Fun with FreeNAS © on Virtualbox ™

For Work and Play

Chin-Fah, HEOH
chinfah@katanalogic.com

Fun with FreeNAS © on Virtualbox ™

For Work and Play

Chin-Fah, HEOH
chinfah@katanalogic.com

TABLE OF CONTENT

CHAPTER 0 (INTRODUCTION)

FreeNAS is the world's most downloaded Storage Operating System. It is open source and it is free to use under the BSD License. It can be downloaded from https://www.freenas.org/download.

Virtualbox by Oracle, is a free and open source Type 2 hosted x86 hypervisor software and can be downloaded from https://www.virtualbox.org/wiki/Downloads. It runs on Windows, Linux, MacOS and Solaris.

The motivation

In my day job, I built FreeNAS solutions with x86 servers, often using enterprise components - Xeon CPUs, ECC RAM, Enterprise SSDs and HDDs, 10Gigabit Ethernet. Often, I do integration work and testing for my customers.

Virtualbox and the FreeNAS downloaded ISO have been very competent in my testing and integration development. So I have decided to document most of my work in this book and hoping to share this experience with the FreeNAS community.

The approach

You can read about the features of FreeNAS in its documentation found here at https://www.ixsystems.com/documentation/freenas/11.2-U5/freenas.html. The document is helpful to understand FreeNAS well but it does not describe most real world situations well, especially when it comes to integration with clients and applications.

The book is like a cookbook, full of step-by-step screenshots of getting FreeNAS and Virtualbox running and working together in NAS and SAN networks. In addition, I also include some real world experience which cannot be duplicated in the Virtualbox environment and hopefully, this is helpful to all.

The versions

In the development of this book, I am using Virtualbox version 6.0.10 r132072 And FreeNAS 11.2-U5. I use Windows 7.0 as my client, Windows Server 2019 for Active Directory and iSCSI, and CentOS 7.6 build 1810 as NFS client.

Continuous development

There is much to do to develop the content and the chapters of this book. It will be continuous and on-going work and I do hope to get as much done as possible.

Trademarks, Copyrights, Registrations and Acknowledgments

FreeNAS® is © 2011-2019 iXsystems

FreeNAS® and the FreeNAS® logo are registered trademarks of iXsystems

FreeBSD® is a registered trademark of the FreeBSD Foundation

Active Directory® is a registered trademark or trademark of Microsoft Corporation in the United States and/or other countries.

Apple, Mac and Mac OS are trademarks of Apple Inc., registered in the U.S. and other countries.

FreeBSD® and the FreeBSD® logo are registered trademarks of the FreeBSD Foundation®.

Intel, the Intel logo, Pentium Inside, Pentium and Xeon are trademarks of Intel Corporation in the U.S. and/or other countries.

Linux® is a registered trademark of Linus Torvalds.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates.

UNIX® is a registered trademark of The Open Group.

VirtualBox® is a registered trademark of Oracle.

VMware® is a registered trademark of VMware, Inc.

Windows® is a registered trademark of Microsoft Corporation in the United States and other countries.

CHAPTER 1 (VIRTUALBOX SETUP FOR FREENAS)

Virtualbox is a Type 2 hypervisor application which runs on Windows, Linux and MacOS. Virtualbox allows the creation of multiple virtual machines (VM) with guest OSes. In this book, we will use Virtualbox to prepare the storage environment for FreeNAS for iSCSI SAN, NAS (both SMB and NFS) as well as Object Storage.

Phase 1: (Preparation)
We start by preparing the Virtualbox VM and its requirement for FreeNAS.

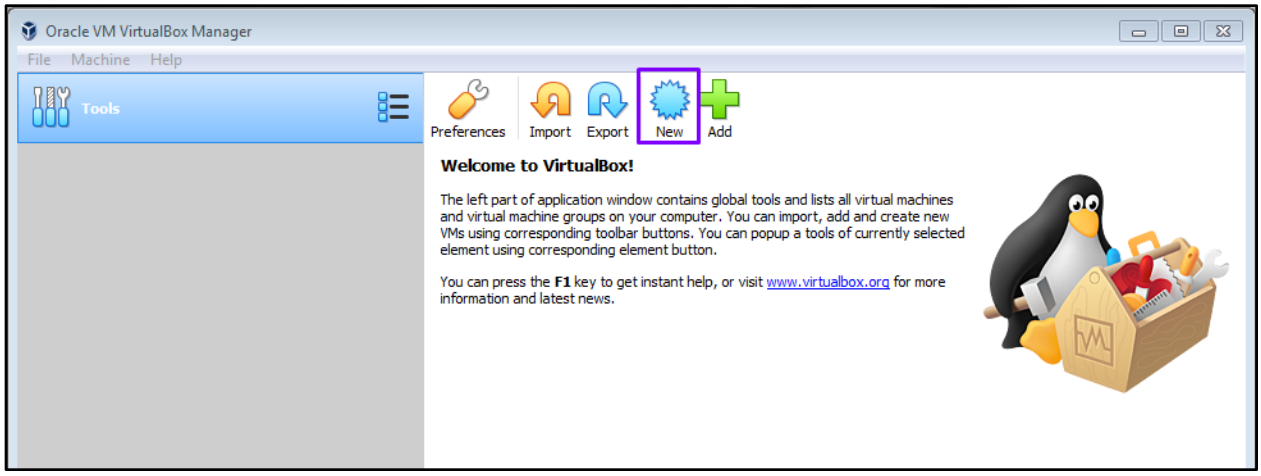


Fig 1.1: This is the Virtualbox interface

Click on 'New' as shown in Fig 1.1.

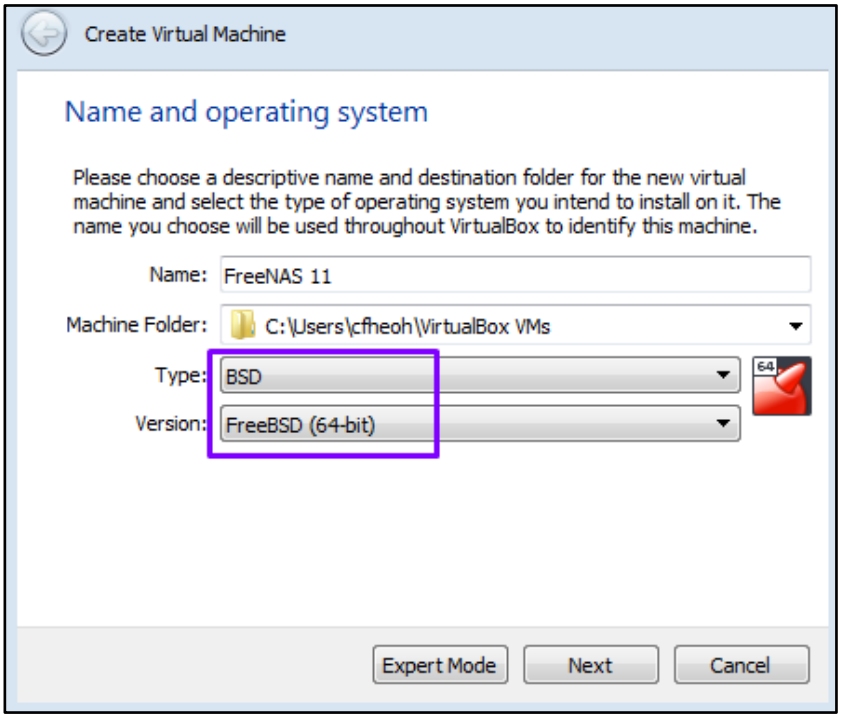


Fig 1.2: Set the name of the VM, Type and Version

In Fig 1.2, set the name of the Virtualbox VM. We use the following:

Name: FreeNAS 11
Type: BSD
Version: FreeBSD (64-bit)

If you can only find FreeBSD (32-bit) but not FreeBSD (64-bit) in the Version in Fig 1.2, you need to enable the Virtualization feature, Intel VT or AMD-V, on your BIOS. Shutdown your OS and boot to BIOS setup. This is a hardware feature of the computer that allows Intel or AMD processors to have virtualization extension. After this feature is enabled, the FreeBSD (64-bit) can be found in Virtualbox and selected.

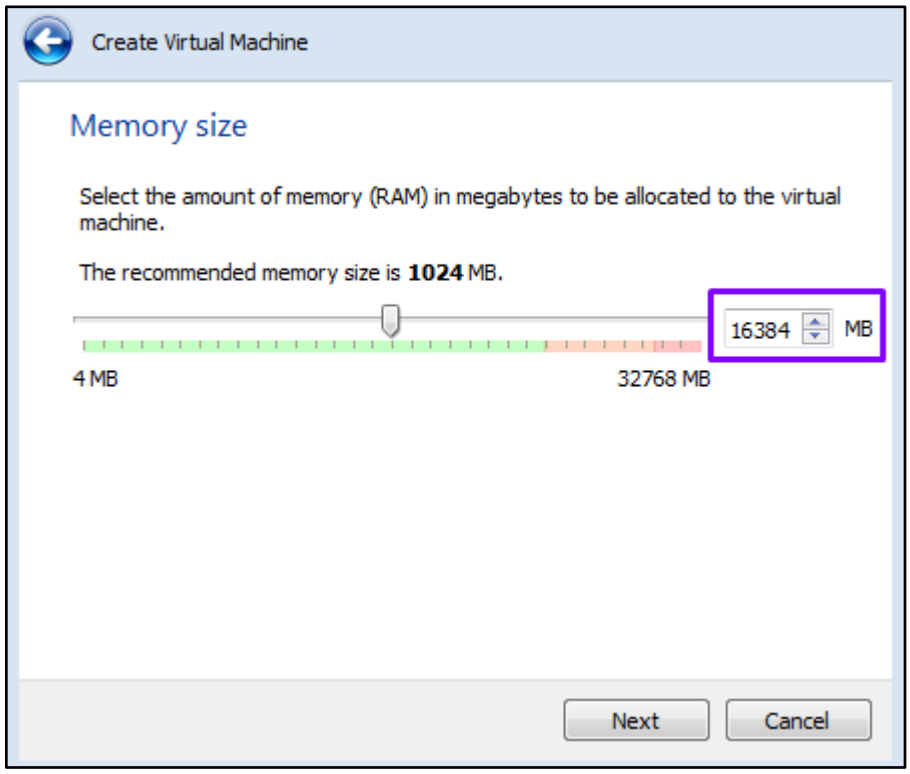


Fig 1.3: Set the memory size for the FreeNAS VM

Memory size 16 x 1024 = 16384 MB. The minimum RAM size to run is 8192MB or 8GB. It can run with 6GB but a message will pop up mentioning insufficient memory to install. This is OK and FreeNAS will run with 6GB without much issue, as long as there is no high performance requirement with the installation or its operations.

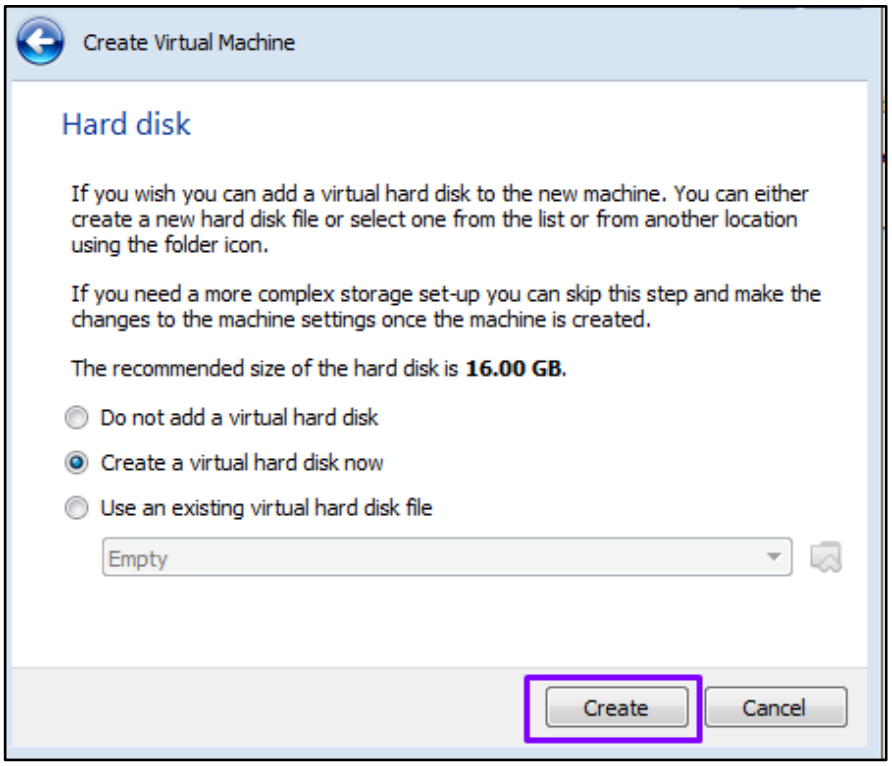


Fig 1.4: Create the hard disk drive for FreeNAS

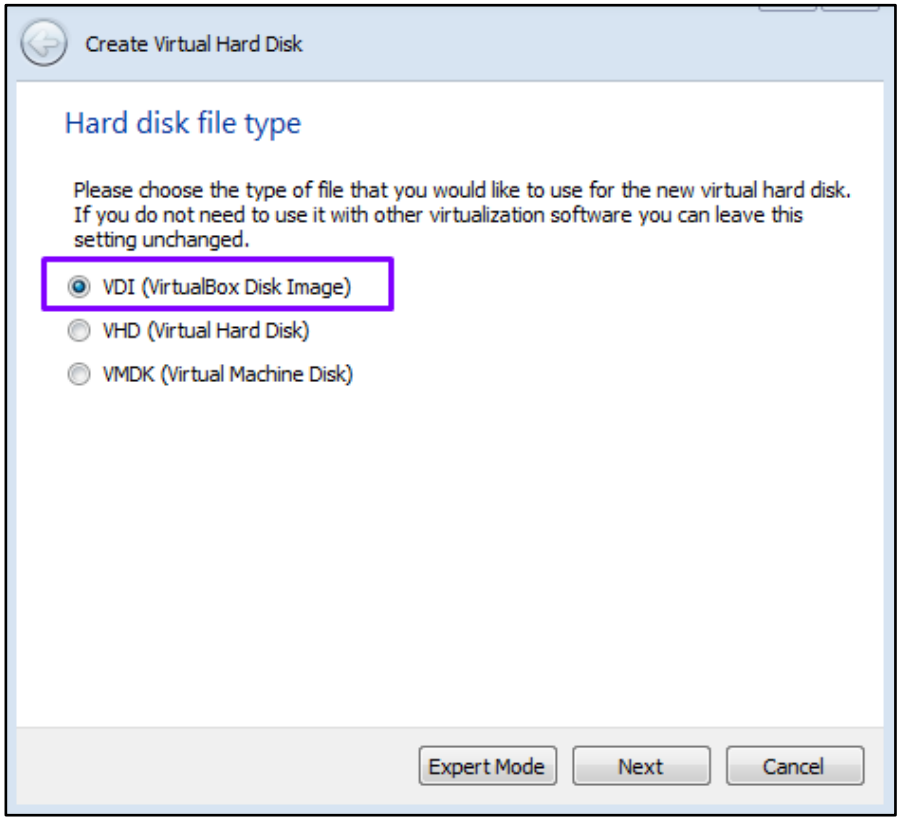


Fig 1.5: VDI is the default disk image of Virtualbox

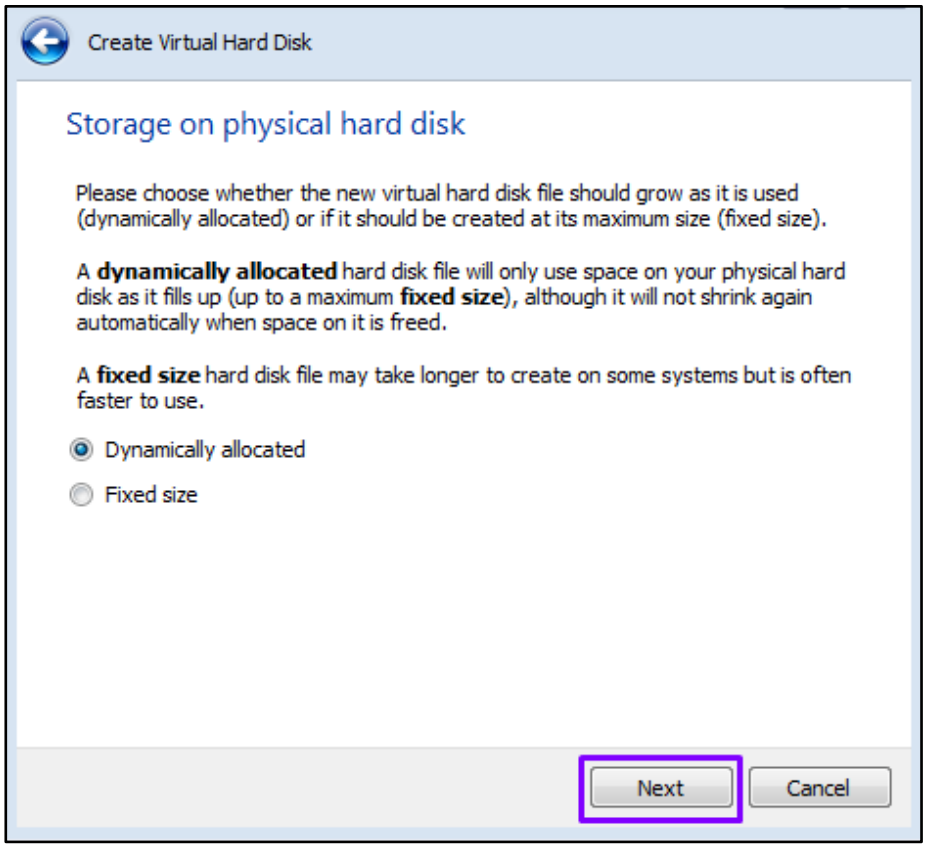


Fig 1.6: Dynamic allocation is the default

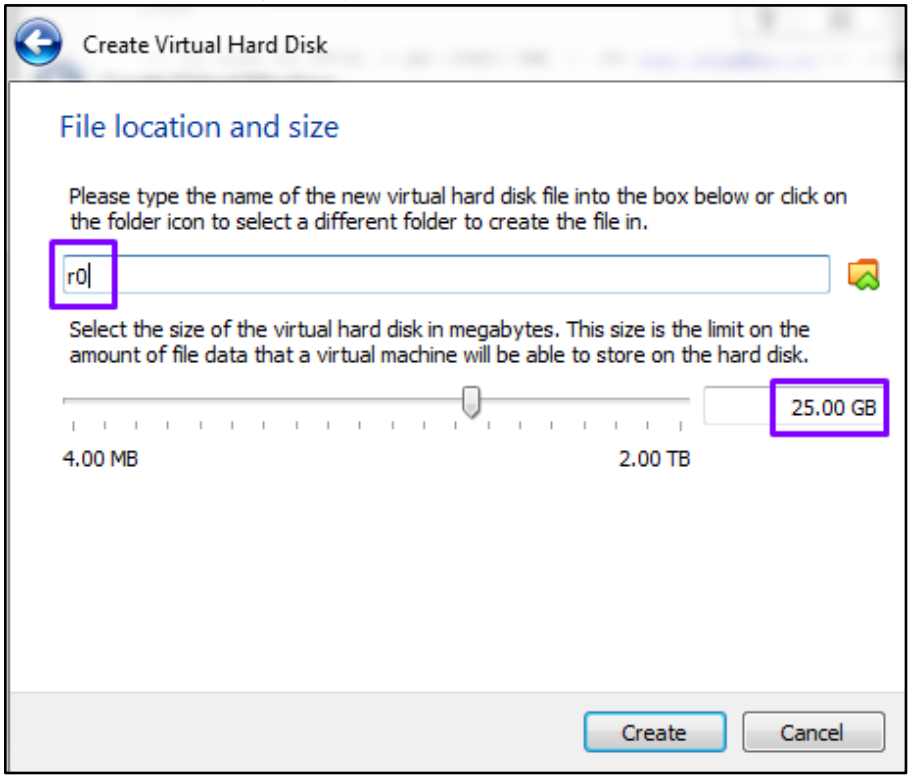


Fig 1.7: Name the disk r0 for the root disk

In our Virtualbox setup, we name the root disk as ‘r0’ and use 25GB instead of the default 16GB. This root disk is where the FreeNAS OS will be installed. In a real setting, we recommend the root disk to be a read-intensive or mixed workload SSD (solid state drive).

