Cybersecurity

Linux Personal File Encryption

CYBER.ORG

Linux Personal File Encryption Lab

- Materials needed
  - Kali Linux Virtual Machine
- Software Tool used
  - GnuPG - GNU Privacy Guard
    - GPG - GnuPG encryption tool

Objectives Covered

- Security+ Objectives (SY0-601)
  - Objective 2.1 – Explain the importance of security concepts in an enterprise environment
    - Data protection
      - Encryption

What is encryption?

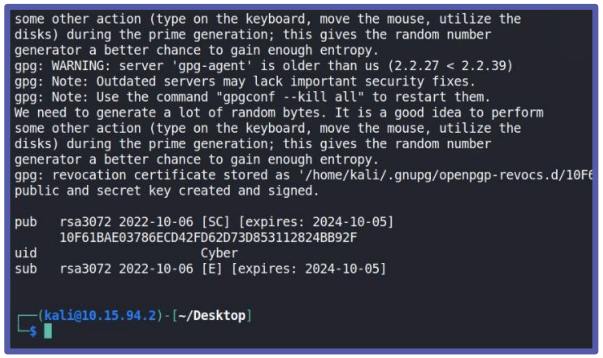
- Encryption is taking normal, plaintext, and applying some sort of mathematical algorithm on it to make it look random. This random looking string is known as ciphertext.

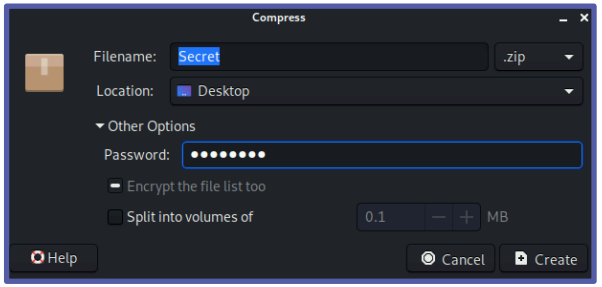
- Simple examples of encryption include a Caesar Cipher (shifting each letter a set amount in the alphabet), old newspaper Cryptoquips, and even the Enigma which the Germans used in WWII.

- The purpose is to protect data so even if someone gains access to the data, they won't be able to understand it.

Linux Personal File Encryption Lab

1. Setup VM environment
2. Create a Secret File
3. Add a Password to the File
4. Remove the Original Directory
5. Open the File
6. Your Turn - Password Protecting a File
7. Create a New File
8. Create a GPG Key
9. Encrypt the File (Using GPG)
10. Decrypt the File





CYBER.ORG

Setup Environments

- Log into your range
- Open the Kali Linux Environment
  - You should be on your Kali Linux Desktop

Create a Secret File

Explore a simple way to password protect a file

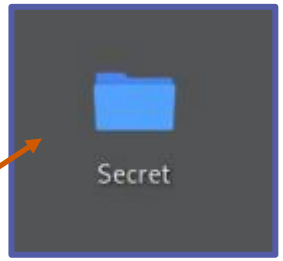
- Open a terminal
- Navigate to the desktop:

  `cd Desktop`
- Create a directory named Secret:

  `mkdir Secret`

```
┌──(kali@10.15.94.2)-[~]
└─$ cd Desktop/

┌──(kali@10.15.94.2)-[~/Desktop]
└─$ mkdir Secret
```



