Crypto Lab
Sec-250
Your Name Here

Caesar Cipher

Caesar Cipher uses a key called a Shift, which transposes cleartext ‘x’ amount of times to the right, which results in ciphertext.

cleartext	c	a	t
Shift (Rotate +10)			
ciphertext	m	k	d

Encrypt & decrypt the following **without** online resources, just like the Romans!

1. (ROT3) smegael = **vphjdho**
2. (ROT4) frodo = **jvshs**
3. (ROT9) **gandalf** = pjwmjuo
4. (ROT13) **caesar salad** = pnrfne fnynq

Brute force time ... There’s no shift provided, so try all possibilities to decode!

5. **They are taking the hobbits to isengard** = Maxr tkx mtdbgz max ahuubml mh blxgztkw.
6. What is the shift? **ROT19**

Vigenère

Vigenère technique switches the alphabet used on each letter (called polyalphabetic), based upon a secret keyword. We start with a table of shifted alphabets:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	D
X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	C
Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	B
Z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

Now, let's encode the message "cyber is life" using the keyword "cncs". We start by repeating the key underneath the message, so the letters line up:

Message:	c	y	b	e	r	i	s	l	i	f	e
Key:	c	n	c	s	c	n	c	s	c	n	s
Encrypted:	e	l	d	-	-	-	-	-	-	-	-

Now, look for the row marked ‘c’ in the left-most column (the key), and the ‘c’ in the top-most row (plaintext). The intersection of that column and row contains ‘e’ (ciphertext).

7. Rinse & repeat for the remaining cyphertext, again **without** online resources … Vive le cryptage!

Message:	**c**	**y**	**b**	**e**	**r**	**i**	**s**	**l**	**i**	**f**	**e**
Key:	**c**	**n**	**c**	**s**	**c**	**n**	**c**	**s**	**c**	**n**	**c**
Encrypted:	**e**	**l**	**d**	**w**	**t**	**v**	**u**	**d**	**k**	**s**	**g**

Completed tabled

Railfence

When you rearrange plaintext in a "wave" pattern (down, down, up, up, down, down, etc.), it is called railfence encryption.

Ex: This plaintext of “happy” looks like this with 3 rails (read: 3 rows) -

```
h   y
 a p
  p
```

Then when read line-by-line, we get the cypertext of “hyapp”.

8. Decrypt the following ciphertext “crsyectba” built using 3 rails.
cybercats

9. Is Railfence a substitution or transposition cipher? Explain why.

It is a transposition cipher because the letters are rearranged but they are not changed(ie not substituted for other letters

---

Hashes

We learned how different hash functions produce a unique fixed string of data that should always match to verify data integrity, as long as the data doesn't change.

1. Open a browser to : https://defuse.ca/checksums.htm

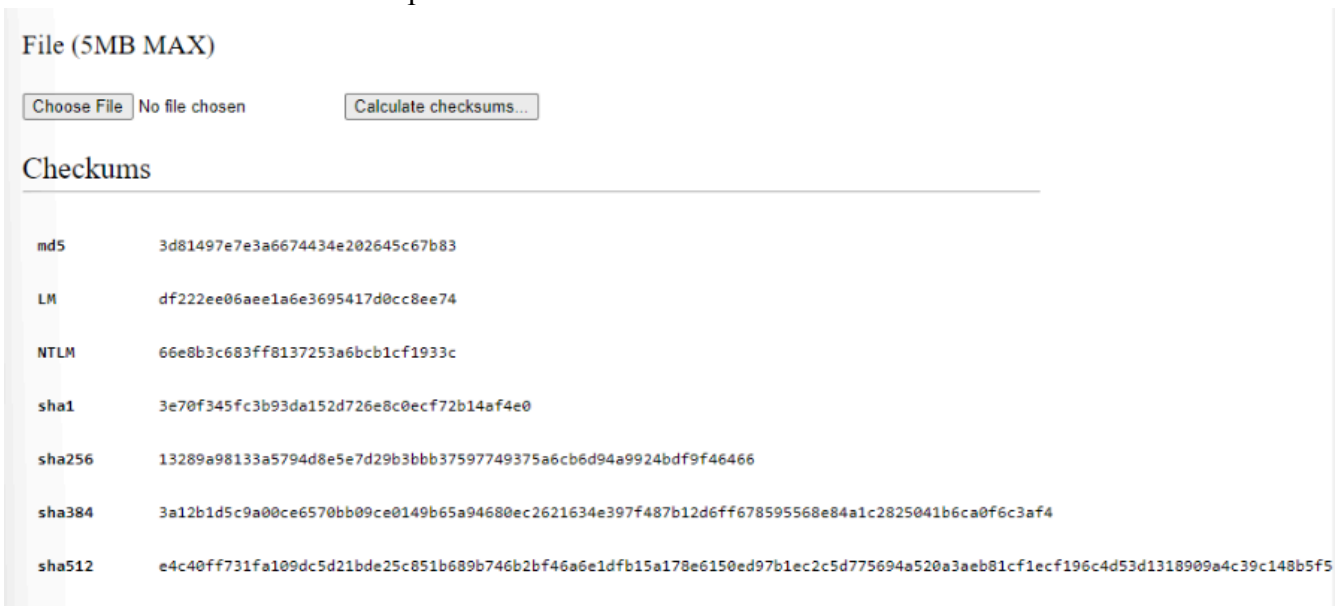
2. Enter the following phrase into the text area:

   Cryptography is as much fun as a person can have!

   a. Click "Calculate Checksums"

   Are the checksum lengths the same? **No**
   How many characters long is each checksum?
   MD5 = **32**
   SHA1 = **40**
   SHA512 = **128**
   b. Please take a Snip of results!

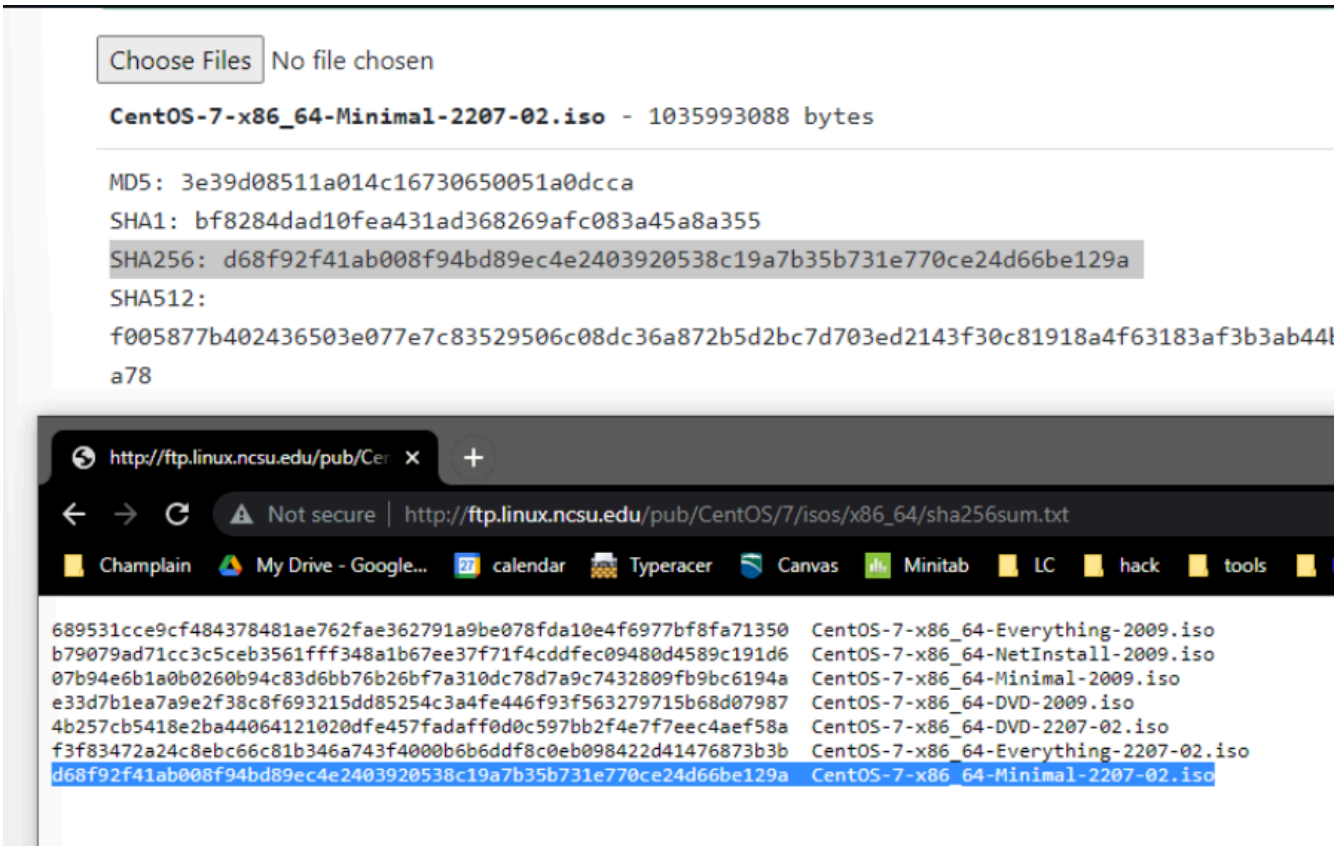
3. Change the phrase you typed in by removing one character and click "Calculate."
   a. Is the checksum different from the previous computation? **yes**
   b. Please take a Snip of results!



