Department of Computer Science

Computer Security

Exam 2

March 31, 2025

Solutions Discussion

Prepared with tender loving care by Paul Krzyzanowski

100 POINTS – 25 QUESTIONS – 4 POINTS EACH – For each statement, select the *most* appropriate answer.

1. In a *Merkle tree*, what is the value stored in a *non-leaf node*?
   a. The raw data of a transaction.
   b. A pointer to the data block.
   c. A digital signature of the data.
   d. The hash of the concatenation of its children's hashes.

Correct answer: d

Explanation: In a Merkle tree, non-leaf nodes store the hash of the concatenation of their children's hashes. This structure allows efficient and tamper-evident verification of large datasets.

a is incorrect because raw transaction data is stored in leaf nodes, not non-leaf nodes.

b is incorrect because Merkle trees use hashes, not pointers, to link nodes.

c is incorrect because digital signatures are not part of the Merkle tree structure.

2. Which of the following best explains why blockchains are considered *immutable*?
   a. Each block contains a digital signature.
   b. All blocks are stored in encrypted form.
   c. Altering a past block requires recomputing all subsequent block hashes.
   d. They are synchronized across many computers worldwide.

Correct answer: c

Explanation: A blockchain is immutable because each block includes a hash of the previous block, so changing a past block would require recalculating all subsequent hashes.

a is incorrect because digital signatures prove authenticity, not immutability.

b is incorrect because encryption protects confidentiality, not integrity.

d is incorrect because synchronization supports availability and redundancy but does not guarantee immutability.

3. What is typically used to *sign* a Bitcoin *transaction*?
   a. The sender's Bitcoin address.
   b. A shared secret between sender and receiver.
   c. The sender's private key.
   d. A digital certificate issued by a trusted authority.

Correct answer: c
Explanation: A Bitcoin transaction is signed using the sender's private key, which proves ownership of the funds being spent.
a is incorrect because the Bitcoin address is derived from the public key and is not used for signing.
b is incorrect because Bitcoin uses asymmetric cryptography, not shared secrets.
d is incorrect because Bitcoin does not rely on certificate authorities.

4. Why is Bitcoin's mining process considered a *proof-of-work* system?
   a. A hash with certain properties must be found to propose a valid block.
   b. Miners must prove that all transactions in a block are valid before adding it to the blockchain.
   c. Each miner must vote on transactions to reach a consensus.
   d. A miner must demonstrate that they used a certain amount of computing cycles.

Correct answer: a

Explanation: Bitcoin uses proof-of-work, where miners must find a hash with certain properties (e.g., below a difficulty target) to propose a valid block.

b is incorrect because while transaction validity is required, it is not the basis of proof-of-work.

c is incorrect because miners do not vote to reach consensus.

d is incorrect because proof-of-work is demonstrated through hash computation, not by showing exact CPU usage.

5. Which trend has reduced the effectiveness of traditional CAPTCHAs in recent years?
   a. Increased use of phones and tablets.
   b. More efficient hash computation algorithms.
   c. Wider use of encrypted sessions.
   d. Advancements in automated machine learning and computer vision.

Correct answer: d
Explanation: Traditional CAPTCHAs have become less effective due to advances in machine learning and computer vision, which allow bots to solve visual puzzles.
a is incorrect because mobile device use affects usability, not the CAPTCHA's effectiveness against bots.
b is incorrect because hash algorithms are unrelated to CAPTCHA solving.
c is incorrect because encryption protects data in transit but does not weaken CAPTCHA security.

6. Which usability advantage is provided by noCAPTCHA compared to traditional CAPTCHA systems?
   a. Many users can pass verification with a single click.
   b. CAPTCHA responses from the server are encrypted end-to-end.
   c. Users are presented with two questions instead of one.
   d. It uses text-based challenges instead of images.

Correct answer: a
Explanation: NoCAPTCHA improves usability by allowing many users to pass verification with a simple checkbox, using behavioral analysis behind the scenes.
b is incorrect because encryption is standard for most web traffic and not a distinguishing feature of noCAPTCHA.
c is incorrect because noCAPTCHA often reduces the number of challenges, not increases them.
d is incorrect because traditional CAPTCHAs also used images; noCAPTCHA attempts to reduce reliance on visible challenges entirely.

7. How does a *capability list* differ from an ACL?
   a. It grants access rights to users based on group membership.
   b. It is typically used to control access to system-level operations.
   c. It associates permissions with subjects instead of with objects.
   d. It allows fine-grained control over object methods.

