SECURITY 1 (BSc)

Examination set, May 22, 2019.

What to hand in

You submit electronically, via LearnIT:

1. A plain-text file `answers.txt` with your answers to multiple choice questions, one line per answer: the question identifier, a colon, and the answer digit (format example below). You can submit only one answer for each question; multiple answers will be awarded zero points.
2. A pdf file `discussion.pdf` containing the answers to discussion questions D1 and D2. It may contain text and figures in any form.

If there are technical or logistical problems with submitting in this way, please turn to available personnel in order to find a backup solution. Feel free to submit a .zip archive containing the above two files.

How to answer

Example multiple choice question:

> **Question 9-7**
>
> This is the question text. It is followed by the possible answers:
>
> 1. The first possible answer.
> 2. The second possible answer.
> 3. The third possible answer.
> 4. The fourth possible answer.

Example answer. You choose "The third possible answer":

```
9-7:3
```

Permissible aids

This examination is A11: Written, on-premise, with access to all aids and internet.

Questions begin on the next page.

Multiple-choice Questions

Your answers to the multiple choice questions counts 70% towards the final grade.

Principles

Question 1-1

The Foxfire firewall is malware-free because it is open source. It comes with default settings that fit any customer requirements, so it does not need any customer-specific configuration. Packet inspection is delegated to an advanced AI engine FoxAI (r) that securely runs in Foxfire protected servers. In fact, for security reasons, Foxfire does not share the details of FoxAI(r) to anyone. In case of a DDOS attack, FoxAI (r) automatically and remotely sets Foxfire to blacklist the attacking IPs without requiring any intervention by the customer. Which security principle is violated by FoxFire?

1. Least privilege
2. Open Design
3. Minimum Exposure
4. Psychological Acceptability

Question 1-2

A train operator offers to its customers the possibility to buy digital travel ticket through an app. Once the customer purchases the ticket, the details of the tickets are securely stored on the train operator's server. To verify a ticket, the ticket inspector is given a device that scans the ticket and send the details to the server. The server verifies that the received details match the one stored and replies back to device whether the ticket is valid or not. Which of the following security principles is violated?

1. Psychological Acceptability
2. No single point of failure
3. Complete Mediation
4. Least Privilege

Question 1-3

The vulnerability CVE-2005-0750 exploited a bug in the Bluetooth stack on Linux kernels from 2.4.6 to 2.6.11. The attack was also possible on embedded devices without any support of Bluetooth devices at all. Which of the following security principles was violated on those embedded devices?

1. Economy of Mechanism
2. Open Design
3. Minimum Exposure
4. Least Privilege

Network Security

Question 2-1

A 4-way switch has the following internal switching table:

```
LINK MACs
1    d4:25:8b:45:20:36 40:98:ad:63:f0:f5
2    38:ca:da:a3:06:c0
3    40:98:ad:63:f0:f5 f8:95:ea:39:8a:85
4    ad:00:fe:01:01:00
```

What's unusual about this switching table?

1. At links 1 and 3 more than one MAC is located
2. At links 2 and 4 only one MAC is located
3. A MAC is located at multiple links
4. Nothing

Question 2-2

An adversary with MAC `38:ca:da:a3:06:c0/IP 169.254.191.148` who wishes to eavesdrop on MAC `f8:95:ea:39:8a:85/IP 169.254.191.221` using an ARP Spoofing attack would broadcast ARP packets:

1. `169.254.191.148 IS-AT 38:ca:da:a3:06:c0`
2. `169.254.191.148 IS-AT f8:95:ea:39:8a:85`
3. `169.254.191.221 IS-AT 38:ca:da:a3:06:c0`
4. `169.254.191.221 IS-AT f8:95:ea:39:8a:85`

Question 2-3

Which of the following is NOT a capability of the adversary in the Dolev-Yao model?

1. The adversary can drop messages
2. The adversary can guess secrets
3. The adversary can decrypt messages (if he knows the key)
4. The adversary can replay message

Question 2-4

Which of the following attacks works only against HTTP servers?

