

v-Soc installation and configuration manual

Montserrat Flores Castillo

05/09/2025

Contents

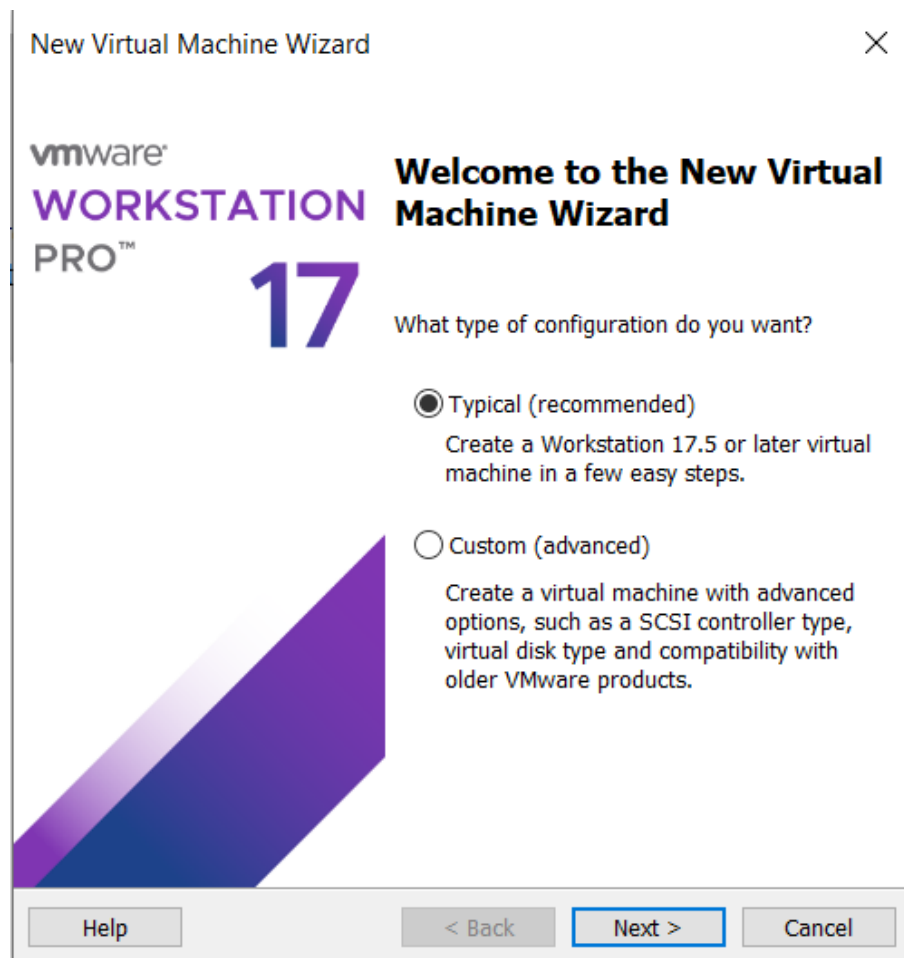
| | |
|---|----|
| 1. Virtual Machine creation and configuration in VMware | 2 |
| 2. Virtual Machine creation and configuration in VirtualBox | 5 |
| 3. Installation of the OS (Ubuntu Server) | 8 |
| 4. Ansible installation step-by-step | 17 |
| 5. Firewall UFW installation and configuration | 20 |
| 6. Snort installation and configuration..... | 21 |
| 7. Syslog-ng installation and configuration | 23 |
| 8. Fluentd installation and configuration | 24 |
| 9. Elasticsearch and Kibana installation and configuration | 25 |
| 10. Grafana installation and configuration..... | 29 |
| 11. Nagios installation and configuration | 31 |
| 12. MITRE CALDERA installation and configuration | 32 |
| 13. Wireshark installation | 34 |
| 14. Links and Resources | 35 |

1. Virtual Machine creation and configuration in VMware

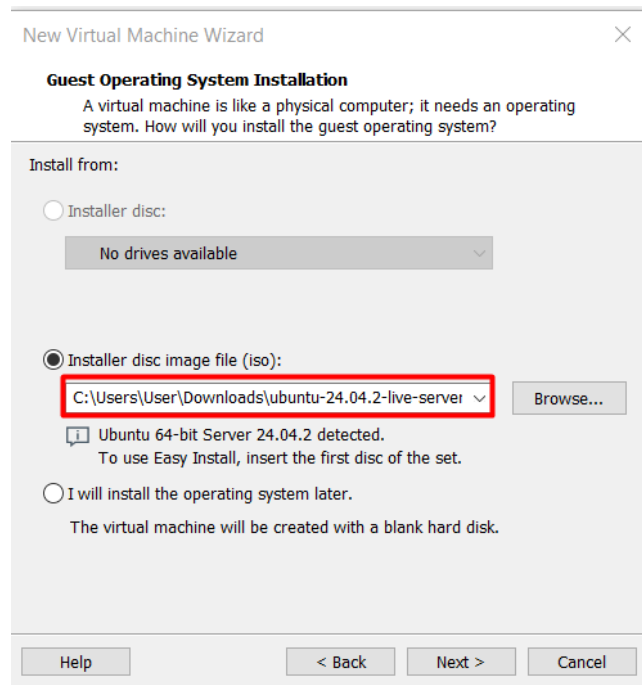
Link for download the software: <https://blogs.vmware.com/workstation/2024/05/vmware-workstation-pro-now-available-free-for-personal-use.html>

For be available to download the VMware pro workstation you will need to create account, and then you will be able to download it.

This is the wizard for creating virtual machines in VMware, normally you will use the typical recommended configuration.



Choose the directory of your ISO image with the Ubuntu OS for example.



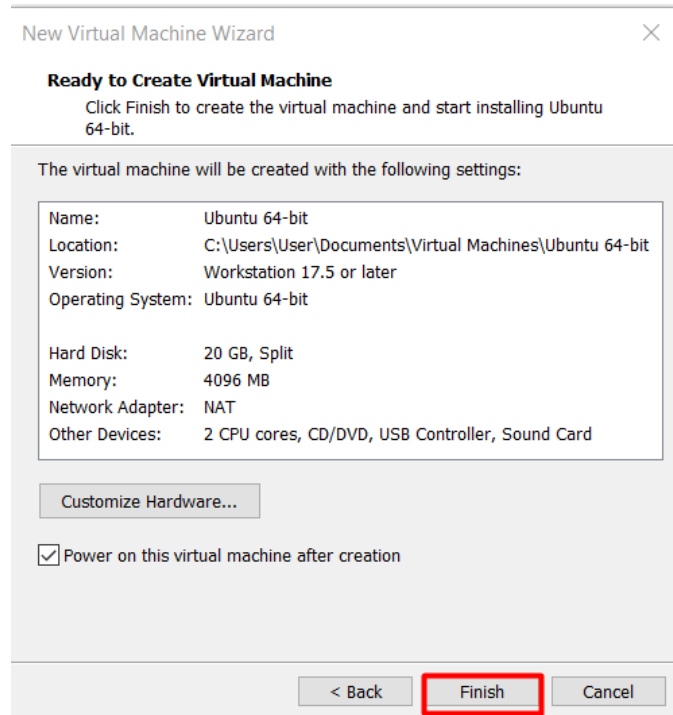
Add a name to your VM and choose the location where is going to be located.

The screenshot shows the 'Name the Virtual Machine' step of the 'New Virtual Machine Wizard'. The title bar says 'New Virtual Machine Wizard' with a close button. The main heading is 'Name the Virtual Machine' with the question 'What name would you like to use for this virtual machine?'. Below this, there is a text box for 'Virtual machine name:' containing 'Ubuntu 64-bit'. Underneath is a 'Location:' section with a text box showing 'C:\Users\User\Documents\Virtual Machines\Ubuntu 64-bit' and a 'Browse...' button. A note states 'The default location can be changed at Edit > Preferences.' At the bottom, there are three buttons: '< Back', 'Next >' (highlighted with a blue border), and 'Cancel'.

Change the size of your disk capacity to the needed one, for example: 120 GB

The screenshot shows the 'Specify Disk Capacity' step of the 'New Virtual Machine Wizard'. The title bar says 'New Virtual Machine Wizard' with a close button. The main heading is 'Specify Disk Capacity' with the question 'How large do you want this disk to be?'. Below this, there is explanatory text: 'The virtual machine's hard disk is stored as one or more files on the host computer's physical disk. These file(s) start small and become larger as you add applications, files, and data to your virtual machine.' This is followed by a 'Maximum disk size (GB):' label and a spinner box set to '20.0', which is highlighted with a red rectangle. Below that, it says 'Recommended size for Ubuntu 64-bit: 20 GB'. There are two radio button options: 'Store virtual disk as a single file' (unselected) and 'Split virtual disk into multiple files' (selected). A note explains: 'Splitting the disk makes it easier to move the virtual machine to another computer but may reduce performance with very large disks.' At the bottom, there are four buttons: 'Help', '< Back', 'Next >', and 'Cancel'.

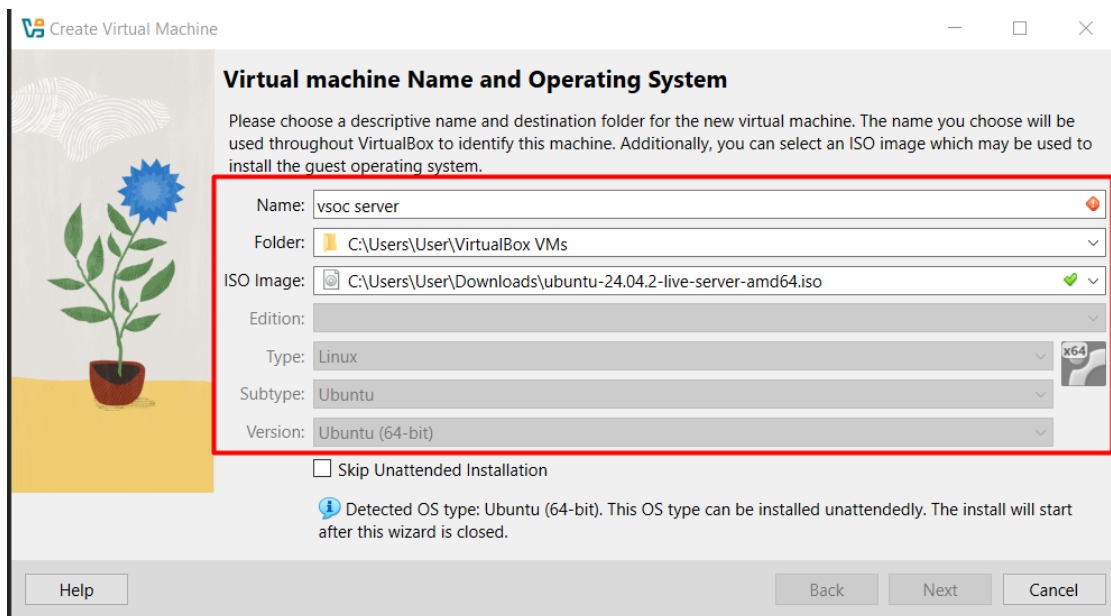
Now you are in the resume page, where you can change any configuration before finishing creating the VM.



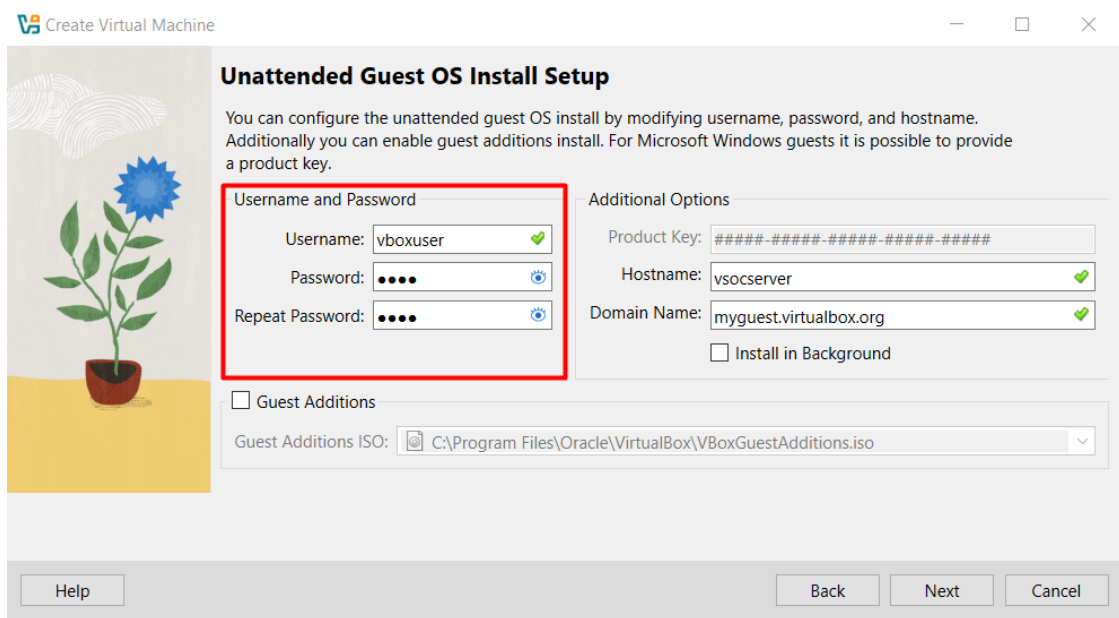
2. Virtual Machine creation and configuration in VirtualBox

Link for download VirtualBox: <https://www.virtualbox.org/wiki/Downloads>

In the vm creation process here we must indicate the name of the machine, where we want to locate it and the ISO of the OS.



Indicate the username for the root user and the password.



