Computer Security

Chapter 1 Fundamental Concepts

1.1 Confidentiality, Integrity, and Availability

C.I.A

- **Confidentiality**
  - Avoidance of the unauthorised disclosure of information
    - Protection of data
    - Providing access for those who are allowed to see it
  - *Encryption:*
    - transformation of info using an encryption key, so the encrypted data can only be read using the decryption key.
    - secure -> encryption key difficult so someone cannot read cipher text with decryption
  - *Access control:*
    - policies that limit access to confidential information to systems with a 'need to know'
    - done by identity, name ,serial number, role of person
  - *Authentication:*
    - the determination of the identity or role that someone has
    - by a key fob, smart card, password
    - something you are, have or know
  - *Authorization:*
    - the determination if a person or system is allowed to access resources, based on access control policy
  - *Physical Security:*
    - physical barriers to limit access to protected computational resources
    - locks on cabinets and doors
- **Integrity**
  - Property that information has not be altered in an unauthorised way
  - *Backups:*
    - the periodic archiving of data
    - data files can be restored if altered in an unintended way
  - *Checksums:*
    - computation of a function that maps the contents of a file to a numerical value

            - depends on contents of file, a tiny change can result in a different value
            - used to detect a breach of data integrity
        - *Data correcting codes:*
            - method for storing data in a way that small changes can be easily detected and automatically corrected.
            - applied to small units of storage
        - All tools use redundancy -> replication of some content or functions
        - Need to protect metadata -> attributes of the file & info about access which aren't part of the content
            - time stamps, groups accessed, users
    - **Availability**
        - Property that info is accessible and modifiable in a fashion by those authorised to do so
        - *Physical Protections:*
            - infrastructure meant to keep info available even in the event of physical challenge
            - building housing, systems to withstand storms, earthquakes & bomb blasts
        - *Computational redundancies:*
            - computer and storage devices that serve as fallbacks in case of failures
            - redundant arrays of inexpensive disks (RAID) use storage redundancies to keep data available to clients

Chapter 5 Network Security

1 Network Security Concepts

1.1 Network Topology

- data packet is a finite-length set of bits -> divided into a
    - header: specifics where the packer is going and contains various overhead/bookkeeping details
    - payload: actual information that is being communicated
- network connection structure -> network topology
- computers in a network are **host nodes** that can be sources and destinations of messages
- routers are communication nodes -> messages flow
- physical connection between nodes define channels where messages travel so that packets move by being passed from one to another to get source -> destination
- private network of computers in close zone -> *local area network (LAN)*
    - internet -> *wide area network (WAN)*
        - Autonomous Systems(ASs) -> wide area networks on the internet -> clusters
            - controlled by single organisation entity -> how packets are routed among nodes
            - using shortest paths so distance is minimised and routing cycles are avoided
            - between ASs -> by contractual agreements, avoid loops

1.2 Internet Protocol Layers

- ***Internet protocol Stack***
  - layers provides a set of services & functionality guarantees for higher layers
  - interface each layer provides to higher levels -> provide essential info, lower level details are hidden
- ***Physical Layer:***
  - move the actual bits between nodes of the network
  - physical wires (abstraction it provides to the higher level is the ability)
- ***Link layer:***
  - transfer data between a pair of network nodes | between nodes in a local-area network and to detect errors that occur at the physical layer
  - deals with logic in sending info & how to find good routing paths in a local-area network.
  - Protocols such as Ethernet -> route packets between computers sharing a common connection
  - The link layer provides a grouping of bits into ordered records, called frames -> 48 bit address called *media access control addresses (MAC address)*
- ***Network layer:***
  - known as the internet layer
  - provide for moving of packets between any two hosts(best effort basis)
  - individually addressing each host using a *IP address*
  - Main protocol is (IP) -> IPv4(32 bit IP addresses), IPv6(128 bit IP address)
  - *Best Effort Basis* no guarantees that any packet will be delivered, if reliable delivery is needed -> upper layer requirement
- ***Transport Layer:***
  - support communication and connections between applications based on IP address and ports(16 bit address) for application level protocols
  - *Transmission Control Protocol(TCP)*
    - virtual connection between a client and a server and guarantees delivery of all packets in an ordered fashion
  - *User Datagram Protocol(UDP)*
    - assumes no setup and delivers packets as quickly as possible but with no delivery guarantees
- ***Application Layer:***
  - protocols that support useful functions on the internet, based on the transport layer services.
  - HTTP -> uses TCP & supports web browsing
  - DNS -> UDP & supports use of useful names for hosts instead of IP address
  - SMTP & IMAP -> TCP & support electronic mail
  - SSL -> TCP & supports encrypted connections
  - VoIP -> UDP & supports Internet telephone messaging
- ***Open Systems Interconnection (OSI)***
  - seven layers

    - Application layer ->
        - Strict application layer -> host applications-to-network processes
        - presentation layer for inter-host communication
- Packet for a layer -> data transmitted + metadata providing routing & control info
    - metadata stored -> header & sometimes footer
    - data portion -> payload
    - payload storing a packet is known as -> *encapsulation*
- ***Internet Protocol Stack***
    - functions and abstractions -> internet
    - layered model uses Internet Protocol Suite -> build software which uses approx services & provides right service guarantees without ruining implementation details

1.3 Network Security Issues

- ***Confidentiality***
    - encryption could be done at application layer (https protocols & IP sec specs)
- ***Integrity***
    - checksums to validate a small number of bits not encryption -> not cryptographically secure.
    - should be done at the application layers | alternative protocols at lower Layers
- ***Availability***
    - become unavailable -> lots of data requests,
    - need to scale with ↑ communications requests & block attacks from illegitimate request
- ***Assurance***
    - by default, a packet is allowed to travel between any source and destination in a network, introduce permissions & policies that control data flow in a network, implemented as explicit additions, firewalls designed to blocked traffic in and out of a network domain, if that traffic violates policies set by the administrators
- ***Authenticity***
    - headers & footers -> internet protocol don't have a place to put digital signatures, no notion of user identities, data exchanged between machines & allow for signatures, explicitly at the application layer & alternative protocol
- ***Anonymity***
    - no default notion of identity of users of internet -> built-in anonymity -> good for human rights worker reporting on abuses & bad if thief steals credit card numbers identity

The Link Layer

- modern operating systems include TCP & IP implementation, allow programs to interact with IP stack
- libraries support upper levels (including passing data to physical layer device drivers)-> starts with the link layer ↑ above the physical layer & concept of group sequences of bits into frames

2.1 Ethernet

- Ethernet refers to physical medium (cable) as well as the link-layer protocol
- frame transmitted on a cable -> impulse is sent through cable & received by other machines that connect to cable on same *local-area network (LAN)*
- portion of local-area network -> same logical connection : *network segment*
- two machines on same network segment transmit a frame at same time, a collision occurs & frames must be discarded and retransmitted.
- ***Dealing with Collisions***
  - The Ethernet protocol is designed so that eventually every machine in a network segment will succeed in transmitting its frame.
- ***Ethernet Hub and switches***
  - a device that logically connects multiple devices together, allowing them to act as a single network segment (particpate in Ethernet Collision resolution protocol), can generate large amounts of traffic
  - Switches act like hubs but overtime learns the address of machines that are connected. a switch will then only forward each frame it receives along the cable it knows is connected to the destination for that frame.
  - reduces possibilties collisions & increases speed of network, effective bandwidth
  - reduces risk of network eavsdropping as network frames forwarded by a switch are less likely to been seen by machines which are not destinations

Media Access Control (MAC) Addresses

- network interfaces are typically identified by a hardware-specific identifier known as its media access control address (MAC address)
- 48 bit identifier assigned to a network interface by its manufacturer
- sequence of six pairs of hexadecimal digits 00:16:B7:29:E4:7D
- used in the link layer to indentify devices in a network - intended to be unique
- first 24 bits are a prefix identity the organisation
- can be changed by software through driver of network
- change -> the second-least-significant bit of the most significant byte is set to 1, while in a manufacturer-issued MAC, this bit is set to 0
- faciliate routing of frames to correct destinations
- what switches use
- format of a ethernet frame:
  - preamble
  - delimiter
  - mac dest/src
  - ethertype length
  - payload
  - CRc-32 checksum ->confirm data integegrity

        - interframe gap

ARP Spoofing

- The Address Resolution Protocol (ARP) is a link-layer protocol that pro- vides services to the network layer. ARP is used to find a host’s hardware address given its network layer address.
- Used to determine the mac adrress associated with a give IP -> valuable service (man in the middle attack agaisnt protocol)

How ARP works ?

- let a machine want to send a packet to a dest machine on the same network.
- at network layer - scr knows the dest ip address, sending the packet is job of the link layer, src needs to idenify the mac address of dest machine
- in arp proctol, resoulution of ip address to mac is done by a broadcast message that queries all network interfaces on a local-are network so proper destination can respond.
- the reply is transmitted in a frame addressed to the machine that made the request.
- stored the IP-MAC address pair in a table called arp cache so doesn't have to do it again, src can send to dest
- lacks Authentication. Any computer on network could claim to request the ip address , any machine can recieve an arp reply even if it didn't make the request will automatically update the cache. Shortcoming, it is possible for malicious parties on a LAN to perform a arp spoofing attack,
- The attacker sned a arp reply to a target, who associates the ip address of the lan gateway with the attackers mac address. the atttacker sends an arp reply to bob associating the target ip's address with the attacker's mac address.this is arp cache poisioning. Bob now things alice ip is connected with eve mac and alice thinks bob ip is connected to eve mac. thus everything is routed through eve.
- Eve now has control of traffic and can sniff passwords or tamper with traffic.
- a denial of service attack is also possible.
- arp spoofing is derived from lack of identity veriffication
- to solve :
    - restrict LAN Access
    - checking for multiple occurrences of the same MAC address on the LAN, which may be an indicator of possible ARP spoofing.
    - Static ARP Tables :
        - requires a network administrator to manually specify a router’s ARP cache to assign certain MAC addresses to specific IP addresses.
        - requests to adjust the cache are ignored
        - reduces flexibility if a new device joins
        - does not prevent an attacker from spoofing a mac address to intercept traffic

3 Network Layer

Network layer move packets between any two hosts in a network on a best effort basis. Relies on the link layer to do this.

IP

- The *Internet Protocol* (IP) is a protcol which performs a best efford to route a data packet from a src node to a dest node.
- In IP, every node is given a unique numerical address:
  - 32 bit number(IPv4)
  - 128 bit number (IPv6)

Routing IP Packets

A host has an algorithm for routing packets from that host: - Packet is addressed to a machine on the same LAN, then packet is transmitted directly on LAN using ARP to determine mac address of dest machine. - packet is address to a machine that is not on the LAN, packet is transmitted to a specially designated machine on LAN(gateway). ARP protocol is used to determine mac address of gateway A host typically stores a list of IP address of the machines on LAN and the ip address of the gateway. Once a packet has reached a gateway node, needs to be further routed to its final dest Gateways and other nodes that handle routing of packets on the internet are called routers. Typically connected to two of more LANSs and use internal data structures known as routing tables to work out the next router. data packet with dest t, a routing table lets the router determine which neighnour it should send the packet to. Based on the numerical address,t, the routing protocol encodes the next hop from this router to each possible destination. There can be a misconfiguration in the routing tables that can cause a packet to travel forever. To stop this each IP packet is given (TTL - time-to-live) count by its scr. Known as a hop limit <= 255 hops and is decremented by each router that processes the packet. if it =0 then packet is discared and error packet is sent to scr.

The structure of the internet

Routers are designed to be very fast. From each packet - it performs :

- ***Drop*** : if packet is expired,it is dropped
- ***Deliver*** : If the destination is a machine on one of the LANs to which the router is connected, then the packet is delivered to the destination.
- ***Forward*** : If the destination of the packet does not belong to the LANs of the router, then the packet is forwarded to a neighboring router.

2 protocols that determine how the next hops are encoded:

- ***Open Shortest Path First (OSPF)*** : determines how packets are routed within an Autonomous system and is based on a policy that packets should travel along the shortest paths
- ***Border Gateway Protocol (BGP)*** : etermines how packets are routed between autonomous systems (ASs) and it is based on policies dictated by contractual agreements between different ASs The routes established by BGP may not be shortest paths.

