Introduction to cryptography

Quiz

Gilles Van Assche
Christophe Petit

INFO-F-405
Université Libre de Bruxelles
2023-2024

Perfect secrecy

Which assertion is correct?
(Note: unconditional security = perfect secrecy.)

A. A cipher is unconditionally secure **implies that** the secret key is at least as long as the plaintext.

B. A cipher is unconditionally secure **as soon as** the secret key is at least as long as the plaintext.

C. A cipher is unconditionally secure **if and only if** the secret key is at least as long as the plaintext.

Perfect secrecy

Which assertion is correct?
(Note: unconditional security = perfect secrecy.)

A. A cipher is unconditionally secure **implies that** the secret key is at least as long as the plaintext.

B. A cipher is unconditionally secure **as soon as** the secret key is at least as long as the plaintext.

C. A cipher is unconditionally secure **if and only if** the secret key is at least as long as the plaintext.

The correct answer is **A**. It is easy to build a cipher with a key longer than the plaintext that does not achieve perfect secrecy.

Not-so-one-time pad

What happens if the one-time pad is incorrectly used and that two distinct plaintexts are encrypted with the same key?

A. The key is compromised.

B. The two plaintexts are revealed.

C. The difference between the two plaintexts is revealed.

D. The authenticity of the plaintext is compromised.

Not-so-one-time pad

What happens if the one-time pad is incorrectly used and that two distinct plaintexts are encrypted with the same key?

A. The key is compromised.

B. The two plaintexts are revealed.

C. The difference between the two plaintexts is revealed.

D. The authenticity of the plaintext is compromised.

The correct answer is **C**.

Computational security

Assume an adversary performs an exhaustive key search on a huge network of 10^9 computers, each capable of testing 10^9 keys per second. After about how much time will a 128-bit key typically be found?

A. A few seconds.

B. A few days.

C. A few years.

D. A few centuries.

E. A few times the age of the universe.

Computational security

Assume an adversary performs an exhaustive key search on a huge network of 10^9 computers, each capable of testing 10^9 keys per second. After about how much time will a 128-bit key typically be found?

A. A few seconds.

B. A few days.

C. A few years.

D. A few centuries.

E. A few times the age of the universe.

The correct answer is **E**.

Nonce

What does “nonce” stand for?

A. Number used only once
B. Non-committing encryption
C. Network neutrality for confidentiality and encryption

Nonce

What does “nonce” stand for?

- A Number used only once
- B Non-committing encryption
- C Network neutrality for confidentiality and encryption

The correct answer is **A**.

Semantic security / IND-CPA

For a cipher to achieve semantic security (or equivalently, to be IND-CPA secure), which condition is necessary?

A. It must be randomized.

B. It must be randomized (if asymmetric) or the diversifier must be a nonce (if symmetric).

C. It must ensure that one cannot recognize whether two identical plaintexts were encrypted with the same key.

D. None of the above.

Semantic security / IND-CPA

For a cipher to achieve semantic security (or equivalently, to be IND-CPA secure), which condition is necessary?

- A It must be randomized.
- B It must be randomized (if asymmetric) or the diversifier must be a nonce (if symmetric).
- C It must ensure that one cannot recognize whether two identical plaintexts were encrypted with the same key.
- D None of the above.

The correct answer is **C**. Answers A and B are sufficient conditions that ensure the property C.

Authentication

For an authentication scheme to be secure (EU-CMA), which condition is necessary?

A. It must be randomized.

B. It must be randomized (if asymmetric) or the diversifier must be a nonce (if symmetric).

C. It must ensure that one cannot tell whether two identical messages were signed / MAC'ed with the same key.

D. None of the above.

Authentication

For an authentication scheme to be secure (EU-CMA), which condition is necessary?

A. It must be randomized.

B. It must be randomized (if asymmetric) or the diversifier must be a nonce (if symmetric).

C. It must ensure that one cannot tell whether two identical messages were signed / MAC'ed with the same key.

D. None of the above.

The correct answer is **D**. The message is public, so it is fine if the tag is a deterministic function of the message (and of the key).

