

Kryptografiopgaver bilag

Historisk kryptografi

1. Caesar ROT

Jeg har klartekst: Hej jeg hedder alperen

Kryptotekst: urw wrt urqqre nycrera

Algoritme: ROT-13

Modtaget fra min medstuderende: Alend

Kryptotekst: Urw wrt urqqre Nyraq

Klartekst: hej jeg hedder Alend

fra mig: til Alend

The screenshot shows the Cryptool software interface. On the left, under 'Recipe', there is a list with 'ROT13' selected. Under 'Input', the text 'hej jeg hedder alperen' is entered. Under 'Output', the encrypted text 'urw wrt urqqre nycrera' is displayed. The interface includes various buttons for file operations and a status bar at the bottom.

fra medstuderende Alend

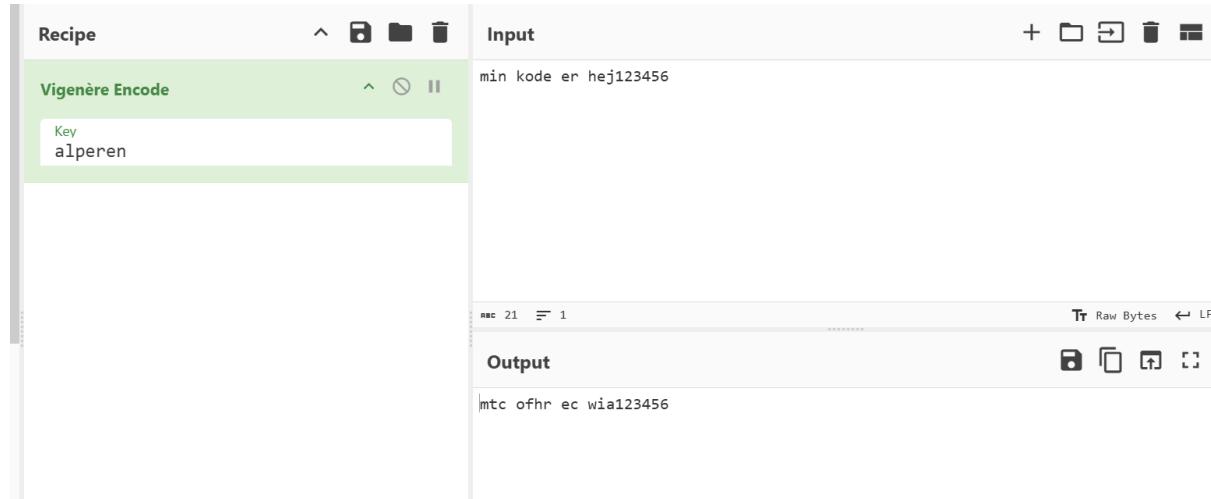
The screenshot shows the Cryptool software interface. On the left, under 'Recipe', there is a list with 'ROT13' selected. Under 'Input', the encrypted text 'Urw wrt urqqre Nyraq' is entered. Under 'Output', the decrypted text 'Hej jeg hedder Alend' is displayed. The interface includes various buttons for file operations and a status bar at the bottom.

2. Vigenére

Fra mig til alend

Min key: alperen

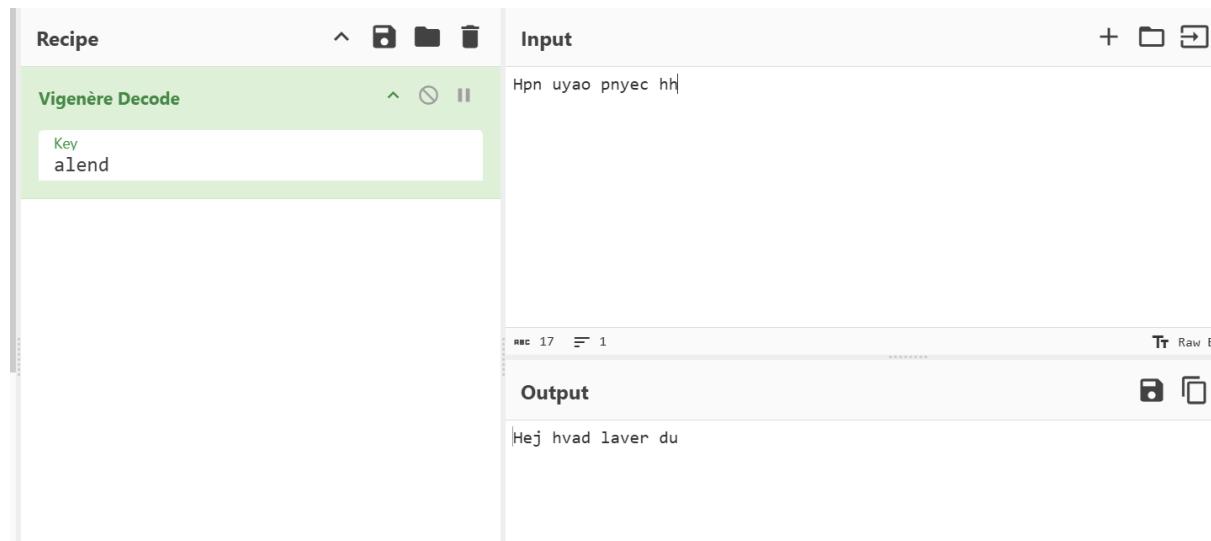
Output: mtc ofhr ec wia123456



Fra alend til mig

hans key: alend

Output: Hpn uyao pnyec hh



3. Steganografi

Vi gør brug af denne <https://stylesuxx.github.io/steganography/>



The screenshot shows a web-based steganography tool. At the top, there's a browser-like header with back, forward, and search buttons, followed by the URL 'stylesuxx.github.io/steganography/'. Below the header is a file input field containing the path 'Vælg fil 162002597-5e9c2b73-2e73-47e2-8b4d-5c49aa1dcc97.png'. To the right of the file input is a 'Decode' button. Further down, there's a section titled 'Hidden message' with a descriptive text about steganography and a 'Decode' button. The main area is labeled 'Input' and contains a blurred image of a person's face.

Hidden message

Steganografi (af græsk steganos, "dække", og gráphein, "at skrive") er et underemne inden for kryptologien, der beskæftiger sig med at skjule beskeder i en eller anden form for kontekst. Steganografi adskiller sig fra kryptering, da man ved kryptering forudsætter, at en opponent kender til beskedtransporten. Ved steganografi prøver man i stedet at

Input



dette er mit billede jeg sender til alend.



The screenshot shows the same steganography tool interface. The file input field now contains 'fenerbahce-logo-1579861520-30402.jpg'. The message input field contains 'hej alend'. To the right of the message input is an 'Encode' button.

Binary representation of your message

0110100011001010110101000100000011000010110110001100101011011001100100

Original



dette er det jeg har modtaget fra alend som jeg har decryptet

Vælg fil alperen.png

Decode

Hidden message

Hej alperen kan du se denne
besked

Input

4. (Ekstraopgave) Enigma og Bomba

ENIGMA

Recipe

Enigma

Model
3-rotor

Left-hand rotor
EKMFLGDQVZNTOWYH...

Left-hand rotor ring setting
A

Left-hand rotor initial value
A

Middle rotor
AJDKSIRUXBLHWTMC...

Middle rotor ring s...

Middle rotor initial...
A

Right-hand rotor
BDFHJLCPRXVZYNE...

Right-hand rotor ring setti...
A

Right-hand rotor initial val...
A

Reflector
AY BR CU DH EQ FS GL IP JX KN M...

Plugboard

Strict output

Input

Hej jeg hedder alperen

Output

ILDZR ZTYRZ NLBYD NPZY

Besked brydes med bombe: Se decryption preview.

