Day 38

Exploitation Analyst

Control Remote Connections:

How can we control remote connections in Linux?

You can control remote connections in Linux by:

1. **SSH configuration** (/etc/ssh/sshd_config) – restrict users, change port, disable root login.
2. **Firewall rules** (ufw, iptables) – allow/block specific IPs or ports.
3. **TCP wrappers** (/etc/hosts.allow, /etc/hosts.deny).
4. **Fail2Ban** – block repeated failed login attempts.
5. **Disabling unnecessary services** (systemctl disable service).

Why we need to control remote connections?

We need to control remote connections to:

1. **Prevent unauthorized access** – stop attackers from logging in.
2. **Limit attack surface** – only allow trusted users/IPs.
3. **Protect sensitive data** – prevent data theft or modification.
4. **Mitigate brute-force attacks** – reduce risk of password guessing.
5. **Ensure system stability** – avoid unauthorized processes consuming resources.

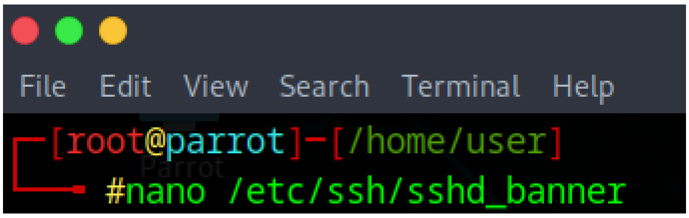
Steps to Control Remote Connections by SSH banner:

Regarding SSH banners:

- Before login: Displayed before authentication. Used for legal warnings or notices. Configured in Banner /etc/ssh/sshd_banner in /etc/ssh/sshd_config.
- After login: Displayed after authentication, usually via /etc/motd or shell profile (~/.bash_profile). Can show system info or messages.

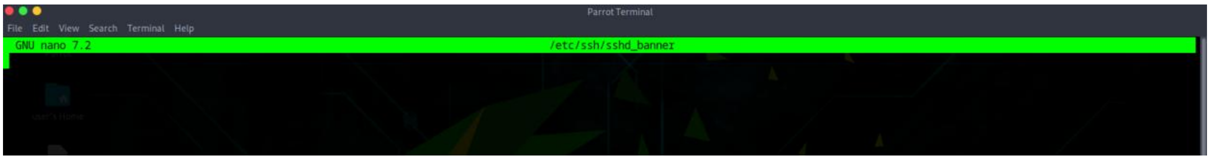
Steps:

Open the /etc/ssh/sshd_banner file using nano text editor:



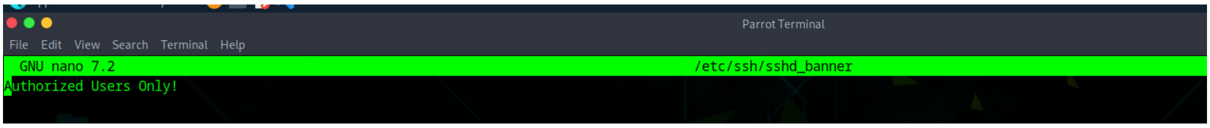
```
┌─[root@parrot]─[/home/user]
└──╼ #nano /etc/ssh/sshd_banner
```

Following screen will appear:



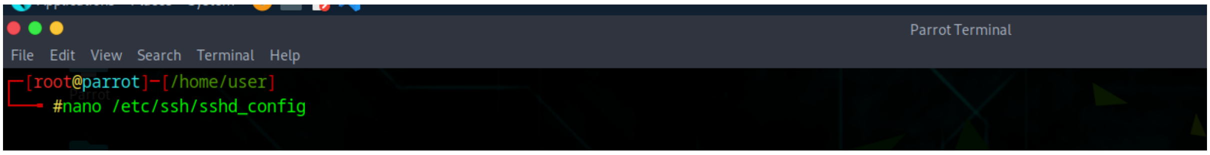
```
  GNU nano 7.2                    /etc/ssh/sshd_banner
```

Write your message:



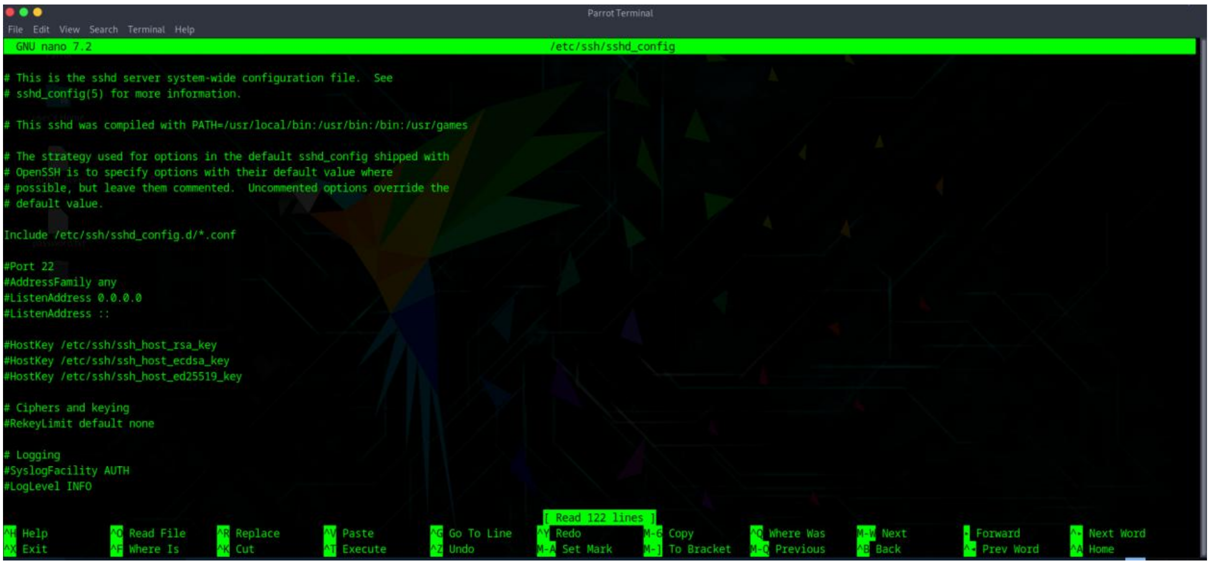
```
 GNU nano 7.2                                  /etc/ssh/sshd_banner
Authorized Users Only!
```