The screenshot shows the 'Unattended Guest OS Install Setup' window in the 'Create Virtual Machine' application. The window has a title bar with the application name and standard window controls. On the left is a decorative illustration of a potted plant with a blue flower. The main content area is titled 'Unattended Guest OS Install Setup' and includes a descriptive paragraph. Below this, there are two main sections: 'Username and Password' and 'Additional Options'. The 'Username and Password' section is highlighted with a red rectangle and contains three input fields: 'Username' (set to 'vboxuser' with a green checkmark), 'Password' (masked with dots and an eye icon), and 'Repeat Password' (also masked with dots and an eye icon). The 'Additional Options' section contains a 'Product Key' field (pre-filled with a placeholder), a 'Hostname' field (set to 'vsocserver' with a green checkmark), a 'Domain Name' field (set to 'myguest.virtualbox.org' with a green checkmark), and an 'Install in Background' checkbox. Below these sections is a 'Guest Additions' section with a checkbox and a file selection dropdown showing 'C:\Program Files\Oracle\VirtualBox\VBBoxGuestAdditions.iso'. At the bottom are 'Help', 'Back', 'Next', and 'Cancel' buttons.

Create Virtual Machine

Unattended Guest OS Install Setup

You can configure the unattended guest OS install by modifying username, password, and hostname. Additionally you can enable guest additions install. For Microsoft Windows guests it is possible to provide a product key.

Username and Password

Username: vboxuser ✓

Password: ●●●●

Repeat Password: ●●●●

Additional Options

Product Key: #####-####-####-####-####

Hostname: vsocserver ✓

Domain Name: myguest.virtualbox.org ✓

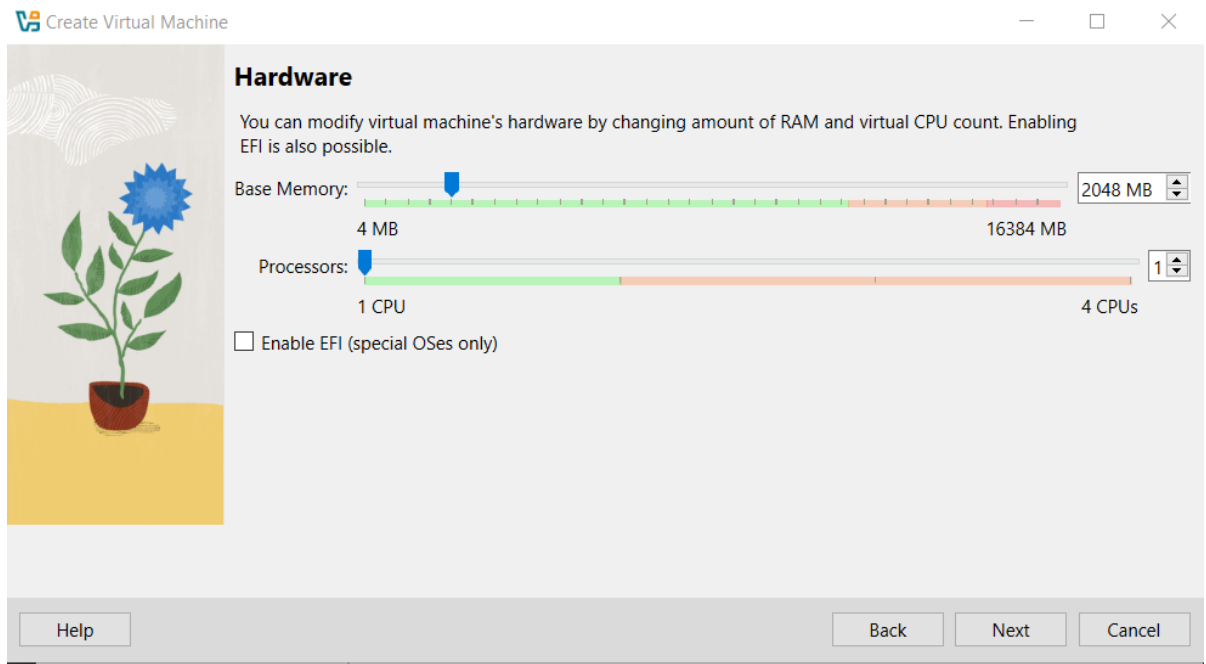
☐ Install in Background

☐ Guest Additions

Guest Additions ISO: C:\Program Files\Oracle\VirtualBox\VBBoxGuestAdditions.iso

Help Back Next Cancel

Here we need to indicate the RAM and CPU we want to use in the VM



The screenshot shows the 'Hardware' configuration window in the 'Create Virtual Machine' application. The window has a title bar with the application name and standard window controls. On the left is a decorative illustration of a potted plant with a blue flower. The main content area is titled 'Hardware' and includes a descriptive paragraph. Below this, there are two main sections: 'Base Memory' and 'Processors'. The 'Base Memory' section features a slider ranging from 4 MB to 16384 MB, with a value of 2048 MB selected. The 'Processors' section features a slider ranging from 1 CPU to 4 CPUs, with a value of 1 selected. There is also an 'Enable EFI (special OSes only)' checkbox. At the bottom are 'Help', 'Back', 'Next', and 'Cancel' buttons.

Create Virtual Machine

Hardware

You can modify virtual machine's hardware by changing amount of RAM and virtual CPU count. Enabling EFI is also possible.

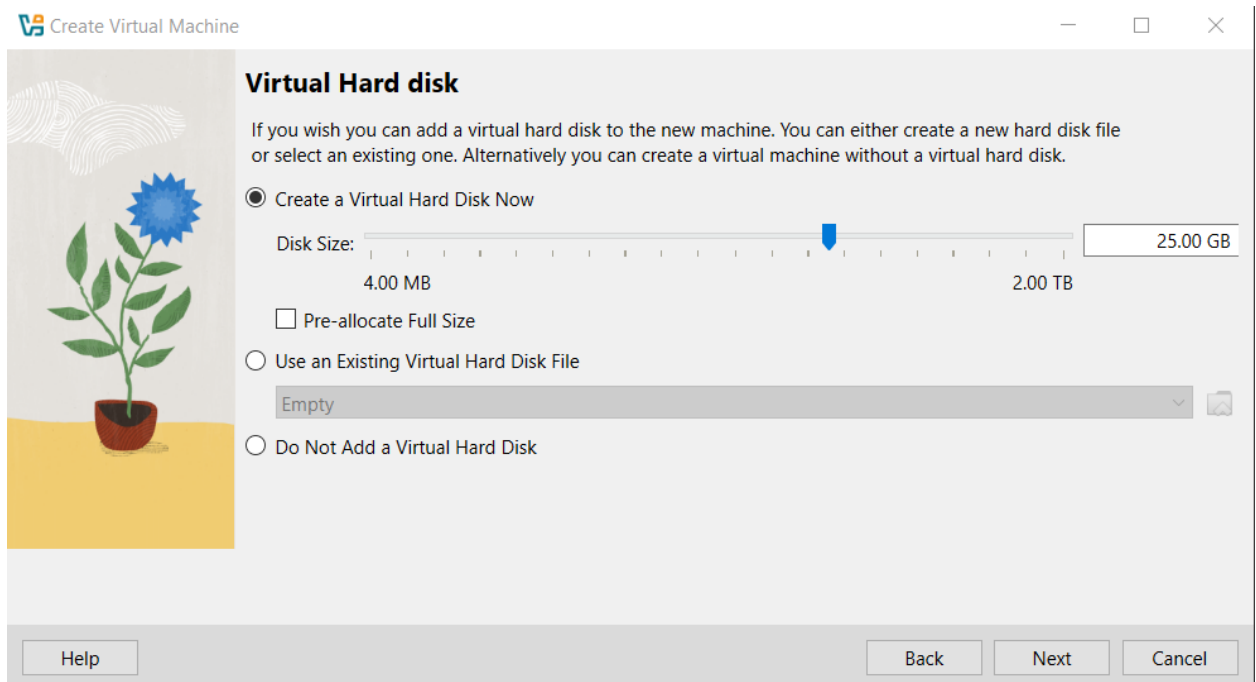
Base Memory: 2048 MB

Processors: 1

☐ Enable EFI (special OSes only)

Help Back Next Cancel

Indicate the size of the disk for the VM.



The screenshot shows the 'Create Virtual Machine' window with the 'Virtual Hard disk' tab selected. On the left is a decorative illustration of a potted plant with a blue flower. The main area contains instructions and options for creating the virtual hard disk. The 'Create a Virtual Hard Disk Now' option is selected. A slider for 'Disk Size' is set to 25.00 GB, with a range from 4.00 MB to 2.00 TB. Other options include 'Pre-allocate Full Size' (unchecked), 'Use an Existing Virtual Hard Disk File' (with a dropdown menu showing 'Empty'), and 'Do Not Add a Virtual Hard Disk' (unchecked). At the bottom are 'Help', 'Back', 'Next', and 'Cancel' buttons.

Create Virtual Machine

Virtual Hard disk

If you wish you can add a virtual hard disk to the new machine. You can either create a new hard disk file or select an existing one. Alternatively you can create a virtual machine without a virtual hard disk.

☒ Create a Virtual Hard Disk Now

Disk Size: 25.00 GB

4.00 MB 2.00 TB

☐ Pre-allocate Full Size

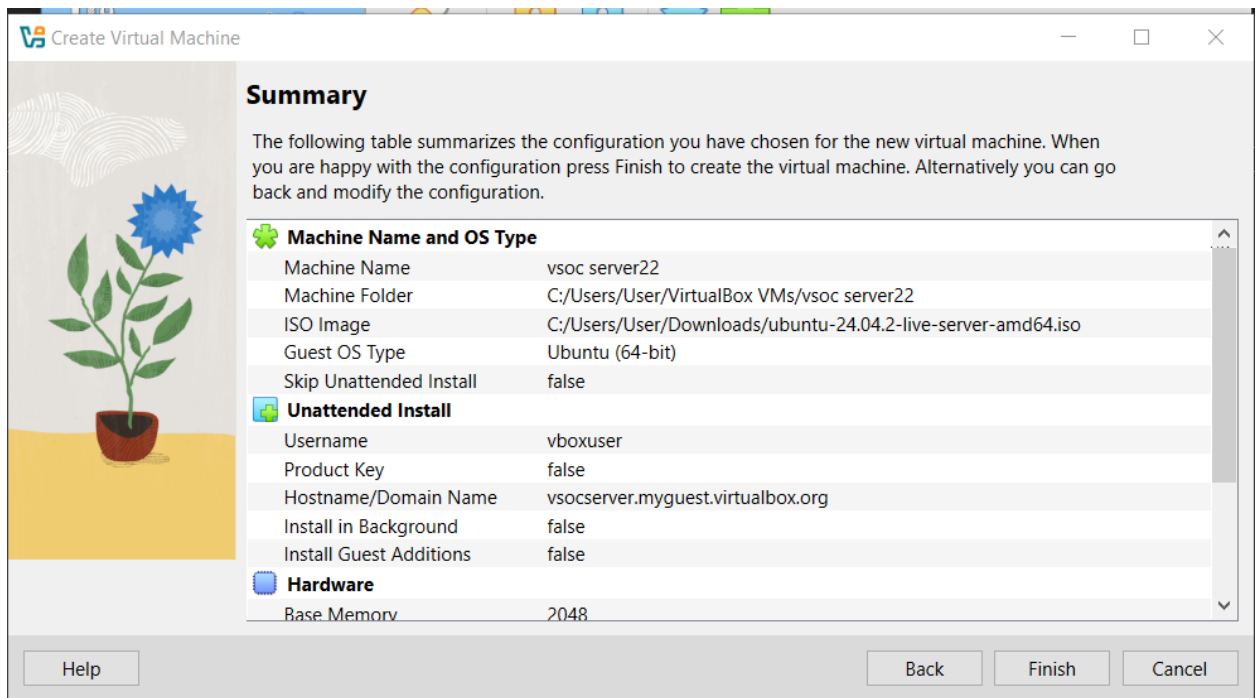
☐ Use an Existing Virtual Hard Disk File

Empty

☐ Do Not Add a Virtual Hard Disk

Help Back Next Cancel

Summary of your VM creation in VirtualBox.



The screenshot shows the 'Create Virtual Machine' window with the 'Summary' tab selected. It displays a summary of the configuration chosen for the new virtual machine. The configuration includes the machine name 'vsoc server22', the ISO image 'ubuntu-24.04.2-live-server-amd64.iso', and the guest OS type 'Ubuntu (64-bit)'. The 'Unattended Install' section shows settings for the username 'vboxuser' and other installation options. The 'Hardware' section shows the base memory set to 2048. At the bottom are 'Help', 'Back', 'Finish', and 'Cancel' buttons.

Create Virtual Machine

Summary

The following table summarizes the configuration you have chosen for the new virtual machine. When you are happy with the configuration press Finish to create the virtual machine. Alternatively you can go back and modify the configuration.

| | |
|---------------------------------|--|
| Machine Name and OS Type | |
| Machine Name | vsoc server22 |
| Machine Folder | C:/Users/User/VirtualBox VMs/vsoc server22 |
| ISO Image | C:/Users/User/Downloads/ubuntu-24.04.2-live-server-amd64.iso |
| Guest OS Type | Ubuntu (64-bit) |
| Skip Unattended Install | false |
| Unattended Install | |
| Username | vboxuser |
| Product Key | false |
| Hostname/Domain Name | vsocserver.myguest.virtualbox.org |
| Install in Background | false |
| Install Guest Additions | false |
| Hardware | |
| Base Memory | 2048 |

Help Back Finish Cancel

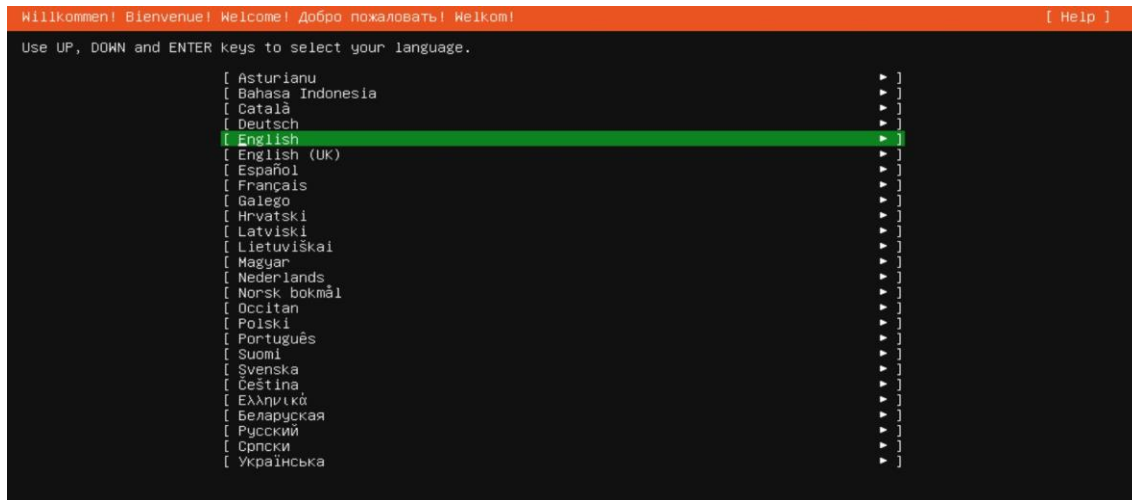
3. Installation of the OS (Ubuntu Server)

At the beginning of the installation we will see this prompt which can take couple minutes until it pop you up the installation of the operating system.

