Octobre
24 2022

Lab 4 : Mise en place de Kerberos

Cour : Cryptographie

PREPARE PAR

RHARIF ANASS

APPROUVE PAR

Ph.D Yassine Maleh

SUMARRY :

INTRODUCTION

Kerberos is a network authentication system based on the principal of a trusted third party. The other two parties being the user and the service the user wishes to authenticate to. Not all services and applications can use Kerberos, but for those that can, it brings the network environment one step closer to being Single Sign On (SSO).

This section covers installation and configuration of a Kerberos server, and some example client configurations.

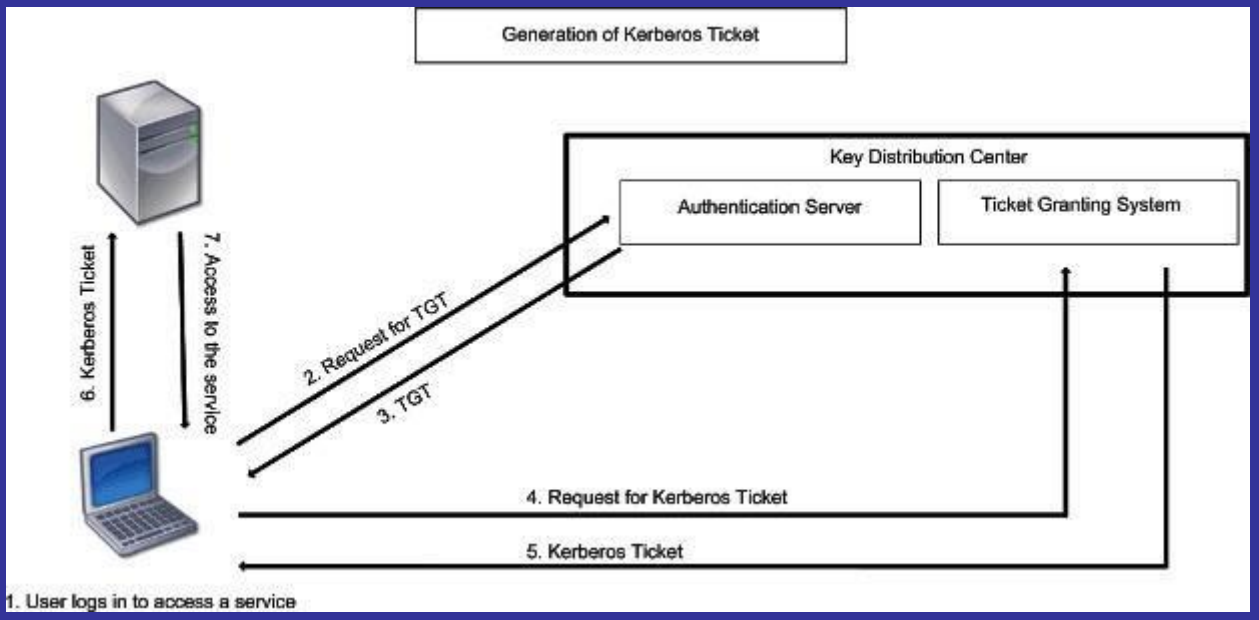
How Kerberos Authentication Works ?

When you need authentication Kerberos works as asymmetric encryption, and this is trusted by the third party, the famous Key Distribution Center (KDC). As soon as authentication happens Kerberos started to store the correct ticket for the session and user who are aware of Kerberos service they only look for authentication via password.

Here you can few steps to get Kerberos Authentication :

1. **PC client will log in to the domain, and Ticket-Granting Ticket wills sent the request for Kerberos KDC.**
2. **After this, KDC returns the session key and TCT to the PC Client.**
3. **Now is the time when ticket gets requested for the application server which the Kerberos KDC sends. This consists of TGT, PC client, and other authenticators.**
4. **After doing this, KDC returns the ticket to the PC Client.**
5. **The final ticket has sent by the application server, which must get authenticated by the PC Client.**

6. **Now is time to the reply the server through the PC Client to another authenticator. After receiving the authentication, PC Client can authenticate the server easily.**



