Lab #7 - November 8, 2017

Due Dec 31 at 11:59pm **Points** 12 **Questions** 12
Available Nov 8 at 9:15am - Dec 31 at 11:59pm about 2 months **Time Limit** None
Allowed Attempts Unlimited

Instructions

[1] Good morning and welcome to the seventh 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 this week's lecture.

Take the Quiz Again

Attempt History

	Attempt	Time	Score
LATEST	Attempt 1	9 minutes	9.5 out of 12

Submitted Nov 8 at 1:45pm

Question 1 — 1 / 1 pts

Multiplicative inverses

Definition 1

- The residues modulo a positive integer n comprise set **Z_n = {0,1,2,…,n - 1}**

Definition 2

- If x, y are two elements in Z_n and x is non-zero, such that **x y mod n = 1** then we say that "y is the **multiplicative inverse** of x in Z_n" and write "$y = x^{-1}$"

Example: Multiplicative inverses of the residues modulo 11 (0 does not have)

x	0	1	2	3	4	5	6	7	8	9	10
x^{-1}		1	6	4	3	9	2	8	7	5	10

Multiplicative inverses (cont'ed)

Fact 1

An element x in Z_n has a **multiplicative inverse** if and only if x, n are **relatively prime**

- example: the only elements of Z_{10} having a multiplicative inverse are 1, 3, 7, 9

x	0	1	2	3	4	5	6	7	8	9
x^{-1}		1		7				3		9

Thus

If p is prime, every non-zero residue in Z_p has a multiplicative inverse

Recall the concept of multiplicative inverses that we discussed in class.

Why, whenever p is a prime, does every non-zero element x in Z_p have a multiplicative inverse?

Correct!

- (selected) Because p is a prime number.

  Correct, because every such element x is, by definition, relative prime to p, since x is less than p and p is a prime.

- Because x is non-zero.
- Because x is a residue modulo p.

Question 2

0.5 / 1 pts

Subsets where inverses always exist

Definition 3

- With respect to set of residues modulo Z_n,
  Z_n^* is the **subset** of Z_n containing all integers that are **relative prime** to n

Thus

- If n is a prime, then all non-zero elements in Z_n have an inverse
  - $Z^*_7 = \{1,2,3,4,5,6\}$, n = 7
  - $2 \bullet 4 = 1 \pmod 7$, $3 \bullet 5 = 1 \pmod 7$, $6 \bullet 6 = 1 \pmod 7$, $1 \bullet 1 = 1 \pmod 7$
- If n is not prime, not all integers in Z_n have an inverse
  - $Z^*_{10} = \{1,3,7,9\}$, n = 10
  - $3 \bullet 7 = 1 \pmod{10}$, $9 \bullet 9 = 1 \pmod{10}$, $1 \bullet 1 = 1 \pmod{10}$

Fermat's Little Theorem

Theorem

If p is a prime, then for each non-zero x in Z_p, we have $\mathbf{x^{p-1} \bmod p = 1}$

- Example, p = 5:

$1^4 \bmod 5 = 1$ $\quad$ $2^4 \bmod 5 = 16 \bmod 5 = 1$

$3^4 \bmod 5 = 81 \bmod 5 = 1$ $\quad$ $4^4 \bmod 5 = 256 \bmod 5 = 1$

What is the importance of Fermat's Little Theorem?

Correct!

☑ It provides a way to compute multiplicative inverses in Z_p^*.

Yes, a Corollary of the this theorem give us that:

If p is a prime, then the multiplicative inverse of each non-zero residue x in Z_p is $x^{p-2} \bmod p$.

