

DCF255 Test 2 Review

Week 6: Routing and Switching

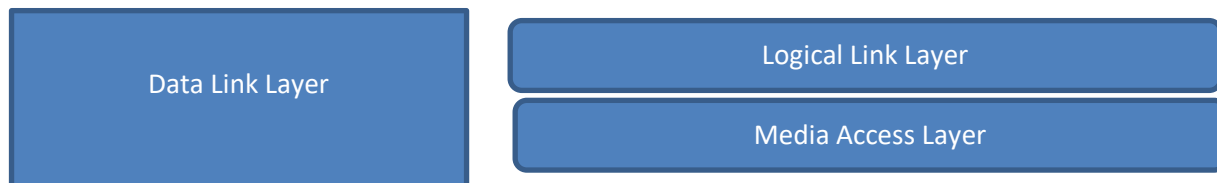
The two most common devices are switches and routers.

- Switches are layer 2 devices, data link layer, which means they forward frames, not packets, using the Media Access Control (MAC) address.
- Routers are layer 3 devices, Internet layer, which means they forward packets, not frames, using the IP address.

The basic device is the switch. Routing was built on top switching so that many independent single networks can be joined into a larger network in order to share resources and exchange data.

Data Link Layer

The data link layer is not a single layer, but 2 layers: Logical Link Layer and the Media Access Layer



- The Logical Link Layer is only used by 802.2 Ethernet (old Ethernet) and is also used by other LAN\WAN technologies. Modern Ethernet, is defined by the IEEE 802.3 standard and does not use the Logical Link Layer.
- All types of Ethernet have one thing in common- they all use the Media Access Layer to access the network. This layer uses a protocol called CSMA/CD to control how network PCs access the communications channel. CSMA/CD (Carrier Sense Multiple Access with Collision Detection).

How CSMA/CD Works?

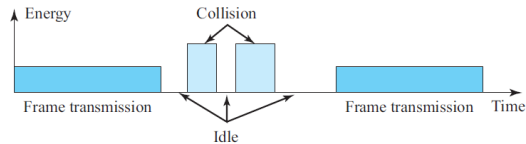
In CSMA/CD when a device wants to transmit data it must first access the transmission channel and determine whether the channel is free. If the channel is not free (noise level is high) it waits and checks again after a brief amount of time. If the channel is free, the node transmits data. Any PC can transmit its data. However, if two nodes transmit at the same time a collision will occur which will destroy the data. This is where the third component Collision Detection comes in. If a collision occurs (a spike in the noise level), both devices stop transmitting. The NIC cards of each device will send a special 32-bit sequence that indicates to the rest of the network, that its previous transmission was faulty and that those

data frames are to be deleted. Each node then waits a random amount of time and, if the network channel is free, it automatically retransmits the data.

CSMA/CA (Carrier sense multiple access with collision avoidance)

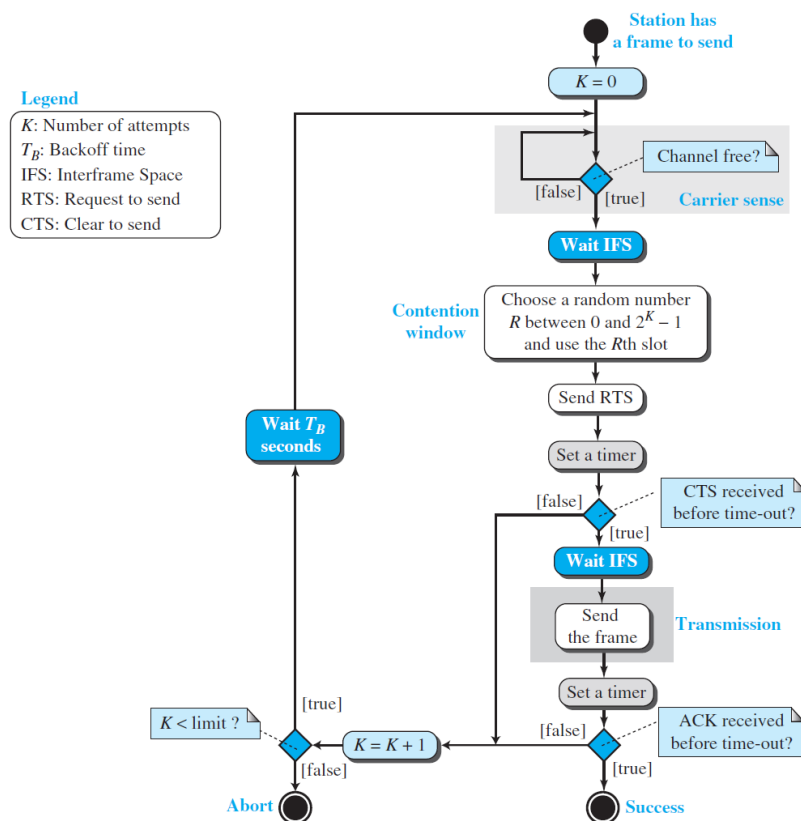
How collision can be detected:

- detecting voltage level on the line
- detecting power level
- detecting simultaneous transmission & reception



CSMA/CA

CSMA/CA uses three strategies: the **interframe space (IFS)**, **contention window**, and **acknowledgments**



Wireless Ethernet is a shared medium and the process use a protocol called CSMA/CA. It works like . It works like CSMA/CD, but has extra steps to avoid collisions). This is the CSMA protocol with collision avoidance.

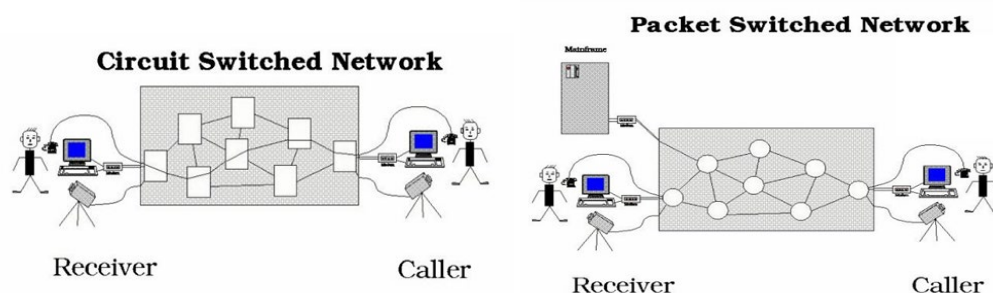
Algorithm:

- The station ready to transmit, senses the line by using one of the persistent strategies. As soon as it find the line to be idle, the station waits for an IFS (Interframe space) amount of time. It then waits for some random time (contention window slot time) and sends the frame.
- After sending the frame, it sets a timer and waits for the acknowledgement from the receiver. If the acknowledgement is received before expiry of the timer, then the transmission is successful. But if the transmitting station does not receive the expected acknowledgement before the timer expiry then it increments the back off parameter, waits for the back off time and re-senses the line.
- Whenever the channel is found idle, the station does not transmit immediately. It waits for a period of time called interframe space (IFS), because it may be possible that same distant station may have already started transmitting and the signal of that distant station. Therefore, the purpose of IFS time is to allow this transmitted signal to reach other stations.

Switching

Switching is a process to forward frames coming in one port and forwarded out the destination port which copies the MAC address of the device attached to it. It is much faster than routing. It is used to improve performance by minimizing broadcast and collision traffic which consume bandwidth. At a broad level, switching can be divided into two major categories:

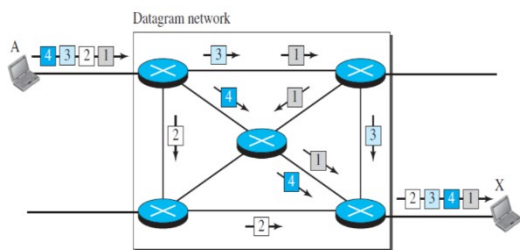
- **Circuit Switching:** In circuit switching a connection is established between two network nodes before they begin transmitting data. Bandwidth is dedicated to this connection and remains available until the users terminate the communication. Applications which use circuit switching may have to go through three phases: establish a circuit, transfer the data and disconnect the circuit. Example: PSTN, DSL.
- **Pros and Cons:** A dedicated circuit is good for voice communication it is not good for transmitting digital data. The dedicated circuit is too inefficient in bandwidth when a million bits can be sent in a fraction of a second.



- **Packet switching:** packet Switching is the most popular form of connection for Ethernet and the Internet. Large messages are fragmented into small individual messages.
 - **Pros and Cons:** Efficient use of bandwidth by not holding a connection open until a message reaches its destination. Packet switching also enhances line efficiency as packets from multiple applications can be multiplexed over the carrier. The Internet uses packet switching; packet switching enables the user to differentiate data streams based on priorities. Packets are stored and forwarded based on their priority to provide quality of service. Because of the time, it takes to reassemble the packets into a message, packet switching requires speedy connections, if used for live audio or video transmission.

Two types of packet switched networks

1. Datagram switching



Datagram switching (Network layer)

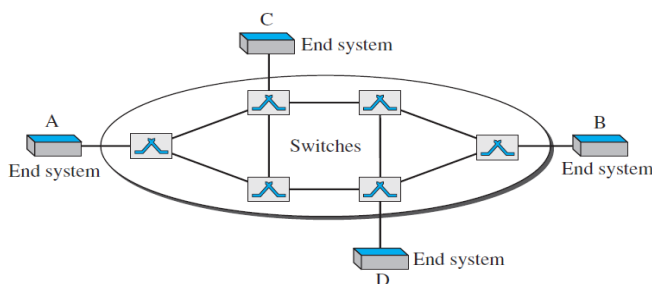
In datagram switching each packet is treated independently to all other packets. Depending on the current network situation (congestion or unavailability of the next hop) packets may take different routes/paths to the destination. Datagram switching is a connectionless approach where no connection establishment or tear down take place.