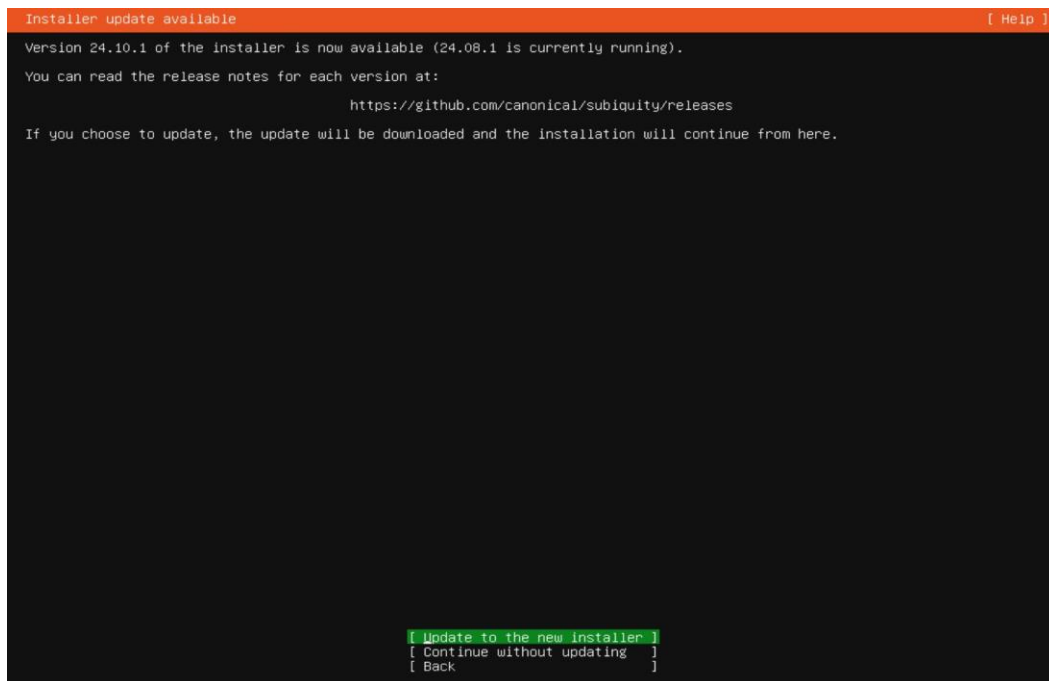
```
[ 12.822540] overlays: null uuid detected in lower fs '/', falling back to xino=off,index=off,nfs_export=off.
Begin: Running /scripts/casper-premount ... done.
done.
Begin: Creating debconf-communicate fifo mechanism ... done.
Begin: Running /scripts/casper-bottom ... Begin: Moving mount points... .. done.
Begin: Configuring fstab... .. done.
Begin: Setting up locales... .. done.
Begin: Setting up automatic login... .. done.
Begin: Disabling systemd's GPT auto generator... .. done.
Begin: Setting hostname... .. done.
Begin: Setting up console keyboard... .. done.
Begin: Applying desktop settings... .. done.
Begin: Regenerating SSL certificate... .. done.
Begin: Loading preseed file... .. done.
Begin: Adding live session user... .. passwd: password expiry information changed.
done.
Begin: Setting up init... .. done.
Begin: Configuring accessibility options... .. done.
Begin: Disabling update-notifier... .. done.
Begin: Configuring power management... .. done.
Begin: Enabling detection of crashes... .. done.
Begin: Disabling unnecessary KDE services... .. done.
Begin: Fixing language selector... .. done.
Begin: Disabling trackerd... .. done.
Begin: Adding APT-CDROM source... .. Using CD-ROM mount point /cdrom/
Identifying... [6b83e7359c62d3be5f1e15a61d28f8d4-2]
Scanning disc for index files...
Found 2 package indexes, 0 source indexes, 0 translation indexes and 1 signatures
Found label 'Ubuntu-Server 22.04.5 LTS _Jammy Jellyfish_ - Release amd64 (20240911.4)'
This disc is called:
'Ubuntu-Server 22.04.5 LTS _Jammy Jellyfish_ - Release amd64 (20240911.4)'
Copying package lists...done.
Begin: Running /scripts/nfs-bottom ... done.
Begin: Running /scripts/init-bottom ... [ 18.238755] loop4: detected capacity change from 0 to 299944
[ 18.251953] loop5: detected capacity change from 0 to 611786
[ 18.266233] overlays: "xino" feature enabled using 32 upper inode bits.
done.
[ 18.905069] systemd[1]: Inserted module 'autofs4'
[ 19.028024] systemd[1]: systemd 249.11-0ubuntu3.12 running in system mode (+PAM +AUDIT +SELINUX +APPARMOR +IMA +SMACK +SECCOMP +GCRYPT +GNUTLS +OPENSSL +ACL +BLKID +CURL +ELFUTILS +FIDO2 +IOM2 +IDN +IPTC +KMOD +LIBCRYPTSETUP +LIBFDISK +PCRE2 +PAQUILITY +P11KIT +QRENCODE +B2IP2 +LZ4 +XZ +ZLIB +ZSTD -XKBCOMMON +UTMP +SYSVINIT default-hierarchy=unified)
[ 19.031125] systemd[1]: Detected virtualization vmware.
[ 19.031787] systemd[1]: Detected architecture x86_64.

Welcome to Ubuntu 22.04.5 LTS!
```

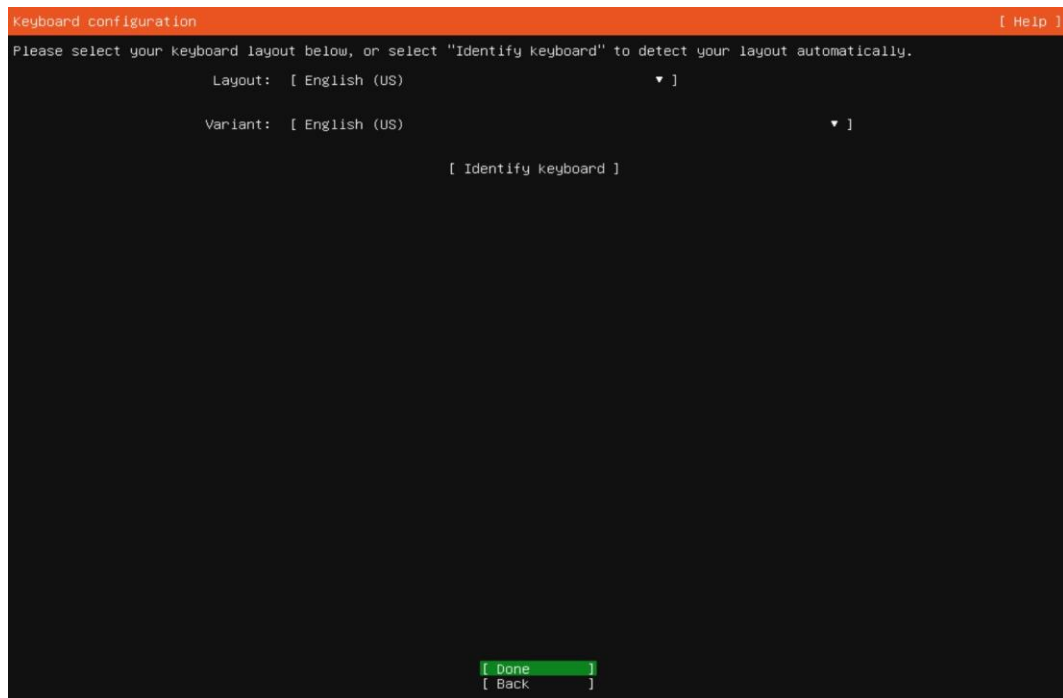
In this section we will need to choose the language of the system:



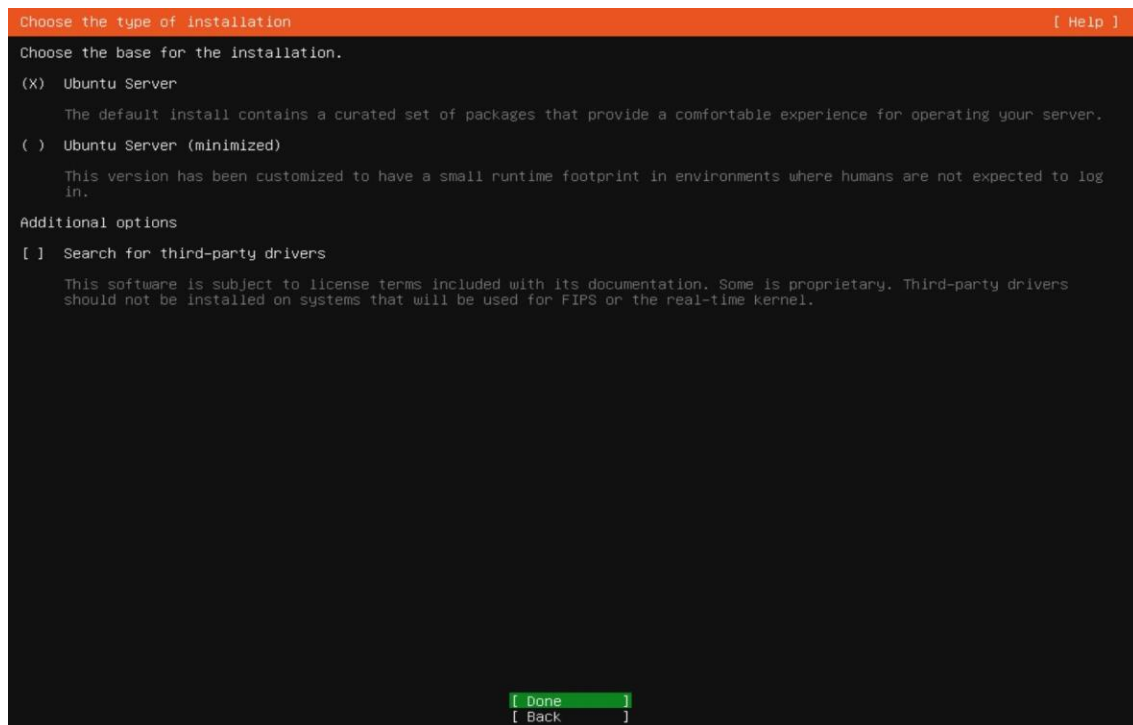
Indicates we have a new update available so is recommended to install the operating system with the newest version:



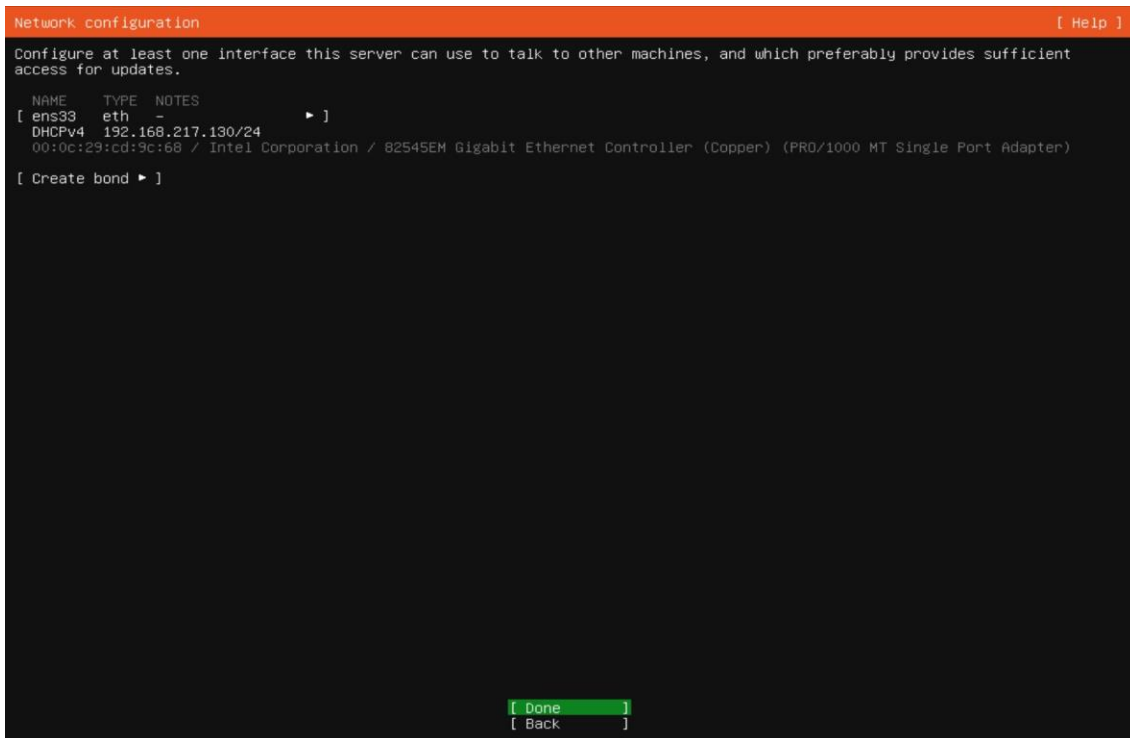
Choose the language of the keyboard and layout of the keyboard:



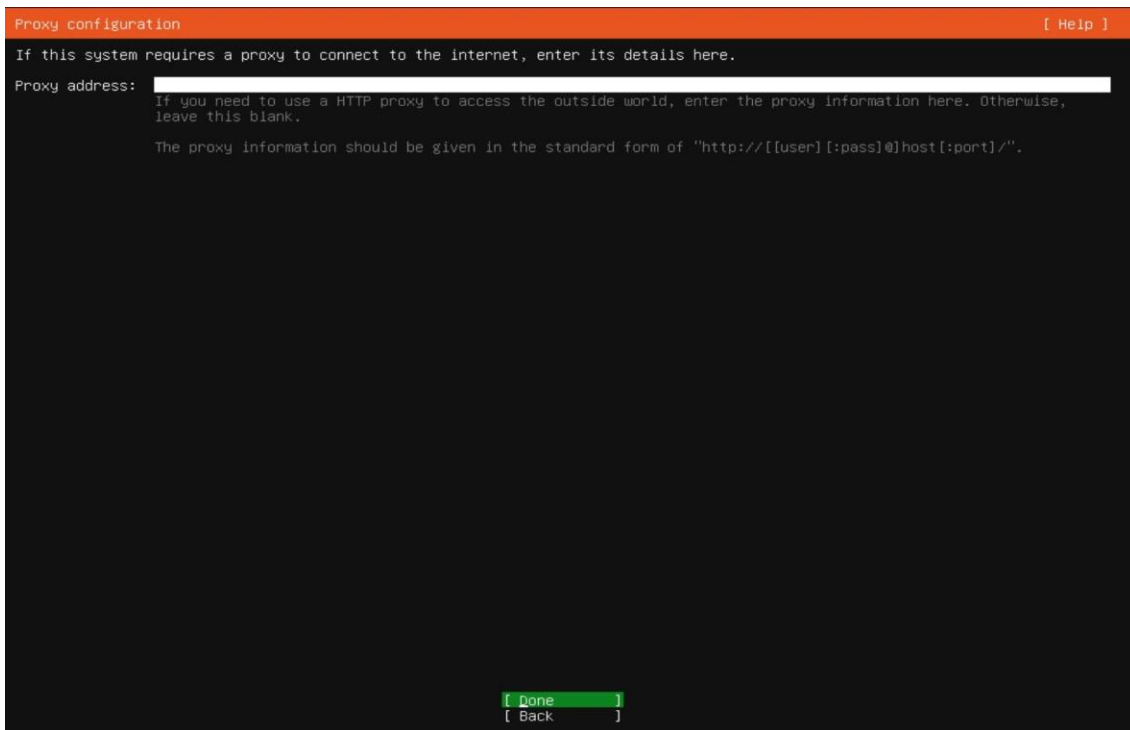
Here we indicate which kind of installation of Ubuntu Server we want, in our case the normal installation of Ubuntu Server:



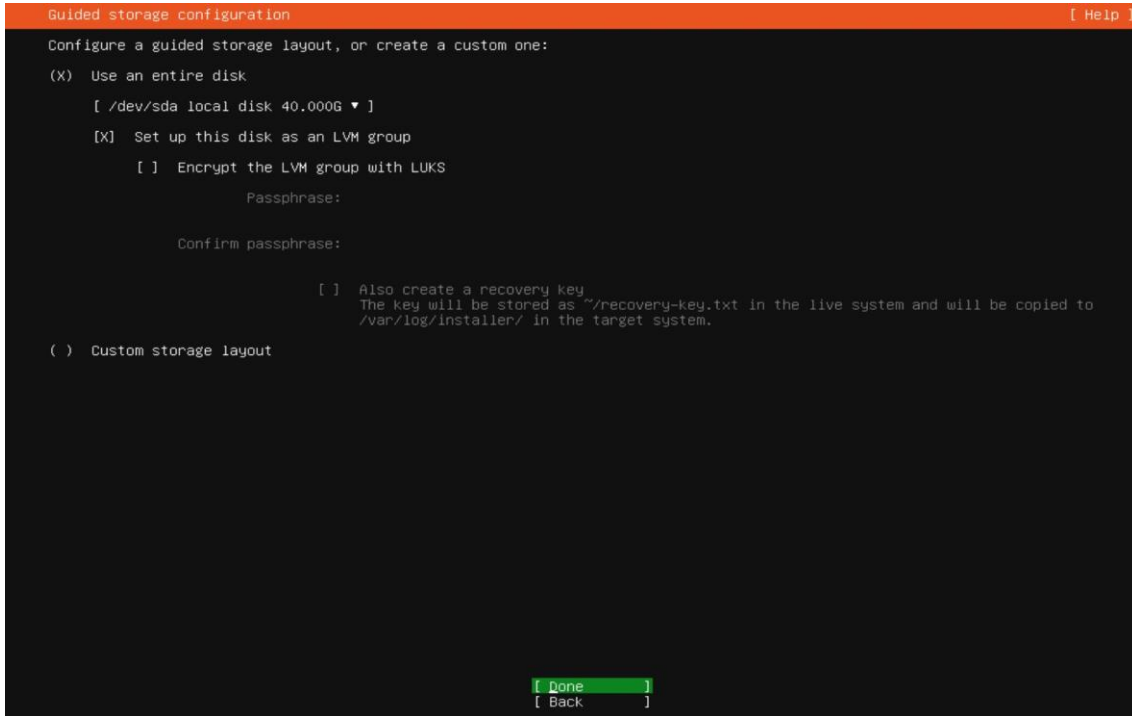
The network configuration we leaved in the default configuration until we need to configure it in the future:



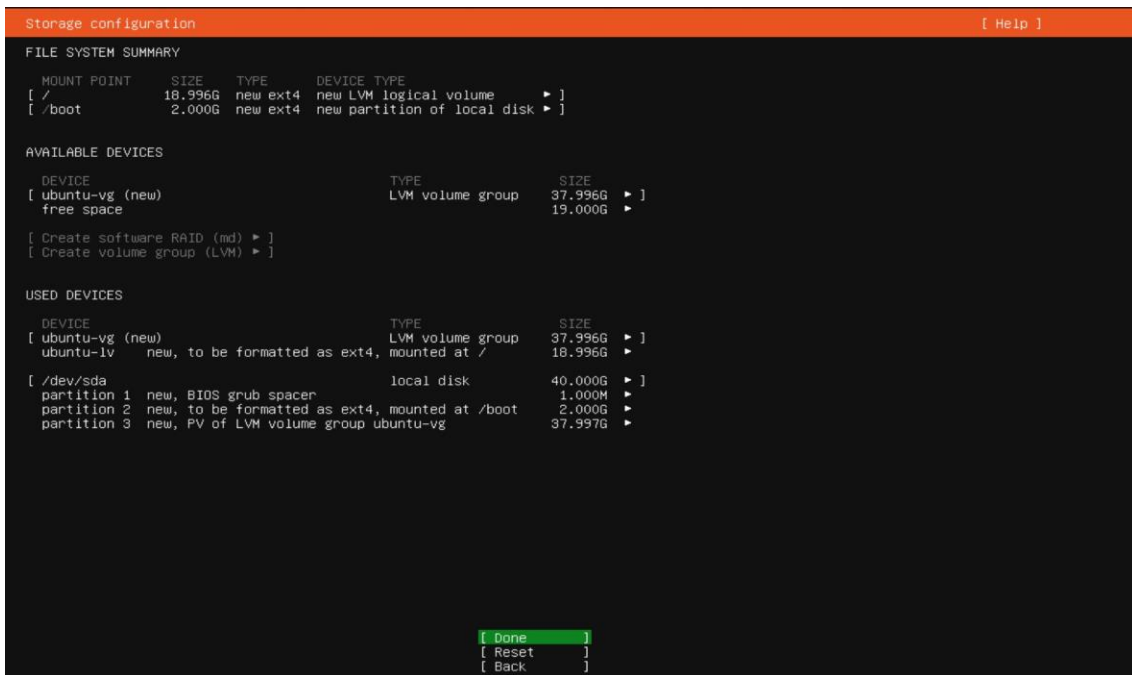
We left in the default configuration empty until we configure our own proxy:



The storage configuration we leave by default here we can change where we want to locate the installation of the ubuntu server.



More extra configuration we can leave in default, like where is locate the /boot or the root system
/:



Creation of superuser and user of your ubuntu server.

Profile configuration [Help]

Enter the username and password you will use to log in to the system. You can configure SSH access on a later screen, but a password is still needed for sudo.

Your name:

Your server's name: The name it uses when it talks to other computers.

Pick a username:

Choose a password:

Confirm your password:

[Done]

Installation process:

```
Installing system [ Help ]

subiquity/load_cloud_config/extract_autoinstall:
subiquity/Early/apply_autoinstall_config:
subiquity/Reporting/apply_autoinstall_config:
subiquity/Error/apply_autoinstall_config:
subiquity/Userdata/apply_autoinstall_config:
subiquity/Package/apply_autoinstall_config:
subiquity/Debconf/apply_autoinstall_config:
subiquity/Kernel/apply_autoinstall_config:
subiquity/KernelCrashDumps/apply_autoinstall_config:
subiquity/Zdev/apply_autoinstall_config:
subiquity/Ad/apply_autoinstall_config:
subiquity/Late/apply_autoinstall_config:
configuring apt
curtin command in-target
installing system
executing curtin install initial step
executing curtin install partitioning step
curtin command install
configuring storage
running 'curtin block-meta simple'
curtin command block-meta
removing previous storage devices
configuring disk: disk-sda
configuring partition: partition-0
configuring partition: partition-1
configuring format: format-0
configuring partition: partition-2
configuring lvm_voigroup: lvm_voigroup-0
configuring lvm_partition: lvm_partition-0
configuring format: format-1
configuring mount: mount-1
configuring mount: mount-0
executing curtin install extract step
curtin command install
writing install sources to disk
running 'curtin extract'
curtin command extract
acquiring and extracting image from cp:///tmp/tmpgdlio3bn/mount /

[ View full log ]
```

After installing the operating system is always recommended to check for last updates and upgrades. Using the command: (`sudo apt update`, `sudo apt upgrade`):

```
root@vsoc:/home/vsoc# apt update
Get:1 http://security.ubuntu.com/ubuntu jammy-security InRelease [129 kB]
Hit:2 http://us.archive.ubuntu.com/ubuntu jammy InRelease
Get:3 http://us.archive.ubuntu.com/ubuntu jammy-updates InRelease [128 kB]
Hit:4 http://us.archive.ubuntu.com/ubuntu jammy-backports InRelease
Fetched 257 kB in 1s (176 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
26 packages can be upgraded. Run 'apt list --upgradable' to see them.
root@vsoc:/home/vsoc# apt upgrade
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
The following NEW packages will be installed:
python3-packaging
The following packages will be upgraded:
cloud-init distro-info-data dmidecode gir1.2-packagekit-glib-1.0 libmbim-glib4 libmbim-proxy
libmm-glib0 libpackagekit-glib2-18 libpam-modules libpam-modules-bin libpam-runtime libpam0g
libpcap0.8 modemmanager packagekit packagekit-tools snapd sosreport ubuntu-advantage-tools
ubuntu-minimal ubuntu-pro-client ubuntu-pro-client-110n ubuntu-server ubuntu-server-minimal
ubuntu-standard xfsprogs
26 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 32.5 MB of archives.
After this operation, 5,667 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y_
```

Now we are going to installing the GUI version in your ubuntu server.

Command: `sudo apt install taskel dialog`

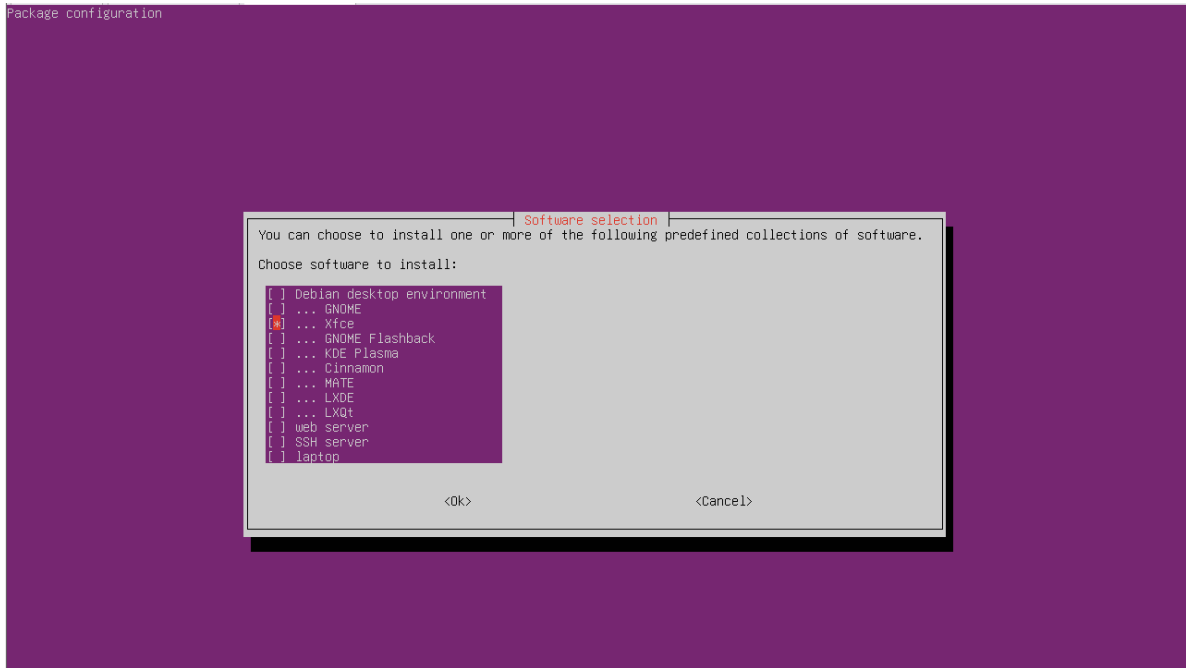
After installing the taskel dialog which is a program to launch an interface version inside of your CLI. Now we will execute this program.