Because, indeed:
$x(x^{p-2} \bmod p) \bmod p = xx^{p-2} \bmod p = x^{p-1} \bmod p = 1$

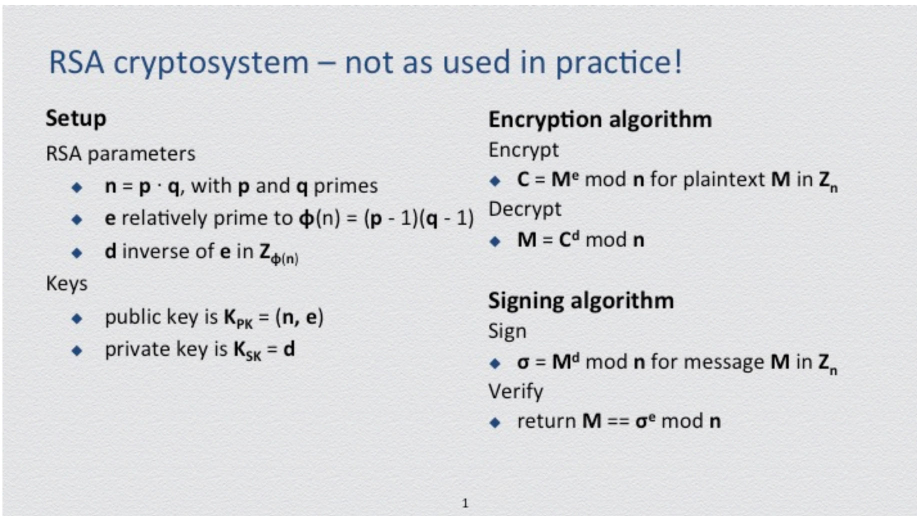
- [ ] It is not of importance for p being a prime.

orrect Answer

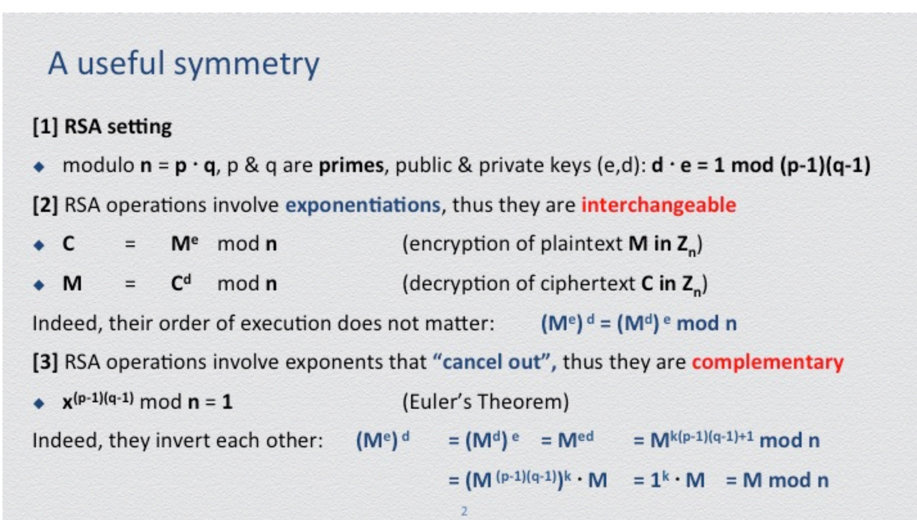
- [ ] It provides a way to compute powers more efficiently.

Question 3

0 / 1 pts



The "pure" or "plain" RSA cryptosystem is described by the above exponentiation algorithms, where interestingly the same public key and secret key are used both for encryption and for signatures (of course, in reserve order). Importantly, when applied consequentially, these exponentiation algorithms have a "cancellation" property in the exponent, as explain below:



Intuitively, M^{ed} = M mod n because M^{ed} mod n equals "1" times "M". What contributes these two factors 1 and M?

orrect Answer

1 results from Euler's theorem; and M results from the fact that d, e are inverses modulo φ(n).