2. Virtual Circuits

Virtual circuit is the combination of circuit switched and packet switched techniques where data is packetized and for sending it to the destination a temporary path known as virtual circuit is established. All the packets followed the same path to the destination during the session. The header of the packet contains the address of the next switch should be and the channel on which the packet is being carried, not end-to-end address like in datagram switching where the destination address is the IP address of the destination node. There are three phases of communication:

Three phases of communication:

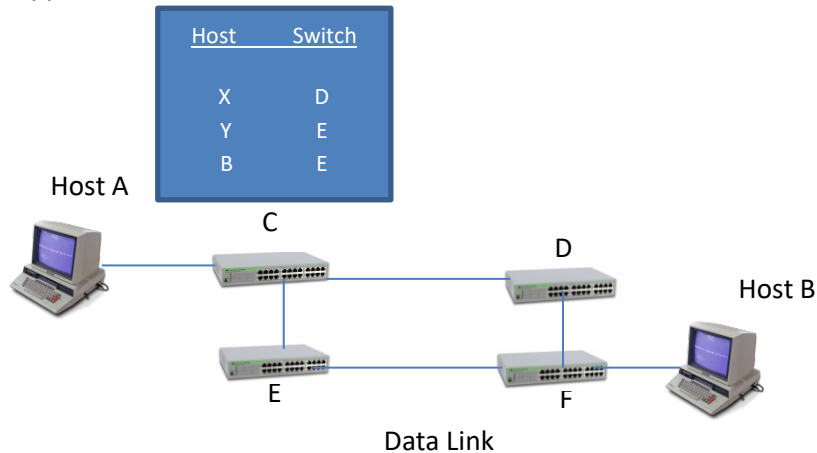
- Connection establishment
- Data transfer
- Connection tear down



In Figure. The network has switches that allow traffic from sources to destinations. A source or destination can be a computer, packet switch, bridge, or any other device that connects other networks.

Switch Operation

Suppose Host A wishes to send information to Host B. We can see there are 2 switched paths CDF and CEF.



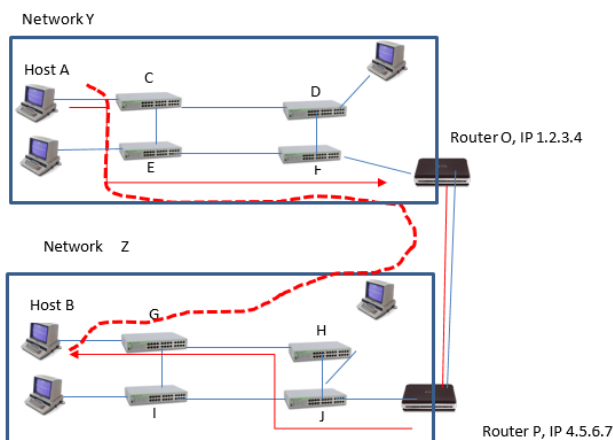
Switch C has a switching table with two columns: one for the destination of the frame, and the other for the switch to use to reach the destination host. This switch-by-switch decision making continues across each physical link until the final switch forwards the frame to Host B. Switches only know their neighbor; they do not know

the entire path. The path from Host A to CEF and final to Host B is called a data link. There is only 1 data link across a single network, regardless of the number of switches used.

Router

Bob Kahn and Vint Cerf devised a new device to link single networks together which they called a “gateway” when ARPANET was developed. Today, we call this device a router (the term gateway is still used as a point of access to a network).

Routers connect different signal networks together. Single networks have no idea what a router is, they only know how to deliver frames to their neighbour. They do not open the packet and look inside to see the message in the frame. Routers work by acting as hosts on the network. Like a PC, a router has a NIC and a unique IP address.



- Unlike switches, routers and PCs have software that can open the frame to see the packet and view the source and destination IP addresses.
- Packets are routed across internets. Frames are switched across single networks.

Type	Number	Description
Physical Links	9	AC,CE,EF,FO,OP,PJ,JH,HG,GB
Data Links	3	AO,OP,PB
Frames	3	AO,OP,PB
Packet	1	AB
Route	1	AB

Example

Host A on Network Y, wants to send an application message to Host B on Network Z. On Network Y, the router acts as a destination host, accepting the frame across Network Y data link. Router O creates a new frame acceptable to Network Z. Looking at its routing table, Router O, knows that to reach Network Z, it needs to send the frame to interface 4.5.6.7. Router P acts like a host accepting the frame from Router O and forwards the frame to switch J. Switch J uses its switching table to forward the frame to switch H which in turn forwards it to switch G and finally to Host B.

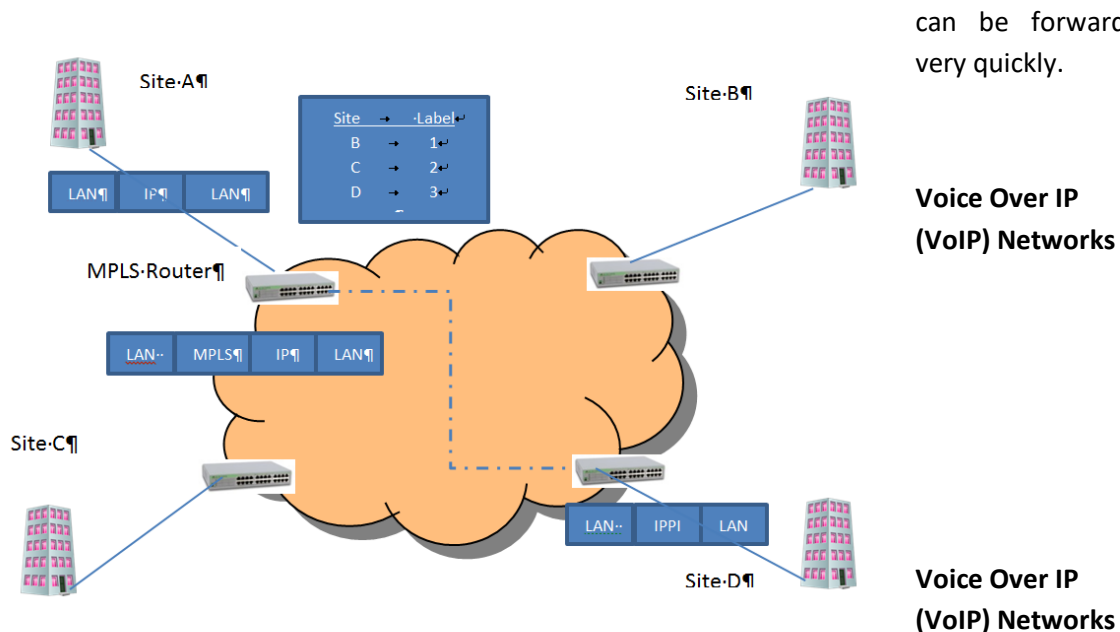
Routing Operation

Switches which are organized in a hierarchical fashion and only have one path to the destination host, a router is organized in a mesh with multiple paths to the destination host. The rows in the routing table are in the thousands with each row identifying a destination path. Thus, the router must make a forwarding decision as to the best route making routing much slower than switching. Routers have specific routing algorithms to help in the decision making, but all routers follow a 3-step process when a packet arrives. This is true for IPv4 and IPv6 routing:

- The router finds the destination IP address. It does this by ANDing the IP address with the network mask. For example, suppose a packet was destined to MySeneca on the Seneca College network, IPv4 142.204.250.120. The router would apply the mask of 255.255.0.0
- The router then compares the network address of 142.204 to every row in its routing table. The router then compares the network address of 142.204 to every row in its routing table. From the list of matches, the router must decide which route is the best-match. The rule the router follows is which route is the longest match (to get the packet as close to the destination as possible). For example, if the routing table showed 2 routes to MySeneca, 142.204.0.0/16 and 142.204.0.0/24, and the packet was destined for network 142.204.250.120; the router would choose the destination row with the 24 bit host.
- Some rows may have the same longest match than the router uses some metric to break the tie. The metric will depend on the routing protocol used. Usually, the tie breaker will be the shortest distance.

Multiple Protocol Label Switching (MPLS)

- Multiprotocol Label Switching (MPLS) is a new standard which greatly simplifies routing and basically allows the router to function like a switch. This technology can be used with any LAN technology, like Ethernet, but is currently being used by third parties to provide MAN and WAN configurations. MPLS operation is completely transparently to the sending and receiving host. MPLS adds a new header in between the existing LAN and IP headers (note: the LAN and IP headers are not changed). Can be used to avoid routing delays.
- On an MPLS network, when two hosts begin to communicate, they do not immediately send packets. Instead, they determine the best path for the packets. This best path is called the “label-switched path”. This dedicated path is slow to set up, but once created all subsequent packets can be forwarded very quickly.



VoIP is a client/server architecture which uses data devices to send real time audio communication as well as data over the TCP/IP Ethernet network. The servers set up the connection, and once made, the clients then send IP packets back and forth until the call has ended and the servers take down the connection.