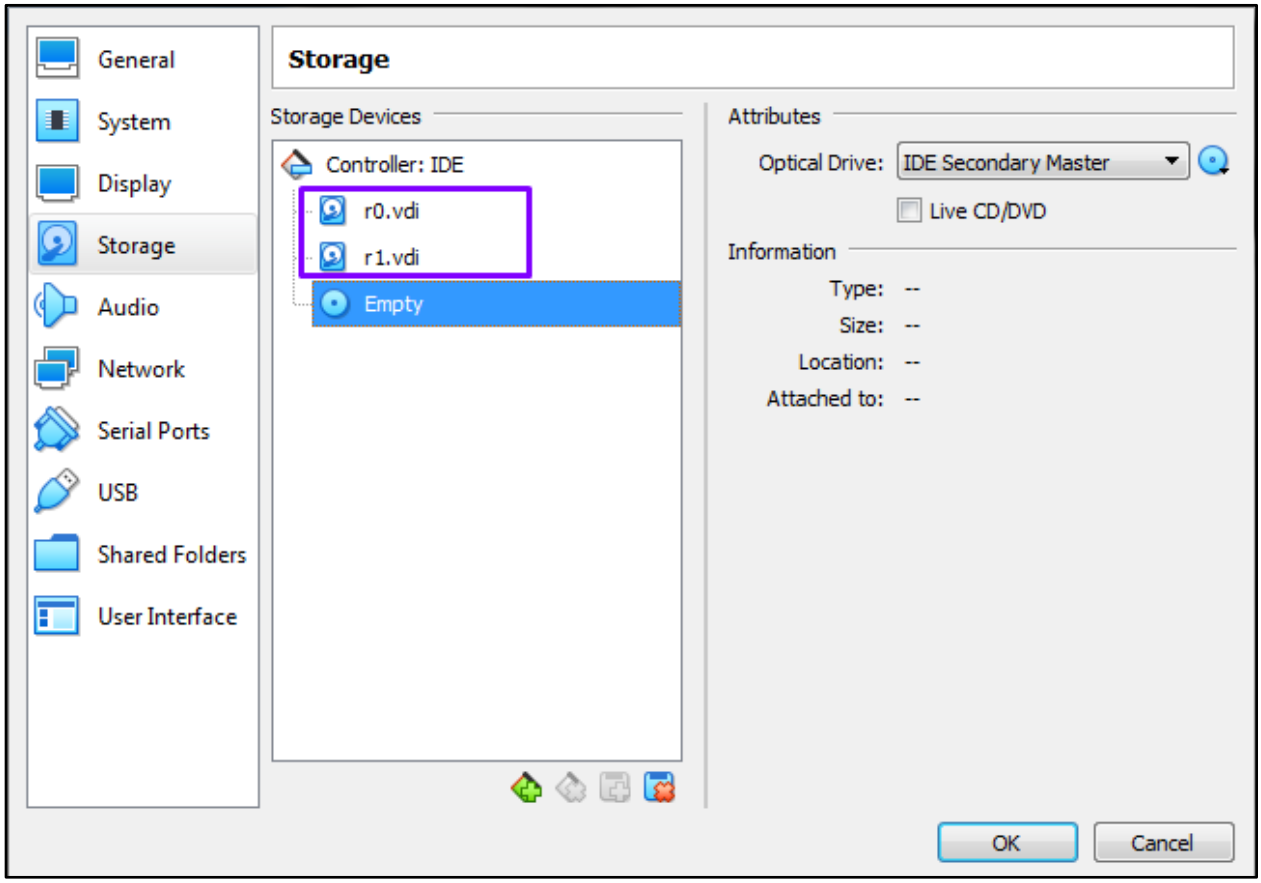


Fig 1.8: 2 OS hard disk drives created for resiliency reasons.

We encourage the creation of a second hard disk drive, named as ‘r1’. During installation, we will mirror ‘r0’ and ‘r1’ in a RAID-1 configuration. This is to provide high resiliency should of the OS disk fails. Again, in a real setting, this should be 2 SSDs mirrored.

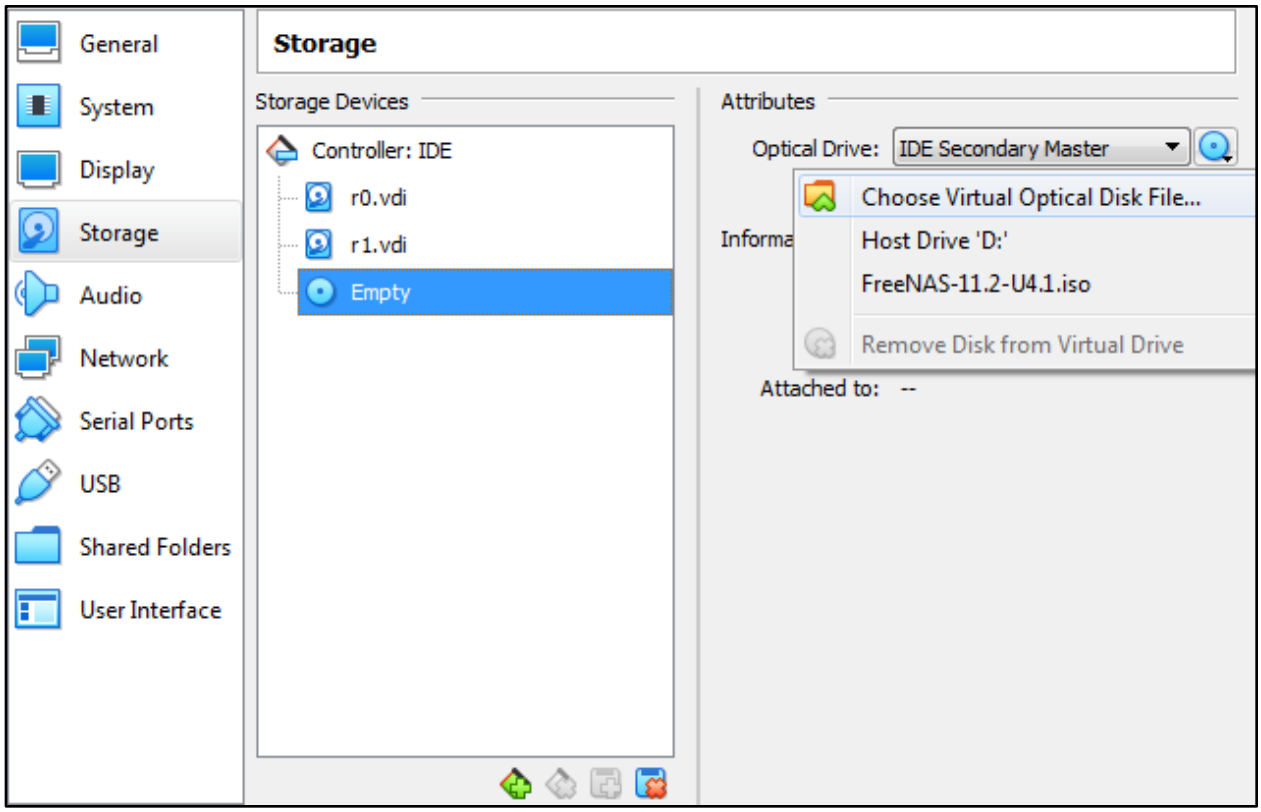


Fig 1.9: Using a virtual CDROM to install FreeNAS ISO

In Fig 1.9, we load the virtual CDROM with the downloaded FreeNAS-11.2U4.1 ISO

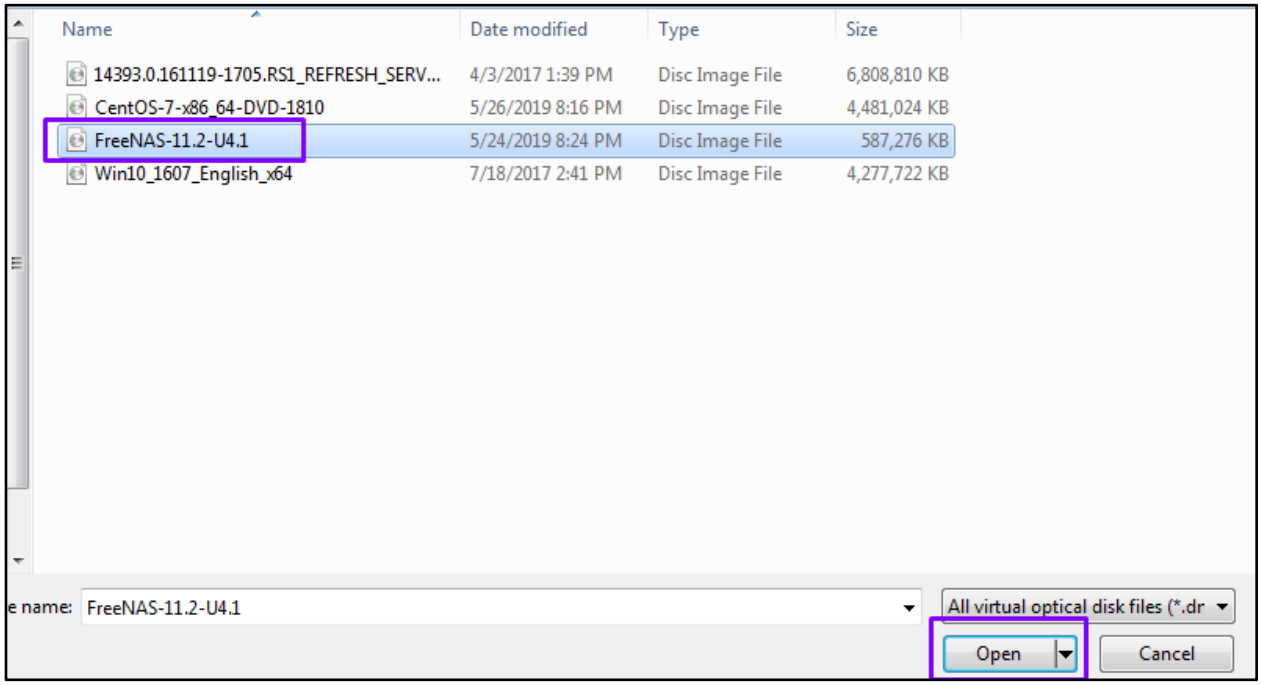


Fig 1.10: Select the FreeNAS ISO from the pop-up window

Phase 2 (Settings)

Click on ‘Settings’ to set up the storage hard disk drives and networking for the FreeNAS VM as in Fig 1.11.

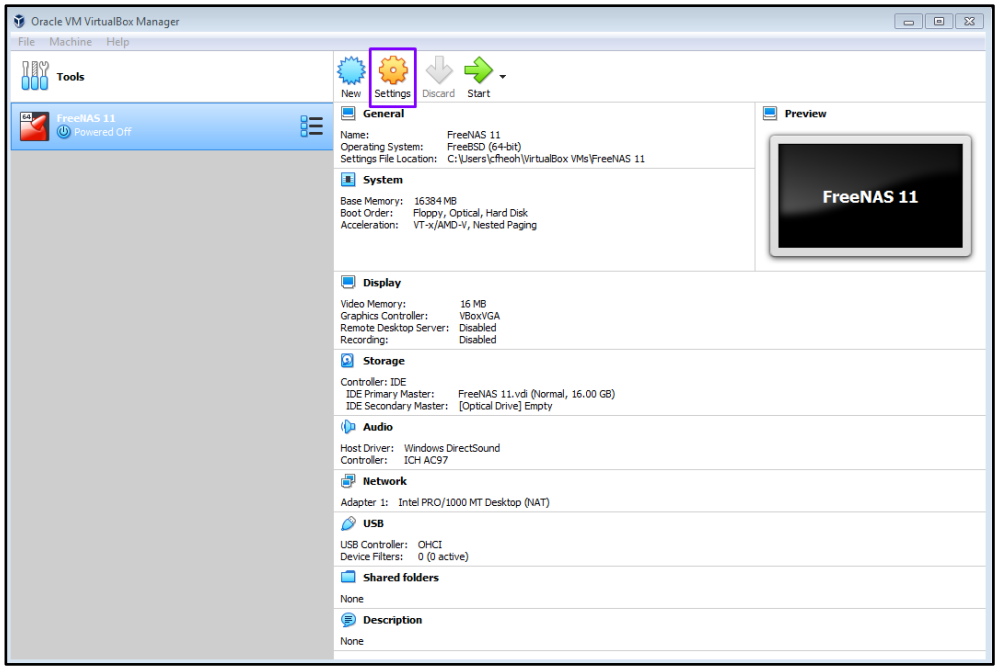


Fig 1.11: Click ‘Setting’ to begin Phase 2 of FreeNAS VM setup

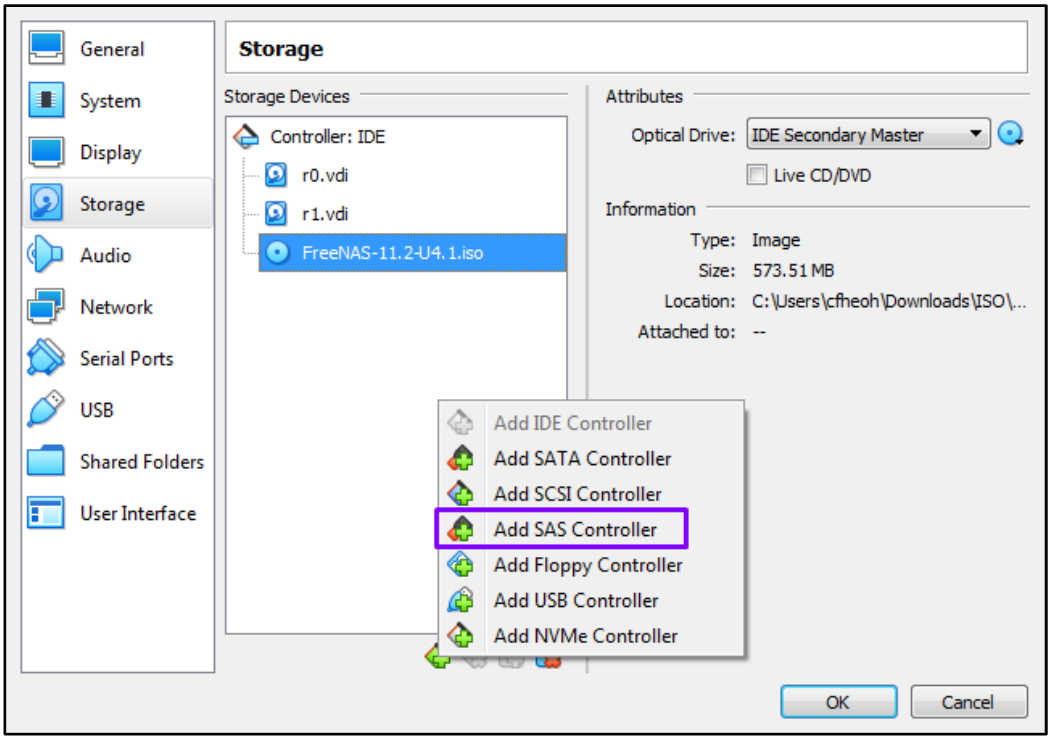


Fig 1.12: Create a new SAS controller for the hard disk drives

In Fig 1.12, we create a new SAS controller for the hard disk drives. In Virtualbox, you can create SATA, SCSI, SAS, NVMe Storage Controller for HDDs (hard disk drives) and SSDs (solid state drives). Each has its performance and resiliency features. In a real setting, it is best to have at least 2 storage controllers of the chosen interface (SAS, SATA, SCSI or NVMe). During creation of a VDEV (virtual device), it is best to select drives from both controllers for resiliency and performance reasons.

More details of VDEV later when we go deeper on the ZFS filesystem.

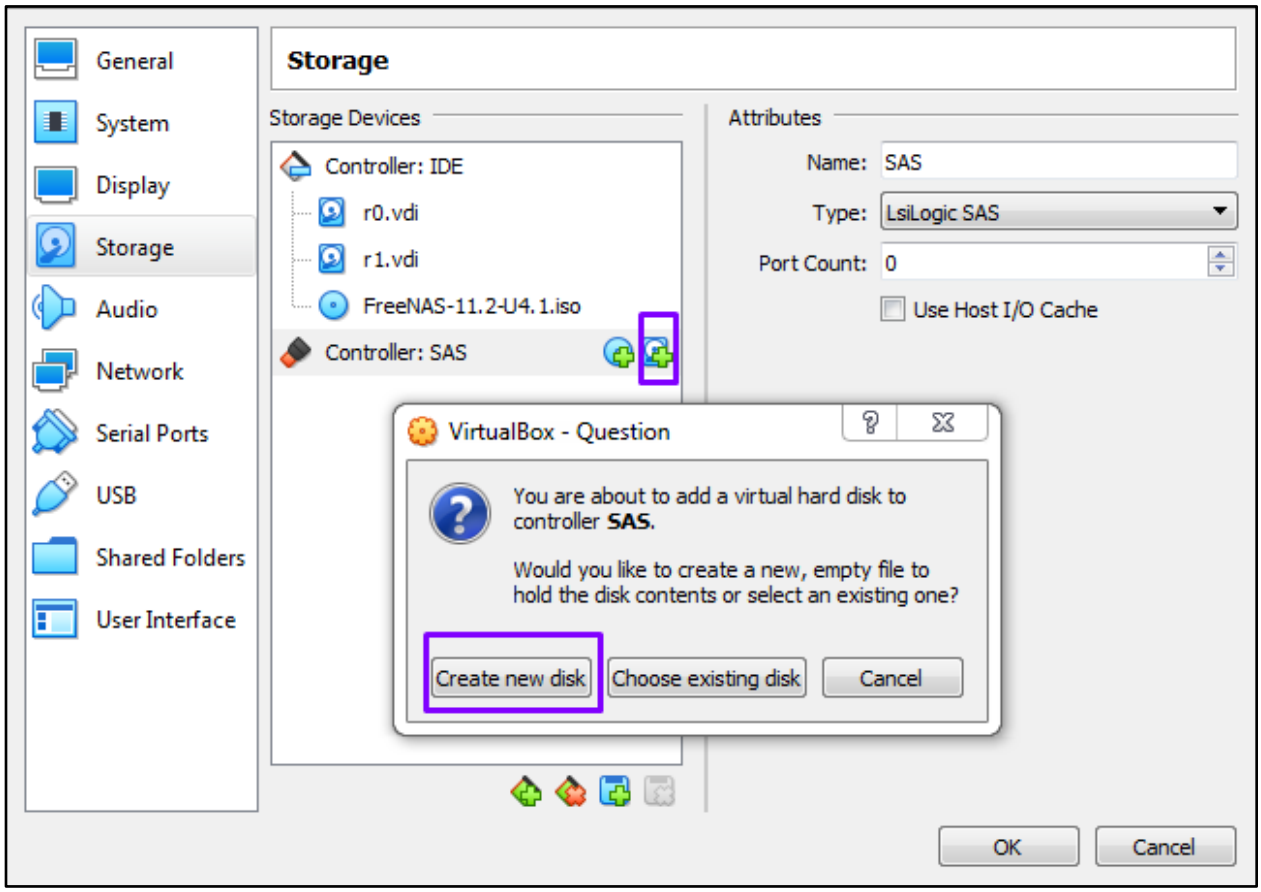


Fig 1.13: Create new disk on the new SAS controller

In this book, we have created 12 SAS HDDs named ‘d0’ to ‘d11’. These HDDs will be used for VDEVs and zpool.

