Lab #8 - November 29, 2018

Due Dec 31 at 11:59pm **Points** 13 **Questions** 13
Available Nov 29 at 9:20am - Dec 31 at 11:59pm about 1 month **Time Limit** None
Allowed Attempts Unlimited

Instructions

[1] Good morning and welcome to the eighth 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,

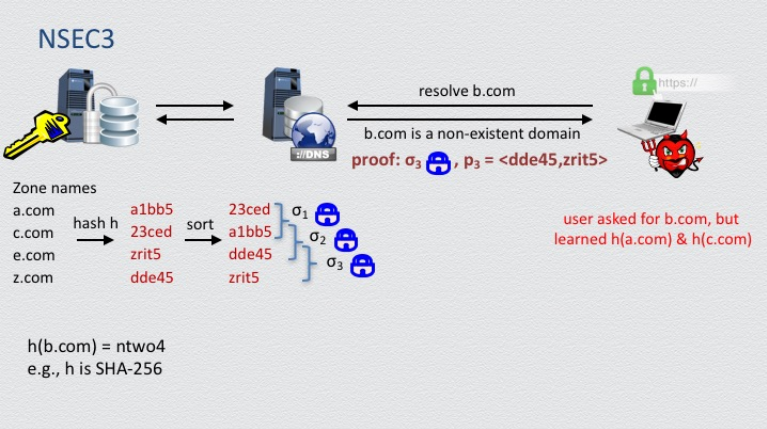
Take the Quiz Again

Attempt History

	Attempt	Time	Score
LATEST	Attempt 1	11 minutes	7 out of 13

Submitted Nov 29 at 1:54pm

Question 1 0 / 1 pts



NSEC3 is a protocol extension for DNSSEC that provides verifiable answers to domain name-resolution queries for names that do not exist.

Does NSEC3 successfully solve the security problem that motivated its design and why?

orrect Answer

No, because offline dictionary attacks can still leak valid domain names previously unknown to the attacker.

'ou Answered

Yes, because non-queried domain names are only leaked in hashed form.

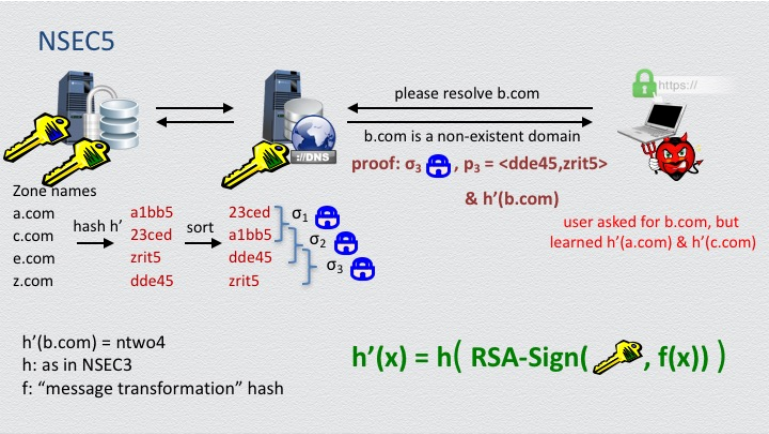
This is the solution that was by design adopted by NSEC3. But since domain names are typically human-memorable names, they are easy to predict, therefore an attacker can easily perform an offline dictionary attack to effectively invert the hashes and learn "near-miss" domain names "in the clear."

Yes, because "near-miss" domain names are authenticated.

The design of NSEC3 was motivated by a vulnerability of its predecessor protocol NSEC, namely the fact that the attacker could trivially learn "near-miss" valid domain names "in the clear", thus allowing zone enumeration attacks against a target organization. By hashing domain names, NSEC3 supposedly removed this vulnerability, since the attacker would now learn "near-miss" valid domain names "in a hashed form." But since domain names are typically human-memorable names, they are easy to predict, therefore an attacker can easily perform an offline dictionary attack to effectively invert the hashes and learn "near-miss" domain names "in the clear."

