Lab #6 - November 1, 2018

Due Dec 31 at 11:59pm **Points** 14 **Questions** 14
Available Nov 1 at 8:45am - Dec 31 at 11:59pm 2 months **Time Limit** None
Allowed Attempts Unlimited

Instructions

[1] Good morning and welcome to the sixth lab session!

Lab sessions provide the opportunity for recitation and more in-depth understanding of the materials covered in class, as well as preparation for upcoming homework assignments.

Your attendance only of a given lab session (and, thus, your participation in the assignments and/or discussions) gives you full credit. You are expected to stay in the lab for the entire session, or until the TAs release the class possibly earlier than scheduled, and to actively participate in the discussions (e.g., asking questions, answering to questions, etc.).

Typically, lab assignments are offered in the form of an ungraded quiz, which should not be interpreted as a test or a mini exam.

[2] Today's quiz is planned so that it reviews materials covered in our last two lectures in class.

Take the Quiz Again

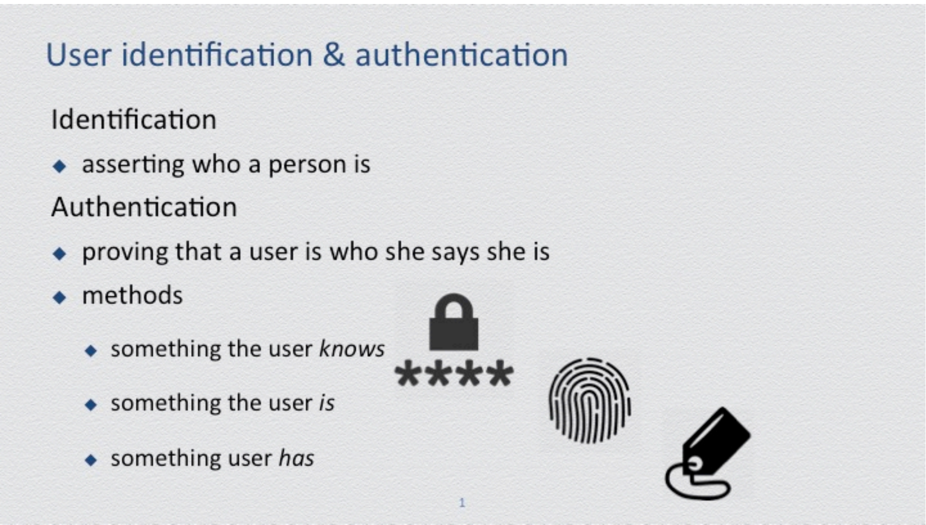
Attempt History

	Attempt	Time	Score
LATEST	Attempt 1	7 minutes	7 out of 14

Submitted Nov 8 at 1:57pm

Question 1

1 / 1 pts



User identification & authentication

Identification

- asserting who a person is

Authentication

- proving that a user is who she says she is
- methods
  - something the user *knows*
  - something the user *is*
  - something user *has*

Suppose that a user provides her (login) name and uses one of the three method above to authenticate into a computer system (either terminal or remote server via a web browser). When does user authentication imply user identification?

○ Always, because the user provides her user name.

○ Only when biometric authentication is used.

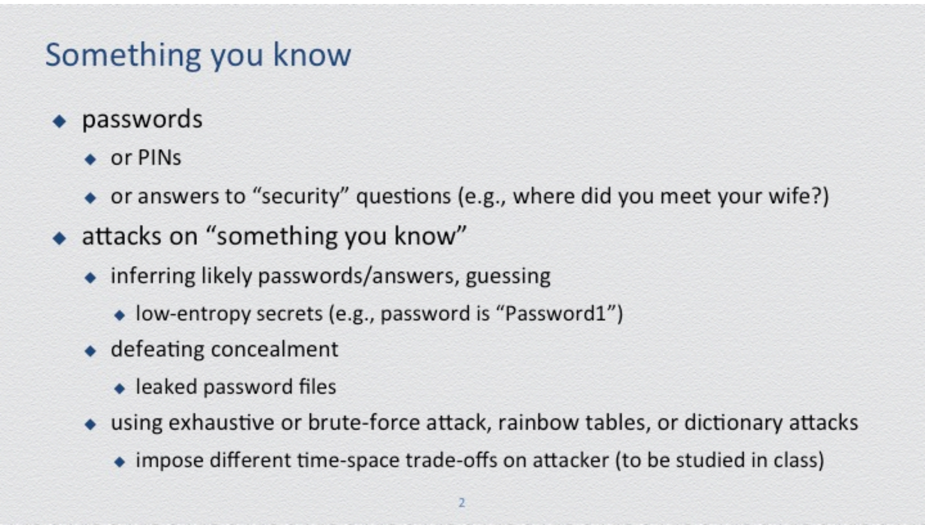
Correct!

● When the methods are: "something the user, and only this user, knows, is or has."