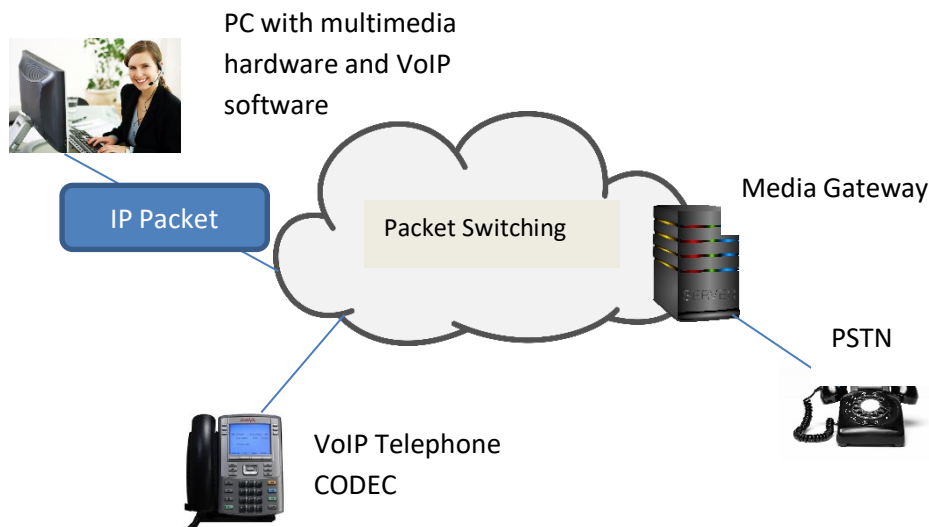
Clients

The clients can be a PC with speakers and microphone and VoIP software. A client could also be a specialized VoIP telephone, that uses an RJ-45 connector. These telephones have built in codecs to convert the digital IP packet to analog voice. A traditional land line can also be a client provided the VoIP communication goes through a media gateway which does the conversion.

Servers

There are two major VoIP signaling protocols, H.323 which is an ITU-T standard or SIP (Session Initiation Protocol) created by the IETF. SIP is the newer standard and will probably replace H.323 in time. All of the major telcos and cable companies offer VoIP services. Most businesses and home users are buying

the service from third party providers. But there is an open source version called OfficeSIP which works with Windows and can be downloaded onto any PC to act as a server.



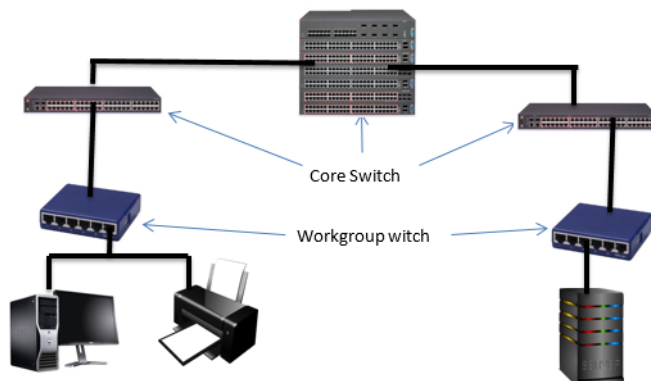
- To provide better voice communication, the UDP, User Datagram Protocol is used. This protocol is connectionless, does not guarantee delivery and reduces the processing load on VoIP phones. To improve UDP the Real Time Protocol (RTP) is used in conjunction. This protocol places a RTP header in between UDP header and the application message.
- VoIP is highly sensitive to jitter
- A common kit for Windows is Ozeki VoIP SIP SDK and a cross platform, open source SDK is Asterisk

Week 8: Wired and Wireless Ethernet Standards

Ethernet standards are controlled by the Institute for Electrical and Electronic Engineers (IEEE) and ISO. They are named IEEE 802.3 which is the most common local area network today with speeds of 100 Mbps speeds. Wireless Ethernet is controlled by the IEEE 802.11 standard.

Wired Ethernet: IEEE 802.3

An Ethernet network is built in a hierarchical manner using Core and Workgroup switches. Core switches



are used to connect switches to switches. Workgroup switches join devices to the network such as printers, workstations and servers. This Switched Ethernet diagram below, is effectively upside down;

- Key design principle is that you have enough bandwidth to handle the aggregate speed of all devices on the network

- 10 devices @ 100 Mbps connected to workgroup switch
- Workgroup switch must be able to carry 1000 Mbps total aggregate speed. Most likely cable is 1000 BASE SX
- 6 Workgroup switches @ 1000 Mbps means that the core switch must carry 6 Gbps of total capacity
- Best – 10G core switch allows for future growth and over-provisioning
- Creating a hierarchical switched network provides the optimal balance between cost and performance. It allows combining less expensive cabling and switches with expensive switches and cabling and using them only where higher performance is required.
- A key design principle of a hierarchical network is that the cabling and switches must be able to accommodate the aggregate of the workgroup switches.

Types of Ethernet LAN Technology

Gigabit Ethernet: IEEE 802.3z

Gigabit Ethernet (GigE) was developed to meet the needs of multimedia applications and Voice over IP (VoIP) where real time transmission is required. It runs over copper or fiber-optic at speeds 10 times faster than 100Base-T. GigE is currently used for Ethernet backbones to interconnect high performance switches and servers. GigE is optimized for full duplex operation.

10 Gigabit Ethernet: IEEE 802.3ae

10 Gigabit Ethernet is the fastest and most recent of the Ethernet standards with a transmission rate 10 X GigE. Based entirely on the use of optical fiber connections and it is used as a high-speed backbone for high volume transmissions.

Power over Ethernet (PoE): IEEE 802.3-2012

PoE provides both power and data transmission over a single cable. This solution is ideal for surveillance equipment, access points and IP telephones where running power would be difficult or expensive. PoE supports fast data rates up to 100 meters in cable length while delivering 25.5 Watts of power.

Metro Ethernet

Metro Ethernet is a Metropolitan area network (MAN) technology based on Ethernet standards. It is commonly used to connect subscribers to a larger service network or the Internet. Larger businesses can also use metropolitan-area Ethernet to connect their own offices to each other and greatly extend the concept of a LAN.

An Ethernet interface is much cheaper than any current MAN technology such as SONET (Synchronous Optical Network) or FDDI (Fiber Distributed Digital Interface) while providing similar bandwidth and speeds. Often metro Ethernet is combined with a IP/MPLS backbone which is used to connect to the

service provider's switches and routers; these MPLS-based deployments are costly, but highly reliable, very scalable and are typically used by large service providers

The Evolution of Ethernet Standards to Meet Higher Speeds				
Date	IEEE Std.	Name	Data Rate	Type of Cabling
1990	802.3i	10BASE-T	10 Mb/s	Category 3 cabling
1995	802.3u	100BASE-TX	100 Mb/s*	Category 5 cabling
1998	802.3z	1000BASE-SX	1 Gb/s	Multimode fiber
	802.3z	1000BASE-LX/EX		Single mode fiber
1999	802.3ab	1000BASE-T	1 Gb/s*	Category 5e or higher Category
2003	802.3ae	10GBASE-SR	10 Gb/s	Laser-Optimized MMF
	802.3ae	10GBASE-LR/ER		Single mode fiber
2006	802.3an	10GBASE-T	10 Gb/s*	Category 6A cabling
2015	802.3bq	40GBASE-T	40 Gb/s*	Category 8 (Class I & II) Cabling
2010	802.3ba	40GBASE-SR4/LR4	40 Gb/s	Laser-Optimized MMF or SMF
	802.3ba	100GBASE-SR10/LR4/ER4	100 Gb/s	Laser-Optimized MMF or SMF
2015	802.3bm	100GBASE-SR4	100 Gb/s	Laser-Optimized MMF
2016	SG	Under development	400 Gb/s	Laser-Optimized MMF or SMF
Note: *with auto negotiation				

Name	IEEE Standard	Data Rate	Media Type	Maximum Distance
Ethernet	802.3	10 Mbps	10Base-T	100 meters
Fast Ethernet/ 100Base-T	802.3u	100 Mbps	100Base-TX 100Base-FX	100 meters 2000 meters
Gigabit Ethernet/ GigE	802.3z	1000 Mbps	1000Base-T 1000Base-SX 1000Base-LX	100 meters 275/550 meters 550/5000 meters
10 Gigabit Ethernet	IEEE 802.3ae	10 Gbps	10GBase-SR 10GBase-LX4 10GBase-LR/ER 10GBase-SW/LW/EW	300 meters 300m MMF/ 10km SMF 10km/40km 300m/10km/40km

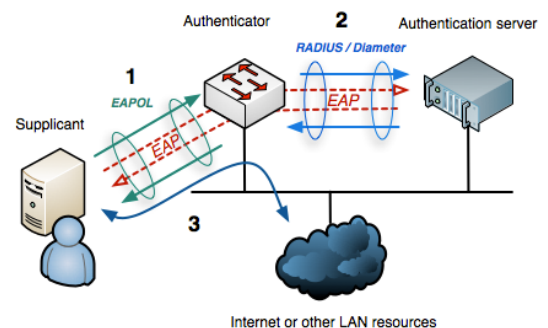
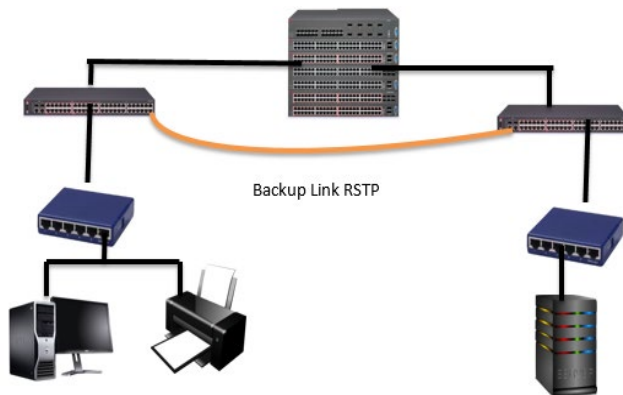
Types of Switches

1. **Store and forward:** A store and forward switch stores each incoming frame and checks for errors. If the frame is good, it forwards it to its destination port, if not it deletes it. With a store and forward switch bandwidth is not wasted on invalid or damaged frames. The disadvantage is that it increases the latency of the switch slightly.
2. **Cut through:** A Cut Through switch, on the other hand, does not error check and begins forwarding the frame immediately upon receiving the destination Address. This results in lower latency, but can propagate errors from one subnetwork to another, wasting bandwidth on invalid or damaged frames. For a Cut Through switch to work, the speed of the transmission coming into the switch must be the same speed leaving the switch
3. **Hybrid:** combining the two types. This switch monitors the frame error rate, and if it is below a level set by the administrator, the switch will function in Cut Through mode, otherwise it uses Store and Forward mode.