Difference between a router and a switch is that a switch handles forwarding of packets on a single network and uses learned associations to reduce use of broadcasting . Router can belong to many networks and use routing tables to determine how to forward packets, can aviod broadcast altogether.

Bits in IP packet have a stucture

- fixed length header
  - total length of packet
  - TTL time to live
  - scr ip
  - dest ip
- variable length data portion

not guarantee that each packet successfully travels from its src to its dest, Ip can detect a packet headers are damaged. Comes with a checksum value, host or router can check -> need to recompute the function and compare value.the time-to-live, are modified with each hop, this checksum value must be checked and recomputed by each router that processes this packet.

Subnetworks (subnets) allow partitions in the networks into logical groups.IPv4 32-bit numbers that are stored as binary but written as 4 bytes

- network portion -> ip prefix used by all machines on the same network
- a host portion -> detect that particular device
- two portions are differentiated by providing a subnet mask along with the IP address.
- The network portion of the IP address can be identified by bitwise ANDing the subnet mask with the IP address, and the host portion can be identified by XORing this result with the IP address.
- subnet masks can be used to define address range of a particular network
- range of ip addtess are based on size of organization in question.
  - ***Class A*** network : largest, has a subnet mask of at least 8 bits and includes up to 224 = 16, 777, 216 unique IP addresses
  - ***Class B*** network : have at least a 16-bit subnet mask and up to 216 = 65, 536 unique IP addresses; they are typically allocated for ISPs and large businesses.
  - ***Class C*** network : 4-bit subnet mask, include up to 28 = 256 unique addresses, and are assigned to smaller organizations. IP addresses with the host portion consisting of all zeros or all ones have a special meaning and are not used for to identify machines. Thus, a class C network has 254 usable IP addresses. Total address space for IPv4 is on the verge of exhaustion: soon, all possible IPv4 addresses will be assigned. Network Address Translation, or NAT delays the exhaustion of the IPv4 address space, it doesn’t solve it, and an actual solution is provided by IPv6, which features 128-bit addresses.

Internet Control Message Protocol

- Network layer protocol (ICMP)

- used by hosts to perform a number of testing and error notfi tasks.
- network diagnostic tasks
- determining if a host is alive & finding the path followed by a packet:
    - ***Echo request*** : Asks the destination machine to acknowledge the re- ceipt of the packet
    - ***Echo response*** : Acknowledges the receipt of a packet in reply to an echo request
    - ***Time exceeded*** : Error notification that a packet has expired, that is, its TTL is zero
    - ***Destination unreachable*** : Error notification that the packet could not be delivered
- ***Ping*** :uses the ICMP protcol to verify whether or not a certain host is receiving packets. ICMP echo request message to the dest host, replies with an ICMP echo response message. Simple protocol is often the first diag- nosis tool used to test if hosts are working properly.
- ***Traceroute*** : ICMP messages to determine the path a packet takes to reach another host, either on a local network or on the Internet. It uses TTL field in the IP header. Attempts to send a packet to the target with a TTL. Receiving a packet with TTL of 1, intermediate router discards the packets and replies to the sender with an ICMP time excceeded message,revealing the first machine along the path to the target.. it sends another packet with a TTL of 2, reaching the first router in the path, the TTL is decremented by one and forwarded to the next router. in turn sends an ICMP packet to the original sender, incrementing the TTL field in this way, traceroute can determine each host along the path to the target

IP Spoofing

IP packet includes a place to specify the IP addresses of the destination and source nodes of the packet. validity of the source address is never checked, it is trivial for anyone to specify a source address that is different from their actual IP address. Nearly every operating system provides an interface by which it can make network connections with arb ip header info so it can make connections. spoofing is specifing the desired ip is scr filed of an ip packet data before sending it to network. IP spoofing does not let an attacker assume an new ip by changed headers, the ip stays the same

How IP Spoofing is used in other attacks

attacker sends an IP packet with a spoofed src address, won't receive a repsonse from dest server. spoofed src ip address on an outbound packet, machine with the server spoofed ip address will receive a response from the dest server not the attacker. attacker is using IP spoofing on his outbound packets, must not car about any response for these packets or has a way to receive response

Dealing with IP Spoofing

- Border Routes: routers span two or more subnetworks can be used to block packets from outside their admin domain that have scr address from inside domain
- IP traceback : tracing path of a packet back to its actual src address. Requests can be made to various autonomous systems along this path to block packets from this location

Packet Sniffing

Comprising confidentiality. Possible to listen in on the traffic . Known as packet sniffing. Performed independlty of wheather the packets are traveling via wireless Internet : attacker resides on the same network segment Frames are transmitted over an ethernet network, received by ever device on same network. segment will normally compare the frame's destination MAC address with its own MAC address, and discard the frame if it doesn't match. Network mode set to *promiscuous* mode, llows an attacker to examine all data transmitted over a particular network segment

Defenses against Packet sniffing

- PS can be used to troubleshoot network related problems -> computer is infected with adware or spyware
- Use Ethernet switches instead of Hubs, less number of machines on attacker network segment,Note that there is no analog to the switch when communicating wirelessly,
- detect when network devices are in promiscuous mode
    - the fact that when a network interface is receiving all network traffic, the operating system behind that network interface is using much more processing power than if these frames were being dropped
    - elicit responses to invalid packets from network devices may provide clues suggesting
    - should use encryption HTTPS instead of HTTP

4 Transport Layer

supports communication between machines.addressing extended is achieved by viewing each machine as having lots of ports. 16-bit source and destination port numbers. Protocols:

- Transmission Control Protocol (TCP)
- User Datagram Protocol (UDP) Extra feature of tcp connection oriented and provides reliable stream of bytes, with a gurantee that info arrives intact. if lost it is resend UDP best-effort communicatiin : where speed is more important than completeness.

Transmission Control Protocol (TCP)

Takes IP Protocol and gurantees transmission of a stream of bits between two virtual ports.

Transmission Control Protocol (TCP)

TCP session starts with connection between sender & receiver, parties can then communicate. Ensures reliable transmission by a sequence number that is initalised during the three-way handshake, Transmission features an incremented sequence number, each part is aware when packets arrive, out of order all or not at all, Incoporates a cumulative acknowledge schemem two TCP sessions , sender and receiver communicating via their established TCP connection. the sender sends the receiver a specified amount of data, the receiver will confirm that it has received the data by sending a response packet to the sender with the acknowledgment field set to the next sequence number it expects to receive. If any information has been lost, then the sender will retransmit it. Manages amount of data that can be sent by one party while avioding overwhleming, processing resources of other or bandwidth of network itself(*flow control*) use *sliding window protocol* -> receiver informs sender of size of receive window(number of bytes of data willing to accept befor sender must pause and wait for a response), sender keeps track of value of last thing. When sending , the sender checks the sequence number of packet and continues only if number is less than last acknowedlenge plus current size of recevive window. Otherwise it waits, stores the number shifting sliding window of sequence numbers. Sender sets a time so if no acknowledgment is received before the times expires, sender assumes data loss and retransmits Checksum field to ensure correctness. not intended to be cryptographically secure, detect inconsistencies in data due to network erros rather than tampoering.

Congenstion Control

attempt to prevent overwhelming a network with traffic which could lead in poor transmission rates and dropped packets. congenstion control is not implemented into tcp packets but based on the info gathered by keeping track of acknowledgements for prev sent data and time required for certain operations. TCP adjusts data transmission rates using this information to prevent network congestion.

TCP Packet Format

Format of TCP :

- dest port
- scr port
- communication connection for this packet and others like it
- connections have a state

TCP Connections

- three-way handshake to establish a reliable connection stream between 2 parties, client sends packef to desired dest with SYN(synchronization) flag, included a random initialization for *sequence numbers*, used for reliable ordering of future transmissions. SErver replies with a packet marked with SYN and ACK(acknowledgement) flags (SYN-ACK) packet, says servers wants to accept. set with an acknowledgement no(set to more than 1 of sequence number and new random seq num) client responds with ACK packet to say successful connect, Final ACK packet features an acknowledgment number (set to one more than the most recently received sequence number, and the sequence number set to the recently received acknowledgment number.)

16 bit port num differenitate multiple TCP connections. packets include src port and dest port. orts may range from 1 to 65,535 (216 – 1), with lower port numbers being reserved for commonly used protocols and services. port 80 - HTTP port 21/22 - FTP , SSH applications can create network connections using sockets

User Data Protocol (UDP)

- UDP doesn't make a gurantee about the order /correctness of packet delivery. No inital handshake to establish connection, allows to send messages (datagrams) asap. if sender wants to communicate via UDP, it only needs a socket.

- UDP features a 16 bit checksum to verfiy the inegrity og each packet, no seq num scheme, so transmissions can arrive out of order or may not arrive at all. Assumed that checking for missing packets is left to application. UCP can be faster than TCP
- Often used in time-senstive applications where data integrity is not as important as speed(DNS or VoIP) TCP is used for applications where order and intergirty matter (HTTP,SSH,FTP)
- Header :
  - Src port
  - dest port
  - length
  - checksum
- Payload
  - Data

Network Address Translation(NAT)

- add network devices to home network, typically don't buy ip address and set up the new address directly on the Internet.
- *Network address Translation* : allows machines on local-area network to share a single public IP address. Public IP add represents the point of contact with internet for entire LAN, while machines on network have private IP add that only accessible from within network
- NAT allows an entire network to be asssigned to a single public IP add, widespread use of NAT, delayed inevitable exhaustion of the IPv4 space, a lot of address capacity for NAT, because there are a number of private IP addresses that such networks are allowed to use which cannot be used on the (public) Internet.
- e IP address are of the form 192.168.x.x, 172.16.x.x through 172.31.x.x, and 10.x.x.x.
- NAT router represents the gate- way between private IP addresses and the public Internet, and this router is responsible for managing both inbound and outbound Internet traffic.

How NAT Works?

- translate by a look up table-> contains entries:
  - private scr IP
  - private scr port
  - dest ip
  - public src port
- dynamically rewrite headers of all inbound and outbound tcp and udp packets,
- machine on internal network attempts to send a packet to an external IP address, the NAT router creates a new entry in the look up table associated with the src machine private ip add, & internal src port of transmitted packet,rewrites the scr ip to be that of NAT device public ip , opens a new public src port, rewrites ip header src port field contain newly opened port. public port and dest ip are recorded with private src ip and private internal port in NAT lookup table. device also adjusts any checksum contained in packet, including used by ip, tcp&udp to reflect changes made, packet is forwared to dest.

- NAT router check look up table for any entries whose public scr port corresponds to dest port of inbound packet & dest ip address corresponds to src ip on inbound packer. NAT router rewrites ip headers of inbound packet in line with lookup table, packet is forwared to correct private ip add and private port
- effecrivley manages outbound traffic, several restrictions on inbound traffic, external machine has no way to start a connection with machine on provate network, since internal m doesn't have a oublicy accessible IP address, can be a security feature since no inbound traffic from internet can reach internal network, NAT devices can function as firewalls, blocking risky contact from the external internet
- violates the ideal goal of end-to-end connectivity for machines on internet by not allowing direct communication between internal and external parties, NAT can cause problems with protcols, if not tcp or udp as a transport layer protocol, cruical in delaying exhaustion of IPv4 address space and simplifying home networking

TCP Session Hijacking

TCP Sequence Prediction

- *Session Spoofing* creates a spoofed TCP session instead of stealing an existing one, type of session hijacking, *TCP sequence prediction* attack triws to guess the inital sequence number sent by the sercer at start of a session. Early TCP stacks implemented sequence numbers by using a simple counter that was added 1 with each transmission.Not using any randomness, trival to predict next sequence number, key to attack. Modern TCP stack implementations use pseudo-random numbers generators to determine seqence numbers, makes TCP sequence prediction attack more difficult, not impossible. Following scenarios could happen:

1. The attacker launches a denial-of-service attack against the client victim to prevent that client from interfering with the attack.
2. The attacker sends a SYN packet to the target server, spoofing the source IP address to be that of the client victim.
3. After waiting a short period of time for the server to send a reply to the client (which is not visible to the attacker and is not acted on by the client due to the DOS attack), the attacker concludes the TCP handshake by sending an ACK packet with the sequence number set to a prediction of the next expected number (based on information gathered by other means), again spoofing the source IP to be that of the client victim.
4. The attacker can now send requests to the server as if he is the client victim.

