

Note:

- During the attendance check a sticker containing a unique code will be put on this exam.
- This code contains a unique number that associates this exam with your registration number.
- This number is printed both next to the code and to the signature field in the attendance check list.

Computer Networking and IT Security

Exam: INHN0012 / Retake **Date:** Thursday 6th April, 2023
Examiner: Prof. Dr.-Ing. Stephan Günther **Time:** 14:00 – 16:00

	P 1	P 2	P 3	P 4	P 5	P 6
I						
II						

Before we proceed with reading the processing instructions, please answer the following questions. This information helps us to examine learning success depending on participation in individual lecture components. The information is **voluntary** and **not considered for evaluation**, i. e., answers to these questions do not give credits. In order to exclude any influence, this page will not be made accessible during the correction.

a) Did you attend the lecture?

- 1 (regularly)
 2 (sometimes)
 3 (never)

b) Did you attend the tutorials?

- 1 (regularly)
 2 (sometimes)
 3 (never)

Working instructions

- This exam consists of **16 pages** with a total of **6 problems** and the cheatsheet distributed with the exam. Please make sure now that you received a complete copy of the exam.
- The total amount of achievable credits in this exam is 90 credits.
- Detaching pages from the exam is prohibited.
- Allowed resources:
 - one **non-programmable pocket calculator**
 - one **analog dictionary** English ↔ native language
- Subproblems marked by * can be solved without results of previous subproblems.
- **Answers are only accepted if the solution approach is documented.** Give a reason for each answer unless explicitly stated otherwise in the respective subproblem.
- Do not write with red or green colors nor use pencils.
- Physically turn off all electronic devices, put them into your bag and close the bag.

Left room from _____ to _____ / Early submission at _____

Problem 1 Multiple Choice (14 credits)

The following subproblems are multiple choice / multiple answer, i. e., at least one answer per subproblem is correct. Sub problems with a single correct answer are graded with 1 credit if correct. Those with more than one correct answers are graded with 0.5 credit per correct answer and -0.5 credit per wrong answer. Missing crosses have no influence. The minimal amount of credits per subproblem is 0 credits.

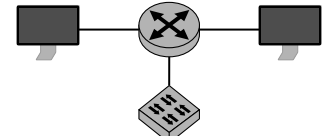
Mark correct answers with a cross

To undo a cross, completely fill out the answer option

To re-mark an option, use a human-readable marking

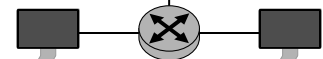
a)* How many broadcast domains does the network to the right contain?

- 3 6 1 5 2 4



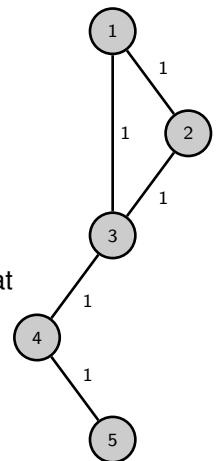
b)* How many collision domains does the network to the right contain?

- 4 2 3 1 6 5



c)* Mark the adjacency matrix for the network to the right.

- $\begin{bmatrix} 1 & -1 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 1 & -1 \end{bmatrix}$ $\begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix}$ $\begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}$

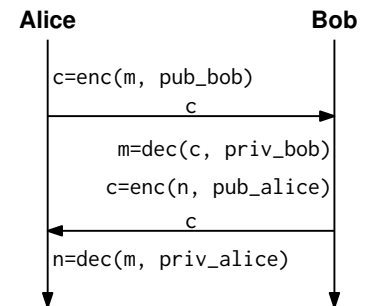


d)* Given the distance matrix D for the network to the right. What is the minimum n such that $D^n = D^{n+1}$ holds?

- $n = 1$ $n = 4$ $n = 3$ $n = 6$
 $n = 7$ $n = 0$ $n = 2$ $n = 5$

e)* Assume that Bob and Alice know each others public key. What is the scheme on the right vulnerable to?

