Computer Security

2019 Exam 3 Review

Paul Krzyzanowski

Rutgers University

Fall 2019

Part 1: Bitcoin

Question 1

version B: 2

A user's *address* in bitcoin is:

(a) A hash of their private key.

(b) Their X.509 digital certificate.

(c) A hash of their public key.

(d) The host IP address of the system that stores their wallet.

- In the first versions of Bitcoin, the user's address was their public key. Now it's a hash the public key (rendered printable and with error-checking characters).
- The address can only be used as a destination of a transaction – you send money to an address
  - You can receive that money in a transaction by providing your public key and signing the transaction with your private key.
  - Anyone can validate your signature and know that only someone with the private key that corresponds to the public key in the transaction could have created it.
  - Anyone can also re-generate your address from your public key to validate that the referenced transactions are really sending money to you.

Question 2

version B: 3

The *change* field in a bitcoin transaction:

(a) Provides a way for the receiver to return any excess money to the payer.

(b) Pays the bitcoin network for processing the transaction.

(c) Directs excess money in referenced past transactions back to the payer.

(d) Enables the bitcoin network to return money if the payer overpaid for processing the transaction.

- A transaction references one or more past transactions as inputs - this identifies where money comes from
- All past transactions must be completely spent: those transactions should never be referenced again by any other transaction
- The change field identifies the transaction owner's address and contains [Σ(inputs) - output]. This is the leftover money that goes back to the transaction creator.

Question 3

version B: 4

The purpose of *proof of work* in bitcoin is to:

(a) Provide a bitcoin reward for nodes to participate in the bitcoin network.

(b) Achieve agreement on the next block in the blockchain.

(c) Validate past transactions to ensure there is no double spending.

(d) Generate a tamper-proof signature for a bitcoin transaction.

The proof of work was created as a consensus mechanism so that in a network of thousands of bitcoin nodes, all receiving transactions, there will be only a tiny chance that multiple nodes will propose a valid new block for the chain at the same time.

(a) The reward is an *incentive* but not the purpose. The reward periodically halves and will eventually reach 0. After that, miners will only make money from transaction fees that accompany each transaction.

(c) Every node that receives a transaction must validate it to ensure there is no double spending before adding it to a proposed block of transactions

(d) The proof of work makes the blockchain tamper-proof since any modification to the block will cause the hash pointers to not have the desired property. However, it is NOT a signature since there is no use of either a signing key (private key) for creation or a public key for verification.

Note: You received partial credit of 2 points for (d)

Question 4

version B: 1

A bitcoin transaction is confirmed when:
(a) All the nodes in the bitcoin network acknowledge receiving it.
(b) A majority of nodes in the bitcoin network acknowledged it.
(c) The transaction is put into a block and added to the blockchain.
(d) A certain number of blocks have been added to the chain after the transaction.

It is expected that occasionally there will be competing blockchains. Some of these may have been created legitimately but some may be malicious. Eventually, one of these will be the highest scoring one and will be accepted as the correct version of the blockchain. Transactions in the competing chains that are not in the correct one will simply be lost.

A malicious attacker who tries to modify past transactions will have to re-compute the proof-of-work in each newer block and keep up with new transactions to have the longest chain.

The more blocks are added after a transaction, the more computational effort will be needed to modify that transaction and still have the longest chain.

Part 2: Network security

Question 5

version B: 6

An attacker can see all traffic on an Ethernet local area network by:
(a) Masquerading as a cascaded (connected) switch.
(b) Sending false responses to ARP (address resolution protocol) queries.
(c) Configuring the operating system to listen on all ethernet MAC addresses.
(d) Sending a lot of Ethernet frames with random source addresses to the switch.

A switch looks at the source address to build up a switch table that associates an address with a switch port. By flooding a switch with thousands of different addresses, you can cause a **CAM overflow** that will flush out earlier entries and cause legitimate incoming ethernet packets to be sent out on all switch ports, turning the switch into a hub.

(a, b): You won't capture all the traffic from the switch.

(c) You can configure an ethernet transceiver to accept any address but that will not change the behavior of the switch.