4. Next, we going to explore HMAC by opening up http://beautifytools.com/hmac-generator.php
5. Copy the same phase into the text box, then type a random string of text in the box "Key" that is displayed when you clicked on HMAC.
    a. Be sure your key is different than your partners.
    b. Is your HMAC checksum the same as your partners? Why or why not?
        i. **No it is not because we both used different keys**

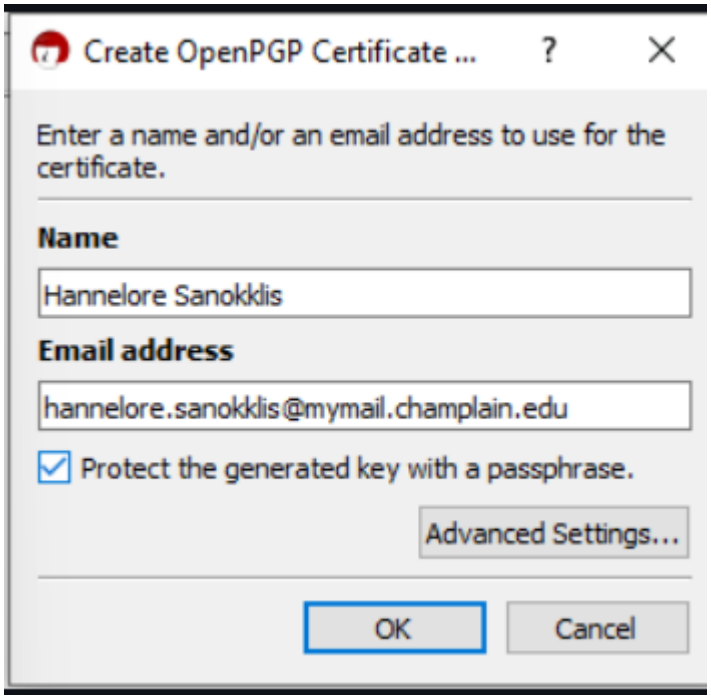
6. Both of you type in "SEC250" as the HMAC key.
    a. Is the checksum the same for you and your partners? Why or Why not?
        i. **Yes it is because we both used the same key**

7. Now, let's download & check an ISO's integrity, something you'll be doing in our program frequently.
8. First, download the CentOS-7 Minimal 1908 ISO here: http://ftp.linux.ncsu.edu/pub/CentOS/7/isos/x86_64/
    a. FYI: This is a popular Linux OS we use in CNCS
9. Also, open the sha265.sum.txt file. This checksum is what we are going to check the download's checksum against.
10. Now, let's go to https://md5file.com/calculator and select the ISO you just downloaded.
11. Once finished calculating, you should see its SHA-256 checksum.
    a. Do the both sets of checksums match? **yes**
    b. Provide one Snip including both the md5file.com's checksum + the sha1sum.txt file's checksum, validating the ISO's integrity.