You should see a folder named “Secret” appear on your Desktop

Create a Secret File

Create a file inside of the directory

- Navigate into the Secret directory

  `cd Secret`

- Create a file called "Meeting“

  `touch Meeting`

- Open the file in the nano editor

  `nano Meeting`

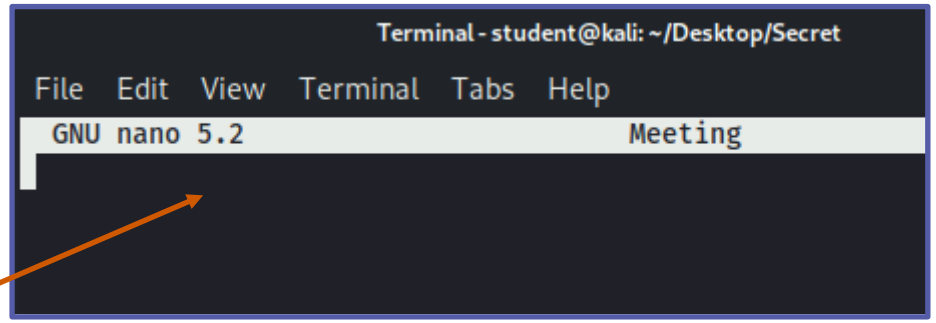
```
┌──(kali@10.15.94.2)-[~]
└─$ cd Desktop/

┌──(kali@10.15.94.2)-[~/Desktop]
└─$ mkdir Secret

┌──(kali@10.15.94.2)-[~/Desktop]
└─$ cd Secret/

┌──(kali@10.15.94.2)-[~/Desktop/Secret]
└─$ touch Meeting

┌──(kali@10.15.94.2)-[~/Desktop/Secret]
└─$ nano Meeting
```



You should see the nano editor open

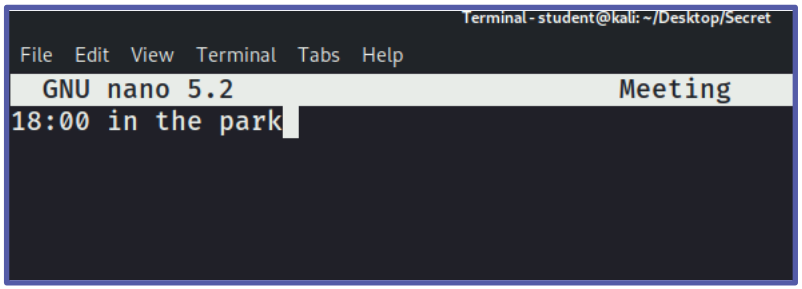
Create a Secret File

Enter some text into the Meeting document

- Enter the following text in the nano editor:

  `18:00 in the park`

- Save and exit the nano editor:
  - Press **CTRL+X**
  - Type '**y**'
  - Hit **ENTER**
- Verify the text was saved
  - **`cat Meeting`**

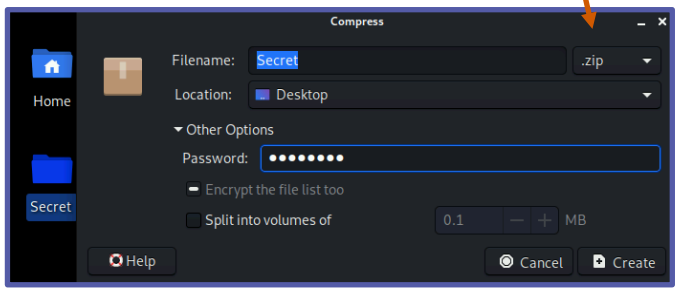


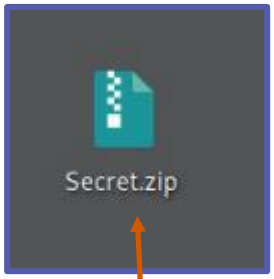
Verify the contents of Meeting were displayed in the Terminal

Add a Password to the File

- On the desktop, right click the **Secret** directory
- Click **Create Archive...**
- Change the extension to **.zip**
- Click Other **Options**
  - This will allow you to enter a Password
- Enter **password** for the password
- Click on **Create**
  - A file named Secret.zip should be on the Desktop