Question 6 version B: 7

An *ARP cache poisoning* attack enables an attacker to:

(a) Install malware onto the attacked computer.
(b) Snoop on all Ethernet traffic originating from the attacked computer.
(c) Redirect traffic on the LAN that is targeted to a specific IP address.
(d) Snoop on all Ethernet traffic on the LAN.

- ARP, the Address Resolution Protocol, is a way for a computer to find the ethernet MAC address that corresponds to a given IP address
- A cache poisoning attack is when an attacker sends an ARP reply message with the wrong MAC address
  - ARP responses don't need to be associated with requests. Systems sometimes send *gratuitous ARP* messages when they start up.
- This allows an attacker to get IP packets that are destined to another system.

Question 7

version B: 8

A problem with the *Dynamic Host Configuration Protocol* (DCHP) is:
(a) A stream of requests can lead to a denial of service attack.
(b) A man-in-the-middle attack can modify the DHCP response.
(c) A system has no way of knowing whose reply is legitimate.
(d) It is too time consuming to validate signatures in DHCP responses.

DHCP has the same problem as ARP: a system has no way of knowing who should provide authoritative data

- An attacker on the LAN can provide spoofed DHCP responses and cause new systems (those making DHCP requests) to be configured in a way the attacker wants (e.g., the attacker's system can be provisioned as the gateway so it gets all outbound traffic).

(a) Possible, but not the best answer here.
(b) Since networks like the Ethernet or Wi-Fi don't route traffic and DHCP

Note: You received partial credit of 2 points for (a)

Question 8 version B: 9

VLAN (*Virtual Local Area Network*) *hopping* attacks take place when:

(a) A computer can forge an ethernet address that belongs to a different LAN.
(b) The switch is hacked with malware.
(c) A computer masquerades as an Ethernet switch.
(d) An attacker overflows the switch's CAM table.

- A computer can spoof itself to look like an ethernet switch that communicates with a trunking protocol and extends the local area network.
- The switch to which it is connected will forward all broadcast, multicast, and non-local-port traffic to this connected “switch”

Question 9 version B: 10

TCP SYN cookies are used to:

(a) Avoid having to allocate TCP resources until the TCP handshake is complete.
(b) Authenticate that the sender's IP address is legitimate at the start of a TCP session.
(c) Add unique data to each TCP packet so that packets injected by an attacker can be detected.
(d) Overwhelm a system with TCP connection requests until it cannot accept any more.

- SYN cookies were created to handle SYN flooding attacks

  SYN flooding: send lots of SYN (TCP connection request) segments but never complete the handshake (usually bogus source addresses). Eventually the OS will not be able to accept any new connections until those time out

- With SYN cookies, the OS does not allocate any TCP session state or buffers until it gets an acknowledgement from the sender.
  – The "cookie" is an initial sequence number that isn't random but computed as hash(src_addr, dest_addr, src_port, dest_port, SECRET)

Question 10

version B: 5

BGP (Border Gateway Protocol) hijacking attacks take place by:

(a) An attacker modifying BGP messages in the network.

(b) Accepting messages without validating the source.

(c) Having a router incorrectly advertise a better route for a range of addresses.

(d) Attackers breaking into a border router and snooping on all traffic going through it.

BGP is a distance-vector routing protocol used by routers on the internet (between ISPs)

- A router advertises lists of IP address prefixes that it knows how to route & the cost (distance) to route to these addresses
- Routers forward the information (increasing the distance) to other routers … eventually every router should know which router it should send a given IP packet to for routing

BGP hijacking occurs when a router sends out incorrect information advertising shorter distances (lower costs) for certain addresses

- This leads to other routers sending IP packets to it, thinking it has the best route

Part 3: Firewalls & VPNs

Question 11 version B: 12

Which algorithm is not supported in the IPsec Authentication Header (AH) protocol?

(a) AES-CBC encryption.

(b) HMAC-SHA2 message authentication code.

(c) Diffie-Hellman key exchange.

(d) RSA public key authentication.