Ethernet Security

RTSP

- To avoid single point failures, the IEEE 802.1 Working Group on Ethernet have provided a way to create backup links using the Rapid Spanning Tree Protocol (RSTP). On a hierarchical network, there can only be one path to each device, loops are prohibited. To avoid single points of failure, the RSTP protocol can be used to create backup links.
- The core switch on the left, before it forwards a packet, is constantly polling if the central core switch is alive. If the central core switch is alive and working the backup RSTP link is NOT used. On the other hand, if the cores switch determines that the central core switch is down, then the RSTP backup link is used to send the frame



802.1x Port Security

- The 802.1x specification is designed to prevent unauthorized access to a data port.
- The 802.1x security protocol uses the Extensible Authentication Protocol (EAP) combined with a RADIUS (Remote Access Dial In User Service) server which stores user accounts and passwords. It divides each data port into 2 virtual switch ports; one for unauthenticated traffic and one for authenticated traffic. Until the user is authenticated all traffic passes through the unauthenticated port, and the only traffic allowed is EAP authentication traffic; all other traffic is dropped. In this sense the authenticator, the switch, acts like a security guard protecting the switch port from unauthorized use.

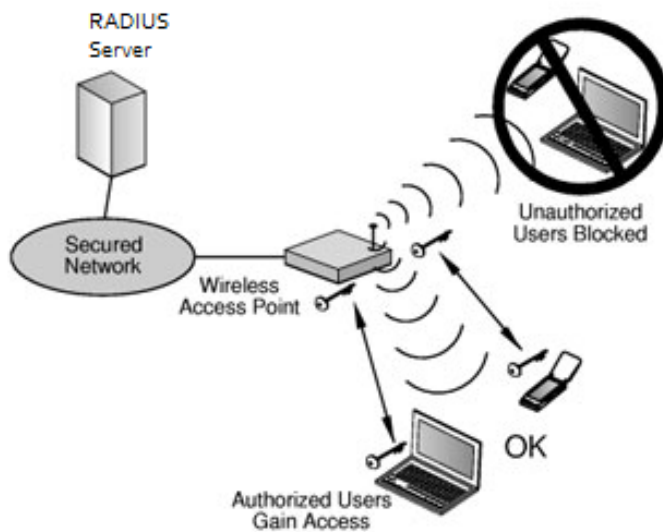
MAC Spoofing

- Another security concern with switched Ethernet is the ability of a malicious person to “spoof” a MAC address. When two devices are communicating, such as when you PING someone, the devices exchange MAC addresses and store this information in memory called the ARP Cache. On a single network the MAC address is used not the IP address to forward frames and there is no way to prevent MAC address spoofing on the client, since the ARP cache must be regularly updated. Thus, it is possible for someone to change the MAC address of a device and replace it with his/her device’s MAC address.
- **Solution:** Most enterprise switches have built-in intelligence to alert if someone is trying to spoof a MAC address. The switch will not allow the MAC address of the switch port to be changed without proper administrative authentication, or the switch may take proactive action and block the switch port alert the administrator that someone is trying to change an address, or there is a duplicate MAC address on the network.

Wireless Local Area Networks (WLANs) - 802.11

- Wireless LANs, or WLANs, use radio frequency technology to transmit and receive data over the air.
- The first WLAN specification was called WEP, (Wired Equivalent Privacy) this technology had some security problems
- WPA (Wi-Fi Protected Access) is backward compatible with WEP products, and uses the temporal key integrity protocol (TKIP) to ensure that keys have not been tampered with and scrambles the keys used for encrypted transmission during the session. WPA also provides user authentication with the extensible authentication protocol (EAP).
- WPA2 uses an encryption device that encrypts the network with a 256-bit key; the longer key length improves security over WEP. WPA2 was designed by the Wi-Fi Alliance standard organization and is basically the same as the IEEE 802.11i standard, with one exception. The WPA2 standard allows the sharing of keys when the network is formed.
- WPA2 works in 2 modes:
 1. **Pre-shared key mode (Personal Mode):** It is ideal for small or temporary networks. The key is only used for initial authentication; after initial authentication, the wireless access point gives each user a new key to access the Internet and randomly changes the key during the session. This prevents attackers, who may be listening to the traffic from cracking the key which was a major problem with WEP.
 2. **Enterprise mode (Infrastructure Mode):** works with IEEE802.11i to provide secure authentication and transmission

IEEE 802.11i is designed to work with enterprise switches using IEEE 802.1x. This provides better security than the shared key approach but requires an enterprise switch and a RADIUS authentication server.



Problem with WPA2

The Wi-Fi networks found in public locations, such as airports, hotels and coffee shops, are open and allow traffic to be sent over them that isn't routinely encrypted, and the traffic can sniffed or hacked during transmission

Solution WPA3

WPA3 fixes this by automatically encrypting all traffic between a device and the Wi-Fi access point by using a unique key, without the need for any prior setup by the user (Opportunistic Wireless Encryption (OWE) – RFC 8110), so even if someone sniffed the traffic but difficult decrypt it.

WPA3 works in 2 modes like WPA2 but with increased security

Personal Mode: Pre-shared key mode with Simultaneous Authentication of Equals (SAE) algorithm, that provides more protection to devices that do not have a strong password by preventing it to brute-force and dictionary password attacks

Enterprise Mode: offers an 192-bit minimum cryptographic strength with the combination protocols. A set of four cryptographic tools replace Wi-Fi 802.1x for WPA2-Enterprise, and the tools are combined together to provide better protection against attacks, such as password cracking on Wi-Fi networks.

Wireless Protocols

IEEE Standard	Year Adopted	Frequency	Max. Data Rate	Max. Range
802.11a	1999	5 GHz	54 Mbps	400 ft.
802.11b	1999	2.4 GHz	11 Mbps	450 ft.
802.11g	2003	2.4 GHz	54 Mbps	450 ft.
802.11n	2009	2.4/5 GHz	600 Mbps	825 ft.
802.11ac	2014	5 GHz	1 Gbps	1,000 ft.
802.11ac Wave 2	2015	5 GHz	3.47 Gbps	10 m.
802.11ad	2016	60 GHz	7 Gbps	30 ft.
802.11af	2014	2.4/5 GHz	26.7 Mbps – 568.9 Mbps (depending on channel)	1,000 m.
802.11ah	2016	2.4/5 GHz	347 Mbps	1,000 m.
802.11ax	2019 (expected)	2.4/5 GHz	10 Gbps	1,000 ft.
802.11ay	late 2019 (expected)	60 GHz	100 Gbps	300-500 m.
802.11az	2021 (expected)	60 GHz	Device tracking refresh rate 0.1-0.5 Hz	Accuracy <1m to <0.1m

Wireless

Transmission Errors

A negative side, wireless transmission as security and propagation problems. Setting up a wireless network is difficult and expensive. In addition, to the errors of crosstalk, EMI, jitter and noise, wireless transmissions have special problems.

- **Attenuation:** A signal strength attenuates as it travels down a medium. With radio waves the attenuation is much greater because the signal travels in all directions at the same time. In addition, plants are the “natural enemy” of radio waves and cause “absorptive attenuation” which greatly limits signal travel.
- **Multipath interference:** Radio waves can bounce off walls and ceiling, and other objects which creates two signals, the original signal and a reflected signal. Often the two signals arrive at the access point out of phase; one signal may arrive at its highest amplitude, and the reflected signal, with a slight delay, arrives at its lowest amplitude, this causes the signal to be unreadable and will require a retransmission.

Wireless Security

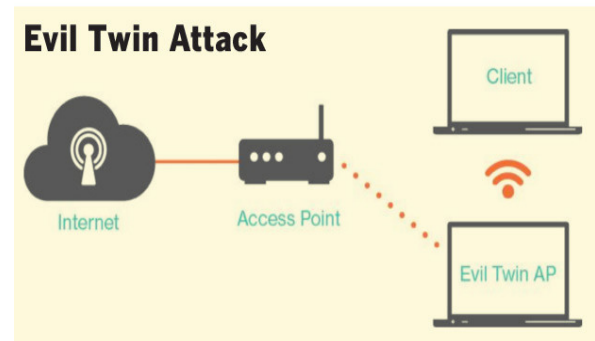
Wireless networks by nature are not as secure as wired networks. The two main threats from hackers are:

Rogue Access Points

A Rouge AP is a Wi-Fi access point that is installed on a network with the SSID as the network ID, but is not authorized by management. It can be used to steal data or create a Denial of Service (DoS) attack. Any client who connects to a rogue access point must be considered a rogue client because it is bypassing the authorized security protocols set by management.

Evil Twin

An evil twin is an access point that is operating at high power, usually in a public area, with the same SSID as the real access point. Wireless devices will connect automatically to the strongest signal; thus the wireless client is associated with an imposter network that is operated by a hacker. The evil twin will establish a secure encrypted connection to the wireless client. The hacker now has access to all communication between the client and the access point. In a public area, this technique can be used to steal personal information. In a corporate environment, it is used to steal encrypted keys, trade secrets, or launch DoS attacks.



VPN (virtual Private Network (VPNs

A VPN network can be used to defeat the evil twin and MITM attacks. A virtual private network (VPN) is a cryptographic connection between the client and a server. VPNs provide end-to-end protection, including authentication.

The network VPN server encrypts outbound network traffic, then wraps the encrypted message in an unencrypted IPv4 packet so it can be routed. If a hacker captures the packet he/she will be unable to read it. The destination VPN server, removes the unencrypted IPv4 packet and decrypts the message to forward onto the destination host.

