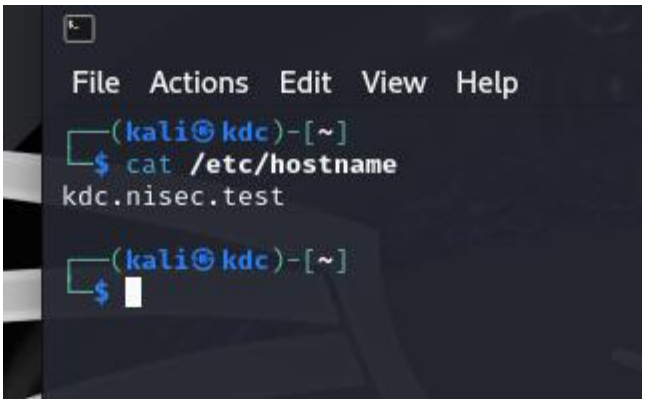
Tutorial 6

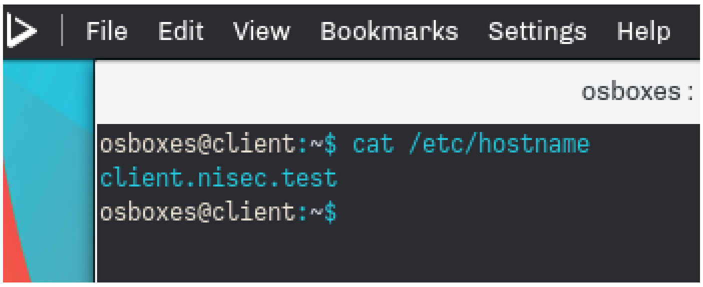
Task 1 – Setup

I use the same Kali linux box as KDC and Nitrux Box as client

Setting hostnames

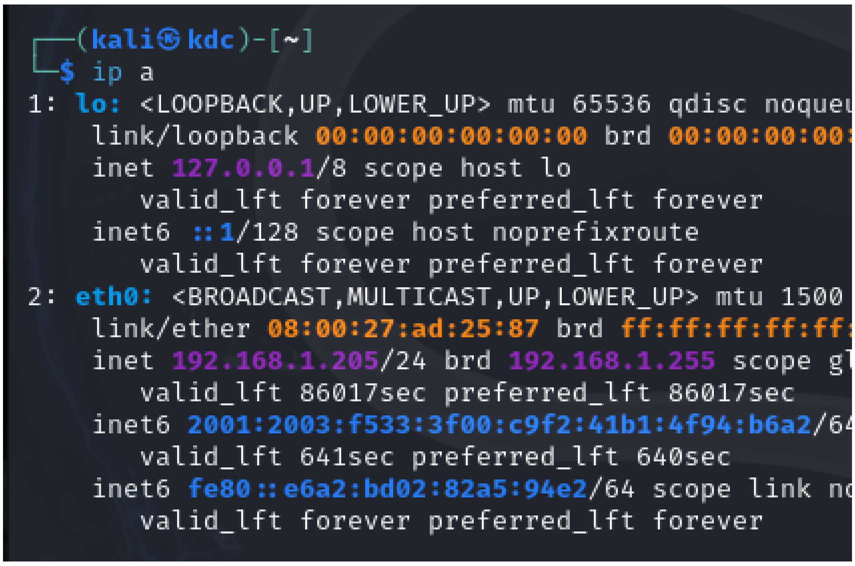


```
┌──(kali㉿kdc)-[~]
└─$ cat /etc/hostname
kdc.nisec.test

┌──(kali㉿kdc)-[~]
└─$ 
```

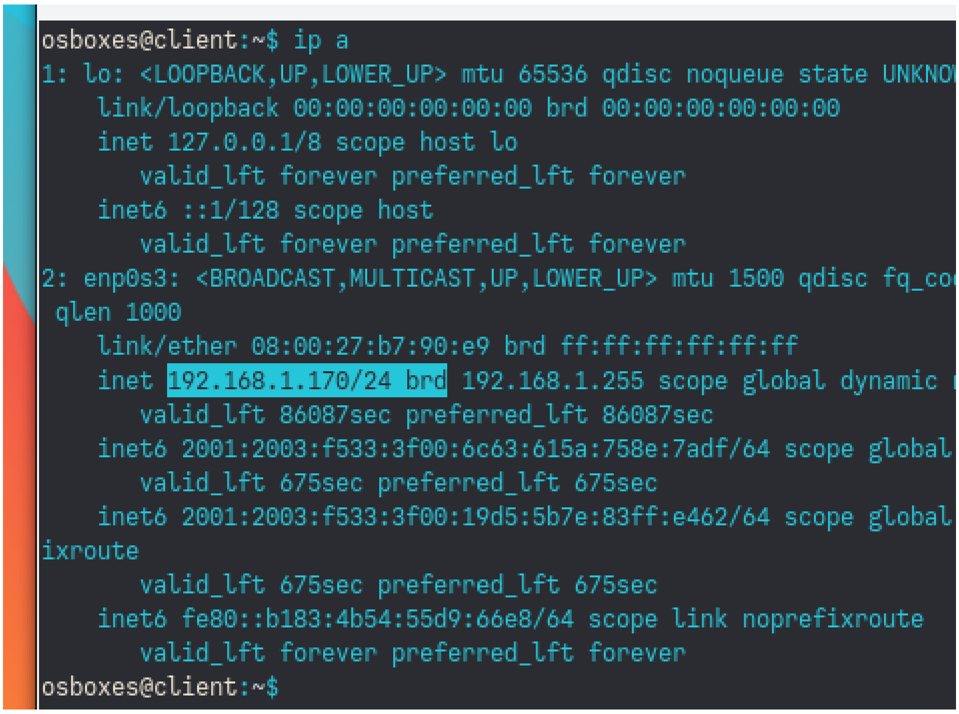


```
osboxes@client:~$ cat /etc/hostname
client.nisec.test
osboxes@client:~$
```

Getting IP addresses – I used Bridged Adapter network for both VMs



```
┌──(kali㉿kdc)-[~]
└─$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueu
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500
    link/ether 08:00:27:ad:25:87 brd ff:ff:ff:ff:ff:
    inet 192.168.1.205/24 brd 192.168.1.255 scope gl
       valid_lft 86017sec preferred_lft 86017sec
    inet6 2001:2003:f533:3f00:c9f2:41b1:4f94:b6a2/64
       valid_lft 641sec preferred_lft 640sec
    inet6 fe80::e6a2:bd02:82a5:94e2/64 scope link no
       valid_lft forever preferred_lft forever
```