- We are presented with algorithms for four functions:
  - (1) Encryption, (2) Integrity, (3) Key exchange, (4) Authentication
- IPsec AH only offers message integrity, not encrypted communication
  - We need authentication to validate the endpoint
  - We need the MAC for message integrity
  - We need key exchange to establish a shared MAC key

Question 12

version B: 13

A *transport mode* IPsec ESP VPN differs from tunnel mode because it:
(a) Does not encrypt the application data.
(b) Allows a client to communicate with only one application.
(c) Sends data at the transport layer (TCP) instead of the network layer (IP).
(d) Does not change the IP header when the packet is routed to the Internet.

A transport mode IPsec VPN encapsulates packets going to a single system rather than to a router

- (a) It encrypts all application data, just like any ESP connection
- (b) It allows clients to communicate with any applications on the target system – apps are unaware that there's a VPN in place
- (c) It still operates at the same
- (d) The IP packet goes directly to the target system in transport mode, so there is no need to encapsulate the IP header itsself

Question 13 version B: 14

Packet filters cannot block traffic based on the:
(a) Source IP address.
(b) URL.
(c) TCP or UDP port number.
(d) Router interface.

- Packet filters filter on:
  - The router interface, network layer headers (layer 3 – IP), and transport layer headers (layer 4 – TCP/UDP)
- They cannot filter on the URL since that is part of the application layer

Question 14

version B: 15

A *DMZ* (demilitarized zone):

(a) Isolates Internet-facing services to another subnet.
(b) Provides a subnet where some traffic does not have to flow through a firewall.
(c) Is an internal network that can be used to test malware since it disallows any traffic to the Internet.
(d) Is a network that can be used by two or more organizations to share data securely.

A DMZ is a separate subnet for Internet-facing services that is separated from other internal systems

All systems, including those in the DMZ should be protected by a firewall

The rationale for a DMZ is that if attackers manage to break into one of the systems in it, they are still on a different LAN and need to figure out how to penetrate the firewall and break into a system on the internal network

Question 15 version B: 16

Deep Packet Inspection (DPI) inspects:

(a) IP header fields such as time-to-live and checksum fields.

(b) Validity of packets with respect to the current TCP session state.

(c) Application-specific data.

(d) A combination of layer 2 (Ethernet), layer 3 (IP), and layer 4 (TCP & UDP) headers.

- Unlike packet filters, DPI enables the firewall to examine application data

Question 16

version B: 17

Deperimeterization is the problem of:

(a) Computers moving in and out of the internal network.

(b) Systems with Internet-facing and internal services on the same network.

(c) Internet-facing services sharing the same computer as internal services.

(d) Not monitoring traffic between the ISP and local network.

- ***Perimeterization*** was the act of creating a perimeter – a boundary – between internal and external systems, where all traffic could be filtered with a firewall (including intrusion detection/prevention systems)
- ***Deperimeterization*** is the problem of people moving devices in and out of that controlled internal network: laptops & phones are regularly moved off premises and connected to other networks (e.g., Wi-Fi hotspots, home networks) where traffic may not be monitored as carefully and they may have a higher chance of getting malware installed that can later attack the network within the perimeter

Question 17

version B: 11

UDP-based protocols are easier to attack than TCP protocols because:

(a) TCP uses sequence numbers.
(b) The payload is not encrypted
(c) UDP datagrams are not signed.
(d) UDP does not guarantee reliable delivery.

(b) Neither TCP nor UDP provide any encryption

(c) Neither TCP nor UDP use any signatures

(d) The ability to retransmit data does not affect one's ability to attack a protocol

UDP packets are autonomous: there is no relationship amongst them. TCP packets are part of a TCP "connection", with sequence numbers in both directions. To be able to insert a packet requires being able to not just forge the right data but to figure out what the proper sequence number should be.

Part 4: Web and mobile security

Question 18

version B: 19

The *SECURE* flag for a web cookie means:
(a) The cookie's contents will be encrypted prior to placing it in an HTTP header.
(b) The cookie will be sent only if there is an HTTPS connection to the server.
(c) A digital signature is attached to the cookie to detect tampering.
(d) The cookie is not accessible to web scripts.