How VPNs Work

Virtual private networks use two things to connect external hosts securely: **authorization and encryption.**

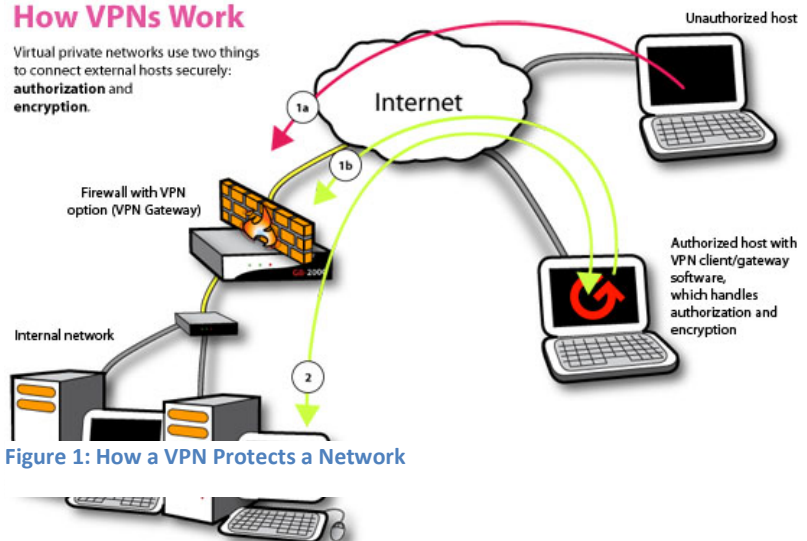


Figure 1: How a VPN Protects a Network

Week 9: Network Security and Management

The creation of new networks or the modification of an existing network is the every day work of networking professionals. Management will select the least expensive technology that will meet user needs. Networks are never static, they are continually changing to the needs and goals of the organization and personnel over time.

Cost and Decision Making

In networking, cost is a major constraint. User demand can never keep pace with corporate budgets. To aid in understanding how network changes can affect performance, management uses simulation tools like Riverbed Academic Modeler, which we are using in this class. There is never enough money to meet all user demands; management must make rational decisions and decide which product is best for the organization. To aid its decision making, management will use a Weighted Criteria Table.

		Product 1		Product 2	
Criteria	Weight	Rating	Points	Rating	Points
Price	5	10	50	8	40
Ease of Installation	4	7	28	5	20
Ease of Use	3	6	18	8	24
Total			96		84

Table 1: Weighted Criteria Table

In the table above, each criterion is given a weight from 1-5, 5 being the most important and 1 the least important factor to management. Then the products are rated by each criterion on a scale from 1-10. Once a product has been selected, it is managements responsibility to provide a multi-year budget to meet the network plan. Network projects have large capital outlays in some years and small outlays in others. The network administrator

Service Level Agreements (SLAs)

Users and consumers expect quality of service (QoS) today. The days of slow speeds, long delays and frequent power outages will no longer be tolerated. Consequently, when companies deal with service providers, such as cloud or web hosting services, they seek a written contract specifying the level of service expected. If the supplier is unable to meet the contractual minimum then a performance penalty will be paid to the company. Some firms are even beginning to require SLAs as a performance measure for internal network staff. The two most common SLAs are **speed and latency**.

In addition to maintaining the network infrastructure and SLAs, managements main responsibility today is security. Regardless, of the industry today, all companies are primarily concerned with “data

processing”. Protecting the collection, storage and retention of data on the network is necessary for management to make timely and relevant decisions.

Reasons for Network Security Failures

Reasons for Security Failures	
<ul style="list-style-type: none"> • Human Factors (love, greed, extortion, ignorance) • Poor Assumptions (overly trusting on data input) • Hardware\Software Misconfigurations (system\application software) • Poor Policy Guidelines-> when management sees security as a “technology” problem and not a “management” problem. 	
	<p>Note: A vulnerability is a known weakness in the software that could be taken advantage by malicious individuals</p>
	<p>An exploit is a tool designed to take advantage of a known vulnerability. A network break-in is also called an exploit.</p>

Security Management

With the increased use of the Internet, network security has become more important. Currently, network administrators often spend more effort protecting their networks than they spend on the actual setup. They must make the following determinations:

- Who will have access to data?
- What resources will users have access to?
- When will users access the resources?

These questions evolve around different levels of trust.

- Divide network into trust spheres
 - Administrators – most trusted (The most trusted are network resources in an organization such as internal servers, domain controllers, and storage devices)
 - Users, DMZ (A de-militarized zone is a separate subnet which contains only public servers)– less trusted
 - Internet users, unauthorized and remote users – untrusted

Plan Protect and Respond

Management of security involves three functions: Plan Protect and Respond. Management cannot protect a network unless it understands the “threat environment”, the types of attacks and attacker methodologies.

- Planning involves understanding the goals of the company and what assets need protection and developing a comprehensive security policy which includes all aspects of the company and network.

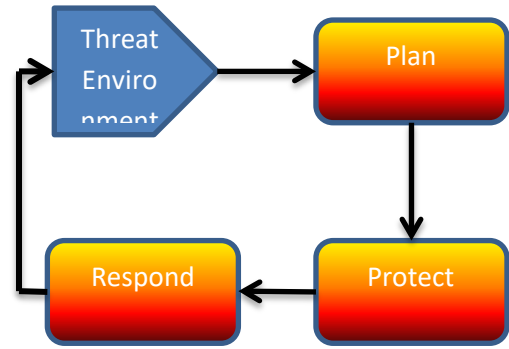
- Protection involves the ongoing protection of the network from firewalls, to account and access control; for highly sensitive data, it could involve cryptography as well. Even the best efforts to protect a network will result in breaches from time to time.
- Respond dictates how management handles the compromise, from detecting and stopping the attack to prosecuting the attacker. The threat environment feeds into the process and as it changes, management must make changes, completing the cycle.

Planning Protect Respond involves three interrelated factors:

- Planning: Risk Analysis
- Protect: Defense in Depth
- Respond: Comprehensive Security Policy

Planning: Risk Analysis

Risk analysis involves assessing the cost of an attack with the protection required. The goal is not to eliminate risk, but to lower it to manageable levels. Good management requires that if the cost of the counter measure is greater than the lost from an attack, then management should not implement the counter measure. Risk is the value of the assets lost X the probability of a loss.



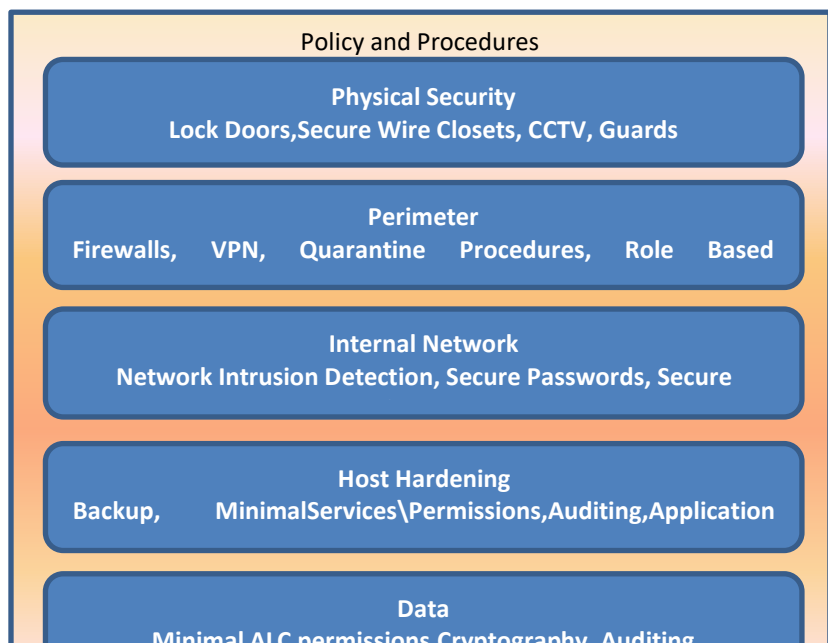
Description	Counter Measure A	Counter Measure B	Counter Measure C
Success Attack Damage	\$500,000	\$1,000,000	\$1,000,000
Probability of an Attack	20%	20%	20%
Annual Probability of Loss	\$100,000	\$200,000	\$200,000
Cost of Counter Measure	\$25,000	\$250,000	\$50,000
Net Counter Measure Value	\$75,000	-\$50,000	\$150,000
Implement Counter Measure	Yes	No	Yes

- **Risk = Value of the Asset X the probability of loss**
- Good Management is not to spend money on a counter measure if the cost is greater than the risk of loss

Protect: Defense in Depth

Defense in depth means that there are multiple layers of network security so that even if an attacker breaches one level there is another level protecting the network and may prevent the attack from succeeding.

Note: Rights refer to what you can and cannot do on the network, such as the right to log in or shut down a PC



Respond: Enforcement of Comprehensive Security Policy

A comprehensive security policy involves protecting the assets of the company by providing guidelines and standards for all employees to follow. An international standard for writing and implementing security policies is ISO 17799.

Hacking

- Hacking is when a person intentionally uses a computer resource without authorization or in excess of his/her authorization
- To protect a network you must understand the hacker motivation and methodology

Hacker Motive and Methodology

- Traditionally, hackers have been adolescents who break into networks or release malware. These individuals are driven by a desire for curiosity, power and peer “bragging” rights.
- The most dangerous group of hackers are “script-kiddies”; the latter are individuals who execute scripts written by others. They do not have a high level of technical knowledge, but their huge numbers make them very dangerous.

Types of Attacks

There are two types of attacks in general use today: random criminal attacks on individuals and targeted criminal attacks on corporations.

Criminal Attacks on Individuals

- To steal personal information, such as credit card or banking information. In this case, the goal is steal a victim’s credit card number so you can make purchases.
- Or, the goal could be to steal several pieces of a victim’s identity to impersonate the victim and get a loan from a financial institution.

In both cases, malware (evil software) is the tool used – viruses, worms or Trojan horses.