“Le Big-MAC”

What happens if a MAC or a signature is too short? An attacker ...

A ... can create a fraudulent message, randomly guess the tag and have some chance that it is valid?

B ... can modify a legitimate message, randomly guess the tag and have some chance that it is valid?

C ... can modify a legitimate message, keep the same tag and have some chance that it is valid?

D ... can recover the key too easily?

E ... can decrypt the message too easily?

“Le Big-MAC”

What happens if a MAC or a signature is too short? An attacker ...

- A ... can create a fraudulent message, randomly guess the tag and have some chance that it is valid?
- B ... can modify a legitimate message, randomly guess the tag and have some chance that it is valid?
- C ... can modify a legitimate message, keep the same tag and have some chance that it is valid?
- D ... can recover the key too easily?
- E ... can decrypt the message too easily?

The correct answer is **A+B+C**. From the point of view of authentication, a message created by an attacker and a legitimate message turned into a fraudulent one after modification are both forgeries. A too short tag can always be randomly guessed—keeping the original tag is as good a guess as any other.

Exhausted DES

The DES has 56 bits of key. What is the complexity of exhaustive key search on it?

A 2^{56} operations

B 2^{55} operations because it has (semi-)weak keys

C 2^{55} operations because of the complementarity property

Exhausted DES

The DES has 56 bits of key. What is the complexity of exhaustive key search on it?

A 2^{56} operations

B 2^{55} operations because it has (semi-)weak keys

C 2^{55} operations because of the complementarity property

The correct answer is **C**.

Exhausted Double-DES

To avoid exhaustive key search and have 112 bits of key, one can use $\mathrm{DES}_{k_1} \circ \mathrm{DES}_{k_2}$. What is the complexity of exhaustive key search on it?

A. 2^{112} operations

B. 2^{110} operations because of the complementarity property

C. 2^{57} operations and negligible memory

D. 2^{57} operations and 56 terabytes of memory

E. 2^{57} operations and 1080 petabytes of memory

F. None of the above

Exhausted Double-DES

To avoid exhaustive key search and have 112 bits of key, one can use $\mathrm{DES}_{k_1} \circ \mathrm{DES}_{k_2}$. What is the complexity of exhaustive key search on it?

- A 2^{112} operations
- B 2^{110} operations because of the complementarity property
- C 2^{57} operations and negligible memory
- D 2^{57} operations and 56 terabytes of memory
- E 2^{57} operations and 1080 petabytes of memory
- F None of the above

The correct answer is **E**. The meet-in-the-middle attack requires two loops of 2^{56} iterations and a table of 2^{56} entries of $7 + 8$ bytes each.

Exhausted Triple-DES

To avoid exhaustive key search and have 112 bits of key, one can use $\mathrm{DES}_{k_1} \circ \mathrm{DES}_{k_2} \circ \mathrm{DES}_{k_1}$. What is the complexity of exhaustive key search on it?

- A 2^{112} operations
- B 2^{112} operations, but it is susceptible to differential cryptanalysis
- C 2^{112} operations, but it is susceptible to linear cryptanalysis
- D 2^{110} operations because of the complementarity property
- E 2^{57} operations and 1080 petabytes of memory
- F None of the above

Exhausted Triple-DES

To avoid exhaustive key search and have 112 bits of key, one can use $\mathrm{DES}_{k_1} \circ \mathrm{DES}_{k_2} \circ \mathrm{DES}_{k_1}$. What is the complexity of exhaustive key search on it?

- **A** 2^{112} operations
- **B** 2^{112} operations, but it is susceptible to differential cryptanalysis
- **C** 2^{112} operations, but it is susceptible to linear cryptanalysis
- **D** 2^{110} operations because of the complementarity property
- **E** 2^{57} operations and 1080 petabytes of memory
- **F** None of the above

The correct answer is **A**. Triple-DES is safe.

Rijndael

In Rijndael, if we change one byte in the input block, how many bytes are *guaranteed* to change in the state after 2 rounds?