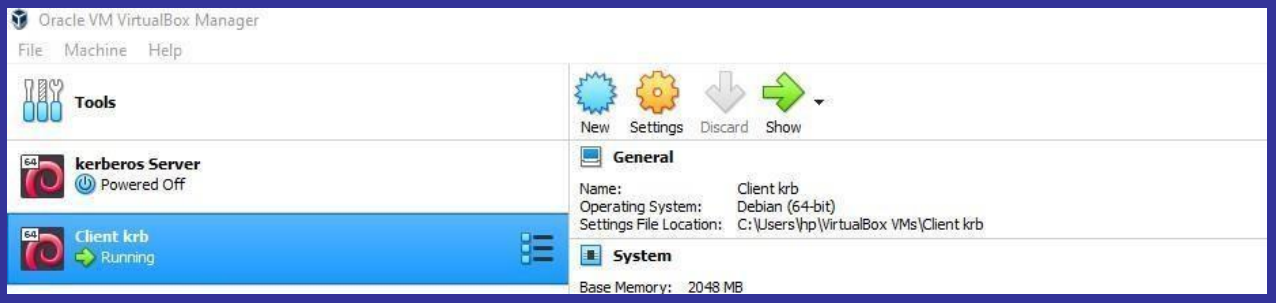
Advantages of Kerberos

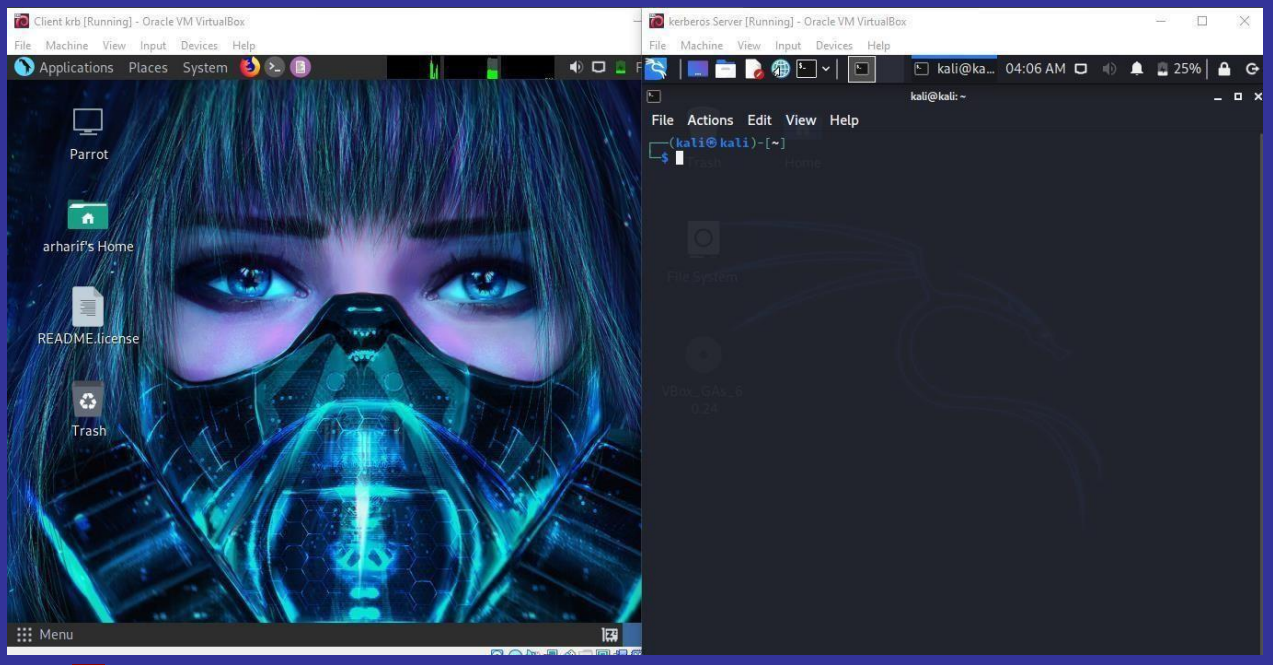
Kerberos has many advantages like other technical solutions. Those are discussing below:

1. It is so safe that passwords will never be sent to the network; only keys are allowed to send.
2. Always mutual authentication happens so that client and server get connected at the same step and communicate with the right counterpart.
3. The best advantage is authentication is always reusable, and it will never expire.
4. It completely depends on the internet standard.
5. Since Kerberos provides security vast number of the industry has adopted this, and they are happy to use its security protocol.

Install and Configure Kerberos Server

Open the terminal on your machine





The `ip` command is a Linux net-tool for system and network administrators. IP stands for Internet Protocol and as the name suggests, the tool is used for configuring network interfaces.

OBJECTS that you will use most often include:

1. link **(l) – used to display and modify network interfaces.**
2. address **(addr/a) – used to display and modify protocol addresses (IP, IPv6).**
3. route **(r) – used to display and alter the routing table.**
4. neigh **(n) – used to display and manipulate neighbor objects (ARP table).**

```
┌──(kali㉿kali)-[~]
└─$ ip r l
default via 192.168.0.1 dev eth0 proto dhcp metric 100
192.168.0.0/24 dev eth0 proto kernel scope link src 192.168.0.170 metric 100
```

```
┌──[arharif@parrot]─[~]
└─$ ip r l
default via 192.168.0.1 dev eth0 proto dhcp metric 100
192.168.0.0/24 dev eth0 proto kernel scope link src 192.168.0.112 metric 100
┌──[arharif@parrot]─[~]
```

Step 0 - Checking the Current Time Zone

A – Checking the Current Time Zone :

timedatectl is a command-line utility that allows you to view and change the system's time and date. It is available on all modern systemd-based Linux systems.

To view the current time zone, invoke the timedatectl command without any options or arguments

The system time zone is configured by symlinking the /etc/localtime file to a binary time zone's identifier in the /usr/share/zoneinfo directory.

Another way to check the time zone is to view the path the symlink points to using the ls command:

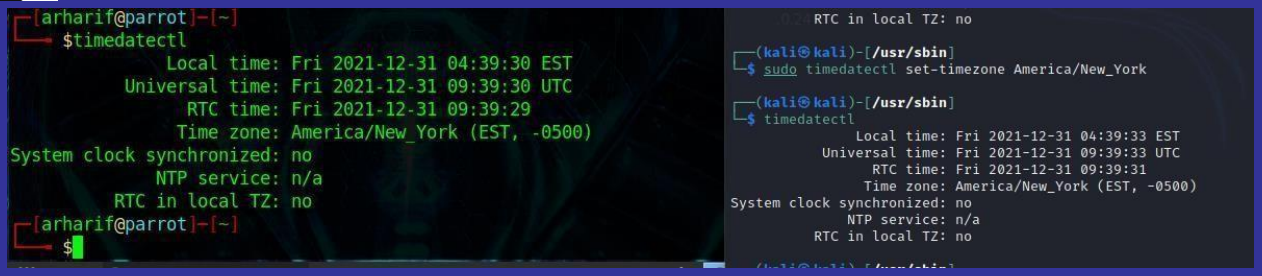
```
┌──(kali㉿kali)-[/usr/sbin]
└─$ ls -l /etc/localtime
lrwxrwxrwx 1 root root 30 Sep  8 05:21 /etc/localtime → /usr/share/zoneinfo/US/Eastern
```

B –Changing the TimeZone :

```
┌──(kali㉿kali)-[/usr/sbin]
└─$ timedatectl list-timezones
Africa/Abidjan
Africa/Accra
Africa/Algiers
Africa/Bissau
Africa/Cairo
Africa/Casablanca
Africa/Ceuta
Africa/El_Aaiun
Africa/Johannesburg
Africa/Juba
Africa/Khartoum
```

Before changing the time zone, you'll need to find out the long name of the time zone you want to use. The time zone naming convention usually uses a "Region/City" format.

