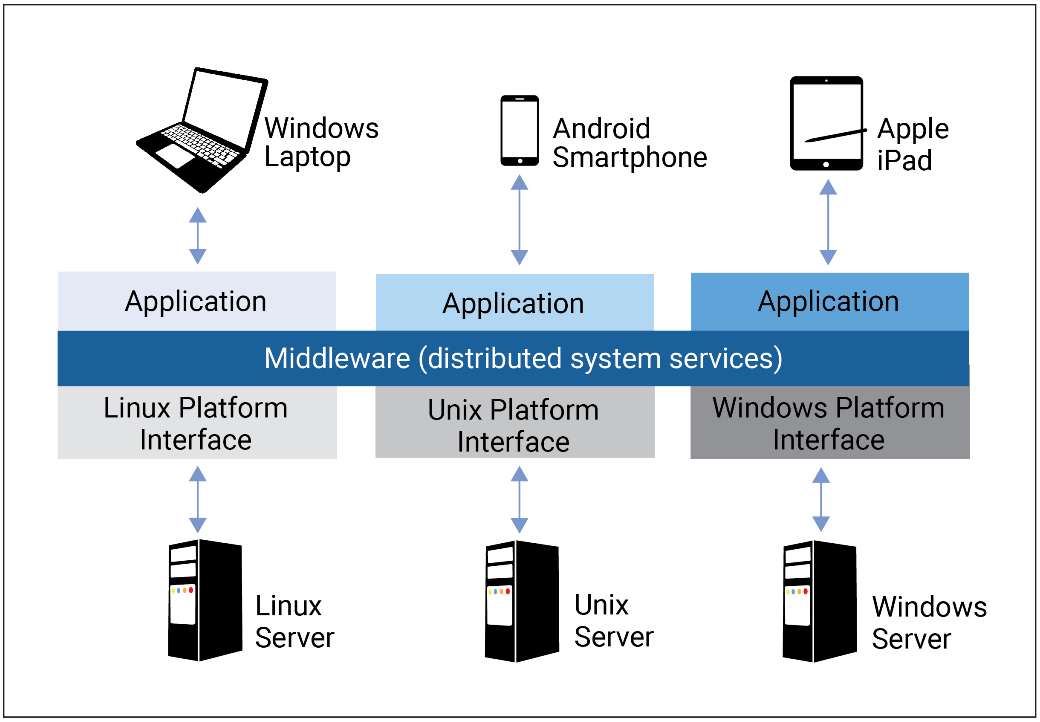


Figure 2-28: Middleware enables a variety of client platforms to link to a variety of server platforms.

facilitates the distribution of client-server applications across enterprise network locations and platforms without changing how users experience them (see Figure 2-28).

Communication services sometimes identified as middleware include data integration services, enterprise application integration services, enterprise service buses (ESBs), and client-server message passing services. On a basic level, these are all message passing services. Some of the message passing methods that middleware uses are identified in Table 2-12.

Table 2-12: Common forms of middleware message passing.

Message Passing Method	Description
Remote procedure call (RPC)	A remote procedure call (RPC) occurs when the client logic for a client-server application sends a request to a server at another network location to perform a procedure or function; this is sometimes called a subroutine call or a function call. When the server receives the request, it sends the required response back to the client. The client's processing of the application pauses while the server is executing the procedure or function and resumes when the server's response is received.
Message-oriented middleware (MOM)	MOM is a software infrastructure that supports the sending and receiving of messages between enterprise network components using a special program called a message broker. A client sends a message to the message broker, and the message broker forwards it to the appropriate server application. The message broker supports message passing in synchronous interactions and message queuing for asynchronous interactions. MOM uses reliable message queues to ensure message passing and provides supportive administrative and security services.
Object request broker (ORB)	An ORB is middleware that allows program calls to be passed from one computer to another computer over a network in a manner that provides location transparency via RPCs. ORBs promote interoperability among different vendor systems by enabling the systems to communicate with one another.

Appendix 2-3: SOA and Microservices

Many applications include services, and some consist almost entirely of services. As noted in section 2.3.5, services are self-contained pieces of software code that perform common business functions, and the functions that services perform produce identifiable results. When the same function is used in multiple applications, it makes sense to develop a service for it that the different applications can use. In SOA, services mirror and have names that reflect the real-world business activities, operations, and processes that they logically represent. Examples of services include "check customer account balance," "check inventory level," "create purchase requisition," "create invoice," and "submit payment."

Service-Oriented Architecture (SOA)

Services are typically designed to be continuously available and able to run on demand. Ideally, they can communicate with clients and other service consumers on demand. Basically, a service is self-contained reusable program code that can be used in multiple applications.

Because they are self-contained, services can be combined to create larger or new services. This means that some services are made up of other services. This ability to create applications from existing services can improve business application development flexibility and agility.

The self-contained nature of a service enables it to be independently revised without having to consider how the revisions might affect other services. This facilitates the development and deployment of new or improved service features and capabilities.

In SOA, services typically communicate with one another via an enterprise service bus (ESB). Messages are passed over the ESB to enable services to communicate and collaborate. Because an ESB, such as that depicted in Figure 2-29, is essentially a message-exchanging system that enables services to collaborate, it is sometimes considered to be a type of middleware.

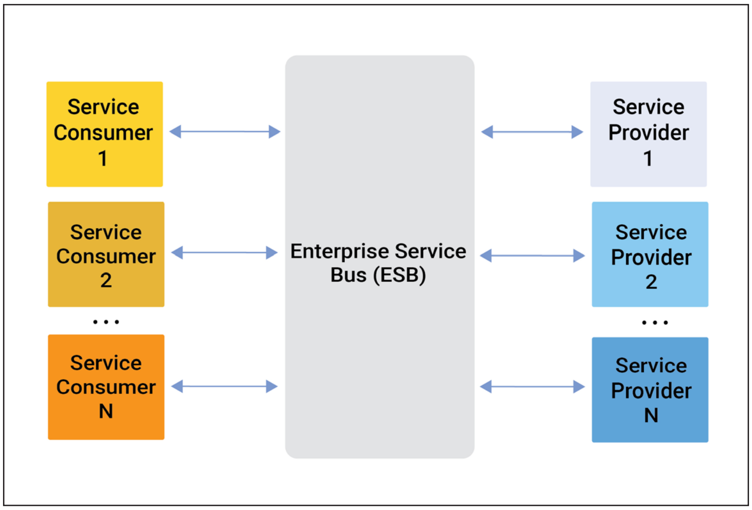


Figure 2-29: A high-level depiction of an enterprise service bus.

ESBs typically employ loosely coupled messaging systems that are platform independent. They often support both synchronous and asynchronous message passing. ESBs may also provide service binding and integration, service registries, service monitoring and management, service orchestration, load balancing, routing, protocol translation, failover, and security services.

Microservices

Like services, microservices are used to build distributed applications, especially cloud applications. Such applications are built as a collection of services that communicate through APIs. This approach separates an application's functions into modular, self-contained programs.

Microservices are a more granular form of services that perform basic and limited functions. They are frequently, but not always, implemented as containers. A *container* is a packaged bundle of applications or services that includes all dependencies, such as configuration files, libraries, and data repositories. This allows the application/service to be deployed easily and consistently regardless of environment.

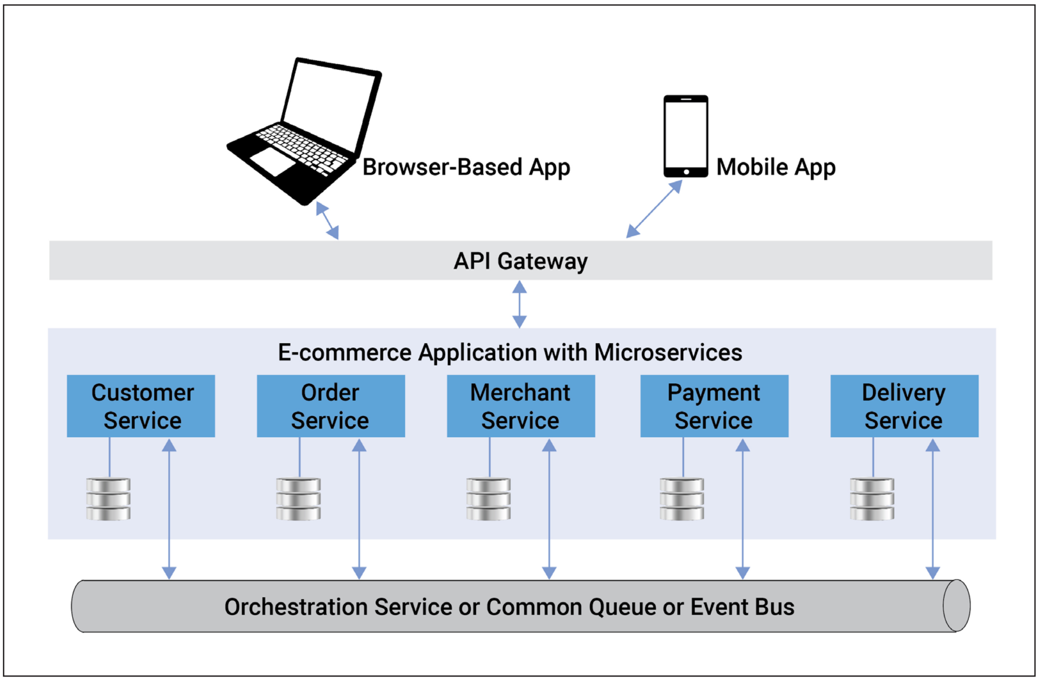


Figure 2-30: A cloud-based e-commerce application with microservices.

When composed of microservices, each function of an application operates as an independent application or service. This software architecture enables each service to be modified/updated without disrupting the application's other functions/services. It also enables each of the services to scale independently of the others. Because on-demand scalability is important for cloud services, it is easy to understand why microservices are popular for cloud applications.

Like services, microservices make it easier for programmers to create and maintain applications. Most microservices are designed for reuse and to have independence/autonomy from other microservices. They are also designed for *composability*, the ability to be easily combined with other microservices to create new applications. Composable microservices are usually published in a way that makes them easily discoverable and accessible to developers.

Applications composed of microservices usually feature loose coupling among the services. This enables service owners/developers to make changes to a microservice without affecting other collaborating services in an application. This also provides applications with a measure of fault tolerance; the failure of one service does not disrupt other services or the entire application—the application continues to run when one of its functions is temporarily unavailable.

As noted previously, today's distributed applications, including e-commerce applications, may consist of cloud microservices. Figure 2-30 illustrates how an online sales process with traditional e-commerce components might be deployed as a collection of microservices with all underlying infrastructure abstracted to a cloud interface.

Appendix 2-4: Business and the Cloud

Cloud computing is a popular option for businesses for multiple reasons, including cost savings, reliability, flexibility and scalability, resilience, and security. However, understanding businesses' attraction to the cloud begins with NIST's description of cloud computing's essential characteristics: broad network access, resource elasticity, measured service, on-demand self-service, and resource pooling, which are summarized in Table 2-13.

Table 2-13: Characteristics of cloud computing that are attractive to business subscribers.

Cloud Characteristic	Description
Broad network access	This refers to the use of standard mechanisms that enable a wide range of client device types, both thick or thin, to access cloud applications, services, and resources. It also means that users can access cloud resources from any location using any type of device.
Resource elasticity	This is the ability of subscribers to expand or reduce their consumption of cloud services in response to changing needs. A business that temporarily needs a large amount of server resources for a specific task can release these resources back to the cloud provider when the computing task is finished.
Measured service	This is the ability of cloud providers to monitor, control, and report resource usage. It provides transparency for the cloud services that subscribers use and their usage fees. Tracking these factors positions subscribers to adjust their contractual arrangements with cloud providers. Measured service enables providers to meter subscriber resource consumption levels and to automatically control and optimize the resources that subscribers need; this complements resource elasticity.
On-demand self-service	This is the ability of cloud consumers to unilaterally and automatically provision cloud services as needed without requiring the intervention of cloud providers. Business subscribers typically sign contracts with providers that specify a specific volume of cloud resources (per month). The ability to temporarily expand access to cloud resources means that additional nonpermanent resources are readily available to meet business needs.
Resource pooling	This refers to the ability of a cloud provider to pool its resources and to dynamically allocate them to multiple cloud consumers. A business subscriber is often a single “tenant” in a multi-tenant pooled resource arrangement, and participating businesses can access needed cloud resources less expensively than they could otherwise.

Comparing traditional IT models and enterprise networks to cloud models reveals other reasons for business adoption of cloud services. In traditional IT models and enterprise networks, business data centers are located on premises to provide employee access to the business’s data and applications. These resources enable a business to provide customized systems and client-server applications to meet its needs.

In traditional IT models, employees use business-supplied devices and connect to servers in the on-premises data center over an internal network. With an expanding workforce, the business may have to purchase additional data center servers and storage equipment and expand its network to support new users. The business is also responsible for deploying software upgrades and redundant hardware and support systems for resilience and security. The business has full control over how its systems are secured but is also responsible for all associated costs. Traditional IT models also require businesses to have an appropriately staffed and skilled IT department to manage their data centers and serve the needs of enterprise network users.

With the cloud, applications and data are off premises, and the business may have no idea where the servers and cloud computing resources used to support its employees are physically located. Businesses rent the servers and storage they need from cloud providers, often on a pay-per-use basis. This enables businesses to forgo investment in in-house servers and storage technologies, even when their workforces are expanding. Cloud providers handle software and hardware upgrades, and the associated cost is included in the monthly fees that the cloud subscribers pay. The availability of cloud computing resources positions businesses to reduce their on-premises infrastructure and the size of their IT support staff.

Cloud applications and services that employees use may be generic, but many allow some degree of customization to meet the various needs of subscribers. Users can typically access cloud resources 24/7/365 via personal devices, and it is no longer necessary for them to only use computers supplied by their employers. This facilitates remote work. The 24/7/365 access is enabled by resource pooling, metering, and elasticity and resilience that stem from virtualization and hardware and data center redundancy.

Table 2-14: Advantages of cloud computing relative to traditional IT models and enterprise networks.

Cloud Advantages	Description
Greater resilience	This stems from the distribution of applications and information among interconnected cloud servers that work as one. If a server fails, processing work shifts to another. There is little or no downtime or data loss when failover measures are used.
Greater elasticity	The greater elasticity of cloud resources contributes to consistent and resilient application performance.
Greater scalability and flexibility	This results from on-demand allocation of cloud resources to meet tenant needs. Increased workload can be accommodated effortlessly and is less expensive than that normally required by traditional IT infrastructure.
Lower administrative burden	This results from outsourcing tasks such as software licensing, hardware maintenance, software upgrades, security, and system monitoring to cloud providers. This is often accompanied by reduced on-premises IT staffing requirements.
Lower costs	This stems from pay-per-use arrangements with cloud providers that are comparable to those that businesses have with utilities. Relative to traditional IT infrastructure, there is less chance of having too little or too much infrastructure. Cost savings may also be realized through reduced downtime and reduced downtime-associated productivity losses.

Relative to traditional IT systems, cloud computing has greater elasticity and resilience, greater flexibility and scalability, and lower administrative burden and costs (see Table 2-14). The elasticity of cloud services contributes to greater consistency in application performance. In contrast, traditional IT systems and enterprise networks have limited capacity and resilience and are more susceptible to disruptions that cause downtime.

Appendix 2-5: CaaS, FaaS, BIaaS, and BDaaS

CaaS and FaaS

CaaS stands for *container as a service (CaaS)* and FaaS is *function as a service (FaaS)*. CaaS is often considered a subset of IaaS that has characteristics of both IaaS and PaaS. This explains its location between IaaS and PaaS in Figure 2-31. As noted in sections 2.3.5 and Appendix 2-3, containers are software that consist of microservices and all needed dependencies that are typically deployed as cloud-native apps.

Like PaaS, CaaS provides software (in this case containers) that subscribers can use to create applications. Apps developed with containers have everything they need to run in multiple environments, including cloud infrastructure from IaaS providers. Because containers are conducive to horizontal scaling (adding more machines to assist in processing the application), CaaS has characteristics that overlap with IaaS. Because security increases may be realized by isolating containers used by different subscribers, CaaS can be more secure than PaaS, while being slightly less secure than IaaS.

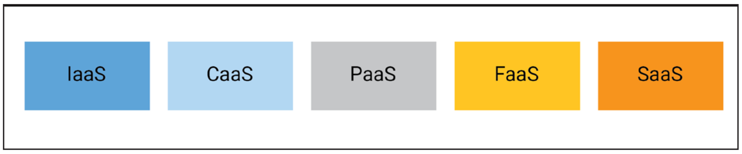


Figure 2-31: CaaS and FaaS relative to IaaS, PaaS, and SaaS.

FaaS is sometimes considered a subset of PaaS that has characteristics of both PaaS and SaaS, which explains its location in Figure 2-31. FaaS enables businesses to focus on application functionalities (microservices) without having to worry about the infrastructure required to run or support them. By enabling subscribers to develop, run, and manage application functionalities, FaaS provides a degree of application customization that is usually beyond that available in SaaS.

BIaaS and BDaaS

Businesses are also consuming an increasing variety of specialized cloud services that combine elements of SaaS, PaaS, and IaaS in innovative ways. *Virtual desktop as a service (VDaaS)* is an example. While it is obviously software (like SaaS), it often includes tools used to create new software (like PaaS). Hence, it may be considered a blend of SaaS and PaaS.

Business adoption of specialized services reflects the evolution and maturation of cloud computing. Cloud brokers and cloud aggregators have important roles in the provision of cloud services to businesses. *Cloud brokers* generally serve as consultants and/or intermediaries between subscribers and providers. *Cloud aggregators*, however, are like system integrators; they assemble and bundle cloud services into proprietary services. Combining/bundling cloud services to meet business needs has resulted in products such as *business intelligence (BI) as a service (BIaaS)* and *Big Data as a service (BDaaS)*.

Like traditional BI tools used with data warehouses (which are described in Chapter 3), BIaaS extracts data from internal and external systems, organizes the extracted data in a data warehouse (DW), and provides user-friendly business intelligence tools to analyze DW data. Ideally, it is an easy to implement and affordable cloud system that provides end-to-end BI solutions. Usually, BIaaS is deployed in a hybrid or VPC cloud environment to protect sensitive organization data while also providing access to public data sources.

Data scrubbing/cleansing and other extract, transform, load (ETL) processes associated with traditional DWs are typically included in BIaaS, but these processes may be more rudimentary and limited in scope than those in traditional DWs to enable the cloud-based DW to be quickly populated with data. Tools to prepare data for the formats required by specific BI tools are also typically included in BIaaS offerings.

BIaaS solutions make sense for organizations with tech-savvy business users capable of creating their own reports and dashboards and for organizations with finite IT resources who are stretched too thin to create and maintain BI reports/dashboards. They are also attractive to organizations in which data/information is hard to quickly access or leverage and to businesses interested in providing easy to deploy BI solutions to business units.

Data analytics as a service (DAaaS) solutions are like BIaaS but provide analytics tools that can usually be applied to data sources outside a DW. DAaaS may include a DW, but this is not required. DAaaS (or *analytics as a service [AaaS])* offerings enable data scientists, developers, and business users to rapidly deploy analytics applications and are typically less expensive than those in traditional on-premises deployments. DAaaS and AaaS enable subscribers to execute scripts and queries and to create reports, data visualizations, and dashboards.

Big Data as a service (BDaaS) uses cloud-based, distributed computing technologies that enable subscribers to store, manage, process, and analyze large data sets (Big Data). As noted in Chapter 1, businesses are increasingly leveraging Big Data to gain useful insights for improving their operations and/or achieving and maintaining competitive advantage. BDaaS solutions often leverage Hadoop clusters and the Hadoop Distributed File System (HDFS) to provide horizontal scalability; the Hadoop aspects of these solutions are described more fully in Chapter 3.

As Figure 2-32 illustrates, BDaaS solutions include Big Data platform, infrastructure, and analytics components. Big Data platform components rely on cloud-based data management solutions such as *data as a service* and *database as a service*. Big Data infrastructure components rely on cloud services such as *storage as a service* and *computing as a service*. Big Data analytics software may include programs/scripts written in R, MapReduce, Apache Spark, and other languages designed for processing large data sets. It may also include specialized Big Data analysis tools.

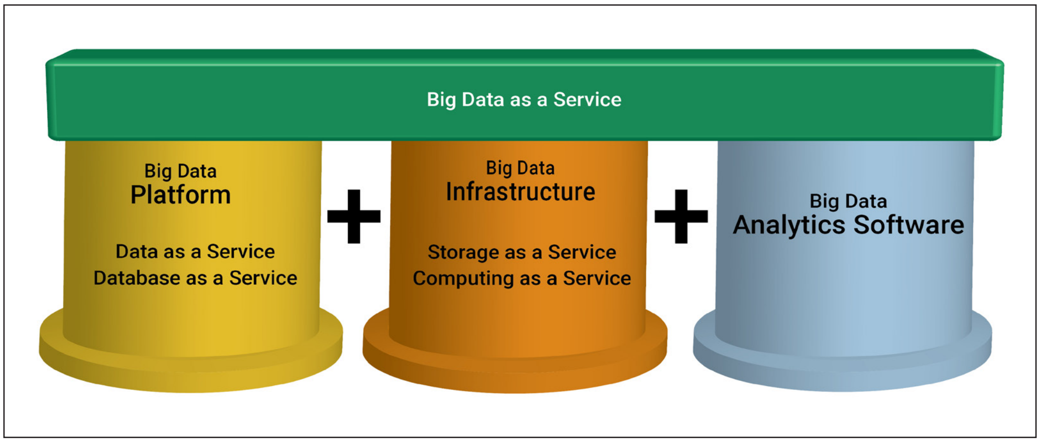


Figure 2-32: BDaaS components.

Table 2-15: BDaaS versus traditional Big Data storage and processing.

Big Data as a Service	Traditional Big Data
On-demand scalability through a combination of cloud computing and distributed architecture	Scalability of storage and processing achievable via a distributed architecture
Virtualized data storage on a distributed platform	Data storage on HDFS or a distributed platform
Structured and unstructured data in a cloud environment	Structured and unstructured data
On-demand computing power with advanced analytics functions	Advanced analytics functions
Combines domain-specific out-of-box algorithms and custom-coded analytics	Custom-coded analytical capabilities
Ubiquitous accessibility	Limited accessibility

Table 2-15 summarizes some of the most important distinctions between BDaaS and traditional Big Data storage and processing. Relative to traditional approaches, BDaaS provides greater processing and storage scalability. BDaaS provides Big Data processing flexibility while also enabling subscribers to run customized processing routines.

BIaaS and BDaaS and their associated tools and services will continue to evolve and mature. As they do, they are likely to attract more business subscribers. This reflects the reality that business-oriented cloud services are increasingly important aspects of enterprise networks.

CHAPTER 3

Appendix 3-1: Big Data Processing Platforms

Big Data processing platforms and analytic tools continue to evolve. Two of the more common Big Data processing platforms, Hadoop clusters and SAP HANA, are briefly described here.

Hadoop Clusters

Hadoop is an open-source software project that includes a distributed file system (the *Hadoop Distributed File System [HDFS]*) that stores large data sets across multiple computers. The HDFS provides data replication and load balancing capabilities that move data among the different machines to maximize performance consistency and reliability.

Hadoop clusters are used to manage numerous types of Big Data processing, including clickstream analysis, marketing analytics, image processing, web crawling, and general archiving. They are used by most Fortune 50 companies, including Facebook and Yahoo!. The servers used in Hadoop clusters typically have Linux operating systems to facilitate the ability of Hadoop's software components to work directly with the underlying hardware.

As illustrated in Figure 3-21, there are three major machine roles in Hadoop clusters: head nodes, worker nodes, and client nodes. It is the network of head and worker nodes that execute the processing tasks across the HDFS. The roles of the *client nodes* are to load data into the cluster, submit data processing tasks, and retrieve or view the results of the data processing jobs.

Head nodes oversee two critical functions: storing lots of data and running parallel computations on it. These functions are carried out by different head node services: the NameNode service and the JobTracker service. Typically, these services run on separate machines. There is also a *Secondary NameNode* service that maintains a backup of NameNode data.

The *NameNode* service oversees and coordinates the HDFS and keeps track of what files are currently being processed. The *JobTracker* function oversees and coordinates the parallel processing of the data, which typically uses the MapReduce algorithm.

Most machines in Hadoop clusters are worker nodes. The *worker nodes* run two important services (DataNode and TaskTracker) that do the actual work of storing and processing the data as directed by the head nodes. The *TaskTracker* program receives processing instructions from the JobTracker, and the *DataNode* program receives instructions from the NameNode about the data to be stored for processing.

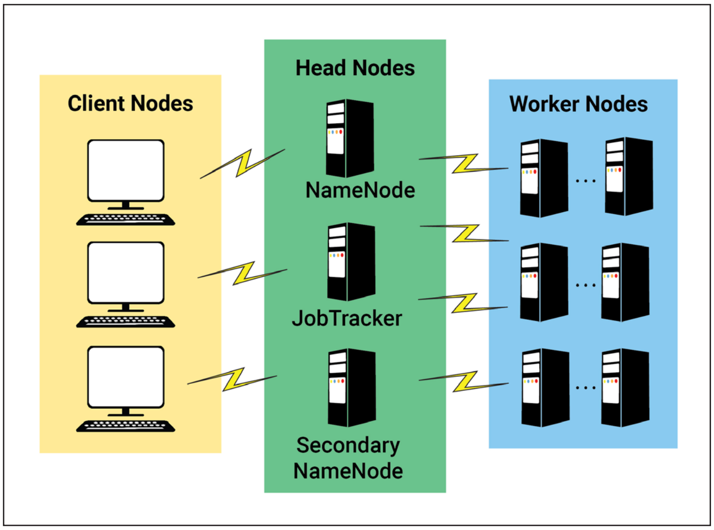


Figure 3-21: Conceptual diagram of a Hadoop cluster.

In smaller Hadoop clusters (those with fewer than 50 nodes), a single server is used for both the JobTracker and NameNode services. In medium and large clusters (consisting of hundreds or thousands of machines), the JobTracker function is typically performed on one head node server while the NameNode functions are performed on one or more other machines.

Hadoop clusters typically employ MapReduce to process large data sets. *MapReduce* is a programming model for processing (and generating) large data sets in a computer cluster using a parallel, distributed algorithm. Although it is not the only programming algorithm used in Hadoop clusters, it is among the most common. When MapReduce is used with a distributed file system such as HDFS, the same computer cluster can process diverse data types (structured and unstructured) simultaneously.

Basically, MapReduce enables a cluster to be a Big Data processing platform by splitting the data into chunks that are mapped and distributed to different worker nodes by Hadoop's JobTracker and NameNode services. The worker nodes process the data in parallel, and the results they produce are merged/aggregated (reduced) into a single set of results. MapReduce programs, submitted by client nodes, are written in languages such as C++, Java, Python, and Ruby.

Apache Spark is another large data set processing tool that interfaces with HDFS and other distributed storage file systems. Like MapReduce, *Apache Spark* is an open-source tool that enables entire clusters to be programmed for parallel data processing. Unlike MapReduce, it enables in-memory access to HDFS data. This makes it faster than MapReduce for many Big Data processing tasks (including interactive queries) and better suited for machine learning applications. Apache Spark has the largest open-source Big Data community, with more than 1000 contributors from over 250 organizations.

SAP HANA

SAP HANA is an in-memory database engine developed by SAP that is used for real-time analysis of large data sets that reside entirely in memory. It is a high-capacity, column-oriented, in-memory relational DBMS that can hold up to a petabyte of data in memory.

Being both column-oriented and in-memory contributes to SAP HANA's performance. Traditionally, row-oriented relational databases have been used to store business data, and Structured Query Language (SQL) has been used to retrieve and manipulate the data.

When SQL is used to query a row-oriented database, the DBMS must go row by row in database tables to retrieve the appropriate data. This can be time-consuming, especially when database tables have millions of rows. When data is stored in columns rather than rows, a DBMS can more precisely and quickly access the data it needs for a query because it is no longer necessary to scan and discard unwanted data in table rows.

SAP HANA is physically implemented as a data center appliance or horizontally scalable cluster that primarily functions as a database server. However, it also has extract, transform, load (ETL) capabilities that can be used to cleanse and update data warehouse data.

SAP HANA is also used as an application server to run SAP ERP and other ES applications. ES applications perform very well on SAP HANA because their entire databases can be run in-memory.

Some organizations use SAP HANA to combine their data warehouses (DWs) and operational databases in-memory to enable real-time analytics that compares current to historical performance. SAP HANA is also used for predictive analytics, prescriptive analytics, text analytics, natural language processing, and streaming analytics (including analysis of social media streams). When integrated with Apache Spark, it provides a robust in-memory data architecture that can be used for real-time interactive analysis and applications that simultaneously process both structured and unstructured data.

Appendix 3-2: Logical Data Repositories

Relational Databases

Data warehouses (DWs) are best suited for clean, structured data from relational databases and other structured data sources; their data is arranged for immediate use and analysis by BI and business analytics tools. Data lakes are large pools of raw data from structured, semi-structured, and unstructured data sources that may be analyzed in the future.

A *relational database* is a collection of integrated records about one or more entities. An *entity* is an object, thing, person, place, or item of interest whose data is captured and placed in one or more database tables. For businesses, entities often include suppliers, customers, employees, materials, products, facilities, transactions, inventory items, etc.

In relational databases, data about an entity is stored in records in one or more tables (relations), such as those illustrated in Figure 3-22. Each record (row) in a relational database table includes columns (fields) that contain information about the attributes of the entity.

Rows (records) in the table represent instances of the entity. Each row (record) also includes a unique key (primary key) to identify it. For example, in Figure 3-22, *customer_id* is used to uniquely identify customer instances (rows) in the Customer table. As this figure illustrates, each

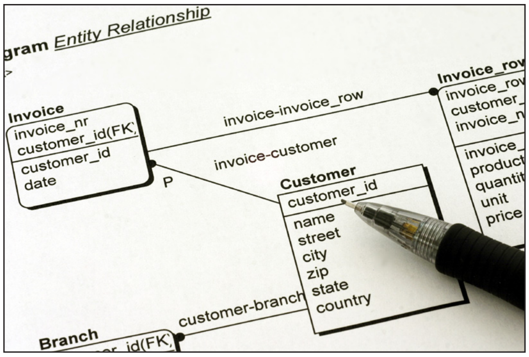


Figure 3-22: An example of relational database tables.

Source: suingu/iStock. Stock photo ID:182660832

customer record in the Customer table also includes the customer's name, street address, city, zip code, state, and country.

Some tables in a relational database, such as the Invoice table in Figure 3-22, include both a primary key and one or more fields that are primary keys in other tables. When a field in one table serves as the primary key for another, it is called a *foreign key*. Foreign keys enable data in different tables to be linked. In Figure 3-22, for example, the customer_id is a foreign key in the Invoice table, which enables data in the Invoice table to be linked to data in the Customer table.

Like other logical data organizations, relational databases include metadata. The metadata for a relational database includes the list of table names, table sizes (often expressed in megabytes, gigabytes, etc.), the number of records in each table, the list of fields (column names) included in the tables, and the type of data (e.g., text, numbers, dates) included in fields.

Relational databases have well-defined data models that specify how the contents of their tables relate to one another. Data models facilitate the use of relational databases in business analytics and BI. ERP and other ES databases are relational databases that may include hundreds or thousands of tables, have complex data models, and reach sizes that exceed several terabytes.

NoSQL Databases

Table 3-12 summarizes other characteristics of NoSQL databases that are attractive to businesses.

Queries and searches are usually faster and more efficient on relational databases than on NoSQL databases because relational databases include structured data and have well-defined data models. NoSQL databases are designed for flexibility and scalability rather than query efficiency. Often, extra processing is needed to run queries on NoSQL databases.

Table 3-12: Factors driving business adoption of NoSQL databases.

Characteristic	Description/Example
Simplicity	Saving a purchase order (PO) as a single document can be simpler than storing PO details in joined relational database tables.
Easy capture of various types of data	A NoSQL database can easily and quickly accommodate new types of unstructured and semi-structured data and is not disrupted when new data is added. For example, new documents with different numbers of attributes can be commingled with documents that have more or fewer attributes.
Query speed	Queries on semi-structured and unstructured data are much faster on NoSQL databases than relational databases, where queries on such data might be impossible.
Cost	NoSQL databases are usually processed by clusters of cheap commodity servers, while relational database management systems (RDBMSs) often rely on expensive and/or proprietary servers and storage systems. In addition, many NoSQL databases are open source and free; licenses for RDBMSs can be expensive.
Horizontal (not vertical) scaling	NoSQL is less expensive to scale. When more processing capacity is needed, inexpensive commodity machines can be added instead of having to invest in bigger and more powerful machines (vertical scaling). NoSQL databases work best on computer clusters. Relational databases work best when run on single computers. NoSQL databases automatically spread data across server clusters without requiring additional programming or requiring applications to be aware of the size or composition of the server pool.
Availability	The ability to add (or remove) servers without disrupting application processing enhances the application's availability.
Data replication	Most NoSQL databases support data replication (storing multiple copies of data across clusters or data centers). This contributes to resilience and business continuity. It also facilitates disaster recovery efforts.

In-Memory Databases

As noted in section 3.2.4, an *in-memory database (IMDB)* is capable of storing large data sets, like an entire ES database and/or DW, in a computer's main memory. This eliminates the need (and required time) to transfer database data in pieces from a storage device and bring it into main memory for processing. Memory limitations in computers traditionally used for database processing have contributed to a business preference for computer clusters (Hadoop) and SAP HANA servers for Big Data processing.

IMDBs are well suited to applications that depend on rapid response times and real-time data management, such as securities trading, banking, e-commerce, geospatial processing, processing sensor data streams, and machine learning. They are also useful for real-time management of enterprise network routers and switches. IMDBs are sometimes called real-time databases (RTDBs) or main memory databases (MMDBs).

Unlike traditional relational databases, some IMDBs, such as SAP HANA, have column-oriented (columnar) rather than row-oriented data models. A *columnar database* stores data in columns rather than rows, and this enables individual data elements to be accessed in groups instead of individually row-by-row. Such access accelerates query processing by ignoring, rather than filtering out, data that doesn't apply to a particular query. For example, a "customer name and zip code" query would not have to go through row-by-row relational data tables to retrieve just those two fields while discarding other data, such as street, city, state, and country.

Columnar databases are increasingly common in DWs, the traditional data repositories used for BI and corporate analytics. Extracting data from tabular data sources and loading it into column-oriented structures can make data warehouses easier to query and process. For this reason, columnar structures are considered the future of BI.

Data Warehouses

A *data warehouse (DW)* includes data that is integrated, time-variant (historical), and nonvolatile. DWs can be physically implemented in storage area networks (SANs) or data centers (discussed in section 3.4), but, like databases, they are logical data collections and arrangements.

DWs have traditionally been used for BI and analytics. Often, their contents are "mined" to identify important patterns and trends across time.

Several important concepts associated with DWs are illustrated in Figure 3-23. For example, DWs include data that has been extracted from multiple sources, both internal and external. The extracted data is cleansed, combined, and loaded into the DW using extract, transform, load (ETL) processes. DWs use metadata to describe their contents, and logical subsets of their data called data marts are often created. Data cubes are also commonly created and are used for data mining and analytics.

Traditionally, DWs have been created from data extracted from structured data sources. Today, however, DWs are also being used to house unstructured and semi-structured data, especially by businesses interested in Big Data processing.

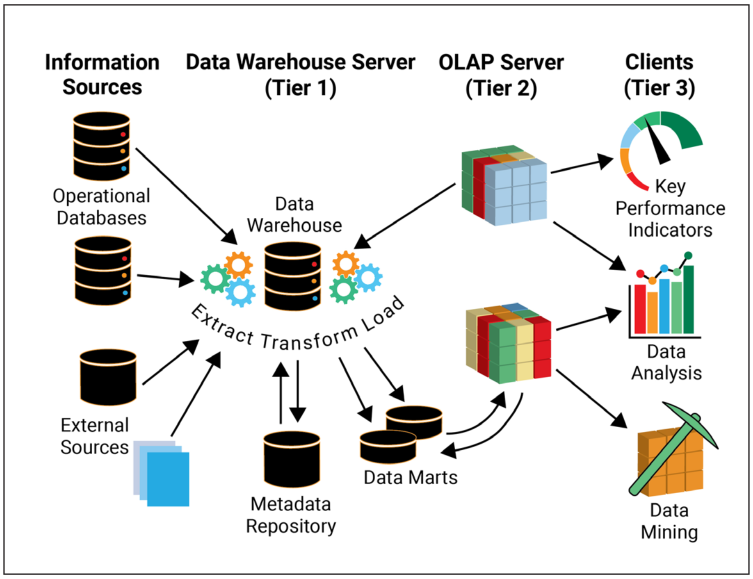


Figure 3-23: Data warehouse concepts.

Data Lakes

Data lakes are among the newer, and larger, logical collections of data in enterprise networks. Data in a data lake comes from disparate sources and often includes a mix of structured, semi-structured, and unstructured data formats. Businesses create data lakes as repositories for data they do not currently need or plan to immediately analyze.

Like a natural lake, a data lake can be fed by multiple continuously flowing streams, in this case, real-time flows of structured, unstructured, and semi-structured data. Data lakes enable businesses to amass large pools of unrefined raw data for later processing and analysis, including data with unidentified or unknown business value. Data lakes may be fed by streams from both relational and NoSQL databases; they may also be sources of data for both types of databases when decisions about processing data lake data have been made. Data lakes are often attractive destinations for audio, image, and video files, as well as IoT device and social media data.

A data lake may be created to collect data for an entire organization, but some are created for data that is specific to certain parts of an organization. The term *data ocean* is sometimes used to refer to all unfiltered and unprocessed data that supports the entire business, especially in organizations that create data lakes or data pools to collect raw data intended to support specific parts of the business. The term *data reservoir* is also used to distinguish between raw data and partially refined data (data that has been partially prepared for use or secured).

Despite the varying terminology, data lakes are commonly contrasted with data warehouses. Table 3-13 summarizes some of the more frequently identified differences between data lakes and data warehouses. A comparison of the relative advantages of data lakes and data warehouses is provided in Table 3-14.

Major cloud providers, including AWS, Azure, and Google Cloud, offer managed data lake services. These services are relatively inexpensive, are highly scalable, and can be integrated with other cloud storage services and/or applications, such as BIaaS (BI as a Service). Data lakes invariably rely on cloud storage. DWs may use cloud storage, on-premises storage, or both.

Table 3-13: Some of the major differences between data lakes and data warehouses.