ILDZR ZTYRZ NLBYD NPZY

Output

Rotor stops	Partial plugboard	Decryption preview
AAZ	??	WPAAUYRXTCHOZFBOBC
AAA	??	HEJJEGHEDDERALPEREN
AAB	??	OBLYTGRGXYMEECSMKFM
AAC	??	GDIGNKHGVVTYHQASANE
AAD	??	QUVGTCAZLYSVPIMYEJJ

Moderne kryptografi

1. Symmetrisk kryptering

Afprøv DES, Triple DES og AES i Cyberchef. Send en krypteret besked, og afkod den når modtaget.

DES

Fra mig til alend

Min key: zealandd

iv: zealandd

Output: 2a8c5b77bb15447c0a62ec8ac7c54f9efc25f92f8c96d37a

mangler screen

Fra alend til mig

hans key: Cykelhah

IV: Hjelmsdd

Output: 904ae99c3295b245

Last build: A month ago - Version 10 is here! Read about the new features [here](#)

Recipe

DES Decrypt

Input

904ae99c3295b245

Key: Cykelhah

IV: Hjelmsdd

Mode: CBC

Input: Hex

Output: Raw

Output

Hej

TRIPLE DES

Triple Des:

Fra mig til alend

Key: Zealandersejduer

IV: Zealandd

OUtput: 3e7447f9011706a91635ff10e27f53ca00041514c0365fd1

Fra alend til mig

Key: Min hemmelig nøgle xddd

IV:cykelhah

OUtput: 1046f18a5072cdd9dc1edea46d68b0efba3866c3f7b5faae

Recipe

Triple DES Encrypt

Key: Zealandersejduer **UTF8**

IV: Zealandd **UTF8** **Mode:** CBC

Input: Raw **Output:** Hex

Input: Zealand er det bedste

Output: 3e7447f9011706a91635ff10e27f53ca00041514c0365fd1

Recipe

Triple DES Decrypt

Key: Min hemmelig nøgle xddd **UTF8**

IV: cykelhah **UTF8** **Mode:** CBC

Input: Hex **Output:** Raw

Input: 1046f18a5072cdd9dc1edea46d68b0efba3866c3f7b5faae

Output: Hej jeg hedder alend

AES

Fra mig til alend

Key: zealandersejhejd

IV zealandersejhejd

OUtput: 6ff8bde309b92a65cc17032796104bb78c419907f1641fea74a0070b12b9f45b

The screenshot shows the CyberChef interface with an 'AES Encrypt' recipe selected. The input text 'hej mit navn er alperen' is being processed. The key is set to 'zealandersejhejd' and the IV to 'zealanderse...'. The mode is set to CBC. The output is displayed in Hex format as '6ff8bde309b92a65cc17032796104bb78c419907f1641fea74a0070b12b9f45b'.

Fra alend til mig

Key: Cykelhjelmxxxxxx

IV:Dukenderikkkoden

OUtput: 0e61b8d203d77ae768d2774b9a77a0193d3d9b324b8fe591ae19b336ef6553d6

The screenshot shows the CyberChef interface with an 'AES Decrypt' recipe selected. The input hex '0e61b8d203d77ae768d2774b9a77a0193d3d9b324b8fe591ae19b336ef6553d6' is being decrypted. The key is set to 'Cykelhjelmxxxxxx' and the IV to 'Dukenderikkkoden'. The mode is set to CBC. The output is displayed in Raw format as 'Kan du læse min bsked XD'.

2. Asymmetrisk kryptering

Skab et sæt RSA nøgler (public & private) med [openssl](#) eller CyberChef

- a. RSA Encrypt din besked med din makkers public key, (Encode med Base64) og send til din makker. Makker skal Decode med Base64 og bagefter RSA Decrypt med sin private key.

```

-----BEGIN PUBLIC KEY-----
MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQDWRMgjMTORB4coI/CZEDS4blym
CGkkfxFF+lpN2lsa4zf2PL4piGC2Q0//DJnQ+uoTGFY09ZPhgvuvsh54re62D0xM
m1joM5v0Idw1SohwK9H51r9HuHN04sbC1kR1AgktACTLwKFk5x8T7wrzMrU+/K01
ML0iaoDRXC172vf5QIDAQAB
-----END PUBLIC KEY-----

-----BEGIN RSA PRIVATE KEY-----
MIICXQIBAAKBgQDWRMgjMTORB4coI/CZEDS4blymCGkkfxFF+lpN2lsa4zf2PL4p
iGC2Q//DJnQ+uoTGFY09ZPhgvuvsh54re62D0xM1joM5v0Idw1SohwK9H51r9H
uHN04sbC1kR1AgktACTLwKFk5x8T7wrzMrU+/KQ1ML0iaoDRXC172vf5QIDAQAB
AoGADV5jQ1baicz3ch1NUeekndITo+tx7op4LzbD4p1rtIjZOCpek9bgEnb1lg
63oMSr707GYEIQjkHTU2036+qv4VNTTxLyGVyLDC8j7z+kzTEwqoUMEy6xEOnZ2
Dyj5Q2TsLHD7D7xwTsQ1f9Y53P9Ajuv519fQ8de50eEugIECQDr8C104hev27dx
Wo9dA1L7p7MYc5yC9nBzOoCQERq05D78U+K/DjkcNcLyISRQYMc7iDEAmGv8CLof
KBUWhpJFAkEA6HzpACxVL5oLeFBQdgiN8uttR7j0Kwdf9zkyBCoMd4AP/7YDD9r
rtxYB1JJXE6hoZVghx1GF5Q5D9yM/qSnIQJBj0iziIGrRChz2Aod/LxzqBq/gvP
iuk/SgNPLAFie+BIttag/ExLaH01zB9+387Dz0q0yMViowUQRqrhQ7Xm6FECQQCV
R4dQ3iaAnzgZPw5bcpxVzX2X09NqL0nwLQUV1WQ6SOL+NIPNbko1y6Pb5FIQYakDs
gP7v1wPbpggrwDskUtMhAkAmI4G4nwlv2VL9LyHKfd02859EpLo8aoIPQZodprdy
y8LU3tPZmw6QAbjQ4Pk1L6CPO3YzKGHheR8UR11raXo
-----END RSA PRIVATE KEY-----

```

Ln 17, Col 65 | 1.160 tegn | 100% | Windows (CRLF) | UTF-8

Recipe	Input
RSA Encrypt	alperen

RSA Public Key (PEM)

```

-----BEGIN PUBLIC KEY-----
MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQDWRMgjMTORB4coI/CZEDS4blym
CGkkfxFF+lpN2lsa4zf2PL4piGC2Q0//DJnQ+uoTGFY09ZPhgvuvsh54re62D0xM
m1joM5v0Idw1SohwK9H51r9HuHN04sbC1kR1AgktACTLwKFk5x8T7wrzMrU+/K01
ML0iaoDRXC172vf5QIDAQAB
-----END PUBLIC KEY-----

```

Encryption Scheme: RSA-OAEP **Message Digest Algorithm**: SHA-1

Output

```

Raw Bytes LF
CAN( .µ•øÀ6•Su•dc3®ÀÑ9-8•#^' ?ð~&íSub;íÉRE•AÈbKjì•E;?jßÍÀ•êýçþþ^•}R•m•dc4BS ò@OïføqYNAKæ•ü
DLE BS •«•àRp1c•VTDC4J2gMSTXEMv•XB³/9ZK, Ü5 FF
HO«•üENG³•ydc1½-Øg!•FS

```

The screenshot shows the Enigma software interface. In the 'Input' pane, there is a large amount of encoded text. In the 'Output' pane, the decrypted message 'hej jeg hedder alend' is displayed.