Indeed. Only under the assumption that the user's password is only known by her, or that the user's biometric "fingerprint" is unique and impossible to clone, or that the user's authentication token is at all times possessed only by her, we can be certain that a successful authentication corresponds to a successful identification.

Question 2

1 / 1 pts



Something you know

- passwords
  - or PINs
  - or answers to “security” questions (e.g., where did you meet your wife?)
- attacks on “something you know”
  - inferring likely passwords/answers, guessing
    - low-entropy secrets (e.g., password is “Password1”)
  - defeating concealment
    - leaked password files
  - using exhaustive or brute-force attack, rainbow tables, or dictionary attacks
    - impose different time-space trade-offs on attacker (to be studied in class)

2

An "online" brute-force or dictionary attack against passwords employs only the authentication system; that is, the attacker attempts to impersonate a victim by trying all possible (short length) passwords or passwords coming from a known dictionary. An offline brute-force or dictionary attack employs a leaked file of hashed passwords.

Can online attacks be prevented and, if yes, how?

Correct!

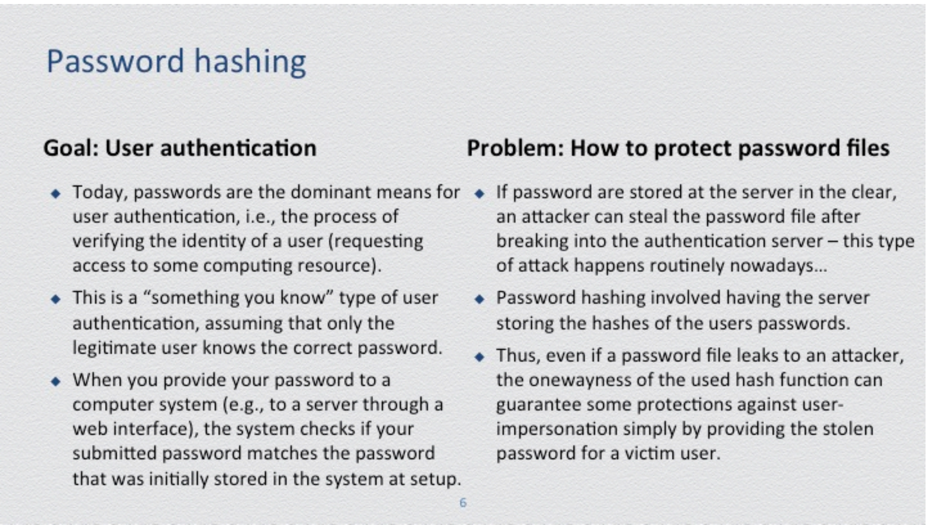
● Yes. By having the system (authentication server or application) blocking any authentication attempt after a (small) fixed number of consecutive failed authentication attempts.

Indeed, this is an easy countermeasure. Even if an attacker employs a botnet to launch an attack from different machines, the authentication server can safely assume that repeated failed authentication attempts by a benign user is very unlikely to happen.

○ They cannot be avoided.

Question 3

1 / 1 pts



Password hashing

Goal: User authentication

- Today, passwords are the dominant means for user authentication, i.e., the process of verifying the identity of a user (requesting access to some computing resource).
- This is a "something you know" type of user authentication, assuming that only the legitimate user knows the correct password.
- When you provide your password to a computer system (e.g., to a server through a web interface), the system checks if your submitted password matches the password that was initially stored in the system at setup.

Problem: How to protect password files

- If password are stored at the server in the clear, an attacker can steal the password file after breaking into the authentication server – this type of attack happens routinely nowadays…
- Password hashing involved having the server storing the hashes of the users passwords.
- Thus, even if a password file leaks to an attacker, the onewayness of the used hash function can guarantee some protections against user-impersonation simply by providing the stolen password for a victim user.

6

What is problematic with the above "naive" approach for password protection via hashing?

Correct!

● Passwords are typically easy to predict.

Indeed: I know that your password starts with a capital letter and end with a number... Doesn't it?

If an attacker predicts a dictionary of popular passwords that most users choose, he can compute their hash values and then compare the leaked hashed passwords against this "hash table" to easily recover in plaintext format a significantly large portion of the stolen passwords!