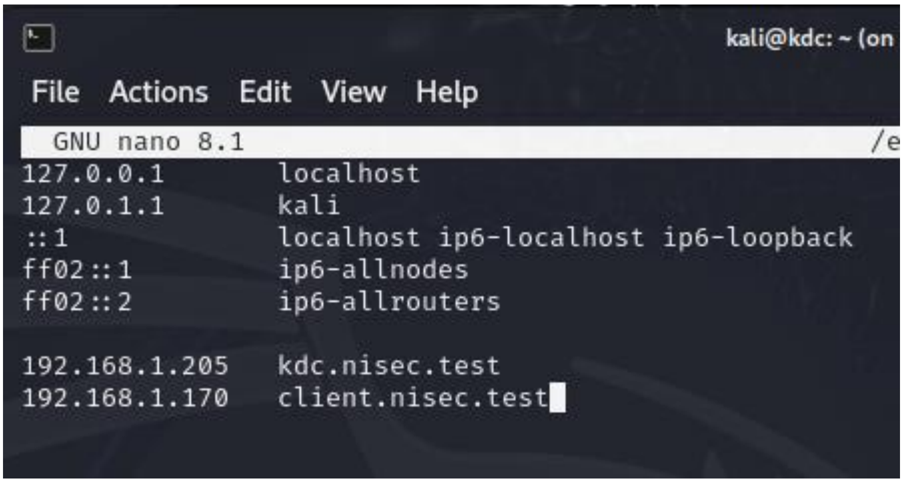


```
osboxes@client:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNO
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_co
 qlen 1000
    link/ether 08:00:27:b7:90:e9 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.170/24 brd 192.168.1.255 scope global dynamic
       valid_lft 86087sec preferred_lft 86087sec
    inet6 2001:2003:f533:3f00:6c63:615a:758e:7adf/64 scope global
       valid_lft 675sec preferred_lft 675sec
    inet6 2001:2003:f533:3f00:19d5:5b7e:83ff:e462/64 scope global
ixroute
       valid_lft 675sec preferred_lft 675sec
    inet6 fe80::b183:4b54:55d9:66e8/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
osboxes@client:~$
```

IP KDC – 192.168.1.205

IP Client – 192.168.1.170

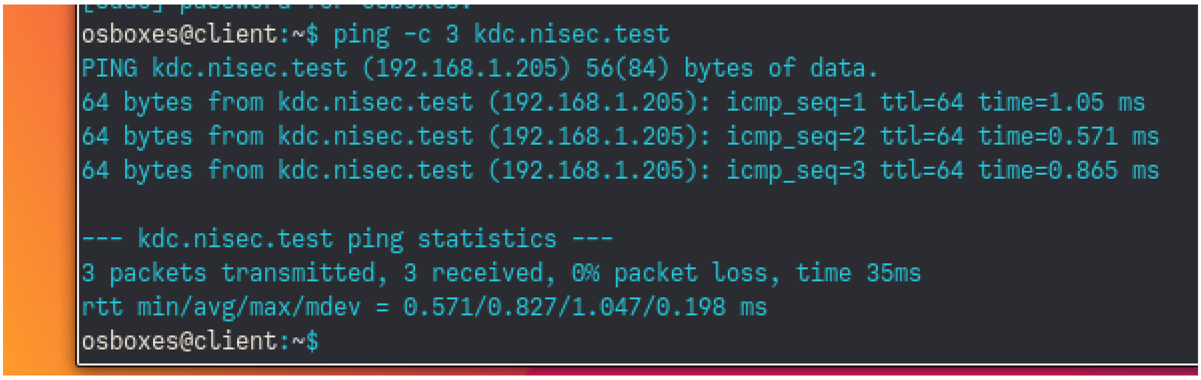
Adding hosts



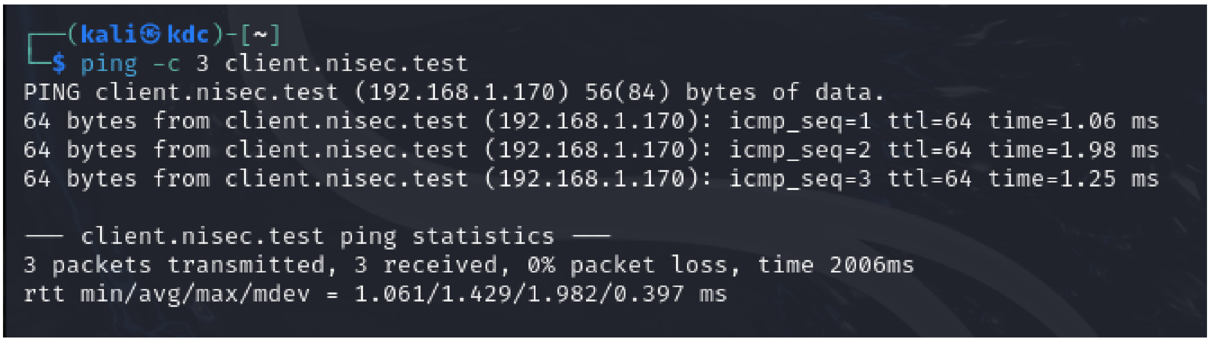
```
 GNU nano 8.1                                              /e
127.0.0.1       localhost
127.0.1.1       kali
::1             localhost ip6-localhost ip6-loopback
ff02::1         ip6-allnodes
ff02::2         ip6-allrouters

192.168.1.205   kdc.nisec.test
192.168.1.170   client.nisec.test
```

Same for nitrux vm

Ping check

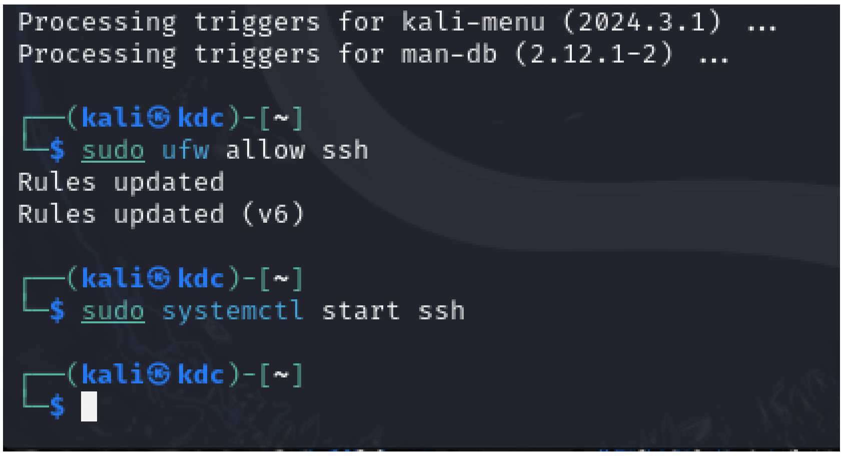


```
[sudo] password for osboxes:
osboxes@client:~$ ping -c 3 kdc.nisec.test
PING kdc.nisec.test (192.168.1.205) 56(84) bytes of data.
64 bytes from kdc.nisec.test (192.168.1.205): icmp_seq=1 ttl=64 time=1.05 ms
64 bytes from kdc.nisec.test (192.168.1.205): icmp_seq=2 ttl=64 time=0.571 ms
64 bytes from kdc.nisec.test (192.168.1.205): icmp_seq=3 ttl=64 time=0.865 ms

--- kdc.nisec.test ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 35ms
rtt min/avg/max/mdev = 0.571/0.827/1.047/0.198 ms
osboxes@client:~$
```