A 1

B 4

C 8

D 16

Rijndael

In Rijndael, if we change one byte in the input block, how many bytes are *guaranteed* to change in the state after 2 rounds?

A 1

B 4

C 8

D 16

The correct answer is **D**.

- After the first MixColumns, 4 bytes in the same column are guaranteed to change.
- ShiftRows will move the 4 changed bytes to different columns.
- The second MixColumns will change 4 bytes in each of the 4 columns.

Primitive

In this course, how is a symmetric crypto *primitive* defined?

A. It is an algorithm whose security cannot be proven but must tested with third-party cryptanalysis.

B. It is the set of elementary operations that must be performed in an encryption or authentication scheme.

C. It is a painter in the Renaissance.

Primitive

In this course, how is a symmetric crypto *primitive* defined?

A. It is an algorithm whose security cannot be proven but must tested with third-party cryptanalysis.

B. It is the set of elementary operations that must be performed in an encryption or authentication scheme.

C. It is a painter in the Renaissance.

The correct answer is **A**.

Mode of operation

What is a mode of operation?

A. The formal security requirements in which an encryption or authentication scheme must operate.

B. An algorithm that implements any type of scheme or another primitive by using a primitive as a black box.

Mode of operation

What is a mode of operation?

A The formal security requirements in which an encryption or authentication scheme must operate.

B An algorithm that implements any type of scheme or another primitive by using a primitive as a black box.

The correct answer is **B**.

Is your attack generic enough?

In symmetric crypto, consider a scheme that is made of a mode of operation on top of a primitive. A generic attack is ...

A ... an attack on the primitive that works independently of the mode of operation?

B ... an attack on the mode of operation that works independently of the primivite?

Is your attack generic enough?

In symmetric crypto, consider a scheme that is made of a mode of operation on top of a primitive. A generic attack is ...

A ... an attack on the primitive that works independently of the mode of operation?

B ... an attack on the mode of operation that works independently of the primivite?

The correct answer is **B**.

Birthday paradox

If we draw 12-digit numbers at random (e.g., 805916763814) with replacement, after how many draws is it likely to have two identical such numbers in the list (i.e., a collision)?

A After 10^3 draws.

B After 10^6 draws.

C After 10^9 draws.

D After $\frac{10^{12}}{2}$ draws.

Birthday paradox

If we draw 12-digit numbers at random (e.g., 805916763814) with replacement, after how many draws is it likely to have two identical such numbers in the list (i.e., a collision)?

A After 10^3 draws.

B After 10^6 draws.

C After 10^9 draws.

D After $\frac{10^{12}}{2}$ draws.

The correct answer is **B**. We have $N = 10^{12}$ possible values, so a collision is likely to appear after $\sqrt{N} = 10^{12/2} = 10^6$ draws.

Salvaging ECB

Assume a user encrypts plaintext using AES-ECB, and the plaintext is highly compressed data. Is this encryption secure in a known plaintext setting?

A. No, it is totally insecure.

B. Yes, it is secure, but up to the birthday bound (2^{64} blocks)

C. Yes, it is secure all the way up to 2^{128} blocks

Salvaging ECB

Assume a user encrypts plaintext using AES-ECB, and the plaintext is highly compressed data. Is this encryption secure in a known plaintext setting?

A. No, it is totally insecure.

B. Yes, it is secure, but up to the birthday bound (2^{64} blocks)

C. Yes, it is secure all the way up to 2^{128} blocks

The correct answer is **B**. ECB will reveal the value of a plaintext block only if the same block occurs. Since the plaintext is highly compressed, they look like random 128-bit values, and information can be revealed only if there is a collision among the plaintext (or ciphertext) blocks.

Inverse-less block ciphers

In which of these block cipher-based mode(s) is the implementation of the inverse block cipher not needed?

A ECB

B CBC

C CTR

D CBC-MAC

Inverse-less block ciphers

In which of these block cipher-based mode(s) is the implementation of the inverse block cipher not needed?

