NTNU – Trondheim
Norwegian University of
Science and Technology

Department of Information Security and Communication Technology

Examination paper for TTM4135 Applied Cryptography and Network Security

Academic contact during examination: Anamaria Costache

Academic contact present at the exam location: No

Phone: 73 55 94 42

Examination date: 2025-05-30

Examination time (from-to): 09:00 - 12:00

Permitted examination support material: (D) No printed or hand-written support material is allowed. A specific basic calculator is allowed.

Other information: –

Language: English

Number of pages: 10

Number of pages enclosed: 2

Checked by:

Date Signature

Instructions

The maximum score is 60 points. The problem set consists of two exercises.

- Exercise 1 consists of the multiple choice questions. There are 30 questions each worth 1 point.

  Answer the multiple choice problems using the separate answer page. *Detach the answer page and hand it in at the end of the examination with your answers booklet(s).* The answer page includes answer boxes for multiple choice problems.

  Check the boxes like this: ☒

  If you check the wrong box, fill it completely, like this: ■. Then check the correct box.

  Other correction methods are not permitted.

  Incorrect answers receive a discount (penalty) of 0.33 marks. In the case that *multiple* answers are correct, *all* correct answers must be checked in order to receive the points for the question. If an imcomplete answer is given, it will count as a wrong answer.

  Note that the multiple choice problems do not receive penalty marks if you do not check any of the four boxes for a given statement.

- Exercise 2 consists of questions requiring written answers. There are 5 questions, each worth a maximum of 6 points. Partial points for each of the questions are divided evenly, so that if there are two parts for a question then each part is worth 3 points and if there are three parts for a questions then each part is worth 2 points.

  The written answers should be written in the answer book(s) provided.

1. Which of the below represents an active threat:
   (a) Replay
   (b) Denial of Service
   (c) Traffic analysis
   (d) Eavesdropping
2. $2^{-1} \bmod 377$ is equal to:
   (a) 1
   (b) 189
   (c) 376
   (d) 0.5
3. In $\mathbb{Z}_{13}^*$, the multiplicative group of non-zero integers modulo 11:
   (a) 2 is a generator
   (b) 3 is a generator
   (c) 4 is a generator
   (d) there is no generator
4. In an alphabet of 27 characters, how many possible permutations are there for the random simple substitution cipher?
   (a) 10^{27}
   (b) $27!$
   (c) 2^{27}
   (d) $2 \cdot 10^{27}$
5. According to Kerkhoff's principle, which of the following should be available to an attacker of an iterated block cipher?
   (a) The round keys
   (b) The number of rounds
   (c) The key length
   (d) The IV
6. Triple DES is the encryption algorithm defined by iterating three instances of the DES algorithm, by evaluating EDE (Encryption, Decryption, Encryption). Let K_1, K_2, K_3 be the three keys used, so that we evaluate triple DES as $C = E(D(E(P, K_1), K_2), K_3)$, for some plaintext P. Which of the following is *not* an option for values of the keys?
   (a) $K_1 = K_3$
   (b) $K_1 = K_2 = K_3$
   (c) $K_1 = K_2$
   (d) K_1, K_2, K_3 are all distinct

7. Recall that for AES, the S-boxes are defined over the finite field $GF(2^8)$. Let's take the easier example of $GF(2^2)$. This is defined via the polynomial $p(x) = x^2 + x + 1$. We recall the multiplication table below. What is the missing value in the table (denoted by $\star$)?

$\times$	0	1	x	$x+1$
0	0	0	0	0
1	0	1	x	$x+1$
x	0	x	$x+1$	$\star$
$x+1$	0	$x+1$	$\star$	x

   (a) 1
   (b) x
   (c) $x+1$
   (d) 0 are all distinct

8. Suppose that 110010 is observed to be a ciphertext from the one-time pad using the binary alphabet. Then we know that:

   (a) The plaintext was 111111 with probability $1/2^6$
   (b) The plaintext must be different from 110010
   (c) The keystream used must be different from 110010
   (d) The plaintext could have been any 6-bit string

9. Recall Euler's phi function ϕ. Let n be an integer, which factorises as $n = p_0^{k_0} \cdot p_1^{k_1} \cdot \ldots \cdot p_{\ell-1}^{k_{\ell-1}}$. What is the formula for computing $\phi(n)$?