- Eve can impersonate Bob Man-in-the-Middle and thereby full message decrypt
 No forward-secrecy Replay Attacks



f)* Which **three** of the following claims are true?

- In AES-CBC, later blocks influence previous ones Common key lengths for AES are 2048 bit and 4096 bit
 ECC is robust against quantum computers SHA-256 is vulnerable to length-extension attacks
 RSA is robust against quantum computers Cipher text blocks in AES-ECB can be cut and pasted unnoticeable
 AES is robust against quantum computers SHA-3 is vulnerable to length-extension attacks

g)* What is the AES-CTR scheme?

- A hash function A block cipher A stream cipher A key exchange

h)* A chain of trust is used in . . .

- Trusted fourth parties Trusted Memory
 Trusted computing Your trusted bike lock

i)* The domain name system . . .

- has a mapping from every single IP address to a domain name is inherently trustworthy
 translates domain names to IP addresses has a single, central authority


j)* The congestion avoidance phase of TCP Reno . . .


- increases the traffic control window linearly increases the traffic control window exponentially
 is the first phase active in a new connection follows the rapid start phase


k)* In 802.11 . . .


- management frames are unprotected when using WEP only two layer 2 addresses are contained in the header
 traffic cannot be sniffed by attacker when not in line of sight arbitrary errors are corrected using the FCS (Frame Correction Sum)


Problem 2 Short Questions: Security (14 credits)


0  a)* Differentiate Authentication from Authorization.


1 


2 


0  b) Argue whether an man-in-the-middle attacker can be passive and/or active. Describe a scenario for each applicable property.


1 


2 


0  c) How is a password hash function different from SHA-256?

1 

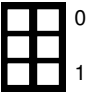
0  d)* Name and describe the three properties of a cryptographic hash function.

1 

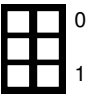
2 

3 

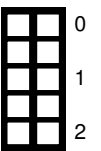
e) Briefly describe a scenario in which the IPsec tunnel mode is used.



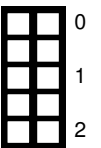
f) Differentiate a block cipher from a stream cipher.



g) Differentiate symmetric encryption from asymmetric encryption. Elaborate on the usage of keys.





h) Describe three functions of a TPM.





Problem 3 CRC (14 credits)


In this problem we consider the binary message 00100110 which should be protected by a CRC as we introduced it for Ethernet-based networks in the lecture. We assume the reduction polynomial $r(x) = x^2 + 1$.


0  a)* Briefly explain what CRC is used for in the context of Ethernet.


1 


0  b)* What is the reduction polynomial being used for?


1 


0  c)* What does it mean if the reduction polynomial is *irreducible*.


1 


0  d)* Reason whether or not CRC requires an irreducible reduction polynomial.


1 

2 

0  e)* Show whether or not $r(x)$ is irreducible.

1 

0  f)* Assuming Ethernet, what is the reaction of the receiving node when a bit error is detected.

1 

Before we continue to fill out the hexdump, we want to mark the end of different headers. Assume that

- the L3 header does not use any options,
- the L4 header uses 12 B options, and
- the total frame length (including checksum) is 111 B.

c)* Mark **the end** of the **L3** and **L4 headers** as well as of the **frame itself** in Figure 4.2. As an example, the end of the L2 header is already marked.

d) Fill in the frame check sequence given as 42 0a f1 73 in Figure 4.2.

We now start with filling in different fields of the L3 header. The start of the L3 header is already given in Figure 4.2. **Do not forget to mark to which subproblem your fill ins belong.**

e)* Fill in the field specifying type and length of the L3 header.

f)* Fill in the L3 source address.

g)* Fill in the L3 destination address.

h)* Fill in the value of the field specifying the type of the L3 SDU.