Recipe: RSA Decrypt

RSA Private Key (PEM):

```
--BEGIN RSA PRIVATE KEY-----
MIICXIAIBAAKBgQC7ZC/WhmQRyIM0cvwF2d9+
36BFexmNAUbUxU3tbnnB16/oq5HT
-----END RSA PRIVATE KEY-----
```

Key Password: (empty)

Encryption Scheme: RSA-OAEP

Message Digest Algorithm: SHA-1

Input: (Large encoded string)

Output: hej jeg hedder alend

rsa sign

The screenshot shows the Enigma software interface. In the 'Input' pane, the message 'hejalend' is entered. In the 'Output' pane, the signed message is shown.

Recipe: RSA Sign

RSA Sign:

```
-----BEGIN RSA PRIVATE KEY-----
QVmxRmf8+dg0CD7Uq2Hnom4Wg2qTg/T33GTE
nc0LIuE=
-----END RSA PRIVATE KEY-----
```

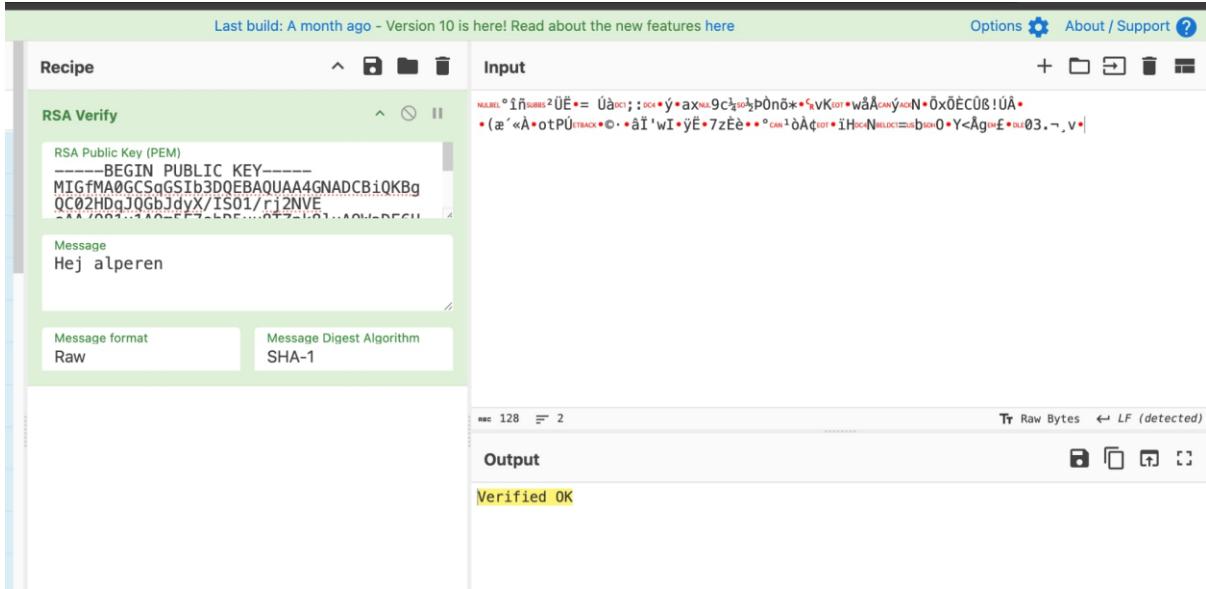
Key Password: (empty)

Message Digest Algorithm: SHA-1

Input: hejalend

Output: (Large signed message)

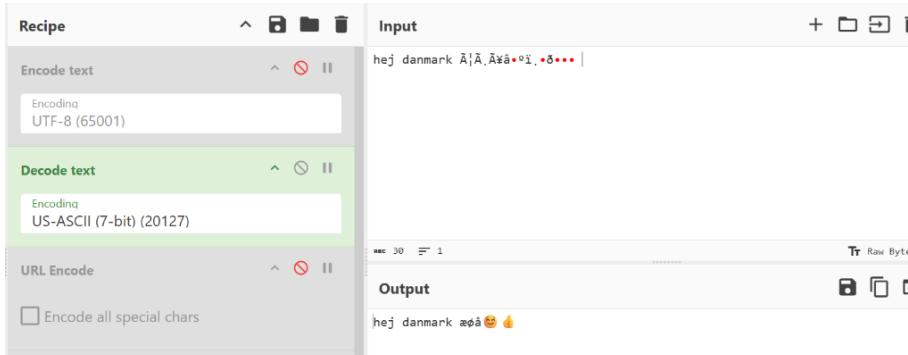
b. makker. Din makker skal *RSA Verify* med din public key.

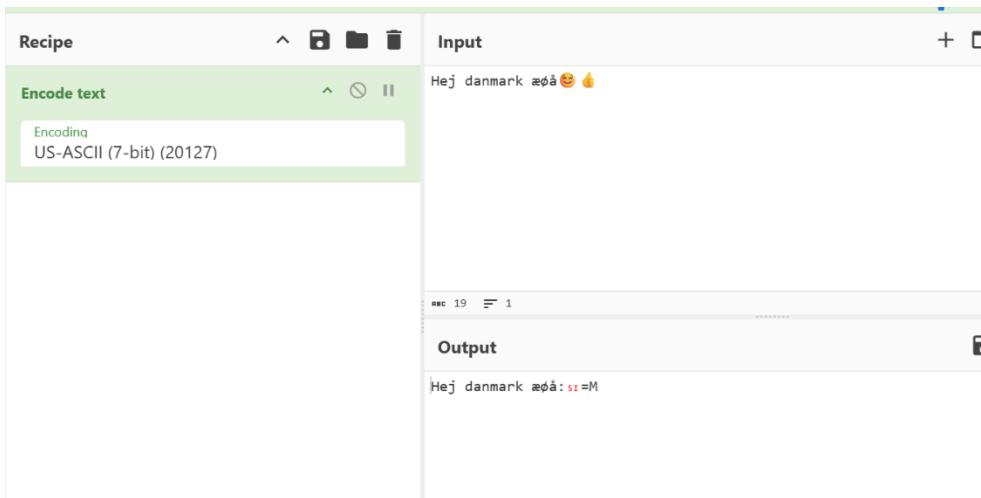


jeg har rsa verify min makkers alends med hans public key.

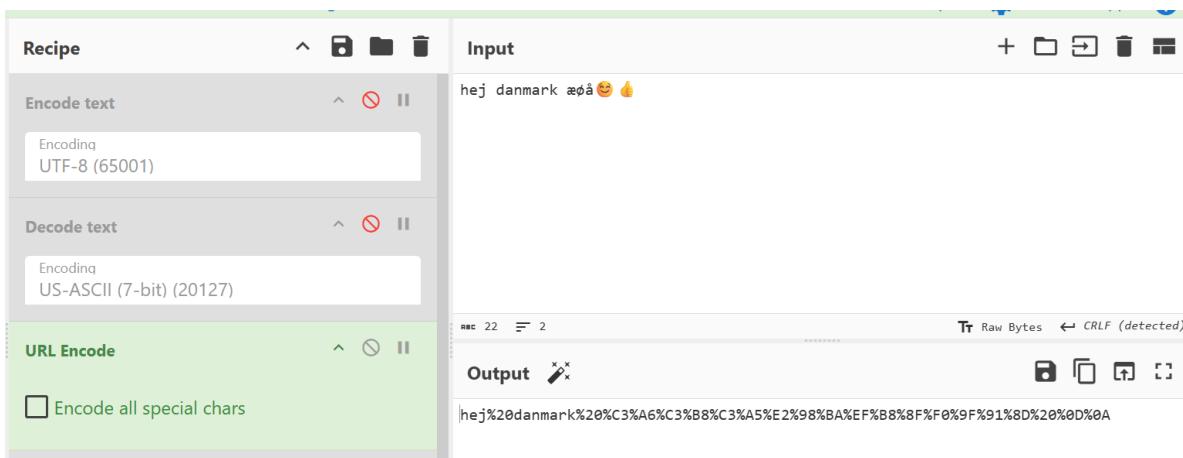
3. Encoding

- Afprøv encoding på en dansk tekst, som indeholder ÆØÅ og emojis 😊 👍
- Prøv at konvertere UTF-8 til ASCII, og læg mærke til datatabet





Prøv URL Encode



Prøv Base64 og Base32

The screenshot shows the CyberChef interface with a "Recipe" sidebar on the left and an "Input" and "Output" area on the right.