Blind Injection

above attack only allows one-way communication as attacker cannot receive any replies from the server due to use of the IP spoofing. method may allow an attacker to subvert a system that excutes a certain command based on scr of IP of requester.possible to inject a packet containing a command that creates a connection back to the attacker.

ACK Storms

possible side effect of blind injection attack, is that can cause a client and server to become out-of-synchronization with respect to a sequence numbers, as server got a synchronized message, client never actually sent. TCP incopoartes a method for clients & servers to become resynchronized when they go out of step, but it doesn't easy tolerate desyncronization that happens after a blind injection attack. Server and client start sending ACK messages to start using correct sequence numbers, known as ACK Storm, can continue until one of these messages is lost by an accident or a firewall detects an ack storm in progress and discards a bad ack message.

Complete Session Hijacking

attacker on same network segment as target server, and or client, attacker can hijack an tcp session. possible due to packet sniffing to see sequence numbers of packets used to establish session. attacker can inject a packer with high probable sequence number to server useing a spoofed src ip address to be like the client. Used in combination with other network attacks, possibility of an attacker who is in same network segment as target server or client victim allows for an even stronger type of session hijacking attack. an attacker on same network segment as client or server can use packet sniffing to see sequence numbers of packets used to establish a tcp session as in a complete session stealing attack, but he can also sometimes go a step further by creating a man-in-the-middle attack using ARP spoofing method. Once done attacker can perform all subsequent actions as if he were the user, masquerading as, he can intercept all responses from both sides.

Countermeasures

- use encryption & authentication, at network layer suce as IPsec or application layer
- web sites avoid creating sessions that begin with secure authentication measures but subsequently switch over to unencryted exchanges.
- trade fair efficiency for security, create a risk with respect a tcp session hijacking attack.

Denial-of-Service Attacks

Bandwidth in a network is finite, number of connections of a web server can maintain to clients is limited. connection to a server needs a minimum amount of network capacity to function. A server has used up its bandwidth or ability of its processors to respond to Requests, additional attempted connections are dropped & potential clients will be unable to access resources provided by server. attacl that is designed to cause a machine or piece of software to be unavailable and unable to perform its basic functionality is known as a denial-of- service (DOS) attack.

Not concerned with receiving responses from a target, spoofing src ip address is commonly used to obscure the identity of the attacker as well as make mitigation of attack more difficult. Servers can stop DOS attacks by dropping all packets from certain blacklisted ip address, attackers can generate a unique src ip address for every packet sent, preventing target from successfully identifying & blocking attacker. IP spoofing can make it more difficult to target the src of a dos attack.

ICMP Attacks

The Ping Flood Attack

- Ping utility sends an ICMP echo reuqest to a host, returns with ICMP echo reponse.

- powerful machine sends massive amounts of echo requests to a single victim server
- attacker can create more ping requests than victim can process & victim has enough network bandwidth to receive all these requests, victim server will be overwhelmed with traffic & start to drop legitimate connections

The Smurf Attack

- takes advantage of misconfigured networks is known as a smurf attack
- networks feature a boradcast address - user can send packet that is received by every IP address on network.
- Smurf attacks exploit property by sending ICMP Packets with scr address set to target and with dest address set to broadcast address of network
- Each packet received by every machine on network at which point every machine sends a reply ICMP packet to indicated src address of target, results in am amplification effect that multiplies number of packets sent by number of machines on network.
- victmins many be on exploited network or traget, identity of attacker is further obscured
- to prevent
  - admin should configure hosts & routers on their networks to ignore broadcasts requests,
  - routers should be configures to avoid forwarding packets directed to broadcast address, poses a security risk in network can be used a ping flood amplifier, server is relatively weak, wise for it to ignore ping requests to avoid ping floods.

SYN Flood Attacks

to initiate TCP session ,client sends a SYN packet to a server, in response to which the server sends a SYN,ACK packet. This exchange is normally then followed by client sending a concluding ACK packet to server, client nevers sends the concluding ACKm server waits for a certain time out period then discards session.

How a SYN Flood Attack Works

attacker sends a large number of SYN packets to server, ignores SYN/ACK replies, never sends expected ACK packets, attacker initaing attack in practice will use random spoofed src address un SYN packets he sends, so replies are sent to random ip address. attacker sends a large amount of SYN packets with no corresponding ACK packets, server's memory will fill up with sequence numbers that is is remebering in order to match up with TCP sessions with expected ACK packets, never arrive so wasted memory blocks out other legiti ate TCP session requests.

Defenses Against SYN Flood Attacks

- SYN Cookies are implemented, rather than dropping connections as memory is filled, server sends crafted SYN/ACK packet without creating a corresponding memory entry. Sever encodes info in TCP seq number as follows:
  - first 5 bits : timestamped, counter incremented every min mod 32
  - next 3 bits : encoded value, max segment size of transmission
  - final 24 bits : MAC of the server, client ip address, server client port numbers , prev time

stamp using secret key

How SYN Cookies work

legit client must reply with a sequence number eqal to pre sent sequence num plus 1,when client replies with ack packet server subtracts 1, to obtain prev sent sequence, compares first 5 bits with current timestamp to check if connection has expired. Server recomputes 24 bit mac usig known IP and port info and compares value encoded in seq number. server decodes middle 3 bits to finish reconstruct syn quee entru, connection can countinue, If good, with syn cookie check, server initaties tcp session.

SYN Cookies Limitations

- max segment size only be 8 possible values, most encode in 3 bits
- SYN Cookies don't allow use of TCP options field, infor stored alongside SYN queue entries

Optimistic TCP ACK Attack

num of tcp packets allowed outstanding during tcp communication session before requiring an ACK - congenstion window. Sever receives ACKs from a client, dynmaically adjusts congestion window size, w to reflect estimated bandwith avail

window size grows when acks are received, shrinks when segments arrive out of order, or not received at all, indicating missing data. TCP helps to keep network congestion down, while also trying to push data through the internet as quickly as possible without overloading the capacity of routers along path that the packets are travelling . Congenstion control nature of TCP automatically adjusts as network conditions change, shrinking congestion win when packets are lost and increase when sucessful acknowldge ew

How the Optimistic TCP ACK Attack Works

denial or service attack, makes congenstion control mechanism of TCP work agaisnt itself.rogue client tries to make server up sending rate until runs out bandwidth. if performed with many servers , creat internet wide congestion. Send ack packets before they have been received, to make server up transmission speed.

Defense Against the Optimistic TCP ACK Attack

implement max taffic limits per client at server level, block traffic whose pattern indicate denial of service attempts

Distributed Denial of Service

server tech allows website to handle enormonus amount of bandwidth. *distributed denial-of-service (DDOS)* attack, malicious users levergae power of many machines to direct traffic against single website. Use botnets

- servers incoporate DOS protection mechanisms that analyze incoming traffic and drop packets from srcs consuming bandwidth. IP spoofing may make DDOS prevention more difficult

IP Traceback

determining the actual origin of a packet on the Internet, without relying on the IP source field contained in that potentially falsified packet.

ip traceback tech relied on logging each packet forwareded by each router. significant space requirements on routers which may not have incentive to cooperate.

packet marking routing mark forwarded packets with info related to the path that packet has taken up to that point.

node sampling single field in ip packet that has only enough room for one address is used. Each router overwrites this field of each packet with own address with some probability p.

Crypography

> used for confidentiality, integrity, authentication, anonymity

Historical Ciphers

- *Rail fence Cipher*

  > shared secret key, plaintext written in columns of size k. The ciphertext is the concatenation of the resulting rows

  - Decryption: ciphertext written in rows of size $\frac{\mod c}{k}$
  - small key space size $k < \mod c \Rightarrow$ brute force attack
- *Substitution cipher*

  > shared secret: a permutation ω of the set of characters, Encryption: apply ω to each character of the plaintext.

  - Decryption: apply ω^{-1} to each character of the plaintext.
  - break using frequency of letters,diagrams, triagrams, expected words
- *Vigenere cipher*

  > shared secret key: a word w over the English alphabet,break the plaintext $m = m_1 \ldots m_n$ in $\frac{m}{w}$ encrypt each block as follows : $m_{i+1} + w_1 = m_{i+1} + w_1$ mod 26

  - Decryption: break the ciphertext $c = c_1 \ldots c_n$ in $\frac{m}{w}$ blocks, and decrypt each block as follows $c_{i+1} - w_1 = this$
  - Can perform frequency analysis on this

Chapter 8.1.3 One time pads

> same approach as vingrene cipher, use a block of keys $(k_1, k_2, k_3, .. k_m)$ to encrypt plaintext M of length n

- length, m, of the block of keys has to be the same as n, the length of the plaintext.
- Each shift amount, k_i, must be chosen completely at random.
- No statistical analysis can be done
- Perfect secrecy

Binary One-Time Pads

- view plaintext messgae M as binary string of length n, view pad P to be a random binary string of length n, $C = M \oplus P$
- Can recover plaintext from ciphertext using $M = C \oplus P$

Chapter 8.1.4 Pseudo-Randomm Number Generators

The Linear Congruential Generator

> numbers its generates are uniformly distributed

start with a random number x_0 (seed) generate the next number x_{i+1} according to this : $x_{i+1} = (ax_1 + b) mod$n assume $a > 0$ and $b \geq 0$ a and n are relatively prime, can prove unformly distrubuted.

Security Properties for PRNG's

- should be difficult to predict next number from prev number.
- numbers will start to repeat itself after period has finished

---

Stream Ciphers

Kerckhoff's principle

- Encryption and Decryption algorithms should be made public
- Security relies on secrecy of key
- subject to two time pad attacks

Block Ciphers

> A block cipher with parameters k and ℓ is a pair of deterministic algorithms (E, D) such that ption

- Encryption E : $0, 1^K \times 0, 1^\ell \Rightarrow 0, 1^\ell$

- Decryption D : $0,1^K \times 0,1^\ell \Rightarrow 0,1^\ell$

8.1.6 The Advanced Encryption Standard(AES)

AES is a block cipher operates on 128-bit blocks, designed to be used with keys that are 128, 192 and 256 bits long, yielding ciphers known as AES-X

AES Rounds

128 bit verison of AES encryption algorithm in 10 rounds. performs an invertible transformation on a 128 bit array called state,. Inital state is X_0 is XOR of the plaintext with the key K: $X_0 = P \oplus K$

Round i(i=1..10) receive state X_{i-1} as input and produce state X_i. ciphertext C is the output of the final round: $C = X_1$

Each round is bulit from four basic steps

1. *SubBytes step*:* an S-box substitution step
2. *ShiftRows step*: a permutation step
3. *MixColumns step*: a matrix multiplication (Hill cipher) step
4. *AddRoundKey step*: an XOR step with a round key derived from the 128-bit encryption key

Implementation of AES

optimised for speed of excution and use serveral look up tables to implement the basic steps of each round. Look up tables stores all possible values of a function into an array that is indexed by the input of the function.