1. SYN flood
2. Slow Loris
3. SYN spoofing
4. Ping flood

Question 2-5

An amplification attack is

1. A binary exploitation technique underpinning SPECTRE and MELTDOWN
2. A DOS attack in which each spoofed request generates more than one reply
3. A DOS attack where an attacker spoofs requests from the target
4. An attack on USB devices inducing a power surge

Question 2-6

Which of the following rules cannot be expressed by a stateless firewall?

1. Drop all incoming packets to port 80
2. Drop all outgoing packets to port 80
3. Drop all established TCP connections to port 80
4. Drop all UDP traffic to port 80

Question 2-7

This is the output of my portscanning the ITU web server:

```
> sudo nmap www.itu.dk -p 1-65535 -A4
```

```
Starting Nmap 7.70 ( https://nmap.org ) at 2019-05-08 14:58 CEST
Nmap scan report for www.itu.dk (130.226.142.6)
Host is up (0.0037s latency).
rDNS record for 130.226.142.6: asterix.itu.dk
Not shown: 65532 filtered ports
PORT     STATE SERVICE
80/tcp   open  http
443/tcp  open  https
1720/tcp open  h323q931

Nmap done: 1 IP address (1 host up) scanned in 4.54 seconds
```

From this, we learn that port 21 is

1. Open
2. Closed
3. Filtered
4. Unfiltered

Applied cryptography

Question 3-1

I download a Kali-linux iso-image, expecting the downloaded file to have SHA224 hash

```
101c20df6b4631881a747c8a292293404d67e21aabb0d0090df1e780
```

Computing the SHA256 hash of the iso-image, I get

```
bc0ef283f57fd5e4f36657053228eae8d2d5b0e4d87c6ee069a9cade39411d63
```

From this we learn that

1. The downloaded file has been corrupted in transit
2. The downloaded file has been decrypted in transit
3. The downloaded file contains a forged signature
4. Nothing

Question 3-2

Which of the following are easy to compute in practice?

1. The SHA224 hash of the text string
   `24736270c237a0678e90b61f968f3babb9ccd7c6c93897e33ad252ea`
2. A (distinct) text string which has the SHA224 hash value
   `24736270c237a0678e90b61f968f3babb9ccd7c6c93897e33ad252ea`
3. A (distinct) text string which has the same SHA224 hash value as the string
   `24736270c237a0678e90b61f968f3babb9ccd7c6c93897e33ad252ea`
4. Two distinct text strings which have the same SHA224 hash value

Question 3-3

Passwords are stored as hashes to preserve:

1. Password confidentiality
2. Password integrity
3. Password availability
4. Password effectiveness

Question 3-4

Passwords stored as hashes uses a salt to:

1. Make brute-force decryption impractical
2. Make reverse hashing impractical
3. Make pre-computation attacks impractical
4. Make password lookup more efficient

Question 3-5

A symmetric encryption algorithm which is theoretically and practically unbreakable exists, however, it requires

1. Short plaintexts
2. Short keys
3. Plaintexts as long as ciphertexts
4. Keys as long as the plaintext

Question 3-6

Cipher-block chaining provides

1. Perfect secrecy
2. Imperfect secrecy
3. A mechanism to perform arbitrary-length encryption from fixed-length keys
4. A mechanism to perform fixed-length encryption from arbitrary-length keys

Question 3-7

I'm making my own MAC scheme. My process is this: Given a message M, compute the SHA256 hash `h(M)`. Send both M and `h(M)` on the network. On receipt of a message M' and hash H, compute h(M') and compare to H; if they are the same, the received message M' is the same as the sent message M. Is there a problem with this scheme? (Below, we write "`M || N`" for the concatenation of the strings M and N.)