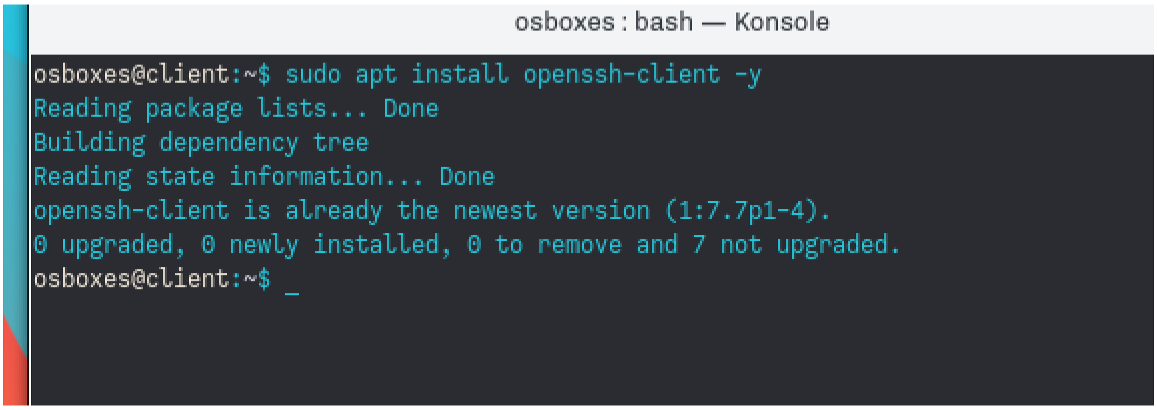


```
┌──(kali㉿kdc)-[~]
└─$ ping -c 3 client.nisec.test
PING client.nisec.test (192.168.1.170) 56(84) bytes of data.
64 bytes from client.nisec.test (192.168.1.170): icmp_seq=1 ttl=64 time=1.06 ms
64 bytes from client.nisec.test (192.168.1.170): icmp_seq=2 ttl=64 time=1.98 ms
64 bytes from client.nisec.test (192.168.1.170): icmp_seq=3 ttl=64 time=1.25 ms

—— client.nisec.test ping statistics ——
3 packets transmitted, 3 received, 0% packet loss, time 2006ms
rtt min/avg/max/mdev = 1.061/1.429/1.982/0.397 ms
```

Installing openssh and other components



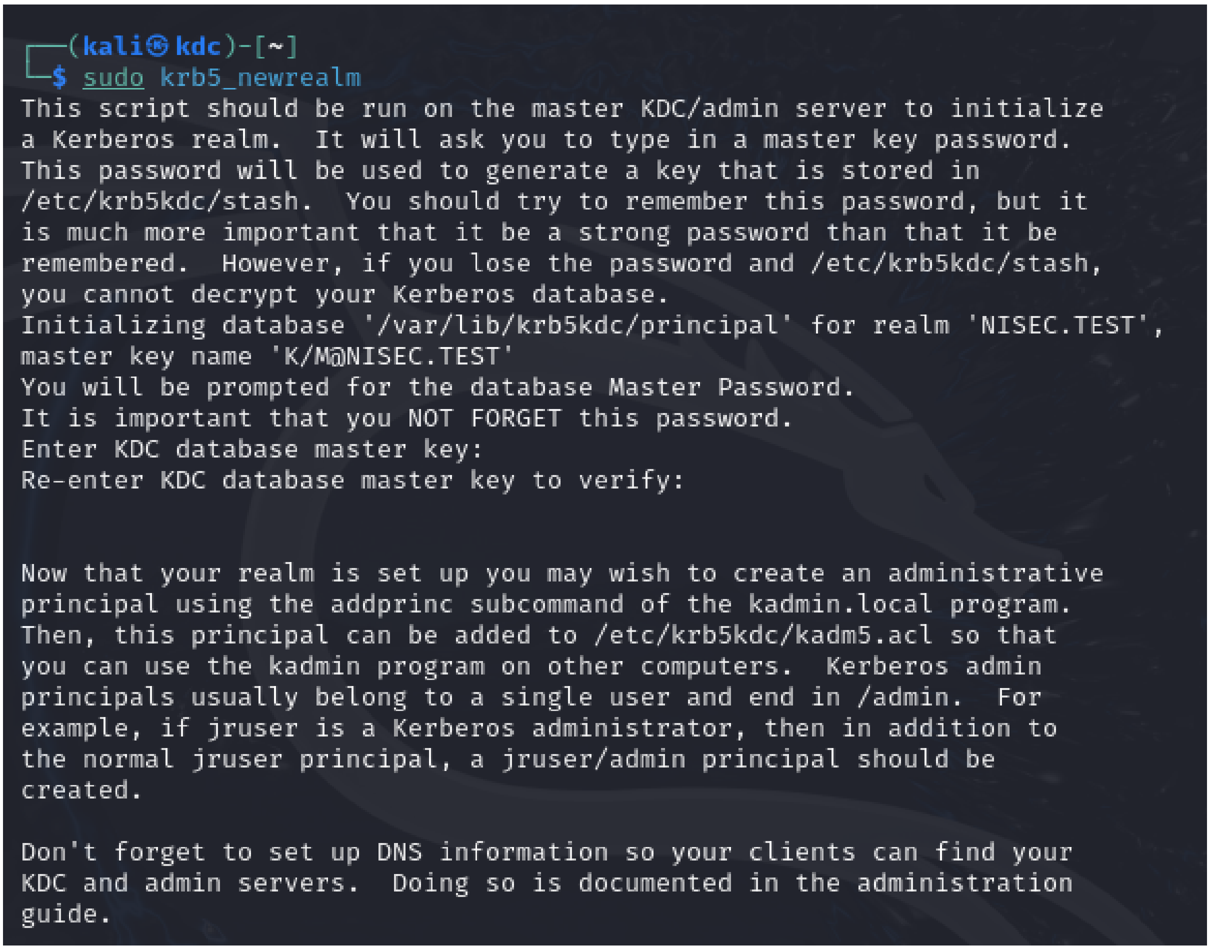
```
Processing triggers for kali-menu (2024.3.1) ...
Processing triggers for man-db (2.12.1-2) ...

┌──(kali㉿kdc)-[~]
└─$ sudo ufw allow ssh
Rules updated
Rules updated (v6)

┌──(kali㉿kdc)-[~]
└─$ sudo systemctl start ssh

┌──(kali㉿kdc)-[~]
└─$ █
```



```
osboxes@client:~$ sudo apt install openssh-client -y
Reading package lists... Done
Building dependency tree
Reading state information... Done
openssh-client is already the newest version (1:7.7p1-4).
0 upgraded, 0 newly installed, 0 to remove and 7 not upgraded.
osboxes@client:~$ _
```