`sudo taskel`

Command:

In the menu of options of desktop environments, we will choose XFCE, because is lightweight and less resource consuming.

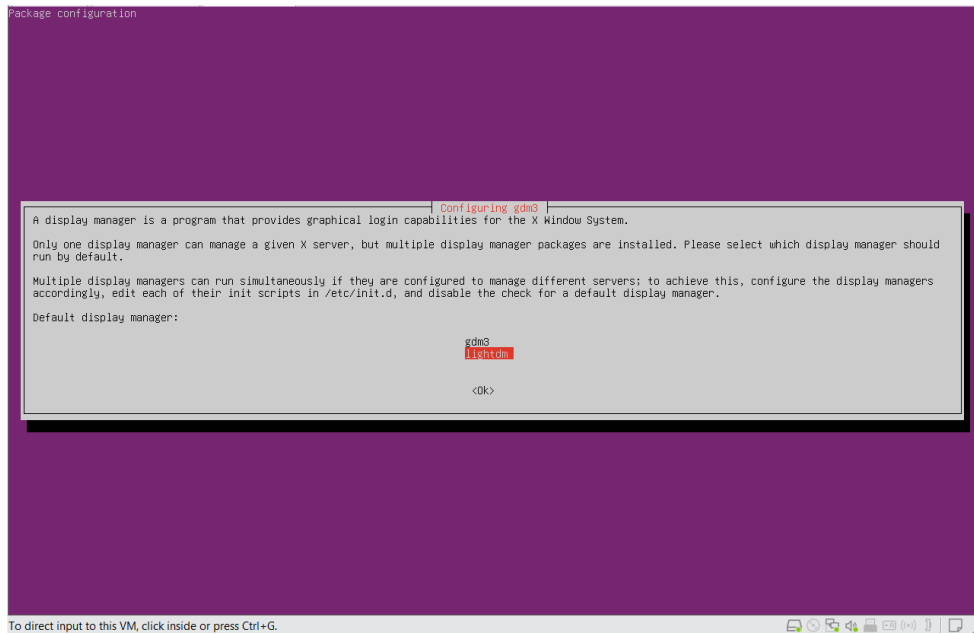
Tip: For move in the menu use your arrows in the keyboard and press space for mark your option. For move to the OK use the TAB and press Enter.



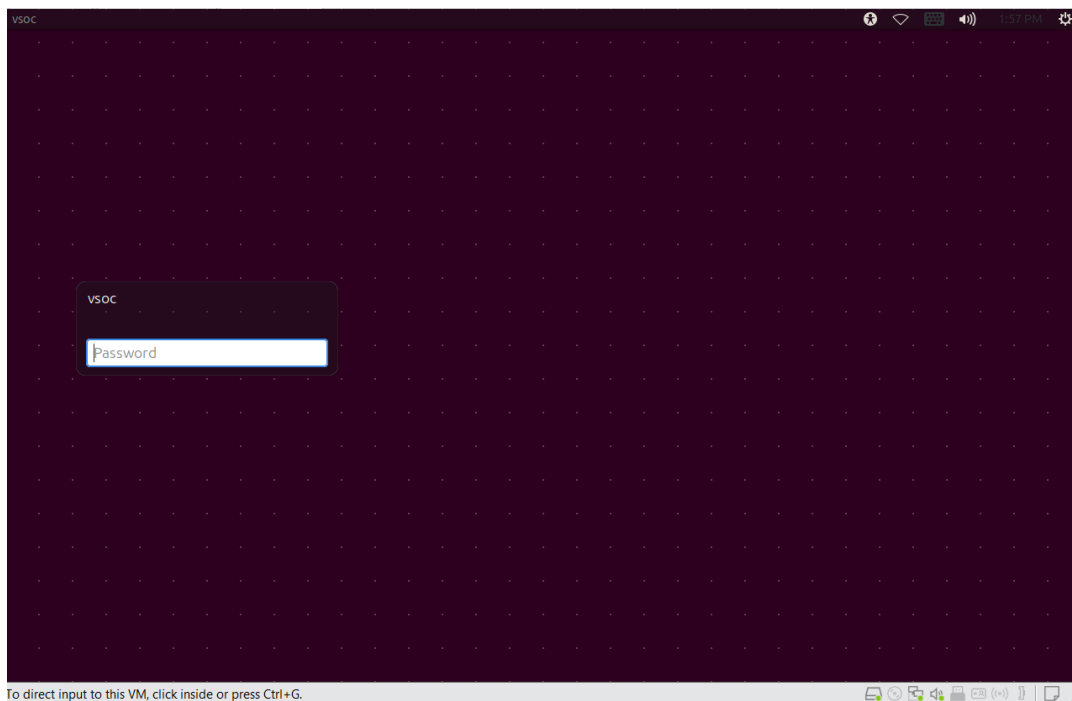
Now after installing the xfce desktop , we need to execute this command for install all the necessities requeriments for ubuntu server have a GUI.

Command: `sudo apt-get install ubuntu-desktop`

Choose the lightdm option for graphical option.



After installing the desktop **RESTART** your virtual machine to activate the GUI.



4. Ansible installation step-by-step

For install Ansible we need first to install the common software properties and add the repository for install the Ansible package.

Command

```
sudo apt install software-properties-common  
sudo add-apt-repository --yes --update ppa:ansible/ansible  
sudo apt install ansible
```

```
vsoc@vsoc:~$ sudo apt install software-properties-common  
[sudo] password for vsoc:  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
software-properties-common is already the newest version (0.99.49.2).  
software-properties-common set to manually installed.  
0 upgraded, 0 newly installed, 0 to remove and 1 not upgraded.  
vsoc@vsoc:~$
```

```
vsoc@vsoc:~$ sudo add-apt-repository --yes --update ppa:ansible/ansible  
Repository: 'Types: deb  
URIs: https://ppa.launchpadcontent.net/ansible/ansible/ubuntu/  
Suites: noble  
Components: main  
'  
Description:  
Ansible is a radically simple IT automation platform that makes your application  
s and systems easier to deploy. Avoid writing scripts or custom code to deploy a  
nd update your applications– automate in a language that approaches plain Englis  
h, using SSH, with no agents to install on remote systems.  
  
http://ansible.com/  
  
If you face any issues while installing Ansible PPA, file an issue here:  
https://github.com/ansible-community/ppa/issues  
More info: https://launchpad.net/~ansible/+archive/ubuntu/ansible  
Adding repository.  
Hit:1 http://us.archive.ubuntu.com/ubuntu noble InRelease  
Get:2 http://us.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]  
Hit:3 http://security.ubuntu.com/ubuntu noble-security InRelease  
Get:4 http://us.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]  
Get:5 http://us.archive.ubuntu.com/ubuntu noble/main Icons (48x48) [106 kB]
```

```

vsoc@vsoc:~$ sudo apt install ansible
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  ansible-core python3-kerberos python3-ntlm-auth python3-requests-ntlm
  python3-resolvelib python3-winrm python3-xlrd python3-xlsxwriter sshpass
The following NEW packages will be installed:
  ansible ansible-core python3-kerberos python3-ntlm-auth
  python3-requests-ntlm python3-resolvelib python3-winrm python3-xlrd
  python3-xlsxwriter sshpass
0 upgraded, 9 newly installed, 0 to remove and 1 not upgraded.
Need to get 20.1 MB of archives.
After this operation, 228 MB of additional disk space will be used.
Do you want to continue? [Y/n]

```

Now we need to get the last version of the file configuration of Ansible:

Command: `ansible-config init --disabled > ansible.cfg`

```

vsoc@vsoc:~$ ansible-config init --disabled > ansible.cfg
vsoc@vsoc:~$

```

Add this lines of code, for configure the ansible configuration file located in /etc/ansible/ansible.cfg

```

vsoc@vsoc: /etc/ansible
GNU nano 7.2      ansible.cfg *
# Also you can now have a more complete file by including existing plugins:
# ansible-config init --disabled -t all > ansible.cfg

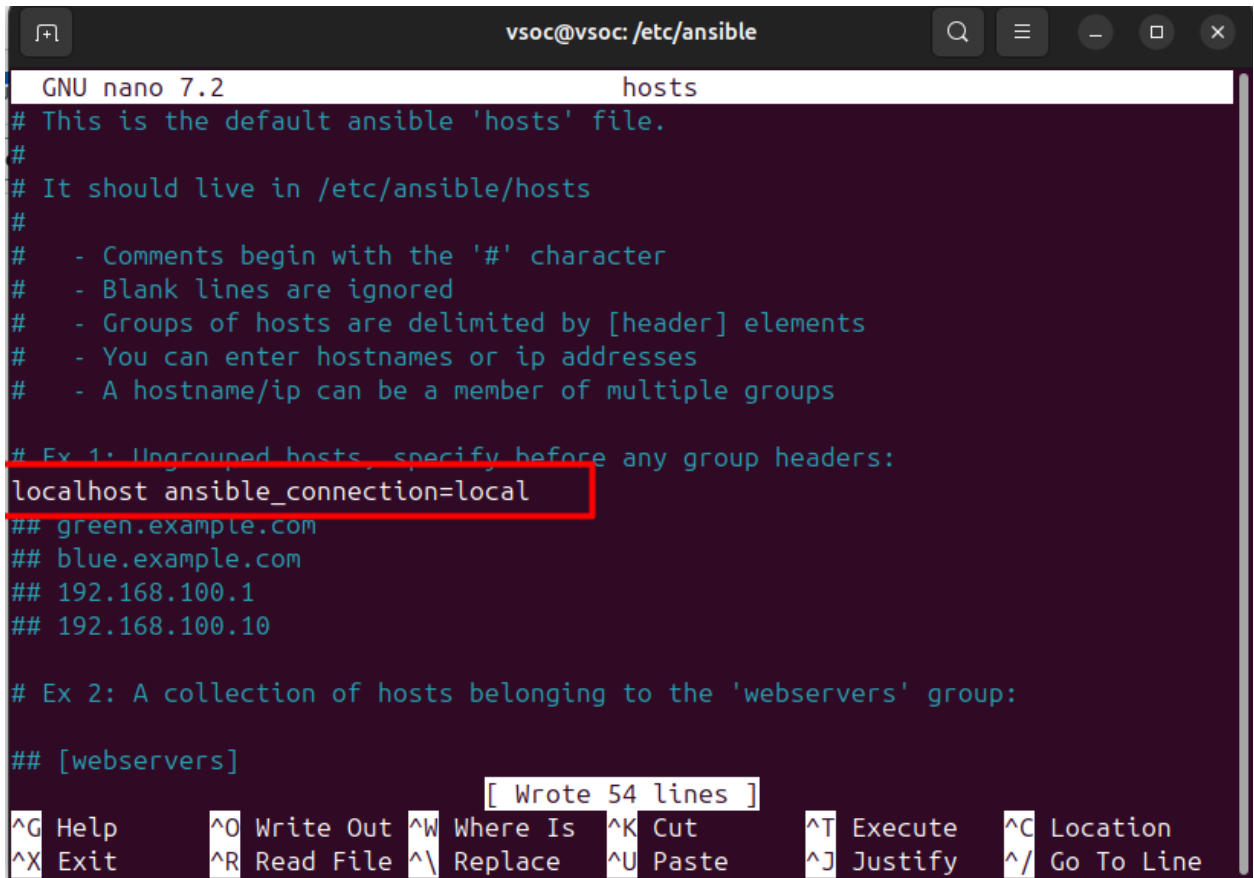
# For previous versions of Ansible you can check for examples in the 'stable' b>
# Note that this file was always incomplete and lagging changes to configurati>

# for example, for 2.9: https://github.com/ansible/ansible/blob/stable-2.9/exam>
[default]
inventory = /etc/ansible/hosts
remote_user = vsoc
host_key_cheking = False
retry_files_enabled = False
interpreter_python = auto_silent
[privilege_escalation]
become = True
become_method = sudo
become_user = root
become_ask_pass = False
[connection]
type = local

^G Help      ^O Write Out ^W Where Is   ^K Cut       ^T Execute   ^C Location
^X Exit      ^R Read File ^\ Replace    ^U Paste     ^J Justify   ^_ Go To Line

```

Now we need to indicate inside of the file hosts inside of the Ansible directory that we want to do our installations locally. Add the line inside of the /etc/ansible/hosts file.



```
vsoc@vsoc: /etc/ansible
GNU nano 7.2 hosts
# This is the default ansible 'hosts' file.
#
# It should live in /etc/ansible/hosts
#
# - Comments begin with the '#' character
# - Blank lines are ignored
# - Groups of hosts are delimited by [header] elements
# - You can enter hostnames or ip addresses
# - A hostname/ip can be a member of multiple groups
#
# Ex 1: Ungrouped hosts, specify before any group headers:
localhost ansible_connection=local
## green.example.com
## blue.example.com
## 192.168.100.1
## 192.168.100.10
#
# Ex 2: A collection of hosts belonging to the 'webservers' group:
## [webservers]
[ Wrote 54 lines ]
^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify   ^/ Go To Line
```

5. Firewall UFW installation and configuration

The UFW is already installed by default in your Linux OS, just need to be configured and enabled.

```
sudo ufw default deny incoming
```

Command: *sudo ufw default allow outgoing*

```
vsoc@vsoc:~$ sudo ufw default deny incoming
Default incoming policy changed to 'deny'
(be sure to update your rules accordingly)
vsoc@vsoc:~$ sudo ufw default allow outgoing
Default outgoing policy changed to 'allow'
(be sure to update your rules accordingly)
```

Now our UFW Firewall can be activated and add the rules.

Command: *sudo ufw enable*

```
sudo ufw allow 9200/tcp | sudo ufw allow 9300/tcp | sudo ufw allow
5000/tcp | sudo ufw allow 5044/tcp | sudo ufw allow 5061/tcp | sudo ufw
allow 80/tcp | sudo ufw allow 443/tcp | sudo ufw allow 9600/tcp | sudo
ufw allow 22/tcp | sudo ufw allow 5040/tcp
```

```
vsoc@vsoc:~$ sudo ufw enable
Firewall is active and enabled on system startup
vsoc@vsoc:~$ sudo ufw status
Status: active
vsoc@vsoc:~$ sudo ufw allow 9200,9300,5000,5044,5601,80,443,22,514,9600/tcp
Rule added
Rule added (v6)
```

```
vsoc@vsoc:~$ sudo ufw status
Status: active

To Action From
--
22,80,443,514,5000,5044,5601,9200,9300,9600/tcp ALLOW Anywhere
22,80,443,514,5000,5044,5601,9200,9300,9600/tcp (v6) ALLOW Anywhere (v6)
```

Command: *sudo ufw reload*
sudo ufw logging high

6. Snort installation and configuration

For install snort we are going to use the ansible playbook named as “snort.yml”. This file must be located inside of /etc/ansible/ and you need to be in that path.