   (a) $\phi(n) = \prod_{i=0}^{\ell-1} p_i^{k_i-1}(p_i - 1)$
   (b) $\phi(n) = \prod_{i=0}^{\ell-1} p_i^{k_i-1}$
   (c) $\phi(n) = \prod_{i=0}^{\ell-1}(p_i - 1)$
   (d) $\phi(n) = \prod_{i=0}^{\ell-2} p_i^{k_i-1}(p_i - 1)$

10. Recall Euler's phi function ϕ. If $n = 92$, what is the value of $\phi(n)$?

    (a) 88
    (b) 45
    (c) 91
    (d) 44

11. The Euler function ϕ is often useful for public key cryptography. For any integer $n > 1$ it is always true that:

    (a) $\phi(n)$ is an odd integer
    (b) $\phi(n)$ divides n
    (c) $\phi(n)$ is an even integer
    (d) $\phi(n) < n$

12. Which of the following pairs of equations cannot be solved using the Chinese Remainder Theorem?

(a) $x \equiv 1 \pmod{39}$ and $x \equiv 3 \pmod{28}$
(b) $x \equiv 3 \pmod{28}$ and $x \equiv 5 \pmod{4}$
(c) $x \equiv 5 \pmod{39}$ and $x \equiv 3 \pmod{4}$
(d) $x \equiv 3 \pmod{25}$ and $x \equiv 5 \pmod{4}$

13. Let p be a prime, and g a generator for $\mathbb{Z}_p^*$. The *Discrete Logarithm problem is:*
    (a) Given x and $y \equiv x \pmod{p}$, recover p
    (b) Given x, p and $y \equiv g^x \pmod{p}$, recover g
    (c) Given x and $y \equiv x^g \pmod{p}$, recover p
    (d) Given g, p and $y \equiv g^x \pmod{p}$, recover x

14. Euler's Theorem states that if n is an integer and $\gcd(a, n) = 1$, then:
    (a) $a^{\phi(n)} \equiv 1 \pmod{n}$
    (b) $a^n \equiv 1 \pmod{n}$
    (c) $n^a \equiv a \pmod{n}$
    (d) $a^{\phi(n)+1} \equiv 0 \pmod{n}$

15. When public key cryptography is used for encryption:
    (a) The private key of the sender is needed during encryption
    (b) The private key of the recipient is needed during encryption
    (c) The private key of the recipient is needed during decryption
    (d) The private key of the sender is needed during decryption

16. When public key cryptography is used for digital signatures
    (a) The public key of the verifier is used for signature verification
    (b) The private key of the signer is used for signature verification
    (c) The private key of the verifier is used for signature verification
    (d) The public key of the signer is used for signature verification

17. A function f is said to be *one-way* if:
    (a) It is computationally hard to compute $f(x)$ for all x
    (b) It is easy to compute $f(x)$ for all x
    (c) It is easy to compute $f(x)$ given x, but computationally hard to compute a pre-image of y, given y
    (d) It is computationally hard to compute $f(x)$ given x, but easy to compute a pre-image of y, given y

18. Below, you can find an incomplete Diffie-Hellman key exchange. What are missing are: the value B sent by Bob to Alice (denoted by $\star$), and the value of the shared secret key K (denoted by $\diamond$). What is the value of the pair $(\star, \diamond)$?

$$\begin{array}{lcl} \text{Alice} & \xrightarrow{A = g^a \mod p} & \text{Bob} \\ \text{Secret: } a & & \text{Secret: } b \\ \text{Computes: } A = g^a \mod p & \xleftarrow[B = \star \mod p]{} & \text{Computes: } B = \star \mod p \end{array}$$

Shared Secret Key: $K = \diamond \mod p$

(a) $(\star, \diamond) = (g^{ab}, g^{ab})$
(b) $(\star, \diamond) = (g^{b}, g^{ab})$
(c) $(\star, \diamond) = (g^{a}, g^{ab})$
(d) $(\star, \diamond) = (b, g^{ab})$

19. Below, you can see the table of modular exponentiation modulo 11. What is the value of the missing entry (denoted by $\star$)?