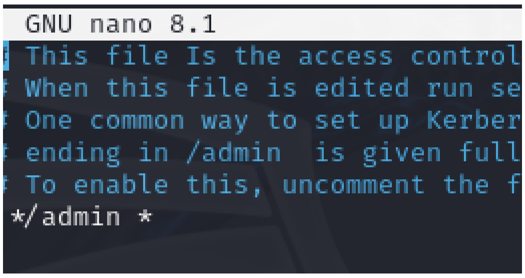
Task – 2 Install and Conf Kerberos

Everything went acc to commands provided

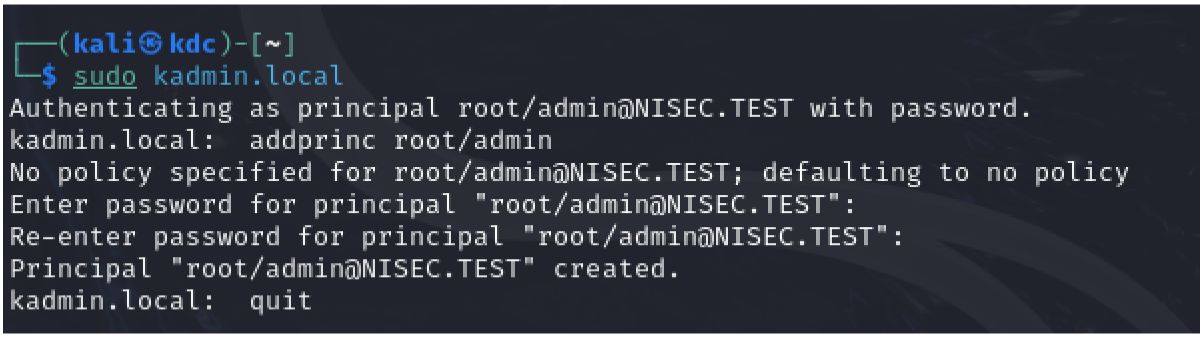


```
┌──(kali㉿kdc)-[~]
└─$ sudo krb5_newrealm
This script should be run on the master KDC/admin server to initialize
a Kerberos realm.  It will ask you to type in a master key password.
This password will be used to generate a key that is stored in
/etc/krb5kdc/stash.  You should try to remember this password, but it
is much more important that it be a strong password than that it be
remembered.  However, if you lose the password and /etc/krb5kdc/stash,
you cannot decrypt your Kerberos database.
Initializing database '/var/lib/krb5kdc/principal' for realm 'NISEC.TEST',
master key name 'K/M@NISEC.TEST'
You will be prompted for the database Master Password.
It is important that you NOT FORGET this password.
Enter KDC database master key:
Re-enter KDC database master key to verify:


Now that your realm is set up you may wish to create an administrative
principal using the addprinc subcommand of the kadmin.local program.
Then, this principal can be added to /etc/krb5kdc/kadm5.acl so that
you can use the kadmin program on other computers.  Kerberos admin
principals usually belong to a single user and end in /admin.  For
example, if jruser is a Kerberos administrator, then in addition to
the normal jruser principal, a jruser/admin principal should be
created.

Don't forget to set up DNS information so your clients can find your
KDC and admin servers.  Doing so is documented in the administration
guide.
```

Set up newreal with master password – masterpassword

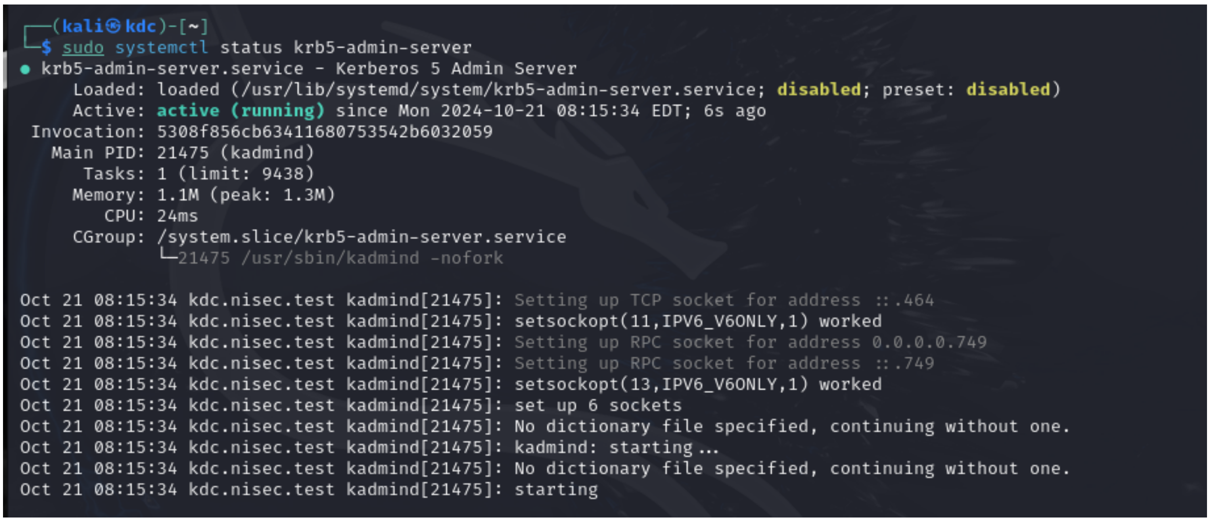


```
 GNU nano 8.1
# This file Is the access control
# When this file is edited run se
# One common way to set up Kerber
# ending in /admin  is given full
# To enable this, uncomment the f
*/admin *
```



```
┌──(kali㉿kdc)-[~]
└─$ sudo kadmin.local
Authenticating as principal root/admin@NISEC.TEST with password.
kadmin.local:  addprinc root/admin
No policy specified for root/admin@NISEC.TEST; defaulting to no policy
Enter password for principal "root/admin@NISEC.TEST":
Re-enter password for principal "root/admin@NISEC.TEST":
Principal "root/admin@NISEC.TEST" created.
kadmin.local:  quit
```