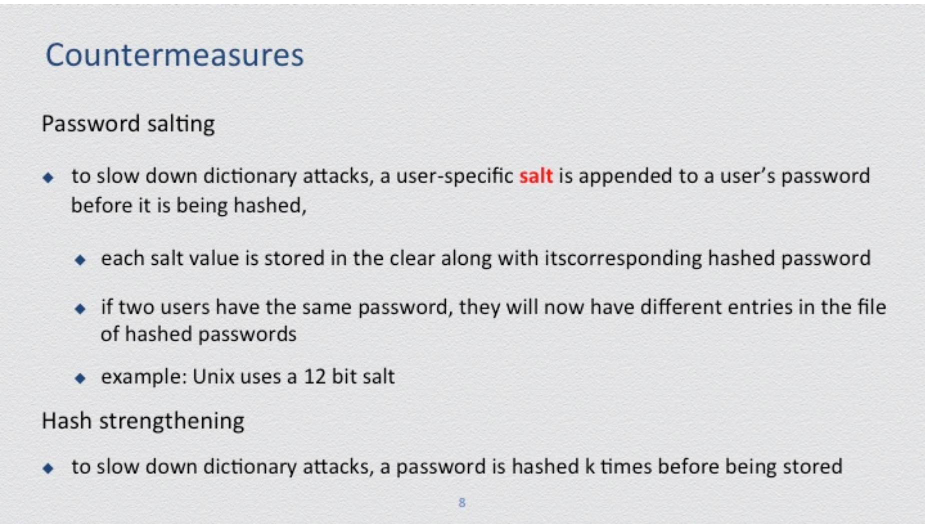
In fact, the above attack strategy is known as an offline dictionary attack, where words from a known English dictionary (and variations of these words) are hashed by the attacker, in order to find and match such a computed hash against one hashed password - in this case, the attacker can impersonate the user associated with this hash password.

○ Nothing is problematic as long as an one-way and collision-resistant hash function is used.

○ Passwords can be leaked during their transmission to the server.

Question 4

0 / 1 pts



Countermeasures

Password salting

- to slow down dictionary attacks, a user-specific **salt** is appended to a user’s password before it is being hashed,
  - each salt value is stored in the clear along with itscorresponding hashed password
  - if two users have the same password, they will now have different entries in the file of hashed passwords
  - example: Unix uses a 12 bit salt

Hash strengthening

- to slow down dictionary attacks, a password is hashed k times before being stored

8

How does password "salting" help?

orrect Answer

○ It makes dictionary attacks user-specific.

○

It does not offer any help unless salts as being kept secret from the attacker.

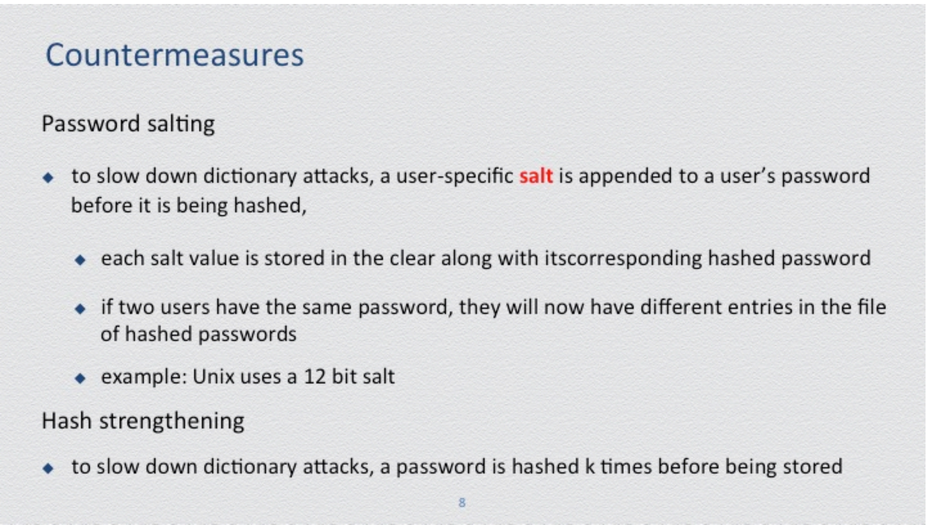
'ou Answered

● It eliminates dictionary attacks.

Not quite. Such attacks are still feasible, but now pre-computation of hash values prior to getting access to a file of hashed passwords is essentially useless.

Question 5

0 / 1 pts



Countermeasures

Password salting

- to slow down dictionary attacks, a user-specific **salt** is appended to a user's password before it is being hashed,
  - each salt value is stored in the clear along with itscorresponding hashed password
  - if two users have the same password, they will now have different entries in the file of hashed passwords
  - example: Unix uses a 12 bit salt

Hash strengthening

- to slow down dictionary attacks, a password is hashed k times before being stored

8

What does repeated password hashing offer?

'ou Answered

● It only makes the attacker's task harder.