A ECB

B CBC

C CTR

D CBC-MAC

The correct answer is **C and D**. CTR does not need the inverse to decrypt, as the block cipher is only used to generate the keystream. CBC-MAC does not need the inverse, as the verification requires to re-compute the MAC.

Sponges

The sponge construction is

A. a mode on top of a keystream generator
B. a mode on top of a block cipher
C. a mode on top of a permutation
D. a factory that produces kitchen appliances

Sponges

The sponge construction is

A a mode on top of a keystream generator

B a mode on top of a block cipher

C a mode on top of a permutation

D a factory that produces kitchen appliances

The correct answer is **C**.

Sponges vs keyed sponges

The difference between a sponge function and a keyed sponge function is:

A. The keyed sponge has one more input, the secret key, that is concatenated to the input string before being absorbed.

B. The keyed sponge allows to alternatively absorb input blocks and request output blocks.

Sponges vs keyed sponges

The difference between a sponge function and a keyed sponge function is:

A. The keyed sponge has one more input, the secret key, that is concatenated to the input string before being absorbed.

B. The keyed sponge allows to alternatively absorb input blocks and request output blocks.

The correct answer is **A**. The keyed sponge construction can be seen as a thin mode layer on top of the plain sponge construction. It is the duplex construction that allows to alternatively absorb input blocks and request output blocks.

Hashing more

SHA-256 is a well-known hash function with 256 bits of output. Let us build the $n = 512$-bit hash function SHA-256+256 this way:

$$\text{SHA-256+256}(x) = \text{SHA-256}(x||0) \,||\, \text{SHA-256}(x||1).$$

What is the collision resistance of SHA-256+256?

A. 64 bits
B. 128 bits
C. 192 bits
D. 256 bits
E. 384 bits
F. 512 bits

Hashing more

SHA-256 is a well-known hash function with 256 bits of output. Let us build the $n = 512$-bit hash function SHA-256+256 this way:

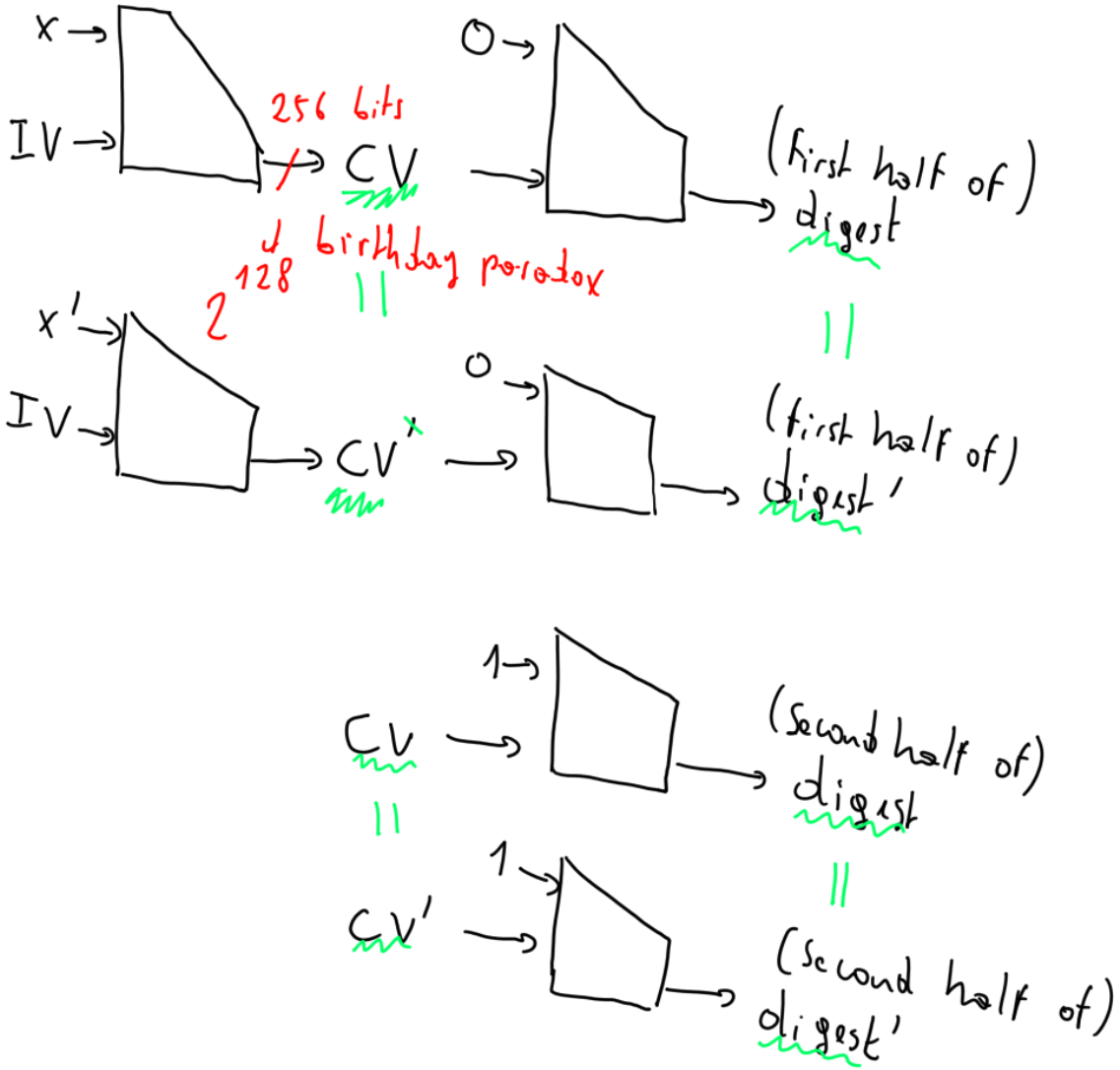
$$\text{SHA-256+256}(x) = \text{SHA-256}(x||0) \;||\; \text{SHA-256}(x||1).$$