Attacks on AES

Timing attack on high performance software implementation of AES were independlty discovered in 2005. based on fact that sache of processor where the AES algorithm is excuted will store portions of look up tables used. Acessing cache faster than accessing memory, time takes to execute algorithm using same key on a series of known cipher texts, attacker can eventually learn key

attacker and AES excuted on same machine, key can be recovered in less than a second, attacker and AES computation on different machines, recovering key takes several hours,defend against timing attacks,

AES should be implenented in a way that the execution time remains constant,

8.1.7 Modes of Operation

Electronic Codebook (ECB) Mode

> simple encryption for a block cipher to encrpty each block independently. $C_i = E_k(B_i)$

- simplicity, can tolerate the loss of a block
- resilience to block loss comes from the fact that deceypting the ciphertext for a block, B_1 does not depend in any way on the block B_{i-1}
- disadvan determinstic , each plaintext has unique associated ciphertext the ECB mode may reveal patterns

Cipher Block Chaining (CBC Mode)

> avoids revelations of patterns in a sequence of blocks in cipher block chaining mode $C_1 = E_k(B_i \oplus C_{i-1})$ B_1 is exclusive-ored with initalization vector C_0

- advan
  - identical blocks appear at different places in input sequence, likely to have differebt encryptions in ourput sequence.
  - doesn not allow the encryption of blocks in a sequence to be done independently
  - decrypt can begin in parallel is all cipher text blocks are available

Cipher Feedback (CFB) Mode

> block encryption is similar to that of the CBC mode, B_i involves the encryption C_{i-1}. Formula is $C_i = E_k(C_{i-1}) \oplus B_i$

- decryption involves encryption of (i-1)st ciphertext

Ouput Feedback (OFB Mode)

> sequence of blocks is encrypted much as one time pad, sequence of blocks that are geenrated with block cipher. Encryption algortihm begins V_0 formula becomes $V_i = E_k(V_{i-1})$ which encryted becomes $C_i = V_i \oplus B_i$

- tolerate block losses, performed in parallel

Counter (CTR) Mode

> start with random speed, compute the ith offset vector according to formula $V_i = E_k(s + i - 1)$, encytpiton is done by $C_i = V_i \oplus B_i$

- generation of pad vectors as well as encryption and decryption, done in parallel

8.5.1 Details for RES

> 128 bit blocks, 16 bytes of 8 bits each arranged in a 4x4 matrix

SubBytes Step

> each byte in matrix is sub with replacement byte by S-box (look up table) for mathematical equation that operates in an esoteric number known as GF(2^8)

ShiftRows Step

> simple permutaion, has effect of mixing bytes in each row , amounts to a cyclic shift

MixColumns Step

> mixes up info in each column of 4x4 matrix, from shiftrow step, mixing by application what amounts to a Hill-cipher matrix multiplication transformation applied to each column.

- arthimetic used to evaluate the polynoimal is modulo 2.
  - addition is XOR
  - multiplication is AND

AddRoundKey Step

> exclusive-or result from prev steps with a step of keys derived from 128 bit secret key

AES Key Schedule

> using pseudo- random number generator

4x4 matrix, applied to plaintext directly before any of the steps in round 1 is simple

secret key, K divided into 16 bytes and arranged into a 4x4 in column major ordering

round 0 key matrix, refer to column as W[0], W[1], W[2], ...

Might need to go through again

Chapter 8.3 Cryptohash functions

Properties and Applications

> one way, given a message M, should be easy to compute a hash value H(M) from that message

given only a value x, should be difficult to find a message such that X=H(M)

Standard hash SHA-256 produces has values for 256 bits

Applications include:

- Commitments
- File intergrity
- Password verfication
- Key derivation
-

Collision Resistance

> hash function, H, is a mapping of input strings to smaller output strings.

> A function f is collision resistant if there is no efficient algorithm that can find two messages m_1 and m_2 such that $f(m_1)$ = $f(m_2)$

- H has *weak collision resistance* if any given message M, computaitonally diffcicult to find another message.
- H has *strong collision resistance* if it has computationally difficult to compute two distinct messages M_1 and M_2 such that $H(M_1) = H(M_2)$

The Merkle–Damga˚rd Construction

> common structure for hash function to use a a bulding block *cryptographic compression function C(X, Y)* X has fixed length m and Y has fixed length n, produces a hash value of length n

- Messgae M , divide into blocks of length m, last block padded with bits to make m.
- Apply compression function C to first block M_1 , fixed string length v of length n, known as initalization vector -> $d_1 = C(M_1, v)$, apply compression fucntion to next one $d_2 = C(M_2, d_1)$
- attacker finds collision between two different messages M_1 and M_2 , then can form other arbitray collisions. $H(M_1||P) = H(M_2||P)$

Practical Hash Functions for Cryptographic Applications

> SHA-256 and SHA-512 MD5 hash function, considered insecure as several attacks

Birthday Attacks

> Brute force way of finding the problem

Chapter 8.2 Public Key Crytography

a and b in $\mathbb{Z}$ are relative primes if they have no common factors

Modular Arithmetic

> $\mathbb{Z}_n = 0, 1, 2, .., n - 1$

Modulo Operator

> x modulo n = a , remember to take the remainder

Modular Inverses

> xy mod $n = 1$

Modular Exponentiation

> x^y mod n

The RSA Cryptosystem

> plaintext and ciphertext message blocks as large numbers using thousands of bits.
> encryption and decryption done using modular exponentiation, based on Euler's Thm

Encryption and Decryption

Receiver BOb to create public & private keys,

> generate two large prime numbers p and q setting n = pq

> picks a number, e that is relatively prime to $\phi(n)$, computes $d = e^{-1} mod \phi(n)$

> public key is pair (e,n) and private key is d

> Alice can encrypt a message M, $C = M^\ell$ mod n

> decrypt the cipher text C, Bob performs a modular exponentiation C^d mod n

Security of the RSA Cryptosystem

> based on difficulty on finding d, given e and n $\phi(n) = (p-1)(q-1)$, easy to compute d from e
>
> - side channel attacks have been done , it is determinstic however

Efficient Implementation the RSA Cryptosystem

- *Primality Testing* : testing if an integer is prime
- Computing GCD
- Computing modular inverse

The Elgamal Crypotosystem

> uses randomization, independent encryption of same plaintext are likely to produce different ciphertexts

- viewing input blocks as numbers and applying arithmetic operations on these numbers to perform encryption & decryption
- $\mathbb{Z}_p$, a number g in this is said to be a generator of primitive root modulo p, if for each postivie interger i in $\mathbb{Z}_p$, there is an integer k such that $i = g^k \mod p$
- the discrete logarithm problem $x = g^k \mod p$, known to be computaionally hard to solve

Bob chooses a random large prime p and finds a generator g. He picks a random number x between 1 and p-2 and $y = g^x \mod p$. The number x is Bob's secret key, public key is (p,g,y)

Alice encrypts a message M, for Bob, by getting his public key. Generates a random number k, between 1 and p-2, uses modular multiplication and exponentation to compute numbers

- $a = g^k \mod p$
- $b = My^k \mod p$

Decryption and Security Properties

> dependent on the choice of random number k, each time Alice does this, she needs to use a different random number cipher text for Bob, he can decrypt by computing $a^x \mod p$, computing the inverse moduluo p and timesing by b modulo p $M = b(a^x)^{-1} \mod p$

Key Exchange

> Symmetric Cryptosystem requires that Alice and Bob agree on a secret key before they can send encrypted message A *key exchange protocol*, which is also called *key agreement protocol*, cryptographic approach to establish a shared secret key.

- Diffie-Hellman key exchange protocol (DH protocol) based on modular exponentiation, assumes two public parameters have been established.
    - prime number p
    - generator g for $\mathbb{Z}_p$
- Alice picks a random positive number x in $\mathbb{Z}_p$ and uses it to compute $X = g^x \mod p$.She sends X to Bob.
- Bob picks a random postitve number y in Z_p and uses it to compute $Y = g^y \mod p$. He sends Y to Alice
- Alices computes the scret key as $K_1 = Y^x \mod p$
- Bob computes the secret key as $K_2 = X^y \mod p$

Both have computed the secret key $K = g^{xy} \mod p = K_1 = K_2$ SEcurity based on assumption that it is difficult for attacker to determine the key K from the public parameters. No methods are known for effciently computing $K = g^{xy} \mod pfromp, X = g^x \mod p, Y = g^y \mod p$ called the Diffie-Hellman Problem

Attack

- attacker picks numbers s and t in Z_p
- When Alice sends the value $X = g^x \mod p$ to Bob, the attacker reads it and replaces it with $T = g^t \mod p$.
- When Bob sends the value $Y = g^y modp$ to Alice, the attacker reads it and replaces it with $S = g^s modp$.
- Alice and the attacker compute key $K_1 = g^{xs} \mod p$
- Bob and the attacker compute key $K_2 = g^{yt} modp$
- When Alice sends a message to Bob encrypted with the key K1, the attacker decrypts it, reencrypts it with the key K2 and sends it to Bob
- When Bob sends a message to Alice encrypted with the key K2, the attacker decrypts it, reencrypts it with the key K1 and sends it to Alice

8.5.2 Details for RSA

Fermat's Little Theorem

> Let p be a prime number and g be any positive integer less than p, then $g^{p-1} \mod p = 1$

Euler's Theorem

> Let x be any positive integer that is relatively prime to integer n >0 then $x^{\phi(n) \mod n=1}$

Corollary

> Let x be a positive integer relatively prime to n, and k be any positive integer. Then $x^k \mod n = x^{k \mod \phi(n)} \mod n$

Euclidean Algorithm:

> The GCD d of two numbers, a>0, and $b \geq 0$ is the samlled positive integer d such that $d = ia + jb$ for integers i and j

Extended algorithm first test if b is zero otherwise recurive call algorithm

---

Efficiency of extended euclidian algorithm, running time of algorithm $\ceil\log a$

Modular Exponentiation Running time is $\ceil\log n$

Repeated Squaring using linear number of multiplications rather than an exponentiation

---

Primality Testing

> Given a integer n, want to determine if n is prime or not

How RSA is Typically Used

1. Encrypt a secret key, K, with the RSA cryptosystem for the AES symmetric cryptosystem.
2. Encrypt with AES using key K.
3. Transmit the RSA-encrypted key together with the AES-encrypted document.

Chapter 8.4 Digital Signatures

> *digital signature* is a way for an entity to demonstrate the authenticity of a message by binding its identity with that message.

- *Nonforgeability*. It should be difficult for an attacker, Eve, to forge a signature, $S_{Alice}(M)$, for a message, M, as if it is coming from Alice.
- *Nonmutability*. It should be difficult for an attacker, Eve, to take a signature, $S_{Alice}(M)$, for a message, M, and convert $S_{Alice}(M)$ into a valid signature on a different message, N.
- if a digital signature scheme acheives these properties, actually achieves *nonrepudiation* , should be difficult for Alice to claim she didn’t sign a document, M, once she has produced a digital signature, $S_{Alice}(M)$, for that document.

The RSA Signature Scheme

> Any third party can verify signature by testing the following information : is it true that $M = S^{\ell} \mod n$ Verification method follows from $de \mod \phi(n) = 1$

-> RSA signature does not acheive non-mutability

The Elgamal Signature Scheme

> document signatures are done throught randomization. To sign a message M, Alice geenrates a fresh one time use random number k and computes the following two numbers $a = g^k \mod p$ and $b = k^{-1}(M - xa) \mod (p-1)$

(a,b) is Alice signature on the message M, verify the signature(a,b) on M, bob performs the following test: Is it true that $y^a a^b \mod p = g^M \mod p$

> based on compuation of b depends on both random number k, and Alice's secret key x

Using Hash Functions with Digital Signatures

> inefficient use of signatures if message is long, digital singatures shcemes are usally applied to cryptographic hases of messages not to actual messages.

Public Key Cerficates

> CAs issue certificates to participants on their public key

contains :

- public key
- subject
- signature

X.509 certificates : defines a framework for the provision of authentication services

Chapter 6.1 The Application Layer and DNS

A sample of Application Layer Protcols

Domainnamesystem(DNS)

> This is the protocol that allows us to use intuitive domain names to refer to Internet hosts rather than using IP addresses. Most application programs and other application-layer services rely on DNS.

Hypertext transfer protocol (HTTP)

> This is the protocol used to browse the Web.

SSL/TSL

> This is the protocol used for secure, encrypted browsing (i.e., with HTTPS).

IMAP/POP/SMTP

> These are protocols that make Internet email pos- sible.

File transfer protocol(FTP)

> This is an old,but still used,protocol that provides a simple interface for uploading and downloading files. It does not encrypt data during transfer.

SOAP

> This is a more recent protocol for exchanging structured data as a part of the web services paradigm.