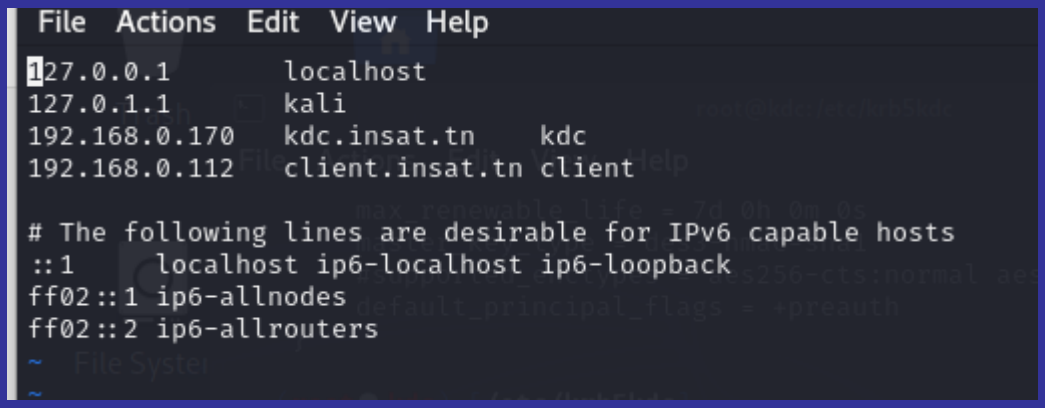
To view all available time zones, use the timedatectl command or list the files in the /usr/share/zoneinfo directory :



Step 1 - Setup FQDN in kerberos server and client.

edit the /etc/hosts file using vim editor.

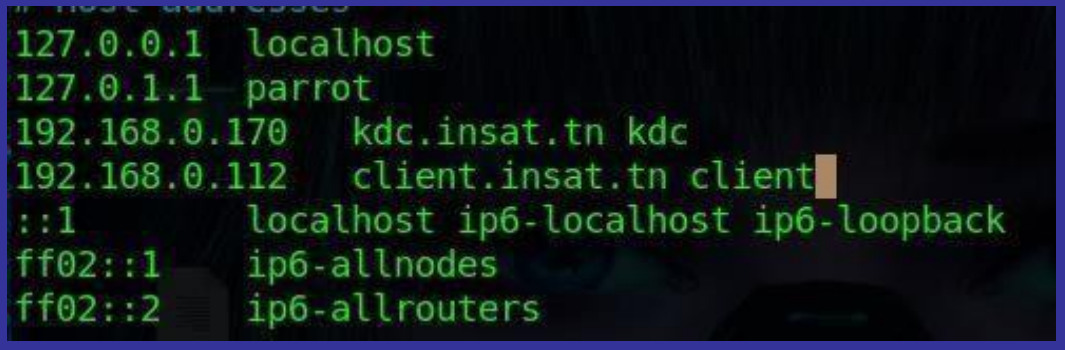
Change the IP address and FQDN with your own and paste into it.



Save and close.

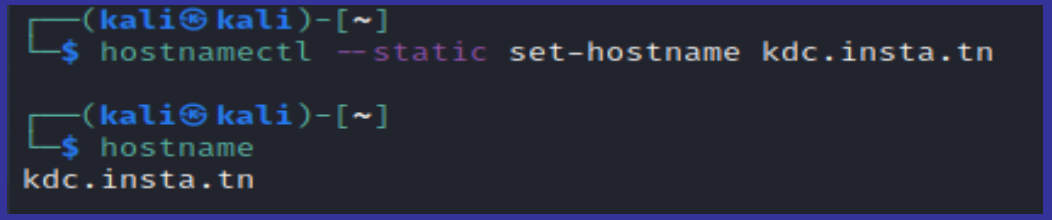
After that, edit the /etc/hosts file using vim editor. Machine client

Paste both KDC Kerberos server and the client as below.



Save and close.

Configure the FQDN on the client machine using the following command.



Configure the FQDN on the client machine using the following command. In client.

```
┌[arharif@parrot]-[~]
└─ $hostname
parrot
┌[arharif@parrot]-[~]
└─ $hostnamectl --static set-hostname client.insat.tn
┌[arharif@parrot]-[~]
└─ $hostname
client.insat.tn
┌[arharif@parrot]-[~]
```

```
┌[arharif@client]-[~]
└─ $ping kdc
PING kdc.insat.tn (192.168.0.170) 56(84) bytes of data.
64 bytes from kdc.insat.tn (192.168.0.170): icmp_seq=1 ttl=64 time=2.64 ms
64 bytes from kdc.insat.tn (192.168.0.170): icmp_seq=2 ttl=64 time=1.18 ms
64 bytes from kdc.insat.tn (192.168.0.170): icmp_seq=3 ttl=64 time=1.89 ms
^C
--- kdc.insat.tn ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2005ms
rtt min/avg/max/mdev = 1.177/1.902/2.636/0.595 ms
```

Now test using the ping command below and make sure the FQDN is resolved to the right IP address.

```
┌──(kali㉿kdc)-[~]
└─$ ping -c 2 client
PING client.insat.tn (192.168.0.112) 56(84) bytes of data.
64 bytes from client.insat.tn (192.168.0.112): icmp_seq=1 ttl=64 time=0.450
64 bytes from client.insat.tn (192.168.0.112): icmp_seq=2 ttl=64 time=0.457

--- client.insat.tn ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1018ms
rtt min/avg/max/mdev = 0.450/0.453/0.457/0.003 ms
```

The nslookup command queries internet domain name servers in two modes. Interactive mode allows you to query name servers for information about various hosts and domains, or to print a list of the hosts in a domain. In noninteractive mode, the names and requested information are printed for a specified host or domain.

```
┌[arharif@client]-[~]
└─ $nslookup kdc
Server:         192.168.0.1
Address:        192.168.0.1#53

Name:   kdc
Address: 192.168.0.170
```

```
┌──(kali㉿kdc)-[~]
└─$ nslookup client
Server:         192.168.0.1
Address:        192.168.0.1#53

Name:   client
Address: 192.168.0.112
```

```
┌──(kali㉿kdc)-[~]
└─$ nslookup 192.168.0.170
170.0.168.192.in-addr.arpa      name = kdc.

┌──(kali㉿kdc)-[~]
└─$ nslookup 192.168.0.112
112.0.168.192.in-addr.arpa      name = client.
```

```
┌[arharif@client]-[~]
└─$nslookup client
Server:         192.168.0.1
Address:        192.168.0.1#53

Name:   client
Address: 192.168.0.112

┌[arharif@client]-[~]
└─$nslookup 192.168.0.170
170.0.168.192.in-addr.arpa      name = kdc.
```