Table 1: $a^k \mod 11$

$k\backslash a$	1	2	3	4	5	6	7	8	9	10
1	1	2	3	4	5	6	7	8	9	10
2	1	4	9	5	3	3	5	9	4	1
3	1	8	5	9	4	7	2	6	3	10
4	1	5	4	3	9	9	3	4	5	1
5	1	10	1	1	1	10	10	10	1	10
6	1	9	3	4	5	5	4	3	9	1
7	1	7	9	5	3	8	8	2	4	10
8	1	$\star$	5	9	4	4	6	5	3	1
9	1	6	4	3	9	2	9	7	5	10
10	1	1	1	1	1	1	1	1	1	1

(a) 5
(b) 3
(c) 4
(d) 10

20. The inputs to a public key verification algorithm are:
   (a) The secret verification key, the message m and the signature σ
   (b) The public verification key and the hash of the signature σ and message m
   (c) The public verification key, the message m and the signature σ
   (d) The secret verification key and the hash of the signature σ and message m

21. In a public key signature scheme, the *unforgeability* property refers to the fact that is is impossible to:
   (a) Construct a pair (m, σ) that passes verification without the secret signing key K_s

(b) Construct a pair (m, σ) that passes verification without the public verification key K_s
(c) Construct a pair (m, σ) that passes verification without the secret verification key K_s
(d) Construct a pair (m, σ) that passes verification without the public signing key K_s

22. A message authentication code (MAC) requires a key as input while a hash function does not. Because of this:

    (a) The output size of a MAC must be longer than the output size of a hash function for the same security level
    (b) The length of the output of a MAC must always be longer than the length of the output of a hash
    (c) A MAC can provide data integrity
    (d) a MAC can provide non-repudiation

23. A cryptographic hash function h often needs the property of collision resistance. To ensure that finding collisions is of the same order of difficulty as brute force key search on a symmetric cipher with a key of 256 bits, the output size of h should be approximately:

    (a) 512 bits
    (b) $\sqrt{256}$ bits
    (c) 128 bits
    (d) 256 bits

24. The Merkle-Damgård construction for hash functions makes use of a compression function h, which acts on successive message blocks. A benefit of this construction is:

    (a) Computation of a hash value requires a fixed number of calls to h, independent of the length of the input message
    (b) If h is collision-resistant then the whole hash function is collision-resistant
    (c) No padding is required for the input message, no matter what the output size of h is
    (d) The length of the input message does not need to be included

25. A plaintext $P = 101101$ is encrypted into a ciphertext $C = 010011$ with a one-time pad. Suppose that an attacker is able to mount a known plaintext attack to partially obtain the three *lowest* bits of P, $P' = 101$ (i.e. $P = 101 \parallel P'$), and the corresponding portion of C. Now, the attacker knows:

    (a) That $P = 101101$
    (b) That $K = 111110$
    (c) That $K = K' \parallel 110$, for some K'
    (d) Nothing at all

26. Galois counter mode (GCM) provides which of the following security services?

    (a) Integrity, but not confidentiality
    (b) Both confidentiality and integrity
    (c) Non-repudiation, but not confidentiality
    (d) Both confidentiality and non-repudiation

27. The purpose of the handshake protocol in TLS is to:

(a) Change the cryptographic algorithms from previously used ones
(b) Signal events such as failures
(c) Setup sessions with the correct keys and algorithms
(d) Provide confidentiality and integrity for application messages

28. An X.509 digital certificate is issued by a certification authority CA for a subject A. Which of the following *must* be included in the certificate:

(a) The subject's private key
(b) The subject's public key
(c) The certification authority's private key
(d) The certification authority's public key

29. A difference between TLS 1.3 and TLS 1.2 is:

(a) The TLS 1.3 handshake protocol always provides forward secrecy
(b) There are no known attacks on the TLS 1.3 protocol
(c) The TLS 1.3 record protocol includes data compression
(d) The TLS 1.3 protocol provides post-quantum security

30. One valid TLS 1.2 ciphersuite is denoted as

TLS_RSA_WITH_AES_128_CBC_SHA256.