Telnet

> This is an early remote access protocol. Like FTP, it doesn't encrypt connections.

SSH

> This is a more recent secure remote access and administration protocol.

The Domain Name System (DNS)

> Resolving domain names to ip address

- arranged in a hierarchy that can be examined from left to right
- top-level domain
- subdomain

Domain Name Registration

- Generic top-level domain: such as .com, .net
- contry-code top level domain .pt(portugal) with restricted to entities within a specific country [://:@:/?]

Domain names are registered and assigned by *domain-name regis- trars* accreditied by *Internet Corporation for Assigned Names and Numbers (ICANN)*

Cybersquatting or domain squatting has become common, a person registers domain name as desirable and cam make money.

How DNS is Organized

> Hierachical nature of domain names reflected in way the internet infrastructure supporting DNS systemm works.

- Resolve a domain name to corresponding IP address DNS hierachy is used to query known as name servers. At the top of name server hierachy, they are root name servers which are responsible for top-level domains such as .com, .it
- Root name servers store root zone database of records, indicating authoritative name server of each top level domain

How DNS Queries Work

- Client wishes to resolve a domain name to ip, contacts designated name server assigned to the machine.
- server issues a DNS query to a root name server, it responds with address fo server that is authoritative for the next level of hierarchy.
- this sequence of requests and responses continues until a name server responds with the IP address of the requested domain. This final name server is therefore the authoritative responder for the requested domain name

DNS Packet Structure

> queries and replies are transmitted via single UDP packet with tcp being used as a subsitue for requests or replies exceeding 512 bytes

- Header : includes 16 bit query identifier(transaction identifier)
- Query: sequence of questions each conisting of domain name quiered and type of record requested, query id is seletced by client sending the query and is replicated in the response from the server
- Sequence of DNS records
  - Name field of variable length
  - 2 byte type field indicates the ytpe of DNS Record,A standard domain-to-address resolution is described by an A record, but other types exist as well, including NS records (providing information about name servers), MX records (providing information about email resolution), and several other less commonly used record types.
  - byte CLASS field denotes the broad category that the record applies to, such as IN for Internet domains.
  - 4-byte TTL field specifies how long a record will remain valid, in seconds.
  - 2-byte RDLENGTH field indicates the length of the data segment, in bytes.
  - variable-length RDATA segment includes the actual record data. For example, the RDATA segment of an A record is a 32-bit IP address.

DNS Caching

> central service utilized by billions of machines connected to the internet, without any additional mechanisms, would place incredible burden on high-level name servers

- DNS cache, table of recently received DNS Records.
- Name server first checks the cache and returns to the client the requested IP if a record is found, designatined server queries the root server and resolves the domain name.
- alue known as the time-to-live (TTL) determines how long a DNS response record remains in a DNS cache. This value is specified in the DNS response, but administrators can configure local settings that override the provided TTL values. Once a cached record has expired, the query process resorts back to asking a higher-level name server for a response

DNS Attacks

Phraming and Phishing

> DNS subverted so that an attacker could control how DNS requests resolve attack known as pharming, resolve domain to site which appeard dientical to requested site but instead designed for malicious intent

Other Pharming Attacks

> distinguish between fake and real sites Email relies on speicalized DNS entries known as MX records ain name used for operating system updates with a malicious IP address, causing victims to automatically download

DNS Cache Poisoning

> attacker attempts to trick a DNS Server into caching a false DNS Record

- attacker decided to launch DNS cache poisioning attack against an ISP DNS Server
- Eve sends a DNS response to her own query spoofing the src ip address
- ISP server accepts eve's foged response

An attacker overcomes:

- attacker must issue a response to her own DNS query before authoritative name server is given a chance to respond
- each DNS request is given a 16-bit query ID. If the response to a query is not marked with the same ID as its corresponding request, it will be ignored

DNS Cache Posioning and the Birthday Paradoz

> guessing is actually more likely if the attacker issues a lot of fake requests and responses to the same domain name lookup

Increase in attack success probability from an increases in fake reuqests in known as brithday paradox

Subdomain DNS Cache Poisoning

> above guessing attack is extremely limited because of its narrow time frame if in cache uses that record rather than going to name server, attacker can only make so much guesses

- each failed guessing attempt, valid reponse will be caches bu tagerted name server, so attack must wait until it expires

new subdomain DNS cache poisoning attack, issues requests for a different nonexistent subdomain for target domain, don't exist, ignores reuqests. he attacker issues responses for each of these requests, each with a guessed transaction ID. Because the attacker now has so many chances to correctly guess the response ID and there is no competition from the target domain to worry about, it is relatively likely that the attack will be successful

Using Subdomain Resolution for DNS Cache Poisoning

> attacker's response include a glue record that resolves the name server of the target domain to an attacker-controlled server sucessfully guessing the transaction ID the attacker, can control not just one DNS resolution for a nonexistent domain but all resoultions for the entire trarget domain

Client-Side DNS Cache Poisoning Attacks

> attacker can construct a malicious web site containing image tags which issue a request to a different non existent subdomain, of the domain attacker wishes to posion,

Identifying the Risks of Subdomain DNS Cache Poisoning

- ReRelying on a 16-bit number as the only mechanism for verifying the authenticity of DNS responses, which is insufficient for security
- Having the response for a nonexisting subdomain request be a nonre- sponse

Some Defenses Against Subdomain DNS Cache Poisoning

- reduce chances of succesful attack, many DNS implementations incoporate source-port randomization (SPR), the practice of randomizing the port from which DNS queries originate (and must be replied to)

- decreases likelhihood of succesfully generating a false reply

DNSSEC

> set of security extensions to the DNS protcol that prevent attacks such as cache poisoning by digitally signing all DNS replies using public-key cryptography

- - deployed at both client and server
- reuquest packet indicates this is supported
- If the queried server also supports DNSSEC, then a resource-record signature (RRSIG) record is returned alongside any resolved queries
- contains a digitak signature of returned records computed by geenrating a hash of returned products with authoriative name sevrer key.
- response to client contains a DNSKEY record containing public key of name server
- verify authenticty
- establish trust in the name server
- To perform signature verification, the client uses the parent name server's DNSKEY to decrypt the RRSIG record, compares this to the DS record, and finally compares the DS record to the child name server's DNSKEY
- until a "trusted key" that the client has existing knowledge of and does not need to verify is encountered

Chapter 6.3 Tunneling

> communication between a client and server is automatically is encrypted, reuqires set up

Secure Shell (SSH)

> SSH was created to use symettic and public key cryptography to communicate across the internet using an encrypted channel

- SSH protocol is used for a variety of tasks in addition to secure remote administration, including file transfer through the simple Secure Copy Protocol (SCP) or as part of the more full-featured Secure File-Transfer Protocol (SFTP)
- clients connects to the server via TCP session
- client and server exchange info such as protcol ver and supported encryption
- client and server initate a secret-key exchange to establish a shared secret session key
- server sends list of acceptable forms of authentication which client will try in sequence

    - If public-key authentication is the selected mechanism,the client sends the server its public key.
    - The server then checks if this key is stored in its list of authorized keys. If so, the server encrypts a challenge using the client’s public key and sends it to the client.
    - The client decrypts the challenge with its private key and responds to the server, proving its identity.
- authentication has been successfully completed, the server lets the client access appropriate resources, such as a command prompt

IPsec

> consists of several protcols, addressing different secuirty needs, operate in one of two modes, transport mode or tunnel mode.

- transport mode, additional IPsec header information is inserted before the data of the original packet, and only the payload of the packet is encrypted or authenticated
- tunnel mode, a new packet is constructed with IPsec header information, and the entire original packet, including its header, is encapsulated as the payload of the new packet. Tunnel mode is commonly used to create virtual private networks (VPNs)
- two parties communcaiting must first set up a set of a security associations(SAs)
- pieces of info that secribe how secure communications are to be conducted between two parties
- contain keys , uni directional,
- Communicating parties store SAs in a security association database (SADB)
- protection for outgoing packets and verifies or decrypts incoming packets by using a security parameter index (SPI) field stored in the IPsec packet header, along with the destination or source IP address, to index into the SADB and perform actions based on the appropriate SA.

Internet Key Exchange(IKE)

> handle negotiations of SAs

1. inital security association is established to encrypt subsequent IKE communications
2. encrypted channel is used to define SAs for the IPsec traffic

Authentication Header (AH)

> used to authenticate the origin and gurantee the data integrity of IPsec packets

Components of the Authentication Header

- secrity parameter index used to identify security assocation associated with packet
- an “authentication data” field that contains an integrity check value (ICV)
- ICV is computed by hashing the entire packet, including the IPsec header

Enscapsulating Security Payload

- H places a header before the payload or original packet, ESP encapsulates its payload by providing a header and a “trailer.” To provide encryption, ESP uses a specified block cipher to encrypt either the entire original IP packet or just its data, depending on whether the tunnel or transport mode is used. ESP also provides optional authentication in the form of an “authentication data” field in the ESP trailer
- doesn't prtotect ip header from tampering allows NAT devices to rewrite scr ip addtess

Virtual Private Networking (VPN)

> allows private net- works to be safely extended over long physical distances by making use of a public network, such as the Internet, as a means of transport

Remote Access VPNs

> allow authorized clients to access a private net- work that is referred to as an intranet. For example, an organization may wish to allow employees access to the company network remotely but make it appear as though they are local to their system and even the Internet itself. To accomplish this, the organization sets up a VPN endpoint, known as a network access server, or NAS. Clients typically install VPN client software on their machines, which handle negotiating a connection to the NAS and facilitating communication.

Site to site VPN

> provide a secure bridge be- tween two or more physically distant networks. Before VPN, organizations wishing to safely bridge their private networks purchased expensive leased lines to directly connect their intranets with cabling.

Different Implementations

- point-to-point tunneling protocol (PPTP) : works by establishing a connection using the peer- to-peer (PPP) link-layer protocol, then encapsulating PPP frames, which are encrypted using Microsoft Point-to-Point Encryption (MPPE), inside IP packets that can be sent across the Internet
- Layer 2 Tunneling Protocol (L2TP) : both header and payload, is encapsulated within a UDP datagram, header and payload, is encapsulated within a UDP datagram. Within the L2TP packet, a number of link-layer protocols can be encapsulated, including PPP and Ethernet. L2TP is commonly used in con- junction with IPsec to ensure authentication, integrity, and confidentiality.

Some Risks in Allowing for VPNs and Tunneling

> payloads of a series of network packets are en- capsulated in a different delivery protocol that might otherwise be blocked by a firewall an information-leakage attack, such as sending company secrets out of a compromised network using HTTP packets, becomes more difficult to detect when protocols relying on tunneling are used

Usable Security

> = Humans + Security Technology think what skills they have and what they don't have

Chapter 6.2 Firewalls

> firewalls can be employed to filter incoming or outgoing traffic based on a predefined set of rules that are called firewall policies

- shiled internal network users from malicious attackers on internet
- censorship
- hardware or software, typically deployed at perimeter of an internal network at point where network connects to internet

Firewall Policies

Packets flowing through a firewall can have one of three outcomes: • *Accepted*: permitted through the firewall • *Dropped*: not allowed through with no indication of failure • *Rejected*: not allowed through, accompanied by an attempt to inform the source that the packet was rejected

Blacklists and Whitelists

- effectively minize vulnerability to the outside world while maintaining the desired functiality for machines in trusted internal network
- network administrators choose a blacklist approach, or default-allow ruleset
  - all packets are allowed through except ones that fit rules fitted in blacklist
  - more flexible in ensuring service not interuppted
- implement a white list or default-deny policy, in which packets are dropped or rejected unless they are specifically allowed by the firewall
  - a network administrator might decide that the only legitimate traffic entering the network is HTTP traffic destined for the web server and that all other in- bound traffic should be dropped
  - configuration requires greater familiarity with protcols used by internal network, provided the greatest possible caution in decding which traffic is acceptable

Stateless and Stateful Firewalls

Stateless Firewalls

> doesn't maintain any remebered context with respect to the packets it is processing, treats each packet attempting to travel through it in isolation without considering pakcets that it has previsouly don't have any memory decidcated to determining if a given packey is aprt of an existing connection Stateless firewalls simply inspect packets and apply rules based on the source and destination IP addresses and ports.