Question 2

1 / 1 pts



As discussed in class, NSEC5 is a protocol, currently being developed in a RFC, that has been recently proposed as a NSEC3 replacement that eliminates some of the vulnerabilities related to leakage of non-queries unknown domain names, thus also eliminating zone enumeration attacks to a large degree.

What is the key technical feature in NSEC5 that allows to successfully address the vulnerabilities in NSEC3?

Correct!

The use of a secretly computable but publicly verifiable cryptographic hash function h'.

Indeed, the protocol operates exactly as in NSEC3, but now the attacker cannot evaluate hash function h'() without explicitly issuing a DNS query. Thus, offline dictionary attacks are no longer feasible (only online dictionary attacks are possible).

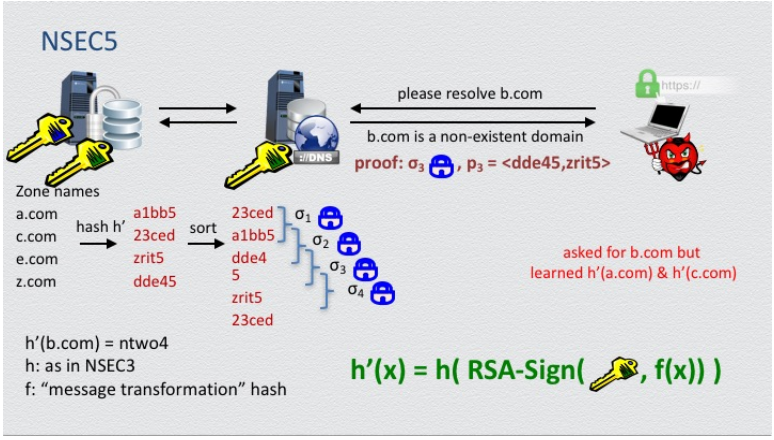
The use of "green" digital signature (namely RSA signatures).

The use of "blue" digital signatures over alphabetically ordered elements.

The key feature is the use of a secretly computable but publicly verifiable cryptographic hash function h'. NSEC5 operates exactly as NSEC3, but now the attacker cannot evaluate hash function h'() without explicitly issuing a DNS query. Thus, offline dictionary attacks are no longer feasible (only online dictionary attacks are possible).

Question 3

0 / 1 pts



Suppose that the "verifiable hash function" h' above is used for password protection, that is, to conceal the passwords that are stored in an

authentication server. How would this improve security?

'ou Answered

By making it harder to find collisions on passwords.

At the end of the day, the new hash function h' makes use of a traditional hash function h (like SHA-2), so it inherits whatever collision-resistant properties are provided by h, thus nothing changes with respect to the hardness of finding collisions.

By allowing users to get a verifiable proof about why their authentication attempt failed.

orrect Answer

By eliminating offline dictionary attacks.

The crucial property here is that the hash function is not "publicly computable," thus, the attacker cannot precompute hashes to implement a dictionary attack for password cracking.

Question 4

1 / 1 pts

Consider the same setting, where the new "verifiable hash function" h' = h(RSA-Sign_{SK}(x)) is used to conceal the passwords that are stored in an authentication server, where SK is the RSA secret key of the authentication server.

What is the new implementation challenge that is introduced by this approach?

The client performance will drop due to the need to verify a new PKI certificate on every authentication attempt.

Correct!

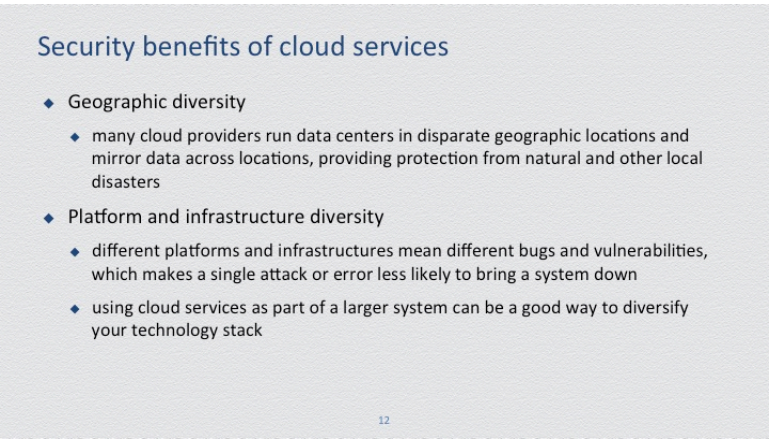
That a second authentication server is required to split the secret state of the system.

