

Break Out The Cage

We began by running an nmap scan and discovered that ports **21 (FTP)**, **22 (SSH)**, and **80 (HTTP)** were open.

```
root@kali: /home/kali/Desktop/boxes/BreakOutTheCage
# nmap -sC -sV -iL 10.10.10.192
Starting Nmap 7.95 ( https://nmap.org ) at 2025-04-14 21:13 EDT
Nmap scan report for 10.10.19.142
Host is up (0.822s latency).
Not shown: 65532 closed tcp ports (reset)
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
80/tcp    open  http
|_ftp-syst:
|_STAT:
|_FTP server status:
|_  Connected to ::ffff:10.14.98.178
|_  Logged in as ftp
|_  TYPE: ASCII
|_  No session bandwidth limit
|_  Session timeout in seconds is 300
|_  Control connection is plain text
|_  Data connections will be plain text
|_  At session startup, client count was 4
|_  vsFTPd 3.0.3 - secure, fast, stable
|_End of status
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)
|_ncf-r-- 1 0 0 396 May 25 2020 dad_tasks
22/tcp open  ssh      OpenSSH 7.6p1 Ubuntu kubuntu0.3 (Ubuntu Linux; protocol 2.0)
|_ssh-hostkey:
|_  2848 df:f8:8b:94:f8:c8:d1:1b:51:e3:7d:f8:1d:dd:62:3e (RSA)
|_  256 3c:ba:38:63:2b:8d:1c:08:13:d5:05:b8:7a:ae:d9:30 (ECDSA)
|_  256 c8:a6:a3:64:44:1a:2c:f4:73:f5:85:f6:1f:78:4c:59:d8 (ED25519)
80/tcp open  http     Apache httpd 2.4.29 ((Ubuntu))
|_http-title: Nicholas Cage Stories
|_http-server-header: Apache/2.4.29 (Ubuntu)
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/
Nmap done: 1 IP address (1 host up) scanned in 22.02 seconds

root@kali: /home/kali/Desktop/boxes/BreakOutTheCage
```

FTP allowed **anonymous login**. Inside was a file called dad_tasks containing **base64-encoded text**. After decoding, the output was a **ciphred message**. I tried **Caesar cipher** without success, but **Vigenère cipher** worked and gave me **Weston's password**.

```
root@kali: /home/kali/Desktop/boxes/BreakOutTheCage
# echo "dG9wIEkxcl - Pr RMKP ... XZW VMUR ... TTI XEF ... LAA ZRGQRD!!!!" | base64 -d
Qpww Eekcl - Pr RMKP ... XZW VMUR ... TTI XEF ... LAA ZRGQRD!!!!
Pfw. Kzjmb xil onwoge
Paz. Tal fhfr ogessl ag oqelbx
Eljwa. Xil Dqi aiklbywe
Wzfv. Zael vvn imel sumek loudofk
Vejr. Tqonl Vow xut "arqjetpohn elnyjamu" wf.

Iz glaw A ykftcf.... Qjhsvbouuoxcxvkwatfllkughhbcbmzdwkbsiduscl

root@kali: /home/kali/Desktop/boxes/BreakOutTheCage
```



Logged in as **weston**. Found that `/usr/bin/bees` could be run as **root**, but it just sent broadcast messages — a **rabbit hole**.

```
weston@national-treasure:~/home$ sudo -l
Matching Defaults entries for weston on national-treasure:
  env_reset, mail_badpass, secure_path=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin/:/sbin:/bin:/usr/sbin:/bin

User weston may run the following commands on national-treasure:
  (root) /usr/bin/bees
weston@national-treasure:~/home$
```

I noticed **broadcast messages** appearing every few minutes. After some digging, I found that **cage** was the user sending them. I searched for files writable by the **cage** group and found `/opt/.dads_scripts/spread_the_quotes.py`, which sends random lines from `.quotes` using `wall`.

```

Broadcast message from age@national-treasure (somewhere) (Tue Apr 15 02:18:01)

Everything I take is prescription - except for the heroin. - Bad Lieutenant: Port Of Call

cat wall
#ls/nobash
bash -p
weston@national-treasure:~$ ls
bash wall
weston@national-treasure:~$ id
uid=1001(weston) gid=1001(weston) groups=1001(weston),1000(cage)
weston@national-treasure:~$ find / -group cage 2>/dev/null
/home/cage
/opt/.dads_scripts
/opt/.dads_scripts/Spread_the_quotes.py
/opt/.dads_scripts/.files
/opt/.dads_scripts/.files/quotes
weston@national-treasure:~$

```

Looking at the python script it seems it choses a random line from ./file./quates and broadcast it

The `.quotes` file was **group-writable**, and Weston was in the **cage** group. I modified the file to include a payload that creates a **SUID shell**. After a short wait, I used `/tmp/bash -p` to become **cage**.

In **cage's home directory**, I found emails. One mentioned Sean left a strange note: h4iinspsyanileph. Another hinted Sean's username might be **root**. I tried many ways to decrypt the note. Eventually, after checking a write-up, I learned it was a **Vigenère cipher** with the key "**face**" — hinted by the joke "FACE THAT!!!". Decrypting gave the **root password**, and I used it to get the **final flag**.



Search for a tool

SEARCH A TOOL ON dCODE BY KEYWORDS:
e.g. type "bookcan"

BROWSE THE FULL dCODE TOOL LIST

Results

FACE
ABCDEFGHIJKLMNOPQRSTUVWXYZ

Vigenere Cipher - dCode

Tag(s) : Poly-Alphabetic Cipher

Share

dCode and more

dCode is free and its tools are a valuable help in games, maths, geocaching, puzzles and problems to solve every day!
A suggestion ? a feedback ? a bug ? an idea ? write to dCode!

VIGENERE CIPHER

Cryptography - Poly-Alphabetic Cipher - Vigenere Cipher

VIGENERE DECODER

PLAINTEXT LANGUAGE: English

ALPHABET: ABCDEFGHIJKLMNOPQRSTUVWXYZ

DECRYPTION METHOD

☒ KNOWING THE KEY/PASSWORD: FACE

☐ KNOWING THE KEY-LENGTH/SIZE, NUMBER OF LETTERS: 3

☐ KNOWING ONLY A PARTIAL KEY: KEY

☐ KNOWING A PLAINTEXT WORD: CODE

☐ VIGENERE CRYPTANALYSIS (Kasiski's Test)

See also: [Beaufort Cipher](#) – [Caesar Cipher](#)

VIGENERE ENCODER

See also: [Beaufort Cipher](#) – [Autoclave Cipher](#) – [Caesar Cipher](#)

Summary

- Vigenere Decoder
- Vigenere Encoder
- What is the Vigenere cipher? (Definition)
- How to encrypt using Vigenere cipher?
- How to decrypt Vigenere cipher?
- How to recognise Vigenere cipher/text?
- How to break Vigenere without knowing the key?
- How to find the key when having both cipher and plaintext?
- What are the variants of the Vigenere cipher?
- How to choose the encryption key?
- What is the naming key vigenere cipher?
- What is the keyed vigenere cipher?
- What are the advantages of the Vigenere cipher versus Caesar Cipher?
- What is a Saint-Cyr slide?
- Why the name Vigenere?
- When Vigenere was invented?

Similar pages

[Beaufort Cipher](#)