Dimension	Data Lake	Data Warehouse
Data filtering	Contains raw, unfiltered data in its native formats	Contains filtered, cleansed data that has been transformed/prepared for business use
Data use	Designed to hold data for future use	Designed for immediate processing, analysis by BI and business analytics tools, and strategic decision-making
Primary users	Data scientists	Business professionals
Metadata amount	Limited metadata; often later added by data scientists to facilitate queries and analyses	Large amounts of metadata, including indexes and data dictionaries
Accessibility	Highly accessible and easy to update	ETL processes make DWs less accessible and complicated or costly to change
Storage location(s)	Cloud data centers	On-premises and/or cloud data center
Scalability	Can scale to several petabytes	More limited scalability; few scale beyond several terabytes
Performance	Less efficient; large data pool size constrains processing speed; limited metadata can also affect performance	More efficient; data preparation, metadata, and smaller size facilitate faster processing and analysis
Security	Because they contain large amounts of raw data flowing in from multiple sources, they have more information security challenges than data warehouses.	ETL and data-cleansing processes contribute to more transparent and well-established information security and data protection controls.

Table 3-14: Advantages and disadvantages of data lakes and data warehouses.

	Data Lake	Data Warehouse
Advantages	• Good for storing and analyzing data from diverse sources where initial data cleaning is problematic • Enables data from IoT devices to be saved for later analysis • Capacity facilitates integration with machine learning applications • Supports storage and analysis of social media and mobile device data	• Ability to analyze relational data, including that from ERP, CRM, and other ESs • Easy to integrate with structured data sources, such as relational databases and spreadsheets
Disadvantages	• Unrefined, low-quality data can create "data swamps." • Large quantities of unused data can be "data graveyards."	• Too much security can limit accessibility, productivity, and collaboration. • Improper cleansing or preparation can undermine data quality and BI results.

Centralized versus Distributed Databases

A **centralized database** is a database that is located and maintained at a single location in a network. Its users access the database via network connections.

A **distributed database** is a collection of separate databases at locations that are managed by a DBMS in a way that makes it appear as a single database to users. Figure 3-24 provides a conceptual depiction of a distributed database.

Some of the major differences between centralized and distributed databases are summarized in Table 3-15.

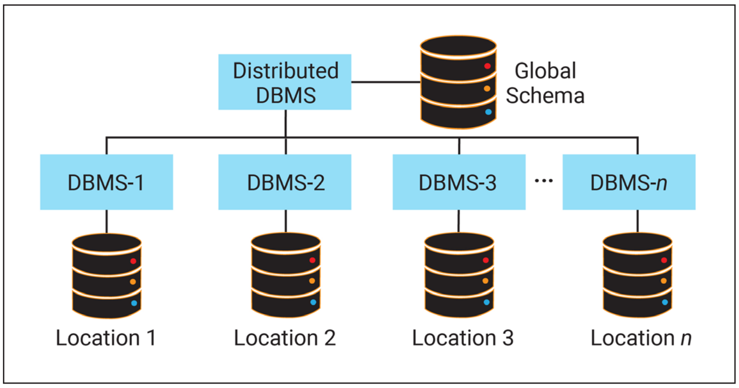


Figure 3-24: A distributed database is a collection of databases at different locations.

Replicated and Partitioned Databases

Most distributed databases in today's enterprise networks are replicated databases. In a **replicated database**, the database is copied to

Table 3-15: Some of the major differences between centralized and distributed databases.

Centralized Database	Distributed Database
A single database at a single network location	A collection of databases at different network locations that appear to users to be a single database
Easier management, modification, and backup	Management, modification, and backup are more challenging; synchronizing the databases at different locations adds complexity.
Data access can be slow during peak usage intervals because a single database entity serves all user requests.	Data access is typically fast because users access the database that is closest to them and do not compete with all other users for data access.
Little data inconsistency or redundancy	Data synchronization across network locations is needed to minimize inconsistencies in the databases.
Single point of failure	Greater availability. If a network location is offline, users can access databases at other locations over network connections.
Less costly than a distributed database	Can be very expensive
Easier data governance	Data governance can be complex.
Increasing database volume may require vertical scaling.	Horizontal scaling may be possible to accommodate data volume increases.
May be easier to secure the data using strong data safeguards	Business continuity may be better because a security breach at one location may have minimal effects on other locations.

two or more network locations. The key advantages of this arrangement are availability, resilience, and business continuity. If a database server, data storage equipment, or communication link failure occurs at one location, users can still access the database at another location.

Real-time replication is often used for databases for transactional systems such as ERP systems. This essentially means that every transaction is simultaneously saved to storage devices at two or more network locations. *Near–real-time replication* (updating in small batches with short time lags [5 to 15 minutes]) or *deferred replication* (updating in larger batches at longer intervals than near–real-time replication) is suitable for some business applications. When either near–real-time or deferred replication is used, temporary data inconsistency across locations is likely.

Some distributed databases are partitioned databases that distribute the database's data across partitions (nonoverlapping segments) at different network locations. Typically, each partition is a nonoverlapping subset of the database. NoSQL databases that store several different types of unstructured data on different servers are sometimes called partitioned databases.

Blockchain Databases

A **blockchain** database is shared by a network of computers. Because it can be viewed and used by all computers in the network, its logical and physical structures are difficult to untangle.

As Figure 3-25 Illustrates, new blockchain records are bundled together in blocks and added to the chain (the database) one block at a time. The basic components of a blockchain are records, blocks (bundles of records), and the chain (the collection of linked blocks).

Each new block includes a hash code of the previous block that links the new block to the previous one. The hash code links make the blockchain tamperproof. Attempts to modify the

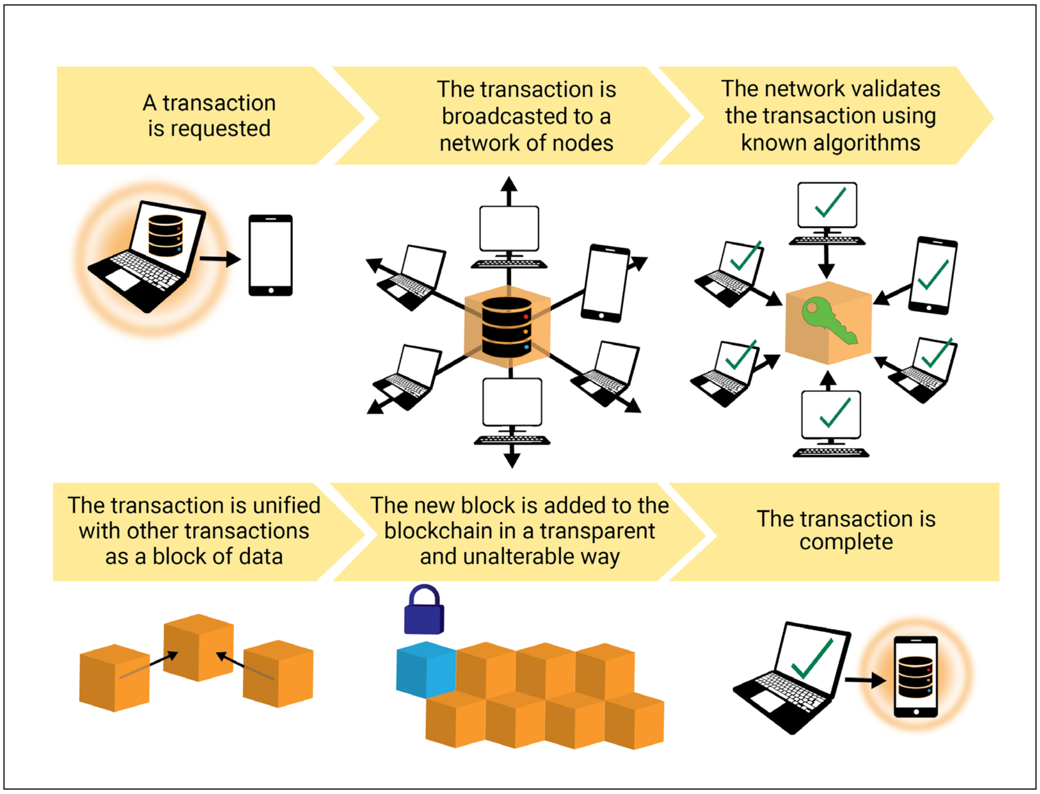

A transaction is requested
The transaction is broadcasted to a network of nodes
The network validates the transaction using known algorithms
The transaction is unified with other transactions as a block of data
The new block is added to the blockchain in a transparent and unalterable way
The transaction is complete

Figure 3-25: An example of how new data is added to a blockchain.

contents of a block make it necessary to change all previous blocks in the chain. Modification attempts would not go unnoticed by the network of trusted computers that continuously monitor the blockchain to ensure that all copies of it are the same. Such persistent monitoring helps ensure the integrity and security of the database.

Basically, a blockchain is a digital ledger that is transparent and accessible to all computers in the network. It is a list of transactions that is constantly being updated. It is supported by a private or public P2P (peer-to-peer) network of computers that uses the same process (algorithm) to verify each transaction. Processing work is distributed across the network to ensure that there is no single point of failure.

Many people recognize blockchain as an underlying technology for Bitcoin. However, businesses are using blockchain for non-crypto applications. The financial services industry is moving toward using blockchain for online banking, funds transfers between banks, and securities trading. Healthcare views blockchain as having the potential to provide secure and validated updates to patient medical histories. Blockchain is also being used to transparently track the movement of materials and products through supply chains.

Appendix 3-3: Data Centers

Some of the major differences between data centers at different tiers are summarized in Table 3-16.

Tier 1 data centers have the simplest infrastructure, while Tier 4 and 5 data center infrastructures are the most complex and have the greatest amounts of redundant components. Data centers at a given tier include the required components of all the tiers beneath them. Data centers at Tier 3 and above are considered fault-tolerant; they enable any data center component to be maintained or replaced without disrupting data center operations.

Table 3-16: Some of the major differences between data center infrastructures at different tiers.

Tier	Required Components	Required Redundancies	Maximum Annual Downtime	Guaranteed Annual Uptime
1	Uninterruptible power supply (UPS) for power sags, spikes, outages; dedicated area for IT systems; dedicated (not shared) cooling system; an electricity generator for power outages	None; but may have backup cooling system and/or generator	28.8 hours	99.671%
2	All Tier 1 components; multiple electricity generators; energy storage (backup batteries); chillers, cooling units, and pumps; heat rejection equipment; fuel tanks and fuel cells for generators and energy storage	Partial cooling redun—dancy and multiple power redundancies	22 hours	99.971%
3	All Tier 2 components; must include enough redundancy to enable any equipment to be maintained or replaced without requiring a data center shutdown; at least 72 hours of exclusive power for power outages	N+1 fault tolerance (amount required for operation and a backup for all required components)	1.6 hours	99.982%
4	All Tier 3 components; no single points of failure; a Tier 4 data center has twice the infrastructure as a Tier 3 data center; all IT equipment must have fault-tolerant power design; 96-hour power outage protection	2N+1 fault tolerance (twice the amount required for operation and a backup for all required components); full redundancy	26.3 minutes	99.995%
5	All Tier 4 components; must be able to run without water; permanently installed energy monitors; securable server racks; must run on local, renewable power sources	Same as Tier 4 but must include fault tolerance for the additional Tier 5 components	Must equal or exceed Tier 4	Must equal or exceed Tier 4

Sources: Uptimeinstitute.com; phoenixnap.com

As Table 3-16 illustrates, data centers typically include numerous redundant components, including Internet connections, power grids, wiring, networking devices, and security controls (see Figure 3-26). Some businesses prefer on-premises data centers, and others choose to be tenants in cloud data centers. For tenants, the tier level of provider data centers may be an important factor when selecting a cloud storage provider or providers.

Today, it is common to save data to two or more storage devices in a data center to ensure its accessibility should a storage device fail. Some cloud providers guarantee that tenant data will be saved to two or more data centers at different locations, thereby using data centers as backups to one another.

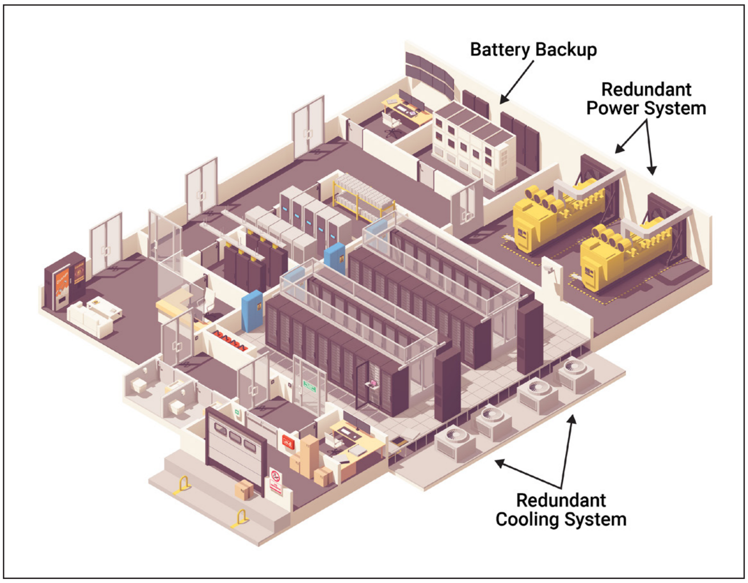


Figure 3-26: Example of redundant power and cooling systems in a data center. **Source**: tarras79 Stock illustration ID:113556313

Large data centers that are industrial-scale operations can consume as much electricity in a day as a small town. As a result, access to reliable and/or inexpensive electricity can be a determining factor for data center locations.

Some data centers have been deployed to mine cryptocurrency. These run continuously, consume a lot of electricity, and make a lot of noise—continuously. As a result, deployments of these data centers in areas where neighbors (including wild furry ones) prefer peace and quiet are resisted.

CHAPTER 4

Appendix 4-1: OSI and TCP/IP Model Similarities and Differences

As you can see in sections 4.2 and 4.3, the OSI and TCP/IP models have commonalities, some of which are summarized in Table 4-7.

As Table 4-7 illustrates, the OSI model is more generic and encompassing than the TCP/IP model, while the TCP/IP model more closely reflects the realities of today's Internet-connected enterprise networks. Standard-setting organizations use these models to develop standards for networking equipment, software, and protocols. The standards that are developed are used to create and enhance standards-compliant networking technologies and have inspired competition among technology manufacturers, including price competition.

Competition is especially intense among manufacturers of Layer 3 (Network), Layer 2 (Data Link), and Layer 1 (Physical) networking equipment. The availability of routers (Layer 3) and switches (Layer 2) from numerous vendors keeps their prices reasonable and affordable. Routers, switches, and cabling (Layer 1) are commodity items, and businesses negotiate attractive prices to lower their overall networking expenses.

Because these models and their associated standards also provide guidance to manufacturers of servers, storage hardware, and client devices, businesses typically have a wide range of vendors and products to choose from. In most instances, comparable standards-compliant products from different vendors are interchangeable and have no appreciable difference in how they or the network perform. This puts businesses in the driver's seat when equipment refreshes are needed or desired.

Table 4-7: Some important commonalities and differences between the OSI and TCP/IP models.

Commonalities	Differences
Both are logical models.	The OSI model is a generic model that is universal in nature. It can be used in different types of networks to guide how communication should occur. The TCP/IP model is compatible with current Internet architecture and is more limited in the types of communication it can address; however, it is well suited to enterprise networks that include Internet and cloud components.
Both are reference models.	TCP/IP is grounded in client-server application communication models in which a user device (a client) is provided a service or network resource by another computer in the network (a server). The OSI model encompasses both client-server applications and applications that are not client-server. However, TCP/IP is widely used in today's networks, which makes it practical to develop systems using the TCP/IP model as a background.
Both are nonproprietary models.	The OSI reference model was developed as a protocol-independent framework, and protocols have subsequently been mapped to its layers. Protocols in the TCP/IP model were developed prior to the TCP/IP model; this means that the TCP/IP model, while nonproprietary, is a protocol-dependent and protocol-oriented framework.
Both provide frameworks for defining and implementing networking standards.	The TCP/IP model exemplifies how the OSI model can be implemented using Internet protocols.
Both divide the network communication process into layers.	The OSI model includes Presentation and Session layers; neither of these are included in the TCP/IP model.
Their lower three layers share common names and are responsible for comparable networking functions and tasks.	TCP/IP's Transport layer includes the functions and tasks performed by the Transport layer in the OSI model as well as some of the functions and tasks performed by the OSI model's Session layer (see Figure 4-9).
Their layers provide approximately the same communication functionalities.	The Application layer in the TCP/IP model incorporates the functions performed by the Application and Presentation layers of the OSI model and some of the functions performed by the Session Layer (see Figure 4-9).
Both provide frameworks for creating and implementing network devices.	The OSI model provides a clear separation of the concepts of services, interfaces, and protocols, whereas the TCP/IP model does not. This makes the TCP/IP model less suitable and useful than the OSI model for describing and assimilating new technologies (or types of networks) within the framework.
They enable network hardware vendors to make devices and components that can coexist and interoperate with those made by other manufacturers.	TCP/IP is compatible with all operating systems, which facilitates communication among different devices and systems. The OSI model primarily helps to standardize routers, switches, and other networking hardware, which contributes to their interoperability.
Both facilitate network troubleshooting and problem resolution.	Because it has more layers, the OSI model typically provides superior guidance for troubleshooting and debugging network issues.

Appendix 4-2: Routing Protocols

Routing Protocols

Routing protocols define the rules that routers use to enable internetworking between source and destination networks. They specify how the routing tables that routers use are updated to determine where to send the packets they receive. Routers are responsible for moving information from a source to a destination, but routing protocols control how routers select the best route or path to the destination network.

Although routing protocols determine how Network layer packets are routed across networks, most are located at other layers in the TCP/IP model (see Figure 4-39). Those closest to the Network layer include OSPF, IGRP, EIGRP, and IS-IS. Routing protocols used by routers inside an organization's network are known as *interior gateway protocols*, and those used by routers that send/receive packets from the networks of other organizations are called *exterior gateway protocols*.

A routing protocol falls into one of two major categories—distance vector or link state—based on the approach used to update the routing tables; these are described in Chapter 5. Table 4-8 summarizes some of the key characteristics of the protocols included in Figure 4-39.

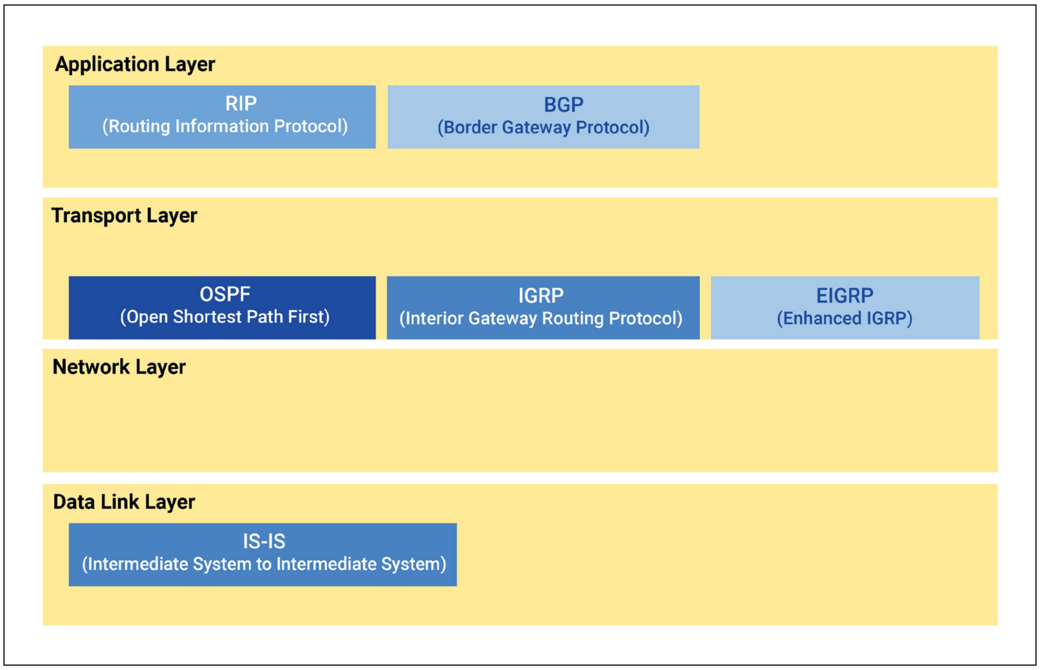

Application Layer
RIP
(Routing Information Protocol)
BGP
(Border Gateway Protocol)
Transport Layer
OSPF
(Open Shortest Path First)
IGRP
(Interior Gateway Routing Protocol)
EIGRP
(Enhanced IGRP)
Network Layer
Data Link Layer
IS-IS
(Intermediate System to Intermediate System)

Figure 4-39: Examples of routing protocols and their locations in the TCP/IP model.

Table 4-8: Key characteristics of routing protocols used in enterprise networks.

Routing Protocol	Key Characteristics
Routing Information Protocol (RIP)	RIP uses UDP for router table update messages. UDP port 520 is reserved for RIP. RIP has two versions: RIPv1 and RIPv2. RIPv1 uses UDP broadcasting to inform all other routers of updates made to a router's routing table; the default value for broadcasts is 30 seconds. RIPv2 uses UDP port 521 to multicast the entire routing table to all adjacent routers using a reserved multicast address; this is more efficient than RIPv1 because it directs routing table updates only to adjacent routers rather than all routers in the network. A revised version of RIPv2, *RIP next generation (RIPng)*, has been developed for IPv6 networking.
Border Gateway Protocol (BGP)	BGP is an exterior gateway protocol used by routers that connect an organization's network to the Internet. Such routers (called *border* or *edge* routers) transfer packets from devices in the organization's network to other networks over the Internet. These routers use BGP to exchange routing and reachability information with border routers in nearby networks. BGP uses TCP port 179 for routing updates and is the only routing protocol that uses TCP as its transport protocol. The current version of BGP is BGP4.
Open Shortest Path First (OSPF)	OSPF is typically used as an interior gateway protocol. It is sometimes considered a Transport layer (Layer 4) protocol that sits on top of IP and encapsulates routing table updates in Network layer (IP) packets. Since OSPF does not use TCP or UDP, reliable table updates are implemented by mechanisms that are built into the protocol.
Interior Gateway Routing Protocol (IGRP)	Like OSPF, IGRP router table updates are encapsulated in IP packets. Unlike OSPF, IGRP lacks built-in reliable mechanisms, and unreliable transport is assumed. Because IGRP sends updates at regular intervals, unreliable transport is not especially problematic.
Enhanced Interior Gateway Routing Protocol (EIGRP)	Like IGRP, router table update messages are encapsulated in IP packets, but, like OSPF, EIGRP includes built-in reliable transfer mechanisms that do not use TCP or UDP.
Intermediate System to Intermediate System (IS-IS)	For efficiency, IS-IS runs at the Data Link layer and is considered an interior routing protocol. While conceptually similar in function to OSPF, IS-IS messages are not carried in IP packets; they are included in Data Link layer frames.

Because OSPF, IGRP, and EIGRP do not use Transport layer protocols for routing table update messages, they are sometimes identified as Network layer protocols. Two routing protocols, RIP and BGP, that rely on Transport layer protocols for routing table updates are often considered Application layer protocols.

BGP4 is the prevailing standard for exterior gateway routing over the Internet. Most Internet service providers (ISPs) are required to use BGP to enable routing between one another. BGP has multihoming capabilities that enable organizations to have redundant access channels to a single ISP or to have separate (and often redundant) connections to multiple ISPs. Such redundancy can enhance enterprise network availability, performance, and resilience to disruption. BGP4 and other routing protocols are described more fully in Chapter 5.

Appendix 4-3: Additional Data Link Layer Protocols

As noted in section 4.4.4, *Carrier Sense Multiple Access with Collision Detection (CSMA/CD)* is the official Data Link layer protocol for wired Ethernet LANs. In Wi-Fi LANs, *Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA)* is the official protocol. Because of their widespread use, they are the most common Data Link layer protocols. Both CSMA/CD and CSMA/CA are formally recognized in standards formulated and maintained by the Institute of Electrical and Electronics Engineers (IEEE). They are described in Chapters 6 and 7.

Other Data Link layer protocols are identified and briefly described in Table 4-9.

Table: 4-9: Examples of Data Link layer protocols.

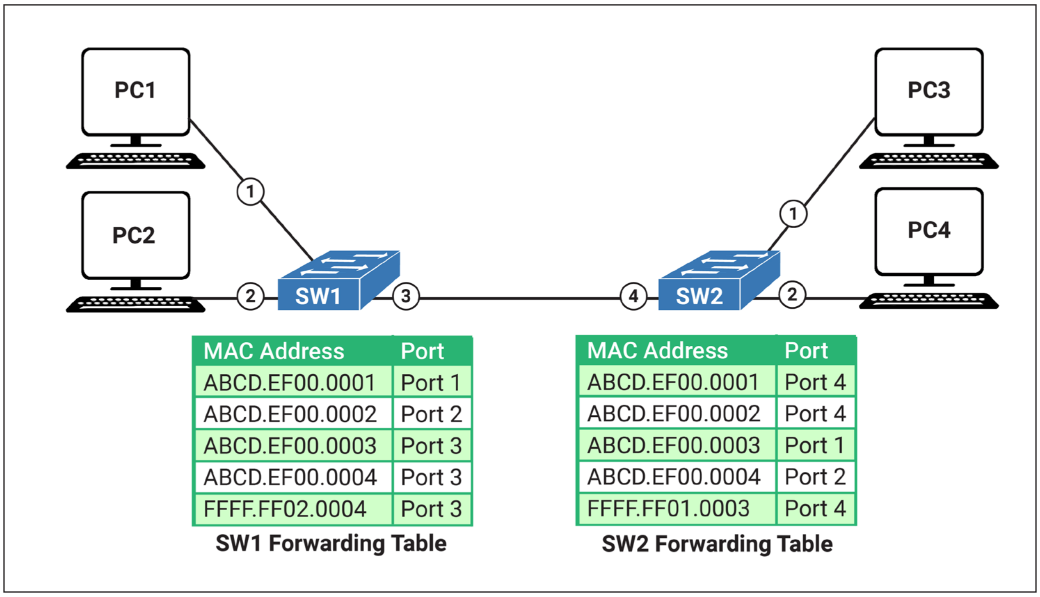
Protocol Name and Acronym	Summary Description
Point-to-Point Protocol (PPP)	Technically, PPP is a family of protocols for transporting data over point-to-point links (direct Physical layer connections between two routers or hosts). PPP is used to tunnel IP or other Network layer packets between directly connected nodes, though IP and Transport layer protocols, such as TCP, are not designed for use on point-to-point connections. PPP can be used on a variety of physical media, including twisted-pair copper wire, fiber optic lines, and cellular telephone or satellite links. It can be used to provide services for dial-up modem, digital subscriber line (DSL), and cable modem connections. A variation of PPP exists for running over Ethernet (PPPoE). PPP use has been declining, as have dial-up modem and other point-to-point connection services; however, it is not dead. PPP specifications are described in the IETF's RFC 1661.
High-Level Data Link Control (HDLC)	HDLC was developed by the ISO in 1979. It was adopted for use in X.25 packet-switching WANs and continues today as an umbrella for numerous WAN protocols that provide best-effort reliable and unreliable services over point-to-point and multi-point connections. It is generally based on SDLC.
Synchronous Data Link Control (SDLC)	SDLC was designed and developed by IBM in 1975 to support communication over point-to-point and multipoint connections within IBM's proprietary Systems Network Architecture (SNA) networks. It introduced the frameworks used by many contemporary networking models, including the TCP/IP model, for ensuring reliable connections, flow control, error detection, and error correction.
Serial Line Internet Protocol (SLIP)	SLIP is an older protocol, superseded by PPP, that is used to encapsulate IP packets in Data Link layer frames. It can and has been used for IP packets, usually among ISPs and home users over dial-up links. It is designed to work with a computer or router's serial communication ports. Unlike PPP, SLIP lacks error detection and error correction capabilities.
Link Control Protocol (LCP)	It was originally developed and created by IEEE's 802.2 committee to provide HDLC-like services in LANs. Basically, LCP is a PPP protocol that can be used for establishing and ending the transmission of frames within a LAN. It can also be used to configure, test, and maintain connections within LANs. PPPoE has superseded LCP in Ethernet LANs.
Link Access Procedure (LAP)	LAP is a family of Data Link layer protocols derived from SDLC for framing and reliably transporting data across point-to-point connections. LAP includes *Link Access Procedure, Balanced (LAPB)*; *Link Access Procedure D-Channel (LAPD)*; and *Link Access Procedure for Frame Relay (LAPF)*.
Network Control Protocol (NCP)	NCP was an older protocol that was implemented by ARPANET, the primary predecessor to the Internet, whose inner workings influenced the development of PPP. It was superseded by TCP/IP in the 1980s.
Source: geeksforgeeks.org	

Address Resolution Protocol (ARP) is sometimes identified as a Data Link layer protocol. As noted previously, ARP is also considered a Network layer protocol or a protocol that spans the border of the Network and Data Link layers. ARP maps logical Network layer addresses (IP addresses) to Data Link layer physical addresses (MAC addresses).

Appendix 4-4: Additional Capabilities of Layer 2 Switches

A Layer 2 switch can be connected to another Layer 2 switch to expand the number of devices that can exchange frames within the local network. This is called *daisy-chaining*, and it can be used to increase the size of the local network. When switches are connected in this manner, each switch's forwarding table must be modified to ensure that frames sent by a device attached to one of the switches is properly forwarded to a device attached to the other switch. This setup is illustrated in Figure 4-40. For example, to send a frame from PC1 to PC4, PC1 would first send the frame to port 1 of SW1. SW1 would use Port 3 as the exit port to send the frame to SW2. SW2 would receive the frame on its Port 4 and forward the frame to PC4 using its Port 2 as the exit port.

Layer 2 switches are usually configured to maximize forwarding speed, but most support several forwarding modes, including those described in Table 4-10.



MAC Address	Port
ABCD.EF00.0001	Port 1
ABCD.EF00.0002	Port 2
ABCD.EF00.0003	Port 3
ABCD.EF00.0004	Port 3
FFFF.FF02.0004	Port 3

SW1 Forwarding Table

MAC Address	Port
ABCD.EF00.0001	Port 4
ABCD.EF00.0002	Port 4
ABCD.EF00.0003	Port 1
ABCD.EF00.0004	Port 2
FFFF.FF01.0003	Port 4

SW2 Forwarding Table

Figure 4-40: Forwarding tables in daisy-chained Layer 2 switches. **Source**: Prospect Press re-rendering of provided image.

Table 4-10: Switching modes in Layer 2 switches.

Switching Mode	Brief Description
Cut-through	In cut-through switching, the switch reads the destination address in the frame header and begins transmitting/forwarding the frame out the exit port without reading the rest of the frame. Both good frames (error-free) and bad frames (frames with errors) are forwarded when this mode is used, but this is the fastest forwarding mode.
Store-and-forward	In this switching mode, the switch waits for the entire frame to be received and placed in a buffer. It checks the frame for errors and forwards frames without errors. Frames with errors are not forwarded. This is the slowest forwarding mode.
Fragment-free	The switch receives and examines the frame header before making the forwarding decision. In an Ethernet network, the frame header is the first 64 bytes of the frame, and 64 bytes is the smallest allowable Ethernet frame size; any frame less than this minimal size is considered a fragment rather than a valid frame. Frames less than 64 bytes in length are not forwarded. Today, this mode is rarely used.

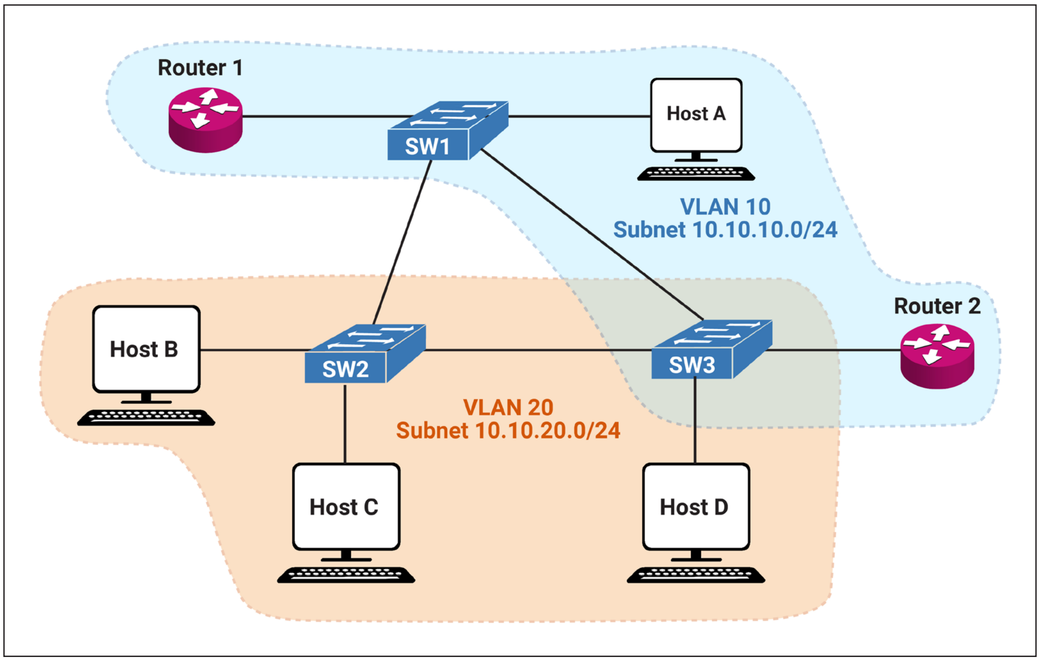


Figure 4-41: Two VLANs in a local network with multiple switches and routers.

Layer 2 switches also support broadcast traffic. When default settings are used, any broadcast frames that a switch receives are forwarded out all ports except the one on which they were received. This defines the switch's default *broadcast domain*. A Layer 2 broadcast frame is a frame with the destination address: FFFF.FFFF.FFFF (or ff:ff:ff:ff:ff:ff or ff-ff-ff-ff-ff-ff). When a device creates a frame with this destination address, the switch knows to transmit it on all of the ports except the one on which it was received. It is important to note that it is the sender that specifies the broadcast address and determines whether the frame will be delivered to the other devices in the local network. Broadcasting is not originated at a switch.

Appendix 4-5: VLAN Fundamentals

Because a VLAN is a logical concept, it cannot be observed from the network's physical topology. A VLAN can be created by grouping ports on a single switch or on multiple interconnected switches (see Figure 4-41).

Best design practices for VLANs recommend a one-to-one relationship between VLANs and IP subnets. This means that devices in a specific VLAN should also be in the same IP subnet.

There are various reasons for creating VLANs, but network security is among the most important. Creating a VLAN can add a logical layer of protection to a network. This is useful in businesses where members of one department or function should not receive data destined for members of another department. Grouping employees into VLANs based on department membership creates multiple self-contained logical LANs, even when each VLAN is created on the same switch (see Figure 4-42).

IEEE standard 802.1Q is the umbrella standard for VLANs. Supporting VLANs in Ethernet LANs requires the addition of VLAN tags to Ethernet frame headers. VLAN implementation in Ethernet LANs is described more fully in Chapter 6.

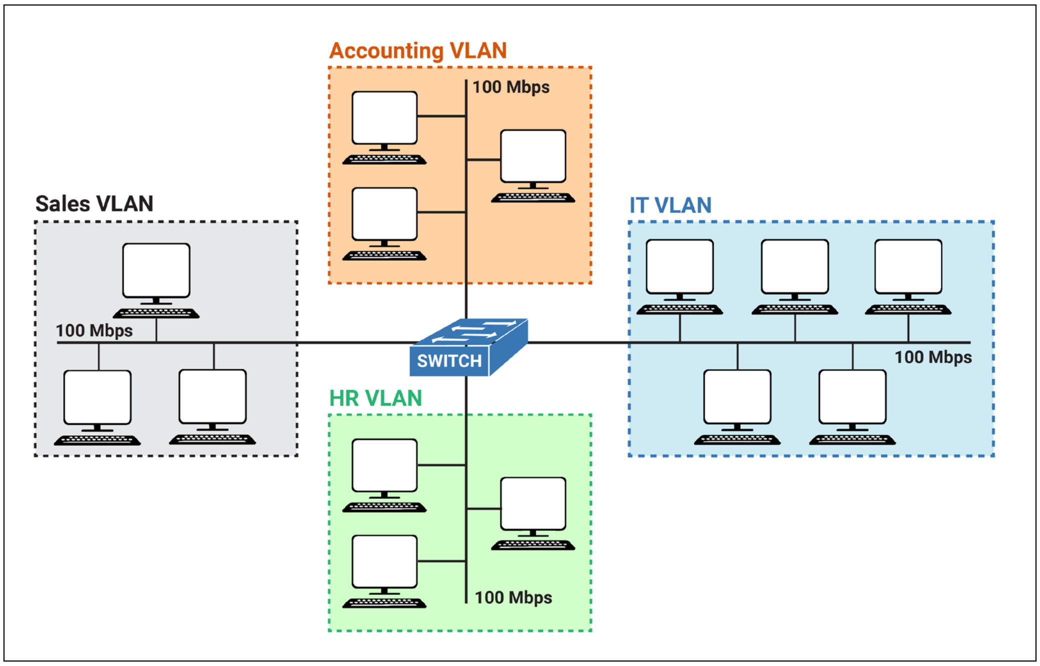


Figure 4-42: Example of multiple business VLANs created on a single switch.

Appendix 4-6: Physical Layer Standards and Specifications

Specifications for connectors, device interfaces, and physical media are addressed in Physical layer standards. Because the circuitry, media, and connectors used at the Physical layer are developed by engineers, the major standards governing Physical layer technologies are defined and maintained by electrical and communications engineering organizations, including the ISO, NIST, and IEEE. Two additional organizations define standards for Physical layer technologies:

- The International Telecommunication Union (ITU)
- The Electronic Industries Alliance/Telecommunications Industry Association (EIA/TIA)

National telecommunications authorities, such as the Federal Communication Commission (FCC) in the United States, also play a role in Physical layer standards by formally adopting standards that the electrical and communications engineering organizations develop.

Network hardware components, such as network adapters, connectors and interfaces, and cable materials and designs, are specified in Physical layer standards. Standards for connectors address the four properties identified in Table 4-11.

Table 4-11: Connector properties addressed by Physical layer standards.