We now continue with filling in different fields of the L4 header. In case a value is not defined, make a reasonable assumption. **Do not forget to mark to which subproblem your fill ins belong.**

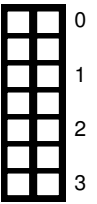
i) Fill in the source port.

j) Fill in the destination port.

k) Fill in the value of the field specifying that offset in the L4 header.

Finally, we come the application layer of the frame's content which is the ASCII encoded string "SSH-2.0-OpenSSH_9.2p1 Debian-2".

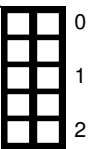
l) Fill in the first 5 B of the L7 PDU.



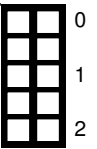
e)



f)



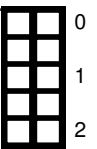
g)



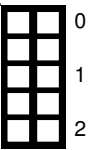
h)



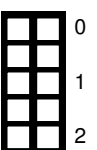
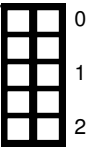
i)



j)



k)



Problem 5 DNS (13 credits)

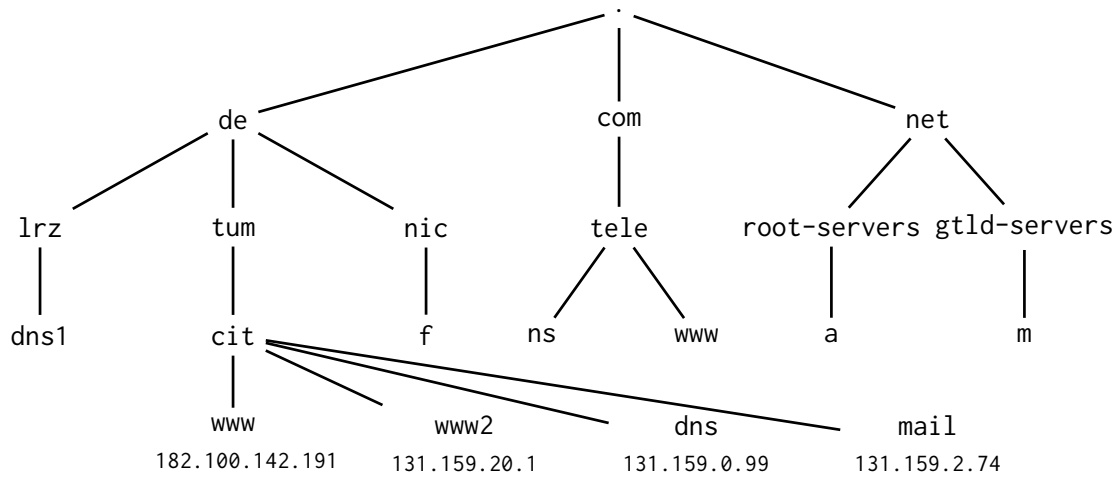


Figure 5.1: A part of the DNS.



a) Briefly describe the purpose of DNS.



b) Briefly describe the difference between a fully and non-fully qualified domain name, also regarding their notation.

Figure 5.1 shows the zone file of the authoritative name server for `cit.tum.de`.

```

1 $ORIGIN cit.tum.de.
2 $TTL 1H
3
4 @ IN SOA dns.cit.tum.de. hostmaster.cit.tum.de. (...)
5
6
7 _____ IN _____ 131.159.2.74
8 _____ IN _____ 131.159.20.1
9 _____ IN _____ 131.159.0.99
10 _____ IN _____ 182.100.142.191
11 _____ IN _____ 20 mail.cit.tum.de
12 _____ IN _____ dns.cit.tum.de

```

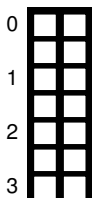
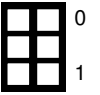


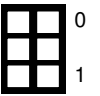
Figure 5.2: DNS zone file on nameserver `dns.cit.tum.de`

c)* Add all other missing data in the zone file depicted in Figure 5.2 based on the information from Figure 5.1.

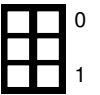
d)* Distinguish a resolver from a nameserver.