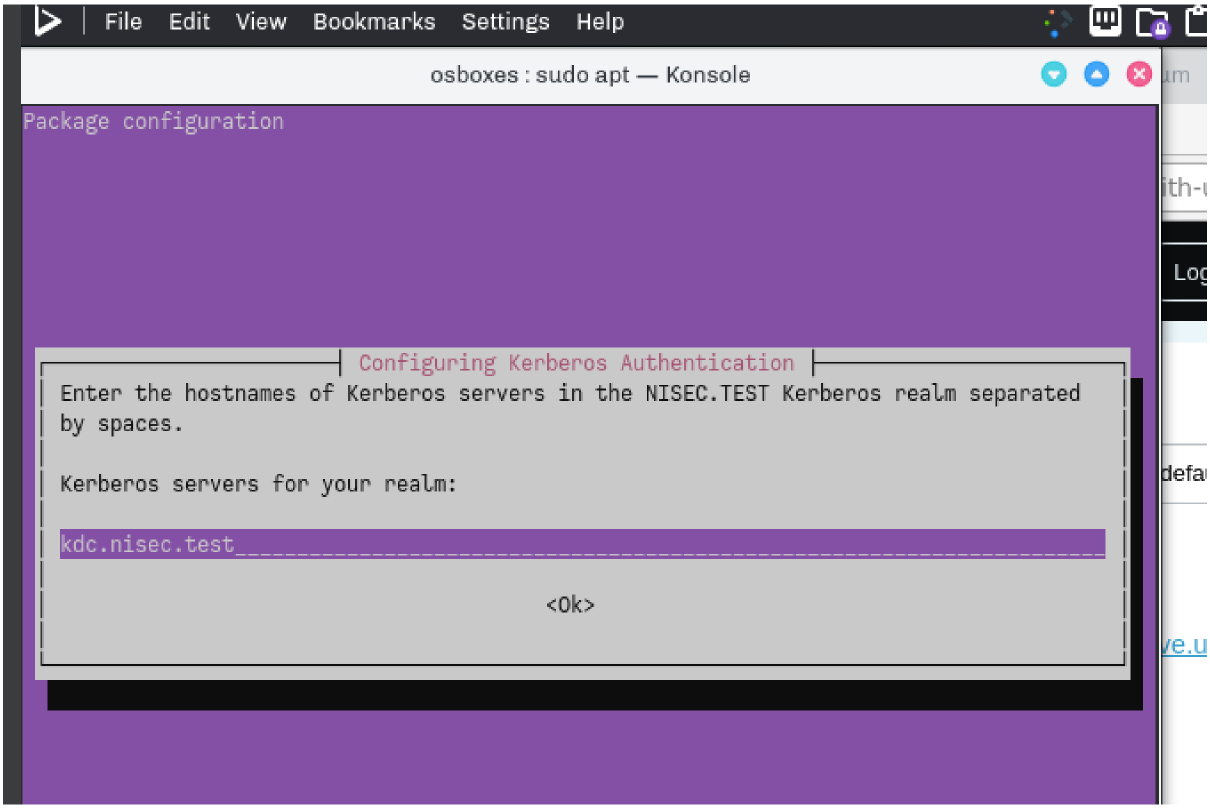
Used password as kali



```
┌──(kali㉿kdc)-[~]
└─$ sudo systemctl status krb5-admin-server
● krb5-admin-server.service - Kerberos 5 Admin Server
     Loaded: loaded (/usr/lib/systemd/system/krb5-admin-server.service; disabled; preset: disabled)
     Active: active (running) since Mon 2024-10-21 08:15:34 EDT; 6s ago
 Invocation: 5308f856cb63411680753542b6032059
   Main PID: 21475 (kadmind)
      Tasks: 1 (limit: 9438)
     Memory: 1.1M (peak: 1.3M)
        CPU: 24ms
     CGroup: /system.slice/krb5-admin-server.service
             └─21475 /usr/sbin/kadmind -nofork

Oct 21 08:15:34 kdc.nisec.test kadmind[21475]: Setting up TCP socket for address ::.464
Oct 21 08:15:34 kdc.nisec.test kadmind[21475]: setsockopt(11,IPV6_V6ONLY,1) worked
Oct 21 08:15:34 kdc.nisec.test kadmind[21475]: Setting up RPC socket for address 0.0.0.0.749
Oct 21 08:15:34 kdc.nisec.test kadmind[21475]: Setting up RPC socket for address ::.749
Oct 21 08:15:34 kdc.nisec.test kadmind[21475]: setsockopt(13,IPV6_V6ONLY,1) worked
Oct 21 08:15:34 kdc.nisec.test kadmind[21475]: set up 6 sockets
Oct 21 08:15:34 kdc.nisec.test kadmind[21475]: No dictionary file specified, continuing without one.
Oct 21 08:15:34 kdc.nisec.test kadmind[21475]: kadmind: starting ...
Oct 21 08:15:34 kdc.nisec.test kadmind[21475]: No dictionary file specified, continuing without one.
Oct 21 08:15:34 kdc.nisec.test kadmind[21475]: starting
```