- Viruses are pieces of code that attach themselves to other programs. A virus executes when the infected program runs. Viruses can be spread by email attachments, links on a web site or file sharing using USB flash drives.
- Worms are stand-alone programs that do not need to attach to other programs because they are written in a macro language like VB. They propagate the same way as viruses. Viruses and worms have payloads, which can erase hard disks, or send users to pornographic sites, steal personal information, or download another program such as a keystroke logger. Sometimes the payload is hidden in a Trojan Horse. The latter is a container which has a legitimate looking purpose, but contains an illegitimate payload.

Criminal Attacks on Corporations

- Attacks on corporate networks are usually targeted attacks to break in and if that is not possible, initiate a denial of service attack. The attacker typically has a 3 step plan.

Stage 1

- Gather public information
- Web sites
- Whois database
- PING
- Social Engineering to steal information

Stage 2

- Research known vulnerabilities
- Attacks increase AFTER patch released – all information to design a tool
- Trial and Error to find an exploit that works – use Metasploit

Stage 3

- Create back door with known password and full privilege
- Delete log files
- Install “keylogger” to steal information or account information

DoS Attacks

If the attacker cannot break-in, he/she will initiate a denial of service attack (DoS). Or the goal is to disrupt the company operations, the hacker does not need to break in. He/she can disrupt company servers or switches\routers.

Week 10: Application Security**Top 10 Vulnerabilities of Web Applications**

Almost all the web vulnerabilities are a result of the programmer being overly trusting on how the

	A1-Injection	A2-Broker Authentication	A3-Sensitive Data Exposure	A4-XML External Entities (XXE)	A5-Broken Access Control	A6-Security Misconfiguration	A7-Cross-Site Script (XSS)	A8-Insecure Deserialization	A9-Using Components with Known Vulnerabilities	A10-Insufficient Logging and Monitoring
C1: Define Security Requirements	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
C2: Leverage Security Frameworks and Libraries	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
C3: Secure Database Access	✓	✓	✓	✓	✓	✓	✓			
C4: Encode and Escape Data	✓						✓			
C5: Validate All Inputs	✓						✓			
C6: Implement Digital Identity	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
C7: Enforce Access Controls		✓								
C8: Protect Data Everywhere			✓							
C9: Implement Security Logging and Monitoring	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
C10: Handle All Errors and Exceptions	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

application will be used. Building secure mobile applications is both an art and a science. It is an art in how the code base is written by the programmer, but it is also a science because we have learned from experience what procedures must be in place for software to be secure.

The top 10 application security vulnerabilities listed in ranked order of occurrence. The column across the top lists the vulnerabilities, and the row headings, lists the security practices which should have

been built into the application development process to prevent the problem.

C1: Define security requirements

Security requirements provide needed functionality that software needs to be satisfied and should be derived from industry standards, applicable laws, and a history of past vulnerabilities.

Instead of having a customized approach standard methods should be used, so they can be reused again in the future.

C2: Leverage Security Frameworks and Libraries

Secure code libraries and frameworks that have embedded security help software developers guard against security-related design and implementation flaws.

C3: Secure Database Access

Secure access to all data stores, including both relational databases and NoSQL databases.

1. Secure queries
2. Secure configuration
3. Secure authentication
4. Secure communication

C4: Encode and Escape Data

Encoding and escaping are defensive techniques meant to stop injection attacks. Encoding (commonly called "Output Encoding") involves translating special characters into some different but equivalent form that is no longer dangerous in the target interpreter. Escaping involves adding a special character before the character/string to avoid it being misinterpreted.

C5: Validate All Inputs

All data that can be entered or influenced by the user must be treated as untrusted. Before being used, including displaying it back to the user, the data must be checked to ensure that it is in the right length (syntactically correct) and in the right format (semantically correct) (and in that order).

C6: Implement Digital Identity

Digital Identity is the way to represent the online transaction. Different levels of authentication should be implemented that may include Password, multifactor authentication and cryptographic authentication.

Every time a user needs to make an important action, such transferring money, or changing the shipping address, he/she should be required to re-authenticate. The server should generate a new session token which should never be written to the local machine.

C7: Enforce Access Controls

Access controls refer to the authorization of a user to access a resource. User or system access should be based on the principle of "least privilege" - granting the least amount of access to do the job for the least amount of time. Application design should check each user's ability to access a resource and the access control policy and the application code should be separated into different layers.

C8: Protect Data Everywhere

It's critical to classify data in your system and determine which level of sensitivity each piece of data belongs to. Each data category can then be mapped to protection rules necessary for each level of sensitivity. For example, public marketing information that is not sensitive may be categorized as public data which is ok to place on the public website. Credit card numbers may be classified as private user data which may need to be encrypted while stored or in transit.

C9: Implement Security Logging and Monitoring

Design your application to log all important application events in order audit activity and conduct compliance monitoring. Logging is essential for forensic analysis and intrusion detection and helps ensure that controls are aligned with real world attacks. Like user input, logging input needs to be checked and encoded to prevent "log injection" attacks prior to writing to the log file.

Mobile applications are at particular risk of data leakage because mobile devices are regularly lost or stolen yet contain sensitive data.

As a general rule, only the minimum data required should be stored on the mobile device.

C10: Handle All Errors and Exceptions

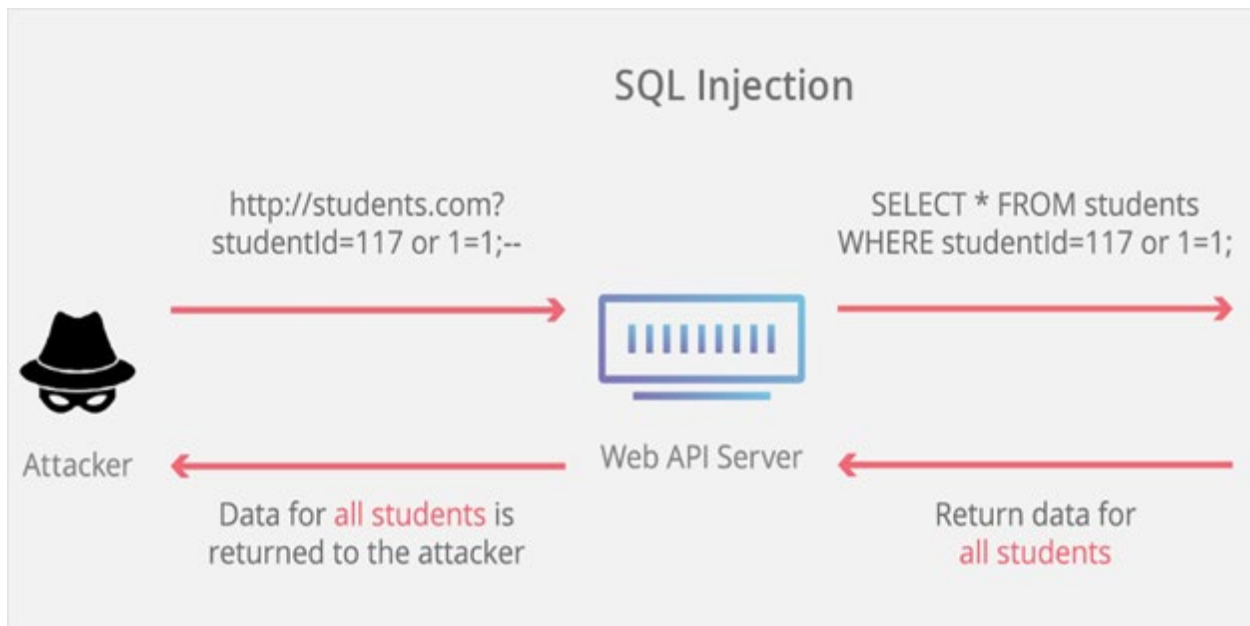
Research at the University of Toronto has shown that lack of error handling or minor mistakes in handling errors, can lead to catastrophic consequences in distributed systems. Error handling should be done in a centralized fashion and error messages to the user should not “leak” critical information about how the application works. All exceptions should be logged for forensic analysis.

Vulnerabilities Examples

Injection Attacks

1. SQL injection Attack

SQL injection is a web security vulnerability that allows an attacker to interfere with the queries that an application makes to its database. let an attacker to view data that they are not normally able to retrieve. With this attack, an attacker can modify or delete this data, causing persistent changes to the application's content or behavior.



String query = "SELECT * FROM accounts WHERE custID='" + request.getParameter("id") + "'";

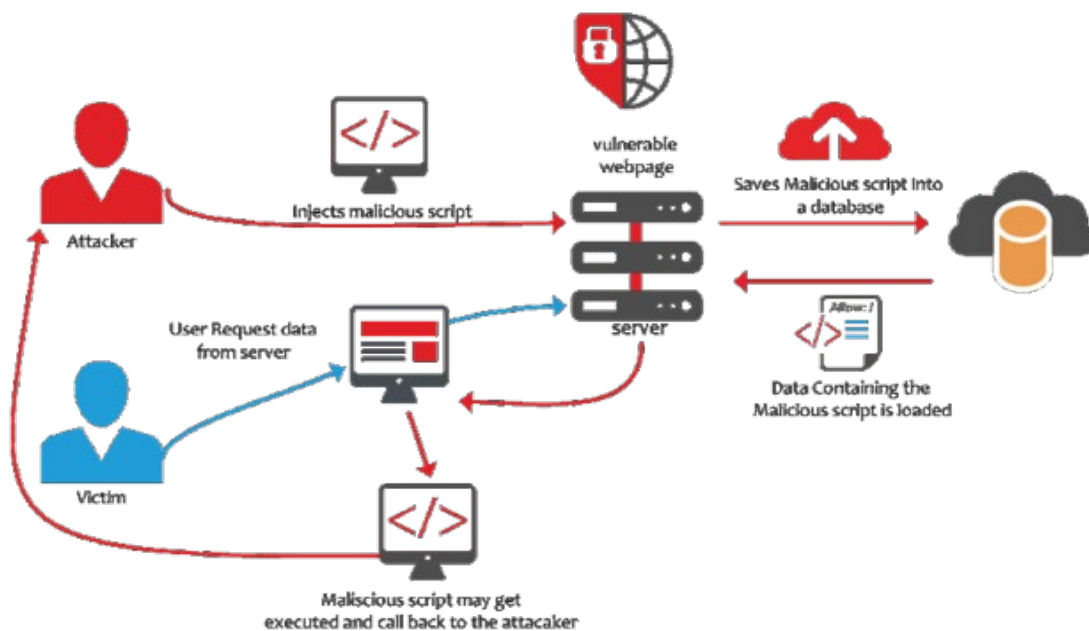
This query can be exploited by modifying the “id” parameter value as follow:

`http://example.com/app/accountView?id=' or '1'='1`

This makes a request to the application to return all records from the account table, other similar and more severe injections can modify the data, and even cause a loss of data.