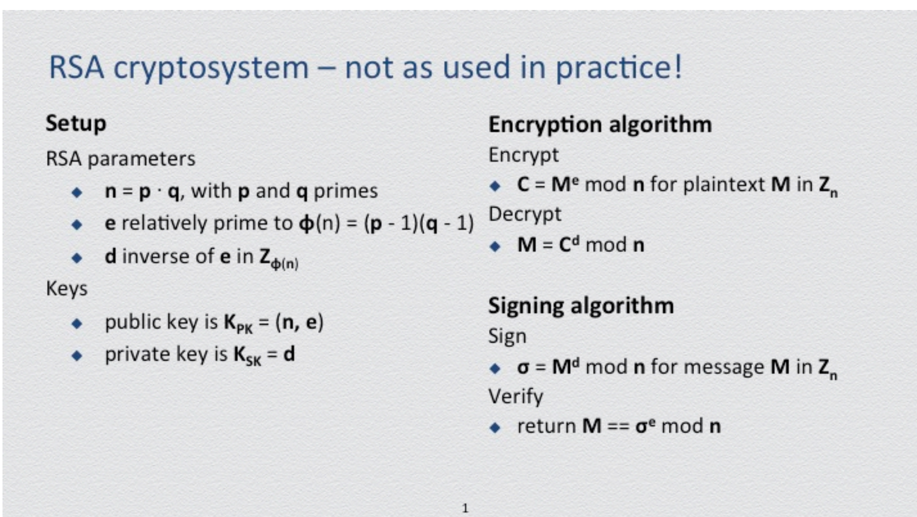
'ou Answered

1 results from the fact that M is relative prime to n; and M results from the fact that d, e are inverses modulo φ(n).

1 results from the fact that d, e are inverses modulo φ(n); and M results from the fact that M is the base in the power M^{ed}.

Question 4

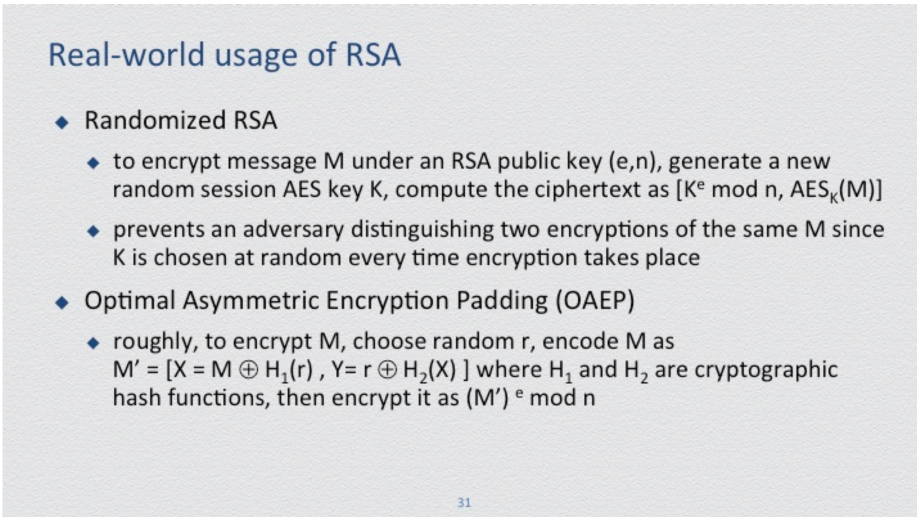
1 / 1 pts



"Plain" RSA comprise only core algorithms that (through their useful symmetry) lend themselves to the design of an RSA public-key encryption scheme and an RSA signature scheme that are used in practice. These real-world schemes are different than plain RSA, however, in that they process (encrypt or sign) a message M not "as is," but instead as a new message M'.

That is, in practice, the RSA exponentiation functions are applied rather to new message M' that is a transformation of the original message M.

In the case of real-world RSA encryption, this message transformation includes:



Why M is encrypted as M' (e.g., as in Padded RSA or RSA-OAEP)?

- Because M must be relative prime to φ(n).
- Because M must be relative prime to n.

Correct!

- Because plain RSA encryption does not provide protections against recovery of all messages, partial plaintext leakage or chosen-plaintext attacks.