Change the extension to .zip





Verify a file called Secret.zip is on the Desktop

Remove the Original Directory

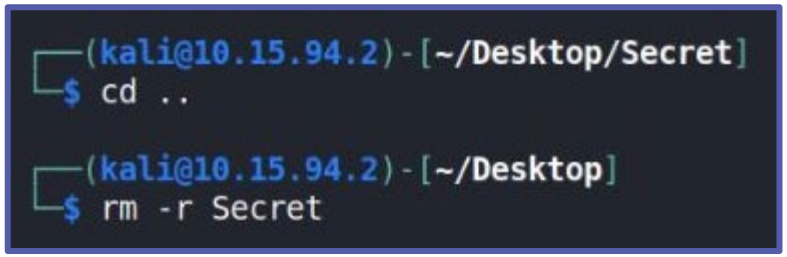
- Navigate to the Desktop

  `cd ..`

- Remove the original directory

  `rm -r Secret`

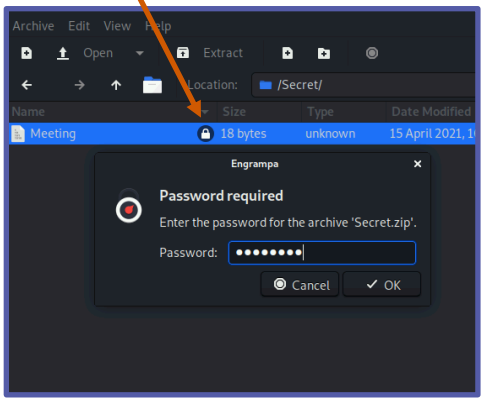
  - Verify that the original directory has been removed from the Desktop



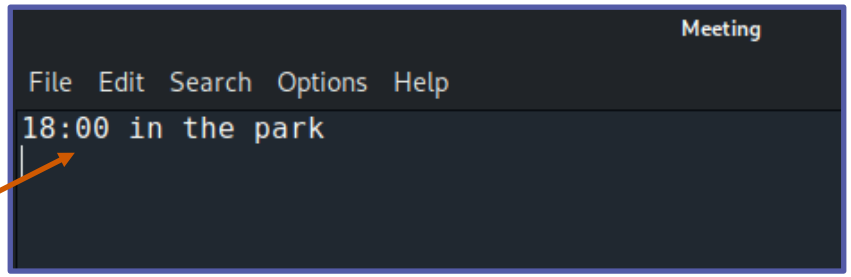
- Your Turn - Attempt to view the file inside of the zip without the password

Open the File

Notice a lock icon next to the Meeting file



- Double click the Zip
  - It should open
- Double click the Secret Directory
  - That should open
- Double click on the Meeting file
  - You should be asked to enter a password
- Enter “password” and hit ENTER
  - This should open the file



Verify the message inside of the Meeting file

Your Turn - With a Partner

- Using the same encryption method
  - Create a new file inside a folder
  - Password protect the file
  - Delete the original file
  - Have another student attempt to open the file
  - Open the file with your password to verify the file

Using GPG to Encrypt Files

- GPG can be used to encrypt files
  - More than just applying a password
- GPG - GNU Privacy Guard
  - Free-software replacement for Symantec's PGP (Pretty Good Privacy) cryptographic software suite
- GPG software will generate a key that could be shared among trusted users
  - Used to encrypt and decrypt messages

- Make sure GPG is installed and/or up to date with the following command:

```
sudo apt-get install gnupg -y
```

Create the File to Encrypt

- Make sure you are on the Desktop directory
- Create a file named Not_Secret.txt