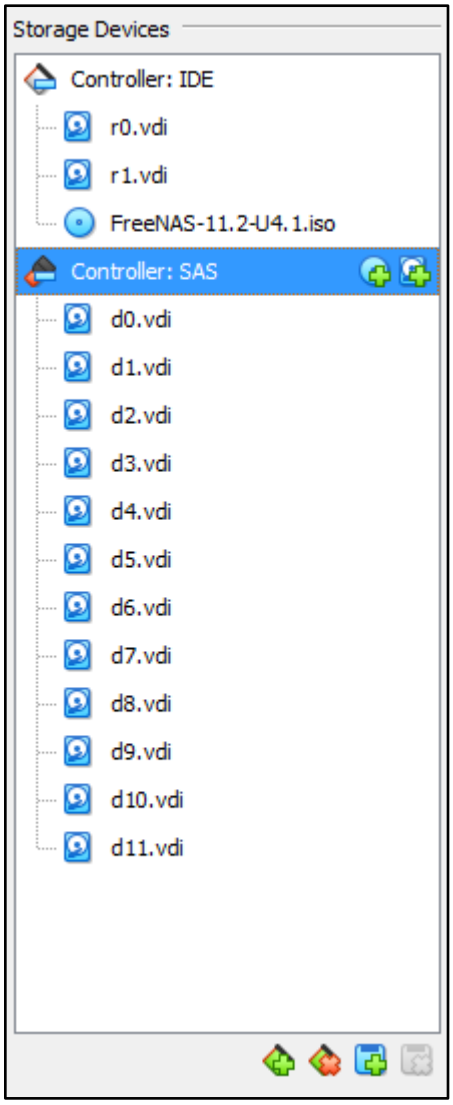


Fig 1.14: Storage configuration for the FreeNAS VM

In Fig 1.14, we have a total of 2 x SSDs which are ‘r0’ and ‘r1’. This is from the IDE controller together with the virtual CD-ROM. The SAS controller has 12 x HDDs, which are ‘d0’ to ‘d11’.

We finalize the FreeNAS VM configuration with the network configuration. In Virtualbox, there are several network modes.

- Not attached
- NAT
- NAT Network
- Bridged Adapter
- Internal Network
- Host-only Adapter
- Generic Driver

More details of each network mode are found at https://www.virtualbox.org/manual/ch06.html

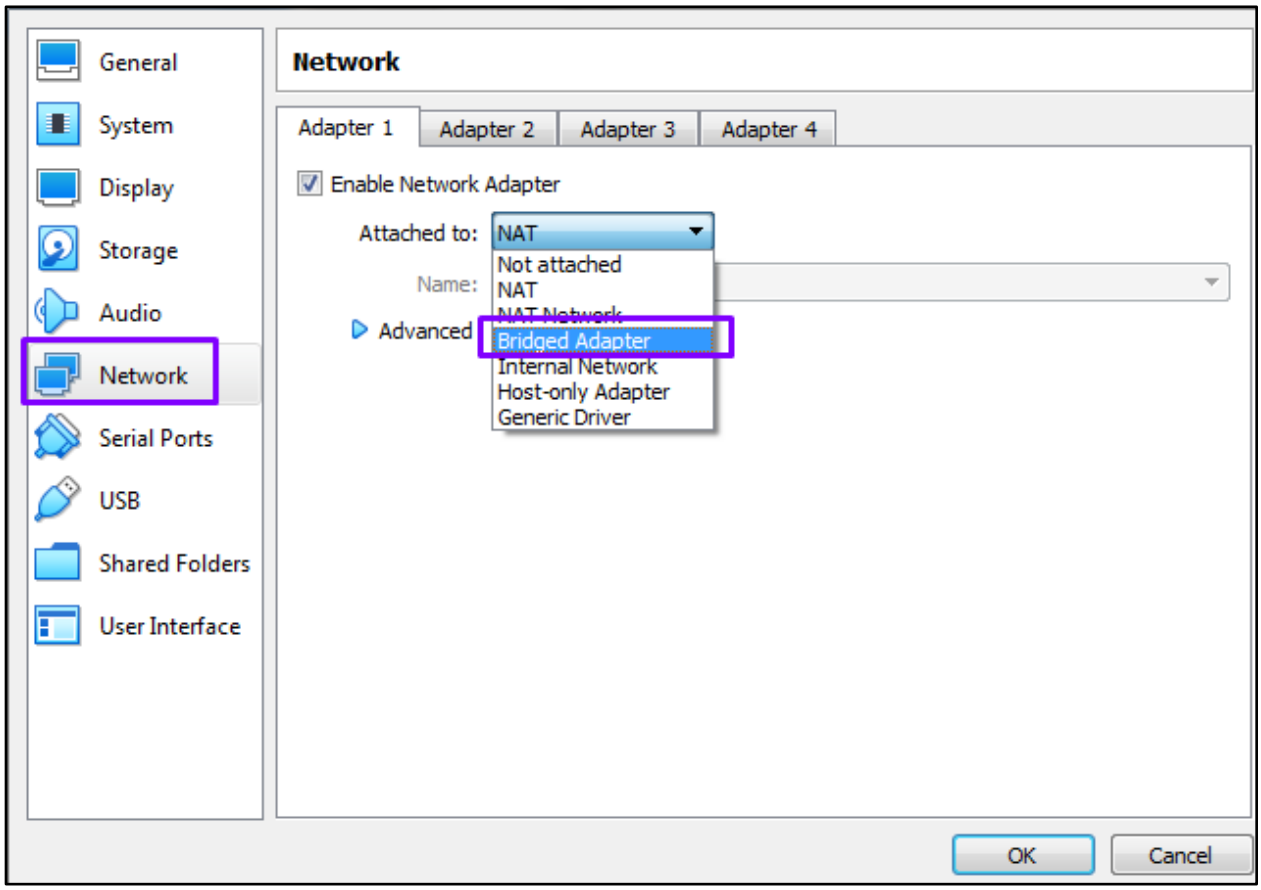


Fig 1.15: Network mode Bridged Adapter

In this book, we choose 'Bridge Adapter' network mode to allow both the FreeNAS to network with the host OS of the Virtualbox and the FreeNAS VM. This network mode makes both the host OS and the FreeNAS VM as 2 separate network entities in a single subnet and is important for both client and server communications.

CHAPTER 2 (FREENAS INSTALLATION)

Phase 1 FreeNAS installation

This chapter begins the FreeNAS VM installation. FreeNAS installation are in 2 phases:

- With the CD-ROM (Phase 1) to install the FreeNAS OS
- Without the CD-ROM to install the remainder of the OS software and drivers

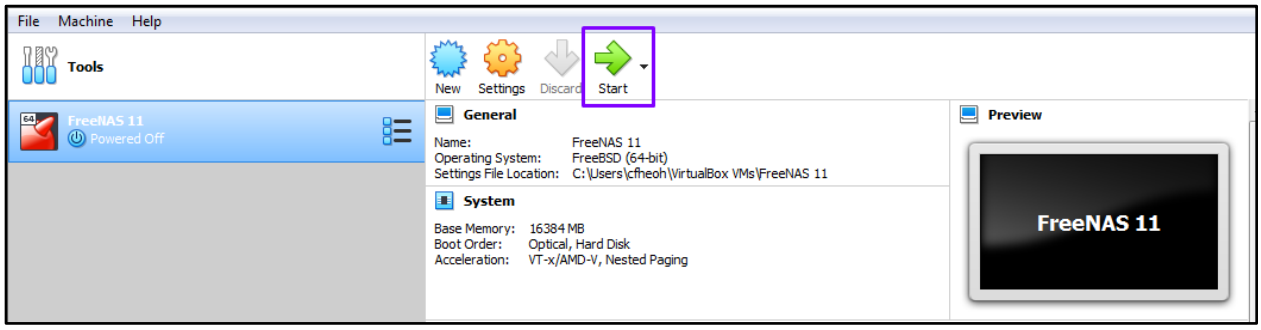


Fig 2.1: Press 'Start' to begin FreeNAS installation

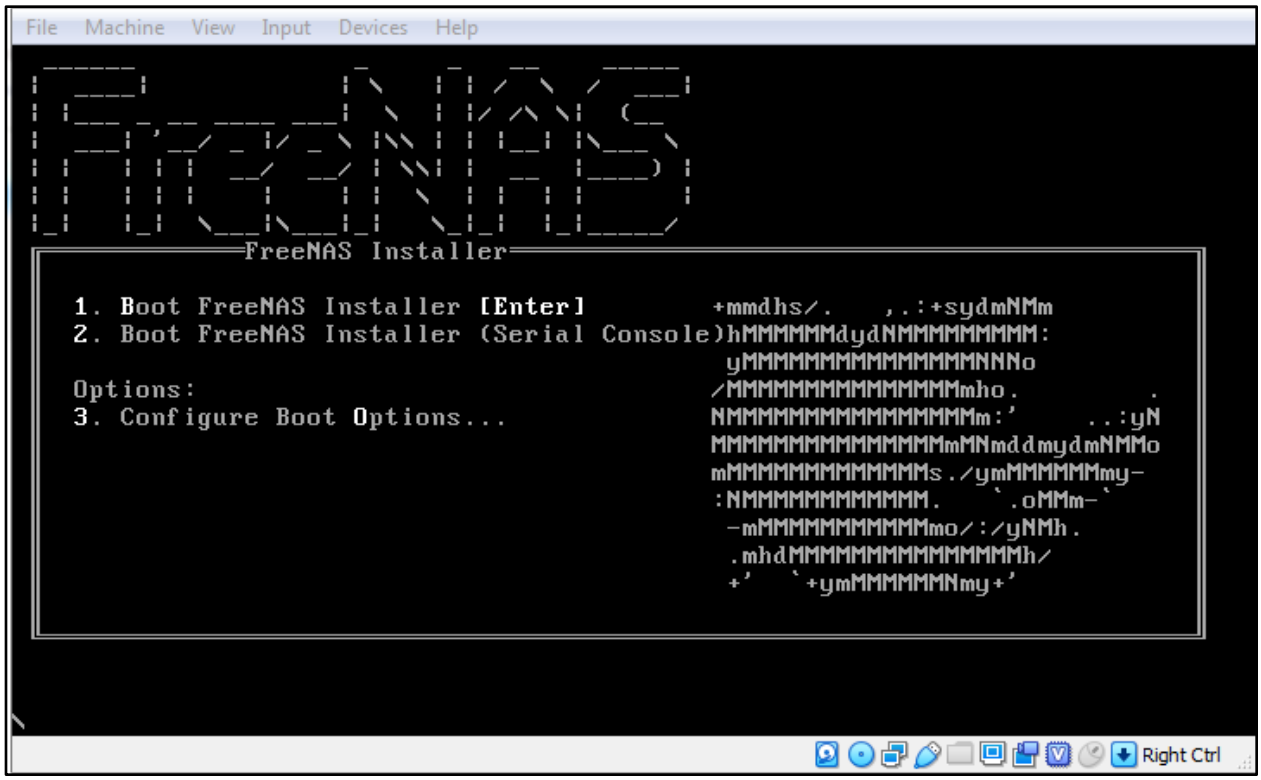


Fig 2.2: The FreeNAS Installer

Hit 'Enter' to the FreeNAS Installation Menu.

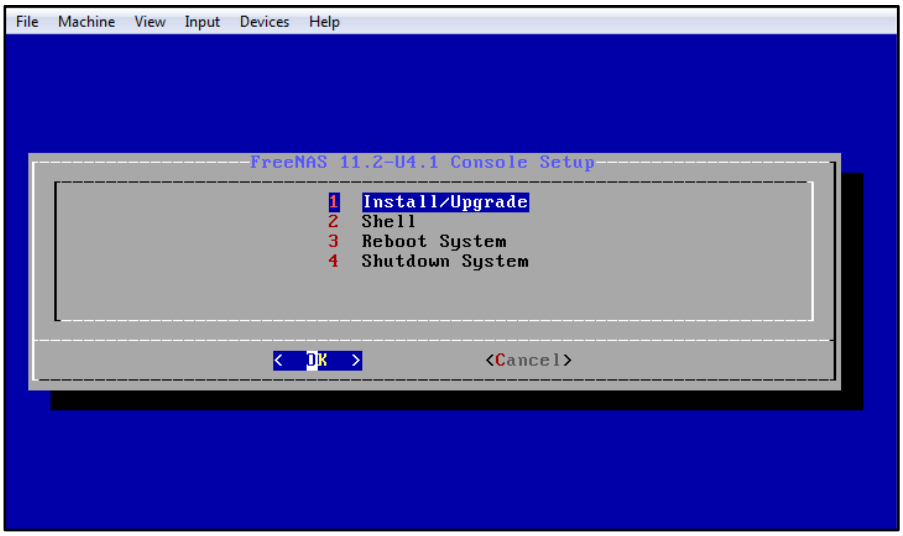


Fig 2.3: FreeNAS installation menu

Select ‘1’ to begin the FreeNAS installation. Hit ‘OK’ to continue.

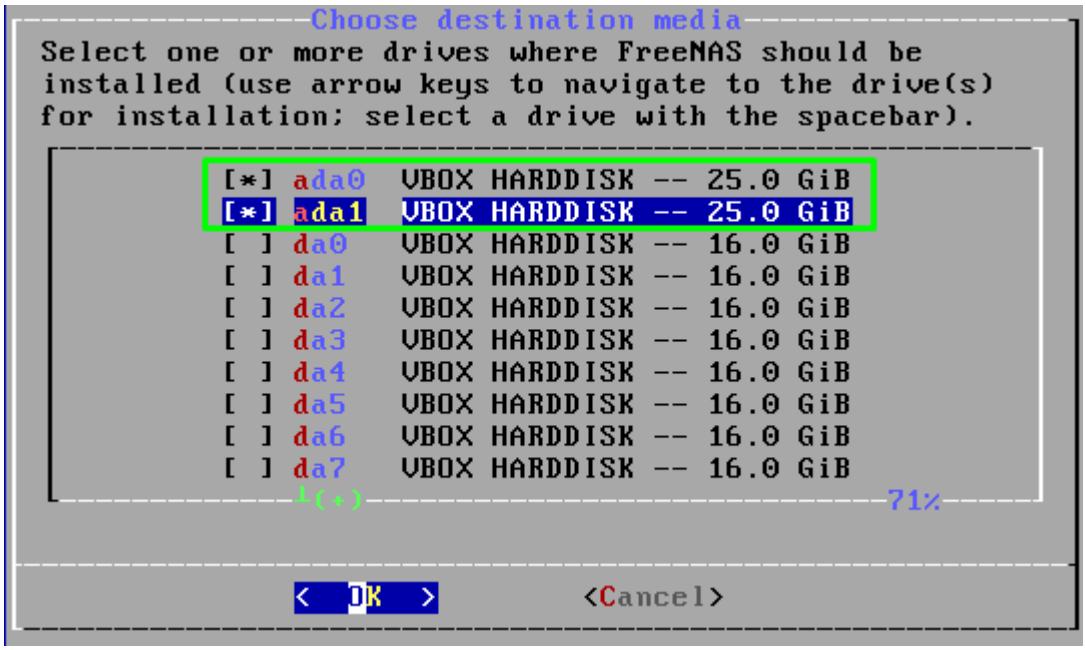


Fig 2.4: Disk selection menu

In Fig 2.4, select ‘ada0’ and ‘ada1’ as the target disks for the FreeNAS OS. These 2 HDDs correspond to ‘r0’ and ‘r1’ in the Virtualbox storage disk setup in the previous chapter. Use the ‘spacebar’ to select the disks. Hit ‘OK’ to continue.

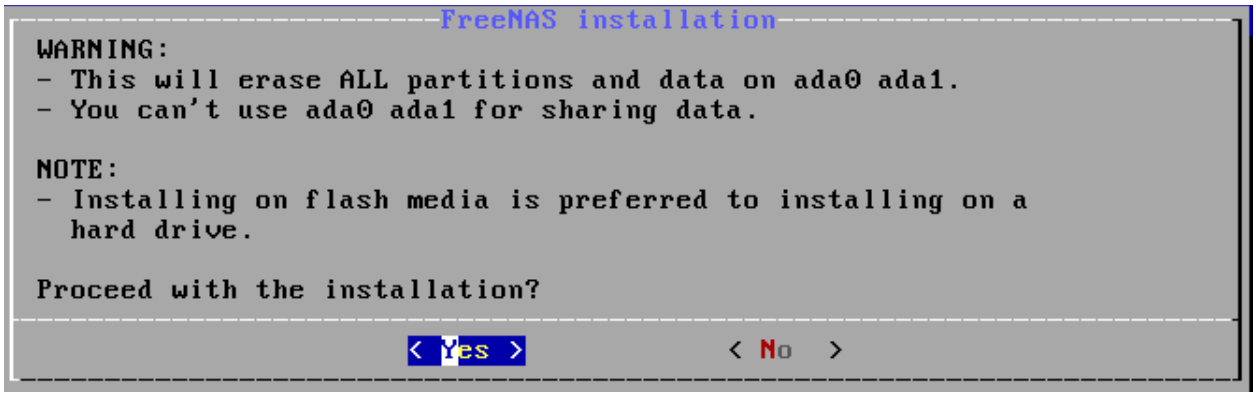


Fig 2.5: A warning message that it will erase the data of the target installation disks

Fig 2.6: Key in the root password

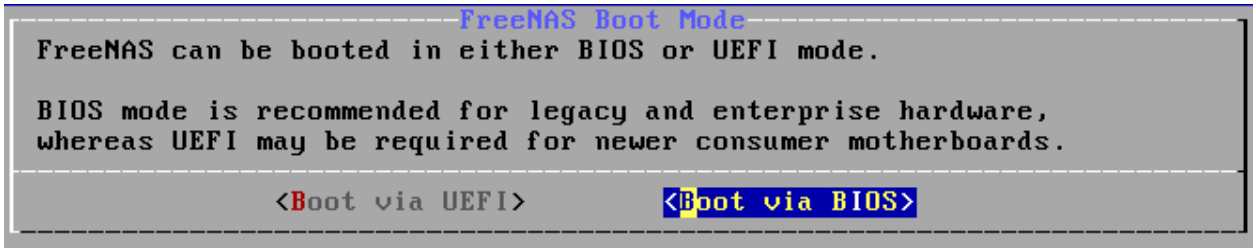


Fig 2.7: Select booting in BIOS or UEFI mode

Both BIOS and UEFI are low-level firmware that help bootstrap your server and load the necessary software code to boot FreeNAS. BIOS is an aging technology and will soon be replaced completely by UEFI. Most modern servers already support UEFI and FreeNAS has no problems booting either in BIOS mode or UEFI mode. However, we encourage you to look at the hardware compatibility lists or the FreeNAS forums for any complications with UEFI.

In this book, we select BIOS since it is the installation default.