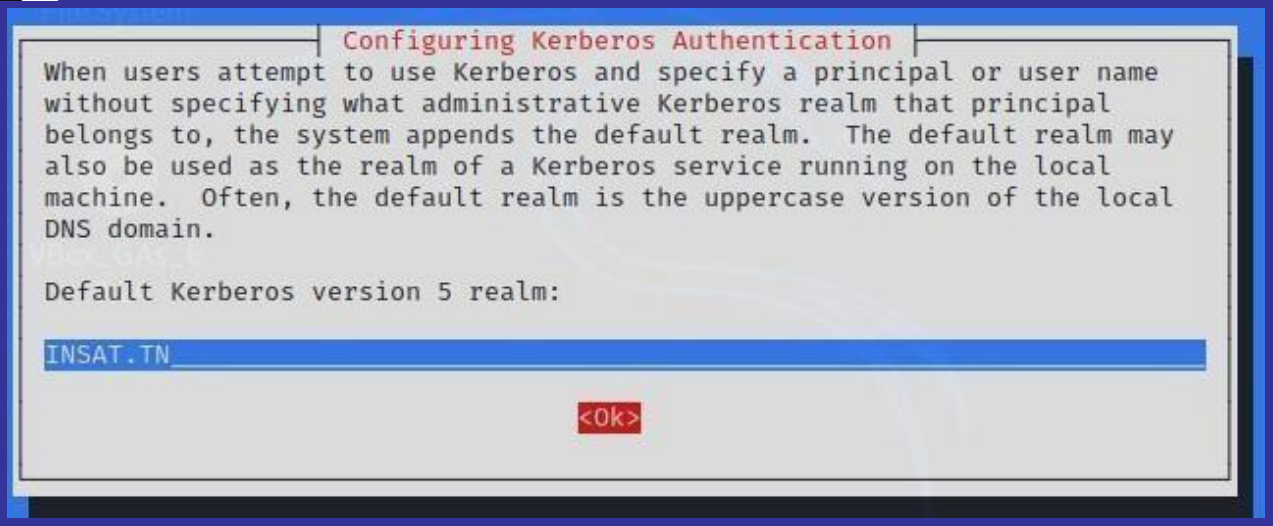
Step 2 - Install KDC Kerberos Server

Install Kerberos server using the following apt command.

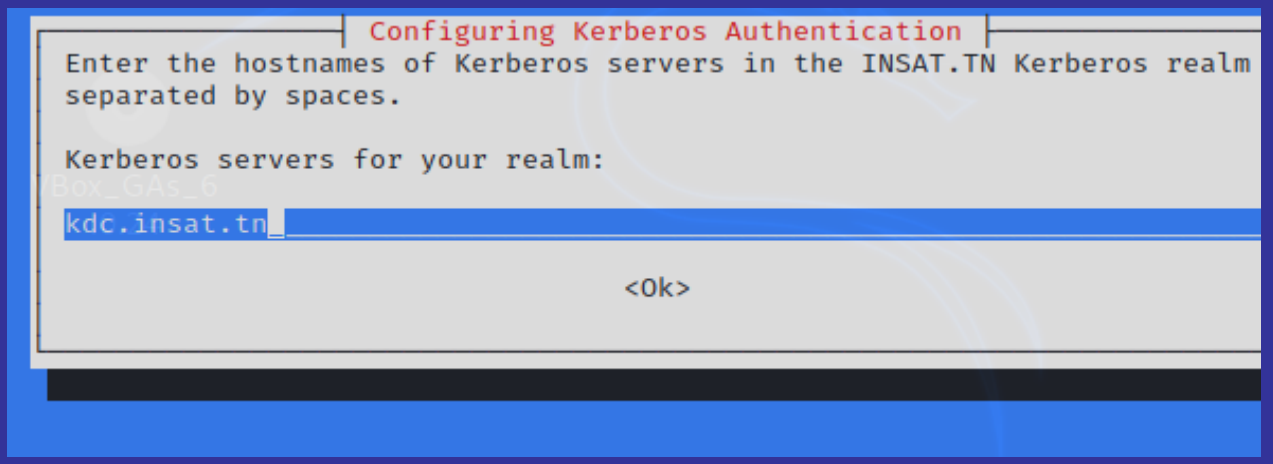
```
┌──(kali㉿kdc)-[~]
└─$ sudo apt install krb5-kdc krb5-admin-server krb5-config
```

During the installation, you will be asked about the Kerberos Realm, the Kerberos server of the Realm, and the Admin server.

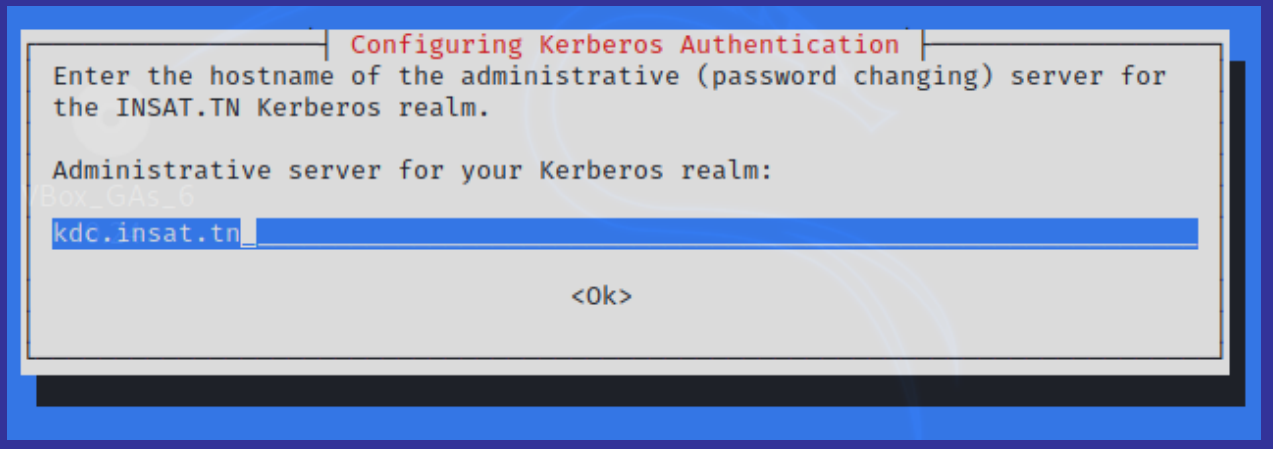
By default, the Kerberos will use the Kerberos server domain name as a REALM, 'INSAT.TN'



The Kerberos server is kdc.insat.tn.