What is the collision resistance of SHA-256+256?

- A 64 bits
- B 128 bits
- C 192 bits
- D 256 bits
- E 384 bits
- F 512 bits

The correct answer is **B**. The attacker can find a collision in the chaining value (CV) when processing x. The size of the CV is 256 bits, so the collision resistance is 128 bits.

Hashing more



MD4, MD5, SHA-1

What is the status of these hash functions w.r.t. collision resistance?

A. SHA-1 is theoretically broken, but not MD4 nor MD5
B. MD5 and SHA-1 are theoretically broken, but not MD4
C. MD4, MD5 and SHA-1 are theoretically broken
D. MD4, MD5 and SHA-1 are practically broken

MD4, MD5, SHA-1

What is the status of these hash functions w.r.t. collision resistance?

- A SHA-1 is theoretically broken, but not MD4 nor MD5
- B MD5 and SHA-1 are theoretically broken, but not MD4
- C MD4, MD5 and SHA-1 are theoretically broken
- D MD4, MD5 and SHA-1 are practically broken

The correct answer is **D**. For these three hash functions, there are concrete examples of collisions.

Indifferentiability

Which statement(s) is/are correct?

- A SHA-1 ...
- B SHA-256 and SHA-512 ...
- C The Merkle-Damgård construction ...
- D SHA-3 ...
- E The sponge construction ...

is/are indifferentiable from a random oracle up to complexity $2^{n/2}$ (A-C) or $2^{c/2}$ (D-E).

Indifferentiability

Which statement(s) is/are correct?

- **A** SHA-1 ...
- **B** SHA-256 and SHA-512 ...
- **C** The Merkle-Damgård construction ...
- **D** SHA-3 ...
- **E** The sponge construction ...

is/are indifferentiable from a random oracle up to complexity $2^{n/2}$ (A-C) or $2^{c/2}$ (D-E).

The only correct statement is **E**. Only the mode can be proven indifferentiable. And Merkle-Damgård is not indifferentiable due to the length extension weakness.

Keccak inverse

When inverting Keccak, which step is the most costly one?

A θ

B ρ

C π

D χ

E ι

F None of the above

KECCAK inverse

When inverting KECCAK, which step is the most costly one?