- Input:** "hej danmark æøå😊🔥"
- Output:** "aGVqIGRhbm1hcmsgw6bDuM0l4pi677iP8J+RjSANcg=="
- Recipe (Steps):**
 - Decode text (Encoding: US-ASCII (7-bit) (20127))
 - URL Encode (Encode all special chars checked)
 - To Base32 (Alphabet: A-Z2-7=)
 - To Base64 (Alphabet: A-Za-z0-9+/=)

The screenshot shows the CyberChef interface with a "Recipe" sidebar on the left and an "Input" and "Output" area on the right.

- Input:** "hej danmark æøå😊🔥"
- Output:** "NBSWUIDEMFXG2YLSNMQMHJWDXDB2LYUYXLX3RD7QT6IY2IANBI=====
- Recipe (Steps):**
 - Decode text (Encoding: US-ASCII (7-bit) (20127))
 - URL Encode (Encode all special chars checked)
 - To Base32 (Alphabet: A-Z2-7=)

● 4. PGP

Afprøv også PGP i Cyberchef, hvor du krypterer og signerer en besked, og du dekrypterer og verificerer. (Du kan lave dine nøgler i cyberchef med PGP Generate Keypair.)

Først laver jeg mine nøgler med PGP Generate Keypair. Derefter kryptere jeg og signere en besked.

PGP Encrypt and Sign

```

-----BEGIN PGP PRIVATE KEY BLOCK-----
Private key of signer
6DRm804=
=bfj3
-----END PGP PRIVATE KEY BLOCK-----

Private key passphrase
hej123

-----BEGIN PGP PUBLIC KEY BLOCK-----
Private key of recipient
PKqig1oJ/F02tJgg81CYXFW9iMt
qjxiMH4cwSLk1TbZnChduJo=
=gXZU
-----END PGP PUBLIC KEY BLOCK-----

```

Output

```

-----BEGIN PGP MESSAGE-----
Version: Keybase OpenPGP v2.1.15
Comment: https://keybase.io/crypto

WYwD0fURSBrsa0sBBACOLJDDcobOTEyy7D4bkYFcwAYqBkdm0R9Qf7Z421HCKmVA
-----END PGP MESSAGE-----

```

Her dekryptere og verificerer jeg beskeden.

PGP Decrypt and Verify

```

-----BEGIN PGP MESSAGE-----
W166eo61GvEcKHPYTxdtYEbpuyk3h
hvwhG07qcxv1lRJF6DRm804=
=7qCL
-----END PGP PUBLIC KEY BLOCK-----

-----BEGIN PGP PRIVATE KEY BLOCK-----
Private key of recipient
hChduJo=
=q7/A
-----END PGP PRIVATE KEY BLOCK-----

Private key password
hej123

```

Output

```

Signed by PGP key ID: 68596268
PGP fingerprint: 9abaca32d22efcbc95d1d8317969b94b6b596268
Signed on Wed, 24 Sep 2015 12:30:47 GMT
-----
hej mit navn er alperen og jeg er sei

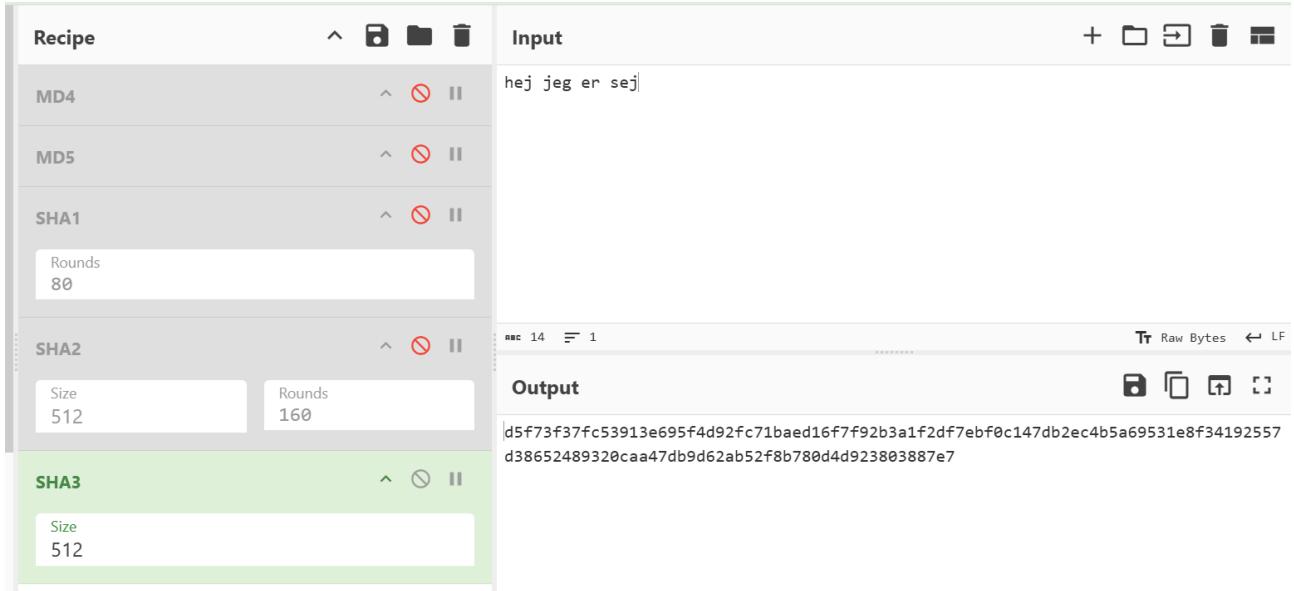
```

5. Hashing

Lav en kort besked, og beregn forskellige hashværdier af den (MD4, MD5, SHA1, SHA2, SHA3).

- b. Send dem til din makker.
- c. Din makker skal vha. hashværdien verificere, at beskeden er ægte.
- d. Gentag evt. med at beregne hashen af en fil

Beregning af hashværdier:



Beregning af hash af en fil: bruger powershell.

```
Windows PowerShell

PS C:\Users\alper> Get-FileHash -Path "C:\Users\alper\Documents\hashingtest.txt" -Algorithm MD5
Algorithm      Hash                                     Path
-----      ----
MD5          2514EBD26B8489510E5136DA49F4A2EE          C:\Users\alper\Documents\hash...

PS C:\Users\alper> Get-FileHash -Path "C:\Users\alper\Documents\hashingtest.txt" -Algorithm SHA1
Algorithm      Hash                                     Path
-----      ----
SHA1          9A78C6B9F8EBA4BE72691287E94794D9D127F0A7          C:\Users\alper\Documents\hash...

PS C:\Users\alper> Get-FileHash -Path "C:\Users\alper\Documents\hashingtest.txt" -Algorithm SHA256
Algorithm      Hash                                     Path
-----      ----
SHA256        EF61F52771BB875E25DBAE407C678AEFAC5311BF057E296247153D0C822289D7          C:\Users\alper\Documents\hash...

PS C:\Users\alper> Get-FileHash -Path "C:\Users\alper\Documents\hashingtest.txt" -Algorithm SHA512
Algorithm      Hash                                     Path
-----      ----
SHA512        B1C10A5362274E9D427377D567B914F68BD96464222C840DE9A895635B7D507FFDD...          C:\Users\alper\Documents\hash...
```

6. Cracking med crackstation

Lav en svag hash af et simpelt, engelsk password. Din makker skal cracke hashen med [Crackstation](#). Snak om, hvordan “salt” kan ændre billedet.