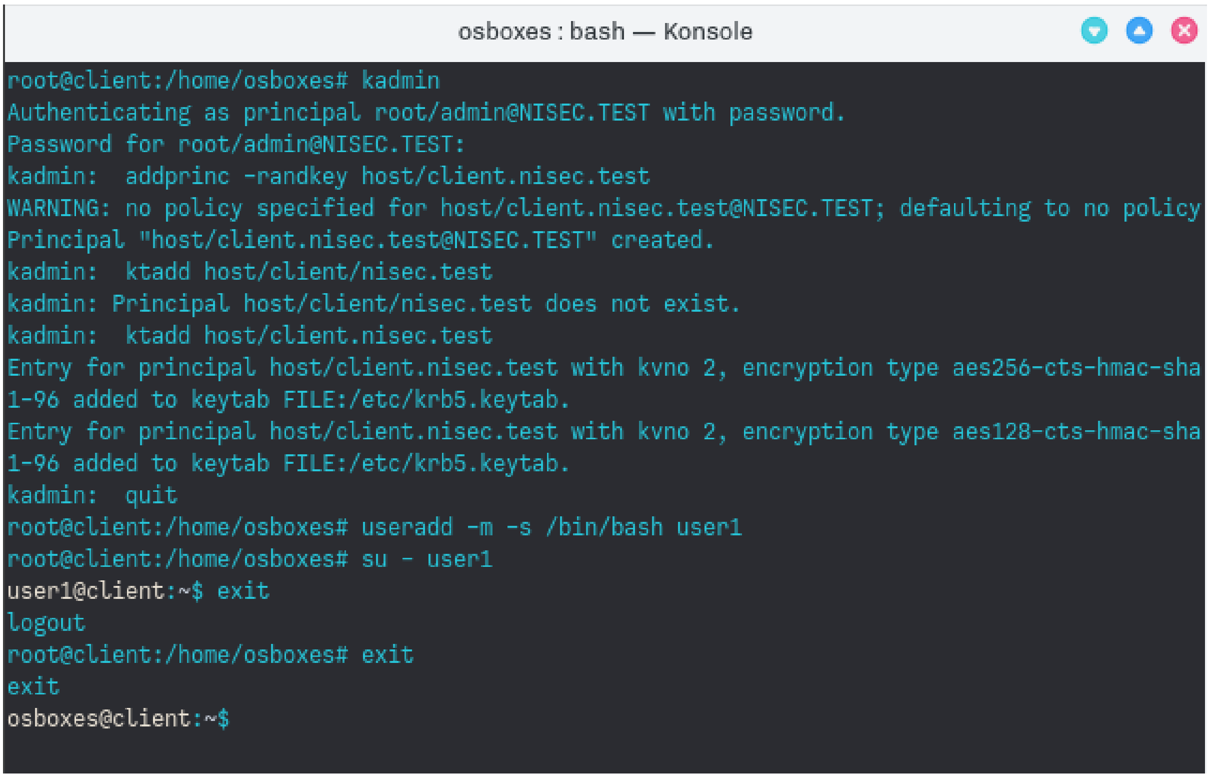
Status

Client side


File Edit View Bookmarks Settings Help
osboxes : sudo apt — Konsole
<Ok>

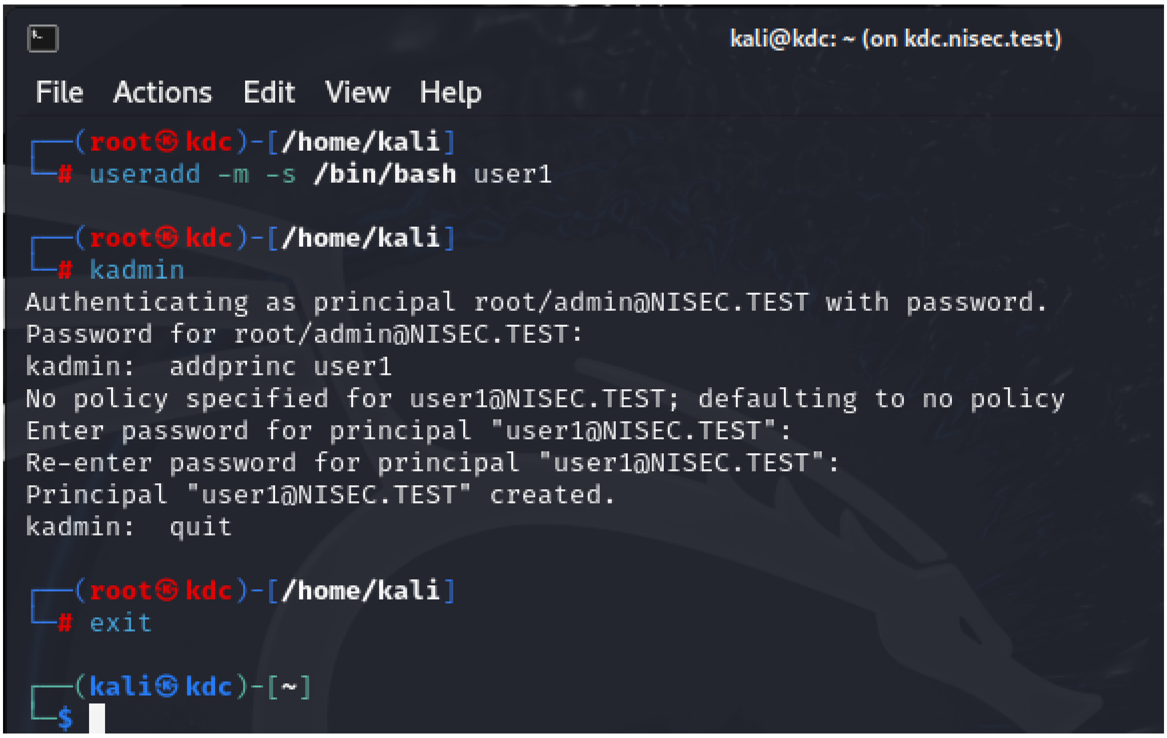
```
Package configuration

Configuring Kerberos Authentication

Enter the hostnames of Kerberos servers in the NISEC.TEST Kerberos realm separated
by spaces.

Kerberos servers for your realm:

kdc.nisec.test
```

Create key table file



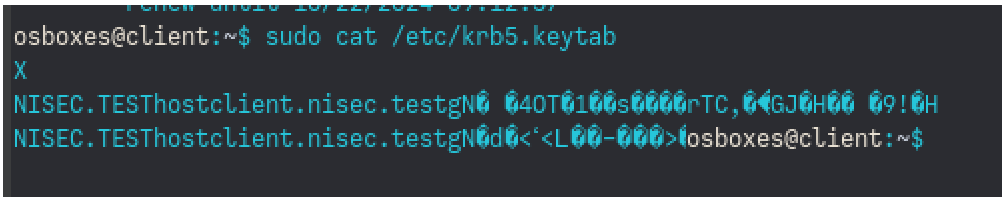
```
root@client:/home/osboxes# kadmin
Authenticating as principal root/admin@NISEC.TEST with password.
Password for root/admin@NISEC.TEST:
kadmin:  addprinc -randkey host/client.nisec.test
WARNING: no policy specified for host/client.nisec.test@NISEC.TEST; defaulting to no policy
Principal "host/client.nisec.test@NISEC.TEST" created.
kadmin:  ktadd host/client/nisec.test
kadmin: Principal host/client/nisec.test does not exist.
kadmin:  ktadd host/client.nisec.test
Entry for principal host/client.nisec.test with kvno 2, encryption type aes256-cts-hmac-sha
1-96 added to keytab FILE:/etc/krb5.keytab.
Entry for principal host/client.nisec.test with kvno 2, encryption type aes128-cts-hmac-sha
1-96 added to keytab FILE:/etc/krb5.keytab.
kadmin:  quit
root@client:/home/osboxes# useradd -m -s /bin/bash user1
root@client:/home/osboxes# su - user1
user1@client:~$ exit
logout
root@client:/home/osboxes# exit
exit
osboxes@client:~$
```

Add dummy user to kdc



```
┌──(root㉿kdc)-[/home/kali]
└─# useradd -m -s /bin/bash user1

┌──(root㉿kdc)-[/home/kali]
└─# kadmin
Authenticating as principal root/admin@NISEC.TEST with password.
Password for root/admin@NISEC.TEST:
kadmin:  addprinc user1
No policy specified for user1@NISEC.TEST; defaulting to no policy
Enter password for principal "user1@NISEC.TEST":
Re-enter password for principal "user1@NISEC.TEST":
Principal "user1@NISEC.TEST" created.
kadmin:  quit

┌──(root㉿kdc)-[/home/kali]
└─# exit

┌──(kali㉿kdc)-[~]
└─$ 
```

