***Remote Access for Kali Linux Powered Machines***

**Abstract:**

Generally, when starting in cyber security the first learning point in the "practical knowledge" zone is undoubtedly Kali Linux. For newcomers in this field it is very important to get their hands dirty with Kali and in-depth understanding of this Linux distribution can build a strong foundation for the future. There are various forms of the distro available that range right from ARM based CPU's to virtual machines to full install options. This self-project can help enhance the user experience and can prove to be very helpful for headless connections.

**Possible Use Cases:**

1. Specifically, ARM based Kali Linux is the most portable and cheapest option. Small card sized computers like the Raspberry Pi running Kali Linux have now become a very vital tool in the cyberspace and get the job done really quick. But to have a headless connection to the Raspberry Pi can save the user from the hassle of wires running all over and provide a clean and tidy solution. Portable hacking machines built using the Raspberry Pi can be connected to a user's laptop or desktop over the network through a headless connection through this script.
2. In Cyber or to be specific in a “Cyber Range” (virtual environment that companies can use for cyber warfare training and software development) when security professionals use many monitors and many laptops for work, this script can prove to be very useful for using many computers on one single computer.

**Implementation:**

🡪Asks the user for permission to install the open-ssh server

🡪If user says Y:

Installs the open-ssh server

If the user says N:

Terminates the program and exits

🡪Creates a new directory for backing up the default ssh keys

🡪Backs up the default ssh keys into the new directory for safety purposes

🡪Regenerates the ssh keys to continue using the ssh server

🡪Enables ssh service to start at every instance of bootup

🡪Enables ssh services for that instance of bootup

🡪Displays the status of ssh service for the user

🡪Displays the sshd\_config file to the user

🡪Instructs the user about the next steps to take

🡪Program terminates

🡪User then configures the sshd\_config file

**NOTE: To preserve the integrity of the system it is very important to configure the file manually.**

**Screenshots:**

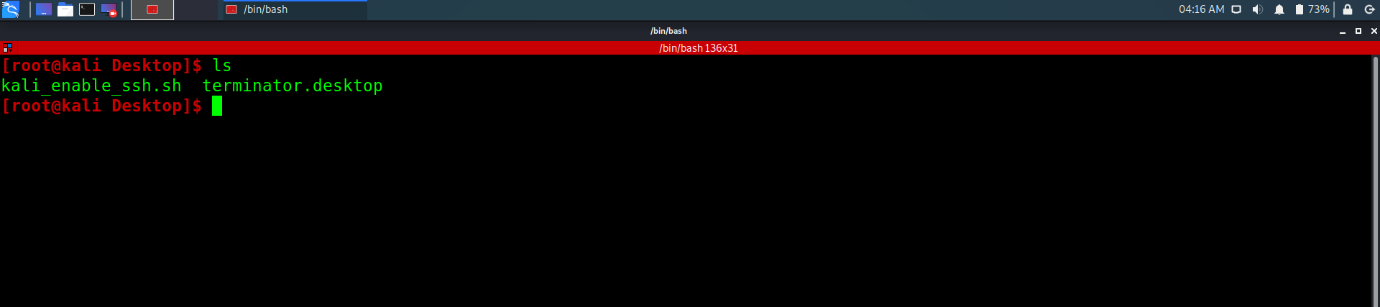
****

Fig. 1 Shows file stored on desktop

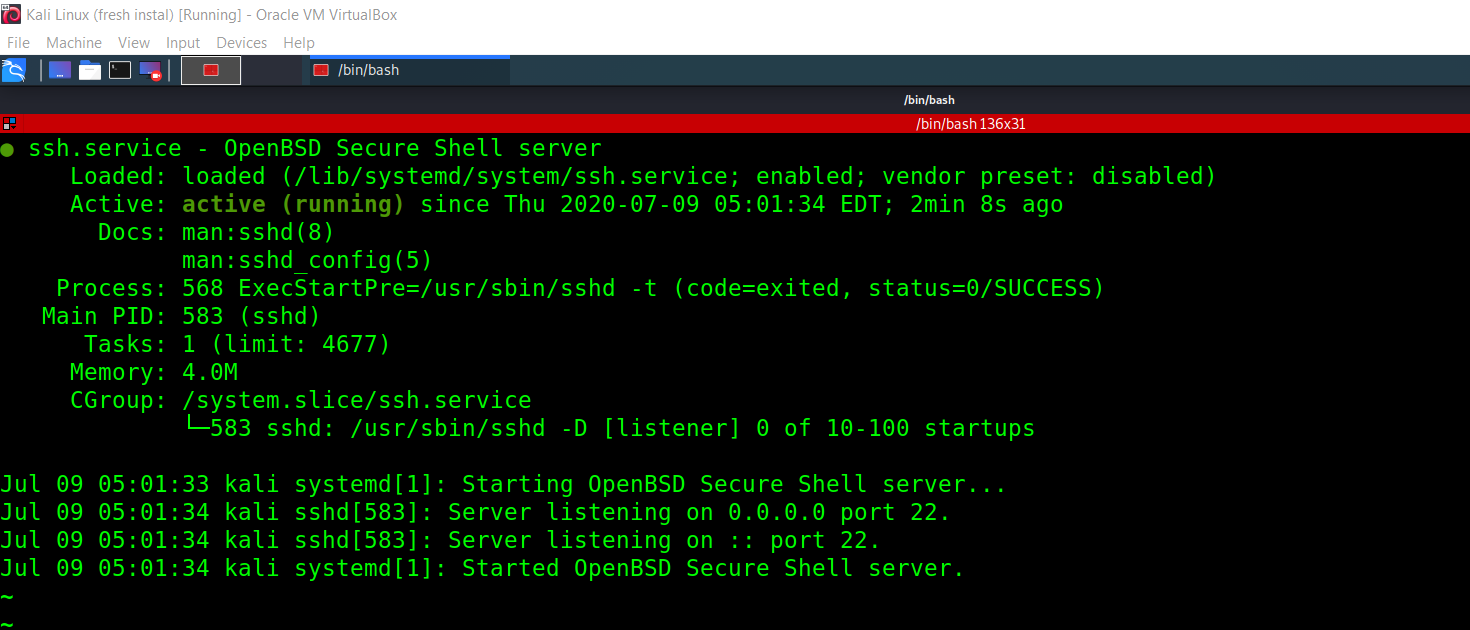


Fig.2 Shows status of the ssh server

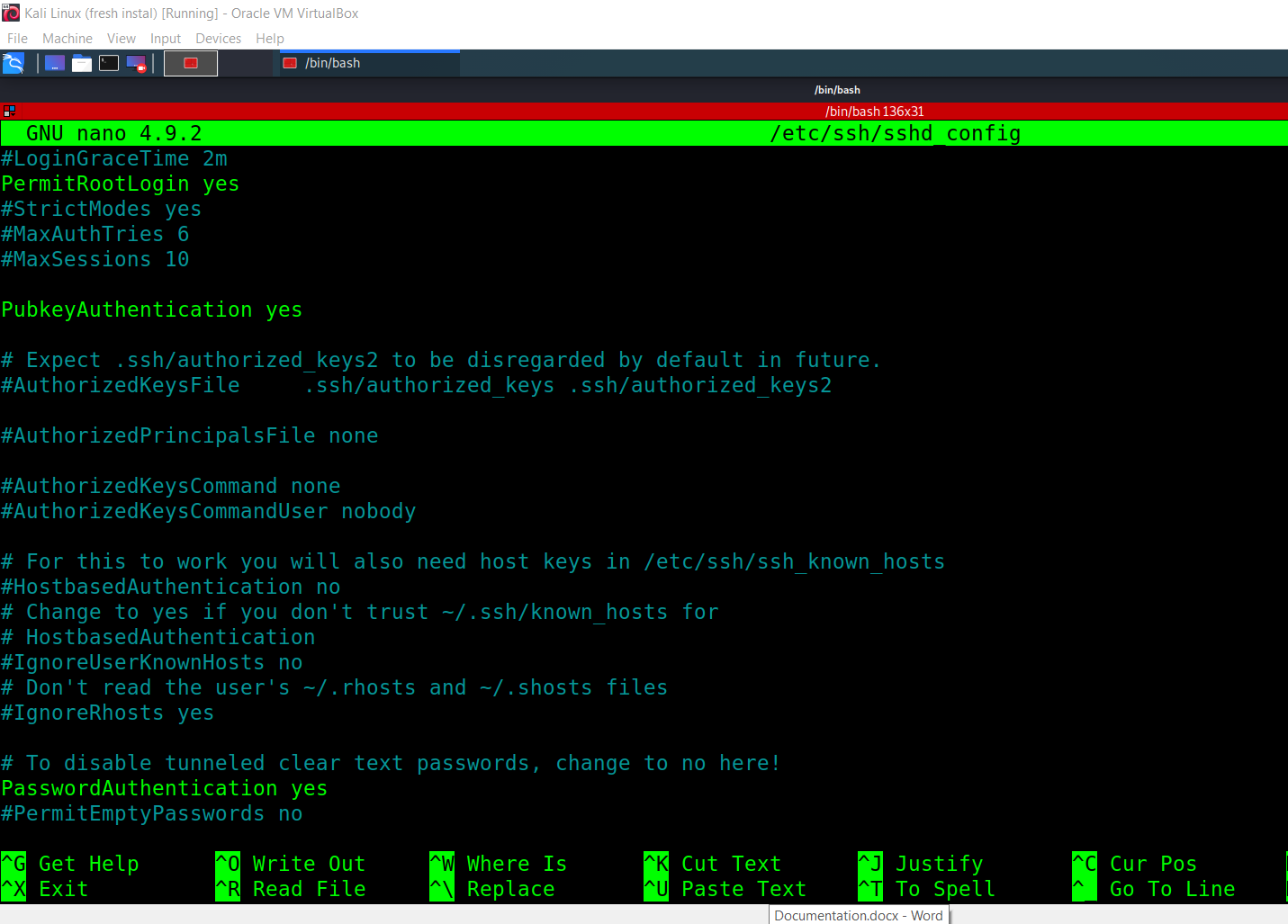


Fig. 3 Lines in fluorescent green changed as required

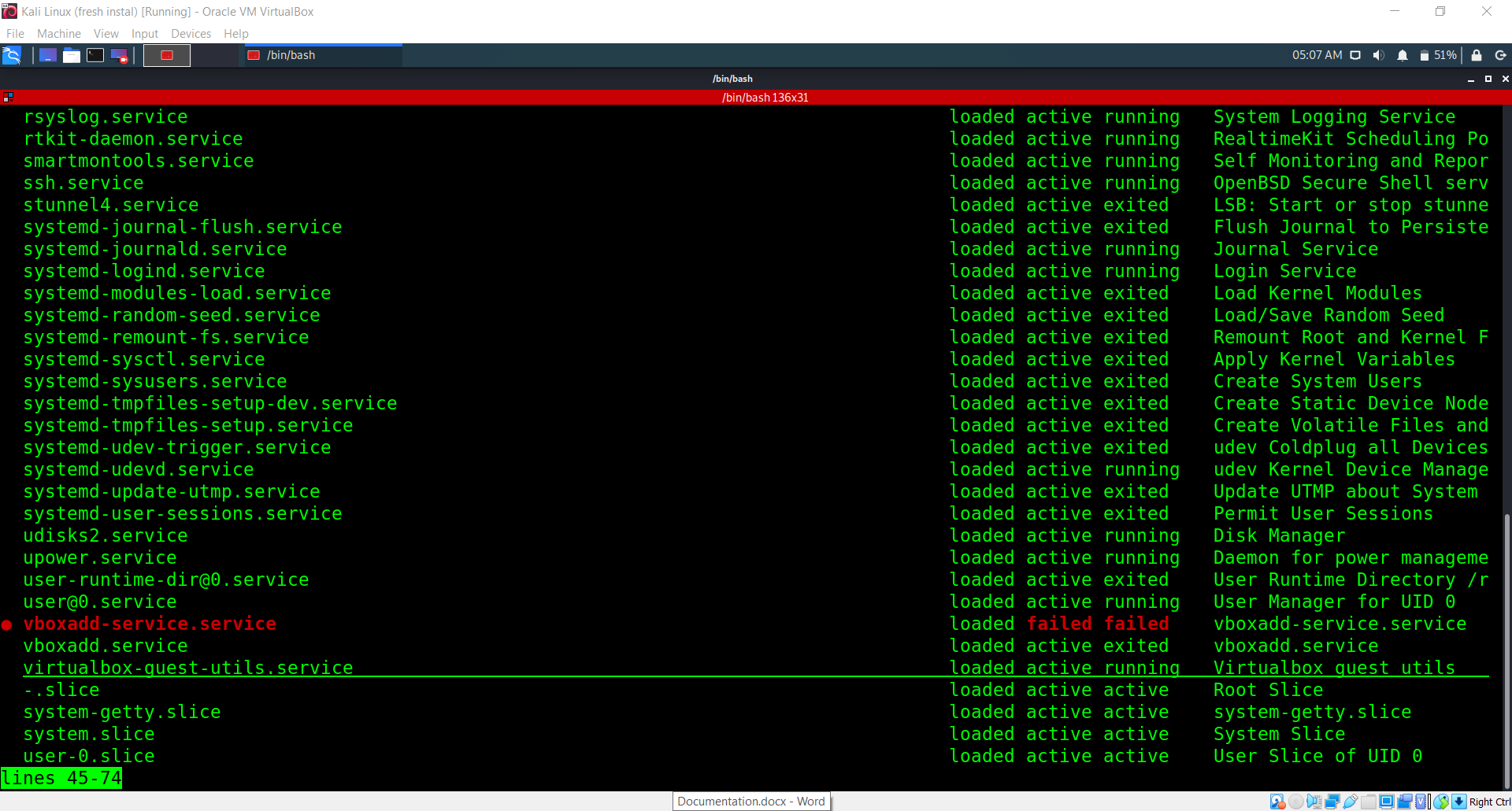


Fig.4 ssh service added to start up programs

Happy Hacking!!!!