Yes, but it also makes makes password verification slower.

orrect Answer

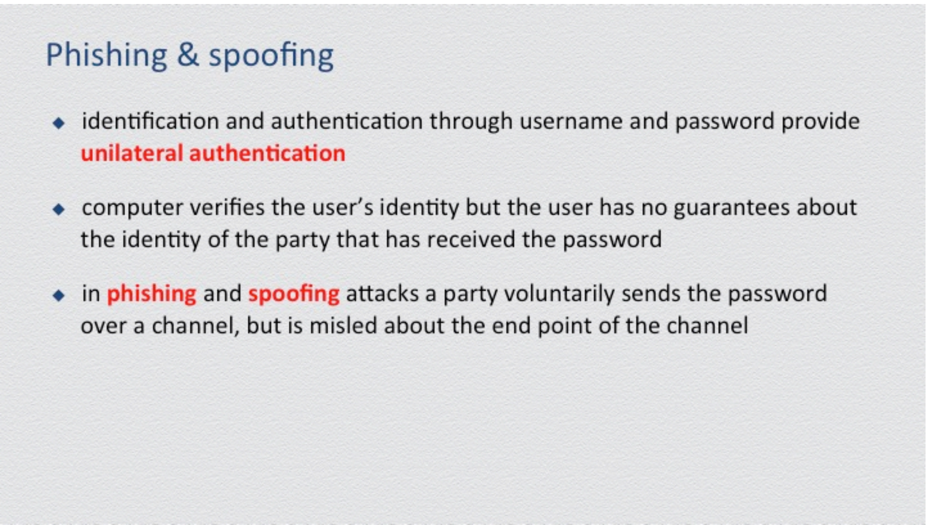
○ A trade-off between security and efficiency.

○ Perfect security.

○ It improves efficiency.

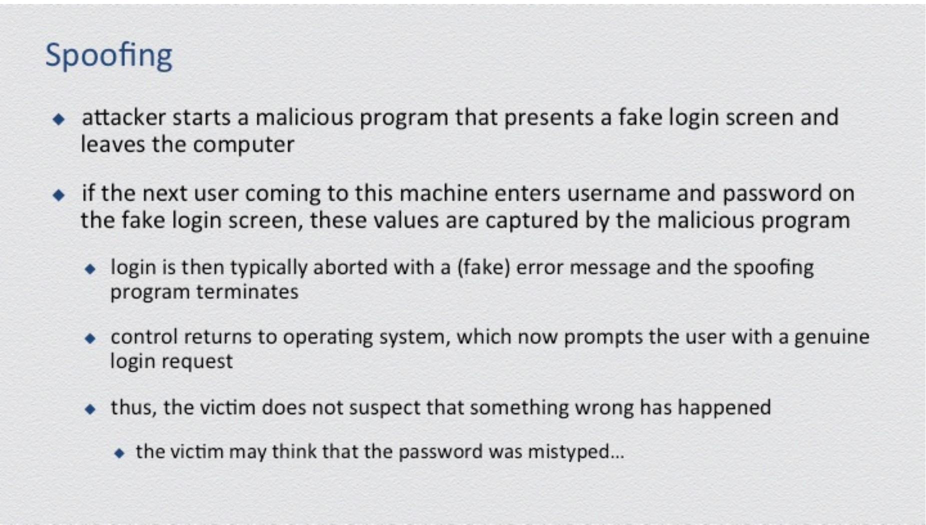
Question 6

0 / 1 pts



Phishing & spoofing

- identification and authentication through username and password provide **unilateral authentication**
- computer verifies the user's identity but the user has no guarantees about the identity of the party that has received the password
- in **phishing** and **spoofing** attacks a party voluntarily sends the password over a channel, but is misled about the end point of the channel



Spoofing

- attacker starts a malicious program that presents a fake login screen and leaves the computer
- if the next user coming to this machine enters username and password on the fake login screen, these values are captured by the malicious program
  - login is then typically aborted with a (fake) error message and the spoofing program terminates
  - control returns to operating system, which now prompts the user with a genuine login request
  - thus, the victim does not suspect that something wrong has happened
    - the victim may think that the password was mistyped...

Which of the following has been used also as a countermeasure against password spoofing?

'ou Answered

● Rebooting the computer.

This could work but it's also very disrupting for the user - it's not much user friendly.

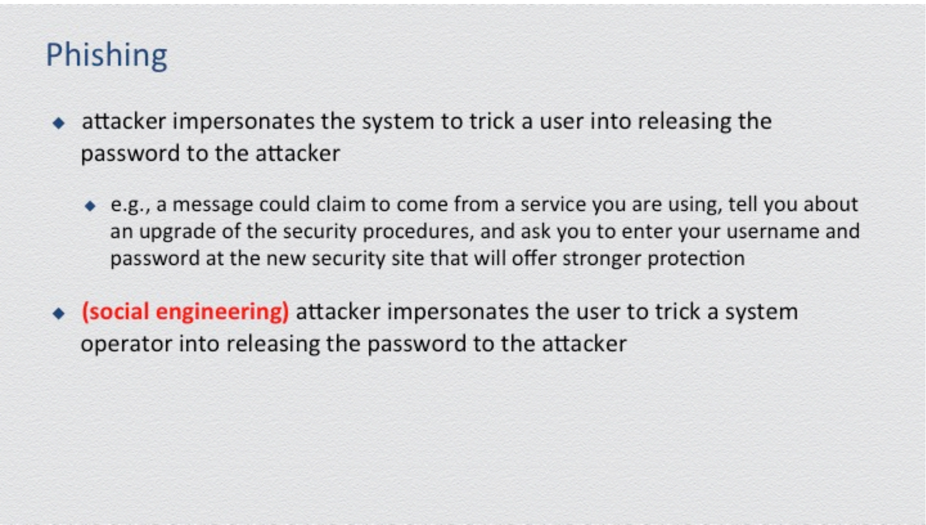
orrect Answer

○ The CTRL+ALT+DEL key combination in some Windows versions.