Command: `sudo ansible-playbook -c local snort.yml`

Now you need to configure snort with the alert’s necessities, and functions for detecting traffic in the network.

Command: `sudo nano /etc/snort/rules/local.rules`

```
# -----
# LOCAL RULES
# -----
# This file intentionally does not come with signatures.  Put your local
# additions here.
drop tcp any any -> any any (msg:"Blocking known malicious IP"; sid:1000002; rev:1;)
drop icmp any any -> any any (msg:"Blocking ICMP (Ping)"; sid:1000003; rev:1;)
drop tcp any any -> any 80 (msg:"Fake AV Domain Access"; content:"Host: fakeav.example.c
eader; sid:1000006; rev:1;)
drop tcp any any -> any 23 (msg:"Telnet Attempt Blocked"; sid:1000008; rev:1;)
#alert icmp any any -> any any (msg:"ICMP Packet Detected"; sid:1000001; rev:1;)
# Alert on any TCP traffic (for testing purposes)
alert tcp any any -> any any (msg:"Alert: TCP traffic detected"; sid:1000002; rev:1;)

# Alert on any ICMP traffic (e.g., ping)
alert icmp any any -> any any (msg:"Alert: ICMP packet detected"; sid:1000003; rev:1;)

# Alert on SSH connections
alert tcp any any -> any 22 (msg:"Alert: SSH connection attempt"; sid:1000004; rev:1;)

# Alert on HTTP traffic (port 80)
alert tcp any any -> any 80 (msg:"Alert: HTTP traffic detected"; sid:1000005; rev:1;)

# Alert on suspicious DNS queries (UDP port 53)
#alert udp any any
```

Att: Also file local.rules is provided for download and substitute directly.

Save the file and restart the services of snort for applying the rules and check if it works properly.

Command: `sudo systemctl enable snort`
`sudo systemctl restart snort`
`sudo snort -A console -c /etc/snort/snort.conf -i ens33`

```
Commencing packet processing (pid=12762)
05/09-09:26:54.849941  [**] [1:1000002:1] Alert: TCP traffic detected [**] [Priority: 0]
20.177.193.443 -> 192.168.159.133:35486
05/09-09:26:56.845518  [**] [1:1000003:1] Alert: ICMP packet detected [**] [Priority: 0]
.168.159.1 -> 192.168.159.133
05/09-09:26:56.845565  [**] [1:1000003:1] Alert: ICMP packet detected [**] [Priority: 0]
.168.159.133 -> 192.168.159.1
05/09-09:26:57.856870  [**] [1:1000003:1] Alert: ICMP packet detected [**] [Priority: 0]
.168.159.1 -> 192.168.159.133
05/09-09:26:57.856907  [**] [1:1000003:1] Alert: ICMP packet detected [**] [Priority: 0]
.168.159.133 -> 192.168.159.1
05/09-09:26:58.878256  [**] [1:1000003:1] Alert: ICMP packet detected [**] [Priority: 0]
.168.159.1 -> 192.168.159.133
05/09-09:26:58.878460  [**] [1:1000003:1] Alert: ICMP packet detected [**] [Priority: 0]
.168.159.133 -> 192.168.159.1
05/09-09:26:59.895903  [**] [1:1000003:1] Alert: ICMP packet detected [**] [Priority: 0]
.168.159.1 -> 192.168.159.133
05/09-09:26:59.895950  [**] [1:1000003:1] Alert: ICMP packet detected [**] [Priority: 0]
.168.159.133 -> 192.168.159.1
^C*** Caught Int-Signal
^Z
[1]+  Stopped                  sudo snort -A console -c /etc/snort/snort.conf -i ens33
```

7. Syslog-ng installation and configuration

For install syslog-ng we are going to use the ansible playbook named as “syslog-ng.yml”. This file must be located inside of /etc/ansible/ and you need to be in that path.

Command: `sudo ansible-playbook -c local syslog-ng.yml`

Now you need to configure syslog-ng for detect logs from UFW Firewall and Snort.

Command: `sudo nano /etc/syslog-ng/syslog-ng.conf`

Att: Also, file syslog-ng.conf is provided for download and substitute directly.

```
# Send UFW logs to Fluentd
log {
    source(s_ufw_log);
    filter(f_ufw);
    destination(d_fluentd_ufw);
};
# Source: Snort alert_fast file
source s_snort {
    file("/var/log/snort/snort.alert.fast" follow_freq(1) flags(no-parse));
};
# Log Path: Snort to Fluentd
log {
    source(s_snort);
    destination(d_fluentd_snort);
};
source s_nagios_log {
    file("/usr/local/nagios/var/nagios.log" follow_freq(1) flags(no-parse));
};
filter f_nagios {
    message("nagios");
};
log {
    source(s_nagios_log);
    filter(f_nagios);
    destination(d_fluentd_nagios);
};
@include "/etc/syslog-ng/conf.d/*.conf"
```

Command: `sudo systemctl enable syslog-ng`
`sudo systemctl restart syslog-ng`

8. Fluentd installation and configuration

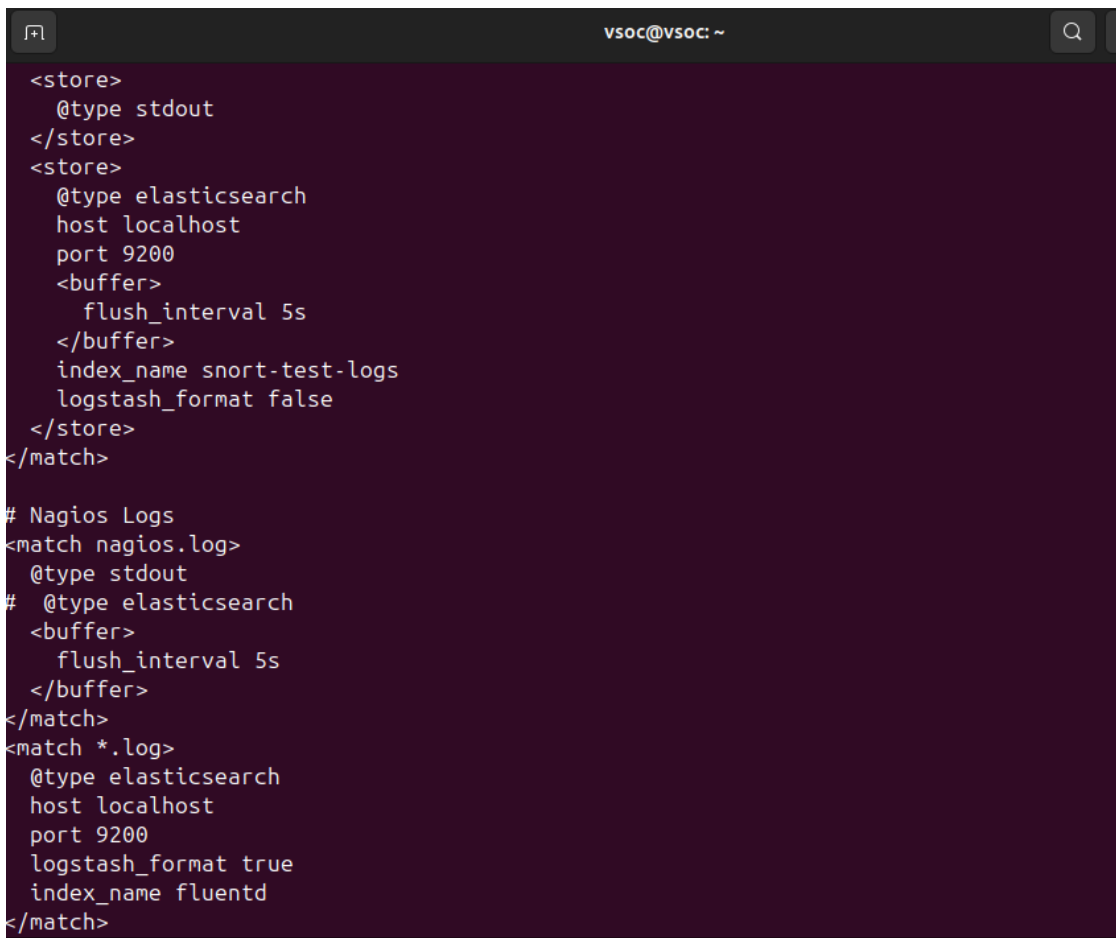
For install Fluentd we are going to use the ansible playbook named as “fluentd.yml”. This file must be located inside of /etc/ansible/ and you need to be in that path.

Command: `sudo ansible-playbook -c local fluentd.yml`

Now you need to configure Fluentd, to get the logs are collected and organized by syslog-ng.

Command: `sudo nano /etc/fluent/fluentd.conf`

Att: Also, file fluentd.conf is provided for download and substitute directly.



```
<store>
  @type stdout
</store>
<store>
  @type elasticsearch
  host localhost
  port 9200
  <buffer>
    flush_interval 5s
  </buffer>
  index_name snort-test-logs
  logstash_format false
</store>
</match>

# Nagios Logs
<match nagios.log>
  @type stdout
# @type elasticsearch
  <buffer>
    flush_interval 5s
  </buffer>
</match>
<match *.log>
  @type elasticsearch
  host localhost
  port 9200
  logstash_format true
  index_name fluentd
</match>
```

Command: `sudo systemctl enable fluentd`
`sudo systemctl restart fluentd`

9. Elasticsearch and Kibana installation and configuration

For install Elasticsearch and Kibana we are going to use the ansible playbook named as “elasticsearch-installation.yml”. This file must be located inside of /etc/ansible/ and you need to be in that path.

Command: `sudo ansible-playbook -c local elasticsearch.yml`

Now you need to configure Elasticsearch and Kibana for get the logs from Fluentd, and be available to visualize this logs.

Command: `sudo nano /etc/elasticsearch/elasticsearch.yml`

Att: Also, file elasticsearch.yml is provided for download and substitute directly.

```
# ===== Elasticsearch Configuration =====
#
# NOTE: Elasticsearch comes with reasonable defaults for most settings.
#       Before you set out to tweak and tune the configuration, make sure you
#       understand what are you trying to accomplish and the consequences.
#
# The primary way of configuring a node is via this file. This template lists
# the most important settings you may want to configure for a production cluster.
#
# Please consult the documentation for further information on configuration options:
# https://www.elastic.co/guide/en/elasticsearch/reference/index.html
#
# ----- Cluster -----
#
# Use a descriptive name for your cluster:
#
cluster.name: elk-vsoc
#
# ----- Node -----
#
# Use a descriptive name for the node:
#
#node.name: node-1
#
# Add custom attributes to the node:
#
```

Command: `sudo nano /etc/kibana/kibana.yml`

Att: Also, file kibana.yml is provided for download and substitute directly.

```
# The number of times to retry temporary migration failures. Increase the setting
# if migrations fail frequently with a message such as `Unable to complete the [...] ste
# 15 attempts, terminating`. Defaults to 15
#migrations.retryAttempts: 15

# ===== Search Autocomplete =====
# Time in milliseconds to wait for autocomplete suggestions from Elasticsearch.
# This value must be a whole number greater than zero. Defaults to 1000ms
#unifiedSearch.autocomplete.valueSuggestions.timeout: 1000

# Maximum number of documents loaded by each shard to generate autocomplete suggestions.
# This value must be a whole number greater than zero. Defaults to 100_000
#unifiedSearch.autocomplete.valueSuggestions.terminateAfter: 100000

# This section was automatically generated during setup.
elasticsearch.hosts: [http://localhost:9200]
elasticsearch.username: kibana_system
elasticsearch.password: '-dyVud-yg3X+vDIFxrDH'
elasticsearch.ssl.certificateAuthorities: [/var/lib/kibana/ca_1745590482890.crt]
xpack.fleet.outputs: [{id: fleet-default-output, name: default, is_default: true, is_def
ring: true, type: elasticsearch, hosts: [http://localhost:9200], ca_trusted_fingerprint:
4b6c44d4df13dd8ce050d641a1be8c0ed7f2833fe2d5b158e133ff}]
```

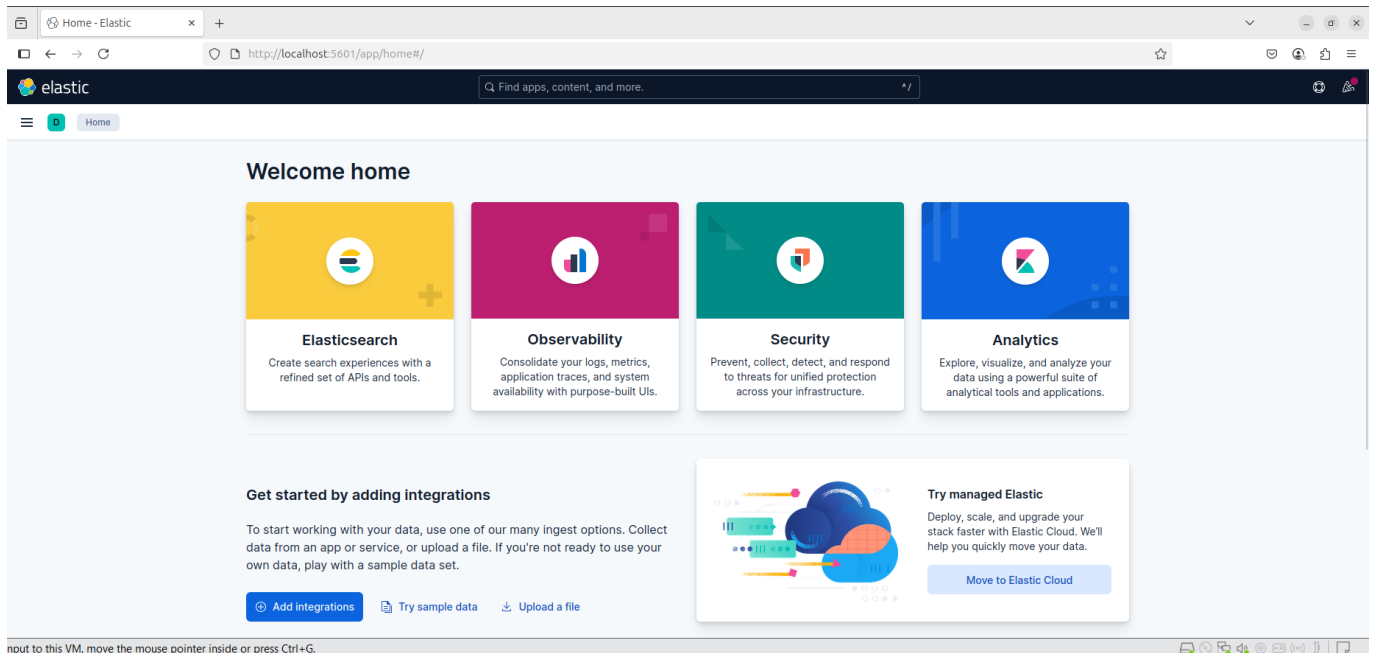
Command: `sudo systemctl enable elasticsearch`
`sudo systemctl restart elasticsearch`
`sudo systemctl enable kibana`
`sudo systemctl restart kibana`

Now we can access to Firefox and go to <http://localhost:5601> , the username is elastic and the password will need to be reset.