2. Cross-site Scripting (XSS) Attack

XSS attacks are essentially malicious injections (client-side) that are added to a web page or app through user comments, form submissions, and so on.



3. Sensitive Data Exposure

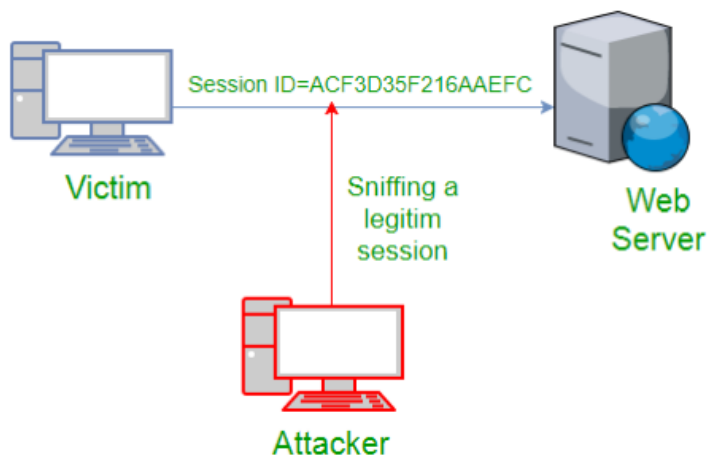
Sensitive data exposure has been one of the most popular vulnerabilities to exploit. It consists of an attacker compromising data that should have been protected. It differs from a data breach.

Data breach-> hacker attacks a network and gets control of the database

Data exposure->insecure transmission of data (data being transmitted in its original form), how data should be store (plain text or encrypted form)



4. Broken Authentication

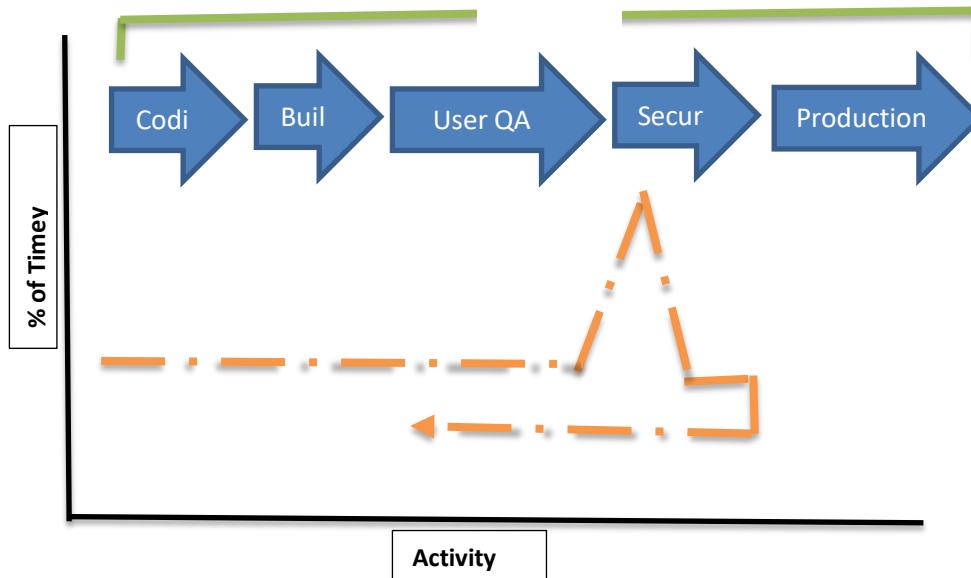


Broken authentication

vulnerabilities allow attackers to use manual or automatic ways to gain control over any account in a system and even gain total control. It allows an attacker to either capture or bypass the authentication methods that are used by a web application. Broken authentication permits automated attacks such as credential stuffing, where the attacker has a list of valid usernames and passwords.

Typical SDLC Process: Resulting in Insecure Software

The systems development life cycle (SDLC), also referred to as the application development life-cycle, is a term used in systems engineering, information systems and software engineering to describe a process for planning, creating, testing, and deploying an information system.



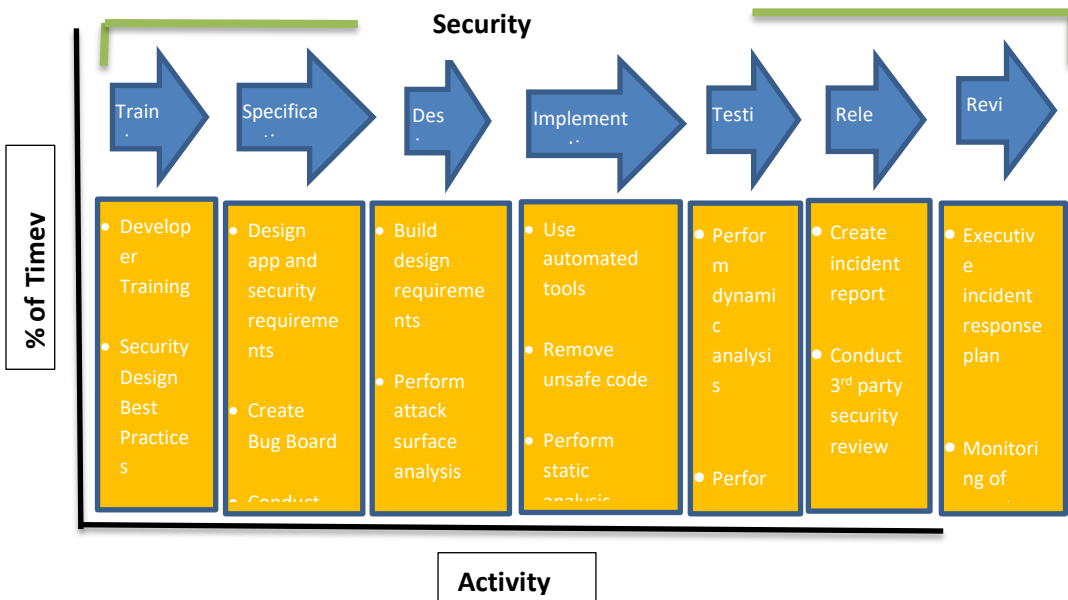
A New Paradigm: A Software Development Framework for Building Secure Software

- For secure software to be developed, attitudes must change. Software vendors should be treated like other businesses and be sued for faulty software.
- Developers should create a professional association and develop a code of ethics where security engineering is a performance criteria.
- Management should see secure software as an investment in building trust with its customers.
- Consumers should place secure software ahead of new features when they purchase software.

Security must be built into the SDLC process from the very beginning and tested throughout the process.

Training

The process begins with developers who are trained on building secure applications. Developers cannot build secure applications, or do proper penetration testing, if they do not understand the tools and methodology of the hacker.



Specifications

It refers to working with the users of the application to design an application which meets organizational and user goals. Notice that the application's security requirements are designed at the same time. During the design stage, developers should also research how the design is affected by governmental regulations and privacy considerations. This is also a good time to develop a "bug board". This is a white board, centrally located, where all "bugs" that are identified in the coding are written down, the date, who identified it, who replicated it, and the date the bug was corrected.

Design

The design phase is the coding phase, where the coding is done to meet the requirements. Unlike the previous diagram, which allowed the entire code base to be built, prior to security testing, the security testing is done concurrently as the code is built.

Testing

In the testing phase, dynamic code analysis is conducted in which the code is executed on a real or virtual processor. The application is first checked to see it performs to specifications. Then the code is reviewed for security faults. The programmer must enter input like a user to test different scenarios: What if the input is not in the length expected, what if the input is not in the format expected, what if the hard drive is full and the input cannot be written to a file. A popular tool is VeraCode which uses the same techniques a hacker uses to find vulnerabilities and design weaknesses.

Release

The first step of the release phase is when management signs off on the project - after the application has been thoroughly tested for performance and security. Typically, management will also, forward the project to a 3rd party security firm for independent review. Once the application is released, bugs will be discovered due to specific configuration issues on the client system which the developers could not anticipate. Thus, it is important to create an incident board to log all the issues, to study the root cause of the problem and develop workarounds and patches to fix them.

Review

Lastly, the development process never ends. The review phase is about ongoing monitoring of the application and testing of new attack surfaces, as they develop. As the product matures, patches and upgrades will be tested and developed. The plan outlines the procedures for handling security related issues from detection to remediation. All this work is done under the framework of a policy statement called the Incident Response Plan.