Correct answer: c
Explanation: A capability list associates permissions with subjects (e.g., users or processes), listing the objects each subject can access. In contrast, an ACL associates permissions with objects, listing which subjects can access them.
a is incorrect because group membership is typically part of discretionary access control, not a defining feature of capability systems.
b is incorrect because both ACLs and capability lists can control system-level and user-level operations; the distinction is in how permissions are stored.
d is incorrect because while some systems support fine-grained control, this is not a distinguishing feature of capability lists specifically.

8. Which of the following scenarios illustrates a weakness of ACL-based access control in large, dynamic systems?
   a. Users may accidentally create capabilities that cannot be revoked.
   b. The system must search all ACLs to determine a user's full access rights.
   c. The kernel must validate all file system operations.
   d. ACLs can only support three classes of users.

Correct answer: b
Explanation: In an ACL-based system, access rights are associated with objects, so determining everything a user can access may require searching through many ACLs, which becomes inefficient in large, dynamic systems.
a is incorrect because it describes a potential issue with capability lists, not ACLs.
c is incorrect because kernel validation is common to most access control models, not unique to ACLs.
d is incorrect because ACLs are not limited to three user classes; that limitation describes traditional UNIX permissions.

9. Which of the following best describes the concept of *privilege separation* in software design?
   a. Denying user processes access to privileged kernel functions.
   b. Dividing a program into components that run with different access permissions or authority levels.
   c. Preventing untrusted users from installing or modifying software.
   d. Managing access control policies in a separate configuration file.

Correct answer: b
Explanation: Privilege separation involves designing software so that different parts run with different levels of privilege, reducing the risk that a vulnerability in one part compromises the entire system.
a is incorrect because denying access to kernel functions is more about enforcing system-wide access control, not internal software design.
c is incorrect because this is a policy decision about software installation, not a design principle.
d is incorrect because separating access policies into a config file is an implementation detail, not the concept of privilege separation.

10. Which of the following actions is *not permitted* under the Bell-LaPadula model?
    a. A top-secret process writing to a secret-level file.
    b. A public-level user writing to a secret-level file.
    c. A top-secret user reading a public-level file.
    d. A confidential-level user writing a confidential-level file.

Correct answer: a

Explanation: In the Bell-LaPadula model, a top-secret process is not allowed to write to a secret-level file because it would violate the no write down rule, which is designed to prevent confidential data from leaking to lower classification levels.

b is incorrect because writing up from a public-level user to a secret-level file is allowed under Bell-LaPadula since it does not violate confidentiality.

c is incorrect because reading down, such as a top-secret user reading public data, is permitted.

d is incorrect because reading or writing at the same classification level is allowed by the model.

11. What is the primary security goal of the *Biba model*, as opposed to the Bell-LaPadula model?
    a. Ensuring availability of data even under high system load.
    b. Protecting the confidentiality of classified information.
    c. Preserving data integrity by preventing modification of high-integrity data by low-integrity sources.
    d. Limiting users from exceeding resource quotas.

Correct answer: c
Explanation: The Biba model focuses on data integrity and enforces "no write up" to prevent low-integrity subjects from modifying high-integrity data.
a is incorrect because availability is not the focus of the Biba model.
b is incorrect because confidentiality is the focus of Bell-LaPadula, not Biba.
d is incorrect because resource quotas relate to system management, not integrity models.

12. Which of the following best describes the goal of *multilateral security*?
    a. To enforce hierarchical access controls based on clearance levels.
    b. To allow data sharing among users within the same classification level.
    c. To ensure that discretionary controls are applied after mandatory ones.
    d. To protect data owned by mutually distrustful parties by isolating it through compartments.

Correct answer: d
Explanation: Multilateral security aims to protect data from being accessed by other parties, even if they share the same clearance level, by isolating it in compartments. This is useful when users are mutually distrustful.
a is incorrect because hierarchical access is characteristic of multilevel models like Bell-LaPadula.
b is incorrect because multilateral security is about isolation, not sharing.
c is incorrect because the order of access control enforcement is not the defining feature of multilateral security.

13. What is a key challenge in implementing the *Chinese Wall* security model in practice?
    a. It requires tracking each user's access history to enforce dynamic access control.
    b. It lacks a defined mechanism for enforcing access control decisions.
    c. It requires encrypting all sensitive data with per-client keys.
    d. It prevents users from accessing any public data once they've viewed confidential information.

Correct answer: a
Explanation: The Chinese Wall model dynamically adjusts access permissions based on a user's prior access history to prevent conflicts of interest, which requires tracking user interactions with data.
b is incorrect because the model does define an enforcement mechanism, though it's more complex.
c is incorrect because encryption is not required by the model itself.
d is incorrect because the model allows access to public data at any time; it restricts access to confidential data from competing entities.