Indeed, the RSA assumption states that, as long as factoring n into p and q is an infeasible task, a random message M in Z_n^* cannot be inferred given M^e mod n. Unfortunately, this does not exclude the possibility that message recovery is possible for specific values of M or that partial information can be inferred about M (e.g., if M is even or odd number). Moreover, plain RSA encryption is deterministic, therefore it cannot possibly protect against chosen-plaintext attacks - e.g., any indistinguishability-based security notion cannot be achieved since the attacker can contrast the challenge ciphertext against the ciphertext of any of the messages M1 or M2 that he adversarially chose and provided to the challenger.

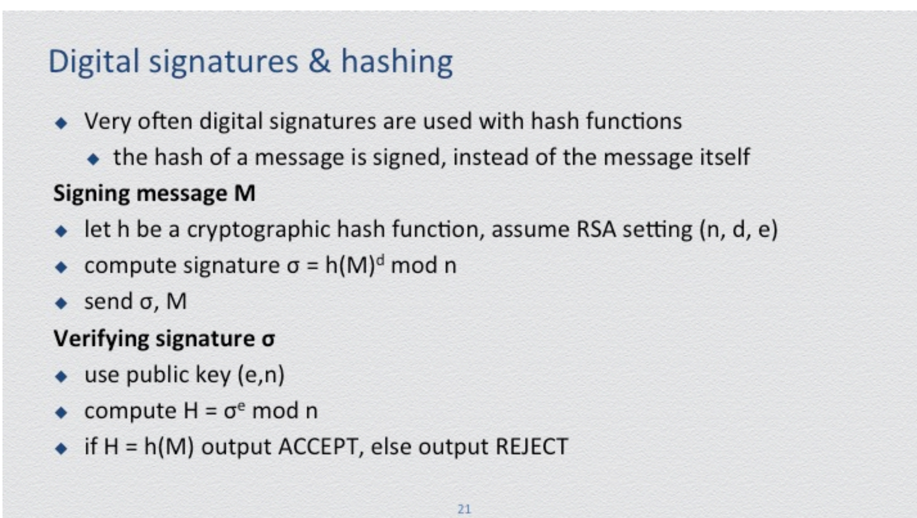
Because smaller values of M are unsafe to be encrypted using plain RSA.

The RSA assumption states that, as long as factoring n into p and q is an infeasible task, a random message M in Z_n^* cannot be inferred given M^e mod n. Unfortunately, this does not exclude the possibility that message recovery is possible for specific values of M or that partial information can be inferred about M (e.g., if M is even or odd number). Moreover, plain RSA encryption is deterministic, therefore it cannot possibly protect against chosen-plaintext attacks - e.g., any indistinguishability-based security notion cannot be achieved since the attacker can contrast the challenge ciphertext against the ciphertext of any of the messages M1 or M2 that he adversarially chose and provided to the challenger.

Question 5

1 / 1 pts

Analogously, in real-world usage of the above RSA signing algorithm, we employ the "hash & sign" paradigm:



Why M is signed as M' = h(M) (e.g., as in RSA-FDH)?

For efficiency reasons, because, as M is typically a large integer, it helps to reduce it to a smaller value M'.

Correct!

Because plain RSA signatures allows for trivial forgeries.

Note that the pair (S^e mod n, S) is a valid message-signature pair for any value S in Z_n^*. Indeed, $(S)^e$ mod n = S^e as required for a valid signature. Also, using the partial homomorphic property of the RSA function, an attacker knowing the signatures σ_1, σ_2 of messages M_1, M_2, can forge signature $\sigma_1 * \sigma_2$ for message $M_1 * M_2$.

For efficiency reasons, because, as M has typically large size, it helps to reduce it to a smaller value M' in Z_n^*.

Plain RSA signatures allows for trivial forgeries.

Note that the pair (S^e mod n, S) is a valid message-signature pair for any value S in Z_n^*. Indeed, $(S)^e$ mod n = S^e as required for a valid signature. Also, using the partial homomorphic property of the RSA function, an attacker knowing the signatures σ_1, σ_2 of messages M_1, M_2, can forge signature $\sigma_1 * \sigma_2$ for message $M_1 * M_2$.