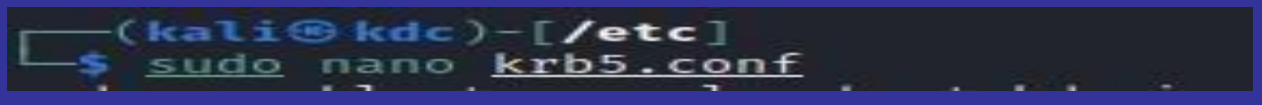


And the Admin server same as the Kerberos server kdc.insat.tn.



Once the installation is finished, you will be shown the Kerberos service is failed to run. It's fine because we will configure on the next stage.

Step 3 - Configure KDC Kerberos Server



```
realms]
        INSAT.TN = {
                kdc = kdc.insat.tn
                admin_server = kdc.insat.tn
        }
```

We haven't changed anything.

we need to add the root admin principle to the access control list by editing the /etc/krb5kdc/kadm5.acl file.

```
File  Actions  Edit  View  Help
┌──(root💀kdc)-[~]
└─# cd /etc/krb5kdc

┌──(root💀kdc)-[/etc/krb5kdc]
└─# ls
kdc.conf

┌──(root💀kdc)-[/etc/krb5kdc]
└─# nano kdc.conf
```

```
root@kdc: /etc/krb5kdc
File  Actions  Edit  View  Help
  GNU nano 5.4                                kdc.conf *
[kdcdefaults]
    kdc_ports = 750,88

[realms]
    INSAT.TN = {
        database_name = /var/lib/krb5kdc/principal
        admin_keytab = FILE:/etc/krb5kdc/kadm5.keytab
        acl_file = /etc/krb5kdc/kadm5.acl
        key_stash_file = /etc/krb5kdc/stash
        kdc_ports = 750,88
        max_life = 40d 10h 0m 0s
        max_renewable_life = 41d 0h 0m 0s
        master_key_type = des3-hmac-sha1
        #supported_enctypes = aes256-cts:normal aes128-cts:normal
        default_principal_flags = +preauth
    }
```

Now generate a new strong master password for the Kerberos REALM using the following command.

```
# 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.
Loading random data
Initializing database '/var/lib/krb5kdc/principal' for realm 'INSAT.TN
master key name 'K/M@INSAT.TN'
You will be prompted for the database Master Password.
It is important that you NOT FORGET this password.
Enter KDC database master key:

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.
```


Type your strong password and the REALM password will be generated at the /etc/krb5kdc/stash file.

```
# cd /var/lib/krb5kdc

# ls
principal  principal.kadm5  principal.kadm5.lock  principal.ok
```



```
# cd /etc/krb5kdc

# ls
kadm5.acl   kdc.conf   stash
```

Add the following configuration

```
root@kdc: /etc/krb5kdc
File  Actions  Edit  View  Help
  GNU nano 5.4                          kadm5.acl
# This file Is the access control list for krb5 administration.
# When this file is edited run service krb5-admin-server restart to activate
# One common way to set up Kerberos administration is to allow any principal
# ending in /admin  is given full administrative rights.
# To enable this, uncomment the following line:
 */admin@insat.tn *
```

Save and close the configuration, then restart the Kerberos service.

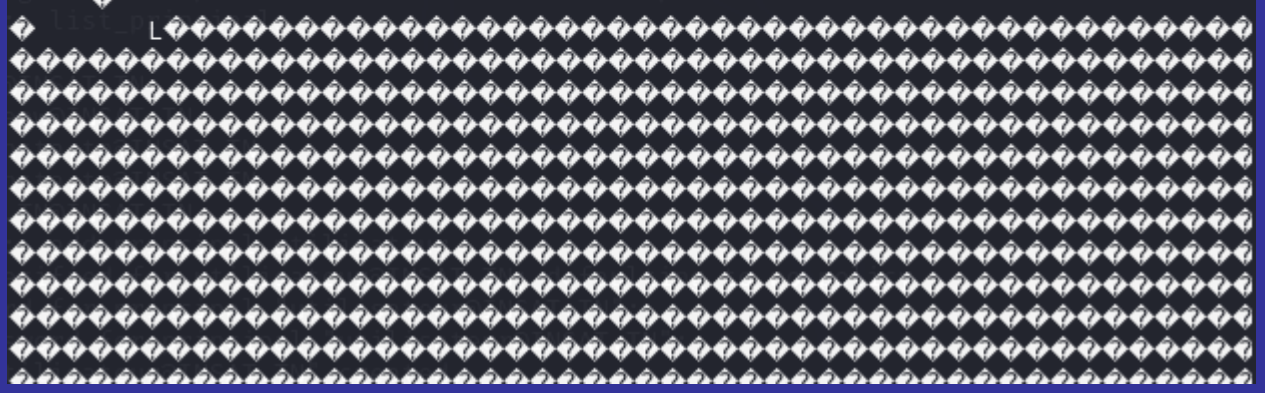
```
┌──(root💀kdc)-[/etc/krb5kdc]
└─# cat /etc/krb5.conf | grep 'INSAT' --color
        default_realm = INSAT.TN
        INSAT.TN = {

┌──(root💀kdc)-[/etc/krb5kdc]
└─# cat /etc/krb5.conf | grep 'insat' --color
                kdc = kdc.insat.tn
                admin_server = kdc.insat.tn
```

```
Authenticating as principal root/admin@INSAT.TN with password.
kadmin.local:  list_principals
K/M@INSAT.TN
kadmin/admin@INSAT.TN
kadmin/changepw@INSAT.TN
kadmin/kdc.insta.tn@INSAT.TN
kiprop/kdc.insta.tn@INSAT.TN
krbtgt/INSAT.TN@INSAT.TN
kadmin.local:  add_principal utilisateur
No policy specified for utilisateur@INSAT.TN; defaulting to no policy
Enter password for principal "utilisateur@INSAT.TN":
Re-enter password for principal "utilisateur@INSAT.TN":
Principal "utilisateur@INSAT.TN" created.
kadmin.local:  list_principals
K/M@INSAT.TN
kadmin/admin@INSAT.TN
kadmin/changepw@INSAT.TN
kadmin/kdc.insta.tn@INSAT.TN
kiprop/kdc.insta.tn@INSAT.TN
krbtgt/INSAT.TN@INSAT.TN
utilisateur@INSAT.TN
kadmin.local:
```

```
┌──(root💀kdc)-[/var/lib/krb5kdc]
└─# ls
principal  principal.kadm5  principal.kadm5.lock  principal.ok

┌──(root💀kdc)-[/var/lib/krb5kdc]
└─# cat principal
b1 "H
   
```



```
┌──(root💀kdc)-[/var/lib/krb5kdc]
└─# strings principal
utilisateur@INSAT.TN
utilisateur@INSAT.TN
aroot/admin@INSAT.TN
[`.1}
`#Nj
#<'S
sQ93
Ig1u'
```