Last build: 3 months ago - Version 10 is here! Read about the new features [here](#)

Options About / Support

Recipe	Input	Output
MD5	admin1	e00cf25ad42683b3df678c61f42c6bda

CrackStation

Defuse.ca · Twitter

CrackStation · Password Hashing Security · Defuse Security

Free Password Hash Cracker

Enter up to 20 non-salted hashes, one per line:

e00cf25ad42683b3df678c61f42c6bda

Jeg er ikke en robot

Servicevilkårene for reCAPTCHA ændres. Foretag handling.
reCAPTCHA
Privatliv · Vilkår

[Crack Hashes](#)

Supports: LM, NTLM, md2, md4, md5, md5(md5_hex), md5-half, sha1, sha224, sha256, sha384, sha512, ripeMD160, whirlpool, MySQL 4.1+ (sha1(sh1_bin)), QubesV3.1BackupDefaults

Hash	Type	Result
e00cf25ad42683b3df678c61f42c6bda	md5	admin1

Recipe	Input
MD5	password123

Raw Bytes LF

Hash	Type	Result
482c811da5d5b4bc6d497ffa98491e38	md5	password123

The screenshot shows the CrackStation interface. At the top, there's a navigation bar with 'CrackStation', 'Password Hashing Security', and 'Defuse Security'. On the right, there are links for 'Defuse.ca' and social media icons for Twitter and YouTube. Below the navigation is a title 'Free Password Hash Cracker'.

A text input field contains the MD5 hash: 482c811da5d5b4bc6d497ffa98491e38. To the right of the input is a reCAPTCHA verification box with the message 'Jeg er ikke en robot' and a checkbox. Below the input field is a note about supported hash types: 'Supports: LM, NTLM, md2, md4, md5, md5(md5_hex), md5-half, sha1, sha224, sha256, sha384, sha512, ripeMD160, whirlpool, MySQL 4.1+ (sha1(shai1_bin)), QubesV3.1BackupDefaults'.

Below the input field is a table with three columns: 'Hash', 'Type', and 'Result'. The first row shows the hash '482c811da5d5b4bc6d497ffa98491e38' with type 'md5' and result 'password123'. A legend at the bottom explains color coding: green for exact match, yellow for partial match, and red for not found.

7. ECC Elliptic Curve Cryptography

Elliptic Curve Cryptography (EC eller ECC) er en anden moderne asymmetrisk krypto-algoritme ligesom RSA, og den kan yde samme sikkerhed med kortere nøgler. Desværre er der ikke kryptering og dekryptering med EC i CyberChef, men du kan prøve at generere en key-pair med Generate ECDSA keypair.

Sign en besked med ECDSA, og verificer samme besked. (Hvis det driller i CyberChef, prøv med <https://emn178.github.io/online-tools/ecdsa/verify/>)

Først har jeg genereret en key-pair med ec i Cyberchef.

The screenshot shows the CyberChef interface. On the left, under 'Recipe', it says 'Generate ECDSA Key Pair'. It has two dropdown menus: 'Elliptic Curve' set to 'P-256' and 'Output Format' set to 'PEM'. The main area is titled 'Input' and 'Output'. The 'Output' section displays the generated public key in PEM format:

```
-----BEGIN PUBLIC KEY-----
MFkwEwYHKoZIzj0CAQYIKoZIzj0OAQcDQgAECCtH7KmYT051JTYut7eLBjrk4kZE
H04yA8BEjkTCY7WraCnBWhgb+bHv0HaxXOFyLx8nhqtyLuJ3KE/OiS/nVGg==
-----END PUBLIC KEY-----
```

ECDSA Sign

Message Digest Algorithm: SHA-256

Output Format: ASN.1 HEX

Input: hej

Output:

```
304502210095ee6a505f0a6160ed5ab54e39267f539fe200cbe403a6d6434c99dcca4681e302206bc
ecd2cc79c6ac261001e19d15ce01fdc16f22763d323486aae95ba461f7dd7
```

nu signerer jeg beskeden.

ECDSA Verify

Input Format: Auto

Message Digest Algorithm: SHA-256

Input:

```
304502210095ee6a505f0a6160ed5ab54e39267f539fe200cbe403a6d6434c99dcca4681e302206bc
ecd2cc79c6ac261001e19d15ce01fdc16f22763d323486aae95ba461f7dd7
```

Output:

```
Verified OK
```

jeg verificerer beskeden, output er som følgende verified ok.

8. Hashcat

Jeg har gennemført en øvelse i brute-force cracking af MD4-hashes med Hashcat. Først genererede jeg MD4-hashes af passwords med 3–7 bogstaver i CyberChef og gemte dem i en fil.

```
(kali㉿kali)-[~/Desktop]
$ hashcat -D 1 -O -m 900 -a 3 hashes.txt '?l?l?l?l?l?l?' --increment --increment-min=3 --increment-max=7

hashcat (v6.2.6) starting
OpenCL API (OpenCL 3.0 PoCL 6.0+debian Linux, None+Asserts, RELOC, SPIR-V, LLVM 18.1.8, SLEEP, DISTRO, POCL_DEBUG) - Platform #1 [The pocl project]
* Device #1: cpu-sandybridge-12th Gen Intel(R) Core(TM) i5-1235U, 1435/2934 MB (512 MB allocatable), 4MCU

Minimum password length supported by kernel: 0
Maximum password length supported by kernel: 55

Hashes: 5 digests, 5 unique digests, 1 unique salts
```

Jeg har brugt denne kommando: hashcat -D 1 -O -m 900 -a 3 hashes.txt '?l?l?l?l?l?l?' --increment --increment-min=3 --increment-max=7

og efter har jeg brugt denne kommando til at tjekke, hvilke hashes der var knækket:

```
cat /home/kali/.local/share/hashcat/hashcat.potfile
```

```
(kali㉿kali)-[~/Desktop]
$ cat /home/kali/.local/share/hashcat/hashcat.potfile
7da57b69cc4d05ccfc1cb3758c999973:alpi
9e68e57d46dcaad88ea688335a7fd7ff:alp
643a76b9441b7d65c9f5686eb97236ea:alper
95f640b5aef2492fc7aadcf8c247bd7e:alpere
d74cf1517f3fa737d5552c8d5693693c:alperen
```

9. Crack et passwordbeskyttet zip-fil

Efterprøv [denne øvelse](#) i Kali, hvor du laver en passwordbeskyttet zip-fil, og du cracker den bagefter.

Jeg har i denne øvelse i Kali, lavet en passwordbeskyttet zip-fil, og cracket den med fcrackzip.

```
(kali㉿kali)-[~/6.10]
$ mkdir 6.10; cd 6.10; touch one two three; zip -e numbers.zip one two three
Enter password:
Verify password:
adding: one (stored 0%)
adding: two (stored 0%)
adding: three (stored 0%)
```

```
(kali㉿kali)-[~/6.10/6.10]
$ ls /usr/share/wordlists/
amass dirb dirbuster dnsmap.txt fasttrack.txt fern-wifi john.lst legion metasploit nmap.lst rockyou.txt.gz sqlmap.txt wfuzz wifite.txt
(kali㉿kali)-[~/6.10/6.10]
$ sudo gzip -d /usr/share/wordlists/rockyou.txt.gz

(kali㉿kali)-[~/6.10/6.10]
$ fcrackzip -u -D -p /usr/share/wordlists/rockyou.txt numbers.zip

PASSWORD FOUND!!!!: pw == password
```