Question 6

1 / 1 pts

RSA-FDH, mentioned in the previous question, is the "hash-and-sign" extension of plain RSA, where message M is signed as M' = h(M), where h is an appropriate "full-domain" cryptographic hash function mapping messages uniformly onto Z_n^*. That is, the signature of M is $\sigma = h(M)^d$ mod n, which is verified by checking whether $\sigma^e = h(M)$ mod n.

How does this "hash-and-sign" extension improve plain RSA signatures?

Primarily, by reducing RSA to use of purely symmetric crypto-primitives.

Correct!

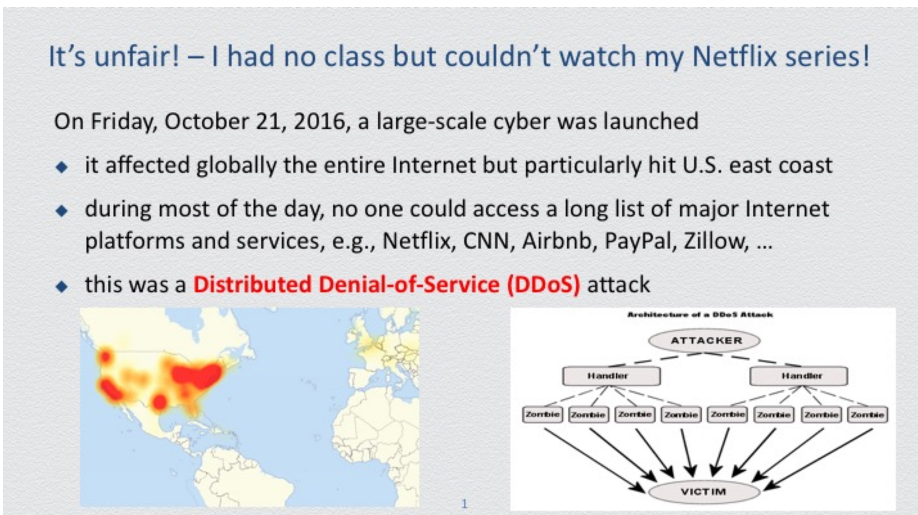
Primarily, by providing stronger security.

Indeed, hashing the message before their are being signed makes infeasible to forge signatures by solely using the public key or the partially homomorphic properties of the RSA function.

Primarily, by providing better efficiency.

Question 7

1 / 1 pts



Please read the brief Wikipedia entry available at:

[https://en.wikipedia.org/wiki/2016_Dyn_cyberattack (https://en.wikipedia.org/wiki/2016_Dyn_cyberattack)](https://en.wikipedia.org/wiki/2016_Dyn_cyberattack)

Which main security property does a Denial-of-Service (DoS) attack attempt to defeat?

Integrity; services or data are modified by an unauthorized user.

Confidentiality; services or data are accessed by an unauthorized user.

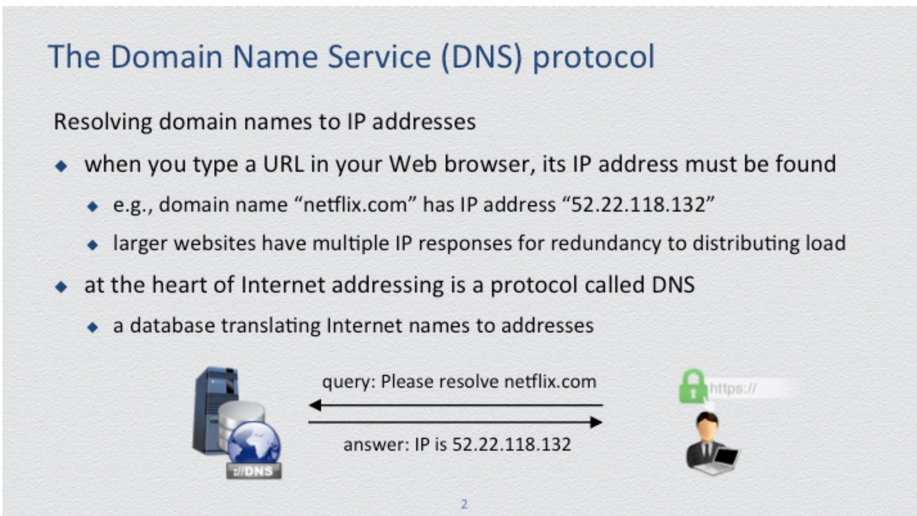
Correct!