When this ciphersuite is chosen, integrity of application data is provided by:

(a) An HMAC tag with SHA-256 as the underlying hash function
(b) A CBC-based tag with AES as the underlying encryption function
(c) Signing each application message with an RSA signature
(d) Appending a hash of each packet, using SHA-256, before encryption with AES

Written answer questions

1. An *affine cipher* has an encryption algorithm as follows: for some secret values a, b, an alphabet Ω of size $\mid \Omega \mid = \omega$, and a message $m \in \Omega$,

$$E(x) = a \cdot x + b \pmod{\omega}.$$

We impose the requirement $\gcd(a, \omega) = 1$ (i.e. a and ω are co-prime), as decryption may not be defined otherwise.

   (a) How many keys can we have (give a general formula)?
   (b) Let $\omega = 28$. How many keys can we have?
   (c) Let $x = 6, a = 9, b = 21$. What is the value of $E(x)$?

2. Recall the Output FeedBack Mode (OFB) for encryption.

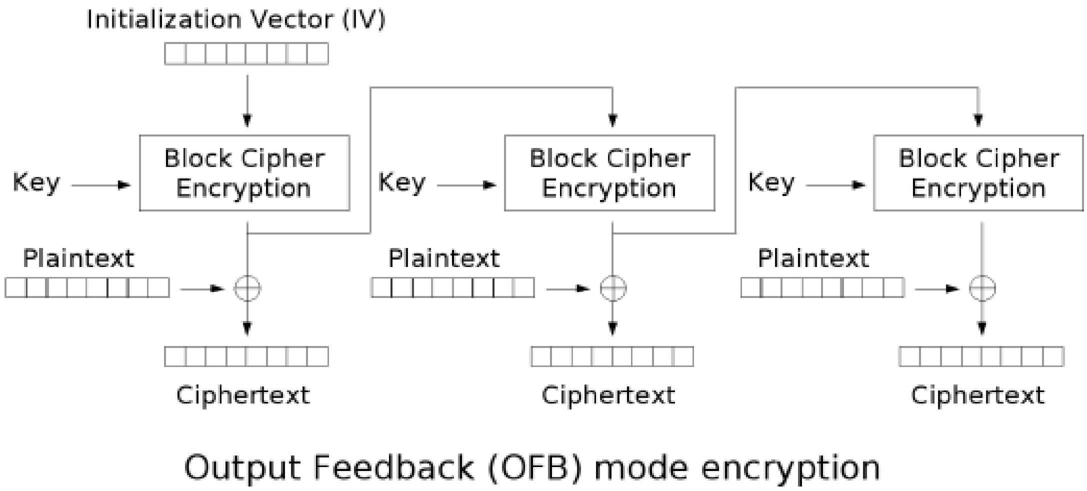

Figure 1: OFB mode of encryption

The keystream is

$$O_t = E(O_{t-1}, K),$$

where $O_0 = IV$ is chosen at random, and K is the encryption key. Then, the encryption equation is

$$C_t = P_t \oplus O_t.$$

   (a) What is the equation for decryption of ciphertext block C_t?
   (b) Is parallel encryption possible?
   (c) Is parallel decryption possible?
   (d) If one bit is flipped in one ciphertext block, how many bits are affected in the plaintext after decryption?

3. Recall that in digital signatures, we have the following parameters/ algorithms:

   – A *private* signing key K_s;

- A *public* verification key K_v;
- A *signing algorithm* `Sign`, which takes in a message m and a signing key K_s, and outputs a signature σ
$$\texttt{Sign}(m, K_s) = \sigma;$$
- A *verification algorithm* `Ver`, which takes in a message m, a verification key K_v, and a signature σ, and outputs a boolean value `True/ False`
$$\texttt{Ver}(m, \sigma, K_v) = \texttt{True/ False}.$$

(a) *Using the terminology above*, recall the *correctness* and *unforgeability* properties of a signature scheme.

(b) Consider this version of the RSA signature.
   - A modulus $n = pq$ is computed from random large primes p and q
   - Two exponents e and d are generated with
   $$ed \pmod{\phi(n)} = 1$$
   - Private signing key is $K_s = (d, p, q)$
   - Public verification key is $K_v = (e, n)$

   RSA Digital Signature Algorithm

   Signature generation: Inputs are the message m, the modulus n and the private exponent d

   i. Compute signature $\sigma = m^d \pmod{n}$
   ii. Output the pair (m, σ)

   Signature verification: Inputs are the message m, the claimed signature σ and the verification key $K_v = (e, n)$

   i. Check whether $\sigma^e \pmod{n} = m$, and if so, output `True`; otherwise, output `False`

   Why is it insecure? (Hint: consider question 3a)).