14. Which of the following best describes how a *heap overflow* can lead to exploitable behavior in a vulnerable program?
    a. Overwriting the return address of the current function.
    b. Modifying nearby data structures that influence the program's control flow or behavior.
    c. Causing a segmentation fault by accessing memory beyond the heap allocation.
    d. Overwriting user data in a different process running on the same system.

Correct answer: b
Explanation: A heap overflow can overwrite adjacent memory regions on the heap, including data structures such as function pointers or object metadata, which can then be hijacked to alter the program's control flow or behavior.
a is incorrect because overwriting the return address typically occurs with stack overflows, not heap overflows.
c is incorrect because while a segmentation fault may occur, it is a crash, not typically exploitable behavior.
d is incorrect because standard memory protection prevents one process from writing directly into another's memory space.

15. How does a *stack canary* help defend against stack-based buffer overflow attacks?
    a. It randomizes the stack memory layout to make exploits harder.
    b. It encrypts the return address by XOR-ing it with the canary to prevent it from being overwritten.
    c. It places a known value before the return address; the program aborts if this value is modified.
    d. It forces all local variables to be allocated on the heap to avoid stack overflow risks.

Correct answer: c
Explanation: A stack canary is a known value placed between local variables and the return address on the stack. If a buffer overflow overwrites the canary, the program detects the change and aborts execution to prevent exploitation.
a is incorrect because stack randomization is a separate defense mechanism known as ASLR.
b is incorrect because the canary is not used to encrypt the return address.
d is incorrect because canaries do not force variables onto the heap; they remain on the stack.

16. Which of the following techniques can bypass *Data Execution Prevention* (DEP)?
    a. Overwriting a buffer with a NOP slide.
    b. Exploiting a format string vulnerability.
    c. Using Return-Oriented Programming (ROP) gadgets.
    d. Injecting code into the stack.

Correct answer: c

Explanation: Return-Oriented Programming bypasses DEP by chaining together small pieces of existing executable code (gadgets) already present in memory, without injecting new code.

a is incorrect because a NOP slide is part of classic scode injection, which DEP blocks by making memory non-executable.

b is incorrect because format string vulnerabilities do not inherently bypass DEP unless combined with other techniques. A format string vulnerability is a memory corruption bug, not a bypass technique.

d is incorrect because injecting code into the stack fails if DEP is enabled, since the stack is marked non-executable.

17. Which of the following attacks is most likely to *fail* if ASLR is properly enabled?
    a. A *return-to-libc* attack that jumps to a function in `libc`.
    b. An integer overflow in image parsing code.
    c. A format string bug that prints out memory addresses.
    d. A buffer overflow attack that crashes the program with a segmentation fault.

Correct answer: a
Explanation: A return-to-libc attack relies on knowing the memory address of library functions, which is made unpredictable when ASLR is enabled, making the attack likely to fail.
b is incorrect because ASLR does not prevent logic or arithmetic bugs like integer overflows.
c is incorrect because format string bugs that leak memory addresses may still work even with ASLR and can sometimes be used to defeat it.
d is incorrect because a segmentation fault may occur regardless of ASLR, and crashing a program is not a reliable security defense.

18. In Return-Oriented Programming (ROP), how is a *ROP chain* typically constructed in a buffer overflow exploit?
    a. By chaining pointers to unused heap memory blocks.
    b. By overwriting the stack with addresses of short instruction sequences ending in `ret`.
    c. By modifying system registers to gain kernel-level privileges.
    d. By injecting encrypted code into the `.text` segment of the binary.

Correct answer: b
Explanation: In ROP, an attacker crafts a payload of addresses pointing to short instruction sequences (gadgets) that end in a return instruction, chaining them together on the stack to perform arbitrary computation.
a is incorrect because ROP does not use unused heap memory.
c is incorrect because ROP does not directly modify system registers; it manipulates control flow through return addresses.
d is incorrect because ROP avoids injecting new code and works with existing executable code already loaded in memory.

19. An *integer overflow* in a memory allocation request in C++ can result in:
    a. A request that allocates much less memory than intended, leading to potential buffer overflow.
    b. An overly large allocation that causes the program to run out of memory.
    c. A processor exception that terminates the process immediately.
    d. The memory allocator switching from heap memory to virtual memory pages.

Correct answer: a
Explanation: Integer overflows in allocation calculations can result in smaller-than-expected memory allocations, leading to buffer overflows when more data is written than the memory can hold.
b is incorrect because the overflow wraps around, usually causing a small allocation, not an oversized one.
c is incorrect because integer overflows in C++ often go unnoticed unless explicitly checked, and do not trigger a processor exception.
d is incorrect because overflow in size calculations does not change the allocator's memory type but leads to unsafe memory usage.