After that, we need to create the admin user (admin principal) for the KDC Kerberos server, add the Kerberos server hostname to the database, and then create the keytab for the Kerberos server.

```
kadmin.local:  get_principal utilisateur
Principal: utilisateur@INSAT.TN
Expiration date: [never]
Last password change: Fri Dec 31 05:57:16 EST 2021
Password expiration date: [never]
Maximum ticket life: 41 days 10:00:00
Maximum renewable life: 7 days 00:00:00
Last modified: Fri Dec 31 05:57:16 EST 2021 (root/admin@INSAT.TN)
Last successful authentication: [never]
Last failed authentication: [never]
Failed password attempts: 0
Number of keys: 2
Key: vno 1, aes256-cts-hmac-sha1-96
Key: vno 1, aes128-cts-hmac-sha1-96
MKey: vno 1
Attributes: REQUIRES_PRE_AUTH
Policy: [none]
kadmin.local:
```


Create a new admin user principal called root.

```
kadmin.local:  add_principal root/admin
No policy specified for root/admin@INSAT.TN; defaulting to no policy
Enter password for principal "root/admin@INSAT.TN":
Re-enter password for principal "root/admin@INSAT.TN":
Principal "root/admin@INSAT.TN" created.
kadmin.local:
```



```
kadmin.local:  list_principals
K/M@INSAT.TN
kadmin/admin@INSAT.TN
kadmin/changepw@INSAT.TN
kadmin/kdc.insta.tn@INSAT.TN
kiprop/kdc.insta.tn@INSAT.TN
krbtgt/INSAT.TN@INSAT.TN
root/admin@INSAT.TN
utilisateur@INSAT.TN
kadmin.local:
```

Then we will manage what we call host create a ticket

```
┌──(root💀kdc)-[/etc/krb5kdc]
└─# kinit -p root/admin
Password for root/admin@INSAT.TN:

┌──(root💀kdc)-[/etc/krb5kdc]
└─# klist
Ticket cache: FILE:/tmp/krb5cc_0
Default principal: root/admin@INSAT.TN

Valid starting     Expires            Service principal
12/31/2021 06:09:25  01/01/2022 06:09:21  krbtgt/INSAT.TN@INSAT.TN
```


Commande kdestroy fo delete ticket

```
┌──(root💀kdc)-[/etc/krb5kdc]
└─# kdestroy

┌──(root💀kdc)-[/etc/krb5kdc]
└─# klist
klist: No credentials cache found (filename: /tmp/krb5cc_0)
```

Add the KDC Kerberos server to the database and create the keytab file for the KDC host.

```
No policy specified for host/kdc.insat.tn@INSAT.TN; defaulting to no policy
Enter password for principal "host/kdc.insat.tn@INSAT.TN":
Re-enter password for principal "host/kdc.insat.tn@INSAT.TN":
Principal "host/kdc.insat.tn@INSAT.TN" created.
kadmin.local:  list_principals
K/M@INSAT.TN
host/kdc.insat.tn@INSAT.TN
```

```
┌──(root💀kdc)-[/etc/krb5kdc]
└─# systemctl restart krb5-kdc

┌──(root💀kdc)-[/etc/krb5kdc]
└─# systemctl status krb5-kdc
● krb5-kdc.service - Kerberos 5 Key Distribution Center
     Loaded: loaded (/lib/systemd/system/krb5-kdc.service; disabled; vendor pr
     Active: active (running) since Fri 2021-12-31 06:14:16 EST; 7s ago
    Process: 1765 ExecStart=/usr/sbin/krb5kdc -P /var/run/krb5-kdc.pid $DAEMON
   Main PID: 1766 (krb5kdc)
      Tasks: 1 (limit: 2294)
     Memory: 832.0K
        CPU: 19ms
```

```
┌──(root💀kdc)-[/etc/krb5kdc]
└─# systemctl restart krb5-admin-server
```

After that, we need to create the admin user (admin principal) for the KDC Kerberos server, add the Kerberos server hostname to the database, and then create the keytab for the Kerberos server.

```
┌──(root💀kdc)-[/etc/krb5kdc]
└─# kadmin.local
Authenticating as principal root/admin@INSAT.TN with password.
kadmin.local:  get_principal utilisteur
get_principal: Principal does not exist while retrieving "utilisteur@INSAT.TN".
kadmin.local:  get_principal utilisateur
Principal: utilisateur@INSAT.TN
Expiration date: [never]
Last password change: Fri Dec 31 05:57:16 EST 2021
Password expiration date: [never]
Maximum ticket life: 41 days 10:00:00
Maximum renewable life: 7 days 00:00:00
Last modified: Fri Dec 31 05:57:16 EST 2021 (root/admin@INSAT.TN)
Last successful authentication: [never]
Last failed authentication: [never]
Failed password attempts: 0
Number of keys: 2
Key: vno 1, aes256-cts-hmac-sha1-96
Key: vno 1, aes128-cts-hmac-sha1-96
MKey: vno 1
Attributes: REQUIRES_PRE_AUTH
Policy: [none]
kadmin.local:  q

┌──(root💀kdc)-[/etc/krb5kdc]
└─# ktutil
ktutil:  add_entry -password -p root/admin@INSAT.TN -k 1 -e aes256-cts-hmac-sha1-96
Password for root/admin@INSAT.TN:
```


Then close the kadmin.local utility.

```
┌──(root💀kdc)-[/etc/krb5kdc]
└─# ktutil
ktutil:  add_entry -password -p root/admin@INSAT.TN -k 1 -e aes256-cts-hmac-sha1-96
Password for root/admin@INSAT.TN:
ktutil:
```

```
ktutil:   wkt /etc/krb5kdc/kadm5.keytab
ktutil:
```

```
┌──(root💀kdc)-[/etc/krb5kdc]
└─# ls
kadm5.acl  kadm5.keytab  kdc.conf  stash

┌──(root💀kdc)-[/etc/krb5kdc]
└─# file kadm5.keytab
kadm5.keytab: Kerberos Keytab file, realm=INSAT.TN, principal=root/admin, type=1, date=Fri
 Dec 31 11:24:40 2021, kvno=1
```