Now, open the SSH config from the file /etc/ssh/sshd_config:



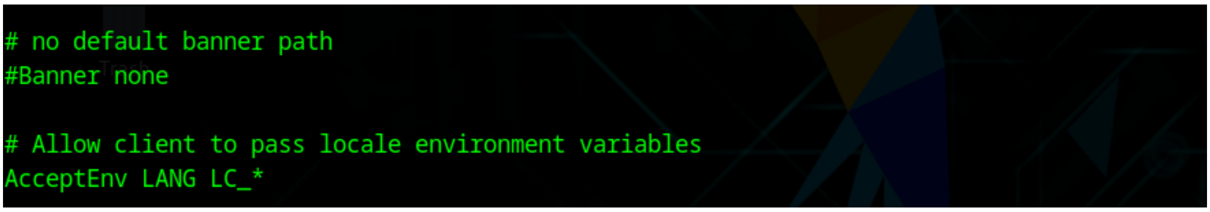
```
┌─[root@parrot]─[/home/user]
└──╼ #nano /etc/ssh/sshd_config
```

Following screen will appear:



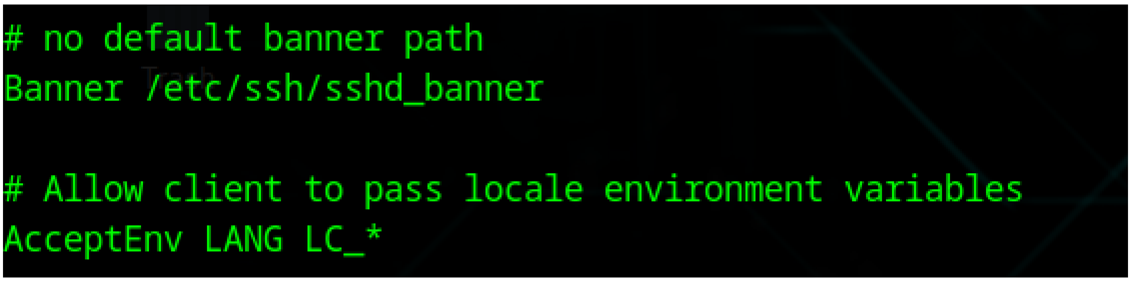
```
 GNU nano 7.2                                  /etc/ssh/sshd_config

# This is the sshd server system-wide configuration file.  See
# sshd_config(5) for more information.

# This sshd was compiled with PATH=/usr/local/bin:/usr/bin:/bin:/usr/games

# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented.  Uncommented options override the
# default value.

Include /etc/ssh/sshd_config.d/*.conf

#Port 22
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

                                         [ Read 122 lines ]
^H Help      ^O Read File  ^R Replace   ^V Paste     ^G Go To Line  ^Y Redo      M-6 Copy        ^Q Where Was  M-W Next  ▸ Forward    ^▸ Next Word
^X Exit      ^F Where Is   ^K Cut       ^T Execute   ^Z Undo        M-A Set Mark M-] To Bracket  M-Q Previous  ^B Back  ^◂ Prev Word ^A Home
```

Press Ctrl + F and search for “banner”: following section should be found commented.



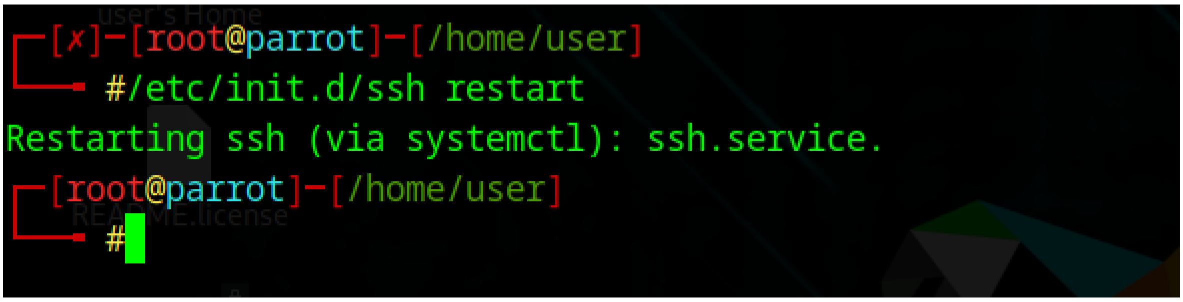
```
# no default banner path
#Banner none

# Allow client to pass locale environment variables
AcceptEnv LANG LC_*
```

Edit it, remove that # symbol, and also add the path of /etc/ssh/sshd_banner over there:



```
# no default banner path
Banner /etc/ssh/sshd_banner

# Allow client to pass locale environment variables
AcceptEnv LANG LC_*
```

Save it, and then restart the SSH: using the command /etc/init.d/ssh restart



```
┌─[✗]─[root@parrot]─[/home/user]
└──╼ #/etc/init.d/ssh restart
Restarting ssh (via systemctl): ssh.service.
┌─[root@parrot]─[/home/user]
└──╼ #
```

You should see the banner before login.

--The End--