Availability; a user is denied access to authorized services or data.

Indeed, the main goal in a DoS attack is implied by its name itself: Availability is concerned with preserving authorized access to assets and a DoS attack aims against this property.

Question 8

1 / 1 pts



What main security property or properties must be preserved in such an important service?

Correct!

All properties in CIA triad.

Indeed, resolving domain names to IP addresses is a service that: (1) must critically be available during all times (availability); (2) must be trustworthy (integrity - or else connections to malicious sites may occur, e.g., as in a DNS-spoofing attacks); and (3) must also protect database entries that are not queried (confidentiality - or else an attacker may find out about the structure of a target organization, e.g., zone-enumeration attacks).

Integrity.

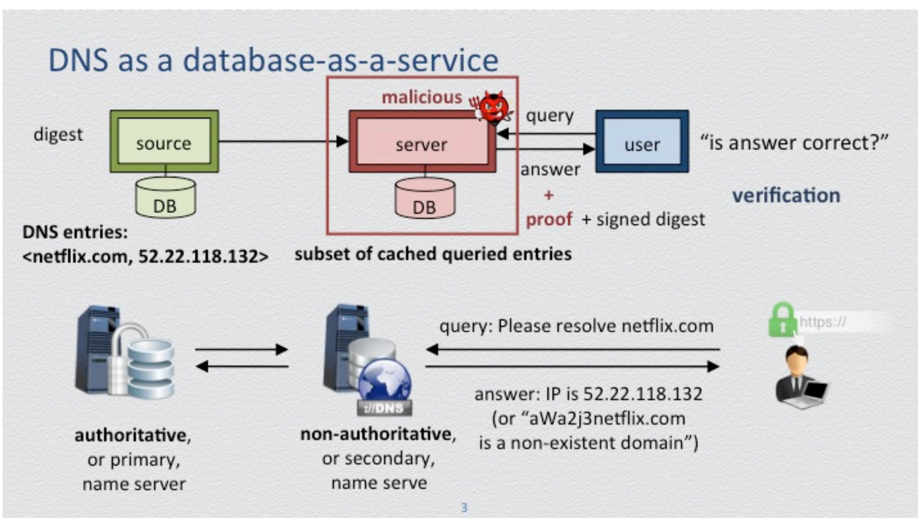
Availability.

Confidentiality.

All main properties of the CIA triad must be satisfied. Resolving domain names to IP addresses is a service that: (1) must critically be available during all times (availability); (2) must be trustworthy (integrity - or else connections to malicious sites may occur, e.g., as in a DNS-spoofing attacks); and (3) must also protect database entries that are not queried (confidentiality - or else an attacker may find out about the structure of a target organization, e.g., zone-enumeration attacks).

Question 9

1 / 1 pts



As discussed in class, DNS resembles the database-as-a-service authentication model. Note that queries may have either positive verifiable answers (an existing domain name with a valid IP address, supported in DNSSEC protocol) or negative verifiable answers (an non-existing domain name with no valid IP address, also supported in NSEC) which are provided as signed "hit key-value" or "near-miss neighboring-key" pairs.

Why DNS uses non-authoritative name servers?

Correct!