Blocking Undesired Packets

ll traffic from a web server originating at the default port for web servers would be allowed through the firewall to the user’s machine, which may be undesirable

observing that the firewall does not need to allow TCP packets marked with just the SYN flag to reach the user

estriction would prevent outside parties from initiating TCP connections to an internal machine, it would not prevent them from probing the network with other packets not marked with the SYN flag.

Stateful Firewalls

can tell when packets are part of legitimate sessions originating within a trusted network

stateful firewalls maintain tables containing information on each active connection, including the IP addresses, ports, and sequence numbers of packets

these tables, stateful firewalls can solve the problem of only allowing inbound TCP packets that are in response to a connection initiat- ed from within the internal network

allow administrators to apply more restrictive rules to network traffic and create more effective policies for inbound versus outbound traffi

n- age traffic based on the actual contents of packets entering and exiting a network rather than merely considering the origin and destination. This is possible through the use of application-layer firewalls

Chapter 6.4 Intrusion Detection

> intrusion detection system (IDS) is a software or hardware system that is used to detect signs of malicious activity on a network or individual computer IDS sensors which collect real-time data about the functioning of network components and computers, and an IDS manager, which receives reports from sensors.

IDS Manager complies data from sesnors to determine if an intrusion has occured. Based on site policies.

Intrustions

---

masquerader

> n attacker who is falsely using the identity and/or credentials of a legitimate user to gain access to a computer system or network

---

Misfeasor

> a legitimate user who performs actions he is not autho- rized to do

---

Clandestine user

> a user who tries to block or cover up his actions by deleting audit files and/or system logs

---

port scans

> information gathering intended to determine which ports on a host are open for TCP connections

---

Denial-of-service attacks

> network attacks meant to overwhelm a host and shut out legitimate accesses

---

Malware attacks

> eplicating malicious software attacks, such as Trojan horses, computer worms, viruses, etc.

ARP spoofing

> an attempt to redirect IP traffic in a local-area network

DNS cache poisoning

> a pharming attack directed at changing a host’s DNS cache to create a falsified domain-name/IP-address association

---

Intrusion Detection Techniques

> NIDS sits at perimeter of a network and detects malicious behaviour based on traffic patterns and content PIDS tailored towards detecting malicious behaviours in a specific protcol, deployed on a particular network host HIDS resides on a single system and moniters activity on that machine, including system calls, interprocess communication and patterns in resource usage

Network IDs usually workby performing deep packet inspection on incoming and outgoing traffic & applying a set of attack signatures or heuristics to determine whether traffic patterns indicate malicious behavior.

Host IDs Monitoring auit files and system logs to detect masquerding and misfeasant users who attempt unauthorized actions and clandestine users who try to deleye modify system monitoring,

use heuristic rules or statistical analysis to detect when a user is deviating from “normal” behavior, which could indicate that this user is a masquerading user. Misfeasant users can be detected by a system that has rules defining authorized and unauthorized actions for each user. Finally, clandestine users can be detected by monitoring and logging how changes are made to audit files and system logs themselves.

Passive IDs log potenitally malicious events and alert network administrator so action can be taken. Don't take any preemptive actions on their own. more sophisticated reactive systems, known as intrusion prevention systems (IPS) work in conjunction with firewalls and other network devices to mitigate the malicious activity.

An IDS Attack

> attempt to launch a denial of service attack on the IDS itself, triggering a high number of intrusion alerts, attacker may overwhlem an IDS to point that it cannot log every event, make it difficult to identify which logged event represents an actual attack & which were used as a diversion. Advanced techniques to evade detecction force IDS developers to employ sophisticated heuristics and signature scheems based on stare of art machine ....

Intursion Detection Events

- *False positive* : when an alarm is sounded on benign activity, which is not an intrusion
- *False negative* : when an alarm is not sounded on a malicious event, which is an intrusion
- *True positive:* : when an alarm is sounded on a malicious event, which is an intrusion
- *True negative:* : when an alarm is not sounded on benign activity, which is not an intrusion

The Base Rate Fallacy

> effectivness of some IDSs can be misinterpreted due to statistcal error known as base rate fallacy, type of error occurs when probability of some coniditonal event is assessed without considering base rate of event

IDS Data Collection and Audit Records

> Inout is a stream of records that identifies elementary actions for a network or host

Event records include: • Subject: the initiator of an action on the target • Object: the resource being targeted, such as a file, command, device, or network protocol • Action: the operation being performed by the subject towards the object • Exception-condition: any error message or exception condition that was raised by this action • Resource-usage: quantitative items that were expended by the system performing or responding to this action • Time-stamp: a unique identifier for the moment in time when this action was initiated

Rule-Based Intrusion Detection

> identify events that should trigger alarms is to use rules

- rules identify types of actions which match such attacks, rule would encode a signature for such an attack
- potential for annoying false-positive alarms is low, because the policy makers themselves have determined the list of rules
- signature- based schemes are fundamentally limited in that they require the IDS to have a signature for each type of attack

Statistical Intrusion Detection

• Count: the number of occurrences of a certain type of action in the given time range • Average:theaveragenumberofoccurrencesofacertaintypeofaction in a given of time ranges • Percentage: the percent of a resource that a certain type of action takes over a given time range • Metering: aggregates or average-of-averages accumulated over a rel- atively long period of time • Time-interval length: the amount of time that passes between in- stances of an action of a certain type

- works out typical profile for each user or host
- Once a user profile is in place, the IDS manager can determine thresholds for anomalous behaviors and then sound an alarm any time a user or host deviates significantly from the stored profile for that person or machine
- tatistical IDSs rely on analyzing patterns in network traffic, it would be difficult for an attacker to hide his behavior from an IDS manager using such techniques
- nonmaliciuos behaviour could generate a significant anomaly.
- sensitivity to normal changes in system or user behavior therefore leads to false positives
- stealthy attacker may not generate a lot of traffic and thereby might go unnoticed by a statistical network IDS, leading to false negatives

Port Scanning

> which traffic is permitted through a firewall and which ports on a target machine are running remote services is a crucial step in an- alyzing a network for security weaknesses

Ports can be :

- open
- closed
- blocker

TCP Scans

> perform scan attempts to initate TCP connection on each of ports on a target machine,

SYN Scans

> performing scan issues a low level TCP packet market with SYN flag for each port on target machine port open, service listening on port will return a packet marked with SYN-ACK flag

Idle Scanning

> relies on finding third party machine known as "zombie" has predicable TCP sequence numbers, attacker uses zombie’s weak TCP imple- mentation as a tool to perform a port scan on a separate target

- attacker sends a SYN-ACK TCP packet, to zombie, packet was unpromted by the zombie,
- reply to attacker with RST packet containing a sequence number, attacker sends a SYN packet to target he wishes to scan, spoofs src IP address
- scanned port is open, target will reply to zombie with "SYN-ACK packet"
- replies to target with a RST packet and icrements sequence num again,
- If it has been incremented, then the chosen port on the target is open, and if not, the port is either closed or blocked

UDP Scans

> connectionless protcol, fewer cues from which to gather infomation, Most UDP port scans simply send a UDP packet to the specified port. If the port is closed, the target will usually send an ICMP “destination unreachable” packet

- scan is not very reliable, however, because open ports and ports blocked by a firewall will both result in no response
- improve the reliability of the response, many port scanners choose to query UDP ports using UDP packets containing the payloads for appropriate applications

Port Scan Security Concerns

- type and version of each remote service and the operating system version may be valuable in planning an attack
- port scanners may exploit the fact that each operating system has slight differences in its TCP/IP stack implementation and, as such, might respond differently to various requests or probes
- versions of remote services may have subtle differences in the way they respond to certain requests, and knowledge of these differences may allow port scanners to determine the specific service running
- known as *fingerprinting*

Honey Pots

> placed on networks in a way that makes it attractive such as having it configured with software with known vulnerabilities and having its hard drive full of documents that appear to contain company secrets or other apparently valuable information

• Intrusion detection. Since attempts to connect to a honeypot would not come from legitimate users, any connections to a honeypot can be safely identified as intrusions. Based on the way in which such con- nections are initiated, an intrusion detection system can be updated with the latest attack signatures. • Evidence. Appealing documents on a honeypot computer encourage an intruder to linger and leave evidence that can possibly lead to the identification of the intruder and/or his location. • Diversion. A honeypot also may appear to be more attractive to potential intruders than legitimate machines, distracting intruders from sensitive information and services.

Crptographic Protocols

> Programs relying on cryptographic primitives and whose goal is the establishment of “secure” communications.

Logical Attacks

> communtative symmertric encryption $m_{k_1\,k_2} = m_{k_2\,k_1}$ attacker can choose any key to encrypt with since no authentication

Needham–Schroeder Public Key (NSPK)

- Authentication: if Alice has completed the protocol, apparently with Bob, then Bob must also have completed the protocol with Alice.
- Authentication: If Bob has completed the protocol, apparently with Alice, then Alice must have completed the protocol with Bob.
- Confidentiality: Messages sent encrypted with the agreed key (k ← h(NA, NB)) remain secret.
- remains secret, alice sends her contribution to Bob with her identity and bob's public key
- he creates a N_B and sends it with N_A enxrypted with Alice's public key
- she sents it back with his public key to say she got it , establish a session key

Attack:

- man in the middle attack
- have someone in the middle, alice is encrypting with attacker's public key
-

Forward Secrecy

> A protocol ensures forward secrecy, if even if long-term keys are compromised, past sessions of the protocol are still kept confidential, and this even if an attacker actively interferred.

Station to station(STS procol)

- establish diffe helman key
- second add authentication, add his signature

The Basic Access Control (BAC) protcol

> Electronic Passport RFID Tag : information printed on passport, a JPEG copy of the picture
> Gives the secret key and MAC key of the address

- Reader get challenge
- Passport picks two (K_p) random number 64 bit numbers , sends N_p to reader
- reader generates N_r, encrypt this with key of passport and encrypt with MAC key
- passport encrypts K_p with long term symmetric key and MAC for integerity
- SEssion key is $K_{seed} = K_p \oplus K_r$
- checks MACS and NCERS

  > passport must reply to all messages

  - Unlinkability: ensures a user may make mutliple uses of a resource or service without other users being able to link these uses together
  - Anoymity: ensures that a user may use of a resource of service without disclosing the user's identity
  - Attacker could work out what is wrong if error type is specificed

Anymous Communication

> Routing information can reveal who you are

-IP address leaves behind digital tracks that can be used to identify you and invade your privacy

3DC Protcol

- Each cryptographer flips a coin and shows it to his left neighbor
- Each cryptographer announces whether the two coins he saw are the same. If he is the payer, he lies
- odd number of “same” ⇒ the NSA paid , even number of “same” ⇒ one of the cryptographers paid

Limitations:

- Requires pair-wise shared secret keys (secure channels) between the participants (to share random bits)
- Requires large amounts of randomness

Crowds

> a crowd is a group of m users; c out of m users may be corrupted

Initator wants to request a webpage creates a path between him and the server

- the initiator selects a forwarder from the crowd and sends him his request
- a forwarder delivers the request directly to the server with probability 1 – pf ; he forwards the request to a randomly selected new forwarder from the crowd with probability pf ; the new forwarder repeats the procedure
- the response from the server follows same route in opposite direction

Chaum's Mix

> message padding and buffering to avoid time correlation attacks dummy messages are generated by the mixes themselves to prevent an attacker sending n – 1 messages to a mix with capacity n, allowing him to then link the sender of the n^{th} message with its recipient

- messages are sent through a sequence of mixes
- some of the mixes may be corrupted
- can be anoymous depending on whole network

Limitations :

- Asymmetric encryption is not efficitent
- Dummy messages are innefficient
- Buffering is not efficient

Onion Routing

> combine advantages of mixes and proxies

- use public-key crypto only to establish circuit
- use symmetric-key crypto to exchange data
- distribute trust like mixes

But does not defend against attackers that control the hole network

TOR Circuit

> Tor directs Internet traffic through a free, worldwide, volunteer overlay network consisting of more than seven thousand relays

- TOR anonymises the origin of the traffic
- TOR encrypts everything inside the TOR network
- but TOR DOES NOT encrypt all traffic through the Internet
- for confidentiality you still need to use end-to-end encryption such as SSL/TLS

If DNS resolution if handled by the client browser defeats the purpose of using TOR

- relays can listed on TOR directory
- local ISP can observe that you are communicating with TOR nodes

LImitations:

- TOR does not provide protection against end-to-end timing attacks
- If the attacker can see both ends of the communication channel, he can correlate volume and timing information on the two sides

Chapter 3.1 Operating Systems Concepts

> provides interface between users of a computer and computer hardware manages ways applications access resources in a computer including disk drives, CPU, main memory, input devices, output devices and network interfaces

---

multitasking multiple application programs to run at the same time

---

The kernel and Input/Output

> Kernel hadnles managment of low-level hardware resources, including memory, processors, input,output.