1. Yes, I need to add a publically-known nonce N to the hash (so we compute `h(M || N)` instead of h(M)) to prevent replay attacks
2. Yes, an adversary may replace M with N, then compute h(N) and add that to the message
3. Yes, SHA256 is broken, duplicates are easy to find
4. No

Question 3-8

I have invented an asymmetric encryption scheme. A keypair is a pair of random values (x,y); x is the public key, y is the private key. To encrypt a message M, compute the hash `h(M||x)`; to decrypt a ciphertext N, compute the hash `h(y||N)`. Which of the following statements proves that my scheme is broken?

1. `h(h(M || x) || y) == M`
2. `h(y || (h(M || x)) != M`
3. `h(x || (h(M || y)) != N`
4. `h(x || (h(M || y)) == N`

Question 3-9

Asymmetric encryption schemes are generally preferred over symmetric schemes when:

1. Performance is a key concern

2. Key-distribution is a key concern
3. Confidentiality is a key concern
4. Availability is a key concern

Question 3-10

Is Diffie-Hellman susceptible to man-in-the-middle (MITM) attacks?

1. No, Diffie-Hellman does not have anything to do with networks
2. No, Diffie-Hellman defeats MITM /provided/ the discrete log is hard to compute
3. This is currently unknown but unsuspected
4. Yes

Internet Security Protocol

Question 4-1

The Needham-Schröder public-key protocol is secure under which assumption:

1. Discrete Logarithm problem
2. Collusion resistant hashing
3. Factorization problem
4. It is not secure

Question 4-2

Which of the following statement about keying in Kerberos is correct:

1. All the 5 keys are symmetric keys
2. The keys shared between AS and TGS (Ktgs) and between TGS and the service B (Kb) are symmetric keys, the rest are asymmetric keys
3. The keys created by AS (Kauth) and TGS (Kserv) are symmetric keys, the rest are asymmetric keys
4. All the 5 keys are asymmetric keys

Question 4-3

How many entities should sign an X.509v3 certificate

1. None
2. One, the CA
3. Two, the CA and the RootCA
4. Many, depending on the number of intermediate CAs

Question 4-4

Which of the following fields are confidential in a certificate

1. The serial number
2. The issuer unique identifier
3. All fields are confidential
4. None of the above

Question 4-5

Which of these certificate revocation mechanisms is more effective against compromised CAs

1. CRL
2. OCSP
3. OCSP stapling
4. CRLsets

Question 4-6

Certificate Transparency is NOT aimed at protecting against

1. A dishonest client
2. A dishonest log server
3. A dishonest CA
4. A colluding log server and CA

Question 4-7

Why does TLS use MAC rather than digital signatures?

1. Digital signatures do not ensure integrity
2. Digital signatures are less efficient than MACs
3. Digital signatures require key establishment
4. Digital signatures are less secure than MACs

Question 4-8

According to the TLS Record protocol, the correct sequence that applies when a secure message is RECEIVED is

1. Decompression, integrity check, decryption
2. Decryption, integrity check, Decompression
3. Integrity check, decryption, decompression
4. Decryption, decompression, integrity check

Question 4-9

The TLS pre master secret key is the result of

1. Hashing all the messages seen during the Handshake protocol
2. Hashing the two nonces generated in the Client_hello and Server_hello messages
3. The specific ciphers agreed in the Client_hello and Server_hello messages
4. Encrypting the public key of the Server

Question 4-10

Which of the following statements is FALSE

1. TLS provides protection against a compromised CA
2. TLS provides integrity
3. TLS provides a way to establish a session key
4. Some messages of the TLS handshake protocol may be skipped

Penetration testing I

Question 5-1

An idle scan has the property that

1. No IP packet is ever sent from the attacker to the target

2. No IP packet containing the IP of the attacker is ever sent to the target
3. No IP packet is ever delivered the zombie
4. No IP packet containing the IP of the zombie is ever sent to the target