The klist command displays the contents of a Kerberos credentials cache or key table.

```
┌──(root💀kdc)-[/etc/krb5kdc]
└─# klist -k kadm5.keytab
Keytab name: FILE:kadm5.keytab
KVNO Principal
──── ──────────────────────────────────────────────────────────────────────────
   1 root/admin@INSAT.TN
```



```
┌──(root💀kdc)-[/etc/krb5kdc]
└─# klist -kte kadm5.keytab
Keytab name: FILE:kadm5.keytab
KVNO Timestamp           Principal
──── ─────────────────── ──────────────────────────────────────────────────────
   1 12/31/2021 06:24:40 root/admin@INSAT.TN (aes256-cts-hmac-sha1-96)
```


The ktutil command invokes a command interface from which an administrator can read, write, or edit entries in a keytab or Kerberos V4 srvtab file.

```
┌──(root💀kdc)-[/etc/krb5kdc]
└─# ktutil
ktutil:  rkt /etc/krb5kdc/kadm5.keytab
ktutil:  l
slot KVNO Principal
──── ──── ──────────────────────────────────────────────────────────────────
   1    1                    root/admin@INSAT.TN
```



```
┌──(root💀kdc)-[/etc/krb5kdc]
└─# ktutil
ktutil:  addent -password -p host/kdc.insat.tn -k 1 -e aes256-cts-hmac-sha1-96
Password for host/kdc.insat.tn@INSAT.TN:
ktutil:
ktutil:
ktutil:  wkt kadm5.keytab
ktutil:  q
```


Save and close the configuration, then restart the Kerberos service.

And the configuration of KDC Kerberos server has been completed.

```
┌──(root💀kdc)-[/etc/krb5kdc]
└─# klist -kte kadm5.keytab
Keytab name: FILE:kadm5.keytab
KVNO Timestamp           Principal
──── ─────────────────── ──────────────────────────────────────────────────────
   1 12/31/2021 06:24:40 root/admin@INSAT.TN (aes256-cts-hmac-sha1-96)
   1 12/31/2021 06:33:23 host/kdc.insat.tn@INSAT.TN (aes256-cts-hmac-sha1-96)
```

Step 4 - Install and Configure Kerberos Client

Install Kerberos client packages by running the following apt command.

```
sudo apt install -y krb5-user libpam-krb5 libpam-ccreds auth-clien
t-config
```

During the installation, you will be asked about the Kerberos Realm, the Kerberos server of the Realm, and the Admin server.

Configure Kerberos Client

From the client machine, connect to the KDC Kerberos server using the 'kadmin' command.

```
kadmin
```

And you will be asked for the password of 'root/admin' principle. Type the password and you will be logged in to the KDC Kerberos administration system. Now add the client FQDN 'client1.ahmad.io' to the Kerberos database and add the keytab file for the client.

```
addprinc -randkey host/client1.ahmad.io
ktadd host/client1.ahmad.io
```

Then close the kadmin Kerberos Administration interface.

```
quit
```

And the configuration of Kerberos client is completed.

Step 5 - Testing

For this testing purpose, we're going to configure the SSH authentication using the Kerberos. The client machine 'client.insat.tn will connect to the server kdc.insat.tn through SSH with the Kerberos authentication.

kinit is used to obtain and cache Kerberos ticket-granting tickets. This tool is similar in functionality to the kinit tool that are commonly found in other Kerberos implementations, such as SEAM and MIT Reference implementations.

The user must be registered as a principal with the Key Distribution Center (KDC) prior to running kinit.

```
┌[✗]-[arharif@client]-[/etc]
└─$kinit root/admin@INSAT.TN
Password for root/admin@INSAT.TN:
┌[arharif@client]-[/etc]
└─$klist
Ticket cache: FILE:/tmp/krb5cc_1000
Default principal: root/admin@INSAT.TN

Valid starting       Expires              Service principal
12/31/2021 08:31:20  01/01/2022 08:31:17  krbtgt/INSAT.TN@INSAT.TN
┌[arharif@client]-[/etc]
└─$kadmin
Authenticating as principal root/admin@INSAT.TN with password.
Password for root/admin@INSAT.TN:
kadmin:
```

Close the Kerberos Administration interface and edit the ssh configuration /etc/ssh/sshd_config.

```
┌──(root💀kdc)-[/etc/krb5kdc]
└─# vim /etc/ssh/sshd_config
```

Uncomment the GSSAPIAuthentication and enable it by changing the value to .

```
59 #PermitEmptyPasswords no
60
61 # Change to yes to enable challenge-response passwords (beware is
62 # some PAM modules and threads)
63 KbdInteractiveAuthentication no
64
65 # Kerberos options
66 #KerberosAuthentication no
67 #KerberosOrLocalPasswd yes
68 #KerberosTicketCleanup yes
69 #KerberosGetAFSToken no
70
71 # GSSAPI options
72 GSSAPIAuthentication yes
73 GSSAPICleanupCredentials yes
74 #GSSAPIStrictAcceptorCheck yes
75 #GSSAPIKeyExchange no
76
77 # Set this to 'yes' to enable PAM authentication, account process
```