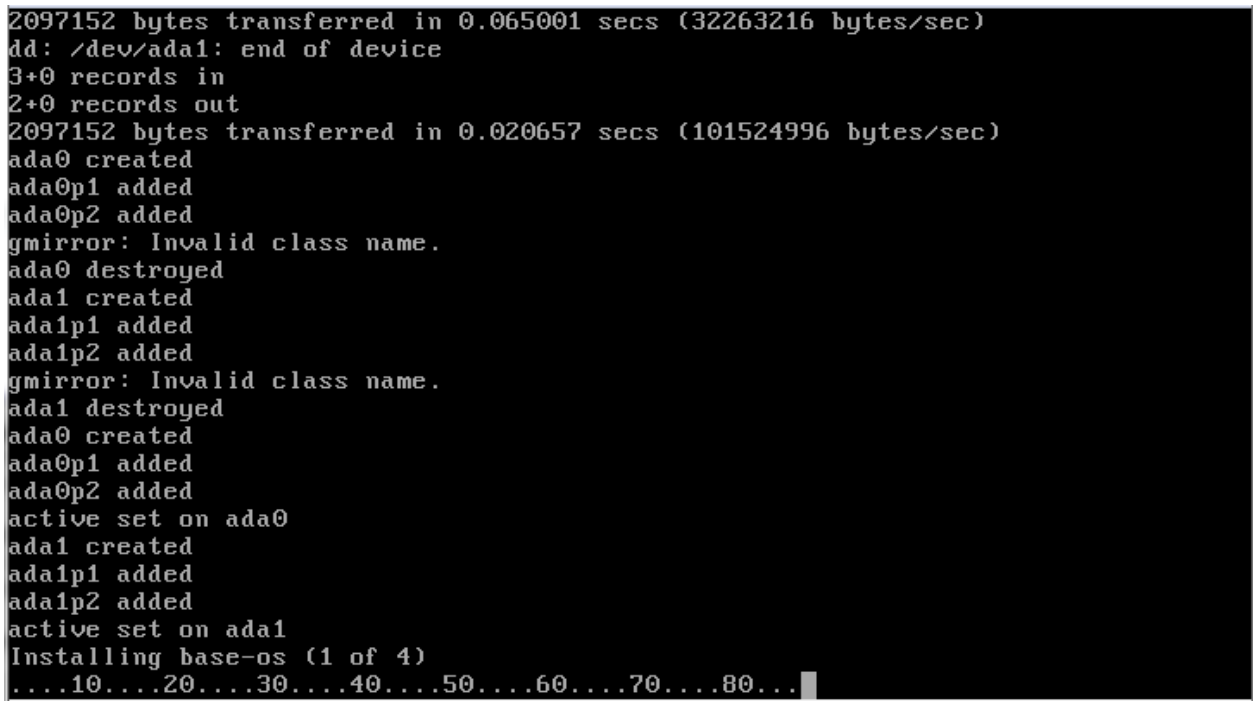


Fig 2.8: Console messages during installation.

Fig 2.9: Phase 1 (with the CD-ROM) is complete

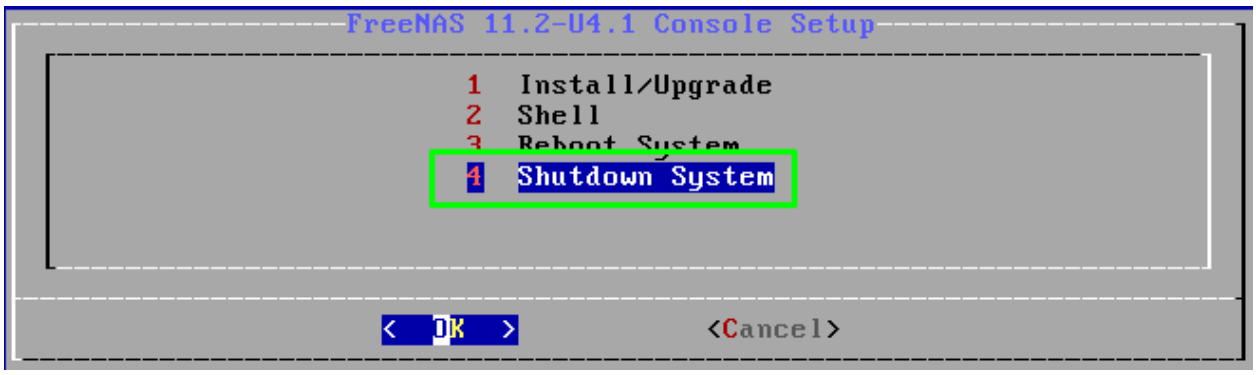


Fig 2.10: Select '4' to shutdown the FreeNAS installation after Phase 1 completion

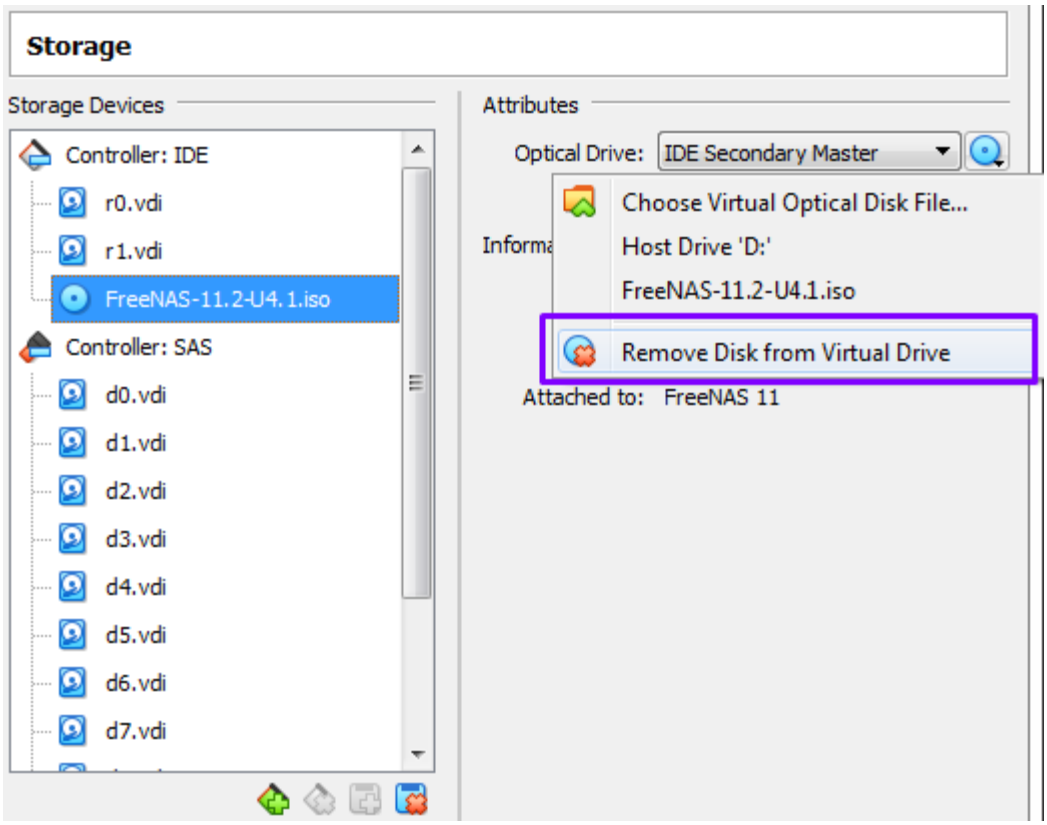


Fig 2.11: Remove the virtual CDROM

Once the FreeNAS is shutdown, it returns to the Virtualbox settings. In Fig 2.11, remove the virtual CD-ROM to continue with Phase 2 of the FreeNAS installation. If the virtual CD-ROM is not removed, it will repeat Phase 1 installation.

Phase 2 FreeNAS installation

Fig 2.12: FreeNAS Installer

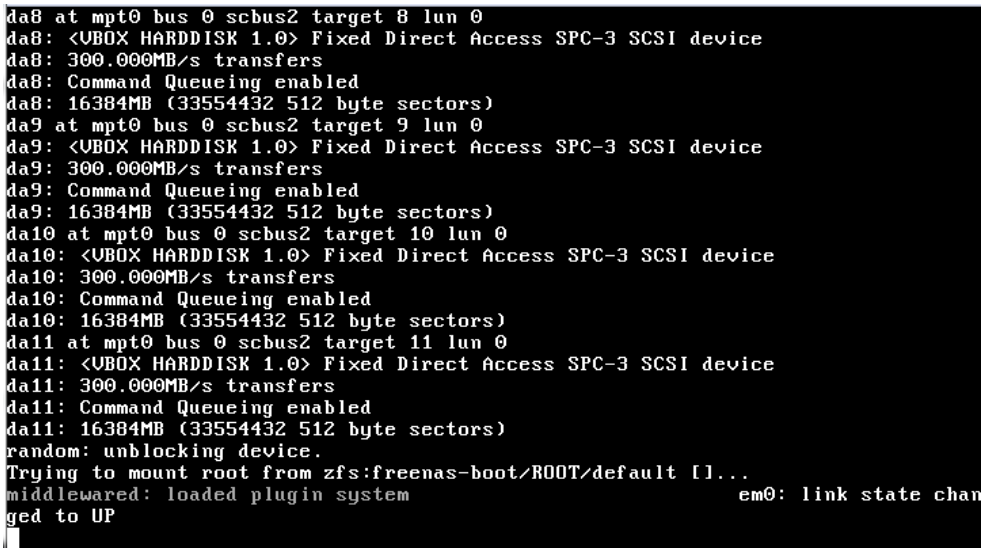


Fig 2.13: Installation messages on the console

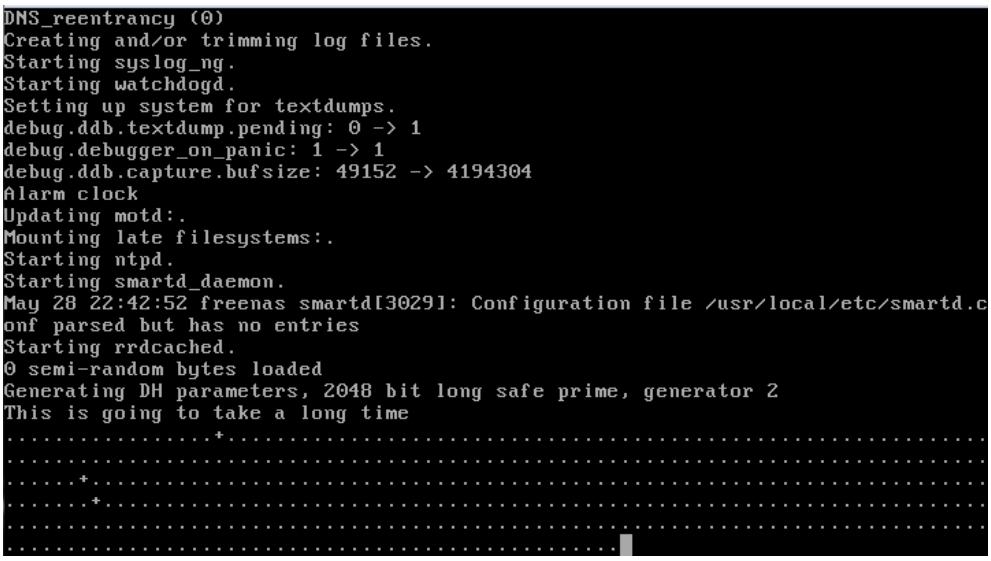


Fig 2.14: More installation messages on the console

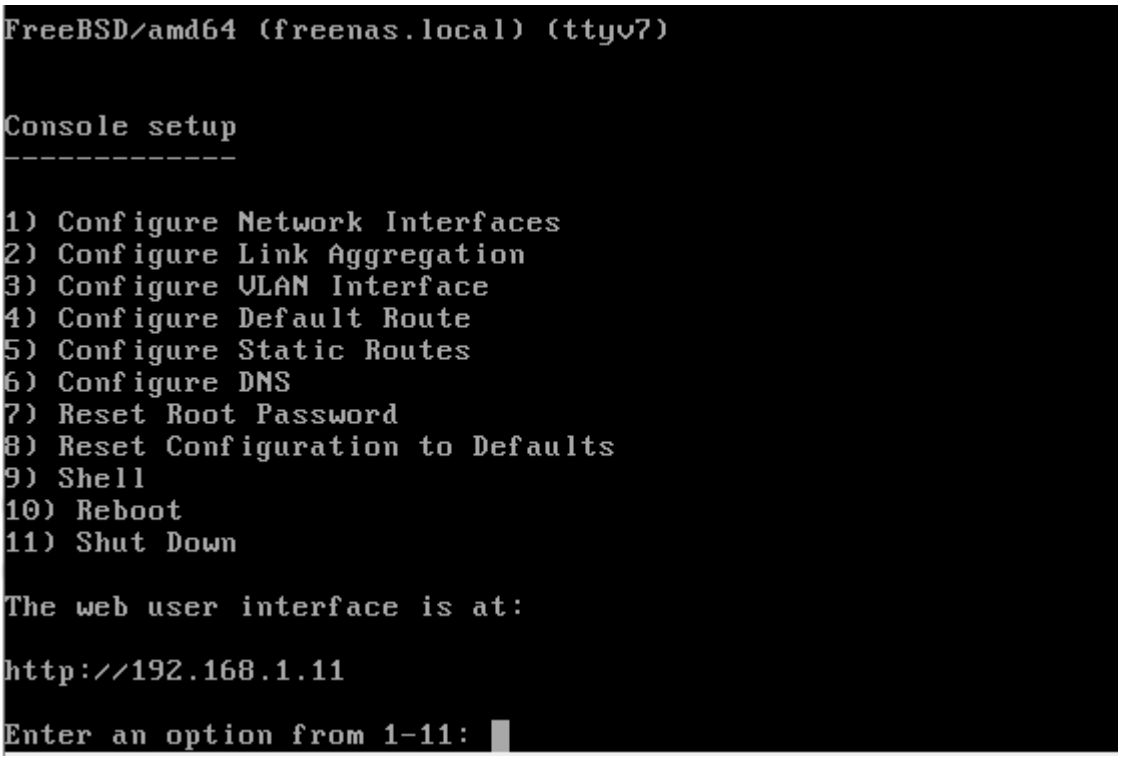


Fig 2.15: The Console Setup 1-11 menu

Once FreeNAS installation is complete, the Console setup will appear. It has a 1-11 Menu of configuration options. In previous versions of FreeNAS, there were 9 or 12 options to configure. By default, the network interface automatically receives a DHCP IP address. In the example in Fig 2.15, the IP address 192.168.1.11 is taken by FreeNAS right after the installation.

It is important to set a fixed IP address for FreeNAS rather than a DHCP given one. FreeNAS is a storage server that provide NAS and SAN services to respective clients and initiators, and the IP address must never change. To change to a fixed IP address, follow the instructions on Fig 2.16 below:

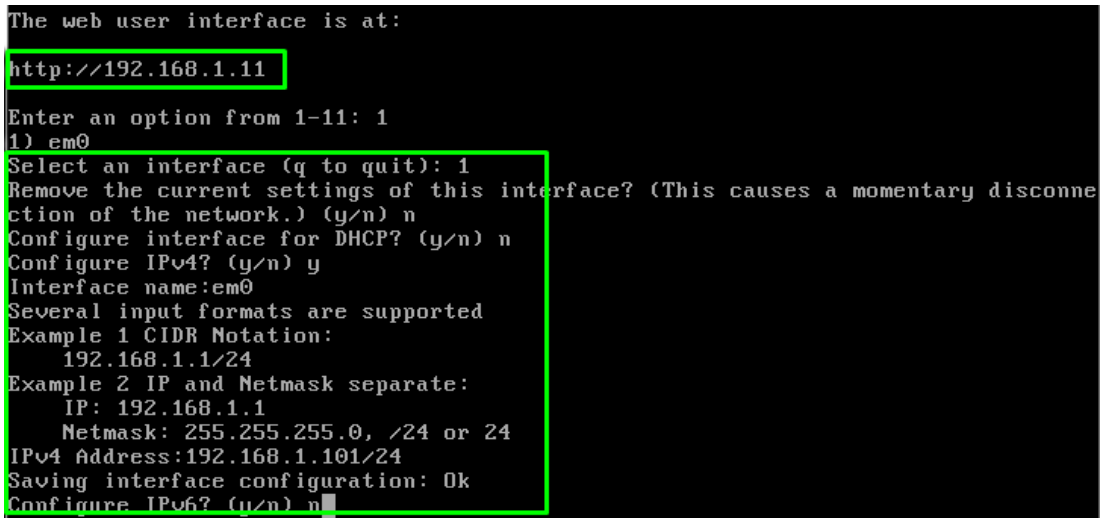


Fig 2.16: Select '1' to configure network interfaces.

The first network interface name is usually em0. If there are others, they default to em1, em2 and so on. In our example, we configure the fixed IP address to be 192.168.1.101/24, which subnets to 255.255.255.0.

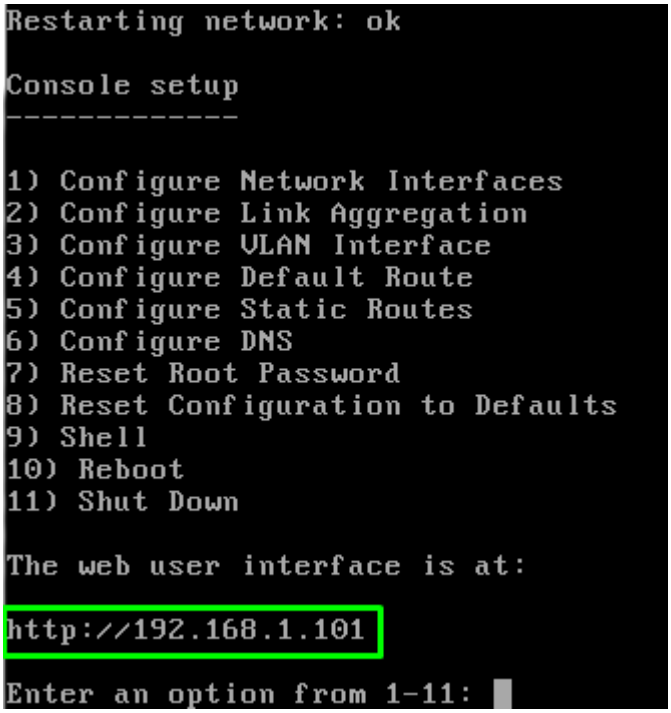


Fig 2.17: The fixed IP address configured

Once the fixed IP address is configured, we can continue the installation and configuration through the GUI interface using a supported web browser. Both Firefox and Chrome work well.

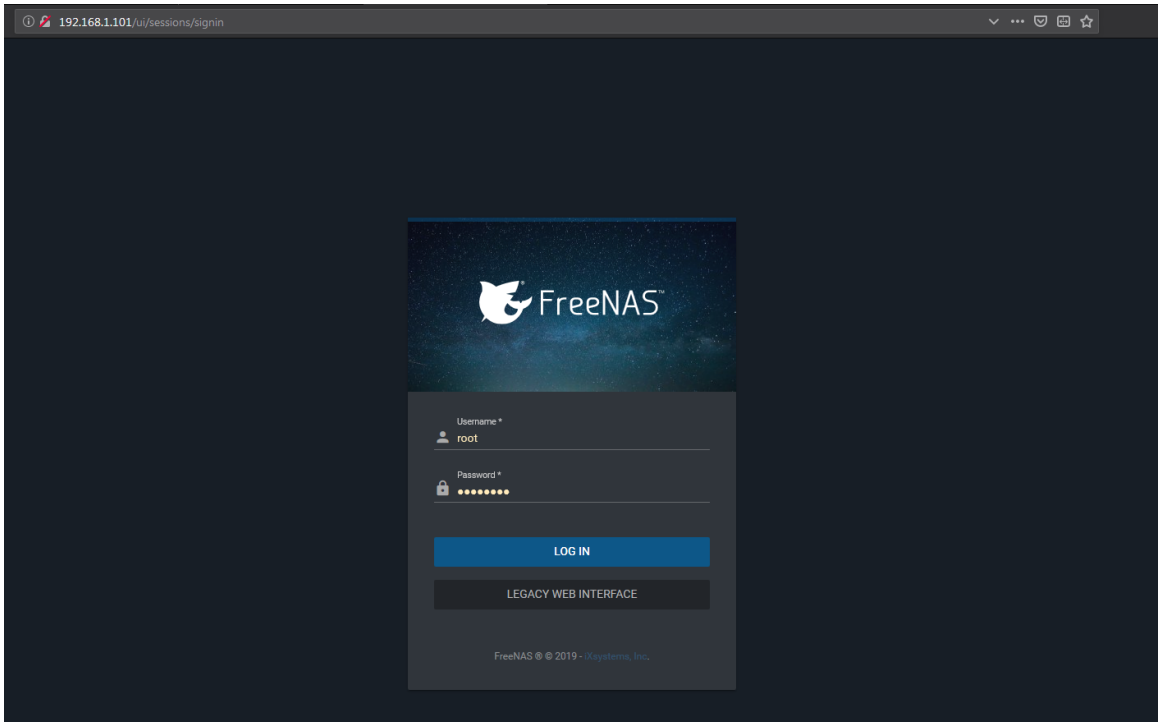


Fig 2.18: New FreeNAS web login GUI

Beginning with version 11.2, FreeNAS is replacing the older web interface with a more modern one. This is based on Angular JS. Users who are familiar with the older, legacy web interface can still switch back but the older web interface will soon be deprecated in future releases. The button below the 'LOGIN' allow users to switch to the older web interface, and switch back to the newer web interface.