Get keytab file -

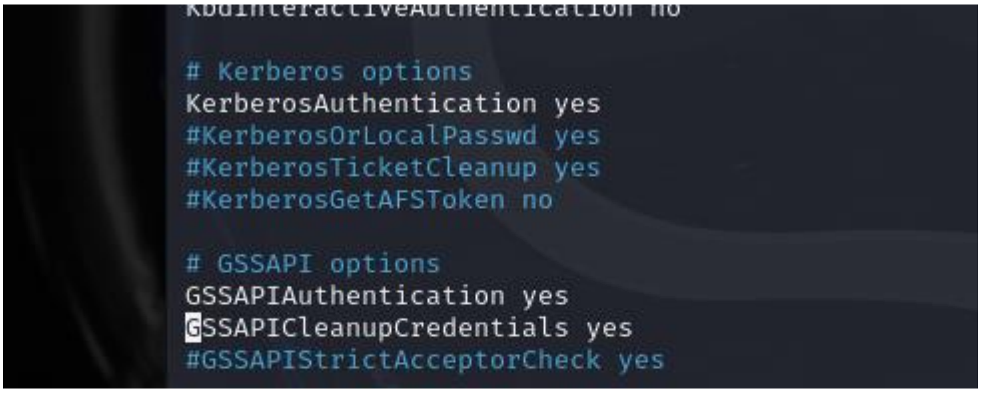


```
        renew until 10/22/2024 07:12:37
osboxes@client:~$ sudo cat /etc/krb5.keytab
X
NISEC.TESThostclient.nisec.testgN� �4OT�1��s����rTC,��GJ�H�� �9!�H
NISEC.TESThostclient.nisec.testgN�d�<‘<L��-���>�osboxes@client:~$
```

We can see its on client machine.

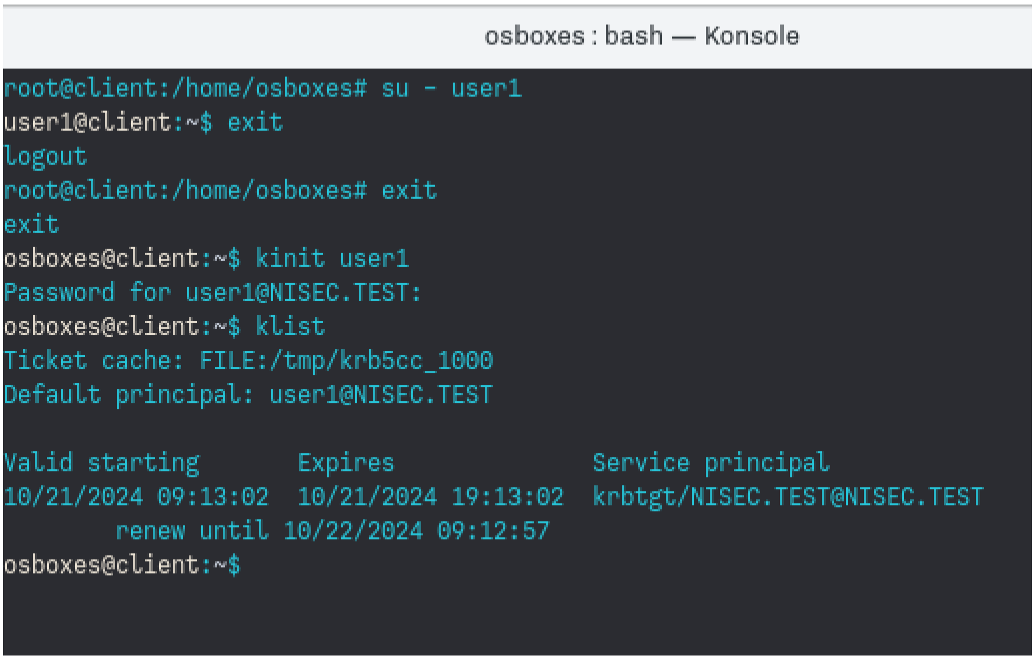
Use scp to get it to kdc machine and from there to local.

Task 3 – SSH conf



```
KbdInteractiveAuthentication no

# Kerberos options
KerberosAuthentication yes
#KerberosOrLocalPasswd yes
#KerberosTicketCleanup yes
#KerberosGetAFSToken no

# GSSAPI options
GSSAPIAuthentication yes
GSSAPICleanupCredentials yes
#GSSAPIStrictAcceptorCheck yes
```