User Applications <-> Operating System(OS Kernel and Non - im os applications) <-> CPU, Memory, Input and Output(Hardware)

Input/ Output Devices

- device is using device driver, encapsulates the details of how interaction with that device should be done
- application programmer interface (API) device drivers present to applications, allow programs to interact with devices

System Calls

> syscall - allow user applications to delegate tasks to kernel to perform tasks called software interrupts : requests by application for processor to stop current flow of execution and switch to handler for interrupt

- trap switching mode to kernel

Processes

> instance of program that is currently executing

- actual contents of programs are in Persistent storage such as a harddrive , to be executed program should be loaded in RAM
- kernel manages all running processes giving each a fair share of CPU
- *Time slicing* -> os gives each running process a tiny bit of time to do something and moves on to the next process

Users and Process Tree

- user creates a new process, kernel see this as an existing process asking to create a new process, which is called *forking*
- *parent process* one is being forked is the *child process*
- most process new child process inherits permissions of its parents unless parent decideds to lower it on purpose
- processes are in a tree, root in linux known as init

Process IDs

> process running on a given computer is identified by a unique non- negative integer, called the process ID (PID), root in linux has 0

Process Privileges

operating system associates info about user on whose behalf the process is being executed with each process, Unix-based systems have an ID system where each process has a user ID (uid), which identifies the user associated with this process, as well as a group ID (gid), which identifies a group of users for this process

Inter-Process Communication

- communicate is to pass messages by reading and writing files
- allows for processes to communicate with each other is to have them share the same region of physical memory.
- *pipes and sockets* act as tunnels from one process to another

Signals

> processes can send direct messages to each other asynchronously

- process receives a signal, os interrupts current flow of execution of that process and checks whether that process has an apporopriate signal handler, signal handler exists, routine is executed, process does not handle this particular signal, then it takes a default action, Terminating a nonresponsive process on a Unix system is typically performed via sig- nals.

Remote Procedure Calls

> allows a process to call a subroutine from another process program

Daemons and Services

> process which runs without any user intervention

- started by init process and have different permissions, Common examples of daemons are processes that control web servers, remote logins, and print servers.

The Filesystem

> external, nonvolatile memory of the computer is organized

File Access Control

> how to delineate which users can access which resources, that is, who can read files, write data, and execute programs, file permissions

File permissions

- checked by the operating system to determine if a file is readable, writable, or executable by a user or group of users
- stored in metadata
- process attempts to access file, os checks identity of process and determines wheather good or not.
- file permission matrix: who is allowed to do what with the file.
- owner class : which determines permissions for the creator of the file
- group class : which determines permissions for users in the same group as the file
- other : class determines permissions for users who are neither the owner of the file nor in the same group as the file.
- read bit

- write bit
- execute bit

Unix File Permissions

- binary , different weights
    - read bit 4
    - write bit 2
    - execute bit 1
- change using chmod
- Unix-based systems employ a path-based approach for file access control. The operating system keeps track of the user's current working directory. Access to a file or directory is requested by providing a path to it, which starts either at the root directory

Memory Management

> organization and allocation of the memory in a computer.

- process executes, it is allocated a region of memory known as its address space, stores program code, data and storage that process needs during its execution, address space is organised:

1. Text. This segment contains the actual machine code of the program, which was compiled from source code prior to execution.
2. Data. This segment contains static program variables that have been initialized in the source code, prior to execution.
3. BSS. This segment, which is named for an antiquated acronym for block started by symbol, contains static variables that are uninitial- ized (or initialized to zero)
4. Heap. This segment, which is also known as the dynamic segment, stores data generated during the execution of a process, such as objects created dynamically in an object-oriented program written in Java or C++.
5. Stack. This segment houses a stack data structure that grows down- wards and is used for keeping track of the call structure of subroutines (e.g., methods in Java and functions in C) and their arguments.

Memory Access Permissions

- processes are not allowed to access the address space of other processes, unless they have explicitly requested to share some of that address space with each other
- Unix memory model, operating systems divide the address space into two broad regions: user space, where all user-level applications run, and kernel space, which is a special area reserved for core operating system functionality.

Contigunous Address Spaces

- Arrays are indexed as contiguous memory blocks

Virtual Memory

computer architectures incorporate a system of virtual memory, where each process receives a virtual address space, and each virtual address is mapped to an address in real memory by the virtual memory system

a virtual address is accessed, a hardware component known as the memory management unit looks up the real address that it is mapped to and facilitates acces

that they allow for the total size of the address spaces of executing processes to be larger than the actual main memory of the computer

Page Faults

> slight time trade-off for benefit we get from virtual memory, however, since accessing the hard drive is much slower than RAM

- block of the address space is not accessed for an extended period of time, may be paged out
- process attempts to access a virtual address that resides in a paged out block, it triggers a page fault
- a page fault occurs, another portion of the virtual memory system known as the paging supervisor finds the desired memory block on the hard drive, reads it back into RAM, updates the mapping between the physical and virtual addresses, and possibly pages out a different unused memory block

Virtual Machines

- software that creates a simulated environment the operating system can interact with
- software layer that provides this environment is known as a hypervisor or virtual machine monitor (VMM)
- system running inside the VM is known as a guest, and the native operating system is known as the host

Implementing Virtual Machines

- Emulation

where the host operating system simulates virtual interfaces that the guest oper- ating system interacts with. Communications through these interfaces are translated on the host system and eventually passed to the hardware. The benefit of emulation is that it allows more hardware flexibility. ownside of emulation is that it typically has decreased performance due to the conversion process associated with the communication between the virtual and real hardware

- Virtualization

removes above conversion, the virtual interfaces within the VM must be matched with the actual hardware on the host machine, so communications are passed from one to the other seamlessly. This reduces the possibilities for running exotic guest operating systems, but results in a significant performance boost.

Advans:

- *Hardware Efficiency* : system admibs to host multi os on same machine ensuring efficient allocation of hardware resources
- *Portability* : to run a program on multiple different machines
- *Security*: maximizing available resources and provid- ing portable computing solutions, virtual machines provide several benefits from a security standpoint.
- *Management Convenience* : ability to take snapshots of the entire virtual machine state can prove very convenient

Chapter 3.4 Application Program Security

Compiling and Linking

> machine code instructions that a processor can execute is known as compiling

Simple Buffer Overflow Attacks

> allocates a fixed-size buffer in memory in which to store information, care must be taken to ensure that copying user-supplied data to this buffer is done securely and with boundary checks

Arithmetic Overflow

- if a program continually adds very large numbers and eventually exceeds the maximum value for a signed integer, 0x7fffffff, the representation of the sum overflows and becomes negative rather than positive
- if a program adds many negative numbers, eventually the sum will underflow and become positive
- the highest number is reached, the next sequential integer wraps around to zero.

Stack-Based Buffer Overflow

> exploits the special structure of the memory stack

- n attacker provides input that the program blindly copies to a buffer that is smaller than the input.

- an attacker could overwrite local vari- ables adjacent in memory to the buffer, which could result in unexpected behavior.
- a stack smashing attack, the attacker exploits a stack buffer vulnerability to inject malicious code into the stack and overwrite the return address of the current routine so that when it terminates, execution is passed to the attacker's malicious code instead of the calling routine.

Seizing Control of Execution

> or the attacker is to guess the location of the return address with respect to the buffer and to determine what address to use for overwriting the return address so that execution is passed to the attacker's code.

- processes cannot access the address spaces of other processes, so the malicious code must reside within the address space of the exploited process,
- the address space of a given process is unpredictable and may change when a program is run on different machines

NOP Sledding

> CPU instruction that does not actually do anything except tell the processor to proceed to the next instruction

Before copying : Buffer -> other program data -> Return address

After Copying: JUnk Padding -> Guessed Address of Shellcode -> NOPs -> Shellcode

Trampolining

> jump-to- register or trampolining, is considered more precise

The return to libc attack

> e attacker can determine the address of a C library function within a vulnerable process's address space, such as system() or execv, this information can be used to force the program to call this function.

Shellcode

> the ability to execute arbitrary code on the machine malicious code included in an exploit is often known as shellcode, written in assembly language

Preventing Stack-Based BUffer Overflow Attacks

- check that you are not overwriting address
- Canary : a value that is placed between a buffer and control data (which plays a similar role to a canary in a coal mine). The system regularly checks the integrity of this canary value, and if it has been changed, it knows that the buffer has been overflowed
- address space layout randomization (ASLR) , rearranges the data of a process's address space at random, making it extremely difficult to predict where to jump in order to execute code.

Heap-Based Buffer Overflow Attacks

> power to allocate memory dynamically and have it persist across multiple function calls. This memory is allocated in a large portion of unused memory known as the heap.

- if don't deallocate, can cause a memory leak
- caused overflows
- can overwrite portions of the next block

Preventing Heap-Based Buffer Overflow Attacks

- ns. Address space randomization prevents the attacker from reliably guessing memory locations, making the attack more difficult.
- some portions not executable
- store heap metadata in seperate location

Format String attacks

> allow an attacker to seize control and execute arbitrary code in the context of the program by overwriting a return address, function pointer, etc

Race Conditions

> any situation where the behavior of the program is unintentionally dependent on the timing of certain events.

The Time of Check/Time of Use Problem

> attacker could exploit this small delay by changing the file in question between the two calls two operations are performed atomically, that is, they should be performed as a single uninterruptible operation

Chapter 3.3.2 Password-Based Authentication

> authentication username and passsword

- typi- cally keep cryptographic one-way hashes of the passwords in a password file or database instead
- guessing passwords from the password file is to conduct a dictionary attack,

Password Salt

> salt would be introduced by associating a random number with each userid

How Salt Increases Search Space Size

> password salt significantly increases the search space needed for a dictionary attack Search space is $2^B \times D$ where B is bits of salts,D size of list of words

Password Authentication in Windows and Unix-based Systems

Windows systems, password hashes are stored in a file called the Security Accounts Manager (SAM) file, which is not accessible to regular users while the operating system is running

Chapter 7 Web Security

The World Wide Web

HTTP And HTML

- process begins with the browser determining the IP address of the web server that is hosting the web site of interest.