A θ

B ρ

C π

D χ

E ι

F None of the above

The correct answer is **F**. The question was tricky on purpose: There is no KECCAK inverse, and the inverse permutation is never invoked in a sponge. Nevertheless, KECCAK-f^{-1} exists and in general θ is the most costly step to invert.

Web of trust

I have the following public keys:

- $\mathrm{PK}_{\mathrm{Xavier}}$ checked and signed by me
- $\mathrm{PK}_{\mathrm{Yves}}$ and $\mathrm{Sign}_{\mathrm{SK}_{\mathrm{Xavier}}}(\mathrm{Yves}, \mathrm{PK}_{\mathrm{Yves}})$
- $\mathrm{PK}_{\mathrm{Zoë}}$ and $\mathrm{Sign}_{\mathrm{SK}_{\mathrm{Yves}}}(\mathrm{Zoë}, \mathrm{PK}_{\mathrm{Zoë}})$

Who do I have to trust so that can I trust Zoë's public key?

A. Xavier only
B. Yves only
C. Both Xavier and Yves

Web of trust

I have the following public keys:

- $\mathrm{PK}_{\mathrm{Xavier}}$ checked and signed by me
- $\mathrm{PK}_{\mathrm{Yves}}$ and $\mathrm{Sign}_{\mathrm{SK}_{\mathrm{Xavier}}}(\mathrm{Yves}, \mathrm{PK}_{\mathrm{Yves}})$
- $\mathrm{PK}_{\mathrm{Zoë}}$ and $\mathrm{Sign}_{\mathrm{SK}_{\mathrm{Yves}}}(\mathrm{Zoë}, \mathrm{PK}_{\mathrm{Zoë}})$

Who do I have to trust so that can I trust Zoë's public key?

A. Xavier only
B. Yves only
C. Both Xavier and Yves

The correct answer is **C**.

If Xavier only, it tells me that I can trust Yves' public key, but not the fact that Yves actually checked Zoë's public key.

If Yves only, nothing tells me that Yves' public key belongs to him, so Yves' signature on Zoë's public key might actually not be from Yves.

Hybrid encryption

How does hybrid encryption work?

A. Alice sends a public key to Bob encrypted with their secret key, and she encrypts the plaintext with his public key.

B. Alice sends a secret key to Bob encrypted with his public key, and she encrypts the plaintext with the secret key.

C. Alice sends a public key to Bob encrypted with **her** private key, and she encrypts the plaintext with **her** public key.

D. Alice sends a private key to Bob encrypted with his public key, and she encrypts the plaintext with his public key.

E. Alice sends a secret key to Bob encrypted with **her** private key, and she encrypts the plaintext with the secret key.

Hybrid encryption

How does hybrid encryption work?

A. Alice sends a public key to Bob encrypted with their secret key, and she encrypts the plaintext with his public key.

B. Alice sends a secret key to Bob encrypted with his public key, and she encrypts the plaintext with the secret key.

C. Alice sends a public key to Bob encrypted with **her** private key, and she encrypts the plaintext with **her** public key.

D. Alice sends a private key to Bob encrypted with his public key, and she encrypts the plaintext with his public key.

E. Alice sends a secret key to Bob encrypted with **her** private key, and she encrypts the plaintext with the secret key.

The correct answer is **B**.

RSA key generation

In the scope of RSA, let $n = pq$ be the product of two distinct large primes. How do the public and private exponents relate? (There can be more than one possible answer.)

A $ed \equiv 1 \pmod{n}$

B $ed \equiv 1 \pmod{\phi(n)}$

C $ed \equiv 1 \pmod{(p-1)(q-1)}$

RSA key generation

In the scope of RSA, let $n = pq$ be the product of two distinct large primes. How do the public and private exponents relate? (There can be more than one possible answer.)

A $ed \equiv 1 \pmod{n}$

B $ed \equiv 1 \pmod{\phi(n)}$

C $ed \equiv 1 \pmod{(p-1)(q-1)}$

The correct answer is **B and C**.