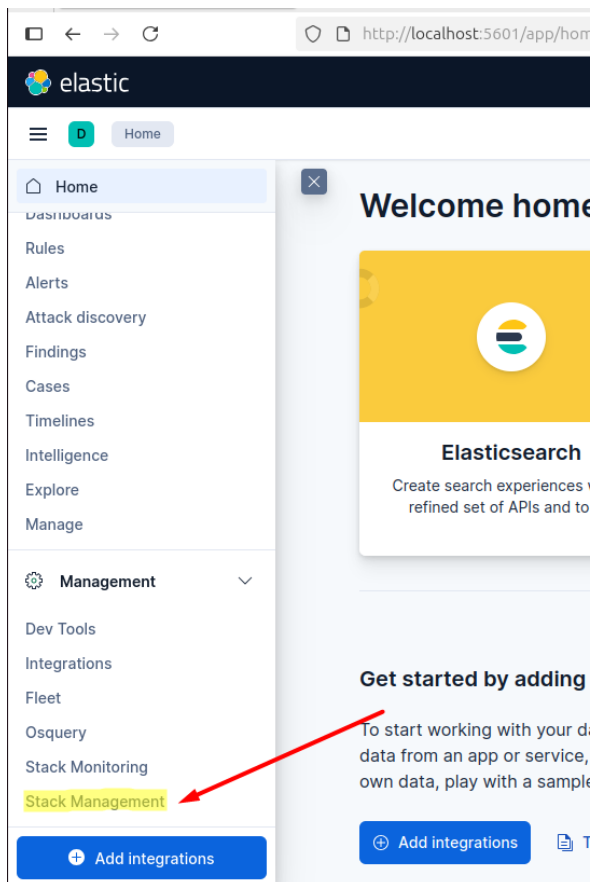
Command: `sudo /usr/share/elasticsearch/bin/elasticsearch-reset-password -u elastic`

This command will prompt you the password for the user, you will need to use.
After that you will have a Access Token from Kibana.

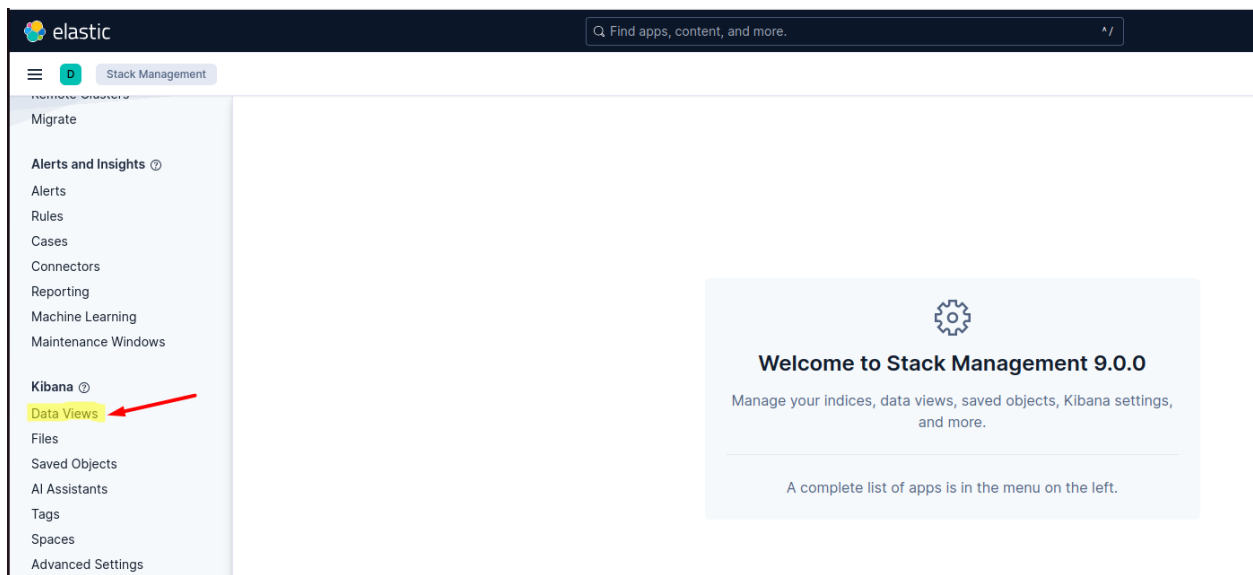
Command: `sudo /usr/share/elasticsearch/bin/elasticsearch-create-enrollment-token -s kibana`



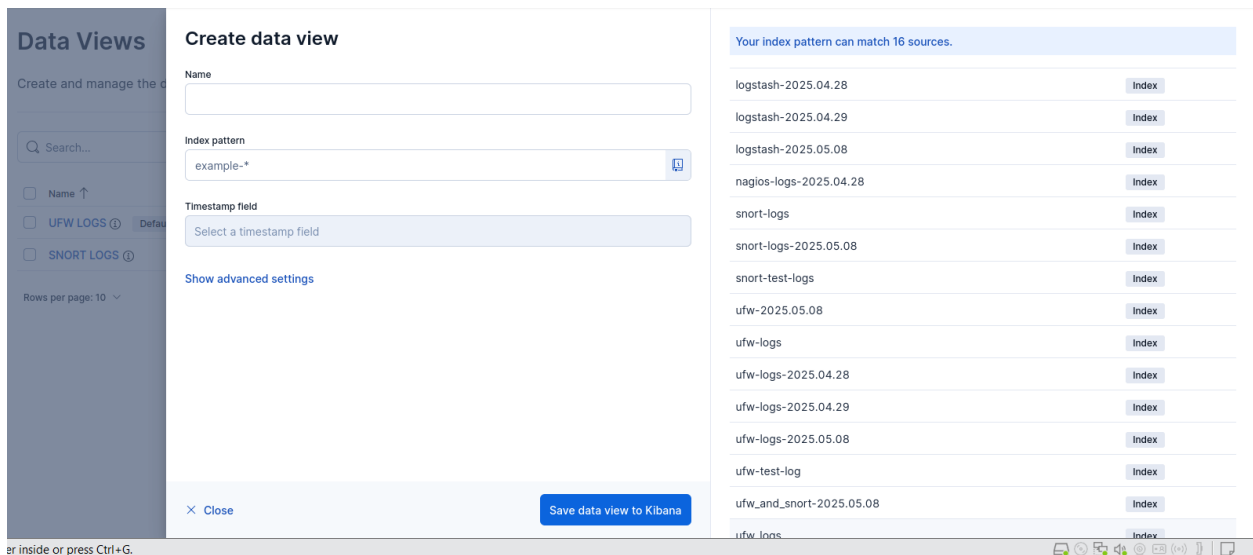
Now we can configure the dashboard in ELK, we need to go to Management → Stack Management.



Then Kibana → Data Views.



Now you can create a new Data View for Kibana, on the right side will show the options of logs is detecting your Kibana are being sent to Elasticsearch. So, you must indicate the index pattern for example: snort-* (This will give you all the snort logs).



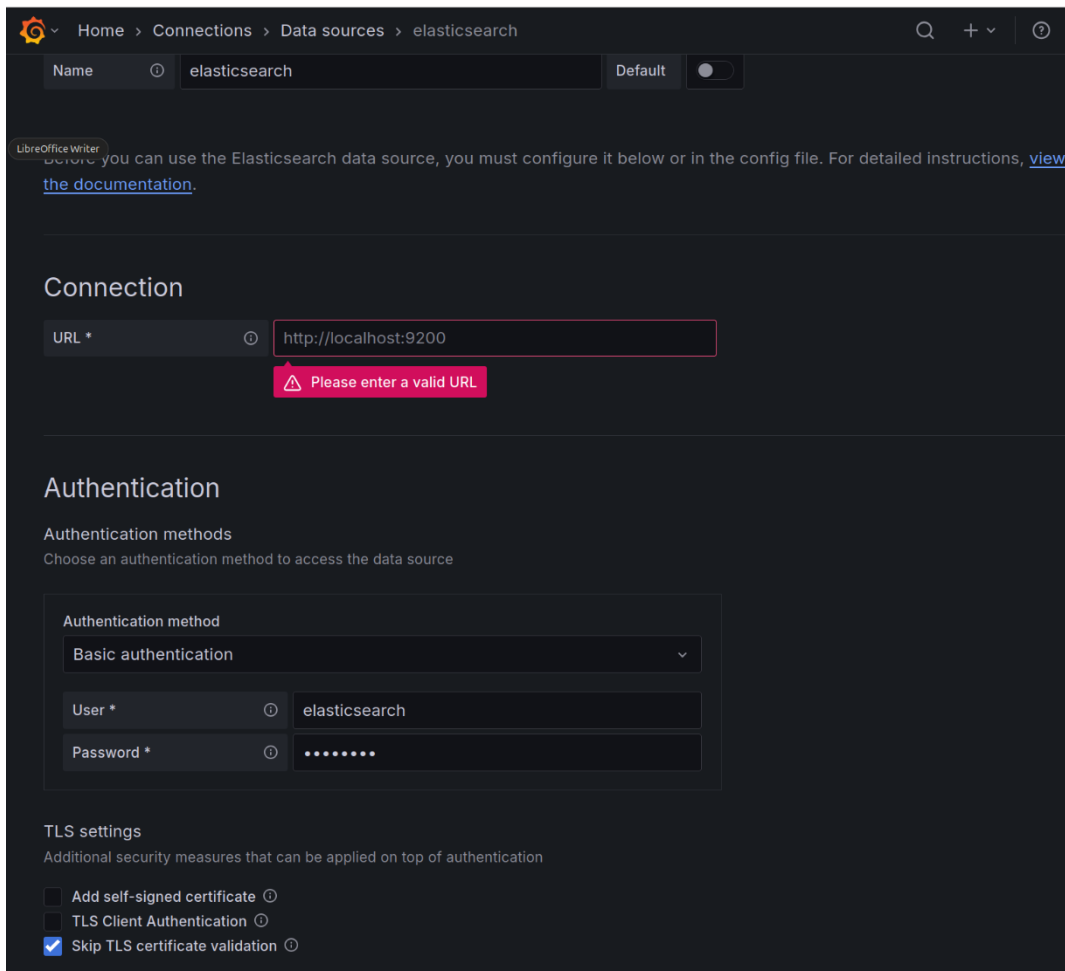
10. Grafana installation and configuration

For install Grafana we are going to use the ansible playbook named as “grafana.yml”. This file must be located inside of /etc/ansible/ and you need to be in that path.

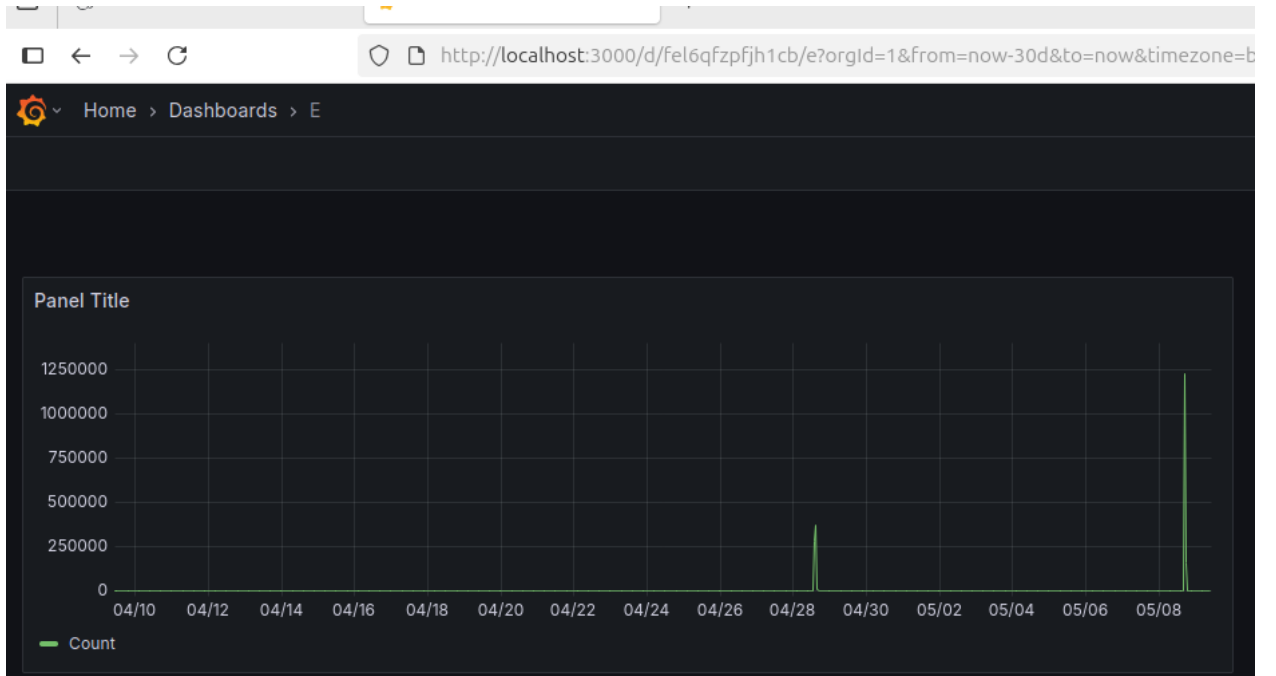
Command: `sudo ansible-playbook -c local grafana.yml`

After you finish your Grafana installation, we need to access to: <http://localhost:3000> , user: admin and password: admin. Will ask to you of changing the password put for example: vsoc2025.