Fig 2.19 shows the login of the older web interface.

Fig 2.19: FreeNAS older web login interface

Shutdown FreeNAS

Shutting down FreeNAS can be completed via the Console or the Web GUI. To complete our installation of this exercise, select '11' to shutdown.

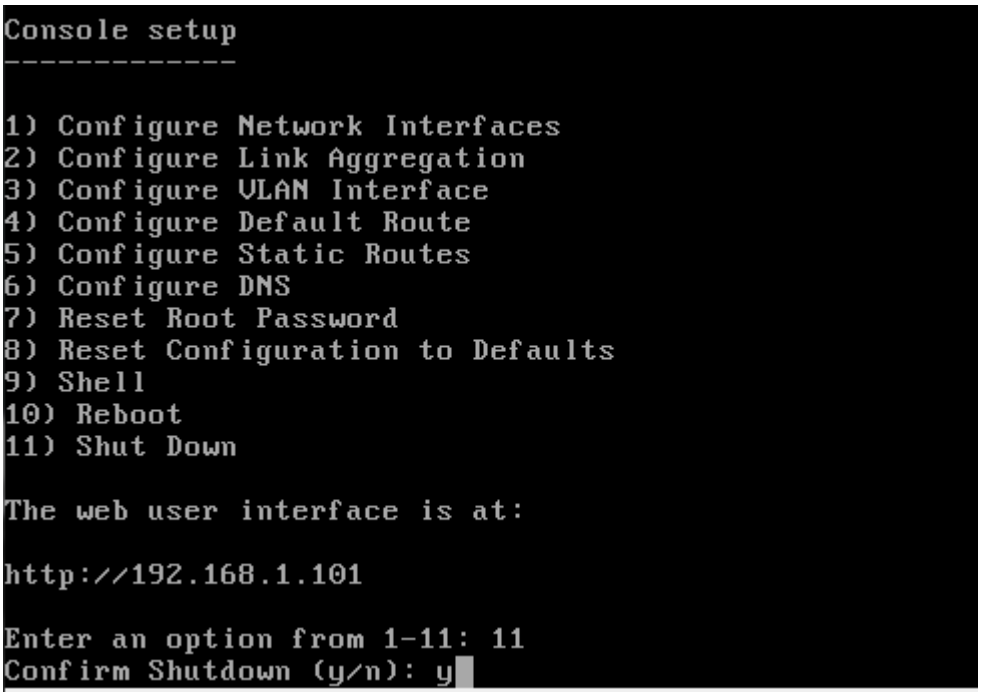


Fig 2.20: FreeNAS shutdown

CHAPTER 3 (POST INSTALLATION CONFIGURATION)

Post Installation Configuration

After the installation is completed, there are some housekeeping tasks to consider. These include post installation configurations which prepare the FreeNAS server to run and operate effectively. These configurations include setting up DNS correctly, default routing, network time services and more.

Configure DNS (for Active Directory and DNS Resolution)

Typically FreeNAS will receive the DNS (Domain Name Server) servers' IP addresses provided by the DHCP service. This is OK if FreeNAS is used in a Windows Workgroup environment. However, in most network setup today where Windows is prevalent, Active Directory (AD) is important for user and service authentication, as well as access control.

In an AD domain, there is a domain controller which is also likely to be the primary DNS server for the domain. Therefore, we can setup FreeNAS to join the AD domain by setting the IP address of the DNS server of the domain. In our setup, the DNS server IP address is 192.168.1.100.

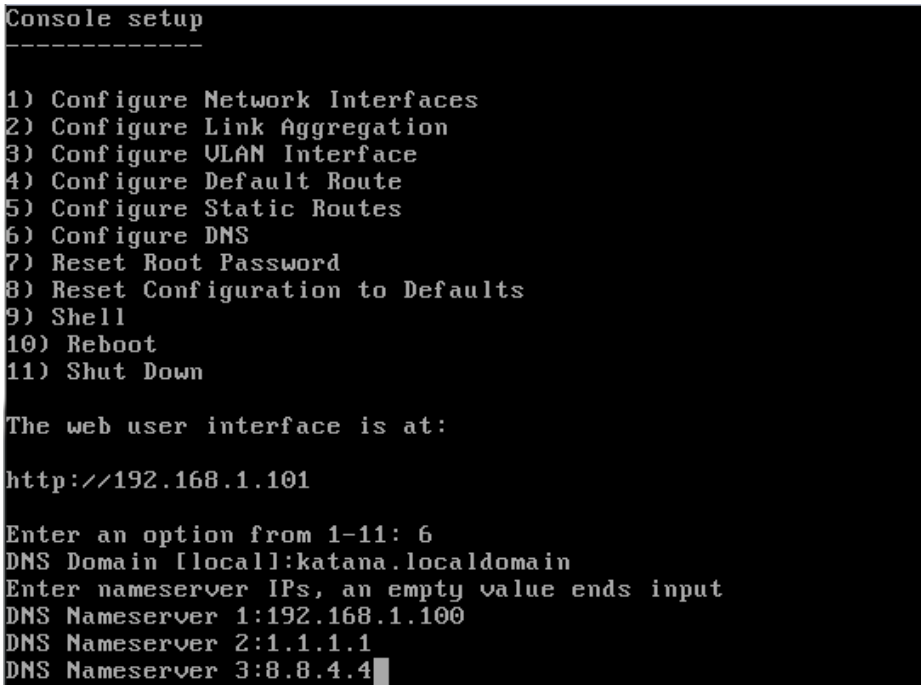


Fig 3.1: Setting up the DNS servers

DNS Name Server 1: 192.168.1.100 (DNS Server of the AD domain)
DNS Name Server 2: 1.1.1.1 (external DNS server - CloudFlare)
DNS Name Server 3: 8.8.4.4 (external DNS server - Google)

Configure Default Route

It is useful to configure the default route to ensure efficient networking.

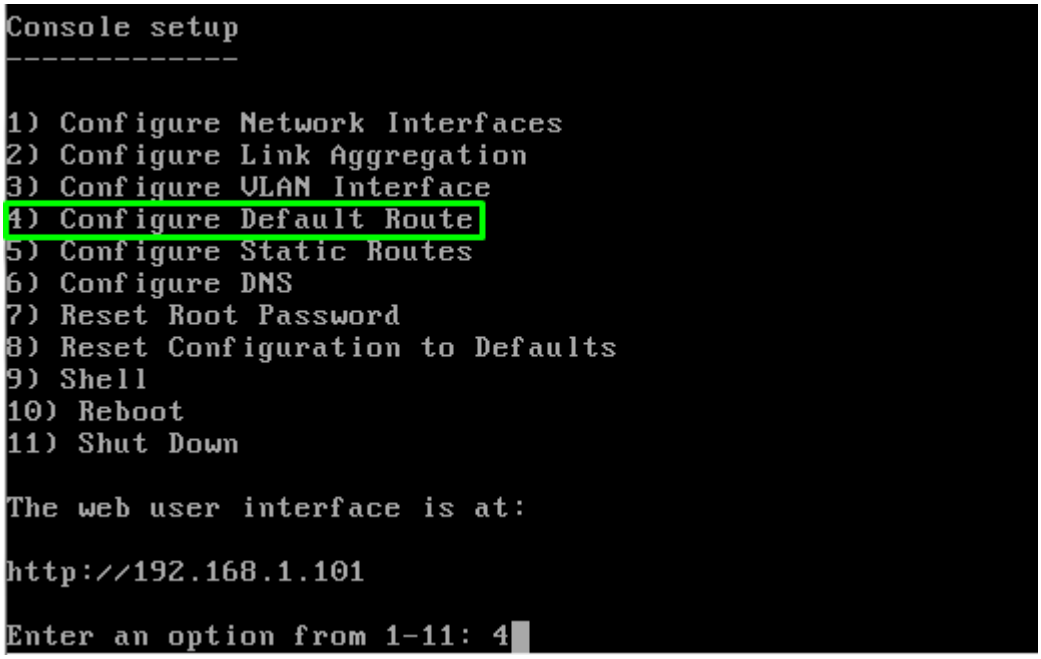


Fig 3.2: Configuring default route

Select (4) from the 1-11 console to Configure Default Route as shown in Fig 3.2.

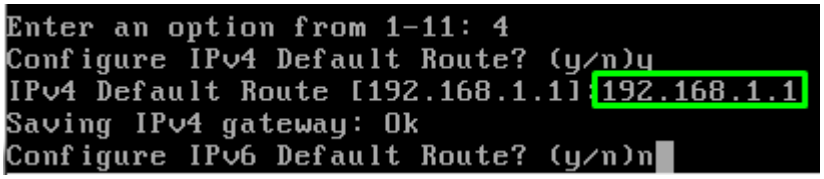


Fig 3.3: Configuring the IPv4 default route

In Fig 3.3,
Configure IPv4 Default Route? : y
IPv4 Default Route [ ] : 192.168.1.1
Configure IPv6 Default Route? : n

After the Default Route has been set, test by pinging to a domain name through the shell.

GUI Post Installation Configurations

The FreeNAS dashboard is the first page shown after logging in.

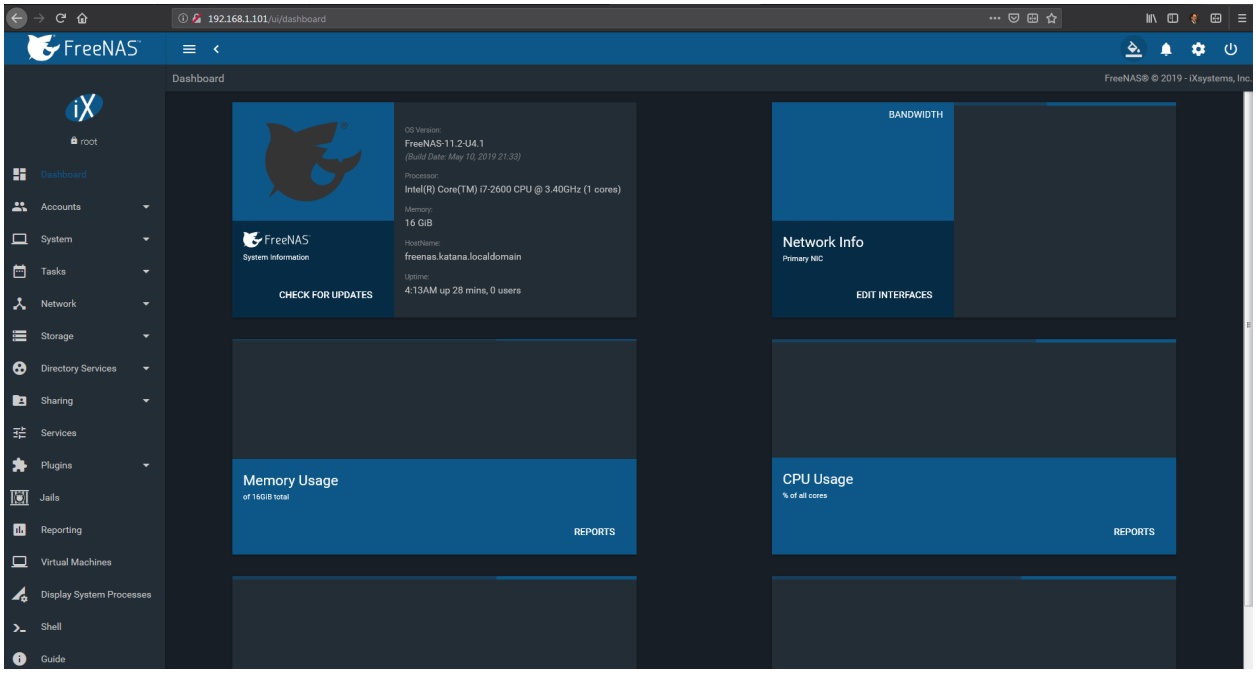


Fig 3.4: FreeNAS Dashboard

The default theme is iX Dark.

Fig 3.5: Change the theme

To change the theme of the WebGUI interface, select the “bucket” logo on the top right corner. We have chosen the theme iXBlue for a better view in this book.

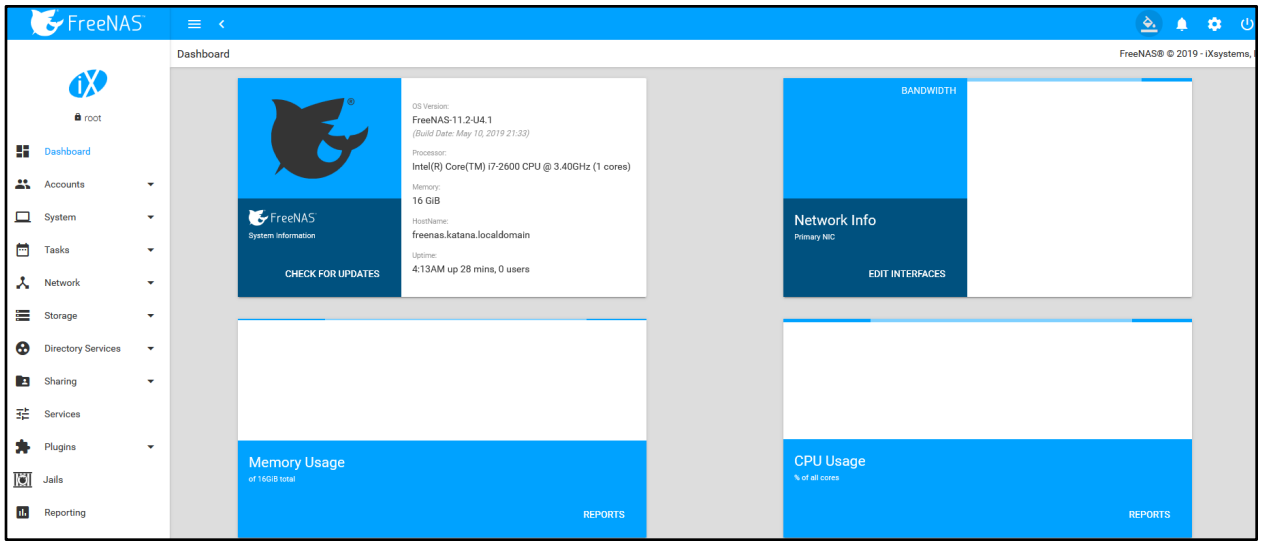


Fig 3.6: The theme is iXBlue

Configuring Common Systems settings

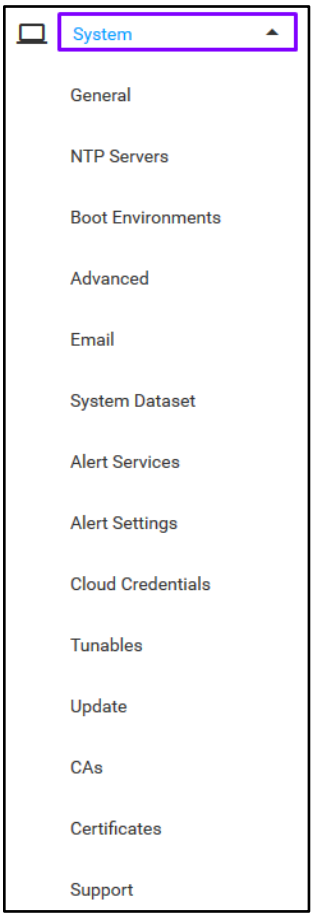


Fig 3.7: System configuration view on the left navigation column

The typical configuration settings, as shown in Fig 3.8 below are:

- WebGUI IPv4 Address: 192.168.1.101 (This is the HTTP browser IP address)
- Timezone: Asia/Kuala_Lumpur

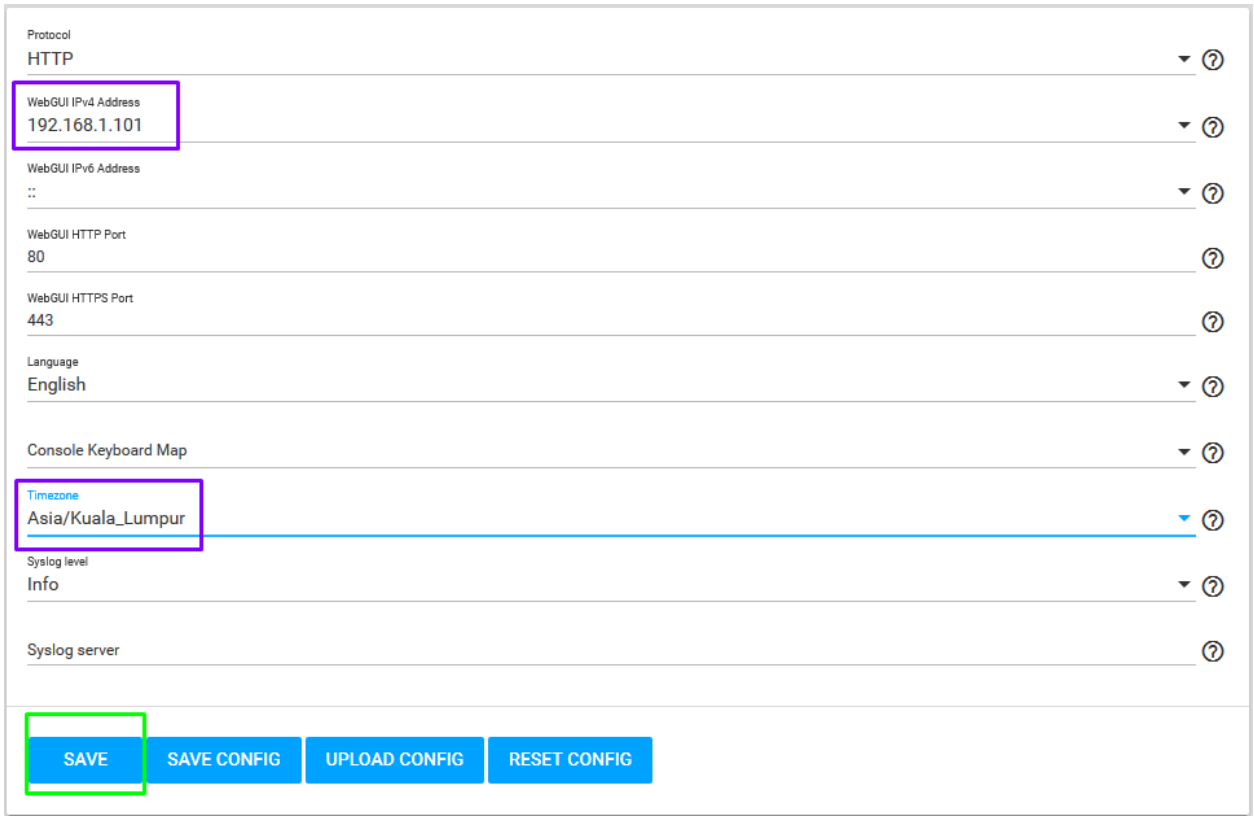


Fig 3.8: System > General

Click ‘SAVE’ to set the settings.