○ Not ever using computers.

Question 7

1 / 1 pts



Phishing

- attacker impersonates the system to trick a user into releasing the password to the attacker
  - e.g., a message could claim to come from a service you are using, tell you about an upgrade of the security procedures, and ask you to enter your username and password at the new security site that will offer stronger protection
- (social engineering) attacker impersonates the user to trick a system operator into releasing the password to the attacker

Which of the following names is relevant to the "social engineering" version of the above phishing attacks?

Try to find information on Wikipedia, e.g., by Googling "wiki <name as below> email hack" (the incident is also described in the course's textbook).

Correct!

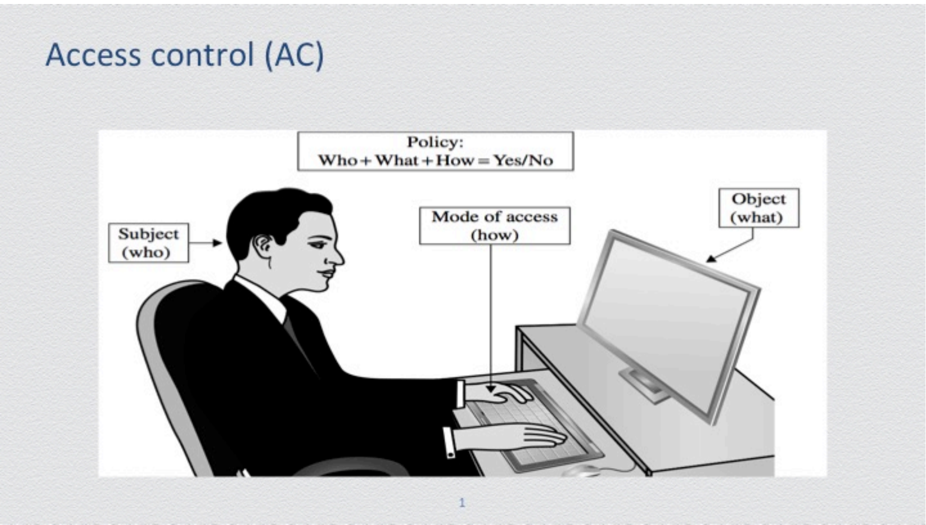
● Sarah Palin.

Yes. When "security questions" (or so-called knowledge-based authentication) are used for password recovery or password reset, then bad things can happen because the answers to typical security questions are easily predictable by prior, background or easily searchable information about the victim.

○ Hillary Clinton.

Question 8

0 / 1 pts



Access control (AC)

Access control refers to a set of mechanisms for defining and enforcing operational security policies in a computing system. Such policies specify which subjects (who) can access which objects (what) and in which mode of access (how), and authorization of such accesses are decided by a reference monitor: Typically, a subject (an "active" entity, e.g., a user) makes an access request on an object (a "passive" entity, e.g., a file), and the reference monitor finds and evaluates the security policy (or policies) relevant to this request.

How do the concepts of "user", "principal" and "subject" relate to each other in the above setting?

○ Subjects are either human users or non-human principals.

ou Answered

● They are all equivalent.

Not quite. Subjects operate on behalf of principals to issue access requests and a granted access is based on the principal’s name bound to the subject (in some unforgeable manner at authentication time).