  `touch Not_Secret.txt`
- Open the file in the nano editor

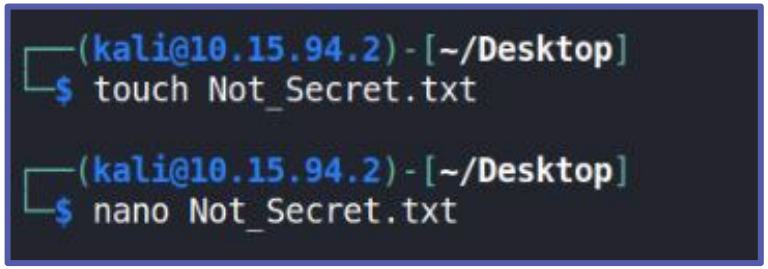
  `nano Not_Secret.txt`
- Edit the file

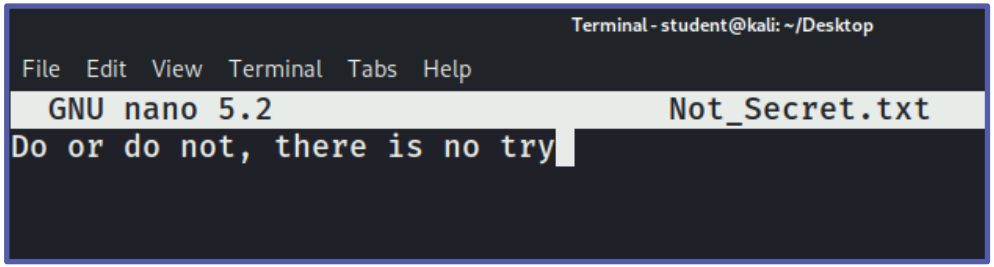
  Enter "**Do or do not, there is no try**"
- Save and Exit the nano editor

  Press **CTRL+X**

  Type '**y**'

  Hit **ENTER**

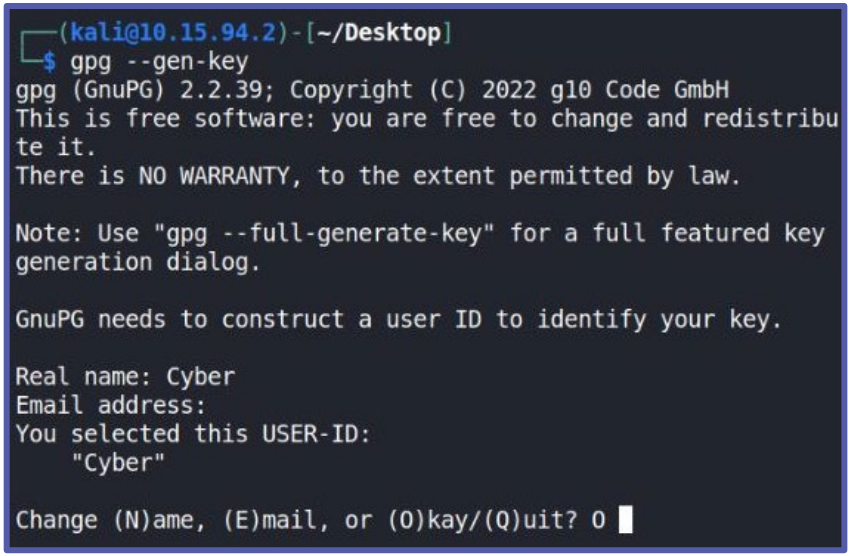


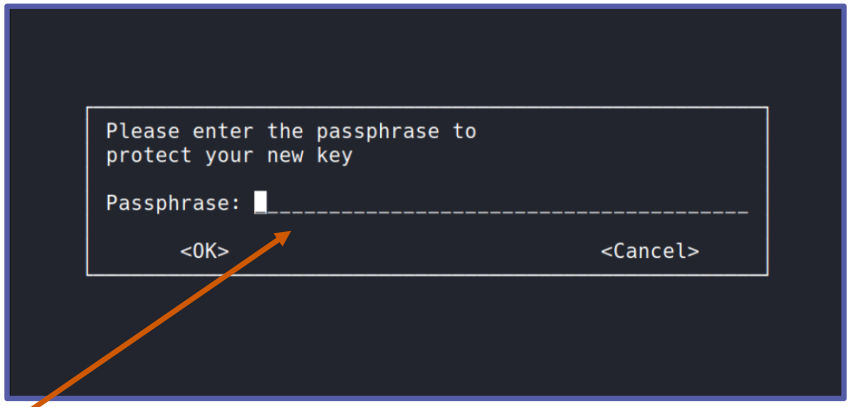


CYBER.ORG

Creating the Key

- Start to create a key
  - **`gpg --gen-key`**
- When asked for real name
  - Enter **Cyber**
- **Leave email address blank**
  - Hit **ENTER**
- Select “Okay”
  - Type **O** and hit **ENTER**