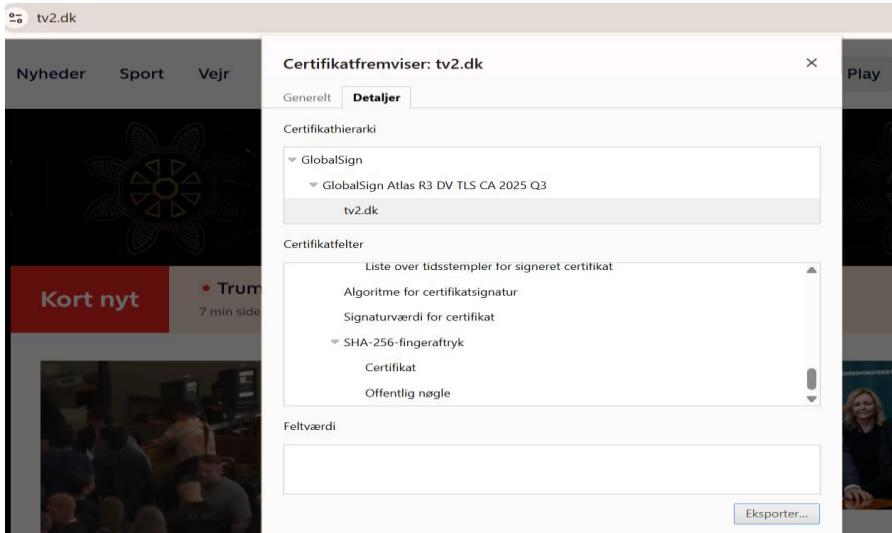
Resultat er således at, password er fundet, som er = password. øvelsen er gennemført succesfuldt og jeg har formået at cracke en pass-beskyttet zip-fil.

Dette viser at svage adgangskoder nemt kan knækkes og afsløres med en brute force angreb.

Anvendt kryptografi

1. TLS certifikater i browsere

Jeg har besøgt tv2, her har jeg undersøgt hvilket certifikat den bruger, og tv2 anvender et TLS-certifikat for at sikre forbindelsen mellem brugerens browser og serveren.



Jeg har også eksporteret den, her er følgende oplysninger:

The screenshot shows a certificate details window with the following information:

Felt	Værdi
Serienummer	014c557689e28dba27517ff6d...
Signaturalgoritme	sha256RSA
Hashalgoritme for signatur	sha256
Udsteder	GlobalSign Atlas R3 DV TLS CA...
Gyldigt fra	19. september 2025 19:02:12
Gyldigt til	21. oktober 2026 19:02:11
Emne	tv2.dk
Offentlig nøgle	RSA (2048 Bits)

Der er spændende informationer, som at certifikatet bruger RSA med sha256, som hashfunktion, samt hash algoritme er baseret på sha256.

2. Keybase.io

Afprøv Keybase.io til at sende sikre beskeder med. (Send til din makker, modtag fra din makker, signer en besked, og verificer en besked.)

(Ekstra: Kast et blik på [Keybase Book](#), hvor du kan lære om hvordan de sikrer informationsoverførsel.)

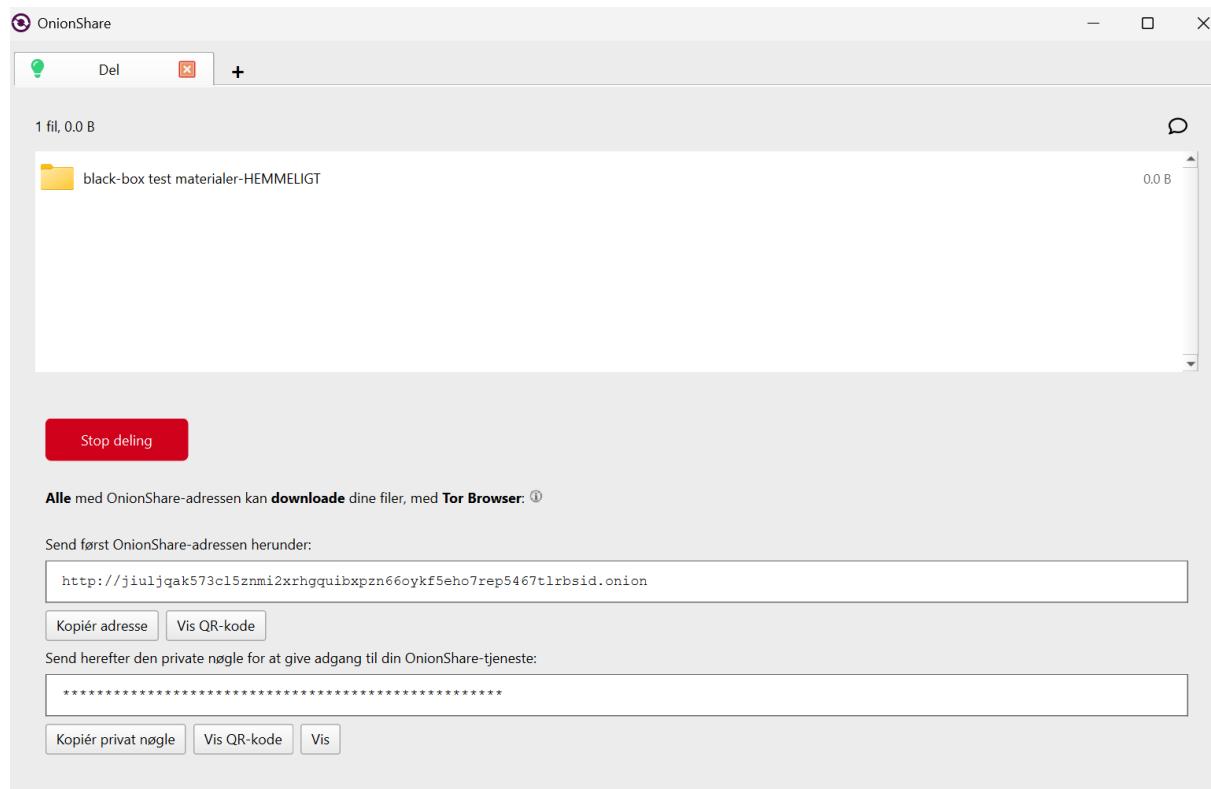
The screenshot shows a Keybase app interface with a sidebar and a main chat window. The sidebar has icons for user, emoji, team, file, checkmark, group, and settings. The main window shows a chat with a contact named "anmi_". The message "This conversation is end-to-end encrypted." is highlighted with a blue background and white text. The messages in the chat are:

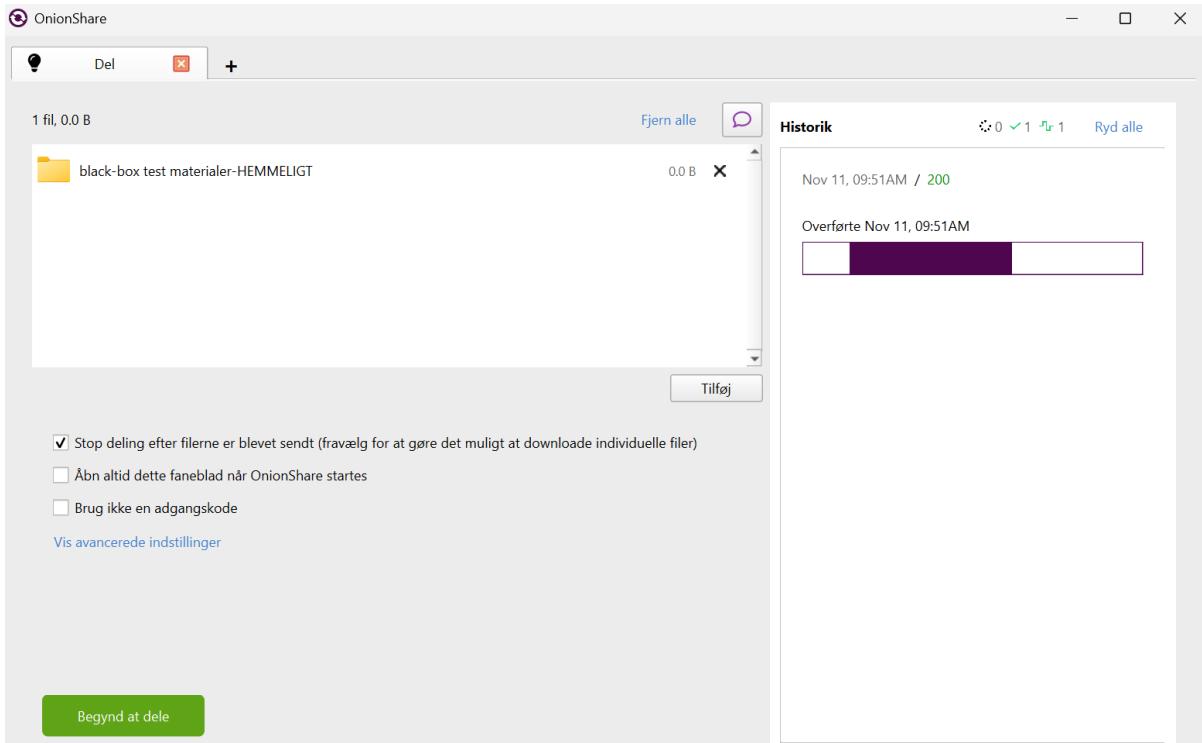
- alo001ao 9:28 AM: hej makker
- alo001ao 9:56 AM: er du klar til noget pentesting i en kommunal bygning kl.15:00
flere oplysninger kommer senere
- anmi_ 9:57 AM: Hej!
Ja, Vi ses kl 15

3. Onionshare

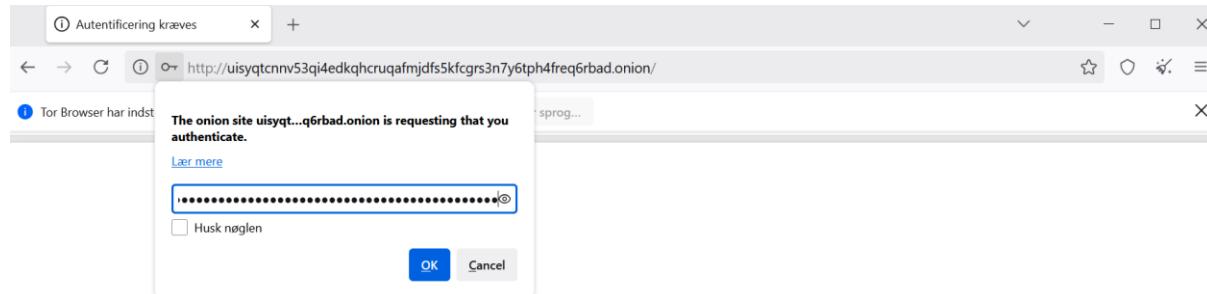
Send en fil til din makker sikkert med [OnionShare](#). Hvordan er det anderledes end Keybase?

Sender en fil til min makker:





Modtager fil fra makker:

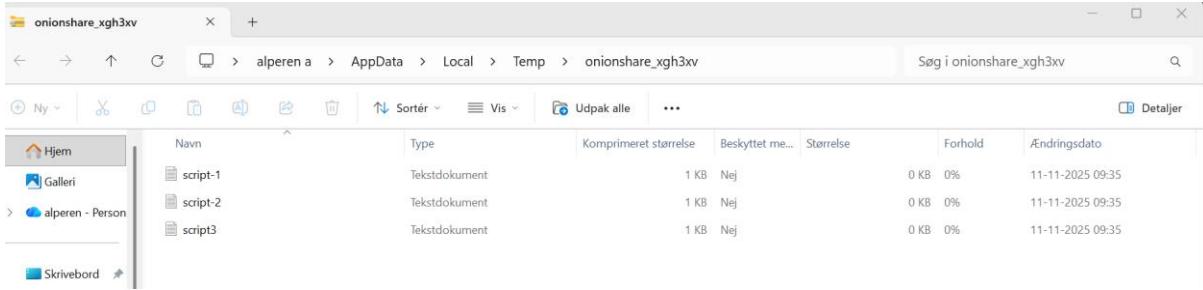


The image consists of three vertically stacked screenshots of a Tor browser window. The browser's address bar shows the URL <http://uisyqtcnv53qi4edkqhcrqafmjdfs5kfcgrs3n7y6tp4freq6rbad.onion/>.

Screenshot 1: A Tor browser warning dialog box is displayed. It says "The onion site uisyqt...q6rbad.onion is requesting that you authenticate." Below it is a password input field with placeholder "*****". There is a checkbox "Husk nøglen" (Remember password) and two buttons "OK" and "Cancel".

Screenshot 2: The main browser window shows the OnionShare interface. It displays a list of files: "script-1.txt", "script-2.txt", and "script3.txt", all with size 0.0 B. At the top right is a "Download Files" button. Below the file list is a message: "Total size: 326.0 B (compressed)".

Screenshot 3: The browser window now shows a modal dialog box titled "Åbner onionshare_xgh3xv.zip". It says "Du har valgt at åbne: onionshare_xgh3xv.zip som er: Compressed (zipped) Folder (326 bytes) fra: ...cnnv53qi4edkqhcrqafmjdfs5kfcgrs3n7y6tp4freq6rbad.onion". It asks "Hvad skal Tor Browser gøre med denne fil?". The options are: "Åbn med Windows Stifinder (standard)" (selected), "Gem fil", and "Gør dette automatisk med filer som denne fremover". There are "OK" and "Annuller" buttons.



4. Pcrypt

Undersøg [Pcrypt](#), som er en lokal virksomhed, der tilbyder kryptografi. Måske en praktikplads?

Jeg har undersøgt Pcrypt, synes det er fedt, at der er en lokal dansk virksomhed, som tilbyder en passwordmanager, som gør at man kan få et overblik over alle ens logins og koder. Samt kryptering foretages på en lokal enhed med 256 bit standard AES og ECC.

Pcrypt skal jeg søge som praktikplads, da det er fedt med at man kan hjælpe virksomheder med sikre adgangskoder og opfylde kravene til datasikkerhed.

5. Open source key management

Find og afprøv et open source password-værktøj, som kan bruges til sikker opbevaring og deling af passwords og andre “secrets”.

Jeg har valgt Bitwarden.

Her fremvises, hvordan man opbevarer password eller andet hemmeligt sikkert:

The screenshot shows the Bitwarden Password Manager interface. On the left is a dark sidebar with navigation links: 'Bokse', 'Send', 'Værktøjer', 'Rapporter', 'Indstillinger', 'Password Manager', 'Share your passwords', and 'Secrets Manager'. The main content area has a 'Komme i gang' section with three steps: 'Opret en konto', 'Importér data' (with a note about importing data from a CSV file), and 'Installér webbrowserudvidelser' (with a note about installing browser extensions). Below this is a table titled 'FILTRE' showing saved items:

	Alle	Navn	Ejer	⋮
<input type="checkbox"/>	facebook	test1	Mig	⋮
<input type="checkbox"/>	vmware ubuntu server: alperentechserver1	SHA256:tdv2FVJRQIUg5mZ/IWbQ4tkwSt+tn5w5v...	Mig	⋮
<input type="checkbox"/>	vmware windows server - server1	admin	Mig	⋮

En passwordmanager, som Bitwarden er noget jeg helt sikkert skal bruge i mit fremtidige arbejde indenfor it-sikkerhed, da man har med mange digitale identiteter og gøre med, desuden opbevarer man adgangskoder og andet hemmeligt sikkert, samt kan dele dem, den er også brugervenligt og beskytter digitale identiteter.