Property	Description
Mechanical	This addresses the mechanical parts of Physical layer technologies, including the physical sizes and shapes of the interfaces and connectors used to connect network components. For example, Ethernet cables have RJ45 connection plugs, and Ethernet switches have RJ45 ports that the cables plug into.
Electrical or Optical	This specifies how bits are represented as symbols on the communication link and includes the symbol type used to represent a 1 or 0 bit (e.g., a voltage level, waveform, or light pulse) and symbol duration.
Functional	This specifies the functions performed by the mechanical components of the connection interfaces, for example, which pins in a RJ45 connector are used for transmitting data and which pins are used for receiving data.
Procedural	This part specifies the sequence of actions necessary to receive or transmit bits on the communication medium.

Network adapters are involved in the implementation of Physical layer standards because they include circuitry used to transmit and receive signals on physical media. They include *transceivers*, the circuitry that creates, transmits, and receives the symbols (signals) used to represent bits on the communication medium. Adapters also include serial-to-parallel and parallel-to-serial circuitry that converts serial transmission on the communication medium to parallel transmission used on the internal buses that interconnect motherboard components, and vice versa.

Appendix 4-7: End-to-End Communications in TCP/IP Networks

Considering end-to-end (host-to-host) communication between a client device in one network and a server in another is a convenient way to summarize many of the observations made in Chapter 4.

Let's assume that a client device in one network (A) requests a web page stored on a web server in a different network (B). Let's also assume that each network is reachable via the public Internet. This situation is illustrated in Figure 4-43.

Figure 4-44 illustrates how the web page request flows from the client device to the web server.

The process begins when an Application layer header is added to the client's web page request before it is passed from the Application layer to the Transport layer.

There, a Transport layer header is added to what it receives from the Application layer. The Transport layer header includes source and destination port numbers for the application. When the Transport layer passes the web page request to the Network layer, a Network layer header is added that includes the client's IP address and the IP address of the web server's network,

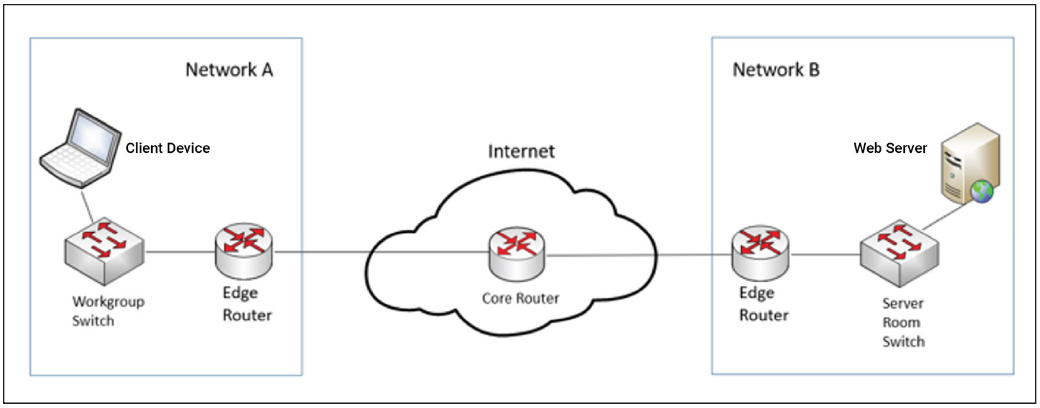


Figure 4-43: An example of internetwork connections between two networks.

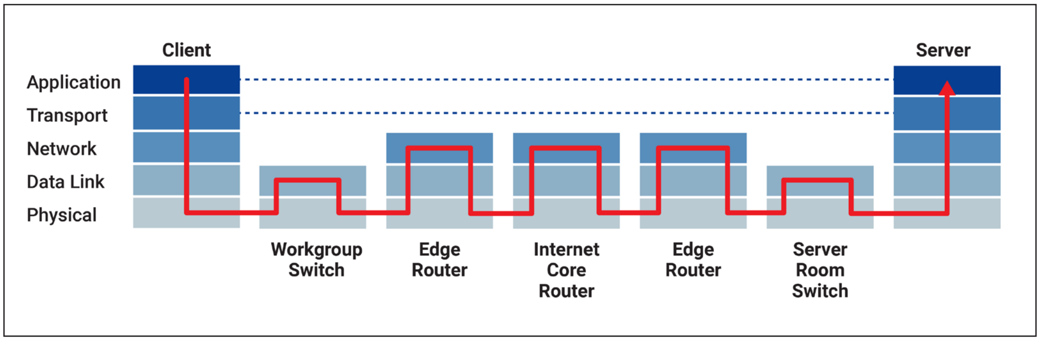


Figure 4-44: A conceptual depiction of the flow of data from client to server in Figure 4-43.

Network B. If the server's IP address is unknown, the client will contact the DNS system to identify it.

After the Network layer header is added, the IP packet is passed to the Data Link layer, where another header is added that contains the client's MAC address and the MAC address of Network A's edge router, the device responsible for sending the web page request over the Internet to Network B. The frame created at the Data Link layer also includes a trailer field to enable error checking. In short, the client's web page request is encapsulated in a frame before it is passed to the Physical layer and transmitted over Network A's physical connections.

The bits of the frame that contain the web page request will be represented as signals on the cable that connects the client and workgroup switch. This is typically an Ethernet cable that transfers bits as electronic pulses, such as the digital signals depicted in Figure 4-35 and Figure 4-36.

When the workgroup switch receives the signals that represent the bits in the frame, it will convert them back to the bits that they represent, reproduce the frame, and pass it to the switch's Data Link layer. The switch will identify the destination MAC address in the frame, consult its forwarding table, identify the port that connects the switch and edge router, and transmit the frame to the router. If a fiber optic patch cable is used to connect the router to the switch, a fiber optic cable port will be used to transmit the frame, and light pulses will be used to represent the frame's bits.

When the edge router receives the light pulses representing the bits in the frame, it converts them back to the bits, reproduces the frame, and passes the frame to the Data Link layer, where it is checked for errors. If there are no errors, the frame header and trailer are removed before the rest of the frame's contents are passed to the Network layer. At the Network layer, the destination address is identified, and the router consults its routing table to make its next-hop decision, identify the outgoing port to use, and create a new packet that includes the IP address of the next-hop router. The packet is then passed down to the Data Link layer, where it is encapsulated in a new frame prior to being passed down to the Physical layer, where its bits are represented as signals on the transmission medium that connects the edge router to the core router on the Internet. Since fiber optic cabling is often used for router-to-router internetworking, the frame that includes the web page request may be encapsulated in a SONET frame for transmission over the fiber optic links.

The Internet core router will receive the signals that represent the frame's bits, remove the SONET header, and convert the signals to the bits they represent before passing the frame to the Data Link layer. Like Network A's edge router, the core router may use the frame's trailer field to check for errors at the Data Link layer. The frame's header and trailer are removed before its contents are passed to the Network layer, where the destination IP address is read, a next-hop decision is made, and a new packet that identifies the next-hop router's IP address is created. The packet is then passed to the Data Link layer, where a new frame is created before it is passed to the Physical layer, where it may be encapsulated in a new SONET frame for transmission over the fiber optic link to Network B's edge router.

When the edge router in Network B receives the signals that represent the frame with the web page request, it removes the SONET header and converts the signals to the bits they represent, reproduces the frame, and passes the frame to the Data Link layer, where its trailer may be used for error checking. The frame is then passed to the Network layer, where the destination IP address is identified. Upon determining that the destination network is the edge router's local network, the router passes the IP packet to the Data Link layer, where a new frame that includes the web server's MAC address as the destination address is created. If the router does not know the web server's MAC address, it will use ARP to identify it. The new frame is then sent to the Physical layer, where its bits will be represented on the local network's transmission medium. Since fiber optic patch cable is commonly used to connect a router to a switch, the frame's bits are likely to be represented as light pulses on the patch cable to the server room switch.

The server room switch receives the signals transmitted by the edge router, converts them to the bits they represent, and reproduces the frame. It identifies the destination MAC address in the Data Link layer frame, consults its forwarding table, and transmits the frame's bits on the cable that connects the web server to the switch. Since fiber optic patch cables are often used for such connections, media conversion may not be necessary.

The server receives the signals from the switch, converts them to the bits that they represent, and reproduces the frame. The frame is passed to the Data Link layer, and the frame header will be removed. The frame trailer is removed after an error check is performed. If no errors are detected, the IP packet is passed to the Network layer, where the Network layer header is removed before it is passed to the Transport layer. The server's Transport layer removes the Transport layer header after ensuring that the sender's message (the web page request) was directed to the web server. It then uses the destination port number to deliver the message to the appropriate application at the Application layer.

The HTTP (or HTTPS) application reads the client's request and initiates the response process required to send the requested web page to the client. The client's request is a simple message that consists of a GET command and the web page's URL, along with an ordered sequence of header fields. The request is included in the Request Line of the HTTP Request packet created by the client's Application layer before being passed down to the client's Transport layer. When responding to the client, the server's HTTP application creates an HTTP Response packet that contains an ordered set of header fields and the contents of the web page that was requested.

Despite having to step through the sequence of communication activities just described, everything is typically accomplished in the blink of an eye. Today, we are accustomed to viewing a web page as soon as we click on its link. To the casual user, the click is all that is required. The reality, however, is that the click initiated a sequence of communication activities with many steps and handoffs among devices and layers and protocols in the TCP/IP model that made it possible for the client to access the web page. A tremendous amount of engineering and software programming has made it possible to have what we have today, and these foundations will be springboards for network performance improvements in the future.

CHAPTER 5

Appendix 5-1: Autonomous System (AS) Numbers

Each AS is identified by an autonomous system number (ASN) that is assigned by the Network Information Center (NIC) in the United States. ASNs are for BGP router configurations. The ASN is an "official" number that is analogous to a business's unique (and official) number on a business license. ASNs are unique 16-bit (between 1 and 65,534) or 32-bit numbers (between 131,072 and 4,294,967,296). As of March 2021, ASNs had been assigned to more than 100,000 companies, educational institutions, nonprofit organizations, and government entities.

ASNs are allocated to ASs by Regional Internet Registries (RIRs) that receive blocks of assignable ASNs from the IANA (Internet Assigned Numbers Authority). A network must meet certain qualification criteria before being assigned an ASN. It must have a distinct routing policy, be sufficiently sizable, and have connections to two or more existing ASs.

ASNs are only required for communications among BGP routers. Internal routers do not need to know the AS's ASN to forward packets to other routers in the AS, and they are not required to use BGP as the routing protocol.

BGP is only required for an AS's edge (border) routers to enable them to be involved in routing packets to/from other Internet-connected networks. The version of BGP used in edge routers is *external BGP (eBGP)*, so, technically speaking, it is eBGP that provides interconnectivity among the Internet's ASs.

The use of eBGP to interconnect ASs is illustrated in Figure 5-32. *Internal BGP (iBGP)* is a version of BGP that can be used for routing among an AS's internal routers; however, an AS can use other protocols, such as OSPF, for internal routing.

Appendix 5-2: Domain Name Registration

Registering a domain name provides an entity (e.g., a company, nonprofit organization, educational institution, government agency) with a unique identity. A domain name, an identifiable website, and identifiable email addresses provide a business with a professional Internet presence. Registering a domain name also helps a business to protect trademarks and copyrights, increase brand awareness, and improve its search engine positioning.

A typical registration process is illustrated in Figure 5-33. Operationally, authorized resellers for ICANN registrars handle domain name registrations and renewals for specific entities.

Appendix 5-3: HTTP Request and Response Messages

Figure 5-9 in section 5.2.1 provides a visual overview of the fields in HTTP request and response messages. Figure 5-34 provides a closer look at the details included in request and header lines in HTTP request messages and illustrates that the request line includes method, URL, and version sections.

Figure 5-35 provides an example of what these lines would look like in an HTTP "GET" request message header. In the request line of this figure, the method is GET, the URL is index.html, and the version is HTTP/1.0.

"GET" is the method used to retrieve or read web pages (or other resources) from a web server. Among methods commonly included in HTTP request lines are POST (used to create new resources, especially subordinate resources), PATCH (used to modify/update existing resources), and DELETE

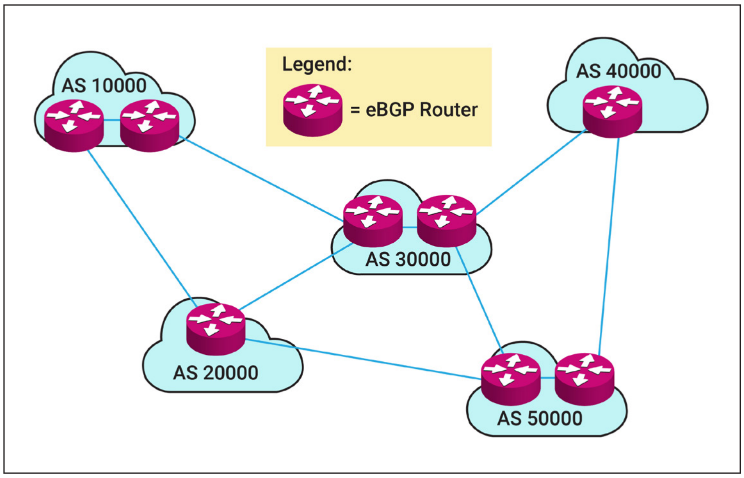


Figure 5-32: eBGP is used in ASs' edge (border) routers for internetworking among ASs.

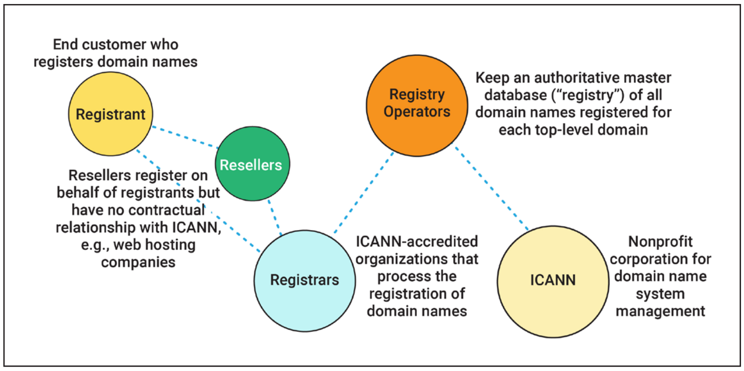

End customer who registers domain names
Registrant
Resellers
Resellers register on behalf of registrants but have no contractual relationship with ICANN, e.g., web hosting companies
Registry Operators
Keep an authoritative master database ("registry") of all domain names registered for each top-level domain
Registrars
ICANN-accredited organizations that process the registration of domain names
ICANN
Nonprofit corporation for domain name system management

Figure 5-33: The domain name registration process.

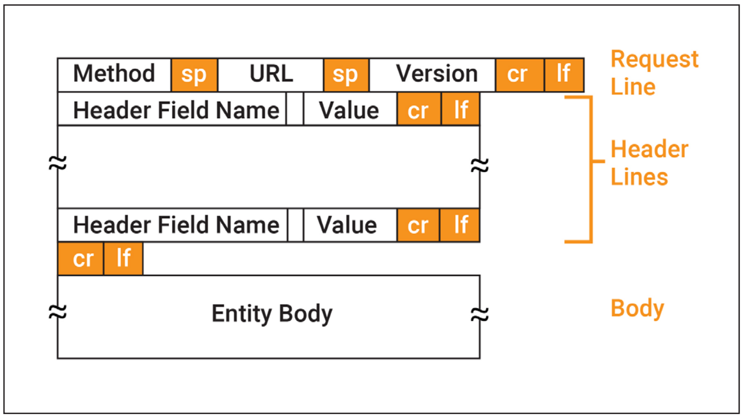
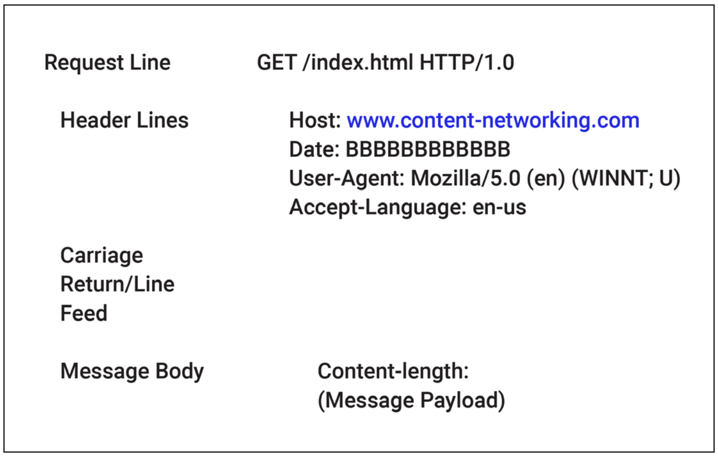


Figure 5-34: HTTP request message format.



Request Line	GET /index.html HTTP/1.0
Header Lines	Host: www.content-networking.com Date: BBBBBBBBBBBB User-Agent: Mozilla/5.0 (en) (WINNT; U) Accept-Language: en-us
Carriage Return/Line Feed	
Message Body	Content-length: (Message Payload)

Figure 5-35: Example of an HTTP GET message.

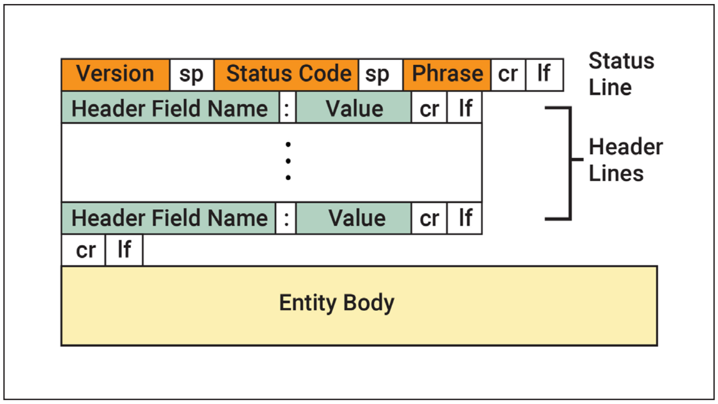
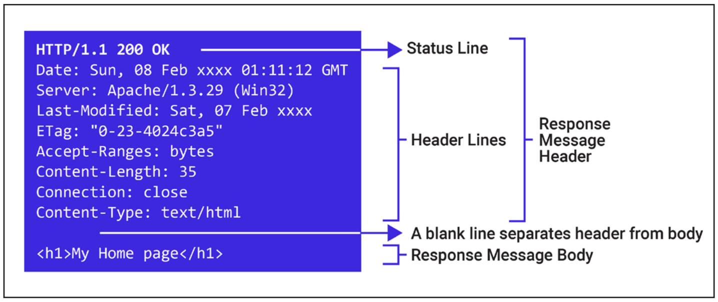


Figure 5-36: HTTP response message format.



```
HTTP/1.1 200 OK
Date: Sun, 08 Feb xxxx 01:11:12 GMT
Server: Apache/1.3.29 (Win32)
Last-Modified: Sat, 07 Feb xxxx
ETag: "0-23-4024c3a5"
Accept-Ranges: bytes
Content-Length: 35
Connection: close
Content-Type: text/html

<h1>My Home page</h1>
```

Status Line
Header Lines
Response Message Header
A blank line separates header from body
Response Message Body

Figure 5-37: Example of an HTTP response message.

(used to remove resources). POST, PATCH, and DELETE are methods that should only be used by individuals who are authorized to make changes to web pages and other resources. Regular users should only be able to view (read) web server resources and should be restricted to using the GET method.

The use of POST, PATCH, and DELETE methods by unauthorized or malicious actors can result in the defacing of a web page, the installation of malware, or the removal/destruction of important web resources. The potential for malicious use of these HTTP methods is one of the reasons why HTTPS has largely replaced HTTP. When HTTP client-server interactions occur over unsecure media such as public Wi-Fi networks, unencrypted HTTP messages are vulnerable to interception. With HTTPS, the contents of intercepted IP packets are unintelligible.

Figure 5-36 illustrates the format of an HTTP response message. It shows that the status line of an HTTP response message includes version, status code, and phrase sections.

Figure 5-37 provides an example of an HTTP response message. Here, the version in HTTP/1.1, the status code is 200, and the phrase is OK. Basically, this informs the client that its request has been received and that the requested web page (or resource) has been retrieved and is included in the body of the response message.

Appendix 5-4: SFTP

SFTP is an SSH extension that secures data transfer between an SFTP server and an SFTP client over any reliable data stream. SFTP assumes a secure communication channel, such as an SSH tunnel, after the SFTP server has authenticated the client. Because it requires only a single port (22) to establish a connection between a client and a server, SFTP is easier to implement than FTP, FTPS, or FTP over SSH. SFTP is widely used for remotely accessing and transferring files over the Internet.

SFTP is a client-server protocol that can be launched via a command line or a graphical user interface (GUI). SFTP client software is needed to connect to an SFTP server. An SFTP server requires both communicating parties to authenticate, either by providing a user ID and password or by validating an SSH key, or both. While ID and password encryption is common in SFTP implementations, it is not required and is sometimes considered an SFTP shortcoming.

SFTP servers enable large files to be transferred easily and efficiently and are superior to email or cloud services for transferring documents, forms, and business-critical files to business partners. By using encryption, public key authentication, data integrity checks, and host authentication, businesses face fewer risks when they use SFTP to exchange files. Transferred data is checked to ensure that it is coming from a trusted source, and hosts are verified before connections are established.

Appendix 5-5: ARQ Protocols and TCP

Sliding windows are a key part of many protocols, including the TCP protocol. They contribute to reliable data delivery and overall network efficiency by addressing unacknowledged packets. Sliding windows are supported in some versions of ARQ.

Automatic Repeat reQuest (ARQ) protocols are considered error control protocols that provide reliability to unreliable networks. These protocols can be used at either the Data Link or Transport layer, but today, they are mostly used at the Transport layer. At the Transport layer, the retransmission is triggered by unacknowledged segments.

There are three main types of ARQ protocols: stop-and-wait ARQ , Go-Back-N ARQ , and Selective Repeat ARQ. They are briefly described in Table 5-7 in section 5.3.1 and illustrated in Figure 5-38. For stop-and-wait in this figure, Frame 1 is retransmitted due to the lack of acknowledgment receipt during its timeout interval.

TCP often uses Go-Back-N ARQ but sometimes uses a variant of Go-Back-N or Selective Repeat ARQ, in which cumulative acknowledgments are used instead of acknowledging individual segments. With *cumulative acknowledgment*, the receiver sends a single acknowledgment in response to a finite number of segments received to signify that the receiver acknowledges that it

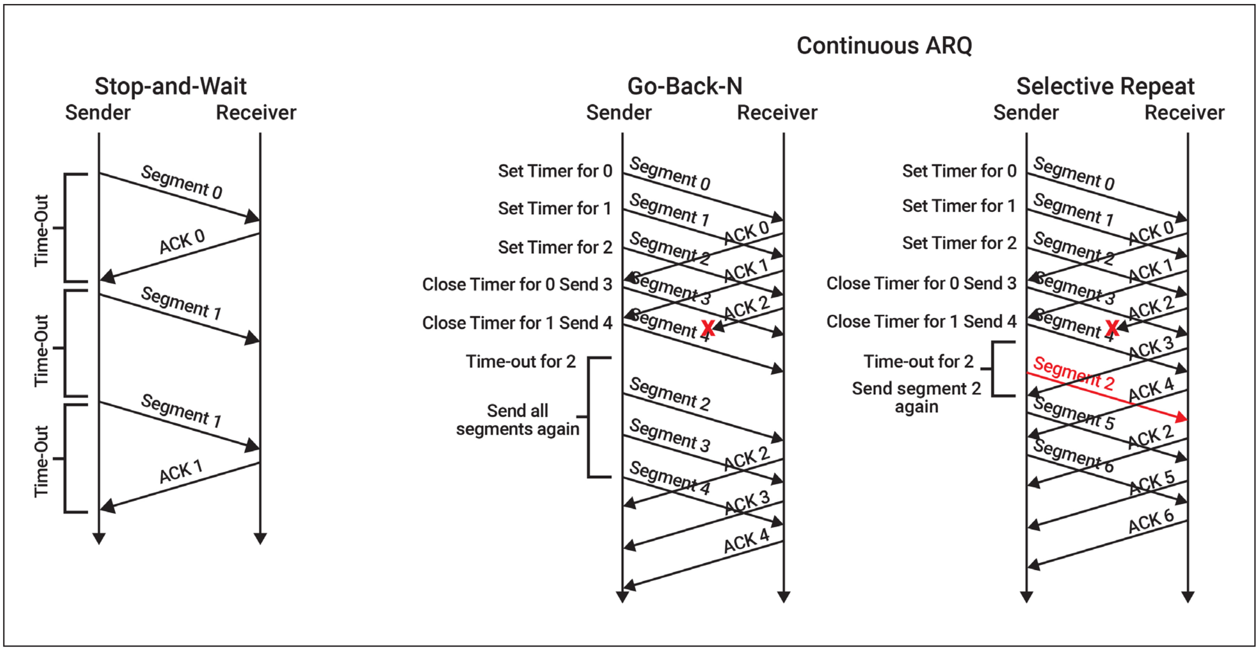


Figure 5-38: Stop-and-wait, Go-Back-N, and Selective Repeat ARQ protocols.

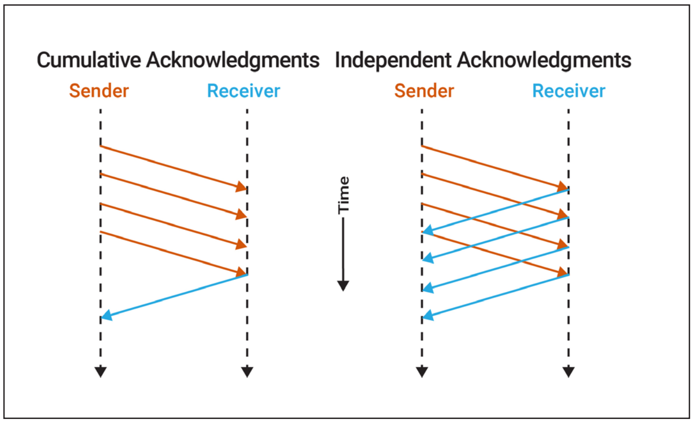


Figure 5-39: Cumulative vs. independent acknowledgments.

has correctly received all previous segments. The difference between cumulative and independent acknowledgments is illustrated in Figure 5-39.

Selective acknowledgment (SACK), a variation on Selective Repeat ARQ, is an example of a cumulative acknowledgment scheme used in some TCP connections. When it is used, the receiver can inform the sender about all segments that have arrived successfully. Like Selective Repeat ARQ, the sender only retransmits segments that have been lost or discarded at lower layers of the protocol stack, but when all segments successfully arrive, the receiver can confirm their receipt in a single acknowledgment. Sender-receiver negotiations during the three-way handshake determine that the SACK option can be used once the connection is established.

Continuous ARQ protocols also contribute to flow control and error control processes. For example, window size (the number of segments/frames per window) may be adjusted during a TCP session to ensure that the receiver receives segments at a rate it can handle. And, when frames are discarded by receivers when errors are detected (at the Data Link layer), senders retransmit missing and unacknowledged segments.

Appendix 5-6: TCP Header Field Descriptions

Table 5-13 provides summary descriptions of TCP headers.

Table 5-13: Descriptions of fields in TCP headers.

TCP Header Field	Description
Source Port	This is a 16-bit (2-byte) field that specifies the port number of the sender's application.
Destination Port	This is a 16-bit field that specifies the port number of the receiver's application.
Sequence Number	This 32-bit (4-byte) number ensures that data is correctly sequenced. It is also used to keep track of transmitted data. When a TCP connection is established (via the three-way handshake), the initial sequence number is a random 32-bit value generated by the sender. Receivers reference segment sequence numbers in their acknowledgments.
Acknowledgment Number	This 32-bit field indicates the next sequence number that the sending device is expecting from the other host. It is used by the receiver to identify/request the next TCP segment. The field's value will be the sequence number incremented by 1.
HLEN (Header Length)	This 4-bit data offset field indicates the length of the TCP header (in bytes) so that the receiver will know where application data begins.
RSV (Reserved)	There are 6 bits in the reserved field. Each bit is always set to 0.
Code/Control Bits/ Flags	There are 6 bits for code bits (a.k.a. control bits or flags). These bits are used to establish and terminate connections and send data: • **URG:** Urgent pointer. When this bit is set, the data should be treated as priority over other data. • **ACK:** Used to acknowledge a received segment. • **PSH:** This is the push function. This tells an application that the data should be transmitted immediately and should not wait to fill the entire TCP segment. • **RST:** This "reset" bit is used to reset a TCP connection. When it is used, the receiver immediately terminates the connection without using the normal four-way handshake. This is only used when there are unrecoverable communication errors. • **SYN:** This "synchronize" bit is used in the three-way handshake to establish a connection and to set the initial sequence number. • **FIN:** This "finish" bit is used to end the TCP connection. Since TCP is a full duplex (bidirectional) communication session, both hosts send a FIN bit to each other to end the connection.

Table 5-13, Continued.	
TCP Header Field	**Description**
Window	This 16-bit field specifies how many bytes the receiver is willing to receive and place in its buffer. Each host specifies its window size during session establishment. Once a connection is established, a receiver can use this field to tell the sender that it would like to receive more, or less, data than it is currently receiving and buffering. Window size is important in TCP's ARQ processes.
Checksum	These 16 bits are used to check for errors in the TCP header. Checksum is a common error-checking approach used to detect errors in a segment or frame.
Urgent Pointer	These 16 bits are used when the URG bit has been set. This field is used to indicate where the urgent data ends.
Options	This field is optional and can be anywhere between 0 and 320 bits in length. It is often used to specify the maximum segment size (MSS) that the receiver's local network can handle.

Appendix 5-7: IPv4 and IPv6 Header Field Descriptions

Table 5-14 provides descriptions of the fields in IPv4 headers that are illustrated in Figure 5-17a, and Table 5-15 describes the fields in IPv6 headers that are illustrated in Figure 5-17b.

Table 5-14: IPv4 header field descriptions.	
Field	**Description**
Version	Identifies which IP version is being used. In an IPv4 header, the decimal value 4 is represented by the bits in this field.
Internet Header Length (IHL)	This 4-bit field identifies the length of the IP header in 4-byte blocks. The minimum length of an IP header is 20 bytes, and an IP header with this length would have an IHL value of 5. The maximum length of an IP header is 60 bytes, so a 60-byte IP header would have an IHL value of 15.
Type of Service	These eight bits may be used to specify the quality of service (QoS) the packet should have during routing, such as whether it should be placed in a prioritized outgoing queue or should take a route with appropriate latency, reliability, or throughput. Today, the original Type of Service field is subdivided into two parts: • *Differentiated Services Code Point (DSCP)* [DiffServ]: a 5-bit field used for QoS routing. • *Explicit Congestion Notification (ECN)*: a 3-bit field used for end-to-end notification of network congestion without dropping packets. ECN is an optional flow-control feature that is available when both endpoints support it and when it is also supported by the underlying network.
Total Length	This 16-bit field specifies the entire size of the IP packet (header and data) in bytes. The minimum size is 20 bytes (if there is no data), and the maximum size is 65.535 bytes, the highest value that can be represented by 16 bits.
Identification	This field is primarily used to uniquely identify the group of fragments associated with a single IP packet. If a packet is divided into fragments by a router, the same 16-bit identification number is included in each fragment to identify the original unfragmented IP packet; this enables the packet to be reassembled by receivers.
Flags	These three bits are used if packets are fragmented: • The first bit is always set to 0. • The second bit is the *DF (Don't Fragment)* bit, and when used it indicates that this packet should not be fragmented. • The third bit is the *MF (More Fragments)* bit and is set on all packet fragments except the last one.
Fragment Offset	This 13-bit field is used to specify the position of a fragment in the original IP packet. Receivers use it for the reassembly of fragmented packets.
Time to Live (TTL)	This 8-bit field limits a packet's lifetime on the network. It is included to prevent a network failure caused by a routing loop. In practice, the field is used as a hop count—when the packet arrives at a router, the router decrements the TTL field by 1. If the TTL field hits 0, a network failure is assumed, and the router discards the packet; the router also usually sends an ICMP "time exceeded" message to the sender.
Protocol	This field defines the protocol encapsulated in the data portion of the IP packet. For example, TCP has value 6 and UDP has value 17.

Table 5-14, Continued.

Field	Description
Checksum	This 16-bit field is used to store a checksum of the header. A receiver can use the checksum field to check for errors in the header. If the receiver is a router, the router will discard the packet if it finds an error.
Source Address	This field specifies the IPv4 address of the sender of the packet.
Destination Address	This field is the IPv4 address of the receiver of the packet.
Options	This field is rarely used; it is optional and, when used, has a variable length based on the options that are specified. When used, the value in the header length (IHL) field will increase. An example of an option is a "source route" request, where the sender requests a specific routing path.

Table 5-15: Summary descriptions of IPv6 header fields.

IPv6 Header Field	Summary Description
Version	Like its equivalent in an IPv4 address, this field specifies the Internet Protocol version. For IPv6, these four bits are 0110, which is the equivalent of the decimal value of 6.
Priority/Traffic Class	Like the Type of Service field in an IPv4 header, this field identifies the router services the packet should receive based on the priority of the packet. If network congestion occurs, packets with the lowest priority may be discarded.
Flow Label	This label is used to maintain the sequential flow of the packets associated with an Internet communication exchange and helps a router determine if the packet belongs to a specific information flow. It helps avoid/prevent the need for packet reordering by receivers. It plays an important part in streaming and real-time applications.
Payload Length	This field informs routers how much information (in bytes) is contained in the packet's payload. A packet's payload consists of its optional Extension Headers and the PDU received from the Transport layer.
Next Header	The Next Header field indicates the type of extension header (if any) that immediately follows the IPv6 header. In some instances, it identifies the protocol included in Transport layer packets, such as TCP or UDP.
Hop Limit	Like the TTL field in IPv4 headers, this field is used to stop a packet from looping infinitely in the network without being delivered to its destination. This field specifies the maximum number of intermediate nodes (routers) allowed to process the packet. Its value gets decremented by one by each node that forwards the packet to another node, and this field causes the packet to be discarded if the value reaches 0.
Source Address	This field contains the 128-bit IPv6 address of the packet's source.
Destination Address	The Destination Address field identifies the IPv6 address of the destination device (in most cases). Intermediate nodes (routers) use this field to correctly forward/route the packet.
Extension Headers	An IPv6 packet may contain zero, one, or more than one Extension Headers. There is a recommended order when more than one Extension Header is used, with each extension header pointing to the next extension header. When Extension Headers are used, the packet's Next Header field points to the first Extension Header, and if there is more than one Extension Header, then the first Extension Header's Next Header field points to the second one, and so on. Extension Headers include Hop-by-Hop or Destination options, a Routing Header (to assist routers in making routing decisions), a Fragment Header (to address packet fragmentation by routers), an Authentication Header (to verify the identity of the sender), and an Encapsulation Security Payload header for the inclusion of encryption information.

Appendix 5-8: Classful IPv4 Addressing

Classful addressing is a concept that divides the available address space of IPv4 into five classes, namely A, B, C, D, and E, where classes have a fixed number of blocks, and each block has a fixed number of hosts. In IPv4, an address is 32 bits in length. Because 32 bits are used, the total address space of IPv4 is 2^{32}, which is equal to 4,294,967,296 unique addresses.

When in use on TCP/IP networks, IPv4 addresses are expressed using the binary notation. However, dotted decimal notation or hexadecimal notation is more readable by humans. A 32-bit IPv4 address is sometimes called a 4-byte or 4-octet address, because the 32 bits of the address can be viewed as consisting of 4 bytes or octets. See Figure 5-40, whose four

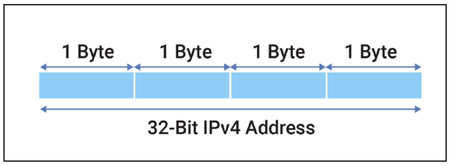


Figure 5-40: IPv4 addresses are 32 bits, 4 bytes, in length.

parts can be represented by dotted decimal notation, such as 192.268.1.10.

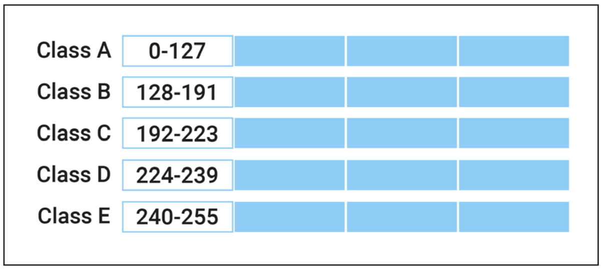
When classful addresses are used, the value of the first byte represents the class of the address and denotes the range of addresses in the class, as shown in Figure 5-41.

For class A, B, and C addresses, the first part of the address identifies the network (this is called the *Network ID*), and the second part identifies the host (called *Host ID*) within the network. The size of these two parts varies with the classes (see Figure 5-41):

- In Class A, the first byte of the address identifies the Network ID, and the other three bytes make up the Host ID.
- In Class B, the Network ID consists of the first two bytes of the address, and the other two byes make up the Host ID.
- In Class C, the first three bytes make up the Network ID, and the last byte identifies the host within the network (the Host ID).

As Figure 5-42 illustrates, a limited number of Class A networks can be created via classful addressing, but each network may include more than 16 million hosts. This figure also shows that more than 16,000 Class B networks can be created, and each one could have more than 65,000 hosts. More than 2 million Class C networks can be created, but each can have no more than 256 hosts.

Figure 5-42 also illustrates some of the disadvantages of classful addressing. For both Class A and Class B, there are more hosts per block than any organization is likely to need. This means that some potential Host IDs are likely to go unused and a portion of each block's total IP address space is wasted. Organizations that are assigned Class C blocks, however, may not have enough Host IDs to meet their requirements. Hence, assigning a fixed size block of IPv4 addresses to an organization can result in IP address wastage or insufficiency. Clearly, assigning addresses according to user organization needs or requirements is a superior approach.



Class A	0-127
Class B	128-191
Class C	192-223
Class D	224-239
Class E	240-255

Figure 5-41: IPv4 address classes.