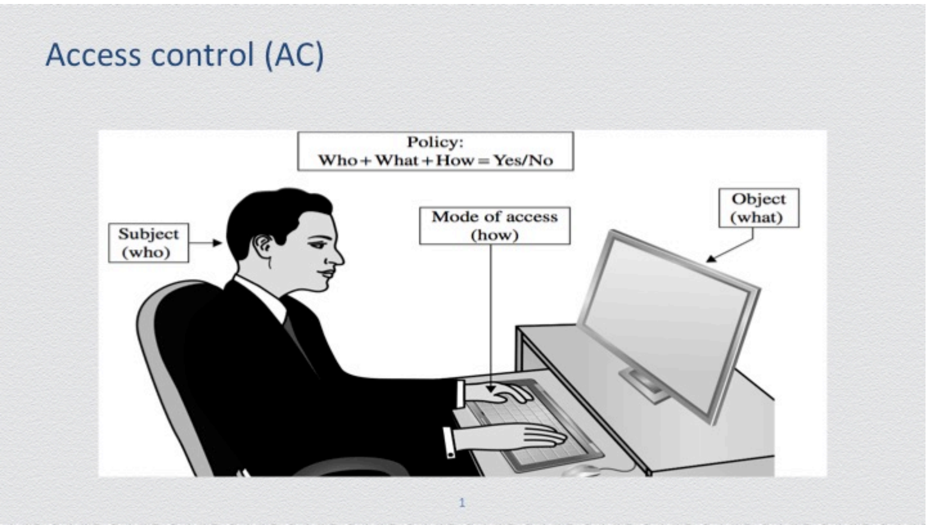
orrect Answer

○ User is a person associated with request; principal is the entity that can be granted access to objects; subject is the process operating on behalf of a principal in issuing an access request.

In reality, when a user wants to access some resources, its user identity (the name used in the system) is often the principal associated with this access request. The subject that actually makes this request is often represented by a surrogate program that is running on behalf of this principal. Thus, a principal is typically associated with a name in the system, which is typically user identity. Subjects operate on behalf of principals to issue access requests and a granted access is based on the principal's name bound to the subject (in some unforgeable manner at authentication time).

Question 9

1 / 1 pts



Access control (AC)

Moreover:

- Objects are things on which an action can be performed, e.g., files, tables, programs, memory objects, hardware devices, strings, data fields, network connections, processors, etc.
- Access modes are any controllable actions of subjects on objects, including, but not limited to, read, write, modify, delete, execute, create, destroy, copy, export, import, and so forth.

Can objects, which are managed under access control, include users?

Correct!

● Yes.

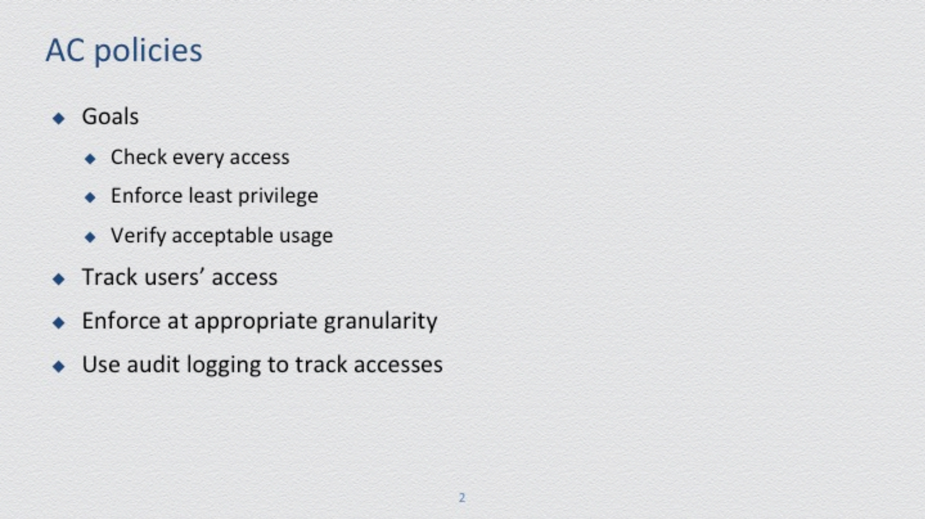
The operating system (a program representing the system administrator) can act on a user, for example, allowing a user to execute a program, halting a user, or assigning privileges to a user.

○ No.

Objects can include users too, or rather programs or processes representing users, because the operating system (a program representing the system administrator) can act on a user, for example, allowing a user to execute a program, halting a user, or assigning privileges to a user.

Question 10

1 / 1 pts



AC policies

- Goals
  - Check every access
  - Enforce least privilege
  - Verify acceptable usage
- Track users' access
- Enforce at appropriate granularity
- Use audit logging to track accesses

2

An AC policy defines specific access-control limitations (who can access what in which ways). In addition to the primary goals described above, an AC policy should be applied at the appropriate level (e.g., file system level Vs. database attributes level) and should be accompanied with a reliable logging systems for auditing past allowed or disallowed accessed.

How "check every access" should be interpreted?

○ Only new access requests should be checked, that is, every new triplet of the form <subject, object, mode> that has not been evaluated before.

Correct!