Now go to Home → Connections → Add new connection → Search for elasticsearch



The screenshot shows the Grafana web interface for configuring a new data source. The breadcrumb navigation at the top reads: Home > Connections > Data sources > elasticsearch. Below this, there is a table with one row for the 'elasticsearch' data source, with a 'Default' toggle switch. A message from LibreOffice Writer is visible: "LibreOffice Writer: Below you can use the Elasticsearch data source, you must configure it below or in the config file. For detailed instructions, [view the documentation](#)." The 'Connection' section contains a 'URL *' field with the value 'http://localhost:9200'. A red error message box below the URL field says: "Please enter a valid URL." The 'Authentication' section has a heading 'Authentication methods' and a sub-heading 'Choose an authentication method to access the data source'. Below this is a form with 'Authentication method' set to 'Basic authentication'. It also has 'User *' set to 'elasticsearch' and 'Password *' masked with dots. The 'TLS settings' section at the bottom has the sub-heading 'Additional security measures that can be applied on top of authentication' and three checkboxes: 'Add self-signed certificate' (unchecked), 'TLS Client Authentication' (unchecked), and 'Skip TLS certificate validation' (checked).



11. Nagios installation and configuration

For install Nagios we are going to use the ansible playbook named as “nagios.yml”. This file must be located inside of /etc/ansible/ and you need to be in that path.

Command: `sudo ansible-playbook -c local nagios.yml`

After you finish your Nagios installation you can access to <http://localhost/nagios/> there will prompt you for indicate username: nagiosadmin, and password: nagiosadmin123 (Change it after log-in).

The screenshot displays the Nagios web interface. At the top, a browser tab shows 'E - Dashboards - Grafana'. The address bar indicates the URL 'http://localhost/nagios/'. A login modal is open, prompting for a 'Username' (nagiosadmin) and a 'Password' (masked with dots). Below the login prompt, the main Nagios dashboard is visible. The left sidebar contains a navigation menu with sections like 'General', 'Current Status', 'Tactical Overview', 'Problems', 'Quick Search', and 'Reports'. The main content area is titled 'Current Event Log' and shows a list of events, including service alerts and notifications. The top right corner features a 'Log File Navigation' section with a timeline view showing events for 'May 09, 2025 10:00' and 'May 09, 2025 09:00'.

12. MITRE CALDERA installation and configuration

For install MITRE CALDERA will be doing it without ansible due the python environment for installation can have some issues for do the job.

First is needed to install the pre-needed packages for the installation.

Command: `sudo apt install git python3 python3-pip python3-venv -`

Then,

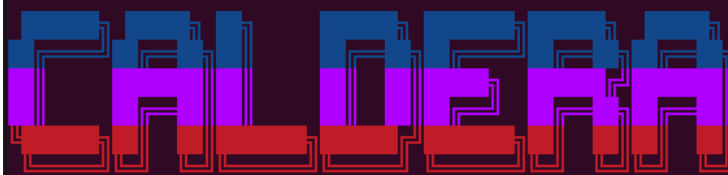
Command: `cd /opt`
`sudo git clone https://github.com/mitre/caldera.git --recursive`
`cd caldera`

Now we can activate the Python environment for make the installation of the software.

Command: `python3 -m venv venv`
`source venv/bin/activate`
`pip install -r requirements.txt`
`python3 server.py --build`

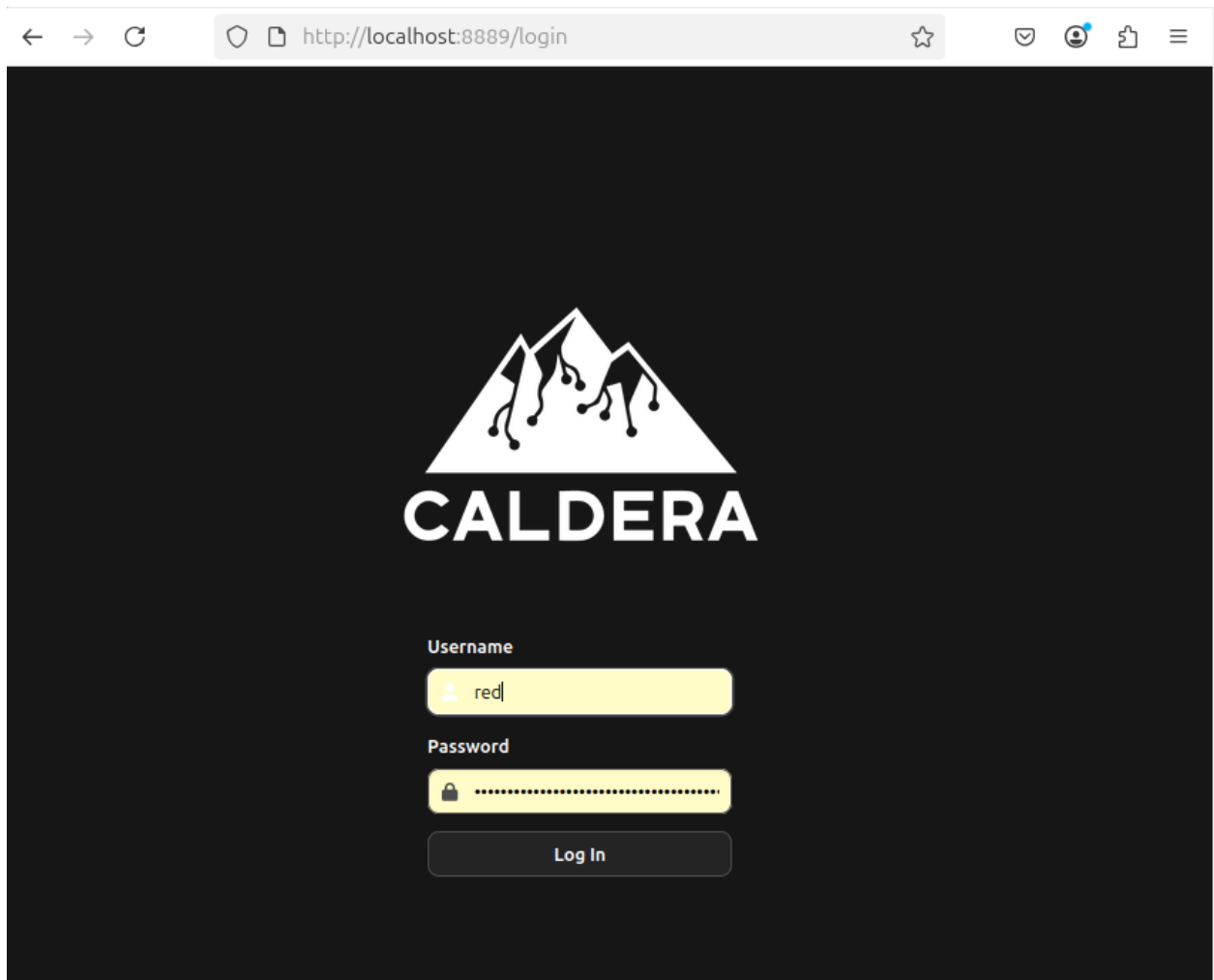
```
vsoc@vsoc:~$ cd caldera/
vsoc@vsoc:~/caldera$ python3 -m venv venv
vsoc@vsoc:~/caldera$ source venv/bin/activate
(venv) vsoc@vsoc:~/caldera$ python3 server.py --build
2025-05-09 11:01:39 INFO      Using main config from conf/local.yml      server.py:228
2025-05-09 11:01:40 INFO      Building VueJS front-end.      server.py:265
⋮
```

```
2025-05-09 11:02:52 INFO      Docs built successfully.      hook.py:60
INFO      All systems ready.      server.py:104
```




Now you can access to <http://localhost:8888> or <http://localhost:8889>.
There are two different log-in red team or blue team, you can find the credentials of these into /caldera/conf/local.yml

```
users:  
  blue:  
    blue: yE3-kuoTVyztCSz9S10eREuH-EdXN6bPFFKZsm9rU4g  
  red:  
    red: uz7m6F33BaLq3jc4f085u4UHfyqHD6Q-VGtW2x43gck  
vsoc@vsoc:~/caldera/conf$
```



← → ↻ http://localhost:8889/login ☆


CALDERA

Username

Password

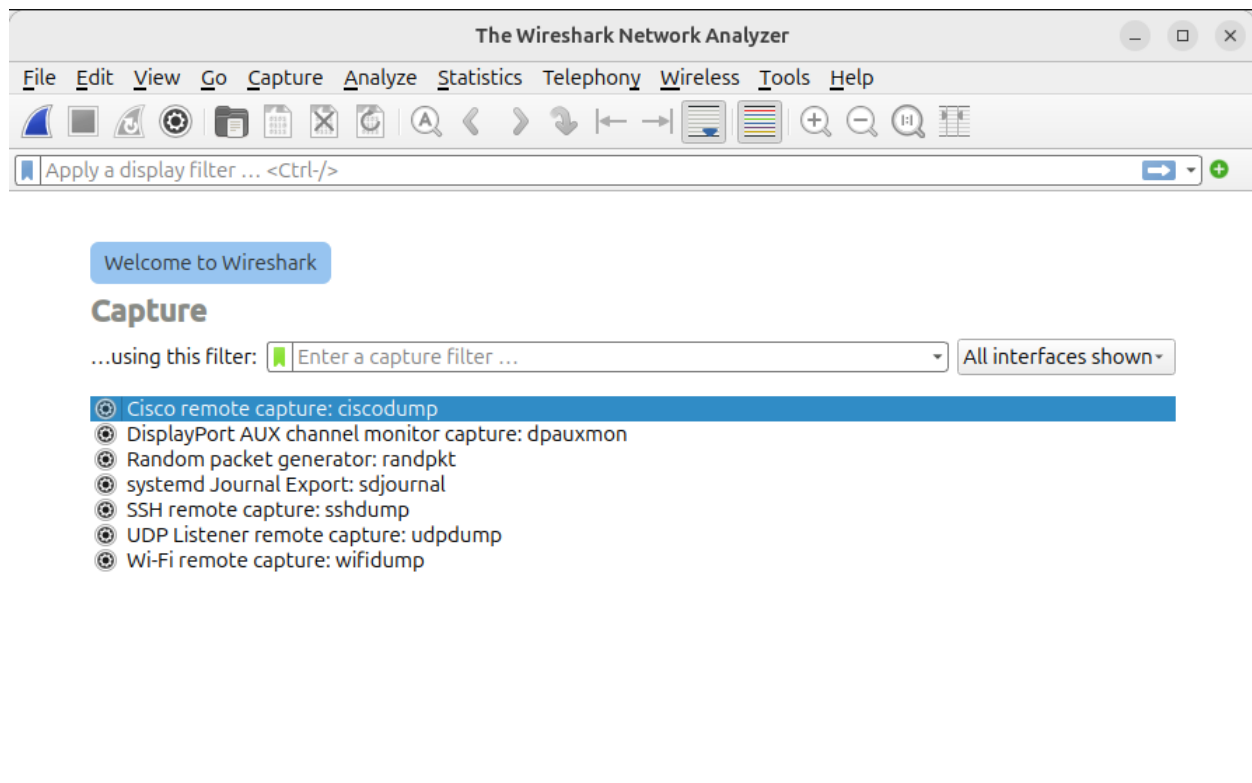
Log In

13. Wireshark installation

Command: `sudo apt install wireshark`

Then execute the next command for opening the application of Wireshark.

Command: `wireshark`



14. Links and Resources

1. <https://www.hostmycode.in/tutorials/install-and-configure-nagios-on-ubuntu>
2. https://www.manageengine.com/products/eventlog/sem/glp/syslog-server.html?camid=19529450106&adgid=175568250399&kwd=syslog%20server%20linux&matchtype=p&adid=738673661453&network=g&adposition=&loc=9008042&placement=&target=&device=c&gad_source=1&gad_campaignid=19529450106&gbraid=0AAAAAChAr7Z_crrrCCamor6xseCCh_2vF&gclid=CjwKCAjwz_bABhAGEiwAm-P8YRoprj1afTVobUEpFnVrPAzekmCQQv7GjT0ul1zR6qqPFtEosivgDRoCH4kQAvD_BwE
3. <https://www.syslog-ng.com/community/b/blog/posts/installing-the-latest-syslog-ng-on-ubuntu-and-other-deb-distributions>
4. <https://www.zenarmor.com/docs/linux-tutorials/how-to-install-and-configure-snort-on-ubuntu-linux#5-running-snort-as-a-service>
5. <https://docs.netgate.com/pfsense/en/latest/packages/snort/setup.html>
6. <https://www.digitalocean.com/community/tutorials/how-to-install-and-configure-elasticsearch-on-ubuntu-22-04>
7. <https://grafana.com/docs/grafana/latest/setup-grafana/installation/debian/>
8. <https://www.digitalocean.com/community/tutorials/how-to-set-up-a-firewall-with-ufw-on-ubuntu>
9. <https://docs.fluentd.org/installation>
10. <https://medium.com/@salim.y.salimov/installing-mitre-caldera-4-on-ubuntu-vm-fac970825352>
11. https://docs.ansible.com/ansible/latest/installation_guide/installation_distros.html