Indeed, the secret key SK cannot be locally stored at the authentication server storing the h'-hashed passwords...

The server performance will drop due to the currently expensive RSA signing.

Question 5

0 / 1 pts



What is an example of the security benefits due to diversification when one employs cloud computing services?

orrect Answer

Raising the bar for the adversary.

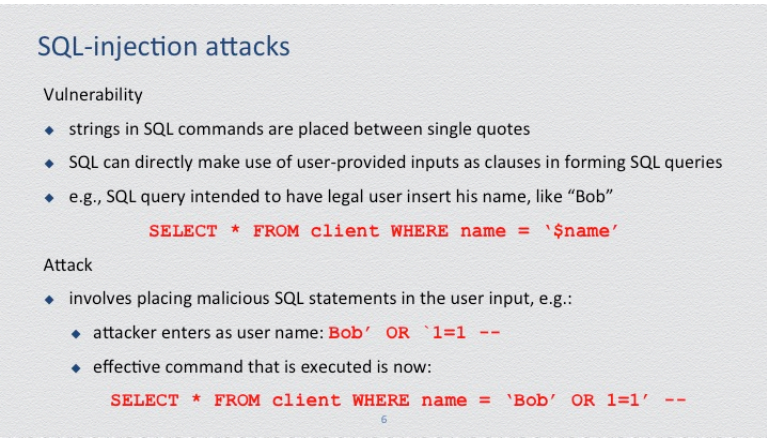
'ou Answered

Increasing availability for the tenant.

Lowering liability for the tenant.

Question 6

0 / 1 pts



SQL injection attacks allow a malicious user to gain unauthorized access to a database by providing malicious data, when prompted to provide his/her authentication credentials, that is, user name and password.

Above "--" denotes the "comment" command in the SQL querying language. How can this seemingly benign command become catastrophic with respect to database security?

orrect Answer

By trivially allowing impersonation.

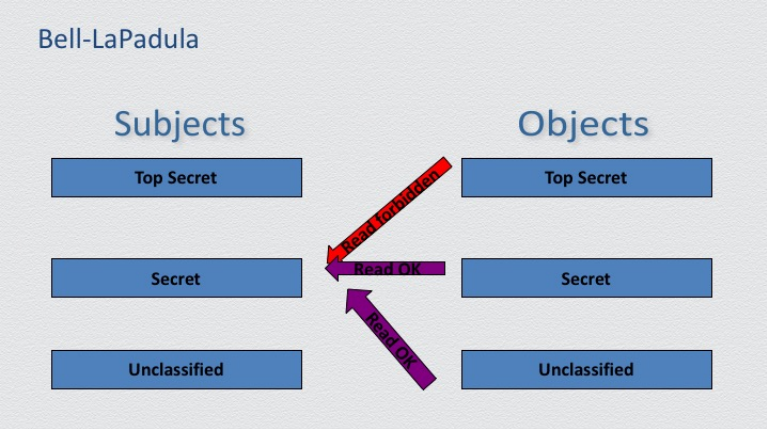
'ou Answered

By harming the availability of the DBMS system.

By weaking the authentication mechanism from 2-factor to 1-factor.

Question 7

1 / 1 pts



What does the "no-read up" property in the above MAC model refer to?