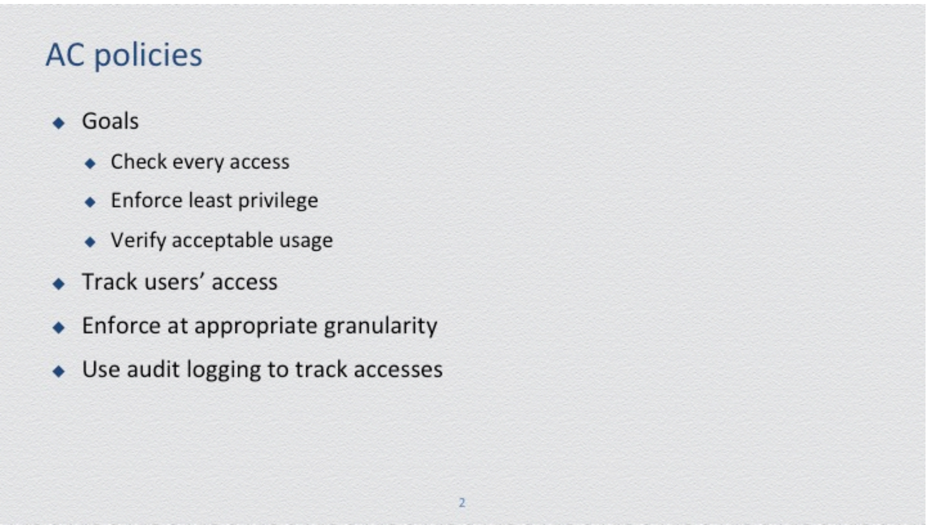
● Literally.

Any request should be checked.

Every access request must be separately evaluated and accepted or rejected, because AC capabilities may change over time. For instance, a user’s privilege to access an object may be revoked, so possible previous authorizations for the user to access the object should not mean anything about a present access request. Revocation of privileges happen when a user should not retain indefinite access to the object or in situations where we want to prevent further access if we suspect that the user behaves maliciously or immediately after we detect that the user is being impersonated. For this reason, we should aim to check every access by a user to an object.

Question 11

0 / 1 pts



AC policies

- Goals
  - Check every access
  - Enforce least privilege
  - Verify acceptable usage
- Track users' access
- Enforce at appropriate granularity
- Use audit logging to track accesses

2

What does "enforce least privilege" mean?

orrect Answer

○ That a given subject should be only granted access to a minimal set of objects that are sufficient for the subject to perform a particular job.

'ou Answered

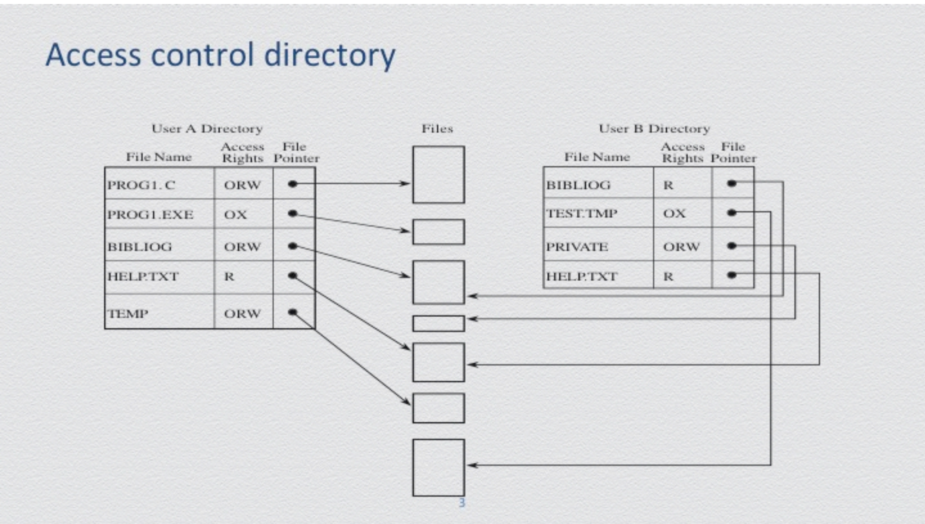
● That subjects (i.e., users) with low privileges should be given access to only low-privilege objects (e.g., less valuable resources).

Not quite.

It means that a subject should have access to the smallest number of objects necessary to perform some task. In information-based settings, the principle of least privilege is equivalent to the principle of "need-to-know" where a user should learn only the minimal information that is needed in order to perform some task and nothing more. Extra information, even useless or harmless, should never be provided. For example, a program should not have access to the absolute memory address to which a page number reference translates, even though the program could not use that address in any effective way. Not allowing access to unnecessary objects guards is a good security hygiene, against security weaknesses should a protection mechanism partially fails.

Question 12

1 / 1 pts



Access control directory

User A Directory

File Name	Access Rights	File Pointer
PROG1.C	ORW	●
PROG1.EXE	OX	●
BIBLIOG	ORW	●
HELP.TXT	R	●
TEMP	ORW	●

User B Directory

File Name	Access Rights	File Pointer
BIBLIOG	R	●
TEST.TMP	OX	●
PRIVATE	ORW	●
HELP.TXT	R	●

One way to represent an AC policy.

Is this a subject-based or an object-based approach?

○ Object-based.