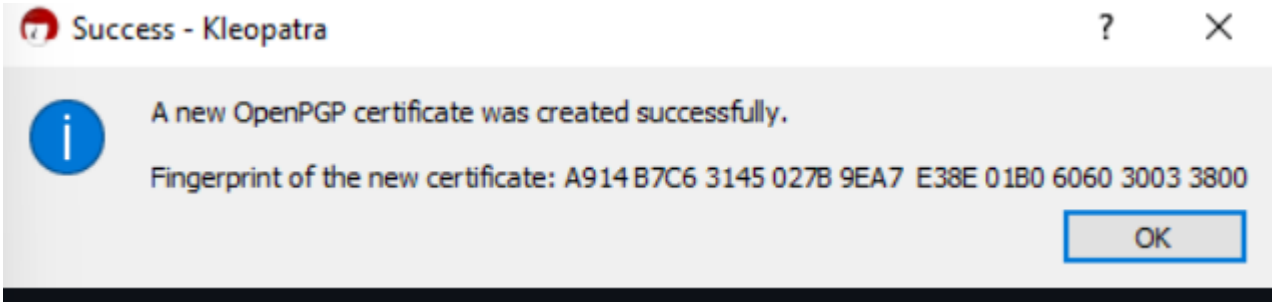


Fun with Public-Private Keys!

1. Download from gpg4win.org (select $0 to download for free) on your Windows 10 VM.
    a. There's an extensive product tech guide with screenshots on how to install & operate this tool https://files.gpg4win.org/doc/gpg4win-compendium-en.pdf
    b. Of course, it's a bit out of date, so the screens nor items may not be exactly the same (because everything online is perfect, right?).
    c. Your mission: Be curious, explore, test, and win!
    d. FYI: This is a small taste of higher level courses.
2. Start the install once downloaded:
    a. Achtung! During the install, please select **ALL** Components to install.
3. Creating a Key pair & Digital Certificate:
    a. File > Key Pair > Create a personal OpenPGP key pair
    b. Insert your First & Last name, along with an email address.
    c. Show all details
        i. Snip for your details.



    d. Create a passphrase for your private key; strong + something you will NEVER forget
        i. Snip your successfully created key pair's Fingerprint

4. 

5. File > Export your Secret Keys & name your output file your First_Last name, and then save it.
    a. Just like your car keys, please do not share nor lose this!
6. Now, open up a new program on your desktop, GPA or GNU Privacy Assistant
7. Once opened, you get what's about to happen … Key Management!
8. You should see your public & private key pair (gold + light blue-ish keys) with your name & email.
9. Download another key from https://ssl.intevation.de/Intevation-Distribution-Key.asc and save it to your Desktop.
    a. In the key that you download, note that everything between and including the blocks

        -----BEGIN PGP PUBLIC KEY BLOCK-----
        -----END PGP PUBLIC KEY BLOCK----

        …pertain to the public key.

        Everything else in the file is ignored. If the block heading "-----BEGIN PGP PUBLIC KEY BLOCK-----" and/or "-----END PGP PUBLIC KEY BLOCK----" are missing, then the file will be ignored as not having a valid public key.
10. Open the file that you downloaded so you learn to recognize a valid public key.
11. Import the new key from your desktop, and after selecting it, you should get a message that 2 public keys are read & imported.
12. Right-click the keys, and now you can sign both which sets the trust for the keys after you've verified it is legit. (Here, we are assuming the verification process has occurred!)
    a. There's a key created on 2010 & another on 2016. When selecting the Details tab …
        i. What are the Key Types? **1,024-bit DSA & 3,072-bit RSA**
        ii. What are their precise expiration dates? **11/2/2021 & 3/16/2020**
        iii. What level is the Owner Trust (that's you!)? **The trust for the key I created is "ultimate" and the trust for the downloaded key is just "full"**

13. Right-click on both keys, and Set Owners Trust to "ultimate"
    a. This is part of that "Web of Trust" which makes online security a thing!
    b. What level is the Owner Trust on the keys now? **You cannot change the level of trust in the newer version of this application**