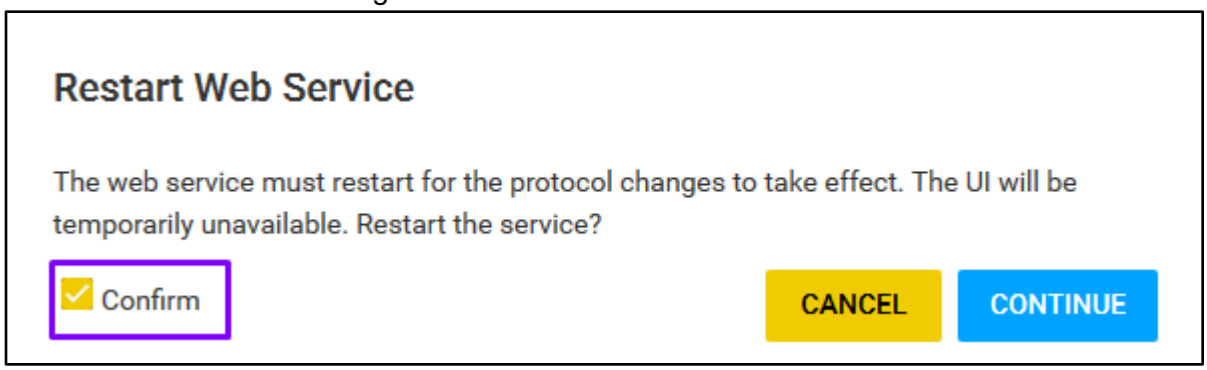


Fig 3.9: Pop up message to restart Web Service after saving setting

A pop up message appears to ‘Restart Web Service’ after saving the settings. Click ‘CONTINUE’ to restart. It takes a few seconds to get a refreshed WebGUI interface.

Another good practice is to save the FreeNAS system configurations in a file. This is useful when FreeNAS is upgraded or reinstalled, but you wish to reuse the previous configuration.

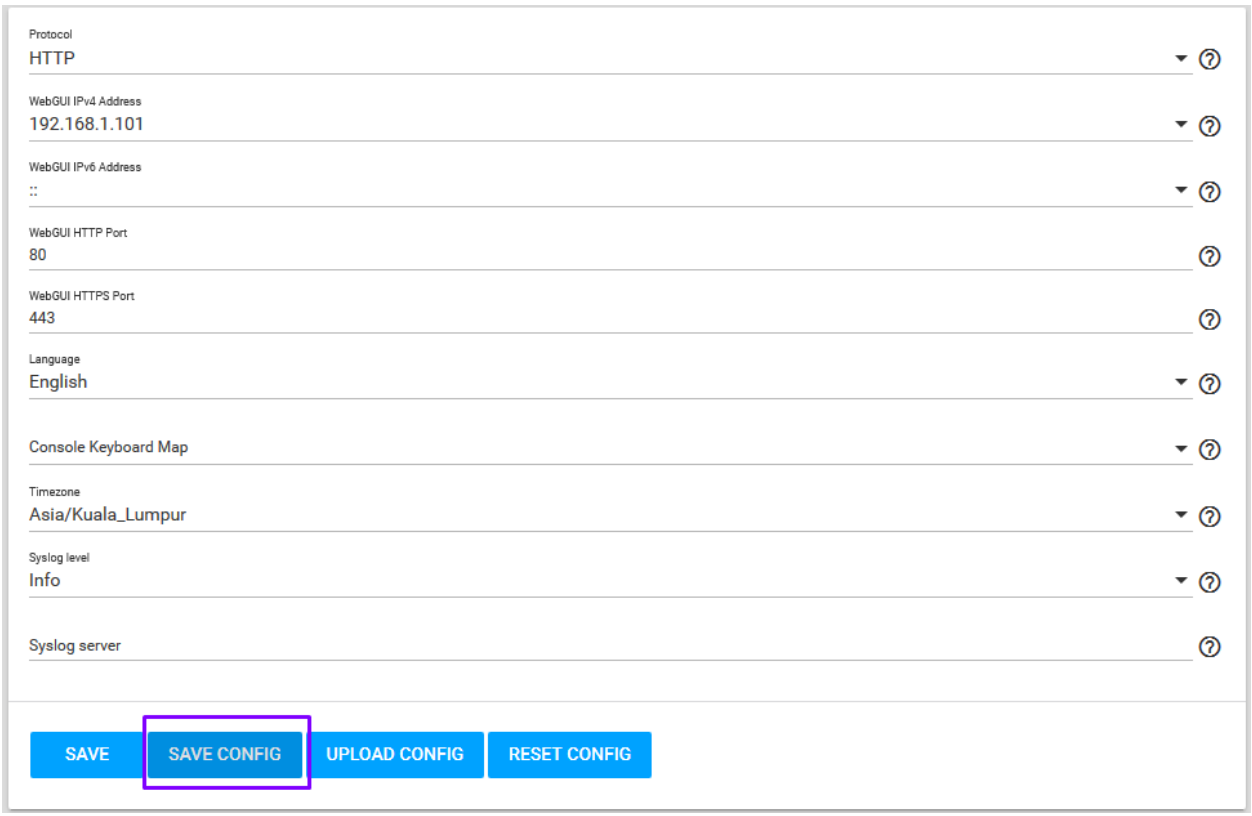


Fig 3.10: Save the FreeNAS system configuration

Click the ‘SAVE CONFIG’ button to save the system configuration to a file on the computer.

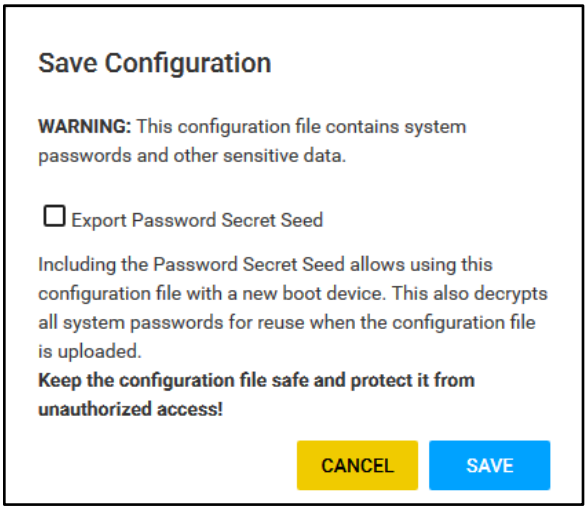


Fig 3.11: A pop up message about Save Configure

After the pop-up message, click ‘SAVE’.

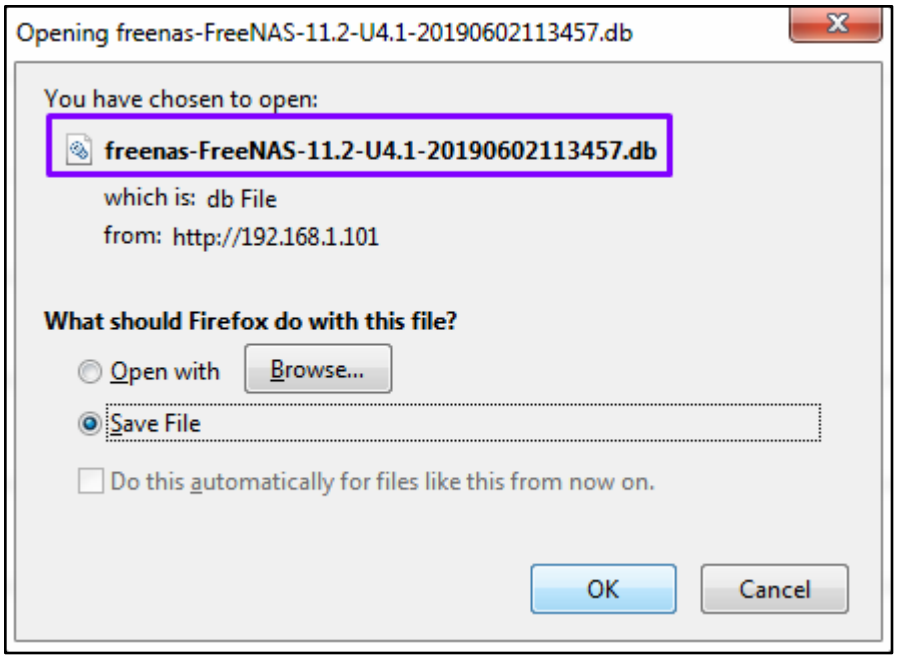


Fig 3.12: Save the configuration file to the computer

The file will be saved to the computer with the format of *hostname-version-timestamp.db* as shown above in Fig 3.12

Fig 3.13: Upload Config

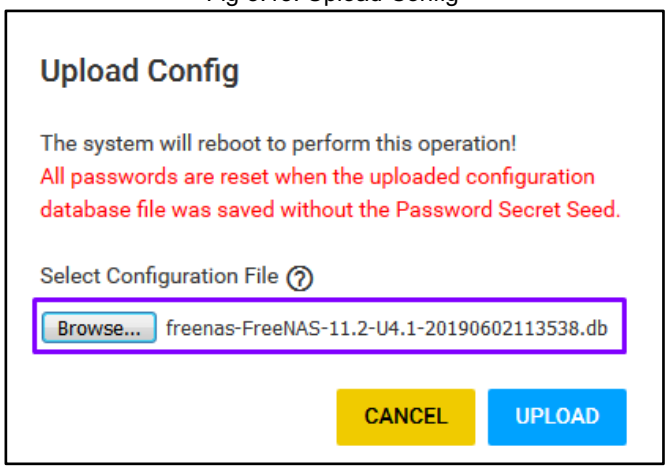


Fig 3.14: Upload Config File in the pop up message

NTP Servers

NTP, or Network Time Protocol servers are important to ensure that there is time synchronization for time-based services. Active Directory relies on NTP for its authentication and authorization services to work.

By default, FreeNAS already has 3 NTP servers from freebsd.pool.ntp.org setup.

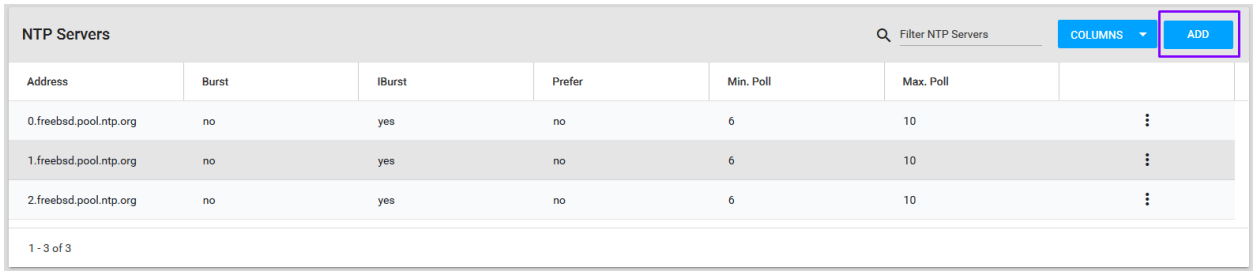


Fig 3.15: Adding new NTP server

For Active Directory domain configuration, select ‘ADD’ on the top right corner to add the NTP server. Typically in most setup, the NTP server is also the AD domain controller.

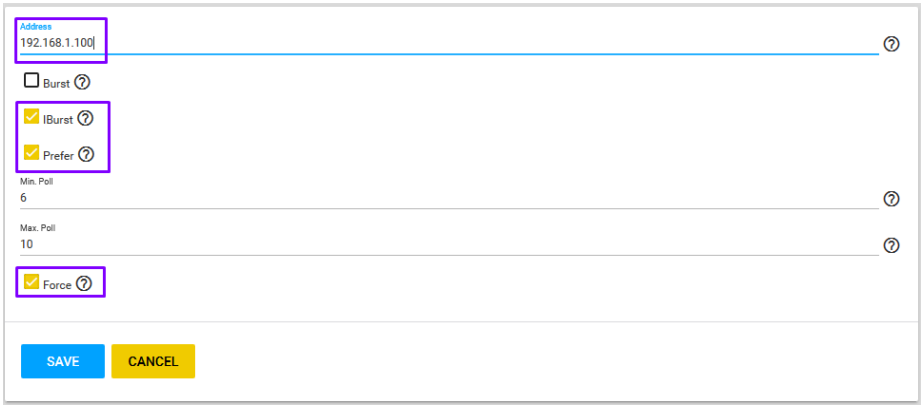


Fig 3.16: NTP Server Configuration

Address: 192.168.1.100
Check boxes for

- iBurst
- Prefer
- Force

Click ‘SAVE’ to add the new NTP Server

Other post install configurations

One useful configuration is to setup the console messages for the WebGUI interface, System > Advanced

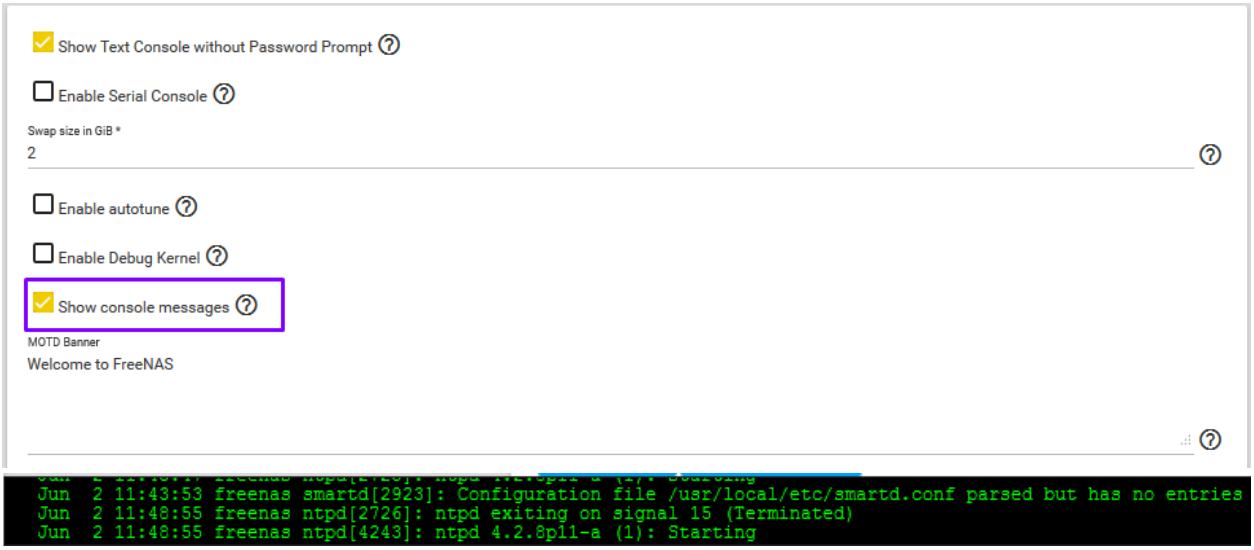


Fig 3.17: Console at the bottom

Check the box 'Show console messages'. The console appears at the bottom of the WebGUI after click 'SAVE'.

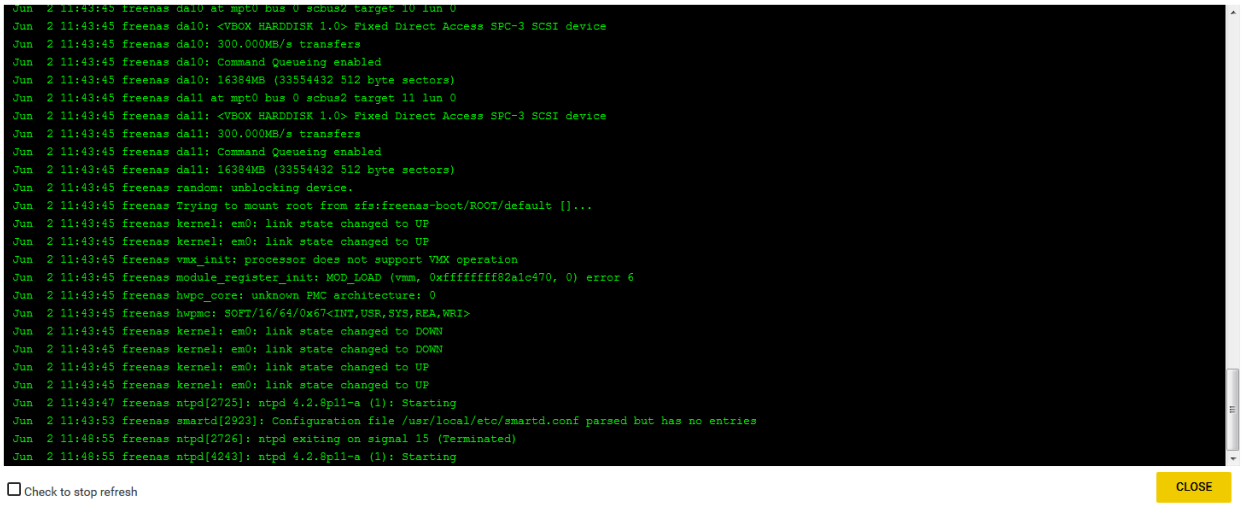


Fig 3.18: Pop up console

When you on the console at the bottom, the console pops up in front of the WebGUI browser. The console shows system messages, alerts and is useful for information and troubleshooting.

Configure Email Alerts

In a production environment, it is important for the FreeNAS to send system messages, disk scrub info, alerts and possible hardware issues to the administrator or the support email.

The root user will send these messages and the first step is to configure the root user's email address.

On the left navigation column, Account > Users as shown below in Fig 3.19.

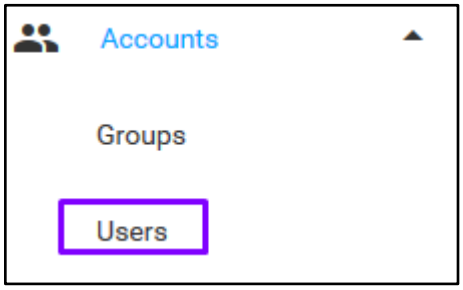


Fig 3.19: Accounts > User

In Fig 3.20, the root user is on top of the Users table. Click on the 'Edit' button.