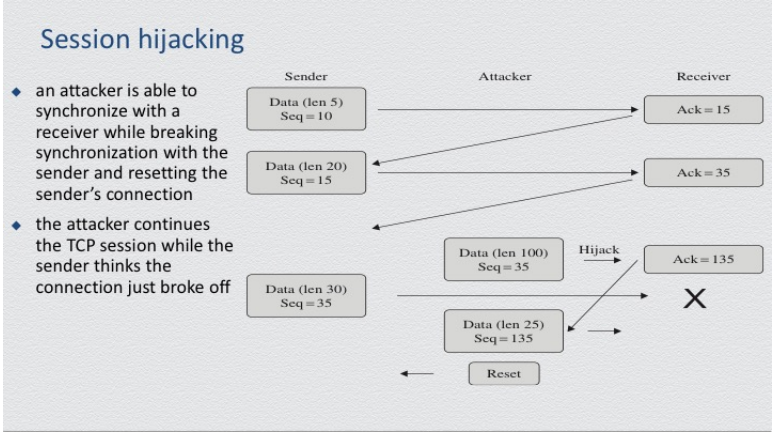
- That no object can read by subjects of greater class.

Correct!

- That any subject can only read objects of equal or lower class.
- That any subject can only write objects of equal or lower class.

Question 8

1 / 1 pts



What type of attack does session hijacking involves?

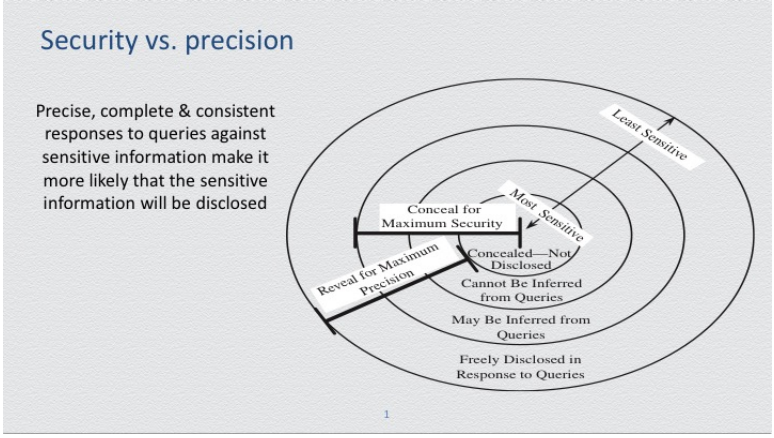
- A replay attack.

Correct!

- (•) A "man or woman-in-the-middle" attack.
- ( ) A length-extension attack.

Question 9

0 / 1 pts



Suppose the records of a database that is outsourced to the cloud, are encrypted using a secret key possessed by the client. What is the utility of this database with respect to data analysis?

- ( ) Limited to simply "get" queries issued by anyone.

orrect Answer

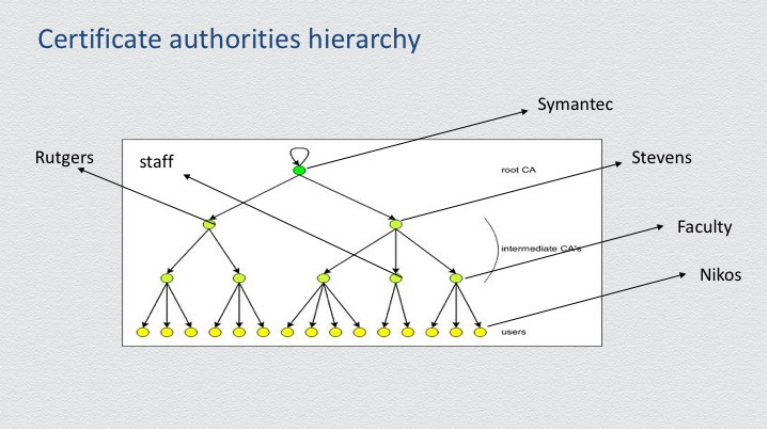
- ( ) Limited to simple "get" queries issued by the client.

'ou Answered

- (•) Limited to SQL queries issued by the client.

Question 10

1 / 1 pts



What is a self-signed certificate?

Correct!

- (•) It is a certificate about a public key of an entity that is signed by the same entity.
- ( ) It is a certificate for the public key of a real person with an identity (e.g., Nikos), not a physical entity (Stevens).
- ( ) It is a certificate on a public key that can only be verified by the entity possessing the corresponding secret key.