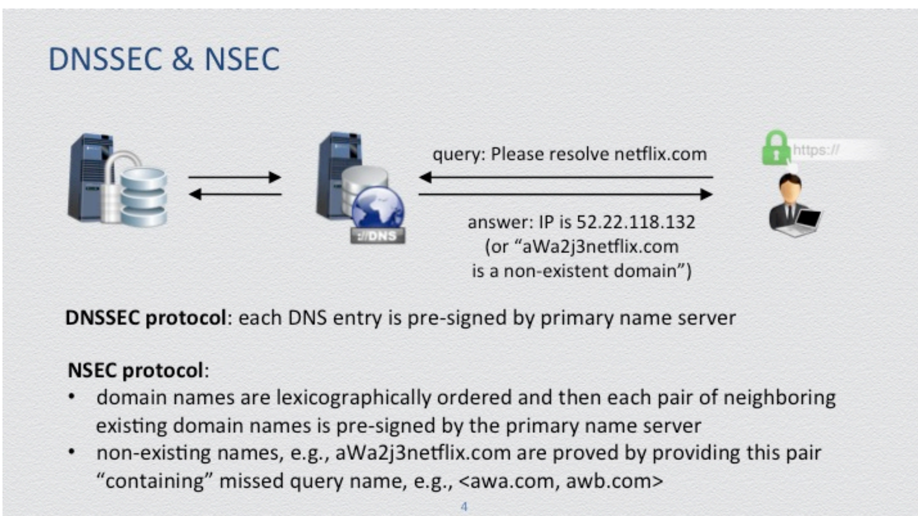
- For more scalability and locality.

Indeed, high traffic loads can saturate the response capacity of authoritative name servers. Also secondary name servers may only cache recent queried domain names without having to store large volumes of DNS entries.

- For added locality.
- For more scalability.
- For added security.

Question 10

0 / 1 pts



What motivated the development of the NSEC protocol as an extension of DNSSEC and what did it result to?

orrect Answer

- Verification of negative answers & a new privacy vulnerability.

'ou Answered

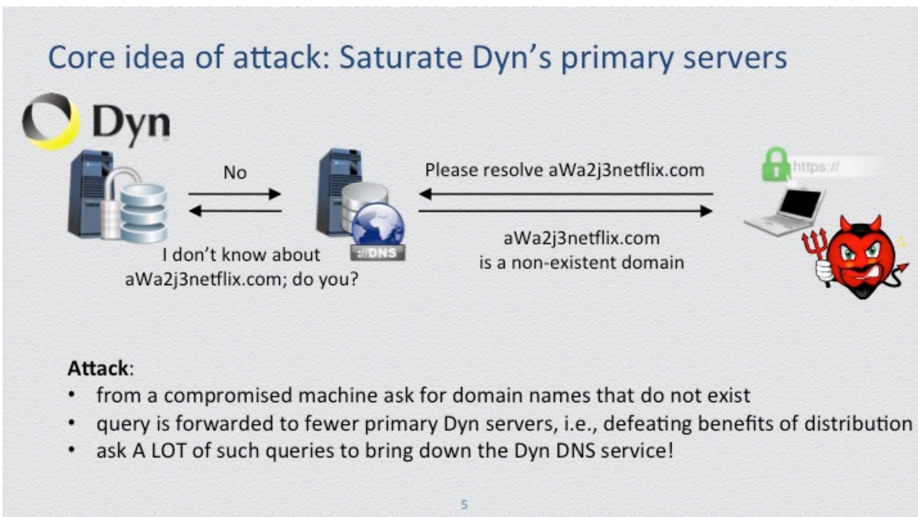
- Lack of efficiency in DNSSEC & stronger integrity protection.