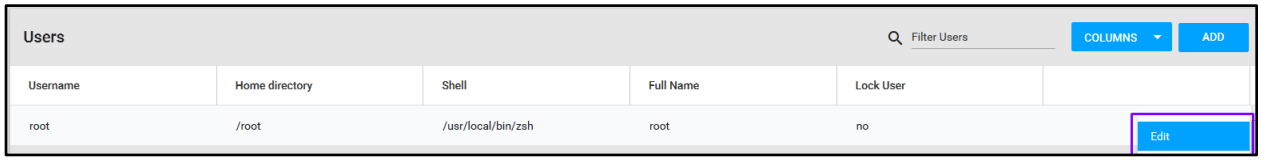


Fig 3.20: Edit root user

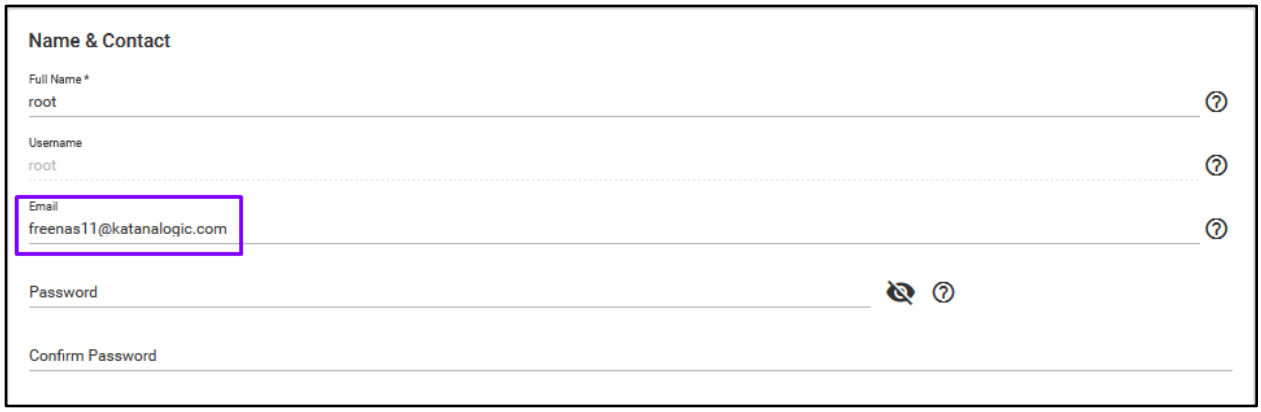


Fig 3.21: Configure email of root user

In our example, we configured the email to be *freenas11@katanalogic.com*.

In the email server, the user *freenas11@katanalogic.com* should have been setup as an email account.

To configure the FreeNAS to send email from root to the email address,

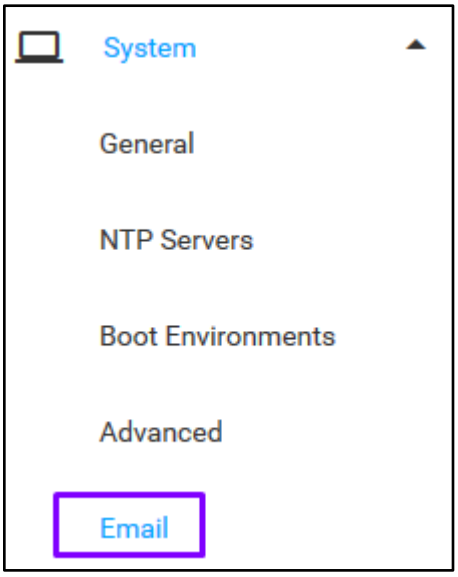


Fig 3.22: System > Email

Fig 3.23: SMTP server configuration

In Fig 3.23,
From Email: root@katana.local domain (any simplified name is helpful)
Outgoing Mail Server: mail.katanalogic.com (SMTP server)
Mail Server Port: 1000 (SMTP server port)
Check SMTP Authentication, with Username *freenas11@katanalogic.com* and Password
Click 'SAVE'.

Click ‘SEND MAIL’ to test the email setup.

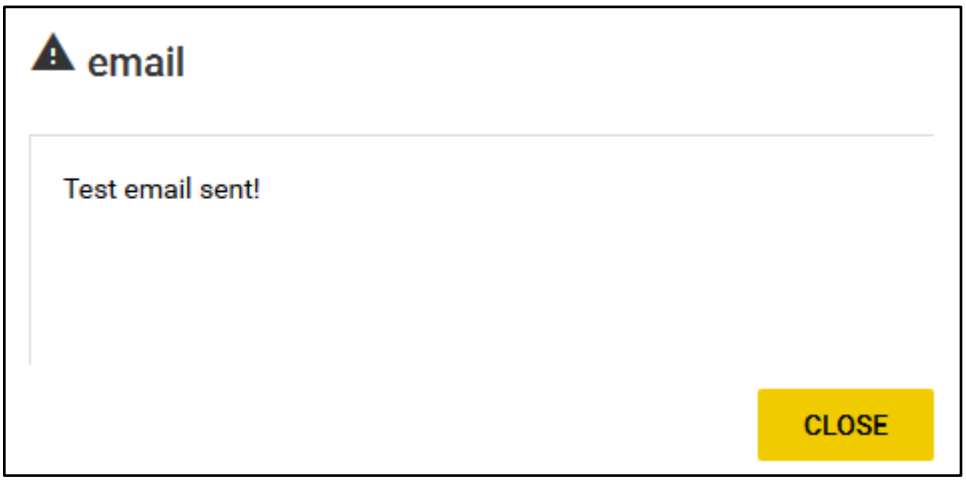


Fig 3.24: Pop up for Email sent.

If this is configured correctly, you should receive a test email from root@katana.localdomain in the *freenas11@katanalogic.com* mailbox.

In our support setup, the *freenas11@katanalogic.com* redirect to support@katanalogic.com

Configure email to a Gmail mailbox

There are situations where you do not have a company email address to handle the FreeNAS emails. You can setup Google Gmail to handle that. The setup is the same as the above, except that Gmail rejects authentication from sources that it considers insecure. When testing ‘SEND MAIL’, you will receive a pop up messages as in Fig 3.25.

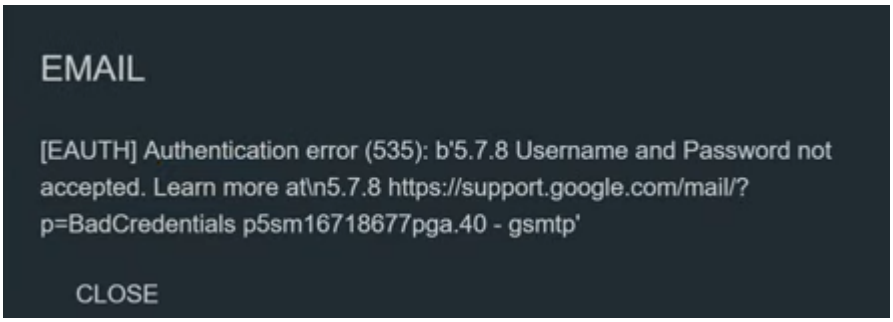


Fig 3.25: Gmail authentication error

Go to your Gmail account and select ‘Security’.

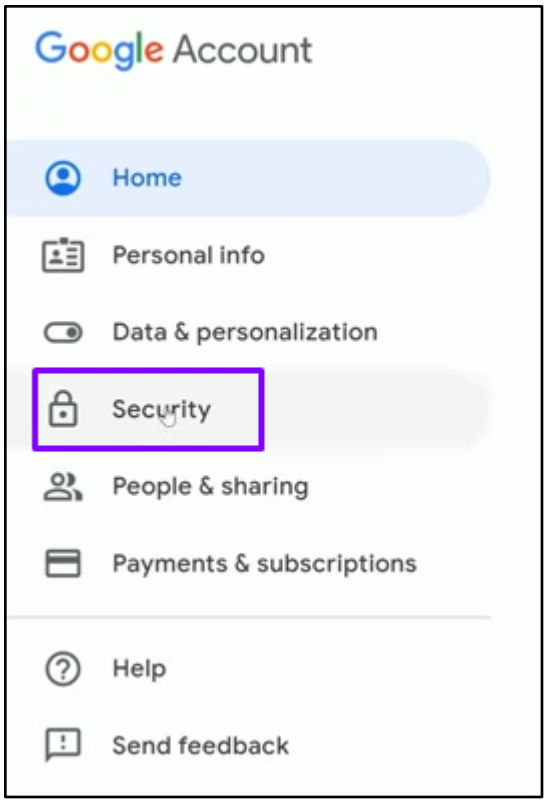


Fig 3.26: Google Gmail Security

Fig 3.27: Turn on access for less secure app access

As in Fig 3.27, click 'Turn on access (not recommended' for Less secure app access.

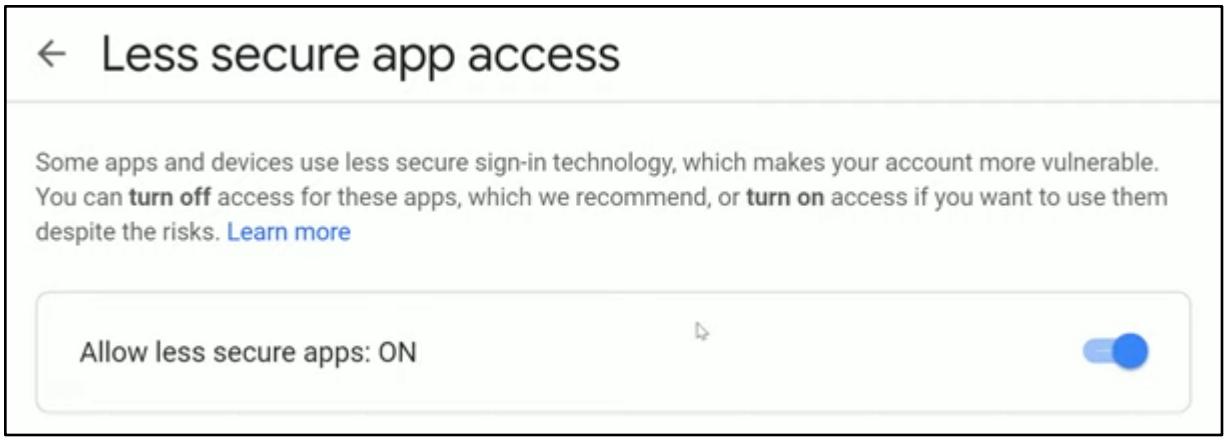


Fig 3.28: Slide Allow less secure apps: ON

Remote Access from Public Internet with Port Forwarding and Dynamic DNS

In a production environment, the FreeNAS storage could be housed in a secure server rack of a data center. Physical access is restricted and the system administrator is required to access the FreeNAS storage from outside the network through the public Internet.

There are 2 things to consider to allow access

1) Port Forwarding
2) Dynamic DNS

In the diagram Fig 3.29 below, the system administrator at a remote location wants to access the FreeNAS in the internal LAN through the public Internet.

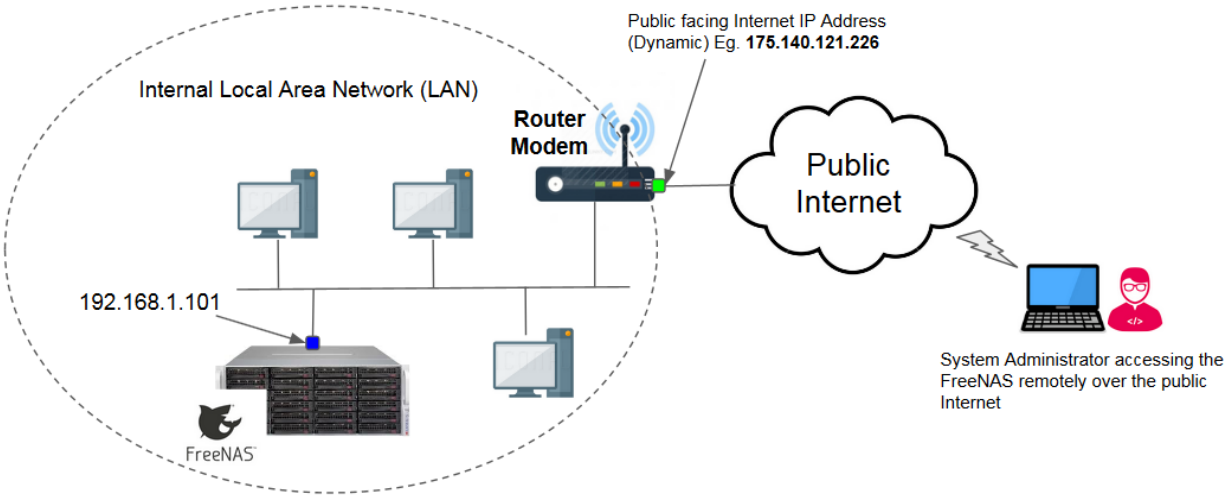


Fig 3.29: Remote access network diagram

He or she can access the FreeNAS either through a secure shell via SSH (well known TCP port 22) or via the WebGUI (well known port HTTP port 80 or HTTPS port 443). In some organizations, the Router Modem is usually the network equipment that is connected to the public Internet. In other organizations there may be firewalls in place. Either or, a connection must be configured to allow the system administrator at the remote location to access the FreeNAS storage. One simple way is to setup and configure Port Forwarding on the Router Modem device.

The Router Modem has a public Internet facing interface. To find out its public Internet IP address, Google “what is my ip”. The result is shown below (Fig 3.30):

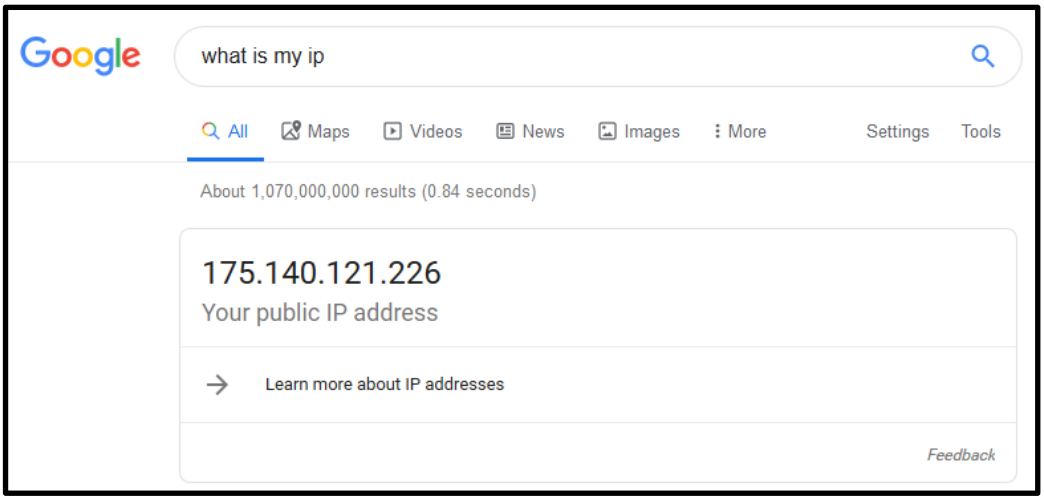


Fig 3.30: Result from Google of ‘what is my IP’

Our example shows 175.140.121.226 as our public Internet IP address. Typically this IP address is not a fixed IP address, but a dynamic IP, where it can change from time to time. We will discuss how we can overcome this part in the Dynamic DNS section, but for now, we will use the given public Internet IP address to setup Port Forwarding.

The setup of Port Forwarding is configured at the Router Modem or the Firewall device. There are many brands and models of routers and firewalls in the market and each is different. The Port Forwarding feature is common and can be intuitively configured.

In my case, I have a ZTE H267A Router Modem (Fig 3.31).

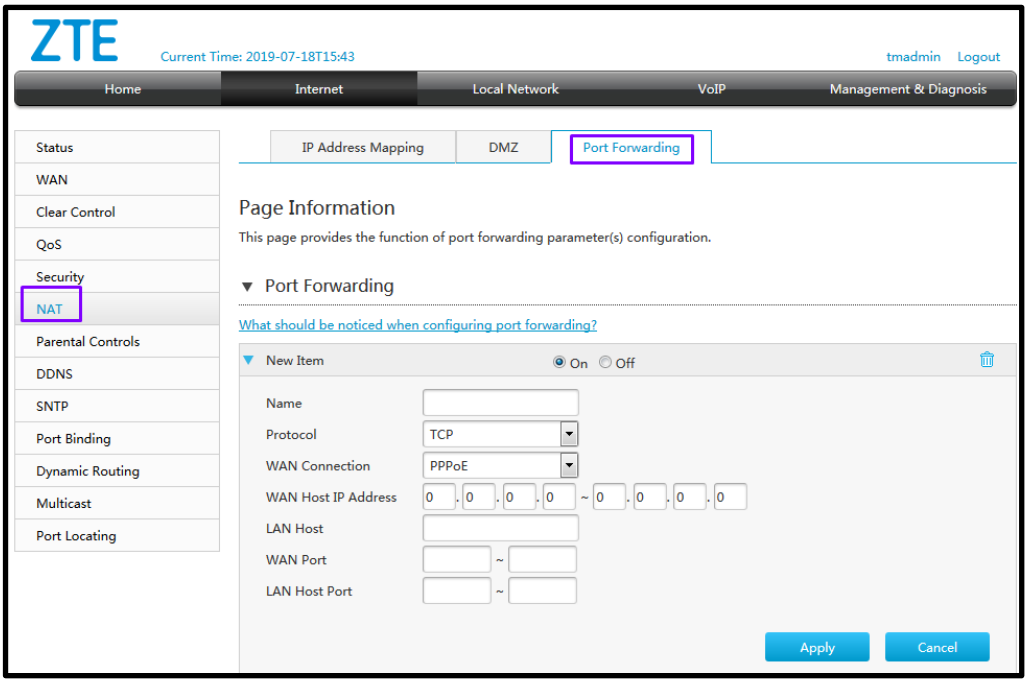


Fig 3.31: Port Forwarding Interface of the ZTE H267A router modem

I have configured 2 port access as shown in Fig 3.32 below.

1) FreeNAS SSH to allow SSH secure shell access via CLI via external port 2222 with public Internet IP address of 175.140.121.226

1) FreeNAS WebGUI to allow browser access to the FreeNAS WebGUI via external port 8088 with public Internet IP address of 175.140.121.226

IMPORTANT: Remember to Enable the Port Access

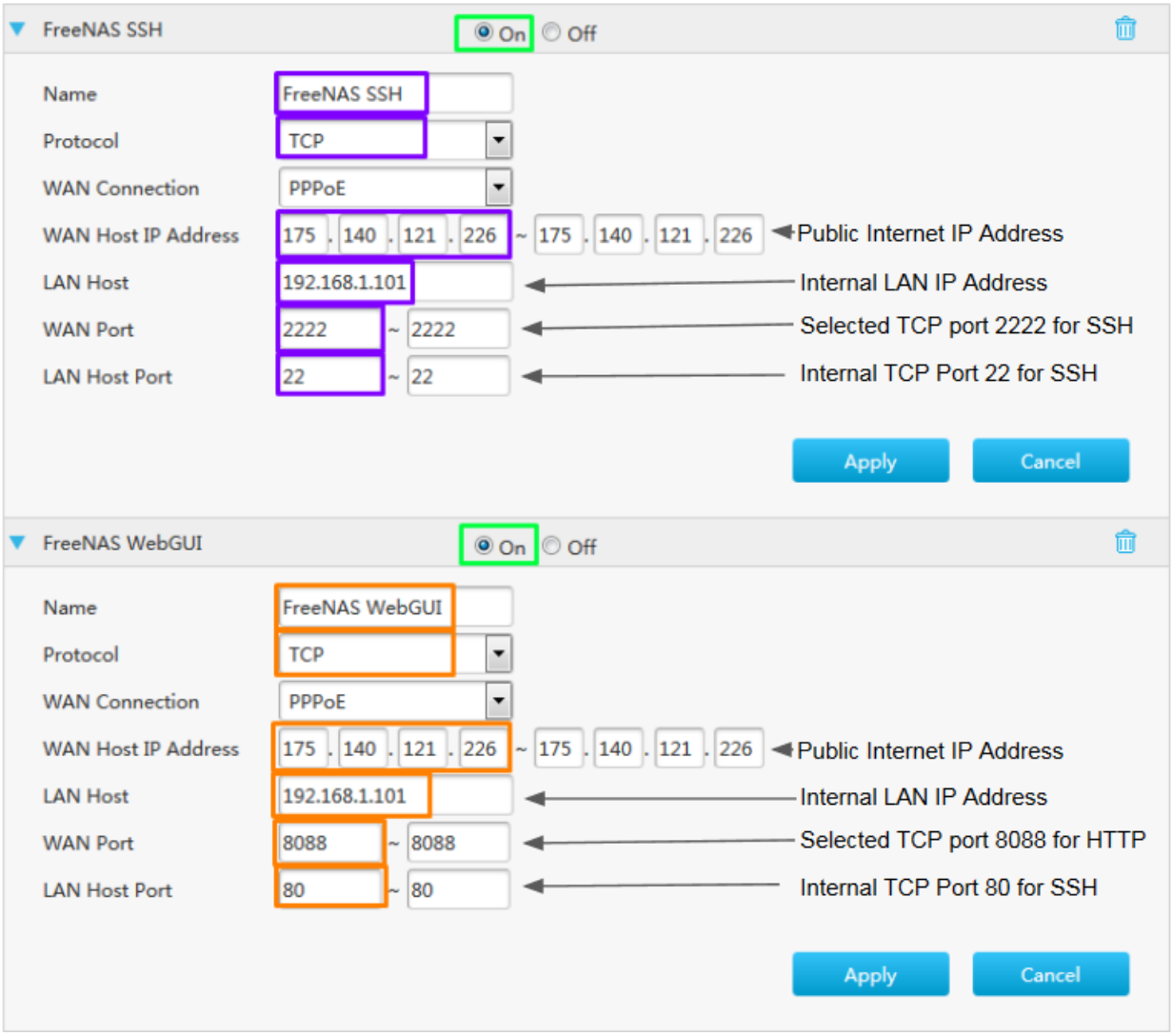


Fig 3.32: Port Forwarding configuration

To ensure that the 2 respective ports are open and accessible from the public Internet, you can test it. There are several websites that provide the checking of the open ports - https://canyouseeme.org, https://portchecker.co and https://www.yougetsignal.com/tools/open-ports/

In this section, I am using a pre-configured port forwarding setup with ports 8888 for HTTP and port 2222 for SSH. I am using https://www.yougetsignal.com/tools/open-ports/ , and it shows that the public IP address has ports 8888 and 2222 (red box) open to allow remote access via HTTP and SSH.