6. Kryptografi i din software

I denne opgave skal du undersøge, hvad Web Crypto API er for noget.

Jeg har besøgt <https://copilot.cloud.microsoft/> og spugt følgende:

Hvad er Web Crypto API, og hvad kan den bruges til? Forklar til en bachelorstuderende i it-sikkerhed

Copilot

Web Crypto API er en **indbygget JavaScript API** i moderne webbrowsere, som giver udviklere mulighed for at udføre **kryptografiske operationer** direkte i browseren — uden at skulle bruge eksterne biblioteker eller sende data til en server.

Her er en forklaring målrettet en bachelorstuderende i it-sikkerhed:

kan du give et eksempel på brug?

Copilot

Selvfølgelig! Her er et konkret eksempel på, hvordan du kan bruge Web Crypto API til at kryptere og dekryptere tekst med AES-GCM, som er en moderne og sikker symmetrisk krypteringsalgoritme.

🔒 Eksempel: Kryptering og dekryptering med AES-GCM

Jeg er blevet klogere på hvad Web Crypto API er og fået eksempel på brug, men synes ikke det er simpelt og gøre brug af, det kræver en vis viden om kryptografi.

7. Sikker e-mail?

I denne opgave skal du afprøve mulighederne for at sende sikker e-mail. Spørg copilot om:
Jeg har spurgt co-pilot følgende:

hvordan kan jeg sende sikker mail fra gmail (eller hotmail eller andet, erstat med din egen mailudbyder)
hvordan kan jeg sende sikker mail fra office 365 (din edumail på skolen)

Copilot

Her er en oversigt over, hvordan du kan sende sikker e-mail fra både Gmail (eller Hotmail/andre udbydere) og fra Office 365 (f.eks. din skolemail):

1. Sende sikker mail fra Gmail (eller Hotmail/andre)

Jeg har prøvet at sende en sikker mail fra min skolemail, men får følgende besked:

You can't sign or encrypt this message until the S/MIME extension is installed. Please contact your IT administrator for help
X
installing the extension.

Send

From: alo001@edu.zealand.dk



S/MIME extension er ikke installeret. men jeg har valgt at bruge Protonmail i stedet for, til at sende en sikker mail til min makker. Som er meget nemt og enkelt at kryptere mails på.

sikker mail

Fra  alp.zea <alp.zea@proton.me>
Til Alendlsmail57@gmail.com

 Denne besked udløber tirsdag den 25. november 2025 kl. 10:46

Hej sender en sikker mail til dig, husk vores møde kl.13.00 idag.
med venlig hilsen
Alperen

Jeg har fået en forståelse for hvordan end-to-end kryptering fungerer i praksis. Så det kun er afsender og modtager, der kan læse mailen. Med dette sørger man også for at beskytte sit eget data eller indhold som beskeder, det er vigtigt især hvis man arbejder med it-sikkerhed.¹

8. Læs artiklen og beskriv det i 6 bullet points (no AI)

<https://samsik.dk/cybersikkerhed/temaer/overgangen-til-kvantesikker-kryptografi/>

Jeg har læst artiklen, og beskrevet det i 6 bullet points for neden, som giver mest mening, det har givet mig en læring oplevelse, at man selv sætter bullet points, som gør at man husker hvad man har læst og får en god viden om emnet.

1. Den nuværende trussel fra kvantecomputere

Kvantecomputere giver nye muligheder for at køre algoritmer, de traditionelle computere kan ikke løse de problemer som kvantecomputere kan. En af de algoritmer er Shors algoritme, der kan bryde de mest udbredte og moderne kryptografiske algoritmer, med en kvantecomputer. Den kan bryde nogle af de mest kendte, som RSA.

¹ <https://proton.me/security/end-to-end-encryption>

2. Indsamlnu – dekryptér senere:

Selvom kvantecomputere ikke findes, kan en fremtidige kvantecomputer udgør en trussel allerede nu, kan angriberne allerede via et angreb, indsamle krypteret data og dekryptere det i fremtiden, når teknologien er tilpas moden.

3. Identifikation og digital signatur

Kvantecomputere vil gøre det muligt at kunne forfalske digitale signaturer og identiteter, der er lavet med ikke kvante sikre kryptografiske algoritmer, en forudsætning er at der er tillid til at digitale signaturer, at de ikke kan forfalskes, en trussel kan være tillidsvækkende mod digitale systemer, som vi bruger i dag.

4. Ny kryptografi er på vej

Det amerikanske National Institute of Standards and Technology (NIST) står bag de fleste kryptografiske standarder, som anvendes i dag. For at man kan forhindre truslen fra kvantecomputere, har NIST i 2016 startet en konkurrence, som skulle finde algoritmer til en ny standard, der vil kunne modstå angreb fra kvantecomputere. NIST's nye standard, post-quantum cryptography (PQC), forventes klar i 2024, der vil blive den første kvantesikre kryptografi.

5. Hvad man kan gøre nu

Det forventes at nuværende kryptografiske algoritmer med tiden vil blive udskiftet med kvantesikker kryptografi. Standardiseringen af NIST PQC forventes afsluttet i 2024, opdatering bør sættes i drift. og systemer med følsom eller værdifuld data, er specifikke grunde til at idrøftsætte kvantesikker kryptografi, allerede nu. samt almindeligt anvendte systemer og software, samt eksisterende systemer, bør opdateres og opgraderes til kvantesikker kryptografi.

6. Mulige løsninger

Hybrid algoritme kan være en mulig løsning, således at man kombinerer dem korrekt. Den ene er en eksisterende standardiseret algoritme, som f.eks. RSA, som er afprøvet, men sårbar for at blive brutt af en kvantecomputer, samt en kvantesikker algoritme, der potentelt er sårbar, da den er ny, men sandsynligvis modstå angreb fra en kvantecomputer. Resultat ved at kombinere dem korrekt er krypteringen både sikker nu og ved angreb med en kvantecomputer, ulemperne kan være flere ressourcer og øget kompleksitet forøger sandsynligheden for fejl, dette kan udgøre en sikkerhedsrisiko.²

9. Blockchain fra bunden

(Ekstra) Diskuter, hvordan I ville kunne udvikle blockchain fra bunden. Beskriv de væsentligste steps i et par sætninger.

Vi ville udvikle en blockchain på følgende måde:

Definere en blokstruktur med data, tidsstempel, hash og previous hash, derefter oprette en genesis-blok som den første blok i kæden, og tilføje nye blokke ved at beregne deres hash og linke til den forrige blok, desuden sikre integritet ved at bruge en kryptografisk hashfunktion (fx SHA-256).

(Ekstra ekstra) Lav en dummyudgave af denne blockchain.

Index: 0

Data: Genesis Block

Hash: 4f8a2c9d7e6b5a4c3d2e1f0a9b8c7d6e5f4a3b2c1d0e9f8a7b6c5d4e3f2a1b0

Index: 1

Data: Alice sender 5 BTC til Bob

Previous Hash: 4f8a2c9d7e6b5a4c3d2e1f0a9b8c7d6e5f4a3b2c1d0e9f8a7b6c5d4e3f2a1b0

Hash: 00d3e4f5a6b7c8d9e0f1a2b3c4d5e6f7a8b9c0d1e2f3a4b5c6d7e8f9a0b1c2d3

Index: 2

Data: Bob sender 2 BTC til Peter

Previous Hash: 00d3e4f5a6b7c8d9e0f1a2b3c4d5e6f7a8b9c0d1e2f3a4b5c6d7e8f9a0b1c2d3

Hash: 0034a1b2c3d4e5f6a7b8c9d0e1f2a3b4c5d6e7f8a9b0c1d2e3f4a5b6c7d8e9f0

Er blockchain valid? True

² <https://samsik.dk/cybersikkerhed/temaer/overgangen-til-kvantesikker-kryptografi/>