Question 5-2

I'm trying to construct a buffer overflow for a C program, and I've used a debugger to determine that just prior to reading unlimited input into a bounded buffer, my stack looks like this:

```
0xbffad5d0:     0x00000000      0x0804865b      0x0000000a      0x00000000
0xbffad5e0:     0x42424242      0x00000000      0x0804862b      0x00000000
0xbffad5f0:     0x00000000      0x00000000      0x00000000      0x00000000
0xbffad600:     0x00000000      0x00000000      0x00000000      0x00000000
0xbffad610:     0x00000000      0x00000000      0x00000000      0x00000000
0xbffad620:     0x00293ff4      0x00000000      0xbffff748      0x0810556b
0xbffad630:     0xbffff640      0x00118fa6      0x0012bff4      0x00000000
0xbffad640:     0x00000000      0x000000ca      0x00000006      0xbffff68c
```

The debugger tells me that the return address is saved at address 0xbffad5e8. It also tells me that the buffer I'm trying to overflow starts at address 0xbffad5d4. I've determined myself that I'd rather have the program return to 0x080485d0. What input should I provide to the program in order to overflow the buffer?

1. 5*4=20 non-zero bytes followed by "\xd0\x85\x04\x08"
2. 5*4=20 non-zero bytes followed by "\x2b\x86\x04\x08"
3. 7*4=28 non-zero bytes followed by "\xd0\x85\x04\x08"
4. 7*4=28 non-zero bytes followed by "\x2b\x86\x04\x08"

Question 5-3

Below is an example program.

```
#include <stdio.h>
#include <string.h>
void pass_function()
{
    int length;
    char bufferA[8];                /* (i) */
    printf("Enter password: ");
    gets(bufferA);
    printf("Returning\n");
    return;                         /* (ii) */
}

int main(int argc, char *argv[])
{
    pass_function();
    return 0;                       /* (iii) */
}
```

Does the program contain a potential for a buffer overflow?

1. Yes, return address may be overwritten at program point (i)
2. Yes, return address may be overwritten at program point (ii)
3. Yes, return address may be overwritten at program point (iii)
4. No

Penetration Testing II

Question 6-1

The following script excerpt is used by a web-server when answering requests. It takes in a user input ($par) and uses it as parameter for a grep over a server file.

```
    system(“grep $par list.txt”);
```

To which kind of attack is this script excerpt vulnerable?

1. Dependency Injection
2. Command Injection
3. SQL Injection
4. XSS

Question 6-2

The following script takes in a user input (name) and prints it into the server webpage.

```
<?php $input = $_GET['input']; ?>
<html>
  <body>
  <p>Hello <?= $input ?></p>
  </body>
</html>
```

To which kind of attack is this script vulnerable?

1. Dependency Injection
2. Command Injection
3. SQL Injection
4. XSS

Question 6-3

The following script takes in a user input (e.g. `http://example.com/test.html#soren`) and prints it into the server webpage.

```
<html>
  <body>
    <script>
      var input = location.hash;
      document.write("hello " + input);
    </script>
  </body>
</html>
```

To which kind of attack is this script vulnerable?