20. What is the core idea behind an *SQL injection* attack?
    a. Trick the system into using cached database results.
    b. Insert malicious input that changes the structure or logic of a database query.
    c. Overload the database server with too many requests to make it unresponsive.
    d. Inject data that causes a buffer overflow and overwrites the return address.

Correct answer: b
Explanation: SQL injection involves inserting malicious input that alters the intended structure or logic of an SQL query, allowing attackers to manipulate the database.
a is incorrect because caching is unrelated to SQL injection.
c is incorrect because overloading the server is a denial-of-service attack, not SQL injection.
d is incorrect because SQL injection targets query logic, not memory corruption like buffer overflows.

21. What is the main idea behind a *path equivalence vulnerability*?
    a. Two files having the same size but different contents.
    b. Two different-looking file paths that actually resolve to the same file or directory.
    c. Multiple users having write access to the same file.
    d. Two different files that have the exact same contents.

Correct answer: b

Explanation: A path equivalence vulnerability occurs when different-looking file paths resolve to the same file or directory, allowing attackers to bypass access checks or manipulate protected files.

a is incorrect because file size and content differences are unrelated to path resolution.

c is incorrect because shared write access may be risky but is not a path equivalence issue.

d is incorrect because file content duplication has nothing to do with how paths are interpreted by the system.

22. Which of the following most clearly introduces a *TOCTTOU vulnerability* in a multi-user environment?
    a. A backup script opens a file for writing without checking if it already exists.
    b. A program uses the `stat` system call to check that a file is smaller than 1 MB, then reopens it for uploading.
    c. A *cron* job deletes temporary files that haven't been accessed in over an hour.
    d. A configuration tool asks the user to confirm settings before writing them to a local config file.

Correct answer: b

Explanation: This is a classic time-of-check to time-of-use vulnerability: the file is checked using stat, then reopened later, allowing an attacker to replace the file between those steps.

a is incorrect because not checking existence before writing does not inherently introduce a TOCTTOU race.

c is incorrect because regularly deleting files is unrelated to timing or check-use conditions.

d is incorrect because user confirmation before writing settings does not involve file system race conditions.

23. Which of the following is true about Linux *capabilities*?
    a. They prevent race conditions in *setuid* programs.
    b. They require containers to run in kernel mode.
    c. They let unprivileged users assign elevated privileges to a process.
    d. They can be used to reduce a process's privileges at runtime.

Correct answer: d
Explanation: Linux capabilities can be applied when a process runs (or is running), allowing a process to get fewer privileges even if it's running as root to reduce the impact of potential exploitation.
a is incorrect because capabilities do not directly prevent race conditions.
b is incorrect because containers do not require kernel mode execution; they run in user space with kernel isolation.
c is incorrect because only privileged users (such as root) can assign elevated capabilities to processes.

24. What advantage does a syscall-based *sandbox* provide beyond Linux's capabilities, control groups, and namespaces?
    a. It allows a process to safely elevate its privileges during execution.
    b. It controls access to hardware devices like GPUs and USB controllers.
    c. It enforces access control using security labels and mandatory access rules.
    d. It can block specific specific system calls, reducing the attack surface even for unprivileged processes.

Correct answer: d
Explanation: A syscall-based sandbox like seccomp-BPF can block specific system calls entirely, limiting the actions a process can take even if it is otherwise unprivileged.
a is incorrect because sandboxing restricts privilege, not elevates it.
b is incorrect because hardware access is more commonly managed by device permissions or MAC frameworks.
c is incorrect because security labels and mandatory rules are features of systems like SELinux, not syscall filtering.

25. Which of the following best describes the difference between AppArmor and seccomp-BPF?
    a. AppArmor enforces access control at the application layer, while seccomp-BPF operates at the kernel layer.
    b. seccomp-BPF restricts memory allocation, while AppArmor blocks system calls.
    c. seccomp-BPF operates only on root processes, while AppArmor is for unprivileged ones.
    d. AppArmor restricts file access based on path names, while seccomp-BPF restricts system calls.

Correct answer: d
Explanation: AppArmor restricts file access based on file paths, while seccomp-BPF controls which system calls a process can make.
a is incorrect because both operate with kernel enforcement, not at the application layer.
b is incorrect because seccomp-BPF does not restrict memory allocation, and AppArmor does not block system calls.
c is incorrect because seccomp-BPF and AppArmor can apply to both privileged and unprivileged processes.

The end.