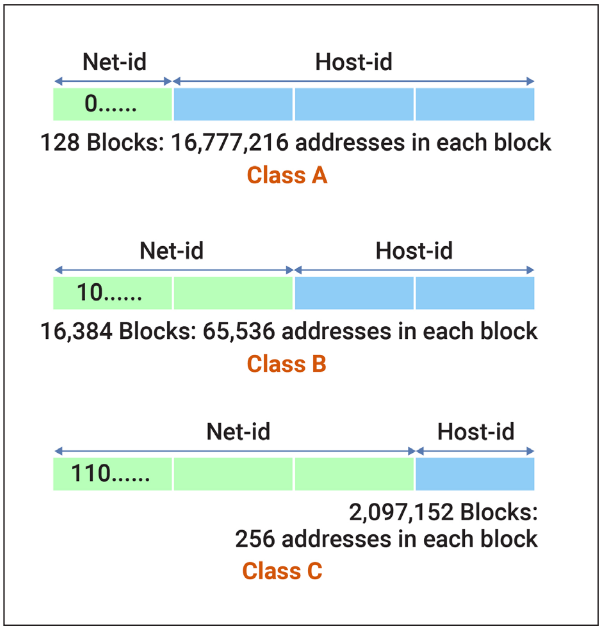

Net-id
Host-id
0......
128 Blocks: 16,777,216 addresses in each block
Class A
Net-id
Host-id
10......
16,384 Blocks: 65,536 addresses in each block
Class B
Net-id
Host-id
110......
2,097,152 Blocks:
256 addresses in each block
Class C

Figure 5-42: Network and Host ID portions of IPv4 Classes A, B, and C.

Subnetting and supernetting emerged as two solutions to classful IPv4 address wastage or shortage. Subnetting enables organizations to divide their total IP address space into smaller subnets that can be shared with other organizations. For example, this would enable an ISP with a block of A or B addresses to create and share subnetworks with their customers. VLANs, introduced in Chapter 2 are another example of subnets.

A *subnet* (or *subnetwork*) is a logical subdivision of an IP network, and *subnetting* involves dividing a network into two or more smaller networks. A subnet is constructed by taking a fixed number of bits from the Host ID and using them to create a fixed number of smaller subnetworks in the original network. This means that the resulting IP addresses will have three parts instead of two: Network ID, Subnet ID, and Host ID.

Supernetting enables smaller networks to be combined into larger ones. When this is done, a fixed number of bits is "borrowed" from the Network ID. This also results in IP addresses that

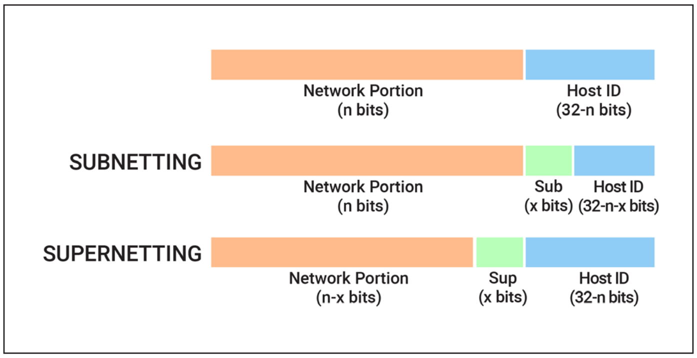
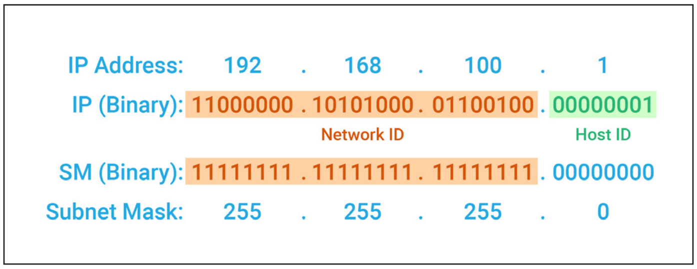
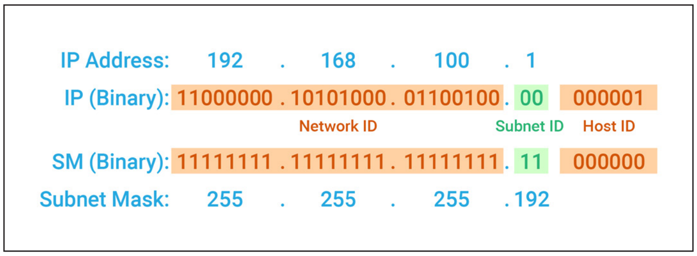


Figure 5-43: IPv4 subnets and supernets.



IP Address:	192 .	168 .	100 .	1
IP (Binary):	11000000 .	10101000 .	01100100 .	00000001
		Network ID		Host ID
SM (Binary):	11111111 .	11111111 .	11111111 .	00000000
Subnet Mask:	255 .	255 .	255 .	0

Figure 5-44: Network ID and Host ID for an IP address with a 255.255.255.0 subnet mask.



IP Address:	192 .	168 .	100 .	1	
IP (Binary):	11000000 .	10101000 .	01100100 .	00	000001
		Network ID		Subnet ID	Host ID
SM (Binary):	11111111 .	11111111 .	11111111 .	11	000000
Subnet Mask:	255 .	255 .	255 .	192	

Figure 5-45: Combining an IP address and a subnet mask in a Class C address divides the address into three parts: Network ID, Subnet ID, and Host ID.

have three (instead of two) parts: Network ID, Supernet ID, and Host ID. Both subnetting and supernetting are illustrated in Figure 5-43.

Subnet masks are used when subnetting breaks a network into smaller networks. Like an IPv4 address, a subnet mask consists of 4 bytes (octets). The combination of an IP Address and a subnet mask defines the network boundary.

A *subnet mask* consists of a string of 1s (from left to right) followed by a string of 0s to make up 32 bits. The 1s in the subnet mask identify the Network ID of the IP Address, and the 0s identify the Host ID portion of the address. For example, in Figure 5-44, the Host ID portion of a network device with an IP address of 192.168.100.1 and a 255.255.255.0 subnet mask (the default subnet mask for Class C) would be the last eight bits of the IP address (00000001).

When a Class C IP address is combined with a subnet mask, such as 255.255.255.192, the original 8 bits for the Host ID are divided into a Subnet ID and a Host ID, as shown in Figure 5-45.

In short, with classful addressing, the size of networks is fixed, and each class has a default subnet mask. A network can be subdivided into subnetworks to enable its block of IP addresses to be assigned more efficiently and/or shared. When subnets are created, some of the bits of the Host ID portion of an IP address are used to identify a host's subnet, and the rest identify the host within the subnet. Nondefault subnet masks indicate how many of the bits in the Host ID portion of the address are used to identify the subnet and how many are used to identify the host.

Appendix 5-9: Private IPv6 Addresses

Because the IPv6 address space is huge, there is less need for private IPv6 addresses. However, private IPv6 address spaces are addressed in RFC 4913 to enable organizations that want private IPv6 addresses to have them. They are called *unique local addresses (ULAs)*.

Like private IPv4 addresses, ULAs are not routed on the Internet. ULAs may be used freely, without centralized registration, within a single organization or site (such as a data center), or a network that spans a limited number of sites or organizations. ULAs are routable within the private networks, but not on the global IPv6 Internet.

FC00::/7 is the reserved range for ULAs. A range of FC00::/7 means that IPv6 ULAs begin with 7 bits with this exact binary pattern: 1111 110L. The L bit is set to 1 to signify that the address is locally generated. This is illustrated in Figure 5-46. Technically, extending the 7-bit prefix with the L bit set to 1 means that ULAs fall in the FD00::/8 range.

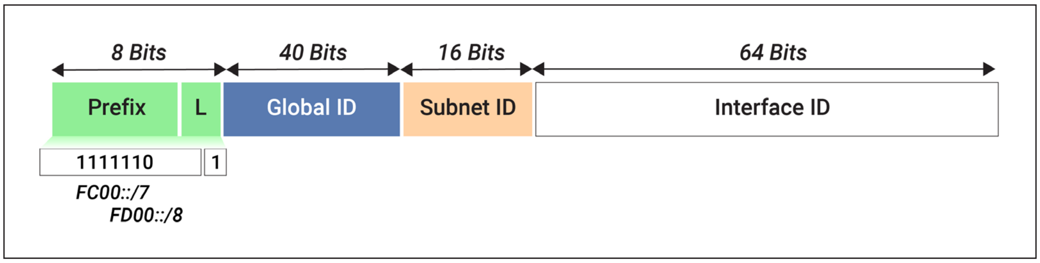


Figure 5-46: ULA format.

Appendix 5-10: IPsec Details

IPsec is incorporated into IPv6, which means that communication between two IPv6 endpoints is either authenticated, encrypted, or both, via Extension Headers, specifically the Authentication Header (AH) and Encapsulating Security Payload (ESP) headers.

The AH provides IP datagrams with data origin authentication and connectionless integrity. "Data origin authentication" is achieved by using integrity check values (ICVs)—error checks—calculated for the packet's payload. Senders append ICVs as packet trailers. If the recipient's independent ICV calculation for the packet payload does not match the ICV in the packet trailer, the packet is discarded. In essence, when the sender's and recipient's ICVs match, the packet sender is authenticated; when they do not, the sender is considered unauthenticated, and the recipient refuses to accept the packet.

"Connectionless integrity" results from the inability of recipients to notify senders when ICVs do not match. This mechanism appropriately assumes that ICVs passed between IPv6 endpoints should match and that sender identity cannot be verified when they do not.

The Encapsulating Security Payload (ESP) header supports IP packet confidentiality and privacy by encrypting the packet's payload. ESP supports either encryption only or authentication only, but because using encryption without authentication is considered insecure, ESP encryption is combined with AH authentication.

Figure 5-47 provides additional details on how the IPsec Transport mode works. AH and ESP headers are added to the Transport layer PDU. When IPsec operates in this mode, it is integrated with IP and is used to transfer the TCP segment or UDP datagram between the sending and destination hosts. After processing, the packet that is passed to the Data Link layer has a single IP header that contains IPsec AH and/or ESP header fields.

When IPsec is used in Transport mode, the IPsec header and the AH and/or ESP extension fields only apply to the IP packet's payload; they are not applied to the IP header. This is why the AH and ESP headers are located between the IP header and the TCP or UDP header in Figure 5-48.

Figure 5-41 illustrates how IPsec's Tunnel mode differs from Transport mode. In Tunnel mode, IPsec encrypts the entirety of the IP packet (the IP header and payload) created at the Network layer. Here, IPsec is not integrated with IP (like it is in Transport mode). IPsec's AH and ESP headers are added before the original IP header, which means that the original IP packet is encapsulated in a new IP packet. Securing and encapsulating a complete IP packet in another IP packet forms a virtual "tunnel" between IPsec endpoints.

Authentication and encryption of IPsec packets is done using *security association (SA)* records stored in a database at each endpoint. SA records identify encryption algorithms, keys, IPsec modes, destination addresses, and other data/information about connections and communications between endpoints.

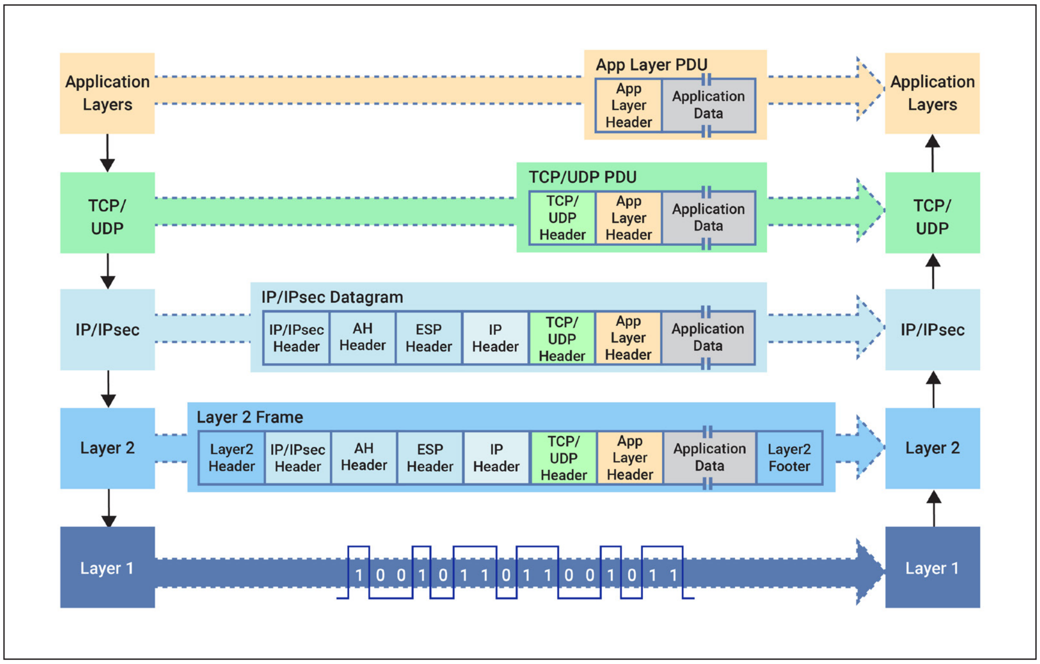


Figure 5-47: IPsec Transport mode operations.

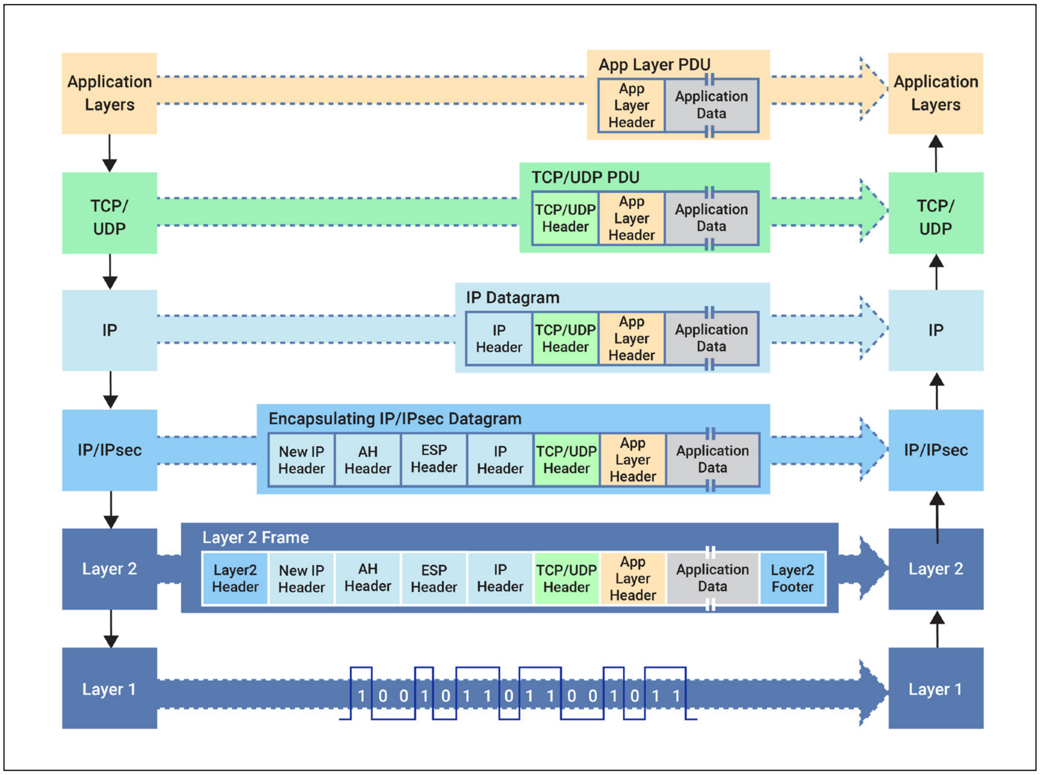


Figure 5-48: IPsec Tunnel mode operations.

SA records may be configured manually via keys that are preshared by endpoints. Alternatively, they can be automatically configured using Internet Key Exchange (IKE) or IKE version 2 (IKEv2). IKE uses a well-known protocol and public key infrastructure (PKI) to securely share encryption keys. IKE handles authentication and ICV calculations, encryption negotiation, and encryption key exchange. Encryption algorithms, keys, and PKI are discussed in Chapter 11.

The greatest security is provided when AH and ESP are used in IPsec's Tunnel mode. This secures and protects IPv4 and IPv6 communications between distributed business locations over the Internet or between an organization and its business partners or remote workers. Though less secure, IPsec's Transport mode is suitable for transferring sensitive data between hosts in the same network.

In sum, IPv6 has an extensive (and evolving) set of features that capitalize on lessons learned from the evolution and use of IPv4. IPv6's required use of IPsec and its ability to cryptographically secure the generation of device addresses provides anonymity and security in today's networks.

CHAPTER 6

Appendix 6-1: IEEE 802.2 Header Fields

The Data Link layer's LLC sublayer adds an LLC header to the Network layer PDU (see Figure 6-37). The MAC sublayer adds a MAC header that includes 48-bit (6 byte) physical (MAC) addresses, not logical addresses. Basically, the MAC sublayer implements physical addressing and uses MAC (physical) addresses to transfer data to another host in the local network. It also adds a trailer field that is used for error checking.

Two of the most important fields in the LLC header are the Destination Service Access Point (DSAP) field and the Source Service Access Point (SSAP) field (see Figure 6-38). These fields facilitate Internet operations in the local network. The DSAP is typically an 8-bit field that

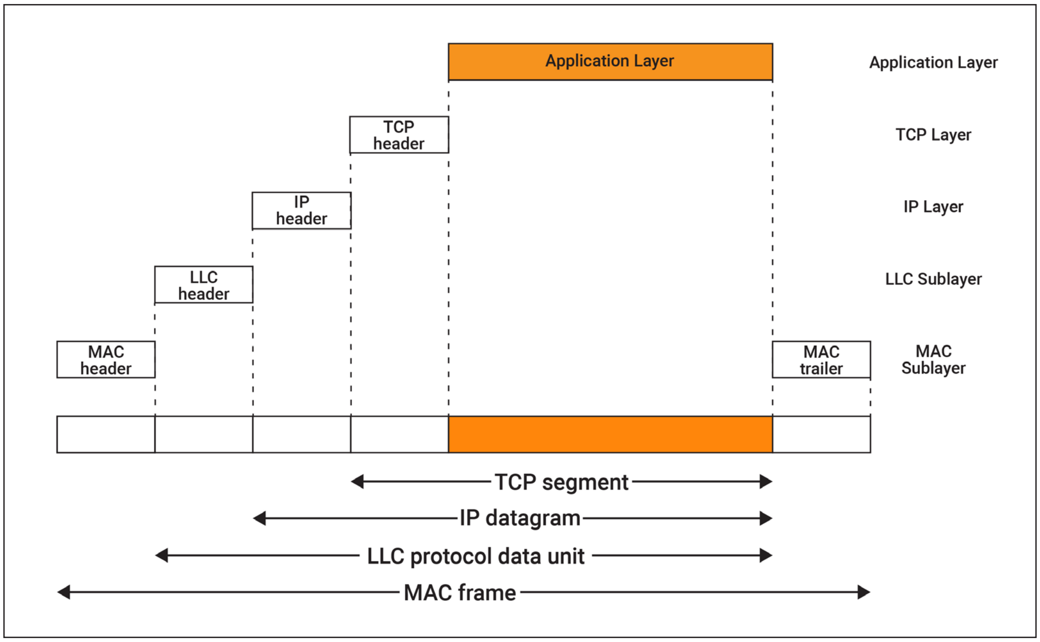
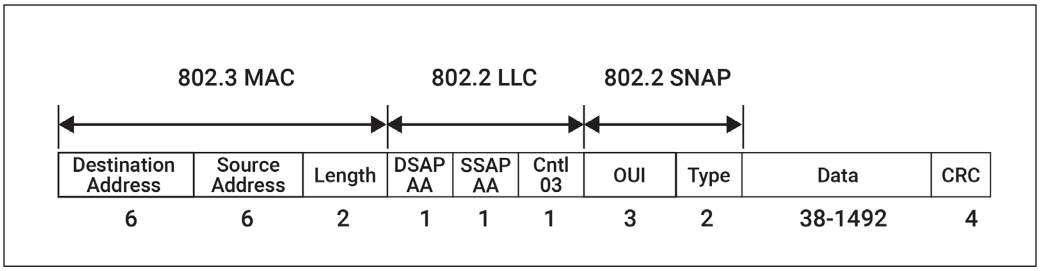


Figure 6-37: LLC and MAC PDUs.



Destination Address	Source Address	Length	DSAP AA	SSAP AA	Cntl 03	OUI	Type	Data	CRC
6	6	2	1	1	1	3	2	38-1492	4

Figure 6-38: The Ethernet SNAP format facilitates TCP/IP use in local networks.

identifies the Network layer entity that should receive the LLC PDU. This may be an individual or group identifier. The SSAP field is an 8-bit field that identifies the Network layer entity that creates the LLC PDU and indicates whether it is a request or response PDU; it essentially identifies the LLC PDU's creator.

The Control field (Cntl) is used to identify the communication mode: Type 1 is a connectionless datagram service with unacknowledged frames; Type 2 is for connection-oriented operations that use sequence numbers for reliable frame delivery; and Type 3 is an acknowledged connectionless service used in point-to-point connections. The Organizationally Unique Identifier (OUI) and Type fields are often called the SNAP (Subnet Access Protocol) extensions. These fields are used to identify the type of IP PDUs included in the Data field for IEEE (e.g., 802.3 or 802.5) or proprietary standards.

Appendix 6-2: Fournier Transforms

Students in electrical engineering and other electronics disciplines learn how digital signals are created by combining analog waveforms when they study Fournier Analysis and Fournier Transforms.

In Figure 6-39, the original wave (top left) is a very rough approximation of the digital (square) wave that overlays the analog wave (top right), but combining harmonics of the original wave with the original wave itself (middle and bottom left) reshapes the wave in ways that cause it to look

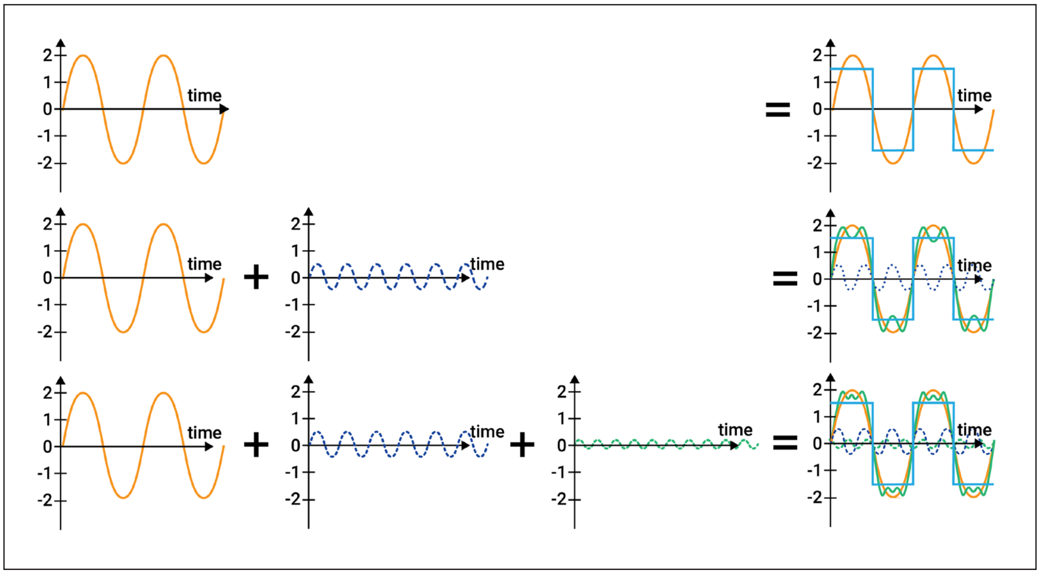


Figure 6-39: A conceptual depiction of a Fournier Transform and how waveform harmonics combine to create digital signals.

more like a digital signal (middle and bottom right). When enough harmonics are combined with the original, the result is indistinguishable from a digital signal.

Knowing this facilitates understanding why digital signals are affected by analog noise waveforms. They can be affected because they are made by combining analog wave harmonics that approximate discrete voltage pulses (digital signals).

Appendix 6-3: Physical Layer Standards

ANSI/TIA/EIA-568

The ANSI/TIA/EIA-568 standard has evolved over time to its current form (TIA 568.2D). This evolution has been driven by the desire to do the following:

- Provide network cabling specifications that enable businesses to avoid being locked into a single vendor's products
- Provide direction for the design and deployment of the telecommunications equipment and cabling in commercial facilities
- Specify structured cabling systems that are generic enough to support both data and voice
- Provide technical and performance guidelines for planning and installing structured cabling systems technology

ISO/IEC 11801

The International Organization for Standardization (ISO) and International Electrotechnical Commission (IEC) also develop telecommunication and IT standards. ISO/IEC 11801 addresses structured network cabling systems in commercial buildings. This standard is reviewed and revised every five years. A major update was released in 2017, and another is expected in 2023. Like ANSI/TIA-568, it provides specifications for both copper and fiber optic cabling used for data and voice services. It also addresses cabling for video services.

Appendix 6-4: Adapter Components and Operations

Adapter circuitry includes a controller that works like a mini CPU or microprocessor (see Figure 6-40). The controller is the core part of an adapter that determines how it performs. It implements a specific Data Link standard (e.g., Ethernet or Wi-Fi) on a specific Physical layer communication medium (e.g., a twisted-pair wire or fiber optic cable) and thereby provides a base for supporting the TCP/IP protocol stack in the local network.

Network adapters also include a direct memory access (DMA) subsystem that is responsible for moving packets to and from the main memory (RAM) of the host's CPU. The DMA subsystem is a shared subsystem that can be directly read from or written to by both the adapter and the host's CPU. This capability facilitates data transfer to/from the adapter and the main CPU over internal buses.

Network adapters operate in one of two modes: cut-through or store-and-forward. When the *cut-through mode* is used, a sending adapter begins transmitting a frame before it is fully read from the adapter's memory, and a receiving adapter can begin storing the frame before it is entirely received.

In the *store-and-forward mode*, an entire frame is read before the adapter begins transmitting it, and receivers do not pass a frame to the device until it is entirely received.

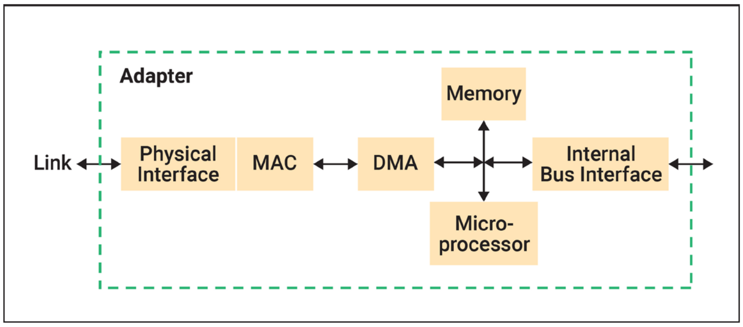


Figure 6-40: Conceptual depiction of network adapter components.

In the cut-through mode, adapters have lower transfer delays, and (thanks to DMA) can sometimes avoid storing frames in the adapter's local memory. However, adapters operating in this mode may pass errored or partial frames to/from the host's CPU, which typically does not happen in the store-and-forward mode.

Appendix 6-5: VLAN Accommodation in Ethernet LANs

When VLANs are supported in Ethernet networks, IEEE standard 802.1Q comes into play. IEEE 802.1Q is the networking standard for virtual LANs (VLANs) on an Ethernet network. 802.1Q defines a VLAN tagging system for Ethernet frames and the procedures that switches and other data communication equipment use for handling frames with VLAN tags. 802.1Q adds a 32-bit field between the Source Address and Type fields (see Figure 6-41) and extends the maximum frame size to 1522 bytes.

As Figure 6-41 illustrates, a VLAN tag field has four segments:

- The Tag Protocol ID (TPID) field identifies the frame as an 802.1Q-tagged frame.
- The User Priority field, also known as the Priority Code Point (PCP) field, identifies the frame's priority level; different service classes (priorities) can be identified for different types of data.
- The Canonical Format Indicator (CFI), which is also known as the Drop Eligible Indicator (DEI), can be used in conjunction with the User Priority field to identify frames that can be dropped when the network is severely congested.
- The VLAN ID field specifies the VLAN to which the frame belongs.

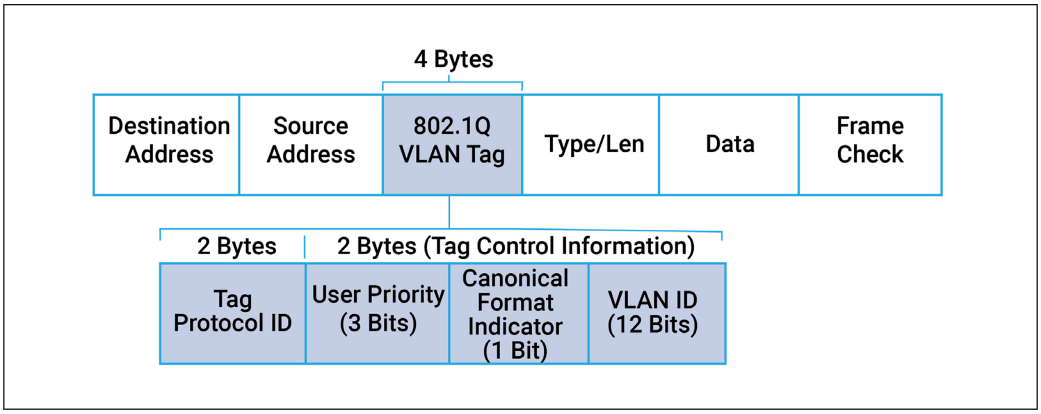


Figure 6-41: VLAN tags in 802.3 frames.

Appendix 6-6: Data Link Layer Security Threats and Remedies

Some attackers attempt to compromise Data Link layer software and protocols. Vigilant network monitoring by network administrators is needed to identify and thwart Data Link layer attacks, including ARP cache poisoning/spoofing, MAC address flooding, MAC address spoofing, spanning tree attacks, and VLAN hopping (see Table 6-16).

These attacks exploit processes, protocols, and software that support physical (MAC) addresses or VLAN tags, and network administrators should be vigilant in monitoring these processes, protocols, and software. They are often precursors to other and potentially more damaging types of attacks. Switch port security features and settings can prevent several of them.

Table 6-16: Examples of Data Link layer security threats.

Data Link Security Threat	Brief Description
ARP cache poisoning (a.k.a. ARP spoofing or flooding)	In this type of attack, the attacker sends bogus ARP messages to a LAN. The attacker requires direct access to the targeted LAN, and the LAN must use ARP. Generally, the goal is to associate the attacker's MAC address with the IP address of a host in the network or the default gateway and to cause traffic meant for that IP address to be sent to the attacker's device. When ARP poisoning succeeds, the attacker may intercept frames, modify traffic flow in the LAN, or stop traffic from reaching the LAN. This is often a precursor to other attacks, such as denial-of-service (DoS), man-in-the-middle (MITM), and session hijacking.
MAC address flooding	In this type of attack, an attacker floods a LAN switch with fake MAC addresses to compromise its security. The attacker saturates (floods) the switch with a huge number of forwarding requests, each with a fake MAC address. When successful, the switch's forwarding table is modified when the table reaches its storage limit and begins removing old addresses and replacing them with the fakes. This is sometimes called a MAC table overflow attack. When legitimate users attempt to send frames to a compromised switch, the attacker is positioned to capture all of the ingoing and outgoing frames and to sniff any confidential data they contain. MAC flooding can be prevented on switches with port security features and settings.
MAC address spoofing	In this type of attack, the attacker hunts the network for valid MAC addresses that can be used to circumvent access control measures by posing as a device with a valid MAC address. Sometimes, this results in the substitution of the attacker's MAC address for that of the default gateway. In this case, the attacker can sniff and copy data passed to the default gateway; this provides details about applications being used and IP addresses. Firewalls often include services for identifying suspected MAC address spoofing that provide some protection against these types of attacks.
Spanning tree attacks	The Spanning Tree Protocol (STP) is designed to close switching loops in local networks. It operates by moving switch ports into blocking or forwarding states depending on the network segments they connect to. Because STP lacks authentication mechanisms, it is vulnerable to attack. Disrupting a switch's spanning trees destabilizes its forwarding table and can cause broadcast or multicast traffic storms that flood the network with frames. Denying access to STP-enabled ports can prevent STP attacks. This is done by disabling STP on user access, enabling port security on all user ports, and physically isolating switches.
VLAN hopping attacks	There are several types of VLAN attacks, but VLAN hopping is among the most common. *VLAN hopping* is a method of attacking VLAN network resources by sending packets to an end system that is not a member of the VLAN. When it succeeds, the attacker has a beachhead that can be leveraged to gain access to other VLANs. VLAN vulnerabilities stem from their key features, which include enabling network administrators to partition a switched network into VLANs without needing to make significant changes to network infrastructure. VLANs generally improve security by restricting users to specific subnetworks, and if a VLAN is compromised, attack damage is limited to that VLAN. However, when a VLAN is breached, an attacker can leverage VLAN trunking (a process that VLAN switches use to look for channels to send or receive data) to compromise security protocols and infiltrate (hop to) other VLANs to steal user passwords and other sensitive information. VLAN hopping can also be used to modify or delete data, or install malware that infects the network with viruses, worms, or Trojans. After a VLAN breach, attackers may double tag VLAN frames to enable VLAN hopping. Proper switch configuration can mitigate the effects of VLAN attacks. This starts by turning off the autotrunking feature on VLAN switches and implementing port security.

CHAPTER 7

Appendix 7-1: Hybrid ARQ (HARQ)

Hybrid ARQ (HARQ) uses a combination of FEC (forward error correction) and ARQ. Errors are corrected by FEC when possible, and when FEC cannot correct all errors, retransmission is requested. The errored frame and all retransmissions of the original frame are retained by the receiver to increase the receiver's chances of correctly reconstructing the frame's original contents. HARQ is compared to ARQ and FEC in Figure 7-39.

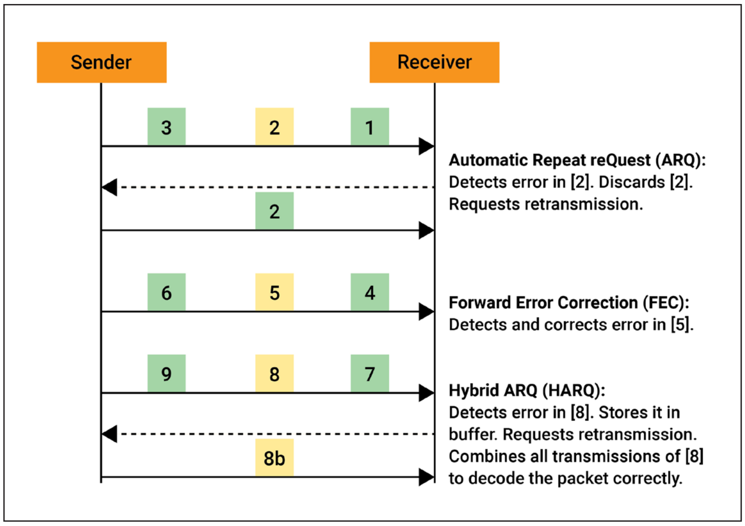

Sender
Receiver
3
2
1
Automatic Repeat reQuest (ARQ):
Detects error in [2]. Discards [2].
Requests retransmission.
2
6
5
4
Forward Error Correction (FEC):
Detects and corrects error in [5].
9
8
7
Hybrid ARQ (HARQ):
Detects error in [8]. Stores it in
buffer. Requests retransmission.
Combines all transmissions of [8]
to decode the packet correctly.
8b

Figure 7-39: ARQ, FEC, and HARQ network error controls.

HARQ is used in Bluetooth networks and 5G cellular networks. It may also be used in future generations of Wi-Fi networks.

Appendix 7-2: Wi-Fi History

Wi-Fi's roots trace back to 1985, when the FCC opened several unlicensed bands of wireless spectrum (900 MHz, 2.4 GHz, and 5.8 GHz). By that time, these bands were already being used by microwave ovens and other equipment, so using them for communications required taking steps to protect communications from interference created by other sources in these ranges. One of the wireless communications solutions was spread spectrum technology, where a radio signal is transmitted over a wide range of frequencies instead of a single, well-defined frequency range. Spread spectrum makes a signal more difficult to detect (and therefore useful for military applications) and less susceptible to interference.

In 1988, NCR Corporation began investigating using unlicensed spectrum to wirelessly connect cash registers to networks and approached the IEEE with the idea of establishing a new networking standard. In response, the IEEE established the 802.11 committee, and in 1997, this committee issued the basic 802.11 standard (with a data transfer rate of 2 Mbps and spread spectrum transmission). Two variants, 802.11b (in the 2.4 GHz band) and 802.11a (in the 5.8 GHz band), were in place by January 2000.

Wi-Fi got a boost when Apple introduced it as an option for its iBook computers in late 1999. Other computer makers quickly copied Apple, and Wi-Fi caught on with consumers. Since that time, Wi-Fi has been the dominant form of home networking. More recently, the IoT has extended Wi-Fi networks to accommodate a wide range of devices and applications. The 802.11 Wi-Fi standard has also continued to evolve, with each new generation bringing higher speed connections and improved capabilities. Recently opened unlicensed spectrum in the 6 GHz band has also been addressed in recent 802.11 amendments.

Appendix 7-3: Wi-Fi DS Station Services

In addition to distribution, integration, and association services, Wi-Fi DS supports *station services (SSs)*. These services are implemented by APs and include authentication, de-authentication, privacy, and MAC service data unit (MSDU) delivery. They are briefly described in Table 7-15.

The 802.11 standard identifies three types of frames and three frame classes. The three types of frames are control frames, management frames, and data frames.

Table 7-15: DS station services.

Service	Brief Description
Authentication	Because WLANs have limited physical security to inhibit unauthorized access, the 802.11 standard addresses authentication services to control LAN access and provide a level of security comparable to what is available for wired LANs. Every station in an IBSS or a BSS or ESS network must use the authentication service prior to establishing a connection (associating) with another station with which it will communicate. To perform authentication, a station sends an *authentication frame* to the other station or AP. This takes one of two forms: • *Open system authentication*. The station that wants to authenticate sends an *authentication management frame* that contains its identity to the other station. The receiving station replies with a frame indicating whether it recognizes the identity of the authenticating station. • *Preshared key (PSK) authentication*. When Wired Equivalent Privacy (WEP), Wi-Fi Protected Access (WPA), and WPA2 Wi-Fi encryption algorithms are used, stations authenticate through shared knowledge of a secret key. This form of authentication assumes that each station possesses a secret shared key that is passed through a secure channel external to the 802.11 network.
De-authentication	To disassociate from another station, a station uses the de-authentication service. This is a notification that cannot be refused by the recipient. A station performs de-authentication by sending an authentication management frame to advise the other station or AP of the termination of authentication.
Privacy	In a WLAN, all stations and other devices can hear data traffic within their transmission ranges. This poses significant security risks, and to address this problem, the IEEE 802.11 standard offers a privacy service option to bring a WLAN's security closer to that of a wired LAN. The privacy service protects the privacy of data frames, and some authentication management frames, to reduce the risks of eavesdropping. This service is based on the 802.11 WEP, WPA, and WPA2 algorithms and is used to encrypt transmitted frames.
MAC Service Data Unit (MSDU) delivery	An *802.11 MSDU* is the payload (data field) of an 802.11 data frame. It carries the Network Layer PDU, which in turn carries the Transport and Application layer PDUs. So, a MAC protocol data unit (MPDU) is essentially a technical term for a WLAN frame. An MPDU consists of a frame header, body, and trailer with the MSDU payload encapsulated in the frame body. MSDU transfer and delivery is considered a primary purpose of a WLAN.