- Two flags in the HTTP protocol for dealing with cookies are:
- HttpOnly: disallows scripts from accessing the cookie
- Secure : send the cookie only if there is an HTTP session

Question 19

version B: 20

CORS (*cross-origin resource sharing*) enables a web:

(a) A web application to download data from another domain.
(b) A web page to include images and scripts from different places.
(c) A web server to redirect a URL to another domain.
(d) A web page to contain links to other domains.

CORS allows a server to define other origins as being equivalent (e.g., other domain names or protocols) so the same-origin policy can apply across these origins

(b) Web pages can always include images & scripts from other sites

(c) A server can send a REDIRECT message anywhere – even to a different origin

(d) Web pages can always link to other origins

Question 20

version B: 21

DNS rebinding attacks are effective because:

(a) A hacked DNS server may return the wrong address for a domain name.

(b) Fake DNS replies coming from a malicious script can confuse a client doing a legitimate query.

(c) Origins are associated with domain names and not addresses.

(d) DNS queries that produce large responses can be used in DDoS attacks.

- With a DNS rebinding attack, the attacker owns the domain name and can manipulate the DNS server that answers DNS queries about the domain (provides IP addresses)
  - Use a really short TTL (time to live) for a DNS response
  - Any JavaScript on the page that will access the same origin again will cause the OS to look up the domain name since the name-address mapping expired
  - Now the attacker's DNS server returns a different address, often one for a system in the victim's internal network
  - The browser is unaware because it only deals with domain names and is still abiding by the same-origin policy

Question 21 version B: 22

One way of defending against *cross-site request forgery* is to:

(a) Validate user input at the browser.

(b) Have the server to check the address of the web page that the request link came from.

(c) Make sure the web page does not use any third-party scripts.

(d) Enforce to the same-origin policy.

XSRF (or CSRF) occurs when an attacker gets the victim to click on a link that performs some action

- Going to the URL causes the browser to send cookies, which include authentication cookies
- E.g., http://mybank.com/?action=transfer&amount=100000&to=attacker_account

One defense: validate the *referrer header* at the server

- The server can then check that the request came from one of its own pages to be considered legitimate

Question 22

version B: 18

Extended Validation (EV) *certificates* differ from other X.509 digital certificates because they:

(a) Use stronger encryption.

(b) Prove the legal entity of the owner of the certificate.

(c) Require validating the entire chain of certificates up to the root certificate.

(d) Perform mutual authentication (the client authenticates the server & the server authenticates the client).

These are essentially the same X.509 certificate but the certification authority (CA) must perform extra checks to validate the entity’s and applicant’s identity:

– Check company incorporation, domain registration, position of applicant, etc.

Question 23

version B: 23

Unlike iOS, *Android* isolates app resources by:
(a) Running each app in a container.
(b) Using a kernel-level sandbox that filters allowable system calls and file access.
(c) Using per-app namespaces.
(d) Assigning a different user ID to each app.

Android assigns each app its own user ID and uses Linux file permissions to isolate access

- No use of containers, kernel sandboxes, or namespaces

Question 24 version B: 25

Unlike Android, *iOS* isolates app resources by:
(a) Running each app in a container.
(b) Using a kernel-level sandbox that filters allowable system calls and file access.
(c) Using per-app namespaces.
(d) Assigning a different user ID to each app.

- iOS uses a kernel-level sandbox to isolate apps
- All apps run under the same user ID

Question 25

version B: 24

iOS's *Secure Enclave* is:

(a) A secure storage service provided by iOS to store keys and other sensitive information.

(b) A gatekeeper component that authorizes messages from one app to another.

(c) An encryption and signing library that is used by iOS apps for data security services.

(d) A separate processor running a different operating system to manage security-sensitive tasks.

- The Secure Enclave is similar to TrustZone found on most ARM processors:
  - Separate OS, isolated programs, communication via messaging
- The Secure Enclave, however, is a separate processor whereas TrustZone just uses a virtual core (separate system registers and memory isolation)

The end