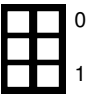
e)* Briefly describe the purpose of a zone's SOA record.



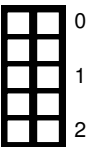
f)* What does "authoritative" mean in the context of DNS?



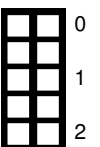
g)* Determine the PTR record of the address 11.42.43.12. You do not need to reason your answer.



h)* Describe the components of the url <https://www.cit.tum.de/webmail?user=user&pwd=pass>.



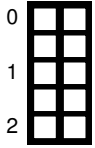
i) Explain the difference between recursive and iterative name resolution.



Problem 6 Side-Channel Information Stealing (16 credits)

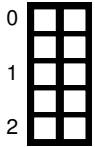
You have breached the data center of a large cloud hosting provider. You intend to extract their private key of 4096 bit length. The key is derived from perfect, uniform randomness. As a very strict network policy is employed there is no way you will be able to do this via the network.

Therefore, you have come up with another way: The cloud hosting provider streams their data center via a live cam (24 frames per second), seemingly to show off their hardware. This stream includes a view of the hard drive activity LED of the relevant server. You can control the LED — and decide to extract the private key by encoding it through LED blink patterns, which you can decode by viewing the live stream.



a)* What is the maximum data rate achievable on the channel?

To properly detect the LED being on or off, a **pulse length of at least** 100 ms is needed.

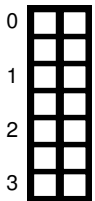


b) What is the resulting transmission rate? How long does it take to transmit the private key?

There is still one problem remaining: you have to properly synchronize the transmission on sender and receiver side. To achieve this you decide to employ a new *8b11b* coding scheme. In this scheme, 8 bit of payload are coded to 11 bit of channel word. The coding transforms a byte to a channel word by prepending a start sequence, thereby marking the start of each channel word recognizable. This start sequence is **three bits** long, and consists of all ones: 111.

To not confuse frame starts with actual data, all left-most occurrences of 11 in the data are replaced with 110. This process is called bit-stuffing. On the receiving side, this process is reversed, thus replacing 110 by 11.

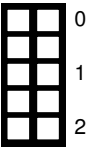
You can neglect any padding that would become necessary for all following sub tasks!



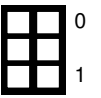
c)* Determine the expected length **increase** of the actual transmitted data. Note, that the bit-stuffing is done on the payload!

If you were unable to solve subproblem c), use 1024 bit as the expected length increase.

d) Using the expected length increase, determine the expected code rate.

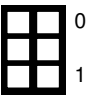


e)* Argue whether it is realistic to calculate the key by brute-force, rather than extracting it via the side-channel.

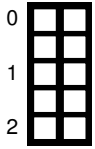


You decide that your current approach is too inefficient.

f) Propose an approach to reducing the overhead while maintaining the synchronization properties.

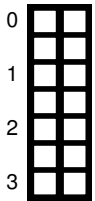


While testing your approach you realize that there is still a $\frac{1}{200}$ chance left that you read a given bit **incorrectly**. Given this, you decide to add redundancy to your coding, extending the *8b11b* coding to the **8b11b_v2** coding. Again, each 8 bit long word of data is translated to 11 bit of channel word. In contrast to before, the coding scheme employed, which shall not be explained in detail, allows for synchronization **without** the use of special symbols. Bit-stuffing is therefore not longer necessary.



g) How long is the resulting data sent? Is this coding on average more efficient than the previous approach?

The new coding additionally allows you to **detect and correct** one flipped bit in each **channel word**.



h) What is the probability $\text{Pr}[\text{incorrect}]$ that the transmission cannot be decoded correctly?

Note: For this sub task, **round to four digits of accuracy**. You are allowed to calculate using rounded interim results

Additional space for solutions—clearly mark the (sub)problem your answers are related to and strike out invalid solutions.

