ELTE
EÖTVÖS LORÁND
UNIVERSITY

Cryptography and security (IPM-18sztKVSZKRBG)

Assignment IV

Minimum requirements

- There are 7 different IT security related problems described in this paper with varying difficulties;
- You need to collect at least **10 points** from this assignment;
- Always provide step by step solutions;
- Submit your final solution in **one PDF** file including every necessary source code;
- Please send your solutions to **ntihanyi@inf.elte.hu** till **20th May 2022**;
- It is **strictly prohibited** to share your solutions with others.

Challenge #1 - scrypt

You were able to dump the Administrator scrypt hash from a database. The password can be found among the top 1000 weakest passwords.

```
Administrator:8:8:1:c2VjcmV0X3NhbHQ:GydhHNhVQP4zvPGf2I/kgzQ6onwF4/+mxWKOmcY+BWA
```

Questions:

a. Recover the cleartext password. (**1 point**)

b. Implement the solution in python programming language. (**3 points**)

Challenge #2 - MD5 cracking

You were able to dump an MD5 hash from a database. It is known that the cleartext of this hash is a Hungarian vehicle registration plate. Examples: “AAA-001”, “CFG-888”.

```
MD5=493ff01f9fd0ccf331f070aebfab3534
```

Questions:

a. Recover the cleartext password using hashcat. (**1 point**)

b. Implement the solution in any chosen programming language. (**3 points**)

Challenge #3 - Wrong implementation

Something wrong with this PHP code:

```php
<?php
$ELTE_hashed_password=
"0e000000000000000000000000000000000000000000000000000000000000";
$mypassword = readline("Please enter your password: ");
if ($ELTE_hashed_password == hash('sha256',$mypassword))
    {echo "Access GRANTED\n";}
else {echo "Access DENIED\n";}
?>
```

Questions:

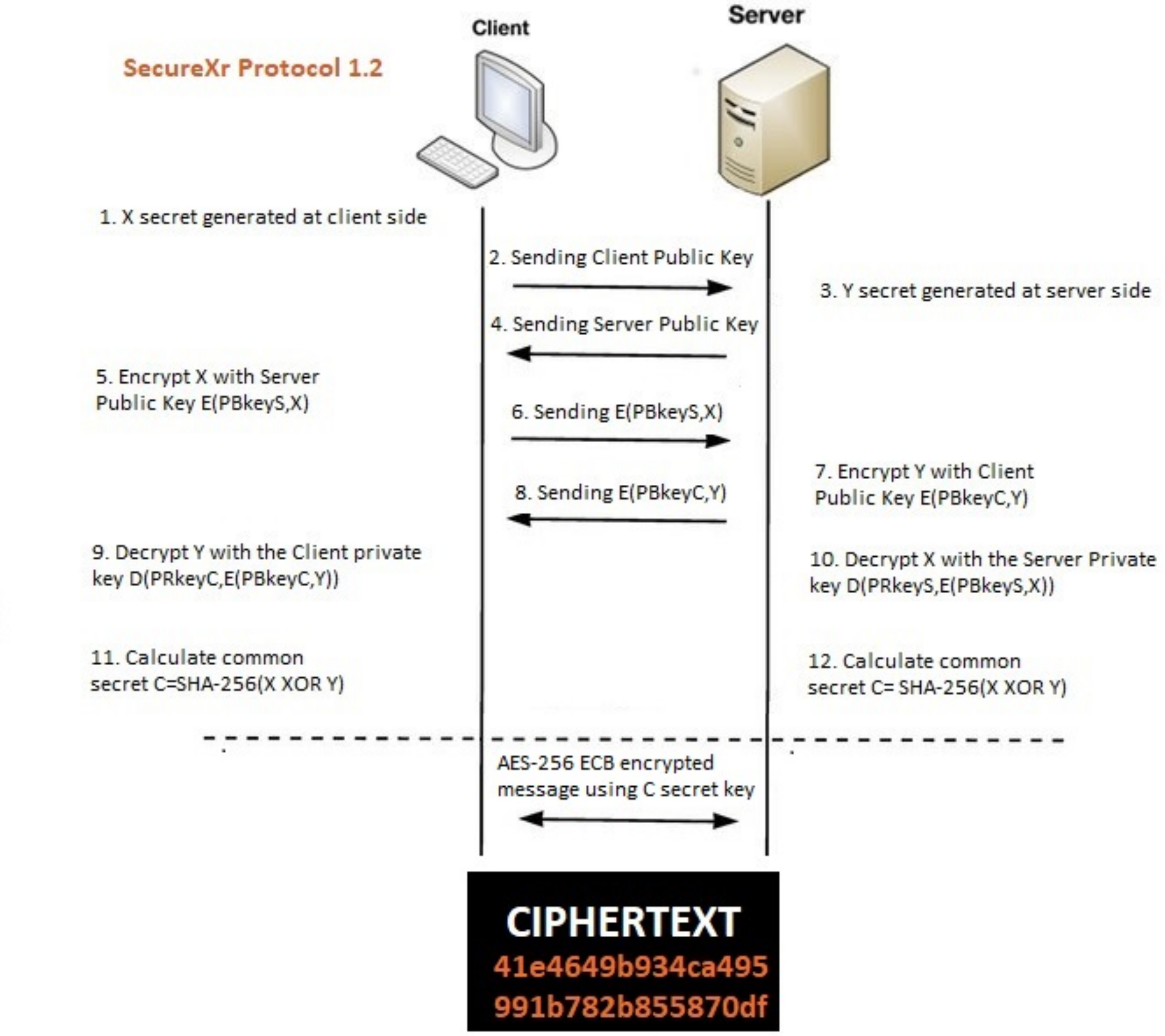
a. Find a password for which the algorithm returns “Access GRANTED”. (**4 points**)