Class 1 frames include all three frame types. RTS (request to send), CTS (clear to send), ACK (acknowledgment), and CF (contention-free) frames are examples of Class 1 control frames. Class 1 management frames include probe request/response frames, beacon frames, authentication frames, and de-authentication frames. Examples of Class 2 frames include association request/response frames, disassociation notification frames, and reassociation request/response frames.

These details about DS illustrate that there is much involved with enabling wireless devices to connect and roam within a Wi-Fi network. Ultimately, the ability of users to transfer and receive data via wireless connections is paramount, but there is much happening in the background that makes this possible, and DS is an important part of that.

Appendix 7-4: DCF and PCF Media Access Controls

Distributed Control Function (DCF)

In 802.11 networks, carrier sensing is the primary method used to avoid collisions. To understand how PCF differs from DCF, a clear picture of 802.11 carrier sensing is needed.

Carrier sensing involves a station's monitoring of ("listening" to) the medium to determine whether it is being used by another station or is free for the station to use and transmit its data.

Basically two mechanisms are used to sense the medium:

- *Check the Physical layer for carrier presence*. This is accomplished by requiring stations to measure the amount of energy received on the communication channel. When that energy is above a threshold, the sensing node determines that another node is currently transmitting and that it must remain silent.

In 802.11 networks, interframe spacing is used alongside carrier sensing to ensure that the shared medium is not busy. When a node is sensing the channel, it must sense that the channel is free for the duration of the *DCF interframe spacing (DIFS) period*. The *short interframe spacing (SIFS) period* is used as a measure of the wait time between the RTS, CTS, DATA, and ACK frames transferred between transmitters and recipients. Since the SIFS is always shorter than the DIFS, requiring the channel to be sensed as being free for the duration of the DIFS ensures that another node does not incorrectly determine that the channel is idle while the RTS-CTS or DATA-ACK handshake is under way and ensures that priority is given to any transmission in progress.

- *Network allocation vector (NAV)*. To complement (or avoid) Physical layer carrier sensing, the NAV is used to inform other nodes how long the current node will need the channel. The NAV is a *virtual carrier-sensing* mechanism—a logical abstraction of carrier sensing that limits the need for physical carrier-sensing.

  Each 802.11 frame header contains a Duration field that specifies the frame's required transmission time and the amount of time the medium will be busy. Stations listening to the medium read the Duration field and set their NAV, which is a timer/counter indicating how long they must wait before trying to use the medium. The NAV counts down to zero at a uniform rate. When the counter/timer is not zero, the virtual carrier-sensing indication is that the medium is busy; when it is zero, the indication is that the medium is free. In IEEE 802.11, the NAV represents the number of microseconds the sending STA intends to hold the medium while transmitting the frame (up to a maximum of 32,767 microseconds).

Figure 7-40 illustrates how DCF is orchestrated in 802.11 networks. When a station sends a request to send (RTS), the receiver (usually an AP) waits one SIFS before sending clear to send (CTS). The sender will then wait one SIFS before sending its data. The receiver will wait another SIFS before sending an acknowledgment (ACK). So, in Figure 7-40, NAV is the duration from the first SIFS to the ending of ACK. During this time, the medium is considered busy by any stations overhearing the frame exchanges.

For stations with data to transmit, it is better to use both physical carrier sensing and virtual carrier sensing to determine whether the medium is idle. This combination best contributes to collision avoidance.

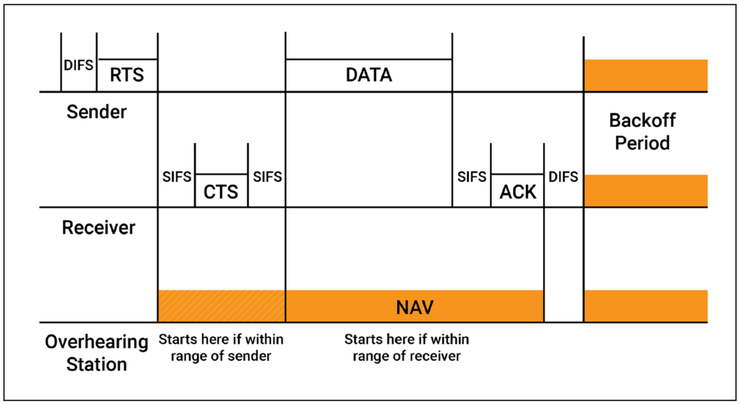


Figure 7-40: Key elements in CSMA/CA's DCF.

Point Control Function (PCF)

Hidden and exposed nodes are problematic for any carrier-sensing–based media access control, including CSMA/CA. As noted in section 7.1.3, a *hidden node* is a station that is within range of the receiving node (typically an AP) but not within range of the transmitting station. Hence, its transmissions will interfere with ongoing data transfers between an out-of-range transmitter and an in-range receiver, and since it cannot hear the transmission from the out-of-range source, it will sense that the medium is idle, transmit an RTS frame, and cause data collisions.

An *exposed node* problem exists when a station is within range of a sending station but is outside the range of a receiving node. This node will determine that any transmission it creates will interfere with any transmission of a sending node whose transmissions it overhears. In reality, since the receiver is outside the exposed node's range, the exposed node should be free to transmit.

802.11 uses the RTS and CTS packets to resolve both of these issues. If a node hears a CTS, it knows that it is in range of the receiver and therefore cannot transmit without causing a collision. If a node hears an RTS but does not hear a CTS, it knows that it is an exposed terminal and is free to transmit.

When hidden nodes exist in a BSS, the AP is the only device with guaranteed communication with all BSS stations. Because of this, the AP is the logical choice to manage the WLAN's shared medium using a controlled-access approach. Controlled-access media access controls, introduced in Chapter 5, include polling and token passing protocols. To provide an opportunity for hidden nodes in a BSS to use the shared medium, 802.11 infrastructure networks may include time periods for both controlled-access and contention-based media access controls.

The PCF is often described as an optional media access control method for infrastructure WLANs; it is not used in IBSSs. To be used, it must be supported by both an AP and a station. When implemented, the AP alternates between PCF and DCF, and during the PCF interval, the AP uses polling to determine which station has the right to transmit. The PCF is often described as being implemented on top of DCF, which is the default media access control method for Wi-Fi.

Figure 7-41 illustrates what happens within the contention-free period (CFP) of a superframe. The CFP begins with the point coordinator (the AP) broadcasting a beacon that informs all stations of the PCF and the start of the CFP. After a short interframe spacing (SIFS) period, it polls one of the stations (T1) in the BSS. If the poll recipient has data to transmit, it transmits a frame to the PC along with a poll acknowledgment (ACK). The AP next sends a poll to a second station (T2) and sends an acknowledgment of frame receipt (ACK1) to station 1. If T2 fails to acknowledge the poll receipt within a DIFS (DCF interframe spacing) period, the AP sends a poll to a third station (T3). If T3 has data to transmit, it sends its frame to the AP as well as a poll acknowledgment. The CFP concludes with the AP sending a frame acknowledgment (ACK3) to T3 and broadcasting a CFP end announcement.

The superframe's contention period (CP) begins when the CFP ends; stations with data to transmit that were not polled during the CFP have another opportunity to transfer their data during the superframe's CP.

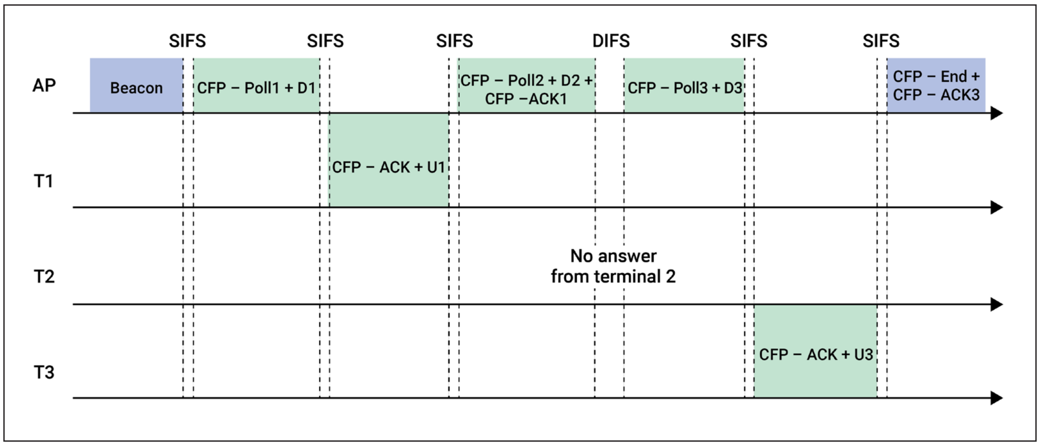


Figure 7-41: An example of PCF operation during a superframe's contention period.

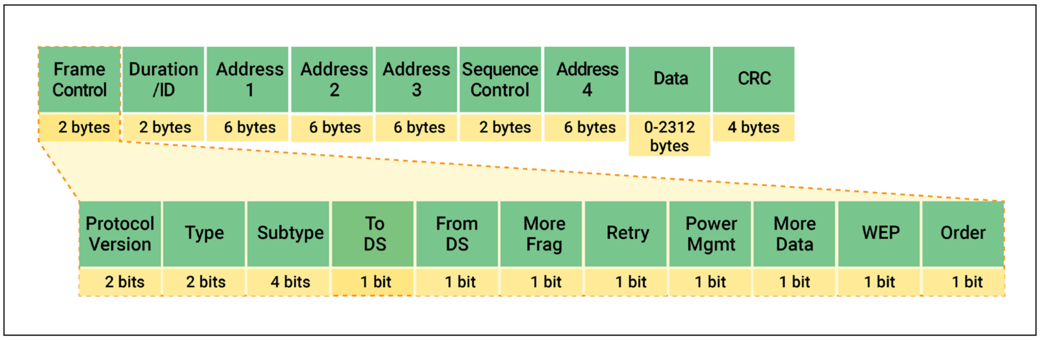


Figure 7-42: Fields within the 802.11 Frame Control field.z

PCF and DCF illustrate how acknowledgments are used to provide reliable frame transfers. Stop-and-wait ARQ is used in DCF. The sender transmits a single frame and waits for an ACK or NACK [negative acknowledgment]. If the receiver is unable to use FEC to reconstruct a corrupted frame, it sends a NACK, and the sender will retransmit the frame before attempting to transmit another.

Appendix 7-5: Wi-Fi Frame Format Details

The Frame Control field in a Wi-Fi frame includes multiple subfields (see Figure 7-42), and the meaning of the address fields depends on the contents of some of these subfields.

The meaning of each address in a Wi-Fi frame depends on the settings in the To DS and From DS fields (see Table 7-16). The contents of a particular address field may specify:

- BSSID—the MAC address of the AP in the BSS
- Source address (SA)—the MAC address of the frame creator/sender
- Destination address (DA)—the MAC address of the frame recipient
- Transmitting station address (TA)—the MAC address of the frame transmitter
- Receiving station address (RA)—the MAC address of the transmission recipient

The DS field settings make sense when viewed as DS (distribution system) involvement in a transmission. For example, in the first row of Table 7-16 (the 0,0 row), the frame is not directed to and is not coming from the DS. This means that the frame is being sent from one station (the SA) in the BSS to another in the same BSS (the DA) via the AP (the BSSID) without DS involvement. In the second row (the 0,1 row), the DS field settings indicate that the frame is addressed to a station in the BSS (the DA) from a device in another BSS (the SA) via the DS, with the AP (BSSID) serving as the relay point between the DS and DA. In the third row (the 1,0 row), the settings indicate that the frame is being sent from a station in the BSS (the SA) to a station in another BSS (the DA) via the DS, with the AP (BSSID) serving as the relay point between the sending station and the DS. The settings for the last row, the 1,1 row, are only used in 802.11 ad hoc networks (IBSSs), so no AP (BSSID) is involved in the transfer. Here, the RA and DA addresses are the same, so are the TA and SA addresses.

Table 7-16: 802.11 Address Field Meanings for Different DS Field Settings.

To DS: Setting	From DS: Setting	Address 1	Address 2	Address 3	Address 4
0	0	DA	SA	BSSID	Not used
0	1	DA	BSSID	SA	Not used
1	0	BSSID	SA	DA	Not used
1	1	RA	TA	DA	SA

Other important fields in Wi-Fi frames include the payload (Data), CRC, Duration, and Sequence Control (SC) fields:

- The Data field in Figure 7-42 includes the Network layer PDU, which in turn includes the Transport and Application layer PDUs.
- The CRC field includes a 32-bit CRC error check value. As noted previously, Wi-Fi uses FEC to reconstruct mildly corrupted frames but uses CRC-32 as a backup to trigger frame retransmission when FEC is unable to fully reconstruct the original data.
- The 4-byte Duration (Duration/ID) field indicates the time (in microseconds) that the shared medium will be used to transmit the frame. This field is used for virtual carrier sensing in DCF, and it may include an association, or connection, identifier.
- The 2-byte (16-bit) Sequence Control (SC) field has two subfields: sequence number (12 bits) and fragment number (4 bits). Some 802.11 acknowledgment mechanisms buffer both original and retransmitted frames, and the sequence number helps receivers filter duplicate frames and reassemble them, or fragmented frames, in their correct order. This contributes to frame delivery reliability.

The subfields in the Frame Control field of a Wi-Fi frame are briefly described in Table 7-17.

As you have probably observed, understanding media access control in Wi-Fi networks requires knowledge of 802.11 architectures (BSS, ESS, IBSS, MBSS), distribution system (DS) services, physical and logical topology, access point (AP) association, and DCF and PCF. These elements converge in various ways in the contents of the fields in 802.11 frames.

Table 7-17: Subfields in a Wi-Fi frame's Frame Control Field.

Frame Control Subfield	Brief Description
Protocol Version	This 2-bit field indicates the IEEE 802.11 protocol version: 0 indicates the 2007 protocol version; 1 indicates the 2020 protocol version.
Type	This 2-bit field identifies the frame type: management (00), control (01), data (10).
Subtype	This 4-bit field indicates the frame's subtype, e.g., 0000 for association request or 1000 for beacon.
To DS	When set to 1, this 1-bit field indicates that the frame is being sent to the DS.
From DS	When set to 1, this 1-bit field indicates that the frame is coming from the DS.
More Frag (More Fragments)	When set to 1, this 1-bit field means that the frame is followed by other fragments.
Retry	When set to 1, this 1-bit field indicates that the frame is a retransmission of an earlier frame.
Power Mgmt (Management)	This 1-bit field indicates the mode of a station after successful transmission of a frame. If set to 1, the field indicates that the station goes into power-save mode; if set to 0, the station stays active.
More Data	This 1-bit field is used to indicate to the frame receiver that the sender has more data to send than the current frame. It can be used by an AP to indicate to a station in the power-save mode that the AP has more packets in its buffers to send it. Alternatively, it can be used by a station to indicate to an AP after being polled that more polling is necessary because the station has more data to transmit.
WEP	When set to 1, this 1-bit field indicates that WEP, the former standard security mechanism of 802.11, is applied.
Order	When set to 1, this 1-bit field indicates that received frames must be processed in order.

Appendix 7-6: OFDM and OFDMA

To illustrate, let's assume that in a single channel each bit in a data stream is represented by a symbol (signal) that lasts 1 nanosecond (ns) and that there is a 0.25 ns spacing between each symbol. To transfer 4 bits, 5 ns would be needed. If OFDM splits the channel into four subcarriers, assigns 4 bits in the data stream to the four subcarriers, where each bit is represented by a symbol that lasts 4 ns and each symbol is separated from the next by 1 ns spacing, the overall data rate is

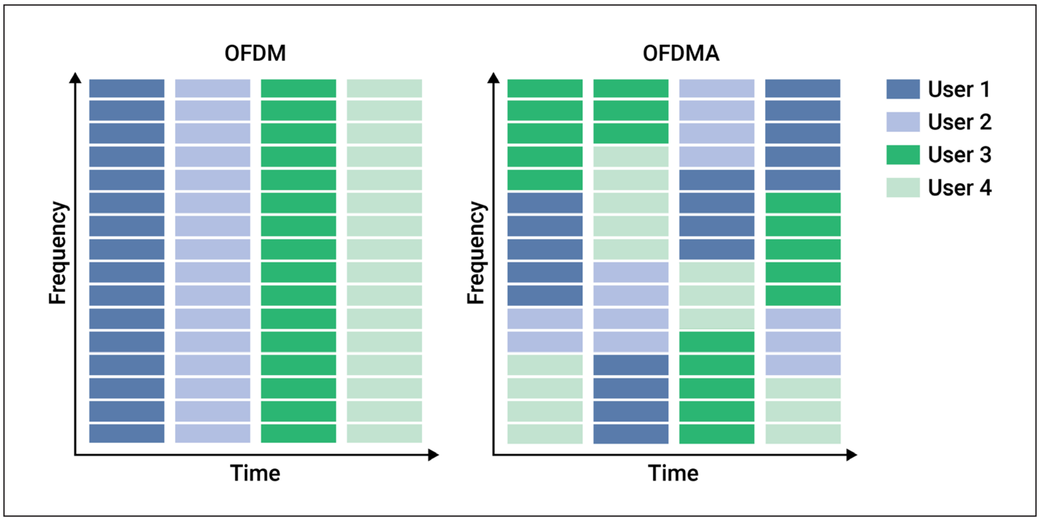


Figure 7-43: A conceptual depiction of the differences between OFDM and OFDMA.

the same: 4 bits every 5 ns. However, the transmission would be less susceptible to interference, so signal integrity would be higher.

Relative to single-channel data transmission, interference is also less of an issue because of the availability of several channels. OFDM assists forward error correction (FEC) by spreading out the data and facilitating FEC's ability to compensate for small errors. Narrowband interference on a single subchannel will not affect the other channels and therefore will not cause a major disruption. OFDM also facilitates the correction of multipath echoes and errors. Lower subchannel data rates and greater intersymbol spacing reduce the potential for intersymbol interference.

OFDM has two main disadvantages when compared to single-channel transmission systems. The first is the need for closely synchronized transmitters and receivers. Timing on signal modulators and demodulators must be closely matched to ensure the tight tolerances needed to resist interference. This also makes transmissions more sensitive to Doppler shifts and less effective in devices or vehicles in motion.

Attempts to visualize the differences between OFDM and OFDMA often result in depictions similar to Figure 7-43, in which users are allocated the full range of a channel's subcarriers to transmit their data for OFDM in each time slice. In contrast, for OFDMA, multiple users are supported on different subcarriers in each time slice, and the size of the RU for a user may vary across time slots.

Appendix 7-7: Wi-Fi Security Threats and Controls for Home and Public Networks

Because today's enterprise networks extend to residential Wi-Fi networks in employee's homes, and because remote workers sometimes rely on nonsecure (public) Wi-Fi networks for Internet access, security issues and precautions for these networks cannot be overlooked. Examples of security threats for these networks are identified in Table 7-18.

Like attacks on Wi-Fi networks in business facilities, several patterns can be observed in attacks on public and residential networks. First, attackers often exploit *rogue (unauthorized) APs* in their Wi-Fi attacks. This happens in evil twin, man-in-the-middle (MITM), Karma, and channel interference attacks. Second, attackers often exploit unsecured APs in public settings and unsecured residential Wi-Fi routers. Third, attackers often capitalize on mobile device tendencies to connect to Wi-Fi networks with the strongest signals. Fourth, compromising a public or residential net-

Table 7-18: Examples of security threats for public and residential Wi-Fi networks.

Security Threat	Brief Description
Channel interference attack	Channel interference attacks degrade the performance of a Wi-Fi network. They occur when two or more Wi-Fi networks are operating on the same channel (or adjacent channels), causing interference, reduced signal quality, slower data transfer rates, dropped connections, and other issues. There are different types of channel interference attacks, including co-channel interference and adjacent channel interference. Both cause interference and reduced signal quality in Wi-Fi networks, so an attacker can use either to intentionally disrupt the performance of a Wi-Fi network.
Evil twin attack	In an evil twin attack, the attacker gathers information about a public network AP, then sets up their device to impersonate it. The attacker broadcasts an SSID that resembles the public network's but uses a stronger signal than the one generated by the legitimate AP, which entices unsuspecting users to connect to the attacker's device. Once connected, the attacker can use specialized tools to read data that victims transfer over the Internet; this may include credit card numbers, username and password combinations, and other personal information.
Karma attacks	Karma attacks sniff, probe, and attack Wi-Fi networks using man-in-the-middle (MITM) methods.
Malware and Ransomware	When connecting to a home or work network, firewalls and anti-malware controls are likely to be deployed, but these protections are often lacking on public Wi-Fi hotspots. Without protection from antivirus software or web filters, malware can be downloaded to a mobile device without the user knowing it.
Mobile device theft	By physically stealing your device, attackers could have unrestricted access to its data and cloud accounts. Personal devices should not be left unattended in public spaces.
Piggybacking	If you fail to secure your residential network, anyone with a wireless-enabled device in range of your AP can use your network to access the Internet. Nefarious users may use your network to conduct illegal activity, monitor and capture your data traffic, or steal personal files.
Rogue hotspots	These are used to fool devices into connecting to a false AP to facilitate evil twin, Karma, MITM, and other types of attacks.
Shoulder surfing	In public areas, malicious actors can simply glance over your shoulder as you type. By simply watching or making a video recording, they can steal sensitive or personal information.
Unauthorized file access	An unsecured public Wi-Fi network combined with unsecured file sharing could allow a malicious user to access directories and files on personal devices.
Wardriving	Wardriving is a type of piggybacking. The broadcast range of an AP (Wi-Fi router) can extend outside your residence. Some individuals have made a hobby out of driving through cities and neighborhoods with a specially equipped laptop computer—sometimes with powerful antennas and Wi-Fi network reconnaissance software—searching for unsecured wireless networks.
Wireless sniffing	Many public APs are not secured, and the traffic they carry is not encrypted. This can put mobile device users' communications or transactions at risk. Malicious actors could use sniffing tools to obtain unencrypted sensitive information, such as passwords or credit card numbers. They may capture encrypted information and try to decrypt it later.

work opens the door to other types of attacks. And fifth, Wi-Fi users are often unaware that their networks or data are being surveilled or stolen.

Employee use of residential and public Wi-Fi networks introduces additional vulnerabilities and attack surfaces for enterprise networks. Wi-Fi routers in home networks have a wide variety of security capabilities, and it is important for remote workers to be aware of and use them. Remote workers also need to be aware of the security issues associated with using Wi-Fi networks in public spaces.

Here are some examples of security measures for home and public Wi-Fi networks:

- Change default passwords in Wi-Fi routers. Many network devices, including Wi-Fi routers, are preconfigured with default administrator passwords to simplify setup. The default passwords are easy for attackers (and forgetful Wi-Fi router owners) to obtain online, and therefore provide minimal protection. Changing default passwords makes it harder for attackers to access the device, and replacing them with complex passwords that are changed periodically provides a level of protection that only determined attackers are likely to attempt to bypass.

- Restrict access. Use mechanisms that identify authorized network users and only allow authorized users to access the network. Wi-Fi routers can be configured to only process frames transmitted by authorized devices via MAC address filtering. Most Wi-Fi routers can also be configured for "guest" access to grant access to guests on a separate wireless channel and password. MAC address filtering can also be performed on enterprise WLANs, even those with guest networks segregated from the corporate network.
- Encrypt the network's data traffic. Depending on their Wi-Fi generation, Wi-Fi routers support encryption protocols, such as WPA2 or WPA3, that encrypt frames transferred between the Wi-Fi router and wireless devices. Many Wi-Fi routers support both the enterprise and personal modes of WPA2 and WPA3; the enterprise mode can provide stronger encryption than the personal mode.
- Protect your service set identifier (SSID). To inhibit piggybacking and wardriving, avoid publicizing your SSID. All Wi-Fi routers can be configured to protect their SSIDs by disabling the SSID broadcast. This "hides" the network and removes it from the list of available networks. Authorized users can connect to hidden networks by providing the network name and password. Disabling the Wi-Fi router's SSID broadcast makes it more difficult for attackers to find the network and reduces the network's attack surface. Additional SSID protection comes from changing the Wi-Fi router's default SSID to something unique. Allowing it to retain the manufacturer's default makes it easier for a potential attacker to identify the type of router and exploit its known vulnerabilities.
- Install firewalls. Installing a (host-based) firewall directly on a wireless device as well as on residential Wi-Fi routers can inhibit potential attackers. Attackers who succeed in circumventing the network (router-based) firewall will also have to get past host-based firewalls in devices to access their information or files.
- Keep antivirus/anti-malware software updated. Install antivirus software in wireless devices and keep virus definitions and malware signatures up to date. This can prevent malware from being unknowingly downloaded to the device when using Wi-Fi networks in public places.
- Use caution with file sharing. File sharing between devices should be disabled when it is not needed. File sharing should only be allowed in home or work networks; it should be avoided on public networks. When file sharing is enabled, it should be restricted to particular directories, ideally just one. This directory should be password protected, and access to all other directories should not be permitted. A wireless device with a shared directory that connects to a public network enables attackers to access the directory's contents, so care should be taken about what it contains.
- Keep AP and Wi-Fi router software patched and up to date. The manufacturer of your Wi-Fi router will periodically release updates and patches for the device's software and firmware. The manufacturer's website should be checked regularly for any updates or patches, since they usually include security updates.
- Check your ISP and your Wi-Fi router manufacturer's security options. The ISP and router manufacturer may have information or resources available to assist in securing the network. Also, check the customer support areas of their websites for instructions or suggestions.
- When using a public network, use a VPN to connect. Many organizations have VPN gateways, and VPNs allow employees to connect securely to their business's network when they are away from the office, even when they use a public network for Internet access. VPNs encrypt frames transmitted by senders, which are then decrypted by

receivers. Data in transit is protected. VPNs prevent ISPs or anyone else from collecting/knowing the contents of the frames being transferred, which helps protect sender/receiver privacy. As a general rule, use a VPN anytime you use a public AP to connect to the Internet.

- Confirm the name and password of a public Wi-Fi hotspot prior to use. This will help ensure that you are connecting to a trusted AP. Also ensure that all APs you connect to use at least WPA2 encryption.
- Encrypt data at rest. Most mobile devices, including laptop computers, have the ability to fully encrypt their stored data. This makes the device's data useless to attackers who cannot provide a proper password or personal identification number (PIN).
- Configure device applications to request login information before allowing access to any cloud-based information. This makes you take some additional steps each time you use an app but helps ensure that you, your device, and the cloud service are properly authenticated.

Appendix 7-8: Business Benefits of 5G Networks

5G is considered an enabler for industrial and commercial IoT (Internet of Things) applications. It is viewed as enabling many new use cases, including factory floor robotic automation, real-time predictive maintenance applications, autonomous vehicles (AVs) in warehouses, and faster and more secure network connectivity. Some potential benefits of 5G networks in business facilities are summarized in Table 7-19.

Table 7-19: Potential benefits of private 5G networks.

Potential Benefit	Brief Summary
Reducing network costs	Installing and maintaining wired connections is both labor intensive and expensive. Historically, businesses have chosen wired over wireless networks because they have offered higher speeds, greater reliability, and higher uptime than wireless networks. The advent of 5G means that private 5G networks have the potential to become a preferred option for an increasing number and variety of business applications. For example, private 5G networks may enable companies to realize faster time-to-value outcomes for their IoT initiatives. Private 5G networks also provide an opportunity to reduce or eliminate the need for costly wired installation and maintenance in new business facilities.
Improving data transmission speeds	Relative to 4G LTE, data transmission speeds in 5G networks are exponentially higher. The primary benefits of higher speeds are the ability to handle greater volumes of data and the increased practicality of applying real-time analytical processing or AI to the data. This is especially important for industrial applications and the level of factory automation needed to unleash the full potential of Industry 4.0. As machine learning and AI continue to advance, the ability to transfer the volumes of data needed and produced by these applications will require faster and more flexible networks.
Lowering latencies	Latency refers to the amount of time that elapses from the moment that data is transmitted to the moment it is received. Latencies for high-band 5G network applications are extremely low, and they facilitate a variety of applications for which low latencies are critical. For example, for AVs carrying humans, reduced latencies can save lives. Reducing latency increases AV safety by enabling virtually instantaneous communication between vehicles on the road and between AVs and traffic signals, such as stoplights. Low-latency 5G connections have the potential to enable companies to have both fast service times and worker safety. For example, a new industrial maintenance technician performing maintenance on an industrial robot could be directly supervised and instructed by a seasoned professional via augmented reality (AR) or virtual reality (VR) without having to send the expert to the robot's location.
Increasing cell (network) densities	5G network cells can support more devices than 4G cells. This means significantly more IoT sensors, wearables, and 5G devices per network. It also enables businesses to scale IoT deployments to levels that are not possible with 4G networks and most Wi-Fi networks.
Enabling new business models	5G has the potential to fuel business innovation. It provides a foundation for supporting highly skilled work from remote locations. For example, it may enable remote robotic surgeries to be performed in rural hospitals without surgeons on staff. Wearable technology innovations are likely to create new markets in health and fitness. 5G is also likely to increase the number of convenience and safety features included in consumer products.

5G business benefits are more likely to be seen in some industries than others, including manufacturing, healthcare, public transit, and retail. Private 5G networks can enable factories to greatly expand the number of sensors used to monitor and manage manufacturing processes. They enable monitoring of more assembly line performance parameters and the collection of more granular data about equipment and process performance. When combined with machine learning and AI, private 5G networks will enable continual improvement and refinement of process quality and efficiency and lower manufacturing costs.

In hospitals, private 5G networks facilitate communications between doctors and first responders that enable patients to be diagnosed before or during transport. This may involve high-resolution video and patient telemetry. 5G networks can also enable telehealth services to be expanded to rural areas.

5G networks are likely to facilitate improvements in several areas of public transit. For example, 5G enables efficient transmission of real-time information on vehicle capacity, scheduling, and routing. Transactional data for ticketing can be collected and transferred more quickly. Since 5G networks can carry large volumes of real-time performance data, public transit may be safer, even as it enables the streaming of digital signage and passenger Wi-Fi services.

5G technology can be used to expand mobile apps and enable new services to attract retail customers. For example, some retailers are developing retail apps that enable customers to "try on" clothes using virtual reality (VR) and augmented reality (AR) dressing rooms. 5G's high-volume and high-speed information flow enables retailers to offer more personalized services; it also enables IoT sensors to be used to track inventory movements and restocking needs.

CHAPTER 8

Appendix 8-1: WAN Technologies and Terminology

A major difference between a LAN or CAN and a WAN is that an organization typically does not own the circuits used to interconnect its different locations and must acquire needed circuits or services from one or more WAN service providers. WAN service providers include **common carriers**, entities that provide fee-based wired and wireless communication services to businesses and consumers. Examples of carriers include telephone networks, cable companies, and satellite services companies. Basically, WAN service providers provide the links and services used to interconnect business locations and transport data. Businesses also use carrier links to access the Internet, connect to organizations, and access external services.

Specialized terminology has traditionally been used to describe WAN connections to carrier networks (see Figure 8-30). Such terms include:

- Customer premises equipment (CPE): CPE consists of the devices and wiring used at a business location to connect to a carrier link. The subscriber (WAN services provider customer) either owns the CPE or leases it from the carrier.
- Data communications equipment (DCE): This is an Electronics Industries Alliance (EIA) term; the International Telecommunication Union (ITU) uses the term data circuit-terminating equipment. DCE consists of devices that provide an interface to connect subscribers to the WAN communication link; DCE puts data on the local loop.
- Local loop: This is the circuit that connects the CPE to the CO of the service provider. This is usually a copper or fiber cable but may be a wireless circuit. The local loop is sometimes called the last mile.

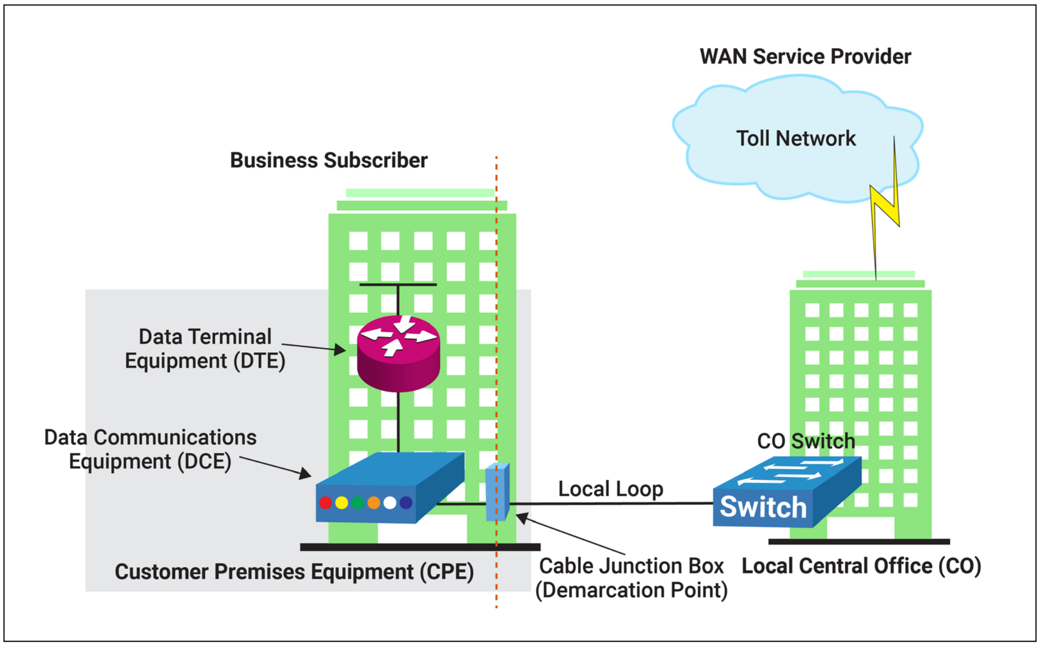

WAN Service Provider
Toll Network
Business Subscriber
Data Terminal Equipment (DTE)
Data Communications Equipment (DCE)
CO Switch
Local Loop
Switch
Customer Premises Equipment (CPE)
Cable Junction Box (Demarcation Point)
Local Central Office (CO)

Figure 8-30: WAN technologies and terminology.

- Data terminal equipment (DTE): DTE is WAN service customer devices that pass the data from the network at the customer's locations to the DCE to enable the data to be transmitted over the WAN. DTE connects to the local loop through the DCE.
- Demarcation point: This point is established in a building or another facility to separate customer equipment from service provider equipment. Basically, it is the point at which the service provider's wiring ends and the customer's wiring begins. Physically, the demarcation point may be a cabling junction box or meter located in or on an outside wall of the customer's facility that a technician can easily access. It is the physical junction of CPE wiring and the local loop. The demarcation point is the place where the responsibility for the connection changes from the customer to the service provider. When communication problems arise, the demarcation point is used to determine whether the customer or service provider is responsible for troubleshooting and resolving the problem.
- Central office (CO): The CO is the local service provider facility or building where local loops to customers terminate. The physical terminal points are called CO switches. The types of CO switches used depend on technologies used in the local loops.
- Toll network: This network consists of the circuits, switches, routers, and other equipment inside the WAN service provider's network. Today, many of the physical circuits in provider networks are long-haul fiber optic cables. The switches and routers in the toll network are usually high-speed devices with high filtering and forwarding rates to enable data to be moved rapidly to/from customer locations.

WANs also include specialized devices associated with specific types of circuits and services. Most can be mapped to one of the categories identified above.

Appendix 8-2: CSU/DSUs and Other T-Carrier Technologies

A CSU and DSU are often combined in a single device, but they can be separate devices or hardware/software capabilities included in multiplexors or routers. Whether located in single or separate devices, CSU/DSUs lie between the leased line and the customer's network at the demarcation point. They serve as the local interfaces between the DTE at the customer's location (such as a router) and the carrier's digital communications line (e.g., a T1 line).

Essentially, CSU/DSUs function as the digital counterpart to modems that convert digital signals to analog signals and vice versa. However, since both the incoming and outgoing signals are digital, CSU/DSUs do not perform the same types of conversions. Instead, CSU/DSUs encapsulate/package outgoing frames from the customer's network in the frames used for transmission over the leased line (e.g., HDLC or PPP frames). They also perform Physical layer framing and ensure that transmissions at that layer are properly formed and timed. They buffer and rate-adapt digital signals to/from the carrier network and provide a protective barrier against electrical disturbances that could harm CPE at customer locations. The CSU/DSU typically has an autosensing feature that adjusts itself to the line speed of the dedicated circuit.

Digital lines, such as T1 lines, usually terminate at customer locations with four-wire connections that have various connector types, including RJ-45, four-screw terminal blocks, and M-block connectors. The four-wire connection is joined to an appropriate interface on the CSU/DSU. The CSU/DSU also connects directly to the router that serves as the interface between the customer's local network and the WAN.

At the other end of the local loop, at the central office (CO), the carrier has a similar CSU that interfaces with the technologies used to interface with the provider's toll network.

Today, instead of having a separate CSU/DSU device, CSU and DSU capabilities are often included in devices that have a leased line interface on one side and one or more Ethernet interfaces on the other side. Examples include T1 routers and T1 multiplexors.

Basically, a *T1 router* is a router with the necessary interface circuitry to connect directly to a T1 line. T1 routers are often set up as Internet access routers in businesses that use T1 lines for dedicated connections to ISPs. *T1 multiplexers (T1 mux)* are used by businesses that use T1 lines to provide voice communications between locations. A T1 connection bundles together 24 64-Kbps (DS0) time-division multiplexing (TDM) channels. TDM allows multiple users to share the T1 using the time slots. Each 64 Kbps (DS0) channel is ideal for audio/voice data that is digitized using pulse code modulation (PCM). PCM is described in Chapter 2.