Question 11

1 / 1 pts

Buffer overflow attacks

Buffer overflow vulnerability

```
/* stack.c */ /* This program has a buffer overflow vulnerability. */

#include <stdlib.h> #include <stdio.h> #include <string.h>
int func (char *str) {
char buffer[12];
strcpy(buffer, str); /* This statement has a buffer overflow problem */
return 1; }

int main(int argc, char **argv) {
char str[517];
FILE *badfile;
badfile = fopen("badfile", "r");
fread(str, sizeof(char), 517, badfile);
func (str);
printf("Returned Properly\n");
return 1; }
```

What describes the security of running the above program?

- When run by a malicious attacker, this program will compromise the host machine.
- Because buffer overflows are not allowed, the program is secure.

Correct!

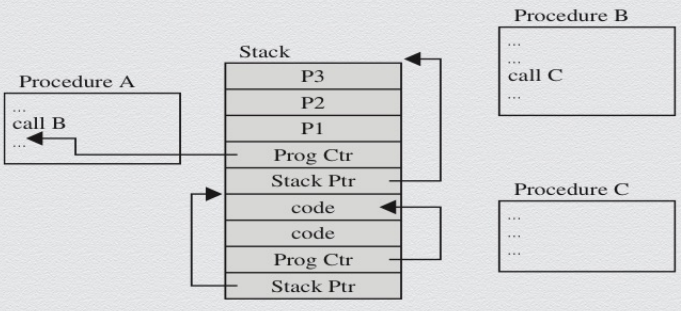
- The program allows for buffer overflows, which can be exploited by a malicious attacker to compromise the host machine.

Question 12

1 / 1 pts

Buffer overflow attacks

Compromised stack



What is the main challenge of the attacker in a successful buffer-overflow attack?

To ensure that the malicious code terminates correctly so that attacked process also terminates.

To write meaningful malicious code that can inclict harm.

Correct!

To overwrite the program counter with the address of the malicious code.

Question 13

0 / 1 pts

Law & legal issues related to IT security

Comparing copyrights, patents, and trade secrets

	Copyright	Patent	Trade Secret
Protects	Expression of idea, not idea itself	Invention—the way something works	A secret, competitive advantage
Protected object made public	Yes; intention is to promote publication	Design filed at Patent Office	No
Requirement to distribute	Yes	No	No
Ease of filing	Very easy, do-it-yourself	Very complicated; specialist lawyer suggested	No filing
Duration	Varies by country; approximately 75–100 years is typical	19 years	Indefinite
Legal protection	Sue if unauthorized copy sold	Sue if invention copied	Sue if secret improperly obtained

A new product makes use of "security by obscurity" in combining and parameterizing standard security practices that are widely used in the industry. What legal protection can be used by the company offering this product?

orrect Answer

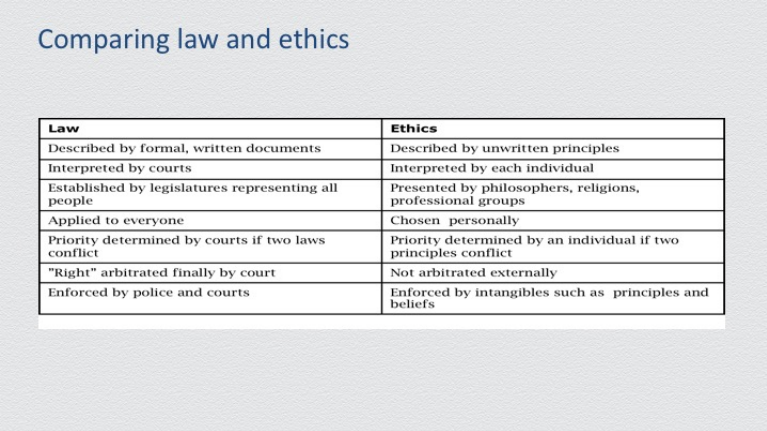
To treat the secret combination and parameterization as trade secret.

To actively seek for violations of copyright rights after the product is launched.

'ou Answered

To file for a patent on the secret combination and parameterization before launching the product.

Ethical issues related to IT security



Free question! Just read the comparison table!