Verify this brings up the Passphrase prompt

CYBER.ORG

Creating the Key

- Use a strong password for the key
  - Type **Password1234!** for the password
  - Press **Enter**
  - Type **Password1234!** again to confirm
  - Press **Enter**

Verify the key has been created

```
disks) during the prime generation; this gives the random number
generator a better chance to gain enough entropy.
We need to generate a lot of random bytes. It is a good idea to pe
rform
some other action (type on the keyboard, move the mouse, utilize t
he
disks) during the prime generation; this gives the random number
generator a better chance to gain enough entropy.
gpg: revocation certificate stored as '/home/kali/.gnupg/openpgp-r
evocs.d/941E0B96EF323653C6059DA33993DB980F5E82A0.rev'
public and secret key created and signed.

pub   rsa3072 2024-03-14 [SC] [expires: 2026-03-14]
      941E0B96EF323653C6059DA33993DB980F5E82A0
uid                      Cyber
sub   rsa3072 2024-03-14 [E] [expires: 2026-03-14]
```

CYBER.ORG

Encrypt the File

Now that the key has been generated, encrypt the file

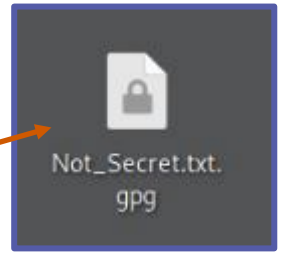
```
┌──(kali@10.15.94.2)-[~/Desktop]
└─$ gpg -e -r Cyber Not_Secret.txt
gpg: checking the trustdb
gpg: marginals needed: 3  completes needed: 1  trust model: pgp
gpg: depth: 0  valid:   3  signed:   0  trust: 0-, 0q, 0n, 0m, 0f, 3u
gpg: next trustdb check due at 2024-10-05
```

- Encrypt the file:

  `gpg -e -r Cyber Not_Secret.txt`

- Looking at the command:
  - **`-e`** is for encrypt
  - **`-r`** is to encrypt for a user's name
  - "**`Cyber`**" is the name you entered
  - **`Not_Secret.txt`** is the file to encrypt

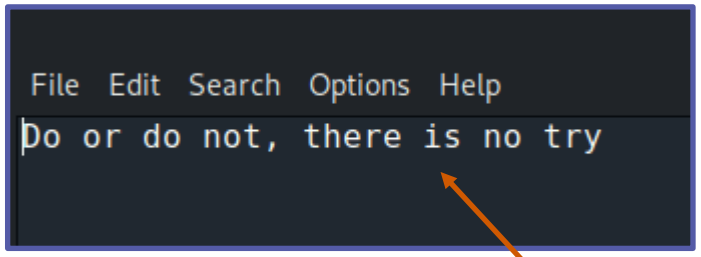
Verify the encrypted file is on the Desktop



Decrypt the Ciphertext File

- Use this command to decrypt the file:
  - **`gpg -d -o Decrypted.txt Not_Secret.txt.gpg`**
- Enter the password you created earlier
  - Type **Password1234!**

```
┌──(kali@10.15.94.2)-[~/Desktop]
└─$ gpg -d -o Decrypted.txt Not_Secret.txt.gpg
gpg: WARNING: server 'gpg-agent' is older than us (2.2.27 < 2.2.39)
gpg: Note: Outdated servers may lack important security fixes.
gpg: Note: Use the command "gpgconf --kill all" to restart them.
gpg: encrypted with 3072-bit RSA key, ID 096D7B5FABF9E50B, created 2022-10-06
      "Cyber"
```



Verify you can see the contents of the file

Your Turn - With a Partner

- Using the same encryption method (GPG)
  - Create a new file inside a folder
  - Encrypt the file
  - Delete the original file
  - Have another student attempt to open the file
  - Open the file with your password to verify the file