Some businesses transmit both voice and data over T1 lines. Voice communications can use a subset of the 24 channels, and the rest can be used for data transfers. A T1 multiplexer manages the 24 DS0 channels when the T1 is used for both voice and data. If the T1 link is only used for data transfer, there is no need for TDM (or a multiplexer); CSU/DSU functions are all that is needed.

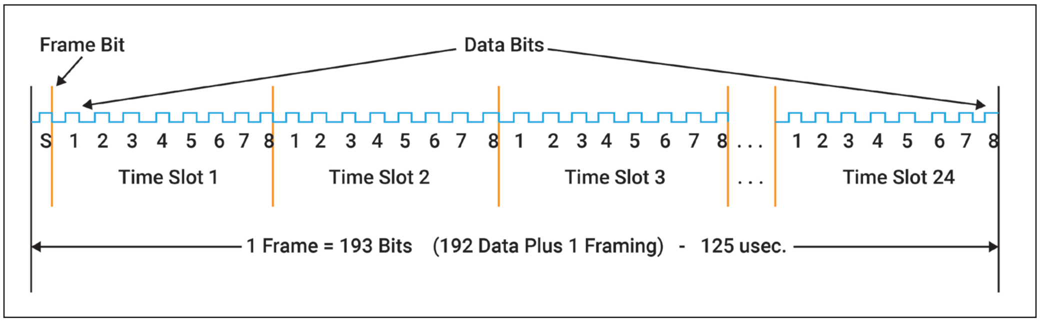

Frame Bit
Data Bits
S 1 2 3 4 5 6 7 8 1 2 3 4 5 6 7 8 1 2 3 4 5 6 7 8 . . . 1 2 3 4 5 6 7 8
Time Slot 1
Time Slot 2
Time Slot 3
. . .
Time Slot 24
1 Frame = 193 Bits (192 Data Plus 1 Framing) - 125 usec.

Figure 8-31: A conceptual depiction of a T1 frame.

Physical layer framing is used on T1 circuits. Although this is considered a type of encapsulation, Physical layer framing is primarily used for sender/receiver synchronization and timing over the physical circuit.

As can be observed in Figure 8-31, the Physical layer frames transmitted on T1 circuits are 193 bits in length. Each frame begins with a single frame bit and is followed by 24 8-bit time slots. As noted above, when a T1 multiplexer is used, the multiplexor uses the time slots for whatever mix of voice and data transfer is needed. If a multiplexer is not used, all the time slots in each frame are used to transfer data.

T1 frame bits contribute to timing and synchronization between senders and receivers. Since each frame is the same length and takes the same amount of time to transmit (125 microseconds), a receiver should expect to receive a frame bit every 125 microseconds when data transfers are occurring.

When a T1 is just used for data transfer, T1 *super frames* may be used. Each super frame includes 24 193-bit T1 frames. The creation of super frames enables the 24 frame bits from the original frames to be used for more than synchronization and timing. They may also be borrowed (robbed) from the 193-bit frames to be used for error detection and communication maintenance.

T1 (DS1) line encoding is bipolar. A plus voltage, a zero voltage, and a minus voltage are important for coding the signal to represent bits. Specifically, the line encoding scheme used on a T1 circuit is *alternate mark inversion (AMI)*. AMI was introduced and briefly described in Chapter 6. Basically, it means that if a 1 bit (mark) is coded as a plus voltage, the next 1 bit must be coded as a minus voltage (see Figure 8-32). Alternating between plus and minus voltages for successive 1 bits helps senders and receivers remain synchronized.

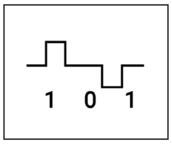


Figure 8-32: An example of AMI (alternate mark inversion) line encoding on a T1 circuit.

T1 extenders (repeaters) exist for signal regeneration. These devices take a weakened incoming signal and regenerate (restore) it to its original form.

The most common Data Link layer protocols used for T-carrier services are HDLC, PPP, and Frame-Relay. *High-Level Data Link Control (HDLC)* was briefly introduced in Chapter 4. It is a relatively simple protocol used for point-to-point connections, and because of its simplicity, it does not require a lot of configuration work to connect two locations. When used on T1 connections, the router at each location would create outgoing HDLC frames for transmission over the connection and would de-encapsulate incoming HDLC frames so that they could be passed to the local network.

Appendix 8-3: Terminal Multiplexers and Other SONET Technologies

Common connection and transmission technologies used for SONET services are illustrated in Figure 8-33. As you can observe in the figure, CPE at a subscriber location interfaces with a terminal multiplexer. Terminal multiplexers may also provide an interface to T-services used by a subscriber.

Terminal multiplexers get their name from being at the endpoints (termination points) of the SONET circuit and performing multiplexing and demultiplexing. Basically, they function as demarcation points for SONET services.

In telecommunications, *multiplexing* (muxing) is a mechanism for combining multiple incoming analog or digital signals into one signal for transmission over a shared medium. There are two major categories of multiplexing: frequency division and time division. Frequency-division multiplexing subdivides the available bandwidth of the communication medium into separate frequency bands and assigns each of the incoming signals to a different frequency band for transmission. If this sounds like OFDM, it should. Time-division multiplexing creates time slots and allocates

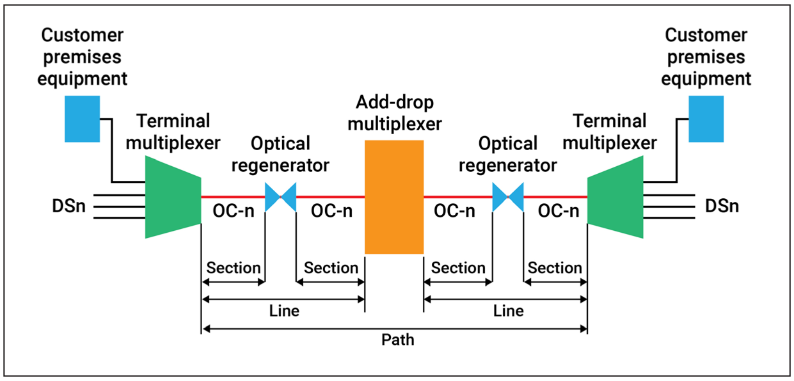


Figure 8-33: SONET services transmission technologies and terminology.

time slots to the different incoming signals. SONET uses time-division multiplexing (TDM).

The optical regenerators depicted in Figure 8-33 function as repeaters and line extenders. *Optical regenerators* take an incoming optical signal and regenerate it (increase its strength to its original level) before retransmitting it to the next device along the circuit.

An *add-drop multiplexer (ADM)* combines, or multiplexes, several lower bandwidth streams of data (e.g., those on several OC-1 lines) into a single beam of light (e.g., that on an OC-3 line). An ADM can also add one or more lower bandwidth signals to the high-bandwidth data stream or extract or drop one or more low-bandwidth signals from the data stream and redirect them to other network paths. Basically, ADMs serve as local "on-ramps" and "off-ramps" to an existing high-speed network, in this case a SONET network.

ADMs are used in the long-haul core networks of carrier toll networks. They are also used in shorter distance "metro" networks. ADMs in long-haul core networks are more expensive to configure and maintain because they also support *wavelength-division multiplexing (WDM)*, which enables multiple carrier signals with different wavelengths (colors) to be carried on a single optical fiber. WDM enables existing fiber cables to scale to higher data transfer capacities without having to install more cabling. WDM is depicted in Figure 8-34.

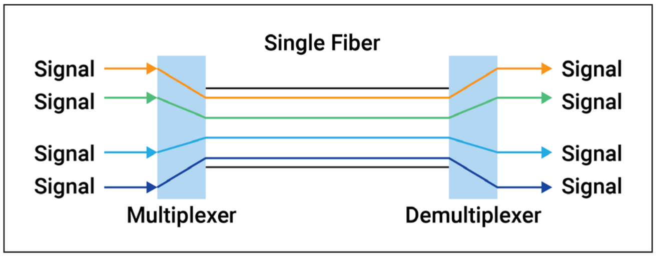


Figure 8-34: A conceptual depiction of wavelength-division multiplexing (WDM).

Figure 8-35 illustrates several functional layers of SONET services. They can be mapped to the Data Link and Physical layers of the TCP/IP and OSI models.

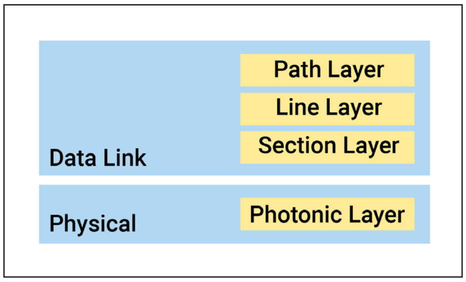


Figure 8-35: SONET services functional layers.

The *Path layer* is responsible for the movement of signals from the optical source to the optical destination. In Figure 8-33, the terminal multiplexers are the optical source and destination, so the path through the SONET network terminate at these endpoints.

The *Line layer* is responsible for the movement of signals across a physical fiber optic line. In Figure 8-33, the OC-n circuits between the terminal multiplexers and the add-drop multiplexer are separate lines.

The *Section layer* is responsible for the movement of signals across a physical section of a line. In Figure 8-33, the parts of the OC-n circuit that connect terminal multiplexers and optical regenerators are examples of line sections.

The *Photonic layer* corresponds to the Physical layer of the OSI and TCP/IP models. This includes the physical specifications for the fiber optic circuits (e.g., presence of light = 1; absence of light = 0).

Appendix 8-4: Communications Satellites

A **communications satellite** is a satellite that creates a communication channel between a transmitter and a receiver at different locations on Earth. To avoid interference, an *uplink* frequency is used to transmit signals to the satellite, and a *downlink* frequency is used to transmit from the satellite to a receiving antenna on Earth. The satellite device that converts the uplink signal to the downlink signal is a *transponder*.

More than 2200 communications satellites are in use, not counting military satellites. Most communications satellites are in *geostationary orbit* 22,300 miles (35,900 km) above the equator. This enables ground station antennas to be permanently aimed at the same point in the sky because there is no need to track the satellite's orbit.

Some communications satellites are not geostationary and are typically classified by their orbital altitudes. *Medium Earth orbit (MEO)* satellites have orbital altitudes from 1243 to 22,400 miles above Earth. The orbital region below MEO is referred to as *low Earth orbit (LEO)* and is located from 99 to 1243 miles above Earth. Ground station tracking and/or handoffs among satellites and ground stations are common for MEO and LEO satellite communications.

Because of the distances that signals must travel, latencies are higher for geostationary satellite communications than for MEO or LEO communications. Noticeable delays may be observed when geostationary satellites are used. Like terrestrial microwave transmissions, satellite transmissions can also be affected by environmental conditions, such as heavy downpours. They are sometimes also affected by sunspot activity and electromagnetic pulses that originate outside the solar system.

A group of satellites that work in concert to provide voice or data communications is called a *satellite constellation*. Iridium and Globalstar are examples of LEO satellite constellations originally designed to provide satellite phone services to remote areas. Iridium's constellation includes 66 satellites. The company SES has a constellation of MEO satellites called O3b that provide broadband Internet services to remote locations, maritime locations, ships at sea, and planes in flight.

Some LEO satellite communications providers use their satellites to provide discontinuous store-and-forward data transfers. Their LEO satellites receive and store data as they pass over one part of the Earth and transmit it later when they pass over another part of the Earth's surface. Orbcomm and CASCADE are examples of satellite services that provide this type of data transfer.

Appendix 8-5: A Closer Look at MPLS

There's been some controversy about whether MPLS is a Network layer (Layer 3) or Data Link layer (Layer 2) service because it does not neatly map to either in the seven-layer OSI model or the five-layer TCP/IP model. As a result, it is sometimes classified as a Layer 2.5 protocol. One of MPLS's key benefits is that it separates forwarding mechanisms from the Data Link layer and enables forwarding tables to be created for any Data Link layer protocol.

When an end user sends traffic into the MPLS network, an MPLS label is added by an ingress MPLS router that sits on the network edge. The MPLS label consists of four subparts, which are illustrated in Figure 8-36:

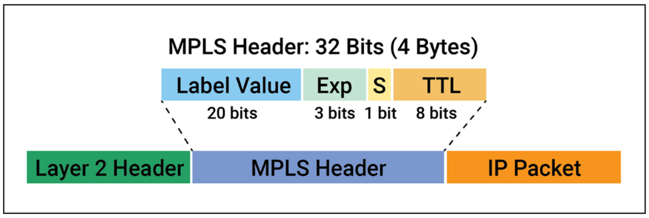


Figure 8-36: The format of an MPLS label and its placement in a Data Link layer frame.

- The Label Value field holds the information used by MPLS routers to identify the route on which the packet should be forwarded.

- The bits in the Experimental (EXP) field specify the quality of service (QoS) priority that the labeled packet should have.
- The Stack (S) field informs routers if the packet is on the last leg of its journey through the network. It informs the router if it is the packet's egress router.
- The Time to Live (TTL) field identifies how many hops the packet can take before it is discarded.

As Figure 8-36 illustrates, if Ethernet is the Data Link (Layer 2) protocol, the MPLS label is inserted between the Ethernet header and the IP PDU in the Ethernet frame's Data field. This MPLS header specifies how the frame should be routed through the MPLS network from the LAN at one business location to the LAN at another location.

Traditional selling points for MPLS networks by WAN services providers include scalability, application performance, superior bandwidth utilization, reduced likelihood of network congestion and detrimental effects on applications, and better user experience with applications supported on the network.

While MPLS does not provide encryption, it enables a business to create a virtual private network through a MPLS network that is separated from the public Internet. As a result, a MPLS network is sometimes considered a secure mode of transferring data between locations. MPLS's use of labels makes MPLS networks less vulnerable to denial-of-service attacks than the Internet and other IP networks.

The biggest downside to MPLS is its expense. MPLS connections are much more expensive than standard Internet connections. Also, since T-carrier and SONET circuits are typically used to interconnect the routers in MPLS clouds, MPLS data rates are comparable to those for T-carrier and SONET services.

MPLS was designed for organizations with multiple geographically distributed locations and private data centers that centrally manage and store applications and enterprise data. As evidenced in Chapters 2 and 3, the increasing importance and use of cloud services in enterprise networks has resulted in many businesses moving away from the types of network architectures that MPLS networks were established to serve.

Today, most businesses have redirected much of the traffic that once went to private data centers to cloud services providers. For most businesses, it is more efficient to send data traffic directly to the cloud than to private data centers. Increased use of video applications, mobile apps, and other bandwidth-hungry applications have spawned growing business network data volumes and bandwidth requirements. Because of their wiring, MPLS services are difficult to scale on demand to meet needs; this is a shortcoming that cloud services do not have.

Software-defined WANs (SD-WANs) are architected with cloud connectivity in mind, if not assumed. This has motivated many businesses to replace or supplement their MPLS WANs with SD-WANs. SD-WANs apply software-defined networking (SDN) concepts to WANs, and SD-WAN edge devices apply rules and policies to send an application's traffic along the best path through the WAN. These devices perform functions comparable to MPLS ingress routers.

Because an SD-WAN can route any type of traffic, including MPLS traffic, MPLS's role is changing in many businesses. Many small and midsize businesses are replacing MPLS with an all-cloud IT model. Larger enterprises with sunk costs in MPLS networks show signs of moving to hybrid WANs where MPLS is used for legacy applications and SD-WAN is used for Internet and cloud traffic.

From 2019 to 2020, business use of MPLS declined by almost 25%, while the use of SD-WANs in enterprise networks increased from less than 20% to more than 40%. Cost differences between MPLS and SD-WANs are likely to fuel further migration of businesses from MPLS WANs to SD-WANs.

The architects of enterprise WANs have many factors to consider when determining the mix of WAN PSN services that they need. When considering MPLS, network architects are making risk/reward calculations between very reliable but expensive MPLS services and less reliable but less-expensive services that leverage the public Internet. Improvements in other networking technologies and PSN services, including Ethernet services, are also important considerations for the architects of enterprise WANs.

Appendix 8-6: VPN Services: A Closer Look

The VPN routers used to connect business locations (such as those depicted in Figure 8-22) usually cost between $6000 and $8000 apiece. When circuit costs to ISPs, Internet access services costs, and user support costs are factored in, a business with 1000 employees may experience annual VPN expenses of approximately $250,000 (about $250 per employee per year).

VPN routers enable virtual circuits called *tunnels* to be created through the Internet. As noted in previous chapters, especially Chapter 5, a VPN tunnel is basically an encrypted frame or packet that is transferred securely across the virtual circuit that is established between the sending and receiving networks. The VPN router at the sending network encrypts and/or encapsulates the packets or frames to be transferred, and the VPN router at the receiving network de-encrypts and/or de-encapsulates the frames or packets that are transferred.

Encryption and encapsulation protect the frames or packets while in transit. Depending on the types of tunnels and encryption used, ISPs are typically unable to see the contents of the packets or frames being sent through their networks to the Internet. This helps the data being transferred to remain private. Sometimes, however, it is possible for ISPs to capture packet source and destination addresses.

Employees working at home or in public settings typically use VPN software to establish a connection with a VPN router at a business location. As noted in Chapter 7, VPN software is recommended when employees use Wi-Fi networks in public settings to communicate with other employees.

VPNs can be implemented at either the Network layer (Layer 3) or the Data Link layer (Layer 2) of the TCP/IP model. IPsec, described in Chapter 5 is considered a Layer 3 VPN protocol because it encrypts the Transport layer PDU carried in an IP packet's Data (Payload) field.

As Chapter 5 explains, IPsec has two different modes: Transport and Tunnel. In the Transport mode, the Data field is encrypted, but the packet's Destination Address and Source Address fields are not, to enable them to be used for routing the packet through the network. In the Tunnel mode, the packet's Data, Destination Address, and Source Address fields are encrypted and placed in the Data field of a new IP packet. The Source and Destination Address fields for the new packet are the IP addresses of the VPN routers at the sending and receiving networks. Because the Tunnel mode encrypts more fields in the original IP packet than the Transport mode, it is considered more secure.

IPsec is used by default in IPv6 networks and is widely used to protect data in transit in IPv4 networks. IPsec is widely supported in VPNs. You are encouraged to revisit Chapter 5 for a more detailed description of IPsec and the differences between its Transport and Tunnel modes.

Generic Routing Encapsulation (GRE) is another example of a protocol used for Layer 3 VPNs. GRE was developed by Cisco Systems. It can be used to encapsulate the payloads of IP packets as well as many other types of Network layer packets. It is frequently used to transfer router table updates across secure links.

Layer 2 VPNs encapsulate and encrypt Layer 2 (Data Link layer) frames, such as Ethernet or Wi-Fi frames. Point-to-Point Tunneling Protocol (PPTP) and Layer 2 Tunneling Protocol (L2TP) are examples of protocols used for Layer 2 VPNs.

PPTP is based on Point-to-Point Protocol (PPP), which is described in section 8.2.2. Historically, PPTP was the first VPN protocol that Windows supported. Although it is compatible with a wide range of device operating systems, PPTP performs best on Windows devices.

PPTP uses 128-bit encryption and has a less noticeable effect on Internet access speeds than L2TP. Today, however, 256-bit encryption is recommended for VPNs. In addition, PPTP users may encounter connection issues on unstable networks, whereas unstable networks are less problematic for L2TP.

One of the main differences between PPTP and L2TP is that PPTP does not require IPsec. As a result, PPTP doesn't require Public Key Infrastructure (PKI) or certificates to be checked when establishing a VPN.

L2TP was developed jointly by Cisco and Microsoft to be a more secure VPN protocol than PPTP. L2TP uses 256-bit encryption, which requires more CPU resources than PPTP and makes it slower than PPTP.

Although L2TP doesn't have any built-in encryption or authentication capabilities (both of which are essential VPN characteristics), it typically leverages IPsec to provide end-to-end encryption, data integrity, and data origin authentication. IPsec also uses PKI (e.g., IKEv2) to provide additional security.

Dynamic multiple VPN (DMVPN) allows secure temporary data exchanges between spoke locations without the involvement of the VPN router, gateway, or server at the hub. The DMVPN protocol creates a mesh VPN that can be selectively applied between locations as needed. This is accomplished using VPN firewall concentrators and routers with DMVPN software that allows the mesh to be applied between two locations on a temporary basis.

A DMVPN often has a single (hub) GRE tunnel interface capable of simultaneously securing several IPsec tunnels. It also involves routing protocols that enable the DMVPN software to identify routes between different endpoints (spoke locations). Cisco's Next Hop Resolution Protocol (NHRP) and other Address Resolution Protocol (ARP)–like protocols are also used to establish temporary on-demand VPN connections between locations.

In sum, relatively inexpensive VPNs are used in some enterprise networks to interconnect distributed business locations. Layer 3 and Layer 2 tunneling protocols enable the creation of a secure private network that uses packet-switched data transfers over permanent or temporary virtual circuits.

Because many businesses use dedicated circuits to connect to ISPs, VPNs usually are implemented over copper or fiber optic cables. However, wireless circuits are also used in carrier and enterprise networks and warrant inclusion in our discussion of WANs.

Appendix 8-7: A Further Look at RANs

In a traditional RAN architecture, a base station includes a *remote radio head (RRH)* that receives radio signals from user equipment (UE). The RRH (or radio unit [RU]) also transmits digital radio signals to UE that originate from other UE in the base station's cell, or from another cell or the CN. It also converts radio signals to/from the form needed by the base station's *baseband unit (BBU)*. When the RRH receives signals from UE, it communicates with the BBU using the *Common Public Radio Interface (CPRI)*. The BBU receives information from the RRH and performs the processing needed to forward it to the core network. Data returns to UE via the reverse process.

Virtual slicing is sometimes used to create private RANs from MNO spectrum. This involves the creation of closed access groups (CAGs) and the use of CAG IDs to disallow network access to nonsubscriber UE.

Three types of private 5G RANs are common—independent private networks, RAN sharing networks, and networks created from slices of MNO networks:

- An independent private network is a mobile network deployed at the business location that is separate and independent of an MNO's public network. These networks do not rely on or interact with the MNO's public 5G network. The enterprise locally stores and maintains user and subscription databases. Network control and data services are also handled locally. This type of network may use dedicated (licensed) spectrum or unlicensed spectrum.
- In a RAN sharing network, RAN CN services are handled locally, but the RAN and its spectrum are shared with the MNO's public network. User and network control are handled by the MNO.
- In a network slice network, the MNO assigns a virtual slice of the public network's spectrum to the enterprise. Other differences among these different types of private 5G networks are summarized in Table 8-6.

Today, base stations in RANs may be equipped with multiple-input and multiple-output (MIMO) antennas to enable simultaneous connections with multiple UE devices. OFDM (orthogonal frequency-division multiplexing) is also supported in 5G RANs. It can be combined with dense QAM constellations to provide high data rates. Figure 8-37 depicts a RAN with two base stations.

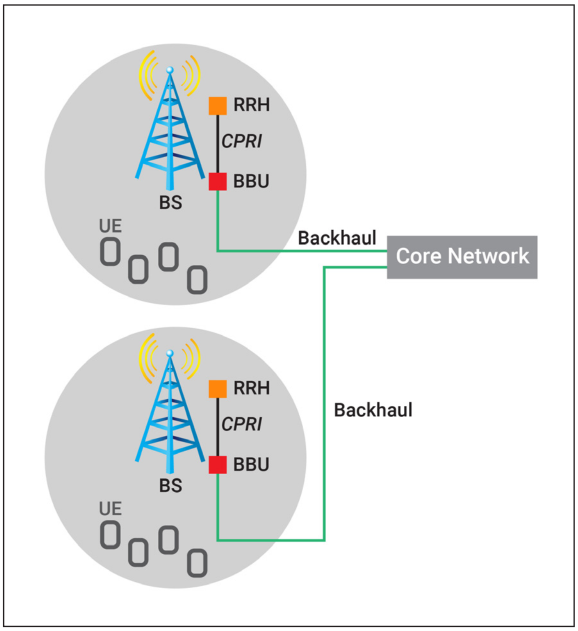


Figure 8-37: Key components of RAN architectures.

4G (LTE) mobile wireless networks use OFDM for the downlink (base station to UE device). The channel is divided into subcarriers with fixed 15 kHz spacing. The modulation used on the subcarriers can be quadrature phase-shift keying (QPSK), 16QAM, or 64QAM.

Table 8-6: Advantages and disadvantages of the three types of private 5G networks.

Private 5G Network Type	Advantages	Disadvantages
Independent	• isolation from public MNOs • local QoS assurance • local data storage and security • predictable latencies by having all RAN components close by • no monthly subscription charges	• high capital expenditure costs (software, hardware, and licensing) • high spectrum cost for licensed spectrum • potential for signal interference if unlicensed spectrum is used • need for appropriately skilled IT staff
RAN Sharing	• uses licensed MNO spectrum • lower capital expenditure costs than independent networks • network is maintained by MNO • local data storage and security • predictable latencies from having all components close by	• not completely isolated from MNO • signaling dependence on MNO network • user/subscriber information is stored in MNO databases • monthly subscription charges for RAN usage • need for appropriately skilled IT staff for troubleshooting local problems
Network Slice	• logical separation from MNO's public network • uses spectrum licensed by MNO • lower capital expenditure costs than independent networks • network is maintained by MNO	• not physically separated from the MNO's public network • enterprise is dependent on MNO network for QoS • higher latencies than independent or RAN sharing networks • user/subscriber information is stored in MNO databases • monthly subscription charges for RAN usage • need for appropriately skilled IT staff for troubleshooting local problems

Source: Aggregated from multiple sources included in Chapter References.

The *5G New Radio (NR) standard* uses OFDM on both base station uplinks and downlinks. The NR standard provides more carrier spacing flexibility than LTE (carriers can have 15 kHz, 30 kHz, 60 kHz, 120 kHz, 240 kHz, or 480 kHz spacing), with up to 3300 subcarriers. Subcarrier modulation can be QPSK, 16QAM, 64QAM, or 256QAM.

So, like Wi-Fi, advances for MIMO, ODFM, and QAM are contributing to increased capabilities and improved performance of 5G RANs. And the maturation of 5G RANs is enabling businesses to consider future scenarios involving liberal use of wireless connections.

Appendix 8-8: Business Rationale for SD-WANs

Businesses are attracted to SD-WANs because of their availability to optimize application performance, especially for ESs and other mission-critical applications. Better application performance can contribute to improved productivity, increased customer satisfaction, and the achievement of other business goals.

SD-WANs support VPNs. This contributes to network security and can enhance application performance across public Internet infrastructure. The proliferation of cloud services in enterprise networks means that connections between applications and users often go through the Internet.

SD-WANs have simple interfaces that make the network easy to configure and manage. This has made them an important part of network management. SD-WANs typically support third-party services such as firewalls and WAN optimization controllers.

SD-WAN products can be physical devices or software-based capabilities. Most are preconfigured appliances placed at network edges such as branch offices and data centers. Some SD-WAN products are "virtual appliances" that can work with existing network hardware or be deployed in cloud environments, such as Amazon Web Services (AWS), or as SaaS to facilitate their inclusion in the WAN.

SD-WAN products supplement or replace traditional WAN routers at business locations and provide the capabilities needed to implement and control the QoS and security policies needed to support the organization's various application classes. SD-WAN products provide a network overlay that causes inexpensive consumer-grade Internet links (e.g., cable modems, FTTH, and DSL) to behave like dedicated circuits.

So there are numerous reasons why SD-WANs are attractive to businesses. Here are some of those reasons:

- Resilience. An SD-WAN can improve network resilience and reduce network downtime. An SD-WAN contributes to resilience through real-time detection of link deterioration or outages and automatically rerouting of traffic to working links.
- Self-healing capabilities. By incorporating AI, an SD-WAN can perform continuous troubleshooting and initiate fixes to WAN problems. This contributes to resilience and downtime reduction.
- Quality of service. QoS is supported by SD-WANs' application awareness, which gives priority to the organization's most critical applications. Dynamic path selection contributes to QoS by sending an application's data traffic over the fastest link or by splitting that traffic between two paths when doing so would improve the application's performance.
- Application optimization. SD-WANs often use caching to store recently accessed information. This improves application delivery by minimizing the time needed to satisfy future requests for the same data/information.
- Scalability. SD-WANs can easily scale up or down to meet business needs. The use of regular Internet connections makes it easier to expand bandwidth. This also makes it simpler to add business locations to, or remove them from, the WAN.

- Easier WAN administration. SD-WANs typically have easy to use network management interfaces. Automatic path selection also facilitates network administration. SD-WAN appliances at network edges can be centrally configured, and configuration updates can be pushed to the edge appliances.
- Online traffic engineering. An SD-WAN provides network managers with a global view of network status. This enables adaptive traffic engineering in response to current link usage and traffic volumes. SD-WAN controller calculations of transmission rates may trigger rate limiting for lower priority applications to ensure that critical applications have sufficient bandwidth.
- Security. IPsec is usually used to secure SD-WAN communication.
- Secure access service edge (SASE). SD-WAN is an important part of SASE solutions that strive to securely connect application users at geographically distributed locations with distributed applications in data centers and the cloud. SASE combines SD-WANs, firewalls, and other network security technologies, such as cloud access security broker (CASB), data loss prevention (DLP), and secure web gateway (SWG). Collectively, these technologies securely connect and protect users and applications. Ultimately, the goal of SASE is to deliver high levels of quality of experience for users of cloud-hosted applications without compromising security.

CHAPTER 9

Appendix 9-1: Vehicle-to-Vehicle Communications

The major components of V2X communications delineate different types of V2V communications:

- V2V communications include applications for maintaining safe distances between vehicles, speed, turning corners, and changing lanes.
- V2I includes vehicle interactions with road infrastructure, such as traffic lights, road signs, and toll gantries.
- V2P includes applications such as sensing a nearby person or cyclist.
- V2N includes vehicle communication with the Internet or another network; this may also include communication infotainment and connected vehicle applications.
- V2D consists of the exchange of information between a vehicle and any electronic device that may be connected to the vehicle itself, such as with a Bluetooth-connected smartphone for hands-free phone calls or running apps.
- V2B involves wireless interactions between vehicles and roadside barriers and is expected to be an important component in next generation intelligent transportation systems (ITSs).

V2V Architecture, Technologies, and Protocols

V2X focuses on enabling a vehicle to have low-latency and high-reliability communication with other vehicles (V2V), pedestrians (V2P), roadside infrastructure (V2I), and networks (V2N) in ways that facilitate road safety and traffic efficiency. Today, vehicles with autonomous driving capabilities are equipped with advanced sensors, cameras, light detection and ranging (LiDAR), radar, global navigation satellite system (GNSS) equipment, and Controller Area Network (CAN) technologies.

Controller Area Network (CAN) is a serial communications protocol developed by the automotive industry that allows multiple electronic systems in a vehicle to share control data. It is a message-based protocol that was originally designed for multiplexing electrical signals from multiple devices over a single cable to save on copper.

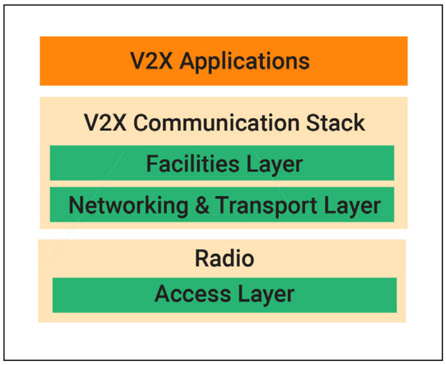


Figure 9-21: V2X communication architecture.

The data for each device is transmitted serially as frames, and through multiplexing, frames from more than one device can be transmitted at the same time over the same cable. Priorities can be assigned to the different devices sharing a cable, and the multiplexing is typically configured to enable the highest priority device to continue while the others back off.

Transmitted frames are received by all devices, including by the transmitting device. You may recall that this also happens when CSMA/CD is used in serial bus or shared media hub architectures. CAN is sometimes called *CAN bus* and is considered a robust bus standard for vehicles that enables microcontrollers and devices to communicate with each other's applications without having to rely on a central host.

V2X architectures are usually depicted as having four layers—access, networking and transport, facilities, and applications—as shown in Figure 9-21. In some depictions of V2V and V2X communications, the access layer is called the *device layer* and the facilities layer is called the *services layer*. Nonetheless, the resemblance between V2X, M2M, and IoT architectures is easy to observe.

V2V communication takes place at the access layer of the four-layer V2X architecture. This is illustrated in Figure 9-22. This figure also illustrates that V2I involves communicating wirelessly with roadside units (RSUs), while V2N and V2P communication involves cloud services.

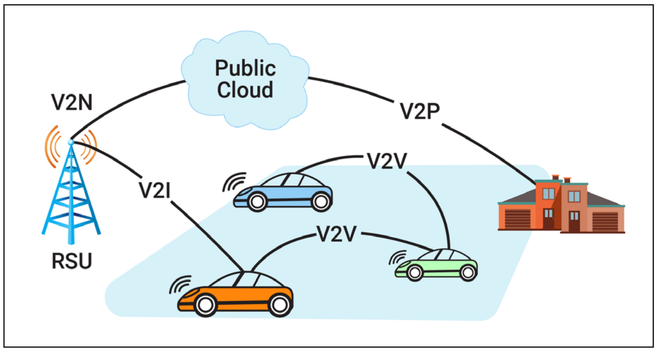


Figure 9-22: V2X communication VANETs include vehicle clusters and RSUs.

As noted previously, V2V communication generally focuses on crash avoidance. It involves *vehicular ad hoc networks (VANETs)*, which are wireless ad hoc networks that enable nearby vehicles to communicate with one another and share data/information about their locations, speeds, and directions.

Like other wireless transmitters and receivers, vehicles and RSUs have limited communication ranges that restrict their communication to vehicles that are within range. However, appropriately equipped vehicles can communicate with vehicles both inside and outside their communication range through relay vehicles. This is illustrated in Figure 9-23.

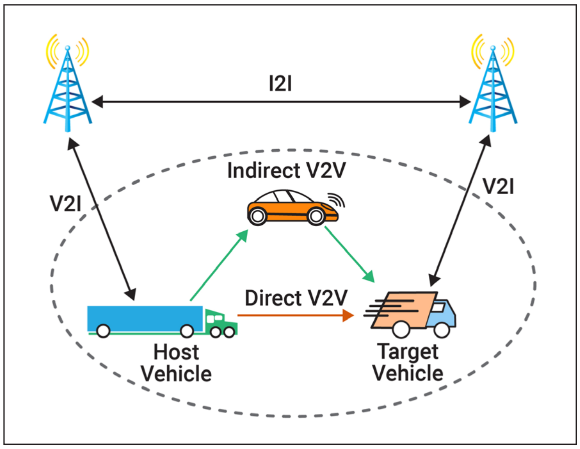


Figure 9-23: Example of indirect V2V communication via a relay vehicle.

The automotive industry has wrestled between using a variant of Wi-Fi called *dedicated short-range communications (DSRC)* and using cellular communications as the basis for V2V. A depiction of DRSC is provided in Figure 9-24.

DSRC is a U.S. Department of Transportation (DOT) initiative based on the ISO's *communications access for land mobiles (CALM) architecture* for vehicle-based communication networks that focuses on vehicle safety services and applications such as toll collection and commerce transactions via cars. The vision of DSRC is a nationwide network that enables communications between vehicles and RSUs and other vehicles.

IEEE 802.11p is the Wi-Fi variant used in DSRC. It amends the IEEE 802.11 standard to add *wireless access in vehicular environments (WAVE)*, which is basically a vehicular communication system that supports ITS applications. The 802.11p standard focuses on data exchange between moving vehicles and between moving vehicles and the roadside infrastructure (RSUs) in the licensed 5.9 GHz frequency band (5.85–5.925 GHz) that has been allocated for ITS systems.

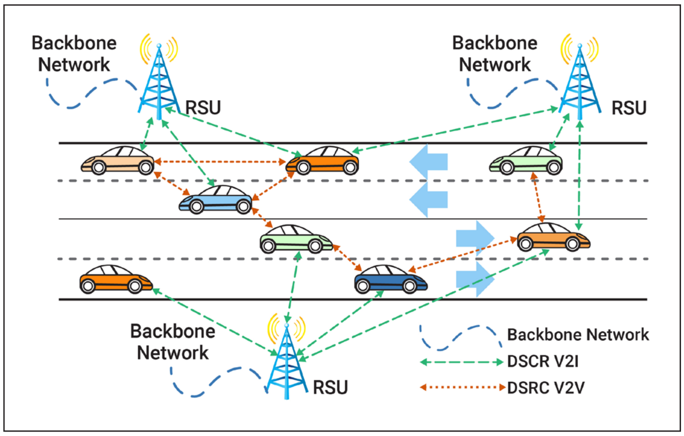


Figure 9-24: A high-level overview of DSRC V2V and V2I communication.

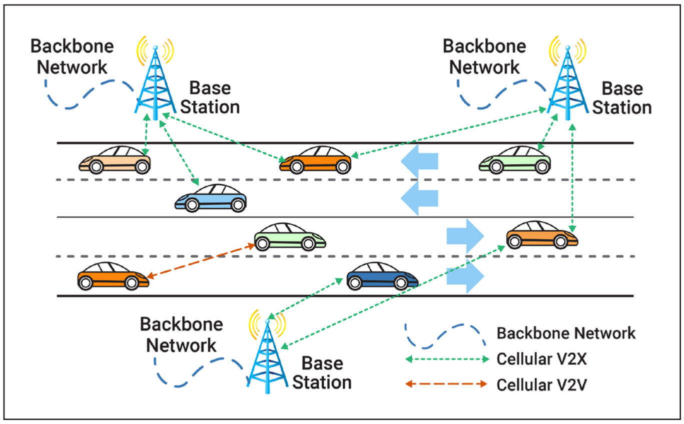


Figure 9-25: A high-level depiction of cellular V2X (C-V2X) communication.

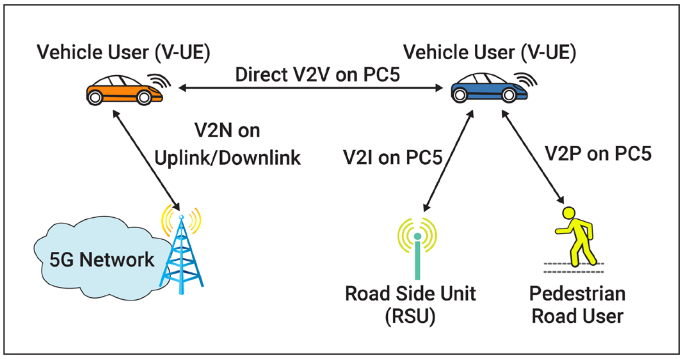


Figure 9-26: A high-level depiction of C-V2X communication modes.