b. Modify the code to make it secure (**1 point**)

HINT: https://link.springer.com/chapter/10.1007/978-3-030-68884-4_8

Challenge #4 - Insecure protocol

The following protocol is used for transmitting secret information between a client and a server.


SecureXr Protocol 1.2
Client
Server
1. X secret generated at client side
2. Sending Client Public Key
3. Y secret generated at server side
4. Sending Server Public Key
5. Encrypt X with Server Public Key E(PBkeyS,X)
6. Sending E(PBkeyS,X)
7. Encrypt Y with Client Public Key E(PBkeyC,Y)
8. Sending E(PBkeyC,Y)
9. Decrypt Y with the Client private key D(PRkeyC,E(PBkeyC,Y))
10. Decrypt X with the Server Private key D(PRkeyS,E(PBkeyS,X))
11. Calculate common secret C=SHA-256(X XOR Y)
12. Calculate common secret C= SHA-256(X XOR Y)
AES-256 ECB encrypted message using C secret key
CIPHERTEXT
41e4649b934ca495
991b782b855870df

Questions:

a. Identify weaknesses and possible vulnerabilities in the protocol. (**3 points**)

b. Suggest improvements to be compliant with FIPS 140-2 standards. (**2 points**)

c. Implement the protocol in any chosen programming language. (**5 points**)

Challenge #5 - Linear cryptanalysis

Linear cryptanalysis described by Mitsuru Matsui who first applied the technique to the FEAL cipher in EUROCRYPT ’92. We have an 8 bits plaintext, ciphertext and key (P,C,K). We know the following linear expressions:

$$P \oplus C = 0x01010101$$

$$\begin{aligned}
P_1 \oplus P_4 \oplus P_3 \oplus C_1 \oplus C_5 &= K_4 \\
P_3 \oplus P_6 \oplus P_1 \oplus C_1 \oplus C_3 &= K_8 \\
P_3 \oplus P_6 \oplus P_8 \oplus C_2 \oplus C_8 &= K_6 \\
P_3 \oplus P_2 \oplus P_7 \oplus C_5 \oplus C_8 &= K_1 \\
P_5 \oplus P_4 \oplus P_7 \oplus C_6 \oplus C_2 &= K_7 \\
P_7 \oplus P_3 \oplus P_1 \oplus C_3 \oplus C_8 &= K_4 \\
P_1 \oplus P_3 \oplus P_5 \oplus C_7 \oplus C_7 &= K_2 \\
P_5 \oplus P_8 \oplus P_7 \oplus C_2 \oplus C_3 &= K_1 \\
P_7 \oplus P_3 \oplus P_7 \oplus C_1 \oplus C_7 &= K_3 \\
P_6 \oplus P_7 \oplus P_2 \oplus C_5 \oplus C_1 &= K_7 \\
P_1 \oplus P_8 \oplus P_6 \oplus C_3 \oplus C_4 &= K_8 \\
P_1 \oplus P_3 \oplus P_7 \oplus C_2 \oplus C_1 &= K_5 \\
P_3 \oplus P_5 \oplus P_1 \oplus C_8 \oplus C_3 &= K_3 \\
P_2 \oplus P_6 \oplus P_7 \oplus C_2 \oplus C_6 &= K_2 \\
P_8 \oplus P_1 \oplus P_7 \oplus C_4 \oplus C_7 &= K_5 \\
P_1 \oplus P_2 \oplus P_3 \oplus C_4 \oplus C_5 &= K_6
\end{aligned}$$

Questions:

a. Find a valid Plaintext, Ciphertext and Key. (**4 points**)

b. Implement the solution in any chosen programming language. (**4 points**)

Challenge #6 - MySQL3.23 hash cracking

We have the following MySQL 3.23 hash:

```
789abffc71d4fbbe
```

Questions:

a. Recover the cleartext password. (**4 points**)

b. Recommend a more secure hash function. (**1 point**)

Challenge #7 - Cascade Ciphers

“Cascade Ciphers: The Importance of Being First” written by Maurer and Massey in 1993. The article can be downloaded from the internet:
https://crypto.ethz.ch/publications/files/MauMas93a.ps

Questions:

a. What is the main conclusion of the article? (25-30 sentences) (**4 points**)