Save and close the configuration, then restart the ssh service.

```
┌──(root💀kdc)-[/etc/krb5kdc]
└─# vim /etc/ssh/sshd_config

┌──(root💀kdc)-[/etc/krb5kdc]
└─# systemctl restart sshd

┌──(root💀kdc)-[/etc/krb5kdc]
└─# sudo systemctl status ssh
sudo: unable to resolve host kdc.insta.tn: Name or service not known
● ssh.service - OpenBSD Secure Shell server
     Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: disabled)
     Active: active (running) since Fri 2021-12-31 08:46:01 EST; 2s ago
       Docs: man:sshd(8)
             man:sshd_config(5)
    Process: 3601 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
   Main PID: 3602 (sshd)
      Tasks: 1 (limit: 2294)
     Memory: 1.1M
        CPU: 27ms
     CGroup: /system.slice/ssh.service
             └─3602 sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups

Dec 31 08:46:01 kdc.insta.tn systemd[1]: Starting OpenBSD Secure Shell server...
Dec 31 08:46:01 kdc.insta.tn sshd[3602]: Server listening on 0.0.0.0 port 22.
Dec 31 08:46:01 kdc.insta.tn sshd[3602]: Server listening on :: port 22.
Dec 31 08:46:01 kdc.insta.tn systemd[1]: Started OpenBSD Secure Shell server.
```

```
66 KerberosAuthentication yes$
67 #KerberosOrLocalPasswd yes$
68 KerberosTicketCleanup yes$
69 #KerberosGetAFSToken no$
70 $
71 # GSSAPI options$
72 GSSAPIAuthentication yes$
73 GSSAPICleanupCredentials yes$
74 #GSSAPIStrictAcceptorCheck yes$
75 #GSSAPIKeyExchange no$
76 $
```

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

# GSSAPI options
GSSAPIAuthentication yes
GSSAPICleanupCredentials yes
#GSSAPIStrictAcceptorCheck yes
#GSSAPIKeyExchange no
```

```
#ClientAliveCountMax 3
UseDNS yes
#PidFile /var/run/sshd.pi
#MaxStartups 10:30:100
#PermitTunnel no
#ChrootDirectory none
```

Create a new system use nada

```
┌[✗]-[arharif@client]-[/etc]
└──╼ $sudo useradd -m -s /bin/bash nada
[sudo] password for arharif:
┌[arharif@client]-[/etc]
```

After that, initialize the Kerberos user principal nada

Type the password of the user and after that check the available Ticket using the following command..

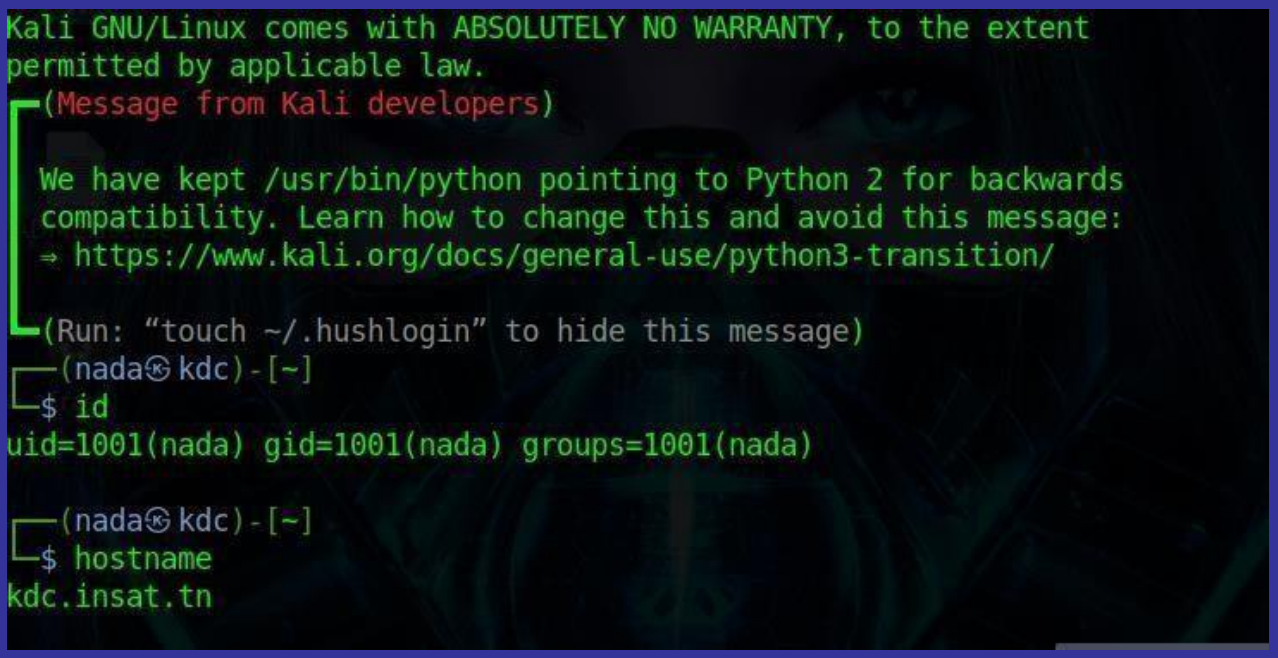
```
┌[✗]-[arharif@client]-[/etc]
└──╼ $sudo useradd -m -s /bin/bash nada
[sudo] password for arharif:
┌[arharif@client]-[/etc]
└──╼ $su - nada
Password:
┌[nada@client]-[~]
└──╼ $kinit nada
Password for nada@INSAT.TN:
┌[nada@client]-[~]
└──╼ $klist
Ticket cache: FILE:/tmp/krb5cc_1001_cT7q3N
Default principal: nada@INSAT.TN

Valid starting       Expires              Service principal
12/31/2021 08:48:12  01/01/2022 08:48:08  krbtgt/INSAT.TN@INSAT.T
N
┌[nada@client]-[~]
```

Now you can connect the krb5.ahmad.io server using the SSH Kerberos authentication.

```
┌[arharif@client]-[~]
└──╼ $su - nada
Password:
┌[nada@client]-[~]
└──╼ $ssh client.insat.tn
The authenticity of host 'client.insat.tn (192.168.0.112)' can't
be established.
ECDSA key fingerprint is SHA256:omuLTpEkCBs8KC81GfIYEZ0pgPV62Awt
+ec2RorYLs.
Are you sure you want to continue connecting (yes/no/[fingerprin
])? yes
Warning: Permanently added 'client.insat.tn,192.168.0.112' (ECDS
) to the list of known hosts.
nada@client.insat.tn's password:
Linux client.insat.tn 5.14.0-9parrot1-amd64 #1 SMP Debian 5.14.9
9parrot1 (2021-10-26) x86_64
```

And you will be connected to the krb5.ahmad.io server through SSH with Kerberos authentication.



- ✓ **Finally, the installation and configuration of Kerberos server and client on kali linux and parrot os has been completed successfully**

CONCLUSION :

Kerberos is one of the best authentication protocols, which lies at the heart of Microsoft's Active Directory. It helps enterprises to protect themselves and keep them far from attack. We hope you like this article and it will be needful for you.