RSA key length

In the scope of RSA, let $n = pq$ be the product of two distinct large primes. How large the primes p and q need to be for a security level that matches the collision resistance of SHA-256?

A p and q each have 64 bits

B p and q each have 128 bits

C p and q each have 256 bits

D p and q each have 1536 bits

E p and q each have 3072 bits

RSA key length

In the scope of RSA, let $n = pq$ be the product of two distinct large primes. How large the primes p and q need to be for a security level that matches the collision resistance of SHA-256?

A. p and q each have 64 bits

B. p and q each have 128 bits

C. p and q each have 256 bits

D. p and q each have 1536 bits

E. p and q each have 3072 bits

The correct answer is **D**.

RSA signature generation

To sign a message m, I compute s and send (m, s). How should I compute s?

A $s = m^d \bmod n$

B $s = m^e \bmod n$

C $s = \text{hash}(m)^d \bmod n$

D $s = \text{hash}(m)^e \bmod n$

RSA signature generation

To sign a message m, I compute s and send (m, s). How should I compute s?

- A $s = m^d \bmod n$
- B $s = m^e \bmod n$
- C $s = \text{hash}(m)^d \bmod n$
- D $s = \text{hash}(m)^e \bmod n$

The correct answer is **C**.

RSA signature verification

To sign a message m, I compute s and send (m, s). How should someone verify my signature?

A. Accept m only if $\text{hash}(s) = m^d \bmod n$

B. Accept m only if $\text{hash}(s) = m^e \bmod n$

C. Accept m only if $\text{hash}(m) = s^d \bmod n$

D. Accept m only if $\text{hash}(m) = s^e \bmod n$

E. Accept m only if $\text{hash}^{-1}(s) = m^d \bmod n$

F. Accept m only if $\text{hash}^{-1}(s) = m^e \bmod n$

RSA signature verification

To sign a message m, I compute s and send (m, s). How should someone verify my signature?

- **A** Accept m only if $\text{hash}(s) = m^d \bmod n$
- **B** Accept m only if $\text{hash}(s) = m^e \bmod n$
- **C** Accept m only if $\text{hash}(m) = s^d \bmod n$
- **D** Accept m only if $\text{hash}(m) = s^e \bmod n$
- **E** Accept m only if $\text{hash}^{-1}(s) = m^d \bmod n$
- **F** Accept m only if $\text{hash}^{-1}(s) = m^e \bmod n$

The correct answer is **D**.

Discrete logarithm vs Diffie-Hellman problems

What is the relationship between the discrete logarithm (DL) and the Diffie-Hellman (DH) problems?

A. An adversary who can solve the DL problem can also solve the DH problem, but not necessarily vice-versa.

B. An adversary who can solve the DH problem can also solve the DL problem, but not necessarily vice-versa.

C. Both problems are equivalent.

Discrete logarithm vs Diffie-Hellman problems

What is the relationship between the discrete logarithm (DL) and the Diffie-Hellman (DH) problems?

A. An adversary who can solve the DL problem can also solve the DH problem, but not necessarily vice-versa.

B. An adversary who can solve the DH problem can also solve the DL problem, but not necessarily vice-versa.

C. Both problems are equivalent.

The correct answer is **A**.

Generic security of the discrete logarithm problem

Without knowing more about the group structure, which group can potentially yield a secure discrete logarithm problem? A group whose size is …

- **A** a prime number of the order of 2^{128}
- **B** exactly 2^{128}
- **C** a prime number of the order of 2^{256}
- **D** exactly 2^{256}
- **E** the product of two secret primes, each of the order of 2^{128}

Generic security of the discrete logarithm problem

Without knowing more about the group structure, which group can potentially yield a secure discrete logarithm problem? A group whose size is …

- A a prime number of the order of 2^{128}
- B exactly 2^{128}
- C a prime number of the order of 2^{256}
- D exactly 2^{256}
- E the product of two secret primes, each of the order of 2^{128}

The correct answer is **C**. Generically, the largest prime factor of the group size needs to be of the order of 2^{2s} for a security strength of s bits.