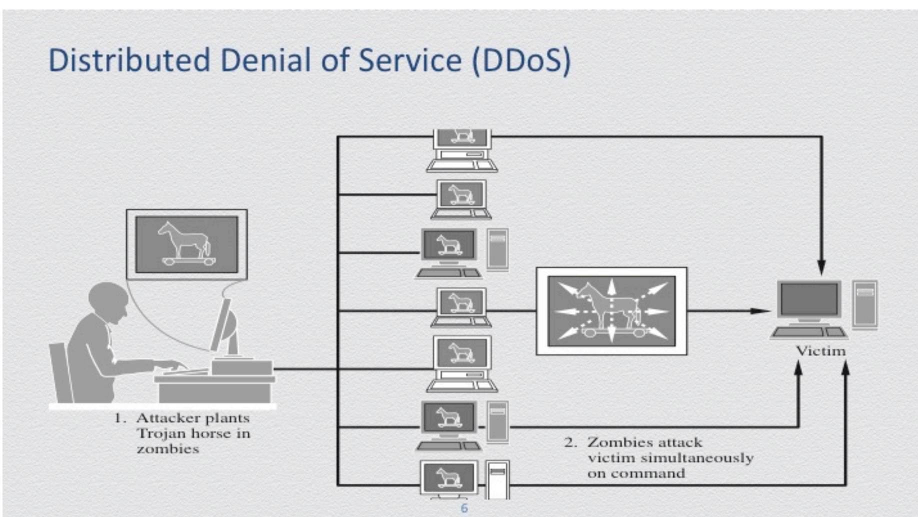
No, the goal was to support verification of non-set membership, but it resulted in introducing a new vulnerability w.r.t. the confidentiality of domain names.

Leakage of unqueried domain names & stronger confidentiality.

Question 11

1 / 1 pts





The attack's core idea is as above. But in practice a distributed DoS attack was launched, involving employing a large army of compromised devices, called zombies, which comprise a botnet that is controlled by the attacker, and bombarding the Dyn servers with millions of DNS queries...

Why a botnet is necessary for an effective DoS attack?

- Avoid effective countermeasures.

Correct!

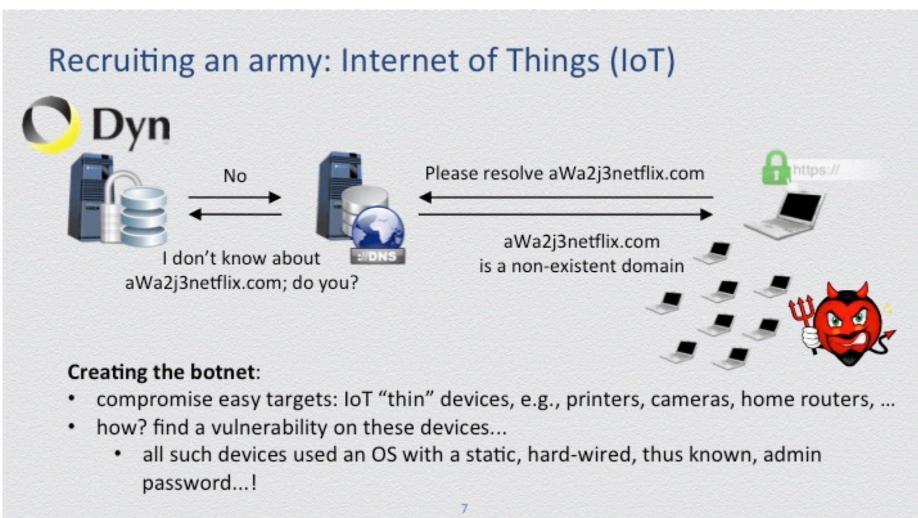
- Avoid effective countermeasures and increase "attack" traffic.

Yes, if the high-volume "attack" traffic comes from few devices, these devices can be filtered out by blocking their connections to the Dyn servers. Also, by employing a large botnet of million of devices the attacker inflicts a larger, more devastating "attack" traffic against the victim Dyn servers.

- Increase "attack" traffic.

Question 12

1 / 1 pts



In the Dyn DDOS attack, the recruited zombie machines were IoT devices that were compromised using the [Mirai (http://https://en.wikipedia.org/wiki/Mirai_(malware))](http://https://en.wikipedia.org/wiki/Mirai_(malware)) malware.

What does the above attack method teache us?

Password security is an important issue.

Correct!

IoT security and password security are important issues.

Yes. IoT devices are "thin," not property administrated, but they are easy targets and interconnected! Passwords are the primary user-authentication method and thus their security is crucial (as we will see next week)!

IoT security is an important issue.