The alternative to DSRC is *Cellular Vehicle-to-Everything (C-V2X)* communications. Figure 9-25 provides a high-level depiction of C-V2X.

C-V2X uses 3rd Generation Partnership Project (3GPP) 4G LTE or 5G NR (New Radio) connectivity to transmit and receive signals. Two transmission modes are used (see Figure 9-26). The first involves direct communications to and from vehicles, road infrastructure, and pedestrians. C-V2X works independently of the cellular networks in this mode and uses a PC5 interface. In PC5, user equipment (UE) communicates with another UE over a direct channel, and communication with a base station is not required.

The second mode involves communications through a cellular network. In this mode, C-V2X employs a conventional mobile network to inform vehicles about road and traffic conditions in their surrounding areas. In 5G networks, UE communicates with a 5G RAN using a Uu interface. A Uu interface (also known as an NG-Uu interface) is a crucial component in 5G network architecture that serves as the interface between the core network and UE devices. A Uu interface enables 5G UE to communicate with the core network via a 5G RAN base station. This is illustrated in Figure 9-27.

Competing DSRC and C-V2X implementation options in VANETs translates into the need to include both 802.11p and C-V2X protocols at the access layer of V2X communication architectures. This is shown in Figure 9-28.

Although it may be obvious, if the United States (and anywhere else) wants to have an easier path toward a future with autonomous self-driving vehicles, it may be important to decide on a single access-layer protocol. While it may be feasible to have VANETs that support both DSRC and C-V2X, such heterogeneous deployments may not be feasible in the long term.

Many pundits expect C-V2X to prevail. As deployment of 5G cellular increases, opportunities are being created for more widespread use of V2V communications. Relative to DSRC, C-V2X can detect potential safety hazards and unsafe road situations over longer distances. It

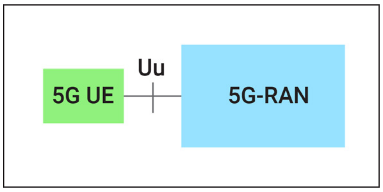


Figure 9-27: A Uu interface is used in 5G networks to enable UE to communicate with RAN base stations.

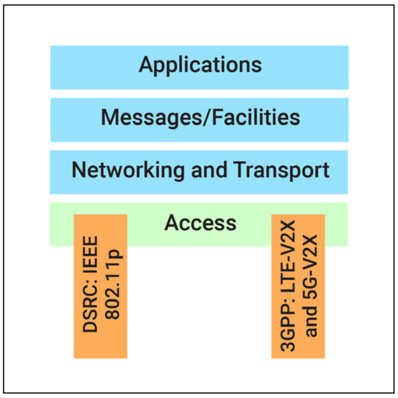


Figure 9-28: Access layer protocols in C-V2X architectures.

can enable a fully equipped autonomous driving vehicle to overcome non-line-of-sight (NLOS) hazards.

For example, a vehicle could use PC5 interfaces for communications with other vehicles, RSUs, and pedestrians; it could also use Uu interfaces with a cellular network for safety hazard information. C-V2X can aggregate information from these various sources to update vehicles with very detailed road and traffic information, which can, in turn, be used to create HD (high-definition) information about local conditions. This information can be combined with blind-spot detection and other applications to increase on-the-road vehicle safety.

A lot is happening very rapidly in V2V communications, and what is being presented here just scratches the surface of where things currently stand and where they are going. Hopefully, this section provides a foundation for further exploration of our migration toward autonomous self-driving vehicles and ITS infrastructure.

Appendix 9-2: A Closer Look at IoT Platforms

IoT Platforms

IoT systems and applications are increasingly used in enterprise networks. Businesses are connecting a wide variety and increased number of IoT endpoints to their networks and are using data from them to manage important physical assets (e.g., manufacturing equipment, vehicles, warehouses, stores, office facilities, etc.). Common IoT-enabled business benefits include asset optimization and new business opportunities and revenue models, such as pay-per-use subscriber services.

IoT platforms are sometimes described as a type of middleware that sits between IoT devices or gateways and IoT applications. They map to the processing layer of the five-layer IoT architecture models and to layers 3, 4, and 5 of the IoT World Forum Reference Model (Figure 9-11). Figure 9-29 provides examples of the different functions that IoT middleware performs.

IoT platforms also provide the capabilities needed to integrate edge computing technologies, cloud storage infrastructure, and the applications deployed to oversee IoT systems. As Figure 9-29 illustrates, IoT platforms (or middleware) support APIs for both IoT gateways and applications and provide the functionality needed to support technologies and protocols used for connectivity and data transport. Because an IoT platform facilitates the deployment of applications that manage and control IoT devices, it is sometimes called an *application enablement platform (AEP)*.

From the perspective of middleware, an IoT platform serves as a mediator or interface that

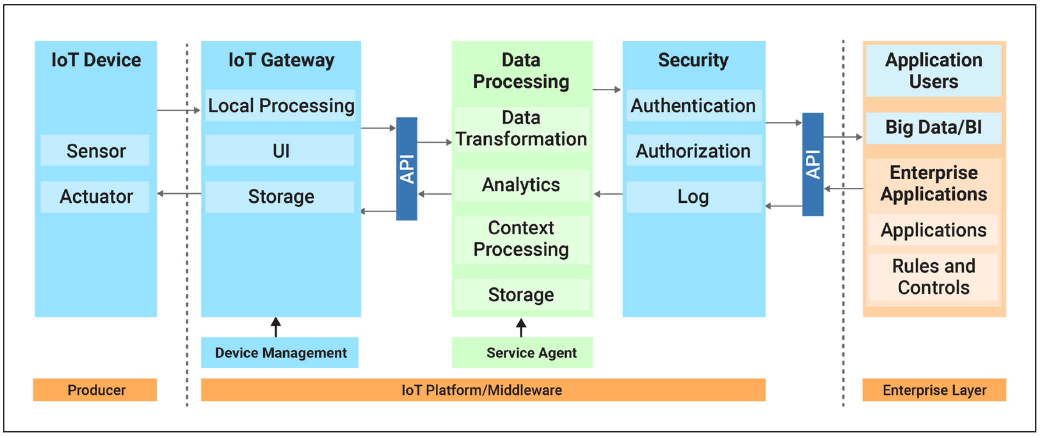


Figure 9-29: IoT software platforms provide middleware needed to integrate IoT devices and applications.

enables interaction between "things" and the Internet. It is the glue that binds the things to the Internet in an IoT network.

An IoT platform hides the heterogeneity of the components, devices, and technologies in an IoT system. It provides solutions to interoperability, security, reliability, and other challenges that IoT systems and applications must overcome to provide value to businesses and other organizations. Important characteristics of IoT platform software or middleware include:

- Flexibility. The ability to establish better connectivity and communication between applications and things.
- Transparency. The ability to hide the complexities of IoT architecture details from both applications and IoT devices; this enables devices to communicate with applications and vice versa with minimal knowledge on either side.
- Interoperability. This enables applications on interconnected networks to exchange data and services even when they have different protocols, configurations, and data models.
- Platform portability. An IoT platform should support any time, any way communication with any device in a manner that provides independence from network protocols, operating systems (OSs), and programming languages.
- Security. IoT platforms should provide multiple security mechanisms, including, but not limited to, authentication, authorization, and access control.

Reusability of application development components is another desirable feature of IoT platforms. It contributes cost efficiency in modifying IoT application and system components to adapt to changing business requirements or preferences. Fault tolerance is another desirable feature of IoT platforms.

Some IoT business platforms are designed to serve vertical markets/applications, such as building management, or specific smart industry types (e.g., smart city, smart grid, smart logistics, smart manufacturing, or smart transportation). Others are application specific or platform-centric to enable flexible deployment of IoT systems and applications across industries.

In sum, enterprise IoT platforms are software components that provide interfaces between IoT devices and applications. They facilitate the deployment of IoT applications and systems that monitor, control, and manage IoT devices. IoT platforms contribute to remote data collection from IoT sensors and to secure connectivity among IoT devices.

Appendix 9-3: A Further Examination of M2M Communication

M2M Technologies

Table 9-7 illustrates how various types of wireless networking technologies are used to support device-to-device and device-to-cloud communications in IoT networks and applications.

M2M area networks may use proprietary non-IP-based communication protocols, but some rely on TCP/IP protocols. Some of the most common protocols used for M2M communication include MQTT, CoAP, and LWM2M. These are specifically targeted for low-power devices that strive to conserve power so that they can operate for a long time.

MQTT and CoAP are introduced and briefly described earlier in section 9.3.2. Lightweight M2M (LWM2M) is a communication protocol developed by the Open Mobile Alliance (OMA) that is designed for remote device management and telemetry in M2M applications. LWM2M is designed to reduce power and data consumption in low-power M2M devices with limited storage and processing capabilities; such devices are known as *resource-constrained* devices.

Table 9-7: Examples of wireless technologies used to support M2M systems and applications.

Wireless Technology	Network Throughput	Network Location	Example Applications
LPWAN	Low	Indoor; outdoor	IIoT, smart buildings, smart cities
Wi-Fi 6	High	Usually indoor	Enterprise networks, residential networks, public Wi-Fi
Bluetooth	Mid	Usually indoor	Audio transfer, wearables, smart home
5G	High	Outdoor; indoor	Consumer mobile communications, connected vehicles, telemedicine, mission-critical IoT

LWM2M protocol specifications address common IoT device management functions, such as firmware and software updates, connectivity monitoring, and remote device actions. They also address cellular management and provisioning.

LWM2M supports four logical interfaces, which contributes to standardizing how device management and telemetry are implemented:

- Bootstrapping interface—This interface enables rapid configuration of services provided by devices.
- Client registration interface—This informs the M2M server about the device's existence in the M2M area network and its supported functionality; this is also used for wireless firmware and software updates.
- Device management and service enablement interface—This enables changes to device settings and parameters.
- Information reporting interface—This enables application users to obtain error reports from devices when their services are not working properly; it also enables users to send device status queries.

Because LWM2M is a compact protocol, it works well in potentially unstable and low-bandwidth networks, such as sensor or cellular networks. It also works efficiently on IoT gateways and routers. It has gained traction in a variety of application areas, including the automotive industry, logistics, manufacturing, robotics, security, telemedicine, and utilities.

Like other communication protocols, the LWM2M protocol specifications are evolving. LWM2M version 1.1.1 extended the protocol's capabilities to support both UDP and TCP at the Transport layer, and to overcome common UDP issues with firewalls and Network Address Translation (NAT). LWM2M 1.1.1 provides non-IP data delivery over cellular networks using Narrowband IoT (NB-IoT) or LTE Cat-M1 (Cat-M1 or LTE-M). In addition, the new version added a security layer called Object Security for Constrained RESTful Environments (OSCORE) to improve security for M2M applications supported by UDP, TCP, or SMS.

This combination of features and capabilities has enabled LWM2M to join MQTT and CoAP as a widely used M2M protocol. Numerous pundits view it as the future of M2M communication.

M2M Platforms

Like an IoT platform, an **M2M platform** is a software solution that sits between the device and application domains of M2M architectures and is designed to simplify and unify the management of M2M devices and applications.

An M2M platform usually has a wide range of features and functionalities, such as those illustrated in Figure 9-30. Typically, an M2M platform simplifies the management of the data transmitted by M2M devices to the back-end systems and applications that process M2M data. It also contributes to managing device software updates and device life cycle administration. Like an IoT platform, an M2M platform leverages software APIs for both devices and applications to ensure that data is appropriately exchanged.

Appendix 9-4: Additional IoT and IIoT Security Considerations

Collectively, Figures 9-31 and 9-32 illustrate the types and extent of security that are important for users of IoT applications, edge technologies, and IoT devices. This is consistent with zero-trust security initiatives and underscores why zero-trust mechanisms are widely considered to be best practices for IoT security. These figures also illustrate how security needs to focus on *all* layers of IoT architectures to sufficiently secure IoT systems and applications.

Figure 9-33 summarizes a wide range of IoT security risks and threats associated with IIoT systems and applications. This figure also identifies some of the major types of security technologies deployed in industrial settings, including advanced firewalls and intrusion detection systems, ML, and vulnerability assessment tools, such as network penetration testing.

The OT at the bottom of Figure 9-33 stands for operational technology. *Operational technology* is software and hardware that monitors and/or controls industrial equipment, assets, processes, and events. OT detects or causes a change in equipment/assets

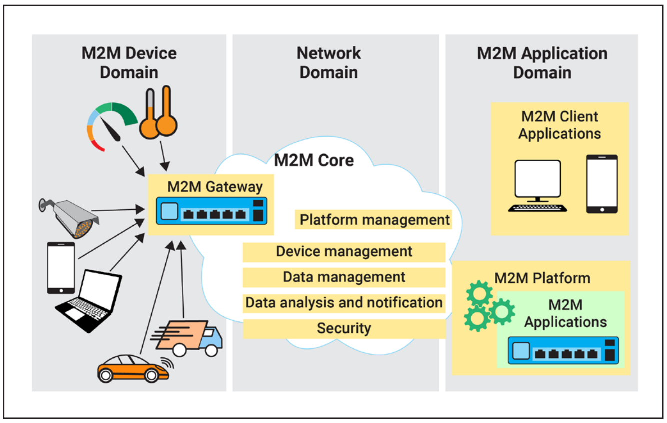
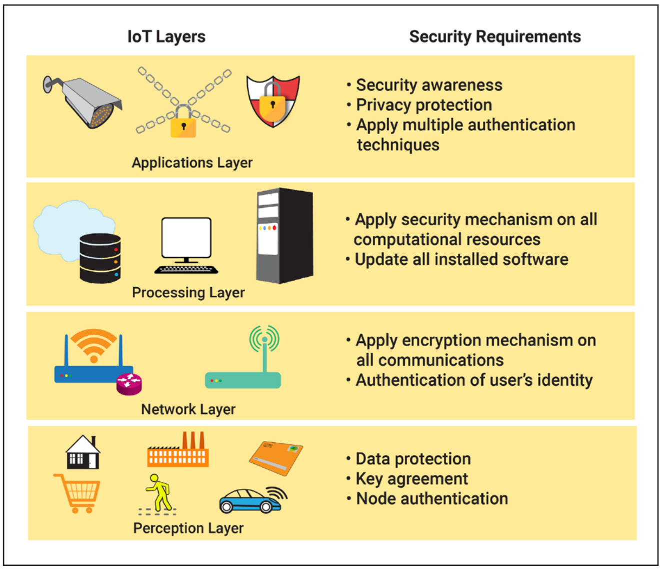
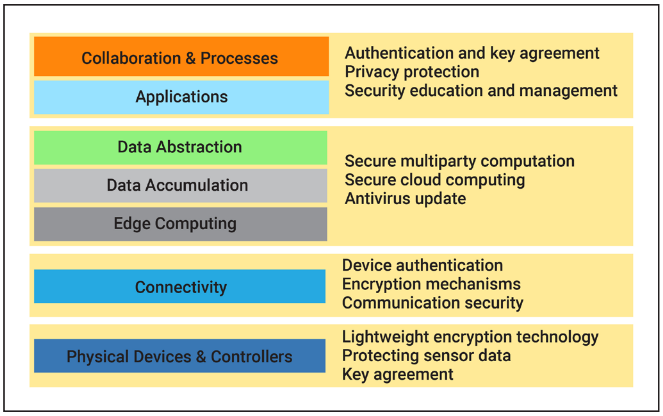


Figure 9-30: Overview of M2M platform functions in M2M architectures.



IoT Layers	Security Requirements
Applications Layer	• Security awareness • Privacy protection • Apply multiple authentication techniques
Processing Layer	• Apply security mechanism on all computational resources • Update all installed software
Network Layer	• Apply encryption mechanism on all communications • Authentication of user's identity
Perception Layer	• Data protection • Key agreement • Node authentication

Figure 9-31: Summary of IoT security requirements at different layers of IoT architectures.



Collaboration & Processes Applications	Authentication and key agreement Privacy protection Security education and management
Data Abstraction Data Accumulation Edge Computing	Secure multiparty computation Secure cloud computing Antivirus update
Connectivity	Device authentication Encryption mechanisms Communication security
Physical Devices & Controllers	Lightweight encryption technology Protecting sensor data Key agreement

Figure 9-32: Examples of IoT security recommendations for different IoT architecture layers.

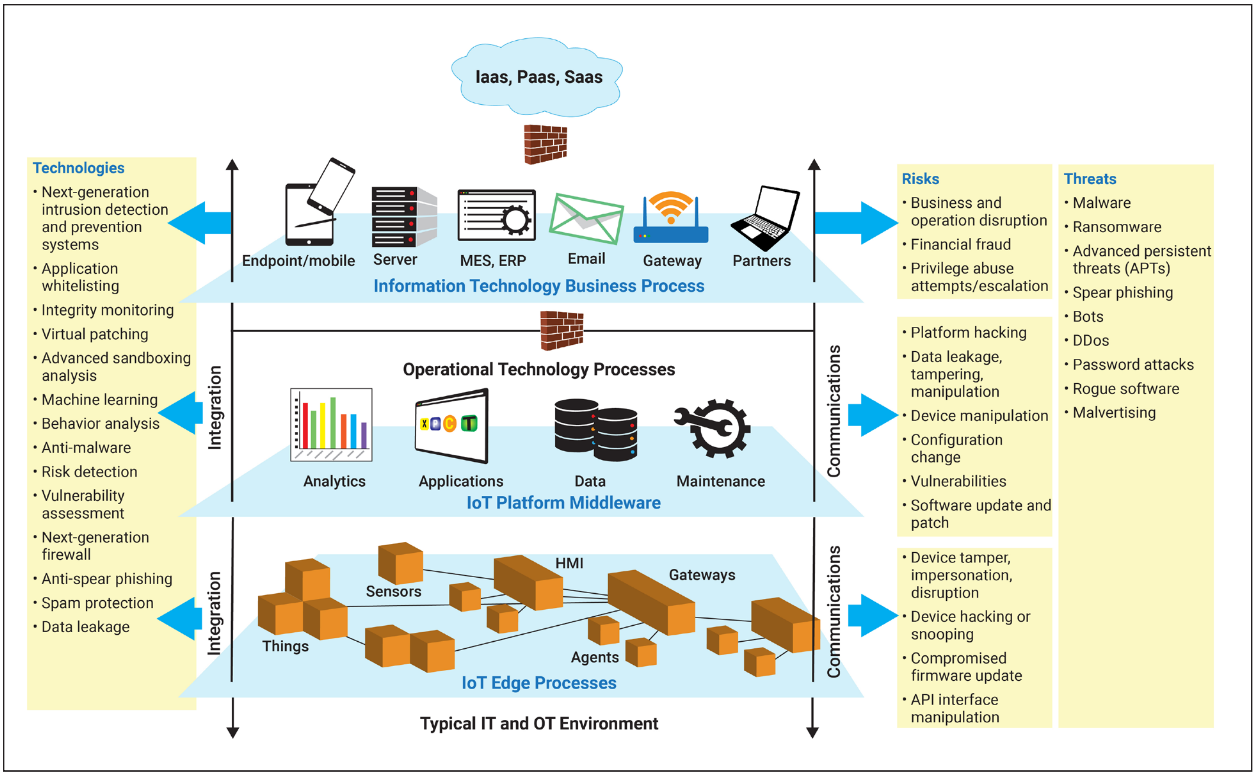


Figure 9-33: Examples of IIoT security risks, threats, and technologies.

or processes. OT is included in SCADA systems and other types of industrial control systems. Because of the functions it performs, OT can play a key role in IIoT security. Most of the security risks, threats, and technologies illustrated in Figure 9-33 are discussed in Chapters 11 and 12.

CHAPTER 10

Appendix 10-1: Additional Benefits of Network Life Cycle Models

In addition to helping to organize network design and management activities, network life cycle models have other benefits. For example, they can contribute to a lower cost of network ownership, increase network availability, facilitate access to applications and services, and improve business agility.

A network life cycle approach can contribute to lower total cost of network ownership by:

- Identifying and validating technology requirements
- Planning for network infrastructure changes and identifying their associated resource requirements
- Developing a network design that aligns with business goals and technical requirements
- Accelerating successful implementation of network infrastructure and applications
- Improving network efficiency and manageability
- Ensuring that the network is properly aligned with operational processes

Network availability is always a top priority. In many enterprises, network downtime results in revenue loss. In an automobile factory, such as the BMW facility in Spartanburg, SC, each minute that an assembly line stops because of a network outage equates to thousands of dollars in lost sales revenues. It doesn't take long before the losses exceed the annual salaries of the network and production managers on duty when an outage occurs.

A network life cycle model can contribute to improved network availability by:

- Assessing the availability of the existing network and the potential of the proposed design to increase availability
- Identifying/specifying hardware and software with high availability (e.g., MTBF) statistics that align with the organization's business goals and technical requirements
- Staging and testing proposed changes to the network before they are fully deployed
- Assessing the security of the current network and the potential for the proposed network design to reduce vulnerabilities and network downtime
- Assessing the manageability of the existing network and the proposed design's potential to improve manageability
- Proactively monitoring the network and assessing availability trends
- Proactively identifying potential security breaches and defining remediation plans to minimize network downtime

In today's businesses, having access to network applications and services is crucial to productivity. A network life cycle model facilities access to applications and services by:

- Assessing the organization's ability to continue to support the existing network, and its preparedness to support planned technologies and services
- Improving the efficiency and effectiveness of network service delivery by increasing network availability, capacity, and performance
- Improving the availability, reliability, and stability of network applications and increasing the uptime of the enterprise network
- Identifying, quickly diagnosing, and quickly resolving problems affecting network services and applications

As Chapter 9 illustrated, many enterprises quickly adopt and use new technologies, such as the IoT, especially when they have the potential to improve operations or achieve business goals. Enterprises with networks that can quickly adapt to new technologies and applications can gain competitive advantages over other businesses. A network life cycle model can help businesses be more agile by:

- Identifying business requirements and technology strategies for enterprise networks
- Integrating business goals and technical requirements into network designs
- Readying business locations and facilities to support new technologies and applications
- Ensuring that business locations and facilities have the resources needed to quickly install, configure, and integrate new technologies and applications
- Continually monitoring the performance of applications and technologies on the network and identifying ways to improve their performance

Appendix 10-2: Core, Distribution, and Access Layer Design Considerations

Hierarchical network architectures are common in the enterprise campus and in enterprise branches because of their numerous advantages when their layers are appropriately designed and deployed.

Core Layer Design Considerations

Common design goals for the core layer include 100% uptime, throughput maximization, and facilitating network traffic growth. Core layer technologies include routers and multilayer switches; hot-swappable or standby (redundant) routers and switches that can automatically take over when a primary router or switch fails; load balancing technologies; routing protocols that scale well and quickly converge; redundant links to automatically overcome link failures or help with load balancing; and standby generators and/or uninterruptible power supplies (UPSs). Partial- or full-mesh topologies are best for enabling the core layer to achieve its design goals.

Distribution Layer Design Issues

The primary technologies at the distribution layer are Layer 3 devices (routers or multilayer switches). These devices perform numerous functions critical for meeting fundamental network design goals, including filtering and managing traffic flows, enforcing access control policies, summarizing and advertising routes to core layer routers, isolating the core layer from access layer failures or disruptions, routing frames between VLANs, and prioritizing traffic sent through the core. Designers often recommend redundant links among distribution layer routers and switches for load balancing and to improve application performance.

Distribution layer networks usually have a partial-mesh topology that provides enough redundant paths to ensure that the network can survive a single link or a single router/switch failure. When distribution layer devices are placed in the same wiring rack, equipment room, or data center, they are typically interconnected by gigabit links; when the devices are further apart, fiber optic cable is used to connect them. Because switches capable of supporting multiple high-speed fiber connections are expensive, designers often carefully plan to ensure that there are enough available fiber ports and redundancy. Routers or multilayer switches are usually deployed in pairs with access layer switches evenly divided between them. With this configuration, the failure of a single router or switch does not cause the network to go down.

Access Layer Design

The access uses Layer 2 access points and switches to provide network access to users; both wired and wireless access is typically provided. Because wired Ethernet has distance limitations, the physical location of access layer infrastructure is an important design consideration, and designers often turn to structured cabling systems for guidance when designing this layer.

Wiring closets/telecommunication rooms serve as termination points for cabling within buildings. Their size and placement should be consistent with network expansion plans and application support needs. Today, most access layer switches have Power-over-Ethernet (PoE) functionality and can be used for both power and communications by a variety of devices, including most Wi-Fi access points.

Access layer devices in data centers or server farms are typically redundant multilayer switches that combine switching and routing functionality. Often they also provide firewall and intrusion protection functions for the servers and storage devices found at these locations. Redundant components and failover strategies can be implemented at the access layer to improve application reliability and to increase availability for access layer devices.

Improving the manageability of the access layer is often a major concern for network designers. This is often driven by increases in the number and variety of devices connecting to the network and new access layer technologies, such as RAN base stations and IoT devices. In addition to providing basic connectivity at the access layer, the designer should consider several factors that can improve access layer manageability, including:

- Access layer naming structures (e.g., host names that indicate physical location)
- VLAN architecture
- Application traffic flows
- Application traffic priorities

As noted in Chapter 6, most Ethernet networks have a physical star topology, and each device has a single direct connection to a single switch (Layer 2 or multilayer). Many businesses consider the cost of having redundant wired links for devices to be too expensive but are supportive of having a wireless overlay network to enable devices to access the network if wired links fail. However, if network affordability is not a factor, it is possible to configure an access network with redundant wired links and switches to ensure availability.

VLANs and IP subnets are the most common methods for segregating user communities and application traffic at the access layer. In today's enterprise networks, designers usually strive to restrict VLANs to a single wiring closet. This design approach increases the number of VLANs in a network and may increase the number of IP subnets; however, the recommended practice is to associate a single IP subnet with a single VLAN. IP addressing at the access layer (e.g., the calculation of IPv6 addresses from MAC addresses) is an important design issue that can affect the scalability of the entire network.

Appendix 10-3: Additional Network Design Considerations

Logical and physical topologies, IP addressing, VLANs, bandwidth, copper versus fiber optic cabling, and Wi-Fi and RAN coverage are among the most obvious network design considerations. However, this is far from an exhaustive list. As this chapter indicates, there is a lot involved in designing a network, and it can be easy for some very important issues to be overlooked, including:

- Regulatory requirements. Legal and regulatory factors can affect the logical and physical design of a network. For example, local fire and business codes can limit options for structured cabling. They may require the use of plenum cable or conduit that minimizes the chances of toxic fumes/smoke in the event of fires. Similarly, the U.S. National Electrical Code includes requirements relevant to providing electrical power to network devices. Other regulations, like HIPAA, PCI DSS (the PCI data security standard), and GDPR (the EU's General Data Protection Regulation), can affect how the network must handle data in transit and data at rest. Network designers must factor in such requirements to ensure that compliant network designs are recommended.
- Network resilience and redundancy. The importance of network availability to business operations means that enterprise networks need some level of fault tolerance. To make that happen, various levels of redundancy, such as the N+1, 2N, or 2N+1 redundancies found at different data center certification tiers, are often part of enterprise network design. Such resilience and redundancy add cost and can affect affordability goals, but aiming for five-nines availability (99.999% uptime) is a common network design goal, especially when the business cost of downtime (e.g., how many dollars per minute/hour the business loses when the network goes down) is high.
- Cloud versus on-premises. Once business and technical requirements are identified, designers should carefully consider how much cloud infrastructure to include in the redesigned or new network, because it's no longer a given that on-premises infrastructure is best for supporting business applications. Designers should avoid being prejudiced toward on-premises solutions when the cloud may be a better fit (or vice versa).
- Cooling and power. It can be easy to overlook a network's power and cooling requirements. If the power requirements for the new network cannot be met, the network cannot be deployed. Similarly, if the design overlooks the heat dissipation needs of new equipment, devices can overheat and prematurely fail. Important points for designers to consider include:

    - Ensuring that electrical panels and outlets can accommodate the new equipment
    - Accounting for PoE loads when sizing UPS backups and other power equipment
    - Ensuring that equipment room cooling can handle the additional heat generated by new network equipment, including consideration of supplemental cooling for equipment rooms
- Network design is never complete. Network design work is never finished. Once the new network is deployed, it should be evaluated to verify that it meets its business and technical requirements. Both performance and security should be evaluated, and tweaks or fixes should be applied where they are needed. There is also an ongoing need to incorporate new services, features, and applications to positively affect network performance. In short, after implementation, network personnel will be maintaining, modifying, updating, and striving to optimize the network.

Technologies selected for the new network should align with bandwidth and QoS requirements, the network topology, business requirements and constraints, and technical goals. Analysis of traffic flows and loads can help designers select devices with appropriate capacities. Assessing the existing network and characterizing its traffic also helps designers select appropriate protocols, WAN technologies, and services for connecting geographically distributed sites in the enterprise network. Often, businesses seek to minimize WAN costs without sacrificing network performance, security, or application support. Selecting WAN service providers and crafting appropriate service level agreements (SLAs) are important considerations for network designers during this phase.

Plans for acquiring components and services for the new network may be developed after selections have been made. For components such as cabling, switches, and routers, organizations may send a *request for quotations (RFQ)* to network hardware vendors. Some organizations issue a *request for proposal (RFP)* to networking consultancies and vendors to solicit their input on network design and technologies. An RFP may be developed after the business and technical requirements for the network are determined and after the current network has been documented. The RFP typically invites interested companies to recommend a network topology and to be involved in selecting and installing network components and services.

Appendix 10-4: A Closer Look at NMSs and Network Management Protocols

As noted previously in section 10.5.1, an NMS is one of the key components of a network management architecture. An NMS includes applications that monitor and control managed devices and provides most of the processing and memory resources required for network management.

Other important architectural components include managed devices, management information bases (MIBs), and agents that report MIB information to the NMS and initiate action in managed devices under its control. The protocols used to transfer management data and information between agents and the NMS are also considered to be a part of network management architecture. They are conceptually illustrated in Figure 10-13.

In a network management architecture, a **managed device** is a network device that can be configured, administered, managed, provisioned, monitored, or otherwise directed by network management software. In many NMSs, managed devices are network nodes that contain Simple Network Management Protocol (SNMP) or other (e.g., streaming telemetry) agents.

An **agent** is a network-management software module that resides in a managed device. It has access to the device's management information base and translates that information into a form compatible with SNMP or another network management protocol.

The NMS and other network software applications can read and display the MIB in managed devices. An *MIB* is a hierarchically organized collection of information about the managed device that contains all the data the device can make available to the NMS.

An MIB stores the information gathered by the local agent on a managed device. Each object stored in an MIB has a unique identifier. Network management applications use the identifiers to retrieve specific objects.

As noted in section 10.5.2, NMSs use network management protocols to communicate with agents and managed devices. SNMP and RMON are examples of traditionally used network management protocols. Both have evolved over the years: SNMPv3 and RMON2 are current versions of these protocols.

SNMPv3 uses an "EngineID" identity to uniquely identify each SNMP entity (device, agent, NMS). The EngineID is used to generate the key used for authenticating messages passed between SNMP entities. As SNMP messages are created, they are given a special key that is based on the EngineID of the entity. The key is shared with the intended recipient and is used by the recipient to decrypt the message. This is how SNMPv3 protects against the modification of management information passed between agents and NMSs. Relative to SNMPv1 and SNMPv2, SNMPv3 makes it easier to remotely configure SNMP agents.

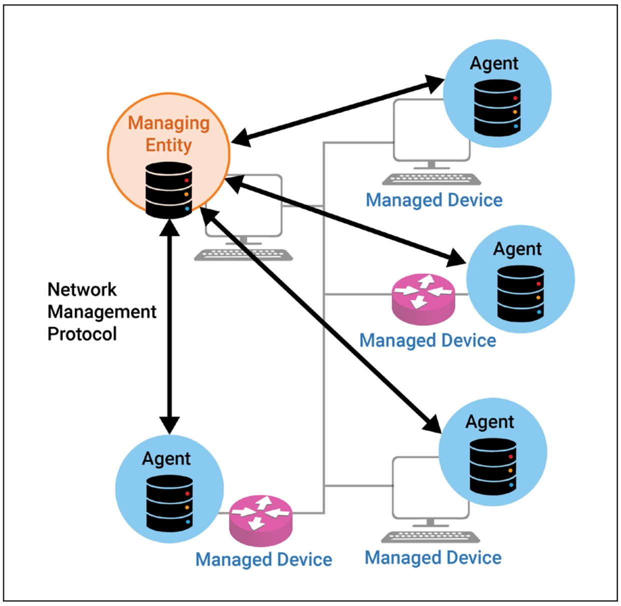


Figure 10-13: Managed devices, agents, and network management protocols.

RMON MIBs enable network managers to acquire information about the health and performance of the network segment on which an RMON agent resides. RMON probes (agents) can be dedicated devices (used for gathering flow statistics) or can be software embedded in a network device like a router, switch, or server.

Appendix 10-5: The FCAPS Model in More Detail

Performance Management

According to the ISO's description of performance management, this category of network management processes focuses on measuring network behavior and effectiveness. This includes examining the behavior of network applications and processes, response time, the reachability of network services and resources, and logging changes to network traffic routes.

Performance management helps organizations determine whether service level agreements (SLAs) are being met, facilitates network optimization, and informs plans for network expansion. Monitoring performance involves collecting data from managed devices, processing and displaying some or all collected data, and archiving some or all the data for future analysis.

The ISO identifies two types of performance that should be monitored: end-to-end performance and component performance:

- End-to-end performance focuses on traffic flows across the network. This can include measuring availability, capacity, utilization, delay, jitter, errors, throughput, reachability, response time, and the burstiness of network traffic. Bursty network traffic is characterized by sudden and unexpected surges in data volume, interspersed with periods of low or no traffic. The data and information that are collected may be used to improve the overall performance of the network (e.g., throughput and response times) for all applications and network users or to better support specific user communities or applications.

- Component performance measures (per-link; per-element) assess the performance of individual devices or links. For example, throughput and utilization of a particular link or network segment may be measured. Throughput (forwarding rates in packets/frames per second) and errors may be measured for individual routers and switches in the network.

In large networks, performance management may include polling remote network sites to assess reachability and response times. Ping packets or ICMP echo packets may be sent and used to measure round-trip response time (RRT). Traceroute may be used to assess traffic route changes.

Protocol analyzers (such as Wireshark) and SNMP tools may be used to assess traffic loads between important network sources and destinations. Documenting source/destination traffic loads is useful for capacity planning, troubleshooting, determining appropriate router configurations, and assessing SLAs that include throughput requirements.

As noted in section 10.5.3, performance management can also be used to identify QoS requirements for applications and for traffic engineering. *Traffic engineering (TE)* is concerned with optimizing the performance of a network. It involves measuring, characterizing, modeling, and controlling network traffic flows to achieve specific performance objectives. Examples of the purposes of TE include:

- Providing more reliable traffic delivery
- Making better use of network resources
- Making network performance predictable

Traffic engineering sometimes involves forcing network traffic to take paths that are different from those chosen automatically by the routing protocol(s) in use in the network. It may also involve *traffic shaping*, which limits the bandwidth that can be consumed by certain applications to ensure adequate performance of critical applications. This basically involves giving the data packets of critical applications a higher priority than the packets for other applications. Shaping slows down or delays some types of network traffic. In some cases, some data traffic may be dropped or discarded when it is outside performance boundaries for that type of traffic flow.

Bandwidth (network) throttling may also be used to manage/control network traffic. This involves the intentional limitation of the communication speed of ingoing or outgoing data from a specific network node or device. For example, an ISP or web server might intentionally slow Internet service to a specific client. This is a reactive mechanism used in communication networks to regulate network traffic and manage congestion, especially that being caused by specific devices or users.

Fault Management

Fault management involves detecting, isolating, diagnosing, and fixing network problems. It also involves reporting problems to network managers and users and processes for tracking trends related to the problems that are detected. In some cases, network managers may have to identify and deploy workarounds until a problem can be corrected.

Enterprise network users expect speedy and reliable resolutions for network problems. Many expect to be informed about the status of network resolution processes and to be provided with estimates for when the network will return to normal operations. Most organizations have trouble ticket systems that document network problem reports and how problems are resolved. A *trouble ticket system* is a software tool an organization uses to track the detection, reporting, and solving of tickets from network users.

A *trouble ticket* is created when a network user submits a help request via a problem tracking system. It typically includes details about the nature of the problem the end user is experiencing with a specific network component. The trouble ticket is electronically forwarded to an appropriate IT technician, who is responsible for addressing the user's issues based on their severity, effect on the organization, when they are received, or other factors. Users typically receive trouble ticket submission confirmation messages as well as trouble ticket closure messages after their problems are resolved. Trouble ticket systems are common features of IT help desks in today's businesses.

A variety of other tools are used for fault management, including monitoring tools that alert network managers of problems and protocol analyzers that are used in problem diagnosis and resolution. Monitoring tools are often based on SNMP and RMON standards. Most operating systems used on network servers and devices enable processing faults/problems to be reported to network managers.

Speedy problem identification and resolution help to minimize network downtime and unacceptable reductions in network performance. They also contribute to helping to optimize the experience of network users.

Configuration Management

Configuration management helps network managers keep track of managed devices; maintain information on how devices are configured; save default configuration details to facilitate the configuration of similar network elements (this is part of element management); modify default configurations; and install configurations on network devices.

Configuration management also helps network managers maintain an inventory of network assets and resources and keep version logs. Version logging involves keeping track of the versions of operating systems and applications running on network devices. Network asset inventories can also involve maintaining information about the hardware configurations of network devices (e.g., amount of RAM, cache, or Flash memory) and the type of cabling that connects devices to the network.

Automated software distribution is an important part of configuration management in many businesses. Software distribution is the process of distributing business software and applications to network users without user intervention. This may involve distributing software updates to all users of specific platforms, such as Windows, Macintosh, or Linux. Software distribution tools enable network administrators to automate and monitor the software applications delivered across the network and supervise their use. Software distribution applications can also help network personnel prohibit the use of certain types of applications or block the installation of unauthorized executable files.

Another commonly used tool for network software management is disk imaging. *Disk imaging* is a type of hard drive backup that places all data on a hard drive in a compressed file that can be stored on another device, in a network file system, or in the cloud. Disk imaging allows businesses or individuals to recover all data that was on a computer when the image was made.

Network backups may also be used. A network backup is the process or result of copying current data for all nodes and end devices in the network and securely storing it. Network backups are an integral part of IT backup and recovery, and they help ensure quick recovery from disasters.

Backing up device configurations is also common in enterprise networks. Configuration backup involves saving existing network configuration files and creating a repository with all versions stored in incremental versions. Configuration backups are typically encrypted before being stored to ensure their security.