10 Best Practices for Building Secure Mobile Applications.

1. **Always Assign Security to a Champion** - security is best applied in a top-down fashion and needs a "champion" to get things done, build relationships with stakeholders, get funding, build a security "culture" and awareness of key projects. (Manager/ CISO (Corporate Information Security Officer))
2. **Always Model Threats During Software Development**- Security requirements for a project must be developed at the onset of the project when the design specifications are drafted. Security requirements for a project must be developed at the onset of the project when the design specifications are drafted.
3. **Always use Modular Code to Separate Parts of the Application**- Dividing your code into modules, submodules and internal APIs creates segregation in your code which isolates parts of the application.
4. **Always use Tested Code Libraries**- Only use encryption libraries that have been available for many years. The longevity of the library speaks for itself. Good libraries have good documentation.
5. **Always Test Application on Different Mobile Platforms and APIs**- Testing for security to ensure that each platform maintains the confidentiality, integrity and availability of the data. Mobile applications are written for IOS, Android, Windows, and other mobile platforms. Each platform uses programming interfaces (APIs) and must be independently tested for performance and security.
6. **Always Add Multiple Levels of Authentication**- Use Two-factor refers to an authentication system in which users are required to authenticate using at least two different "factors" something you know, password/ pin, combined with something they have, smart phone or computer, before being granted access. As a programmer, you must strike the balance between ease of sign up/login with smart security measures which protect privacy and confidentiality.
7. **Always Require Minimal Data from the User**- When designing your application, focus on requesting the least amount of data possible from users; this limits the amount of data you must protect. As a programmer, ask yourself what information do you need for the software to function. Once you have made that determination, design your application to hold a signal "check point".
8. **Always Use Automated Tools and Auditors for Testing**- Application developers should include static and dynamic automated tools to help the code review. These tools attempt to automatically identify security flaws in the code. Once the app is written, it should be sent to independent penetration testers. Always Use Automated Tools and Auditors for Testing
9. **Always Build User Trust** - To encourage users to download your app display your privacy policy and what you will do if a data breach occurs. Be clear and specific. Users look for how many downloads an app has had as indicator of trust (i.e. 250,000 downloads). They also look to user reviews of your app and court the "star" rating. (i.e. 4 stars indicates favourable reviews)

- 10. Research and Stay Updated on Latest Exploits, Vulnerabilities, and Security Trends-** The threat environment is always changing and to a large extent, you will feel that you are playing “catch-up” to the hackers. In order to stay on top of the latest threats and how they may impact your software, you need to maintain regular subscription service. The three most important are: CERT, CVE and OWASP.

Week 11: Living in a connected world

The Internet of Things pertains to the concept of devices connected to the Internet where data gathered by such devices are reported to users. People can then act on the said data or the devices themselves are empowered to act on it. The ease of data transmission, reception and implementation are all meant to improve people's quality of life.

Electronic Communication: The Bad and The Ugly

Electronic communication has proven an effective vehicle for cybercriminals to distribute spam and malware. Malware hiding in email attachments, can install spyware or ransomware. Cybercriminals can distribute malware through drive by downloads. This technique is very popular with high volume online shopping websites.

To protect yourself always do the following:

- Install and keep up to date antivirus and anti-phishing software on your personal PC.
- Don't open any email attachment unless you checked the source, even if the email is sent from one of your contacts.
- Never comparison shop using a search engine.
- Never buy goods from an unknown retailer, regardless of how good the deal seems. Think before you click.

Examples of IoT: Smart Home, Robots and Driverless cars

Components of IoT



Sensors

Sensors are low powered devices that have a capability to monitor some physical or environmental phenomena and collect that information for further processing. Sensor nodes have limited battery power, but now there are some sensors available that can recharge themselves. They consume very little battery power for their operation, so even if they cannot recharge the battery, they still can live on batteries for a long time.

Local processing and storage devices

Each sensing node is equipped with low power processor and a limited amount of memory. The sensed data is briefly stored in the sensing node before transmitting it to data collecting device. Some analog to digital data conversion and data aggregation operations are performed by the small power processor.

Network and Internet

The hardware is needed which connects sensor nodes to the internet for the transmission of sensed and processed data to the cloud to be stored. The protocols that are used for this communication are: HTTP, XMPP which functions as a chat, CoAP, MQTT (less secure and designed for machine to machine communication)

Cloud

The last phase of operation is done at the cloud. At this level, the received data is processed according to nature of data as well as the type of application for which it has been collected.

3 Stumbling Blocks Preventing IoT

1. Lack of Overarching Standards

The main stumbling block for IoT devices is the lack of connectivity standards. There are several proprietary networks, such as X10 and HomePNA. Presently, only products from the same vendor can be networked – similar to network operating systems in the early 80's. The new industry standard organization, AllSeen Alliance, was launched in December 2013, to improve interoperability.



The AllSeen Alliance proposes to unite industry leaders with a shared vision, a common language and a collaborative path to advance the Internet of Everything. The AllSeen Alliance proposes devices be AllJoyn certified. The latter is an open source universal software framework and core set of system services that enable interoperability among connected products. It lets compatible devices and applications find each other, communicate and collaborate across the boundaries of product category, platform, brand, and connection type. Target devices include products in the Smart Home, but will later expand to Broadband Gateways and Driver-less cars. Presently, the communication layer (and thus hardware requirements) is limited to wi-fi.

2. Lack of Internet Bandwidth

The other disadvantage is IoT devices need a lot of data and information about the environment in which they operate. This data can include personal schedules, shopping habits, medicine intake schedule and GPS location of the user at any given time. Each device only uses a small amount of bandwidth, but multiple this by an estimated 30 billion devices and the Internet bandwidth must greatly increase for IoT to become a reality.

3. IoT devices are Insecure

This is not an idle problem. According to a recent study, [70% of IoT devices are vulnerable to cyber-attacks](#). This list includes thermostats, TVs, webcams, sprinkler control systems, home alarms, and door locks- just to name a few.

IoT and Programming Languages

Internet of Things (IoT) development projects are springing up at businesses all over the world. Choosing which language to use to write the project is as big a decision as which hardware platform to use. New languages and platforms are making it easier to engineer IoT projects than ever before. Prior to the IoT, your choice of hardware platform dictated your choice of language. However, with the AllSeen Alliance open source project modern platforms can support multiple languages, increasing developer flexibility. Here are the top 7 languages that you should consider:

C and C++

C was first developed to program telephone switches and it is still a contender for IoT projects. It's available on nearly every advanced embedded system platform and requires little processing power. The language is ideal for programmers who write for the lowest layer of software, the one closest to the hardware. The language hides nothing from you, and that means you can fiddle with every part of the code to squeeze out the best performance from an underpowered device. Every bit can be flipped. Every value on the stack is available. C++ is an alternative if the IoT device requires more complex tasks, think

thermostats and smart toasters rather than devices that detect moisture or heat. C++ adds data abstraction, classes, and objects. All of these features make C++ a popular choice for those who are writing embedded and IoT code with an interface. This programming language still is going strong after more than 30 years in the field.

Java

The motto of Java "write once, run anywhere" makes it an ideal choice for an IoT project. Also, the Java compiler has very few hardware dependencies built into it. Developers can create and debug code on their desktop and then move it to any chip with a Java Virtual Machine. That means the code can run not just on places where JVMs are common (servers and smartphones), but also on the smallest machines. Today, most of the focus is on Java SE Embedded, which is much closer in capability to the Standard Edition. Developers can use the latest features of the Java 8 platform and then move their code to a smaller, embedded device.

Python

Python started as a scripting language to glue together real code, but has become one of the "go-to" language in Web development, and its use has spread to the embedded control and IoT world. The syntax is clean and simple, which greatly improves readability. If the project requires taking data and putting it into any sort of database format, then draw upon the tables for control information, Python is a very real contender provided the device has the processing power for the application. For very small devices there is MicroPython and a software package for very small microcontrollers optimized to run Python on a small board that's only a few square inches. Python is very flexible in many ways. For example, it is an interpreted language that can either be submitted to a run-time compiler or run through one of several pre-compilers so that compact executable code may be distributed.

JavaScript

JavaScript is not an interpreted version of Java. It started as a scripting language, but has grown into a very full-featured language. The two languages were developed separately (JavaScript was developed by Netscape) and shares no syntax or semantics (however, there are libraries which allow Java and JavaScript to work together). JavaScript is heavily used for building Web-front-end applications. Forty-two percent of server based web applications use JavaScript. If you wanted to use the Apache server on a Raspberry Pi to gather data from a network of Arduino-based sensors, for example, JavaScript would be a good starting point for the effort. It's not for lightweight embedded controllers because its interpreted structure requires more processing power, but it works well with RaspberryPi.

Swift

Swift is an Apple programming language, replacing Objective C. The fact that many IoT devices will need to interface with iPhone or iPad makes Swift a good choice for an IoT project. There are other good reasons to use this language, Apple wants to make its iOS devices the center of the smart home network of sensors, so it's been creating libraries and infrastructure that handle much of the work. These libraries are the foundation of its "[HomeKit](#)" platform, which provides support for integrating the data feeds from a network of compatible devices.

B#

B# was designed from the ground-up as a very small, very efficient embedded control language. The embedded virtual machine (EVM) that allows B# to run on a variety of different platforms only takes 24k of memory -- much less than the overhead of other development languages. B# looks like C# (which will

be familiar if you or your team is accustomed to working on Microsoft .NET projects), but it strips out many of the features not required for embedded projects and adds support for the real-time control functions that are critical when making things happen in the real world.

C#

C# is a good choice for an IoT project. Microsoft's strategy for IoT devices is to link them to the Azure cloud. Data collected from an almost limitless number of internet-connected sensors needs to be collated, analyzed and acted upon, and a public cloud is the logical route to do this. Presently, car companies such as Corus, are using the cloud to provide mapping and integration to customers.

Microsoft has launched the Azure IoT Suite to aid developers in application development. The Suite acts as a bridge between customers' devices and the back-end application for storing, analyzing and acting on IoT data in real time. The Suite is scaled to handle billions of devices. The Suite supports multiple protocols and languages including C, Python, Java and JavaScript. Microsoft is positioning itself to use the cloud as an interface for end-to-end solutions of IoT devices from multiple vendors. This makes C# and Visual Studio a real contender for IoT projects.