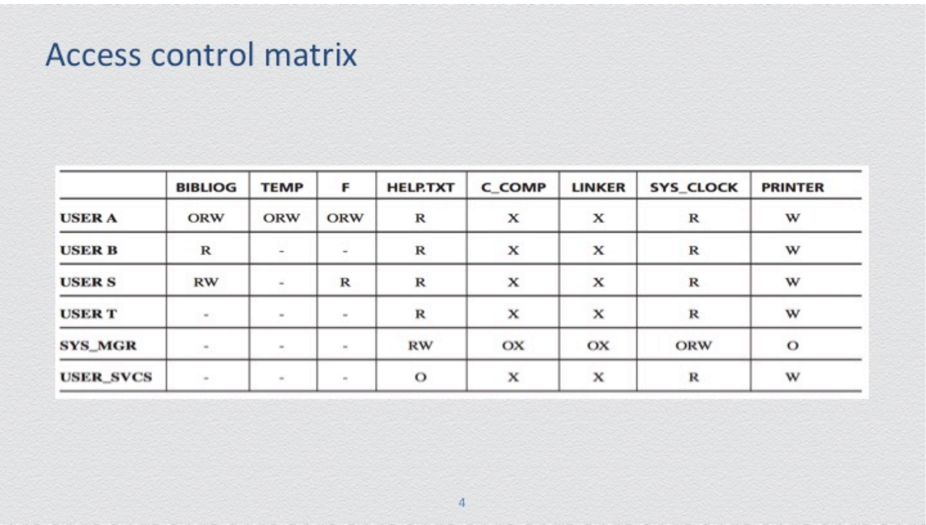
Correct!

● Subject-based.

Access validation starts with the subject issuing the access request.

Question 13

0 / 1 pts



Access control matrix

	BIBLIOG	TEMP	F	HELP.TXT	C_COMP	LINKER	SYS_CLOCK	PRINTER
USER A	ORW	ORW	ORW	R	X	X	R	W
USER B	R	-	-	R	X	X	R	W
USER S	RW	-	R	R	X	X	R	W
USER T	-	-	-	R	X	X	R	W
SYS_MGR	-	-	-	RW	OX	OX	ORW	O
USER_SVCS	-	-	-	O	X	X	R	W

4

Alternatively, a matrix is used to represent all access rights for all possible subject-object pairs.

Why this method is only useful conceptually (and is not used in practice)?

orrect Answer

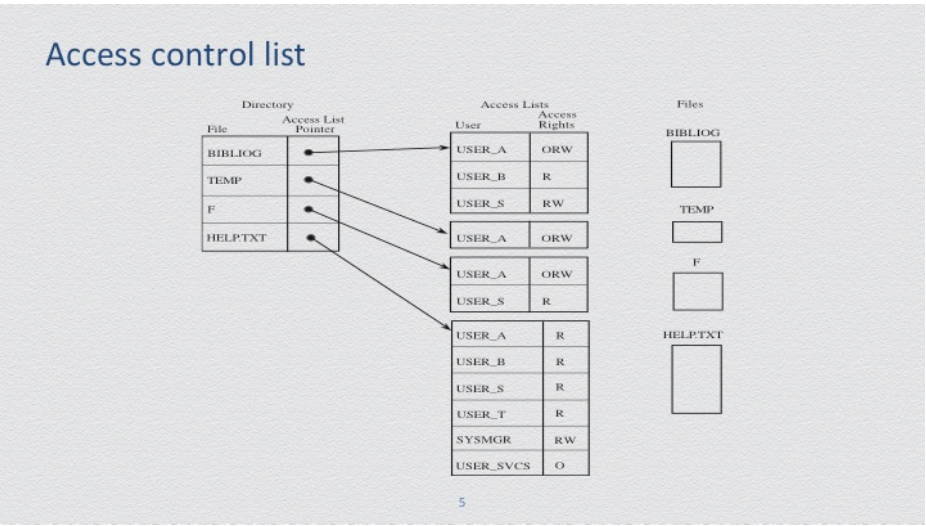
○ Because it corresponds to a huge table that is most of the times very sparse.

'ou Answered

● Because searching in this matrix is computationally expensive.

Question 14

0 / 1 pts



Access control list

Directory

File	Access List Pointer
BIBLIOG	●
TEMP	●
F	●
HELP.TXT	●

Access Lists

User	Access Rights
USER_A	ORW
USER_B	R
USER_S	RW

USER_A	ORW

USER_A	ORW
USER_S	R

USER_A	R
USER_B	R
USER_S	R
USER_T	R
SYSMGR	RW
USER_SVCS	O

Alternatively, objects each carry an access control list (ACL) specifying which subjects have access rights for which actions types (modes of actions). Access control in Unix follows this method, where the concept of a "group" is used to map many users (of similar access control rights) into a single principal entity. How do such groups help?

'ou Answered

◉ They improve security.

Not really. It's an efficiency feature.

orrect Answer

○ They improve efficiency.