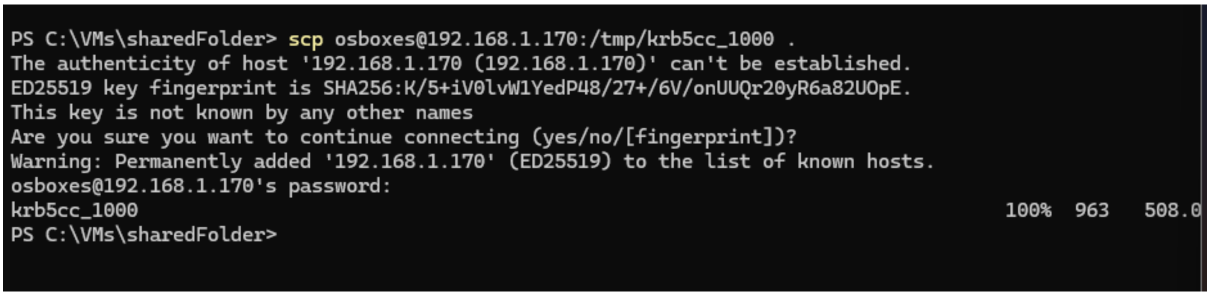
Task 4 – Generating ticket for client



osboxes : bash — Konsole

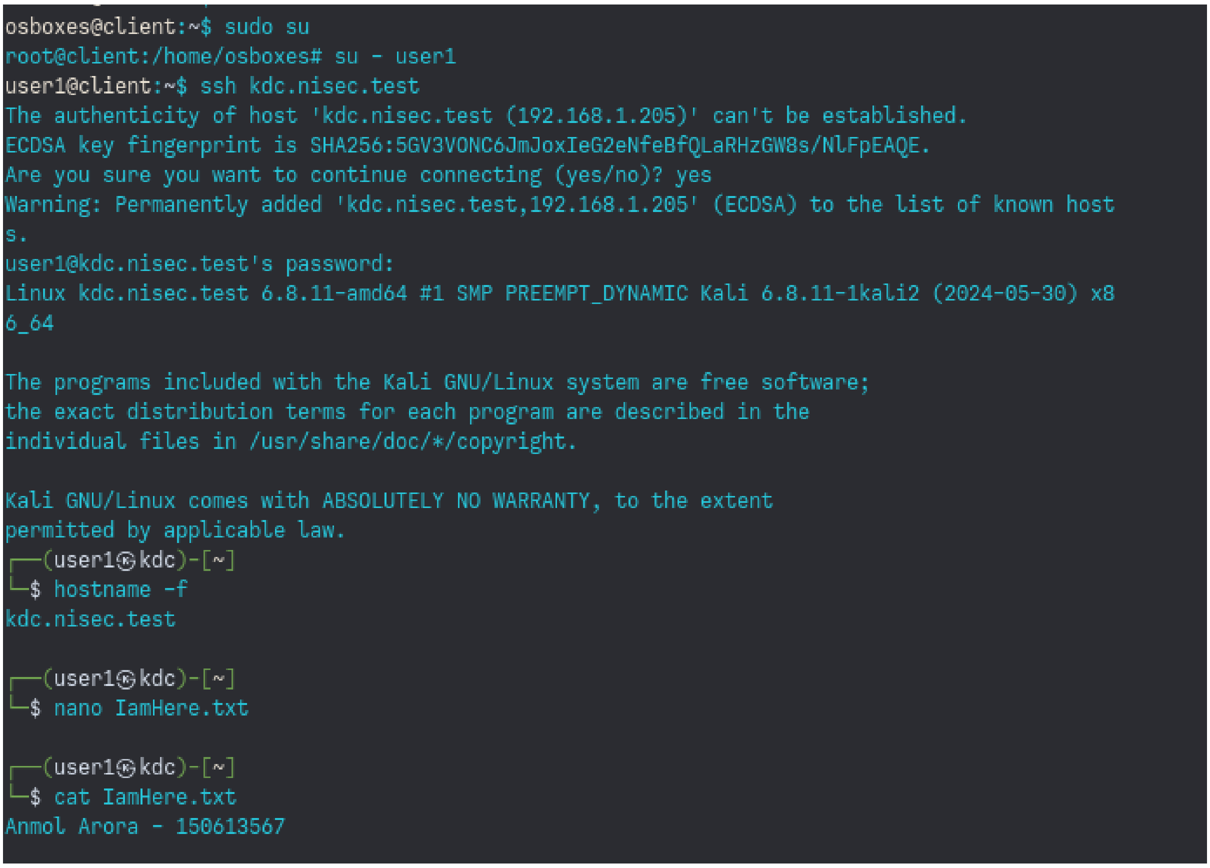
```
root@client:/home/osboxes# su - user1
user1@client:~$ exit
logout
root@client:/home/osboxes# exit
exit
osboxes@client:~$ kinit user1
Password for user1@NISEC.TEST:
osboxes@client:~$ klist
Ticket cache: FILE:/tmp/krb5cc_1000
Default principal: user1@NISEC.TEST

Valid starting       Expires              Service principal
10/21/2024 09:13:02  10/21/2024 19:13:02  krbtgt/NISEC.TEST@NISEC.TEST
        renew until 10/22/2024 09:12:57
osboxes@client:~$
```

Use scp to get file again

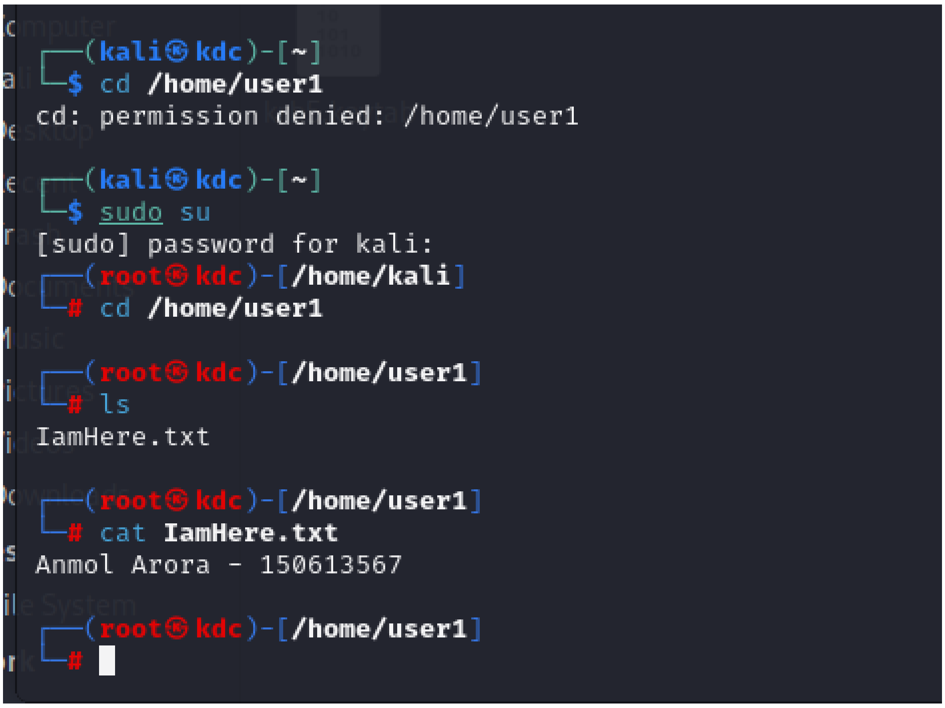


```
PS C:\VMs\sharedFolder> scp osboxes@192.168.1.170:/tmp/krb5cc_1000 .
The authenticity of host '192.168.1.170 (192.168.1.170)' can't be established.
ED25519 key fingerprint is SHA256:K/5+iV0lvW1YedP48/27+/6V/onUUQr20yR6a82UOpE.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])?
Warning: Permanently added '192.168.1.170' (ED25519) to the list of known hosts.
osboxes@192.168.1.170's password:
krb5cc_1000                                                                 100%  963    508.0
PS C:\VMs\sharedFolder>
```

Task – 5 Access and create file



```
osboxes@client:~$ sudo su
root@client:/home/osboxes# su - user1
user1@client:~$ ssh kdc.nisec.test
The authenticity of host 'kdc.nisec.test (192.168.1.205)' can't be established.
ECDSA key fingerprint is SHA256:5GV3VONC6JmJoxIeG2eNfeBfQLaRHzGW8s/NlFpEAQE.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'kdc.nisec.test,192.168.1.205' (ECDSA) to the list of known host
s.
user1@kdc.nisec.test's password:
Linux kdc.nisec.test 6.8.11-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.8.11-1kali2 (2024-05-30) x8
6_64

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

Kali GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
┌──(user1㉿kdc)-[~]
└─$ hostname -f
kdc.nisec.test

┌──(user1㉿kdc)-[~]
└─$ nano IamHere.txt

┌──(user1㉿kdc)-[~]
└─$ cat IamHere.txt
Anmol Arora - 150613567
```



```
┌──(kali㉿kdc)-[~]
└─$ cd /home/user1
cd: permission denied: /home/user1

┌──(kali㉿kdc)-[~]
└─$ sudo su
[sudo] password for kali:
┌──(root㉿kdc)-[/home/kali]
└─# cd /home/user1

┌──(root㉿kdc)-[/home/user1]
└─# ls
IamHere.txt

┌──(root㉿kdc)-[/home/user1]
└─# cat IamHere.txt
Anmol Arora - 150613567

┌──(root㉿kdc)-[/home/user1]
└─# 
```