Uniform Resource Locators (URLs)

> in HTML

Connecting to a Web Server

> http used for retreiving requested web page browser checks local DNS cache, if not there queries a server to resolve IP address, makes a TCP connection port

- 21 : FTP

- 80 : HTTP
- 443 : HTTPS

HTTP Request

- establish tcp , broswer sends http requests in a data portion of tcp packet, request line(GET or POST),Headers section, identifies additonal info

HTML Forms

> provide input to a website in form of variables,

- users submit a form using GET variables, the name-value pairs for the variables are encoded directly into the URL, separated by &

Lack of Confidentiality in HTTP

- does not encrypt data, attacker could intercept the packets being sent between a web site
- could get a man in the middle attack

HTTPS

HTTPS (hypertext transfer protocol over secure socket layer)

- identical to HTTP but either has SSL (Secure Socket Layer), TLS(transport layer security)
- reley on certicate to verify identity of server , browser sends a list of cryptographic ciphers and hash functions supported by both, picks one, sends back certificate which contains public key of server. Broswer verifies authenticity of certifcation, geenrate secret keys

Web server Certificates

> verifying the identity of the requester and ownership of the domain name for the website, the CA signs and issues the certificate, which the web server then sends to browsers to provide proof of its identity SSL server certificate,

• Name of the CA that issued the certificate • Serial number, unique among all certificates issued by the CA • Expiration date of the certificate • Domain name of the web site • Organization operating the web site and its location • Identifierofthepublic-keycryptosystemusedbythewebserver(e.g., 1,024-bit RSA) • Public key used by the web server in the HTTPS protocol • Identifier of the cryptographic hash function and public-key cryp- tosystem used by the CA to sign the certificate (e.g., SHA-256 and 2,048-bit RSA) • Digital signature over all the other fields of the certificate

Extended Validation Certifcates

> confirmation that the domain on the certificate being signed is in fact owned by the certificate requester new class of certificates can only be issued by CAs who pass an audit demonstrating that they adhere to strict criteria for how they confirm the subject's identity

Certificate Hierarchy

> top-level certificate is known as the root certificate root certificate is known as a self- signed certificate, where the issuer is the same as the subject the highest authority within an organization, are referred to as anchor points in the chain of trust used to verify a certificate

Trustworthiness and Usability Issues for CErtificates

- certificate revocation list
  - private key compromise or change of organization operating the web site

Dynamic Content

> pages featuring dynamic content can change in response to user interaction or other conditions, such as the passage of time.

Document Object Model (DOM)

> representing the content of a web page in an organized way

- makes a truee

Javascript

> handles events

Sessions and Cookies

> encapsulates information about a visitor that persists beyond the loading of a single page

- passing session information via GET or POST variables using cookies

concept of a session is a class of attacks known as *session hijacking*

SEssions using GET or POST

- user's session information and inserts it into the page being delivered to the client using the mechanism of hidden fields

- Each time the user navigates to a new page, this code passes the user's session information to the server allowing it to "remember" the user's state.
- method is particularly susceptible to man-in- the-middle attacks

Cookies

> sent to the client by the web server and stored on the client's machine, the user revisits the web site, these cookies are returned, unchanged, to the server, which can then "remember" that user and access their session information

Cookie Properties and Components

> cookie defaults to being deleted when the user exits the browser.

- Only hosts within a domain can set a cookie for that domain. A subdomain can set a cookie for a higher-level domain, but not vice versa
- the cookie can only be accessed within a specific subdirectory of the web site, and defaults to the root directory of a given domain.
- cookies are transmitted unencrypted using HTTP, and as such are subject to the same man-in-the-middle attacks
- Cookies can set an HTTP-Only flag, scripting language, Finally, cookies can set an HTTP-Only flag. If enabled, scripting lan- guages are prevented from accessing or manipulating cookies stored on the client's machine
- preventing scripting languages from accessing cookies signif- icantly mitigates the risk of cross-site scripting (XSS) attacks

How Cookies Support Sessions

- client automatically includes any cookies set for a particular domain and path in the Cookie field of any HTTP request header being sent to that server.

Security Concerns for Cookies

- dangerous to store any sensitive information unencrypted in the body of a cookie, since cookies can typically be accessed by users of the system on which they are stored

Server-side sessions

- servers typically use a session ID or session token
- server then employs one of the two previous methods, o store this token on the client side
- session ID should be hard to guess by an attacker.

Attacks on Clients

Session Hijacking

> attacker can take over a TCP session in an attack called session hijacking

attacker intercept communication between a web client and web server, but also requires that the attacker impersonate whatever measures are being used to maintain that HTTP session

Defenses Against HTTP Session Hijacking

- protect against packet sniffers and TCP session hijacking.
- to encrypt such session tokens
- server-side session IDs should be created in ways that are difficult to predict, for instance, by using pseudo-random numbers.
- replay attacks, which are attacks based on reusing old credentials to per- form false authentications or authorizations.
- erver can protect against such attacks by incorporating random numbers into client-side tokens, as well as server-side tokens, and also by changing session tokens frequently
- a session token with the IP addresses of the client so that a session token is considered valid only when connecting from the same IP addres

Trade Offs

- little long term risk on client end
- server-side sessions are terminated when the client closes the browser.
- server- side session techniques that use random session tokens that are frequently changed can result in a reduced risk for HTTP session hijacking on the user's end

Phishing

> attacker creates a dummy web site that appears to be identical to a legitimate web site in order to trick users into divulging private information.

URL OBfuscation

> disguise the URL of the fake site

- own as the Unicode attack, more formally known as a homeograph attack. Unicode characters from international alphabets may be used in URLs in order to support sites with domain names in multiple languages, so it is possible for phishers to register domain names that are very similar to existing legitimate sites by using these international characters.

Click Jacking

> click-jacking is a form of web site exploitation where a user's mouse click on a page is used in a way that was not intended by the user.

Other Actions that can be Click Jacked

- click-jacking might be used is ad- vertisement fraud.
- Click- jacking can be used to force users to unwillingly click on advertisements, raising the fraudulent site's revenue, which is an attack known as click fraud.

Vulnerabilities in Media Content

The Sandbox

> refers to the restricted privi- leges of an application or script that is running inside another application. Different scripting languages and media applications are granted vary- ing access to different components inside most web browsers.

Media Content and Adobe Flash

- embedded media player used by a web browser to play this content has application-level flaws, malicious media files may be created to escape the sandbox of the victim's browser and execute code on the victim's machine.

Java Applets

- sandbox restrictions significantly mitigate the risk of dangerous behavior by Java applets
- developer of Java applets can obtain a code signing certificate from a CA and create signed applets with the corresponding private key
- signed applet requests to operate outside of the sandbox, it presents the certificate to the user, who, after verifying the validity of the certificate and the integrity of the applet code

ActiveX Controls

- ActiveX controls are granted access to all system resources outside of the browser.
- ActiveX controls can effectively be used as a vector for malware
- a digital signature scheme is used to certify the author of ActiveX controls

Privacy Attacks

Third Party and Tracking Cookies

- cookies are used by advertisers to track users across multiple web sites and gather usage statistics

Protecting Privacy

- specify policies regulating how long cookies are stored and whether or not third-party cookies are allowed
- to protect a user’s anonymity on the Web, proxy servers can be used.
- web browsers have a “private browsing” mode, which can be entered using a single command, preventing the storage of any cookies and the recording of any browsing history while in this mode

Cross– Site Scripting(XSS)

> improper input val- idation on a web site allows malicious users to inject code into the web site, which later is executed in a visitor’s browser

Persistent XSS

> code that the attacker injects into the web site remains on the site for a period of time and is visible to other users

- Javascript has the ability to redirect visitors to arbitrary pages, so this is one possible avenue for attack. Malicious users could simply inject a short script that redirects all viewers to a new page that attempts to download viruses or other malware to their systems

Nonpersistent XSS

> most real-life examples of cross-site scripting do not allow the injected code to persist past the attacker’s session

Defenses againsnt XSS

- users to set restrictive policies on when scripts may be executed
- users choose to eliminate all scripts except for specific sites on a white list. Others allow scripts on all sites except for those listed on a public blacklist.
- XSS scanner might prevent execution of any script lines that attempt to append a cookie directly to the end of a URL, because this code might indicate an XSS attack

Other XSS Attacks

- plagued by these worms, since the ability to communicate with other users is built into the function- ality of the site, and is therefore accessible by Javascript
- social networking site would execute some payload, and then automatically send itself to friends of the victim, at which point it would repeat the process and continue to propagate.

Cross Site request Forgery

> opposite of cross-site scripting

- While XSS exploits a user's trust of a specific web site, CSRF exploits a web site's trust of a specific user. In a CSRF attack, a malicious web site causes a user to unknowingly execute commands on a third-party site that trusts that user

Login attack : a malicious web site issues cross- site requests on behalf of the user, but instead of authenticating to the victim site as the user, the requests authenticate the user as the attacker

Defenses Against Client-Side Attacks

Mitigation of these attacks by the user can be facilitated with two primary methods: • Safe-browsing practices • Built-in browser security measures

Safe Browsing Practices

- https
- the legitimacy of the site should be confirmed by examining the URL and ensuring that there are no certificate errors

Bulit in Browser Security Measures

web sites are placed in the Internet Zone. Users can then delegate sites to Trusted and Restricted zones, Each zone has its own set of security policies, allowing the user to have fine-grained control depending on whether or not they trust a particular web site

Attacks on SErvers

Server side scripting

useful to utilize code on the server side that is executed before HTML is delivered to the user

allow servers to perform actions such as accessing databases and modifying the content of a site based on user input or personal browser settings

executed on the server, and because of this only the result of this code's execution, not the source, is visible to the client

PHP

> PHP is a hypertext pre- processing language that allows web servers to use scripts to dynamically create HTML files on-the-fly for users, based on any number of factors, such as time of day, user-provided inputs, or database queries

Server-Side Script Inclusion Vulnerabilities

> web security vulnerability at a web server is exploited to allow an attacker to inject arbitrary scripting code into the server, which then executes this code to perform an action desired by the attacker.

Remote- File Inclusion

desirable for server-side code to execute code contained in files other than the one that is currently being run

n attack is known as a remote-file inclusion (RFI) attack. code an attacker might execute in such an attack is a web shell, which is a remote command station that allows an attacker to navigate to the web server and possibly view, edit, upload, or delete files on web sites that this web server is hosting.

Local-File Inclusion (LFI)

> ocal-file inclusion (LFI) attack causes a server to execute injected code it would not have otherwise performed (usually for a malicious purpose). The difference in an LFI attack, however, is that the executed code is not contained on a remote server, but on the victim server itself. locality may allow an attacker access to private information by means of bypassing authentication mechanisms FI attacks can allow an attacker to access files on the web server's system, outside of the root web directory.

Databases and SQL Injection Attacks

> database is a system that stores information in an organized way and produces reports about that information based on queries presented by users.

SQL Structured Query Language

• SELECT: to express queries • INSERT: to create new records • UPDATE: to alter existing data • DELETE: to delete existing records • Conditional statements using WHERE, and basic boolean operations such as AND and OR: to identify records based on certain conditions • UNION: to combine the results of multiple queries into a single result

queries to query database

SQL Injection Attack

> SQL injection allows an attacker to access, or even modify, arbitrary information from a database by inserting his own SQL commands in a data stream that is passed to the database by a web server. The vulnerability is typically due to a lack of input validation on the server's part.

Unintended Information Disclosure

constructing the query to the database, the server-side code does not check to see whether the GET variable, id, is a valid input, that is, that it is in proper format and is referring to an id value that actually exists.

BYpassing Aunthentication

he server creates an SQL query using the POST variables,umber of rows returned by this query is greater than zero (that is, there is an entry in the users table that matches the entered username and password, access is granted.

Other SQL Injection Attacks

> allow for inserting new records, modifying existing records, deleting records, or even deleting entire tables

to access information from a database even when the results of a vulnerable database query are not printed to the screen., using mutliple injected queries, known as blind SQL Injection Attack

sert malicious code into the database that could at some point be sent to users' browsers and executed

SQL injection worm. orms propagate automatically by using the resources of a compromised server to scan the Internet for other sites vulnerable to SQL Injection

Preventing SQL Injection

- strip dangerous characters

Denial of Service Attacks

> single point of failure : major web site uses a single web server to host the site

over- load a web server with so many HTTP requests that the server is unable to answer legitimate requests

multiple web servers for an important web site can also serve as protection

Web SErver Privileges

he general principle of least privilege, the web server application should be run under an account with the lowest privileges possible an attacker may first compromise the web server, and then exploit weaknesses in the operating system of the server or other programs on the machine to elevate his privileges to eventually attain root access

Defenses Against Server-Side Attacks

Developers

- development practices is the principle of input validation

Admin

- least privilege
- web servers should be granted read privileges only to the directories in the web site's root directory, write privileges only to files and directories that absolutely need to be written to
- group policy, which is a set of rules that applies to groups of users
- apply security updates and patches as soon as they are released