Configuration backups help networks recover from network disasters, such as data breaches or full network outages. In such instances, network administrators can use a stable configuration version from the repository to restore network device configurations.

Configuration management contributes to performance management by helping to ensure that managed devices are appropriately configured to support network applications and services. It can also contribute to security management by ensure that security patches are effectively deployed to devices throughout the enterprise network.

Security Management

As you have probably gathered from previous chapters, security management in today's enterprise networks is multifaceted. It includes processes for authenticating devices, authorizing access to network resources, and controlling network access. It also involves processes for generating and distributing encryption keys and the use of tools for analyzing router and switch configurations for compliance with security standards.

Security management also involves network intrusion prevention and detection, minimizing damage from data breaches and other attacks, and recovering from security breaches. The importance of robust network security is an important theme throughout this book, so it should come as no surprise that it is also a critical consideration for network design and management. You will gain a broader understanding of security management in Chapters 11 and 12.

Security depends on network management to configure, monitor, manage, and verify security levels throughout the network. Since there is often a need for network managers to access the network during attacks, it can be important to have backup out-of-band access to network devices during attacks where in-band access is not available.

As noted previously, security and performance pose design trade-offs because security mechanisms can affect network performance. When high security is a network design goal, security mechanisms that affect traffic flows can negatively affect application behavior and performance. Achieving an acceptable balance between security and the end-to-end performance of critical applications or user communities can be challenging.

Accounting Management

The goal of accounting management is to gather usage statistics for network users. Also known as billing management or simply "administration," accounting management involves processes for identifying network usage data and compiling a comprehensive inventory of network assets.

Accounting management tracks network utilization information for individual users, user communities, departments, or business units. In the past, some businesses had "charge-back" schemes that allocated network costs based on the utilization of network resources. Such schemes date back to the days when IT was considered a cost center and there was a perceived need for an accounting system that identified the individuals or business units most responsible for running up the costs of IT services. Today, such schemes are less common and "administration" replaces "accounting" in the FCAPS acronym. Still, it is often useful to track network statistics, such as utilization, data storage, and cloud service usage for applications or various groups of network users for capacity planning, network expansion planning, and other types of proactive network management.

Today, accounting management is primarily concerned with determining the utilization of network services by applications, individuals, or groups of users. This enables network services to be apportioned fairly among applications and users while also reducing the potential for network congestion.

Network User Management

User management is part of accounting/administration management. It is also a key part of *identity and access management (IAM)* processes and network directory services. Managing and controlling user access to network resources is a fundamental part of network security. User management enables network administrators to control user access to the network and its various resources and is an essential part of on-boarding and off-boarding network users. Network directory services authenticate, authorize, and audit user access to network resources based on what network administrators allocate to individual users.

User management enables network administrators to manage and maintain the security of network resources while provisioning users based on their needs and roles in the organization. For example, the marketing team typically requires access to different resources than the accounting team. Someone on the marketing team is unlikely to need access to internal financial systems, and someone in accounting is unlikely to require access to Salesforce. For end users, the tasks of user management are usually behind the scenes and invisible. They are usually content to have secure, frictionless access to the network resources they need to do their jobs.

User management includes user account creation, maintenance, and deletion. Today, it often includes the assignment of users to VLANs, end-to-end management of user accounts (including user registration, login, and authentication), single sign-on (SSO), and permissions management. User management functions include:

- Preventing unauthorized access to network infrastructure, applications, and data
- Storing user details and credentials
- Providing convenient login mechanisms for network users
- Enabling users to set and reset passwords
- Ensuring multifactor authentication (MFA)
- Assigning user rights to network services, applications, and systems
- Managing user rights (entitlements) within services and applications

User management also contributes to zero trust security initiatives in enterprise networks. Zero trust calls for strict authentication for all connections, internal and external. It maintains that networks and systems should only grant users the minimal privileges they need to perform their roles.

Strict implementation of user access and identity verification is a core principle of zero trust security strategies. Strict access management means that if attackers compromise a network user's account, they cannot do anything beyond the account's authorized privileges. If attackers compromise a network administrator's account or device, zero trust access systems can flag anomalous account use and block malicious activity. Zero trust also requires continuous monitoring of network user behavior, which enables rapid detection of and response to user accounts that have been compromised.

Appendix 10-6: A Further Look at Network Audits

Network auditing typically uses both automated and manual techniques to gather data and evaluate the network. Network audits often include reviews for:

- Each network node
- Network control and security processes
- Network monitoring and management processes

Network auditing helps organizations take a close look at their network infrastructure. For many enterprise networks, network infrastructure includes both physical and virtual devices, both on premises and in the cloud. There may be multiple networks in the mix, some public and some private, some wired and some wireless.

Performing networking infrastructure audits helps organizations determine whether their networking assets are up to date. It also provides visibility for the network architecture and helps identify real or potential bottlenecks that can affect network performance.

When network infrastructure is audited, all switches, routers, PCs, servers, mobile devices, and wireless access points that exist on the network should be identified and accounted for. This should include both on-premises and remote hardware, including that used by remote workers. Data for all these devices should be included in a database. In addition to identifying devices and their locations, the database should specify:

- The last time the device was updated or replaced
- The end-of-life date for the device
- Whether the device is properly labeled with a physical ID tag or similar identifier and, if so, what the identifier is
- Whether the device's environmental conditions are adequate
- The static IP address (or addresses) of the device, if applicable, or a note specifying dynamic IP address allocation via DHCP
- Special requirements related to the IP address (For example, does the device need to have a specific IP address or be associated with a specific subnet or VLAN?)
- The device's MAC address

Network infrastructure audits are opportunities to update network diagrams and maps that show the physical locations of devices and the networks that connect them. Network audits also consider network software. There are two major types of software on today's enterprise networks. The first type is network applications and the operating systems that support/host them. The other type includes software-defined networking components or resources such as virtual machines, VPNs, and software firewalls. Network audits should assess the current state of software resources and the network's ability to support them. For this part of a network audit, it is important to determine:

- Whether the resource is a standard software resource (an application or operating system) or software-defined component
- The last time the resource was updated or replaced
- The amount of bandwidth that the resource consumes and whether there is enough bandwidth available to adequately support it
- Any IP addresses that the resource uses and how they are assigned
- If applicable, licensing information for the resource and whether licenses are up to date

Today, enterprise network audits primarily focus on network security and control. However, many also review the processes and measures that the organization uses to ensure the availability and performance of critical applications. When network security data is gathered, vulnerabilities and threats are identified in the formal audit reports that are sent to network administrators.

Network security audits should focus on:

- Network data security. This should consider both data in motion and data at rest, access controls that restrict who can view the data, inclusion of personal data in data repositories, and processes used to anonymize personal data.
- Network access controls. This should identify authentication protocols and procedures used to protect the network, how new devices are added to the network, access controls used for both wired and wireless connections, the extent to which up-to-date authentication protocols are used, and avoidance of protocols (such as WEP) with known security flaws.
- Physical security. This should specify how physical access to the network is secured and where additional physical security controls should be used.
- Network security and use policies. This should identify user password strength and password change policies, BYOD policies, and how such policies are enforced.

In short, network audits involve collecting and analyzing large amounts of data. Regular network audits should be viewed as an important part of network management. They help organizations identify opportunities to optimize and to identify and address security vulnerabilities before they become critical issues.

Appendix 10-7: A Closer Look at Enterprise Mobility Management

Enterprise mobility management (EMM) essentially consists of a set of services and software technologies deployed on employees' mobile devices that contribute to securing corporate data, applications, and intellectual property. For example, if an employee's device is lost or stolen, EMM technologies can lock or wipe it. EMM can also enforce the use of multifactor authentication that includes at least one biometric identifier when logging in to the enterprise network. In sum, EMM provides enterprise network managers with the ability to enforce security policies on mobile devices. MDM, MAM, and MIM are some of the key aspects of EMM:

- MDM (mobile device management) is an underlying EMM technology that enables mobile devices to be managed remotely. Basically, MDM enables network managers to provision, configure, and manage mobile devices that are permitted to access corporate systems and data. MDM involves the installation of unique profiles on mobile devices that enable organizations to remotely manage, control, encrypt, and enforce security policies on the devices.
- MAM (mobile application management) tools allow network managers to manage applications on mobile devices. MAM addresses the deployment and updating of mobile apps and enables security policies to be enforced for specific apps. It also enables network administrators to selectively remove apps and their associated data from mobile devices.
- MIM (mobile identity management) is another EMM component. MIM can take various forms, including user and device certificates, single sign-on, app code signatures, and authentication. The primary goal of MIM is to ensure that only trusted devices and users can access enterprise network applications or data.

CHAPTER 11

Appendix 11-1: The SolarWinds Supply Chain Attack

The SolarWinds supply chain attack that affected hundreds of private companies and government agencies went undiscovered for more than nine months. It was a sophisticated malware campaign attributed to Russian intelligence personnel that leveraged tainted software from SolarWinds, an IT management company, and other hacking techniques to breach the networks. The attackers also succeeded in breaching the email accounts of top officials at the DHS, the government agency responsible for defending the United States from foreign cybersecurity attacks.

SolarWinds sells network monitoring and management software that enables an organization to see what's happening on its networks. In the attack, hackers inserted malicious code into an update of Orion, the company's software platform. The tainted update was installed in the Orion platforms of about 18,000 SolarWinds customers, and attackers then used the malware to breach the networks of about 60% of the victims. Hackers deeply penetrated nine federal agencies and 100 private companies. The full extent and damage of the attack are still being investigated and assessed.

Ransomware and some other types of attacks, such as DDoS attacks, are among the easiest to detect because their effects are the most immediate. Cyber espionage and passive attacks, such as eavesdropping, are more difficult to detect because the attackers typically do everything possible to hide their presence. Sometimes, they are not discovered until months after the initial penetration.

The SolarWinds attack was discovered when FireEye, one of its victims, identified the theft of its cybersecurity tools. Other APT attacks have been discovered when unusual password activity, missing data, or higher than normal network usage has been identified.

Appendix 11-2: A Closer Look at SIEM Systems

SIEM combines *security information management (SIM)* and *security event management (SEM)* functions in a single security management system. The guiding principles of SIEM solutions include the aggregation of relevant data from multiple sources, identification of deviations from normal operations, and taking appropriate actions in response to unusual or suspicious activity. For example, when a SIEM system detects a potential issue, it generates an alert, logs additional information, and provides instructions for stopping the unusual activity.

Basically, SIEM solutions enable organizations to collect, centrally aggregate, and analyze log data from all of their digital assets. Organizations can re-create past security incidents, analyze new ones, investigate suspicious activity, and identify opportunities to improve their responses to security incidents and other security processes.

Some SIEM systems are rules-based, and others use a statistical correlation engine to establish relationships between event logs used to determine whether network activities are normal or unusual. When the SIEM software identifies suspicious events/activities, it generates security alerts. Using predefined rules, organizations can set these alerts as low or high priority. For example, a user account that generates 25 failed login attempts in 25 minutes may be flagged as suspicious but low priority because the user's behavior is consistent with the profile of a user who has forgotten their login information. However, a user account that generates more than 100 failed login attempts in 10 minutes would be flagged high priority because it is consistent with the profile of a brute-force attack.

Advanced SIEM systems often include user and entity behavior analytics (UEBA) to help detect security incidents; these lateral movement and APT detection tools are used to protect internal networks. SIEM systems also include security orchestration, automation, and response (SOAR) to support incident response teams.

SIEM systems work by deploying multiple collection agents to gather security-related events from security equipment (e.g., firewalls, intrusion prevention systems), networking equipment (e.g., routers and switches), servers, user devices, and applications. The collectors forward events to a centralized management console to support security analysts monitoring the network for security incidents. This system is illustrated in Figure 11-23.

In some SIEM systems, edge collectors perform preprocessing to reduce the amount of data/information sent to the central security management console and SIEM database. Advancements in AI, machine learning, and deep learning are helping SIEM systems to do a better job of flagging anomalies and continuously educating the system about the network's security environment.

Many SIEM systems include threat intelligence feeds from reputable external sources, and most provide regulatory compliance assessment and reporting tools. Forensics capabilities are common, and many systems enable packet headers and contents to be recorded to provide additional information about security events.

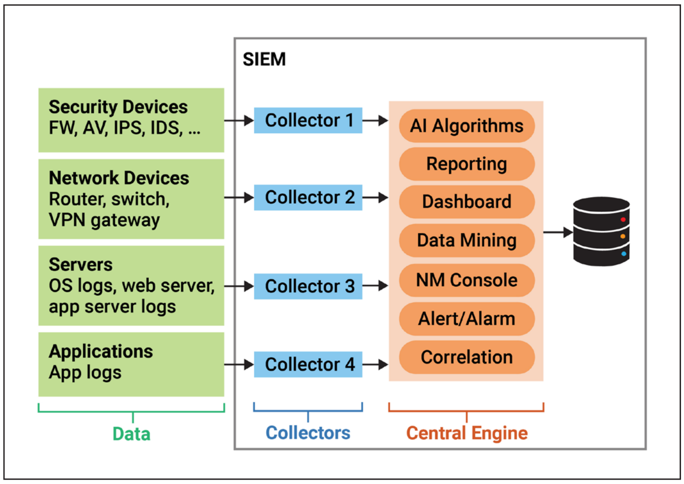


Figure 11-23: A high-level depiction of SIEM system operations.

SIEM makes it easier for enterprise network managers to filter large amounts of security data and prioritize security alerts. It increases the likelihood of detecting incidents that might otherwise go unnoticed. By analyzing log entries from different sources, SIEM enables security analysts to re-create the timeline of an attack, determine the nature of the attack, and assess its effects on business operations.

A SIEM system enhances incident response management by enabling the organization's incident response team to discover and contain breaches, identify compromised network resources, and deploy automated tools to combat attacks in progress. This shortens how long it takes to act to minimize attack damage.

Despite SIEM's potential benefits, there are several drawbacks. For example, it can take several months to successfully implement a SIEM system because it requires integration with existing security controls and network infrastructure. Also, SIEM systems are expensive; initial investments can be hundreds of thousands of dollars, which does not include costs for training network personnel and ongoing system support costs. Configuring the system takes time and expertise, and misconfigured SIEM systems can miss important security events.

SIEM products may be fully on premises (e.g., Cisco's Splunk) or cloud based (e.g., IBM's QRadar on Cloud). LogRhythm is a popular SIEM system among smaller organizations.

Appendix 11-3: Security Management Processes

Security architecture is an important topic in network design and network management. Designers strive to develop security architectures that appropriately align with enterprise network architectures. And security architectures provide guidance for deploying security management processes, such as those included in Figure 11-24.

As Figure 11-24 indicates, security management is multifaceted; however, its various aspects contribute to achieving confidentiality, integrity, and availability goals.

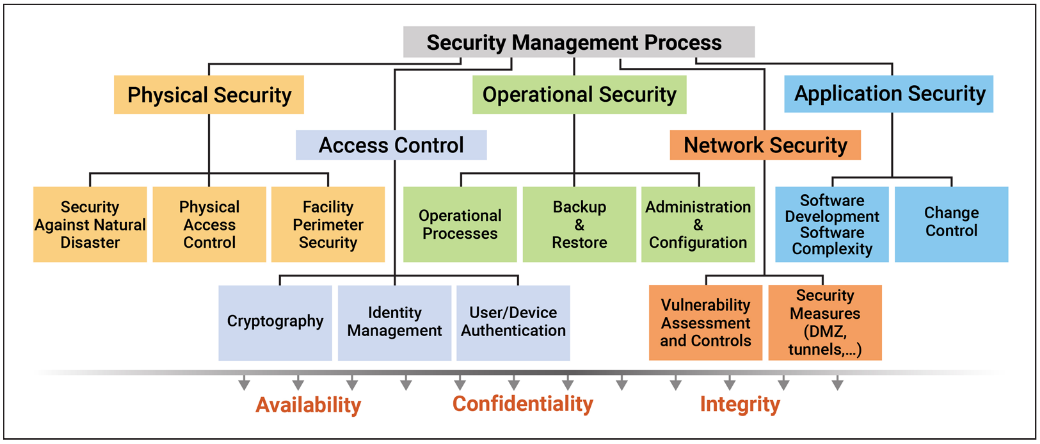

Security Management Process
Physical Security
Operational Security
Application Security
Access Control
Network Security
Security Against Natural Disaster
Physical Access Control
Facility Perimeter Security
Operational Processes
Backup & Restore
Administration & Configuration
Software Development Software Complexity
Change Control
Cryptography
Identity Management
User/Device Authentication
Vulnerability Assessment and Controls
Security Measures (DMZ, tunnels,...)
Availability
Confidentiality
Integrity

Figure 11-24: Network and computing infrastructure security management components and processes.

Appendix 11-4: The DiD in More Detail

Physical security defenses are found at the second layer of the DiD model. The purpose/intent of this layer is to provide physical safeguards against access to network and computing assets and to protect them from theft, sabotage, terrorist attacks, and other physical threats. Controlling physical access facilities that house networking and computing hardware is an important first line of defense for computing and communication facilities and data centers.

Identity and access controls may be considered physical or perimeter layer defenses. They involve ensuring that identities are verified before access is granted. They also ensure that access is limited to only what is needed, and that any changes made by those allowed entry are logged. Multifactor authentication may be used to verify identities, control access to infrastructure, and control who can make changes to network assets or conduct security scans or audits.

The goal of *network perimeter* defenses is to provide safeguards against network breaches or intrusions and network-based attacks against network assets and resources. Identifying breaches/attacks, generating alarms and alerts when they occur, and minimizing their effects are important aspects of network perimeter security. Examples of security mechanisms used at the network perimeter include DMZs, firewalls, proxy servers, network address translation (NAT), DoS and DDoS identification, and filtering tools and other technologies that identify attacks against the network and generate alerts to incident response teams.

Internal network defenses focus on limiting connectivity to network resources. Network users should only have access to resources that they require. It should not be easy for network users to move laterally through the network or access resources they are not authorized to use. Examples of security controls used at this layer include network segmentation and access controls, limiting inbound access to network resources via the Internet, implementing secure connectivity to on-premises networks, and using "deny by default" access controls for network resources. As Figures 11-15 and 11-17 illustrate, IDS and encryption are other security mechanisms used to protect the internal network.

The *host layer* of DiD focuses on ensuring that the network's endpoints and computing resources (e.g., servers, user devices) are secure. This is sometimes called the compute or endpoint layer. This layer strives to protect the operating systems of servers and devices that run applications by ensuring that they are current and patched. It also involves providing appropriate security for virtual machines. Anti-malware software, patch management systems, and endpoint protection (such as UEM) are examples of security mechanisms used at this layer.

Security at the *application layer* strives to ensure that applications are secure and free of vulnerabilities. As has been noted throughout this book, enterprise networks are application-centric. They are designed to support the applications that keep businesses running. Enterprise systems software (ERP, CRM, etc.) are usually critical systems that support core business processes and generate large volumes of transaction data. The data that they generate and use (e.g., customer, supplier, employee, and materials master data) is stored on secure storage media. Many businesses mandate that the applications they develop or use be developed using secure coding practices that strive to reduce vulnerabilities in program code. Authentication and authorization may be used to restrict access to applications.

Data security is the focus of the innermost layer of the DiD model. This is because data is the ultimate target of most attackers. The data they seek may be stored in a database, on disks that support virtual machines, in cloud-based SaaS applications or enterprise storage systems, or all of the above. Data in transit or data processed by applications may also be targeted. Some organizations are subject to regulatory requirements that mandate the controls they must use to ensure data confidentiality, integrity, and availability. For example, regulations may require sensitive data at rest to be stored in encrypted form. Data security involves but is not limited to ensuring that data is securely stored and access to data is controlled.

Defense in depth does not assume that it is possible to achieve complete or total security for network assets by deploying a collection of security mechanisms. Instead, it assumes that determined and persistent attackers are likely to find a way to thwart the defenses that they encounter. However, DiD layers are considered stumbling blocks that hinder the progress of attackers toward their goals and potentially slow and frustrate their progress until they give up or choose to seek easier targets.

Slowing/hindering their progress can also increase the chances of their attacks being discovered and triggering responses that minimize any damage they can do. Since data is the target of most attacks, it makes sense for organizations to place it behind several other layers of security and force attackers to find ways around the defenses at those layers before they can get to the data.

There are several other important observations to make about DiD. To start, each DiD layer includes one or more of the following types of controls:

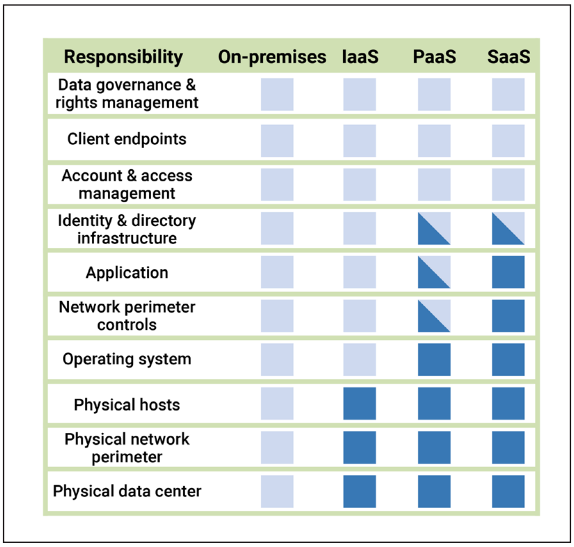
- Technical controls, such as software and hardware capabilities to keep threats (e.g., data breaches, DDoS attacks) from affecting network assets. Examples of technical controls are firewalls, IDS, IPS, and anti-malware software.
- Physical controls that strive to protect data centers, servers, and other physical assets from theft, tampering, and nonpermitted access. Access controls, alerts, alarms, online behavior monitoring, and surveillance systems are examples of physical controls.
- Administrative controls include security policies and procedures and their enforcement. Threat intelligence monitoring may also be considered an administrative control.

Also, some security practices, such as the following, are becoming increasingly common in organizations that embrace DiD:

- Least-privilege access. This is used to control user access and to minimize risks associated with unauthorized access. It basically means that a user's access privileges should be restricted to only the network resources that are needed for their job. It also means that their access rights to any given resource should also be limited. For example, a user may need access to certain types of records in a database but may not need to see all fields in the records.
- Multifactor authentication combines various user authentication processes to verify the user's identity and confirm authorized access to network assets. For example, a network user may be required to provide a strong password, a biometric identifier, and a single-session access code (sent by text or email) to access an application supported on the network.
- Network segmentation is commonly used to protect internal systems and data from third-party and "guest" network users by limiting what they access and use. For example, an organization may have separate wireless networks for internal and external users.
- Behavioral analysis monitors the behaviors of online users to keep tabs on nonstandard user or device behavior and to enforce security policies. Online behaviors that do not conform to security policies, or appear suspicious or aberrant, trigger alerts and may cause the user or device to be immediately disconnected from the network.
- Zero trust security practices are also being increasingly interwoven with DiD defenses.
- Encryption is universally used in DiD architectures to protect both data at rest and data in transit.

Appendix 11-5: Shared Security with Cloud Providers

As noted in Chapter 2, SaaS, PaaS, and IaaS are common cloud services that businesses use. Figure 11-25 compares security responsibilities for on-premises (private cloud) services to how security responsibilities are typically shared between IaaS, PaaS, and SaaS providers and their subscribers. In this figure, light color squares indicate that the business is fully responsible, and dark color squares indicate that the cloud provider is fully responsible. When the square includes both light and dark colors, the responsibility area is shared by the provider and subscriber.



Responsibility	On-premises	IaaS	PaaS	SaaS
Data governance & rights management	□	□	□	□
Client endpoints	□	□	□	□
Account & access management	□	□	□	□
Identity & directory infrastructure	□	□	◣	◣
Application	□	□	◣	■
Network perimeter controls	□	□	◣	■
Operating system	□	□	■	■
Physical hosts	□	■	■	■
Physical network perimeter	□	■	■	■
Physical data center	□	■	■	■

Figure 11-25: Typical division of security responsibilities between cloud providers and subscribers.

For all of these examples, data governance and rights management, endpoints, user accounts, and access management remain subscriber responsibilities, but many of the other responsibilities shift in whole or part to cloud providers.

Appendix 11-6: A Further Look at Email Security

Email security encompasses technologies and techniques that organizations use to protect email accounts and communications. Email is often considered to be an organization's largest attack surface and is typically the primary target of phishing attacks focused on compromising user accounts or spreading malware, including ransomware. Email security best practices include:

- Spam filters—helpful because spam attacks are a type of DoS attack
- Email encryption—to ensure that intercepted messages cannot be read
- Antivirus protection—to screen messages and attachments for viruses/malware
- Secure email gateways (SEGs)—to filter out potentially dangerous emails
- Multifactor authentication (MFA)—to ensure that more than one authentication factor is used to access email accounts
- Employee education—to help network users recognize social engineering, phishing, and other types of attacks that are typically executed using email

Appendix 11-7: Change Management and Internal Network Security

In cybersecurity, *change management* refers to the process whereby changes to network assets are announced, implemented, validated, and logged in ways designed to reduce the risk of the changes affecting the availability of network assets or creating new vulnerabilities. Validation takes place after a change has been made and involves testing systems, applications, and devices to ensure that changes produce their desired results.

Network managers who approve changes should evaluate the effect of those changes and notify users and other stakeholders about why the changes are needed and when they will take place. Generally, changes should be scheduled during time spans when network traffic and network asset usage are low to minimize the effect of network/system downtime on users.

Change management should also include the ability to undo or reverse changes that do not produce their intended results. There should always be a way to bring the system back into a functioning state by reversing changes that fail to achieve their desired outcomes.

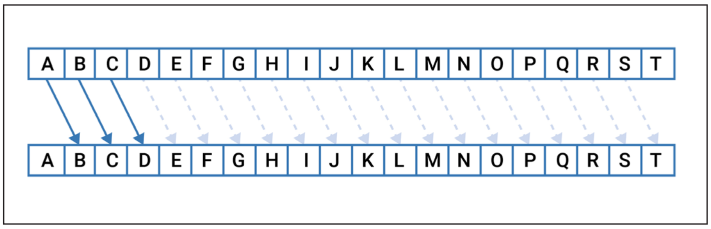
Device users often experience change management processes firsthand in the form of updates for device operating systems. Device owners are typically notified of the availability of software updates and often can choose the time for the update to be installed.

Users typically schedule installs during times when the device is not needed to perform job responsibilities. Should something go awry during the install, automatic reversal of changes typically occurs to return the device to its state prior to the installation failure.

While change management may not seem to be a big deal or major cybersecurity control, it is very important because most updates made to enterprise network components include security upgrades. Keeping software patched and up to date is an important aspect of the overall security profile of internal networks.

Appendix 11-8: A Closer Look at Encryption

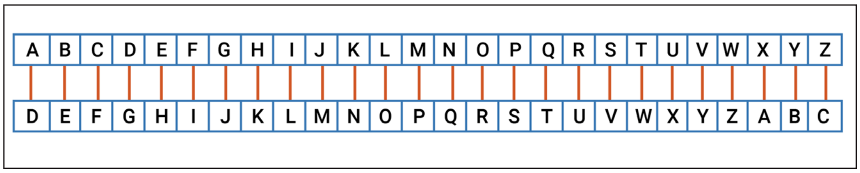
All encryption algorithms involve substituting one thing for another. This is easiest to see in simple encryption schemes such as a Caesar cipher. In a *Caesar cipher* for English text, each letter in the plaintext message is substituted with a letter that is *k* letters later in the alphabet. Figure 11-26 illustrates a one-letter shift (*k = 1*), and Figure 11-27 illustrates a shift of three letters (*k = 3*). If the latter were used to encrypt ENTERPRISE, the result would be HQWHUSLVH.



A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T

Figure 11-26: A Caesar shift of one letter (k = 1).

If someone intercepted a message encrypted using a Caesar cipher and recognized that a Caesar cipher was used, it would not take them long to correctly decipher the message because there are only 25 possible ways to substitute one letter for another using this type of encryption. For this type of encryption, the size of the shift (the value of *k*) is the encryption algorithm, and the letters used to substitute for the original letters (such as the bottom row of Figure 11-26) are the encryption key.

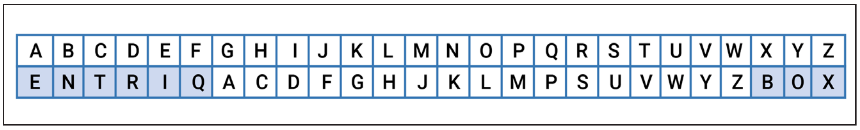


A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C

Figure 11-27: A Caesar shift of three letters (k = 3).

Monoalphabetic ciphers are more complex than Caesar ciphers. Like Caesar ciphers, *monoalphabetic ciphers* substitute one letter in the alphabet for another but without the regular pattern used for Caesar ciphers. Figure 11-28 is an example.

Because any letter in the alphabet can substitute for another, monoalphabetic ciphers have 26 (26 factorial) or 403,291,461,126,605,635,584,000,000 unique pairings of letters! If someone intercepted the message and recognized it as a monoalphabetic cipher, they would have to work harder to correctly decipher IKVISMSDUI as ENTERPRISE because it would take them longer to identify the key (the bottom row of Figure 11-28) even if they used brute force calculations.



A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
E	N	T	R	I	Q	A	C	D	F	G	H	J	K	L	M	P	S	U	V	W	Y	Z	B	O	X

Figure 11-28: Monoalphabetic cipher example.

Polyalphabetic ciphers are ciphers where different substitution alphabets are used for different parts of a plaintext message. They are more complex than monoalphabetic ciphers because they use multiple monoalphabetic ciphers to encode plaintext letters; the monoalphabetic cipher used for a specific plaintext letter depends on its position in the message.

A monoalphabetic cipher uses a fixed substitution scheme for the entire message, while a polyalphabetic cipher does not. When the same plaintext letter is used in different positions in the message, it might be encoded differently. This means that an intruder who intercepts a ciphertext message using polyalphabetic encryption would have a greater decoding challenge than if a monoalphabetic cipher was used.

Caesar, monoalphabetic, and polyalphabetic ciphers are described to help you understand some of the fundamental cryptography and mathematical principles on which today's advanced encryption algorithms are based. They also are discussed to provide a basic understanding of how/why some encryption algorithms are more complex than others and are more challenging for interceptors to decipher.

Symmetric Encryption

As noted in section 11.9.2., two major types of encryption algorithms are used in today's enterprise networks: symmetric and asymmetric. Symmetric encryption algorithms use the same key (cipher) to encrypt and decrypt the plaintext message. As discussed in section 11.9.2, symmetric encryption can be divided into two types: block ciphers and stream ciphers.

Block ciphers, which break plaintext messages into fixed-size blocks before using an encryption key to convert them to ciphertext, are used in many cryptographic processes and enterprise network technologies. They are used just about everywhere in cybersecurity.

A stream cipher, on the other hand, encrypts and decrypts a data flow. Unlike block ciphers, stream ciphers encrypt data in long streams. Encryption key bits are added to plaintext bits to create the ciphertext stream.

Keystreams, the streams of encryption bits, are based on an encryption key and a *seed*, called a *nonce* (number used only once). A nonce is typically a randomly generated prime number that serves as the start of the keystream. Keystream bits are then added (using binary math) to plaintext bits, one at a time, to produce the ciphertext stream. The process is reversed at the other end to convert back to plaintext. This is analogous to a Caesar cipher that substitutes one letter for another, one letter at a time.

Stream algorithms are faster and more efficient than block ciphers because plaintext is encrypted one bit at a time rather than in entire blocks. They are also less mathematically complex. Examples of stream cipher usage and algorithms are identified in section 11.9.2.

Symmetric Key Management

One of the main challenges of symmetric cryptography is finding a secure way to share the cryptographic key that senders and receivers use. Several strategies have been proposed over the years to tackle this challenge, including key-agreement protocols, key encapsulation mechanisms, and out-of-band sharing procedures. Differences between these strategies are summarized in Table 11-17.

Various key-agreement protocols have been developed to enable senders and receivers to co-create the key that they will use to encrypt and decrypt data. They contrast with key encapsulation mechanisms, in which one of the entities in the communication exchange, the sender or receiver, generates a symmetric key and then uses public key encryption to share it with the other entity. Out-of-band procedures use non-network procedures or a different type of network to share keys;

Table 11-17: Common Characteristics of Symmetric Key Management Strategies.

	Brief Definition	Requires Asymmetric Cryptography?	Internet/Network Involvement?
Key-agreement protocol	A *key-agreement protocol* is a protocol that two or more parties use to agree on an encryption key. Symmetric cryptography requires the initial creation or exchange of a shared key in a manner that is private and ensures integrity.	No	Yes
Key encapsulation mechanism	Integrity-assured digitally signed keys (signed by CAs) may be part of key agreement processes. Hybrid systems use public key cryptography to exchange secret keys.	Yes	Yes
Out-of-band procedures	These are non-networked procedures to share a symmetric key.	No	No

this may include exchanging keys during a face-to-face meeting, using a courier service, or using phone calls.

Session Keys

Session keys are symmetric encryption keys. A *session key* is randomly generated to ensure the security of a single communications session between a sender or receiver. It is a temporary key that is only used once to secure communications between endpoints.

Asymmetric Encryption

Asymmetric encryption algorithms use different keys for encryption and decryption.

Asymmetric encryption uses one-way mathematical functions to encrypt plaintext and decrypt ciphertext. One-way functions produce results (answers) that are impossible to calculate in the reverse direction using the same numbers used to calculate the results.

As an analogy, in a one-way function A + B may equal C, but subtracting B from C (C – B) does not equal A. If A is plaintext, B is the encryption key, and C is ciphertext that results from applying B to A, subtracting B from C will not reproduce A. However, there is a corresponding decryption key D, which, when applied to the ciphertext C, will produce A, the plaintext. Hence, for a one-way function, A + B = C, but C + D = A.

Because one-way functions are used for encryption keys in asymmetric encryption, they have corresponding decryption keys that are mathematically related to them.

Asymmetric Key Management

Standard-setting organizations, including NIST, recognize the critical nature of asymmetric key management and have developed standards for key administrators. Regulations, including HIPAA, FIPS, and PCI DSS, also emphasize the need to maintain the security of cryptographic keys used to protect sensitive data.

Public key infrastructure (PKI) is an important part of asymmetric key management. It works through the implementation of two technologies: certificates and keys. Certificates, which confirm the identity of organizations or people online, are issued by a *certificate authority (CA)*.

Organizations and individuals who want to use a CA register with the CA and provide proof of identity during the registration process. Certificates may also be associated with devices for use in authenticating devices. The issuance of PKI certificates also involves a *registration authority (RA)*, which receives and processes requests for certificates and certificate renewals. RAs facilitate the issuance and renewal of certificates for organizations, devices, people, and applications.

After the registration process is complete, the CA creates the public and private keys for the organization, person, or device. The public key is added to the CA public registry, and the private key is sent to the registrant for safekeeping and use.

Basically, a PKI certificate grants permission to an entity to engage in PKI key exchanges. Like passports, they include a unique identifier for each of their holders and an official attestation from a trusted source (in this case, a CA). Without a PKI certificate, the entity is not permitted to participate in the exchange of PKI-encrypted data. The PKI certificate confirms the identity of the entity in the data exchange.

PKI certificates are stored in a certificate database on a server hosted by the CA. CA information and the certificate holder's private key are also kept on the local device or computer used for Internet communication.

When an entity wants to use PKI encryption to exchange messages with a certificate holder, it contacts the CA with a request for the certificate holder's public key. The request is encrypted using the CA's public key. The CA uses its private key to decrypt the request and then creates a response message to the request using the requestor's public key for encryption. The response is also "signed" by the CA using its private key. Hence, the requestor uses the CA's public key to verify that it is the CA that is responding to the request message.

Once the requestor has the certificate holder's public key, it can use it to encrypt a message to the certificate holder, knowing that the message can only be decrypted with the certificate holder's private key. Hence, the validity of a certificate can be verified through a system that checks whether it is authentic.

PKI certificates are not issued indefinitely. They must be periodically renewed. You have probably encountered warning messages from a CA indicating that an entity's certificate could not be validated. Such messages are frequently triggered by an entity's failure to complete the renewal process in a timely manner.

There are numerous common uses of PKI certificates. These include:

- HTTPS—In HTTPS, certificates identify each website the Internet user tries to reach. This helps to ensure that users are communicating with valid websites. Today's browsers routinely warn users when HTTPS is not used by the website to flag websites whose identities cannot be verified through PKI infrastructure. HTTPS and PKI certificates contribute to deterring man-in-the-middle (MITM) attacks by making it more difficult for attackers to decrypt and change/steal the contents of intercepted messages.
- SSL—As noted in previous chapters, SSL operates between the Application and Transport layers of the TCP/IP protocol stack. It encrypts outbound packets as they are passed from the Application layer to the Transport layer and decrypts inbound packets as they are passed from the Transport layer to the Application layer. In SSL, the client-server handshake involves PKI authentication. The server subsequently identifies its preferred symmetric key algorithm (e.g., AES) in a digitally signed message to the client, which the client can decrypt using the server's public key. The client then generates a shared key for the communication session and sends it to the server in a message encrypted using the server's public key. The shared key is then used to encrypt and decrypt the message(s) exchanged between the client and server.
- IPsec—As noted previously, IPsec can be implemented at the Network (Layer 3) or Data link (Layer 2) layer. When used at Layer 3, IPsec sits between TCP/UDP at the Transport layer and IP at the Network layer. Also noted previously, IPsec can support a variety of encryption techniques. Because of this, the first step in IPsec security is determining the encryption technique and encryption key that will be used. This is done using the Internet Key Exchange (IKE). IKE is a key management protocol

standard used to provide security for VPN negotiations; basically, it is a method for exchanging encryption keys for encryption over an unsecured medium, such as the Internet. In IKE, the encryption process (e.g., 3DES) is negotiated using PKI infrastructure. Also, each party generates a random key and sends it to the other using PKI infrastructure. The two keys are combined to produce the symmetric key used to encrypt subsequent message exchanges between the two entities.

- PGP—PGP (Pretty Good Privacy) is a freeware public key encryption product that is sometimes used to encrypt email. It enables users to post their public key on a web page so that other Internet users can send them encrypted messages.

Poor PKI execution poses security and communication risks. For example, if a CA is offline or not reachable, it may be impossible for Internet users to interact with certificate holders until availability is restored.

Unsecured digital identities pose serious security risks. Attackers/intruders may attempt to use an expired certificate to pretend to be a valid certificate holder. Also, a compromised CA can result in nefarious use of PKI certificates. Hence, it is important to effectively manage and secure PKI infrastructure. Key management best practices have been developed by NIST and federal regulators to help protect the integrity of PKI infrastructure.