1. Dependency Injection
2. Command Injection
3. SQL Injection
4. XSS

Question 6-4

Which of these attacks CANNOT be done remotely

1. Code Injection
2. Command Injection
3. Privilege Escalation
4. None of the above

Advanced Crypto

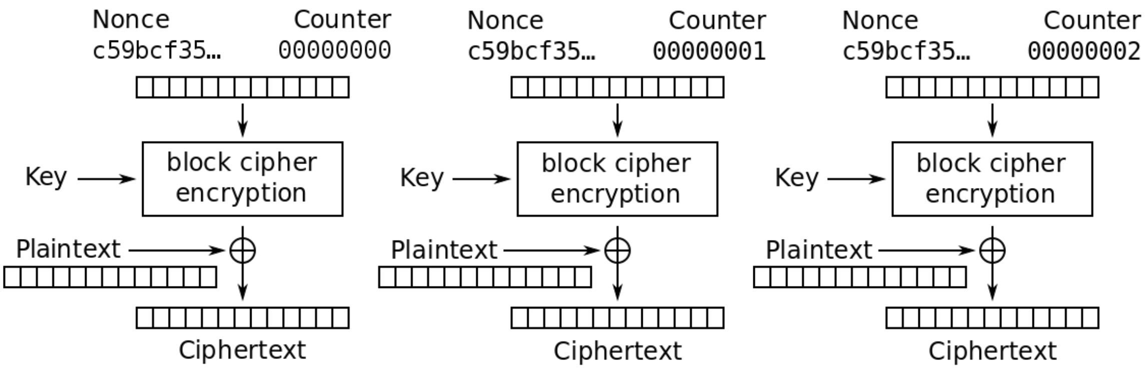
Question 7-1

Which of the following internal AES functions is NOT invertible

1. S-box
2. Shift Row
3. Mix Columns
4. None of the above

Question 7-2

AES can be used in CTR mode as in the figure below. CTR mode of operation requires a unique nonce for each plaintext. What would happen if the same nonce is reused to encrypt two different plaintexts P1 and P2 in to the respective ciphertexts C1 and C2?



Counter (CTR) mode encryption

1. C1 xor C2 = P1 xor P2
2. P2 = C2 xor P1
3. P2 = C1 xor C2
4. C2 = P1 xor C1

Question 7-3

To encrypt a message M in the ElGamal encryption system, Bob takes in a public key PK=g^x mod p, chooses a random y in $\mathbb{Z}^*_p$, and computes the ciphertext c=g^{xy} m. Bob then sends the pair (c, g^y mod p) to Alice, who can decrypt the ciphertext c as m=c/g^{yx} mod p. Imagine a different encryption algorithm in which Bob computes the ciphertext as c'=$g^{xy}h^m$, where h is a generator in $\mathbb{Z}^*_p$, and sends the triplet (c', g^y mod p, h) to Alice. (Note that h here has nothing to do with hashes.) How can Alice decrypt c' and get m?

1. The very same way as in ElGamal
2. By m=c'/g^{yx} h^x mod p
3. By m=c' hx/g^{yx} mod p
4. She cannot decrypt c'

Question 7-4

A proposed commitment scheme based on symmetric cryptography is defined as follows: for the commit phase, the sender generates a key k and sends to the receiver the commitment c=AES(M,k), i.e., the AES encryption of M under key k. For the reveal phase, the sender sends the key to the receiver so that the message can be decrypted as m=AES^{-1}(C,k), i.e., the AES decryption of ciphertext C under key k. Which goal(s) are met by this scheme?

1. Only hiding
2. Only binding

3. Both hiding and binding
4. Neither hiding nor binding

Discussion Questions

Question D1 (Applied Cryptography)

Your answer to this question counts 15% towards the final grade.

A major hardware vendor provides firmware updates as binary files on its website. To provide integrity guarantees, the SHA-224 hash of each binary file is also provided on the web-site; however, the web-site supports only the `http` protocol, not `https`.

1. Explain why this scheme is insufficient to provide an integrity guarantee
2. Propose a change so that the mechanism does in fact provide an integrity guarantee. You may assume a CA.

Answer this question in at most 500 words.

Question D2 (Access control)

Your answer to this question counts 15% towards the final grade.

A bank uses NemID as a service for authenticating its customers. NemID allows the bank's customer to login using either single-factor authentication or two-factor authentication. The customer provides username and password for single-factor authentication, while, for two-factor authentication, they provide username, password, and a onetime code from a small paper that comes with their NemID. An authenticated customer can make dispositive operations (e.g. payments, create loans, sign agreements) or documentary operations (e.g. print bank statements, overview of existing loans and agreements).

The bank wants to create an access control policy that allows

- a customer who authenticates via single-factor authentication to make only documentary operations;
- a customer who authenticates via two-factor authentication to make dispositive and documentary operations.

Answer the following questions:

1. Explain why Discretionary Access Control (DAC) is not the most appropriate model to enforce the bank's policy
2. Identify and motivate the adoption of a more appropriate access control model, and provide a high-level specification of the policy, which fits better the chosen access control model

Answer this question in at most 500 words.

(End of questions.)