(c) How can we fix this? (Hint: recall hash functions)

4. Cryptosystems based on discrete logarithms often make use of a prime number and a generator g of the integers modulo p, $\mathbb{Z}_p^*$ (recall that in class, we often wrote $\langle g \rangle = \mathbb{Z}_p^*$, i.e. *$g$ generates* the group $\mathbb{Z}_p^*$).

   (a) Show that when $p = 19$, the value $g' = 5$ is not a generator but the value $g = 2$ is a generator. (Hint: for the following question, you may find it helpful to list all the powers of $g = 2$ modulo 19).

   (b) Consider Diffie–Hellman key exchange in $\mathbb{Z}_p^*$ when $p = 19$ and $g = 2$. Alice chooses random secret input value $a = 7$, and so sends the value $A = 14 = 2^7 \pmod{19}$ to Bob. Bob sends his message B back, and they both compute the shared secret. If the value of the shared secret is 17, *what is Bob's secret b*? (Hint: you can go back to Question 18 to see a partial summary of the Diffie-Hellman key exchange).

5. The TLS handshake protocol is complex and allows several variants. In this question we focus on the TLS 1.3 version of the handshake protocol.

(a) It is optional for a client to use a certificate. What are the security consequences when a client does not have a certificate?

(b) TLS 1.3 offers several different ciphersuites which are negotiated in the handshake. State and explain one advantage and one disadvantage of allowing different ciphersuites.

(c) How does 0-RTT enhance the protocol compared to TLS1.2? Are there any security trade-offs?

TTM4135 Examination 2025-05-30
Answer page for Exercise 1 Multiple Choice Questions

Detach this page and hand it in together with your written answers

Candidate number: ☐☐☐☐☐

1.	(a) ☐	(b) ☐	(c) ☐	(d) ☐
2.	(a) ☐	(b) ☐	(c) ☐	(d) ☐
3.	(a) ☐	(b) ☐	(c) ☐	(d) ☐
4.	(a) ☐	(b) ☐	(c) ☐	(d) ☐
5.	(a) ☐	(b) ☐	(c) ☐	(d) ☐
6.	(a) ☐	(b) ☐	(c) ☐	(d) ☐
7.	(a) ☐	(b) ☐	(c) ☐	(d) ☐
8.	(a) ☐	(b) ☐	(c) ☐	(d) ☐
9.	(a) ☐	(b) ☐	(c) ☐	(d) ☐
10.	(a) ☐	(b) ☐	(c) ☐	(d) ☐
11.	(a) ☐	(b) ☐	(c) ☐	(d) ☐
12.	(a) ☐	(b) ☐	(c) ☐	(d) ☐
13.	(a) ☐	(b) ☐	(c) ☐	(d) ☐
14.	(a) ☐	(b) ☐	(c) ☐	(d) ☐
15.	(a) ☐	(b) ☐	(c) ☐	(d) ☐
16.	(a) ☐	(b) ☐	(c) ☐	(d) ☐
17.	(a) ☐	(b) ☐	(c) ☐	(d) ☐
18.	(a) ☐	(b) ☐	(c) ☐	(d) ☐
19.	(a) ☐	(b) ☐	(c) ☐	(d) ☐
20.	(a) ☐	(b) ☐	(c) ☐	(d) ☐
21.	(a) ☐	(b) ☐	(c) ☐	(d) ☐
22.	(a) ☐	(b) ☐	(c) ☐	(d) ☐

23.	(a) ☐	(b) ☐	(c) ☐	(d) ☐
24.	(a) ☐	(b) ☐	(c) ☐	(d) ☐
25.	(a) ☐	(b) ☐	(c) ☐	(d) ☐
26.	(a) ☐	(b) ☐	(c) ☐	(d) ☐
27.	(a) ☐	(b) ☐	(c) ☐	(d) ☐
28.	(a) ☐	(b) ☐	(c) ☐	(d) ☐
29.	(a) ☐	(b) ☐	(c) ☐	(d) ☐
30.	(a) ☐	(b) ☐	(c) ☐	(d) ☐