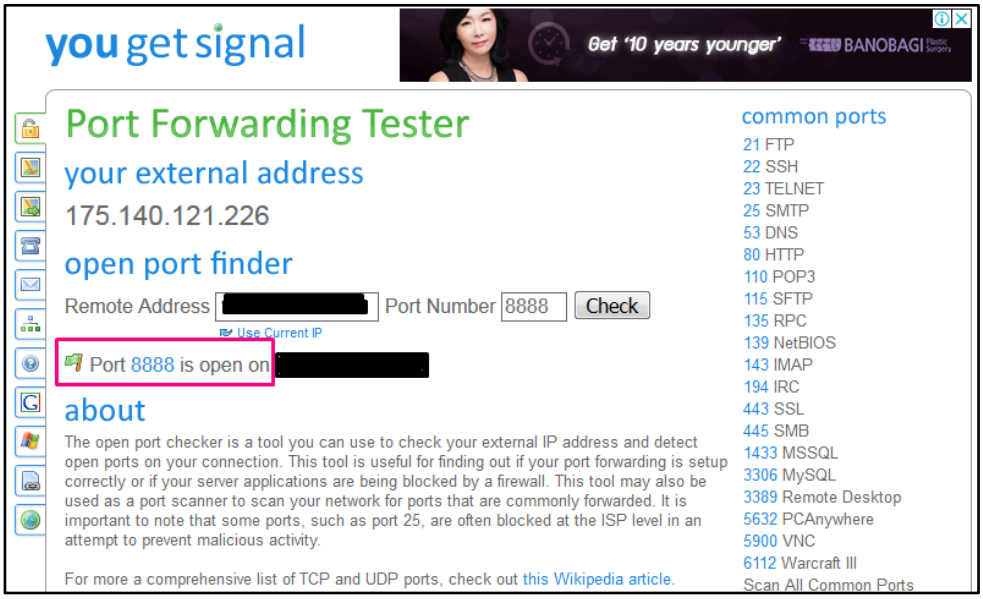


Fig 3.33: Port Forwarding test for TCP port 8888

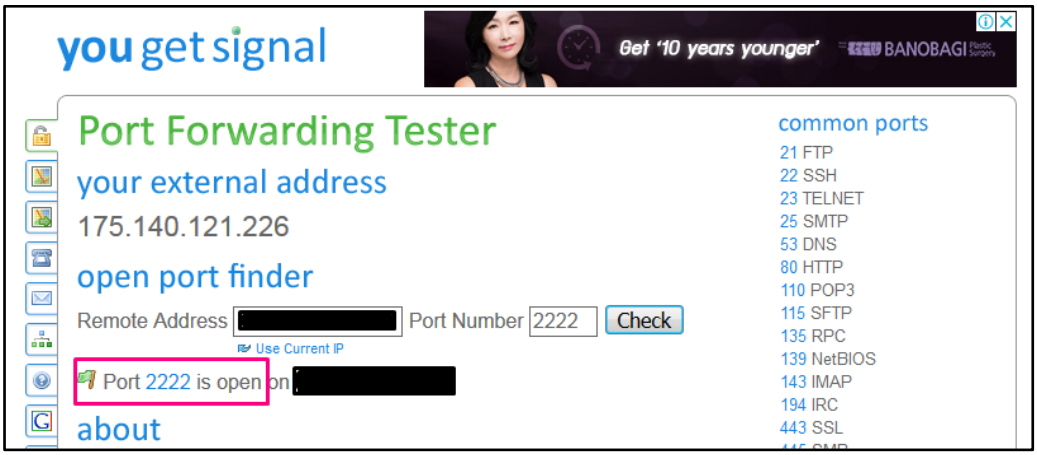


Fig 3.33: Port Forwarding test for TCP port 2222

Once the Port Forwarding is setup, you would be able to access your FreeNAS system from external remote client. From your browser, you can input http://175.140.121.226:8888 to access the FreeNAS WebGUI and from your SSH client (eg. Putty), you can set the IP of 175.140.121.226 and TCP port 2222 to access FreeNAS CLI remotely.

The Public Internet IP address may change over time before most ISPs do not prescribe a fixed IP unless it was subscribed. Thus Dynamic DNS (DDNS) automatically updates the name server records in the DNS in almost real time and provide convenience for remote access without checking the latest Public Internet IP address (if it was dynamically changed by the ISP).

The ZTE router modem provides DDNS configuration and there are several providers as shown below in Fig 3.34.

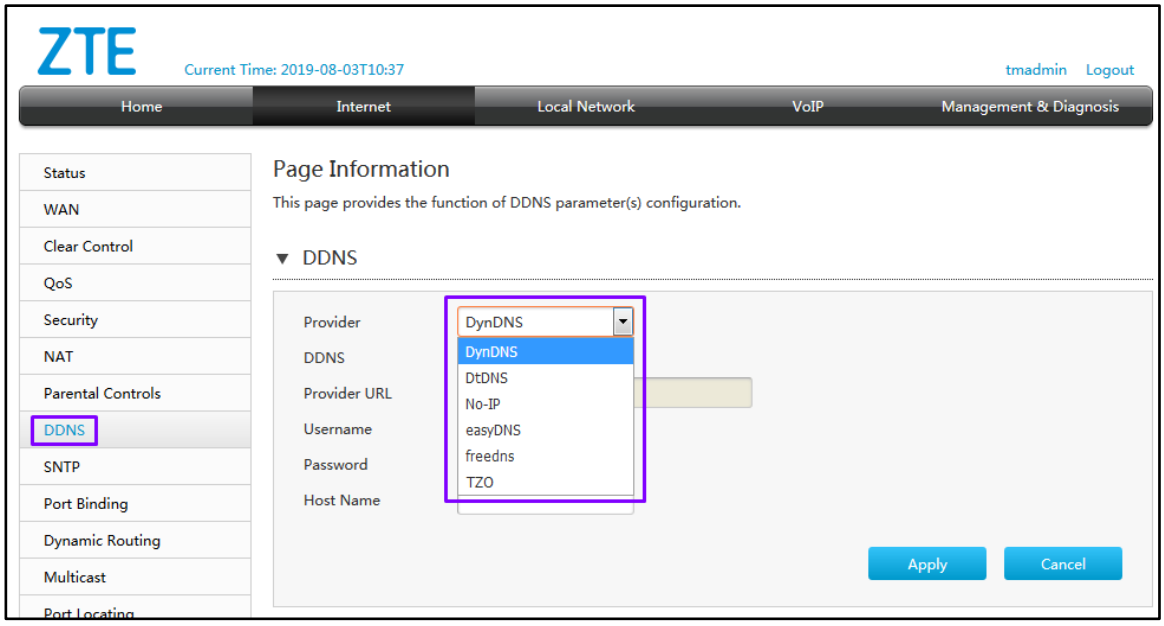


Fig 3.34: DDNS config page of the ZTE router modem

I have chosen 'No-IP' as my DDNS service provider. You have to register for a free account at http://www.no-ip.com

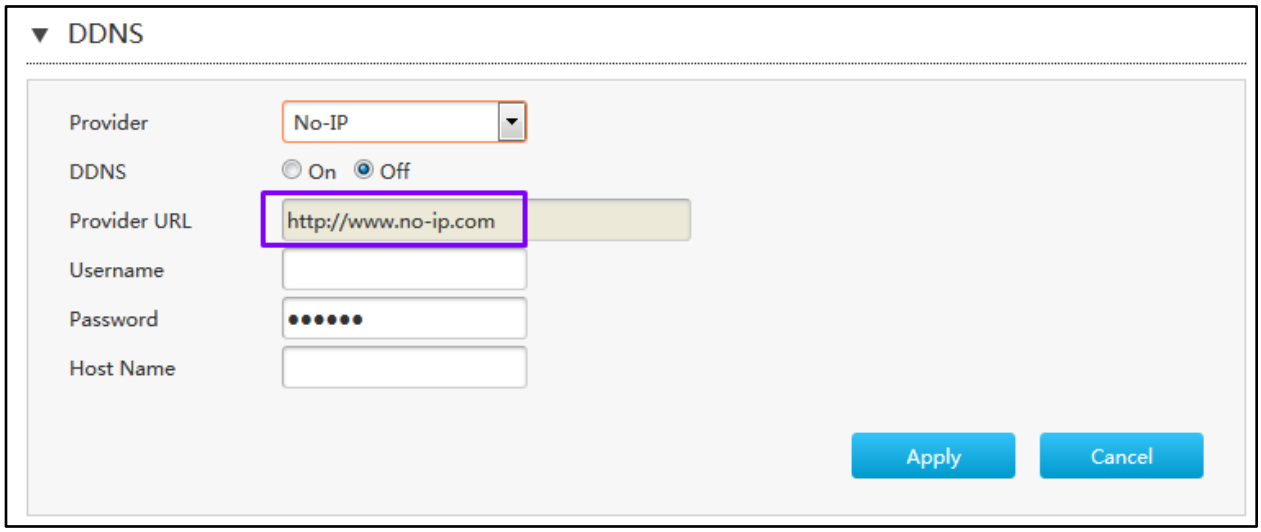


Fig 3.35: Dynamic DNS service provider No-IP

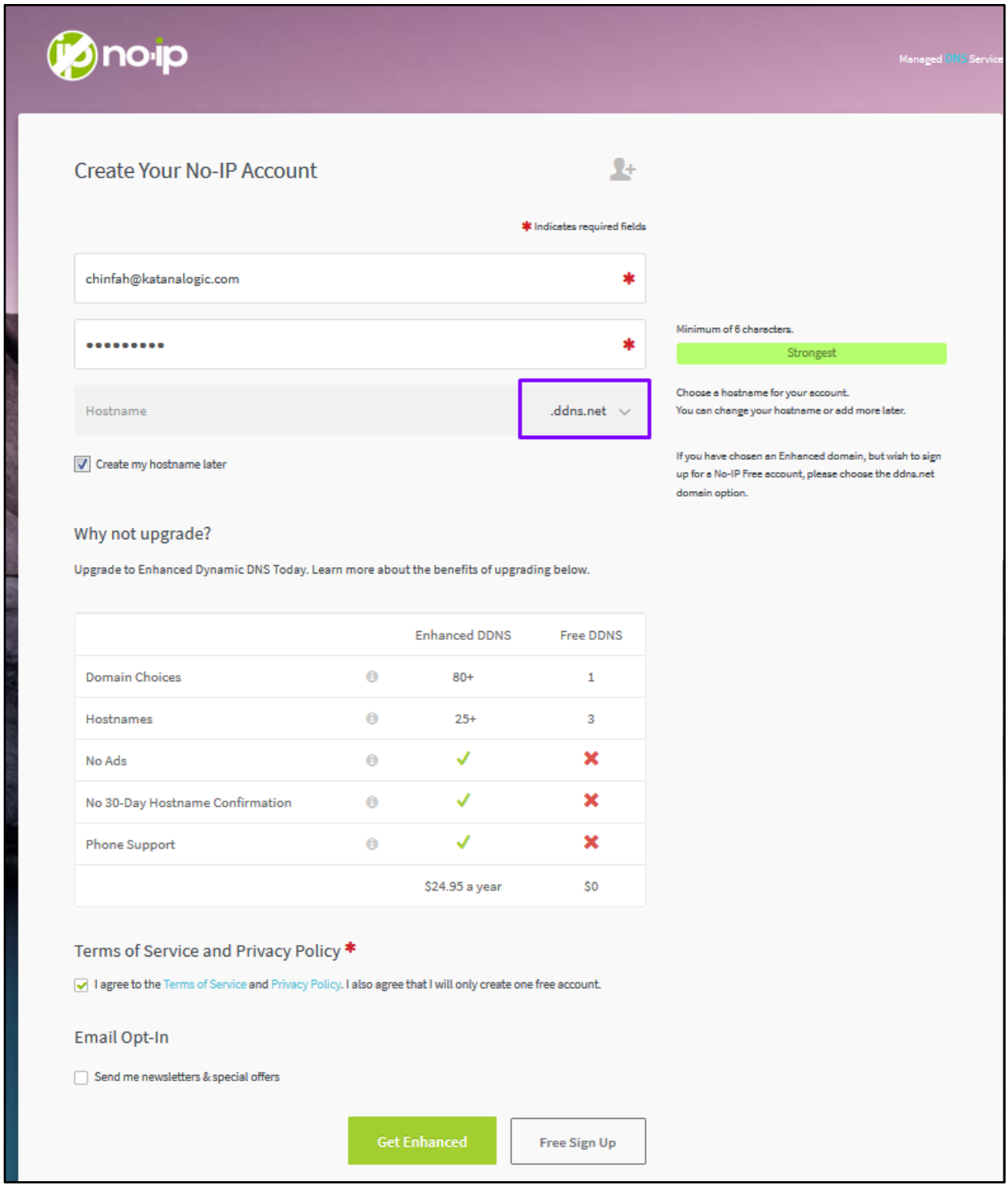


Fig 3.36: Username registration at no-ip.com

The new user account is registered with No-IP and a domain name ‘ddns.net’ is selected, shown in Fig 3.36 above.

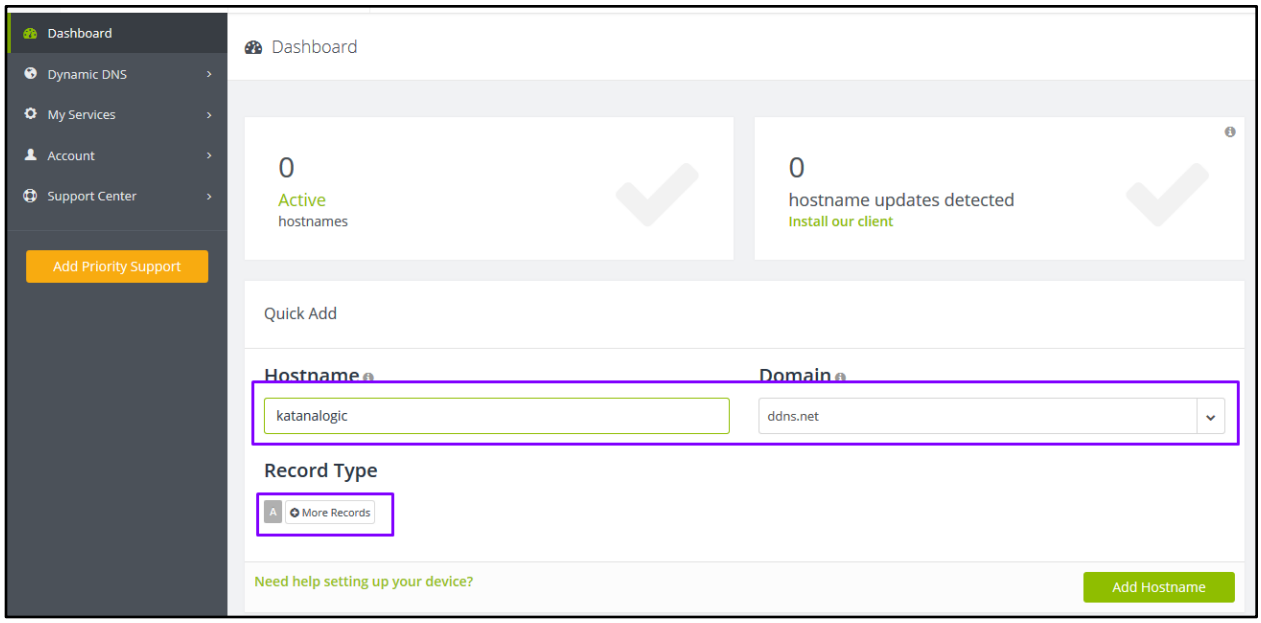


Fig 3.37: Adding a hostname to ddns.net

Add a hostname to the ddns.net domain name selected and choose the Record Type A. In this example on Fig 3.37, I have added 'katanalogic' to be katanalogic.ddns.net.

Once this has been configured without any errors, you can access the FreeNAS server remotely via http://katanalogic.ddns.net:8888 without the need to remember or even ask for the Public Internet IP Address from the end user side.

FreeNAS has a Dynamic DNS client under Services but DynDNS or DDNS is better handled at the device that is public Internet facing, which is the router modem or the firewall. Here is the FreeNAS DynDNS configuration page.

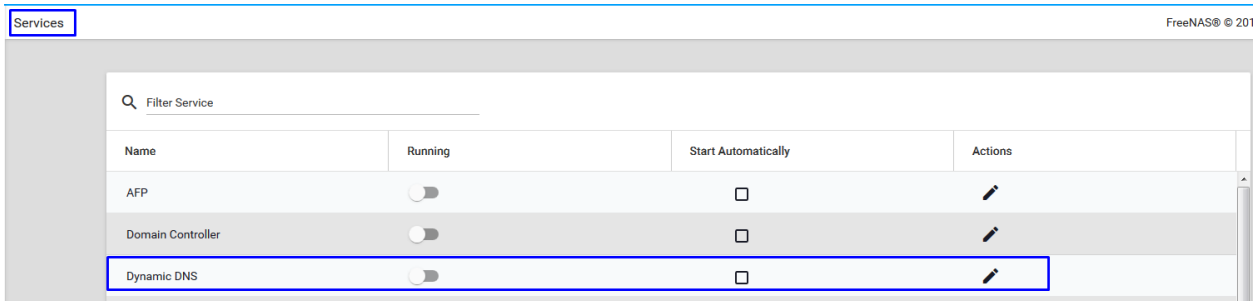


Fig 3.38: FreeNAS Dynamic DNS client

Daily Security Run info and alerts

By default, FreeNAS is configured to do a security check on the logs and can send the output to the emails of the system administrator and other intended recipients. It is configured to run and at 3.01AM every day and the cron schedule can be viewed in the /etc/crontab file, as shown in Fig 3.39 below.

```
# /etc/crontab - root's crontab for FreeBSD
#
# $FreeBSD: src/etc/crontab,v 1.33.2.1 2009/08/03 08:13:06 kensmith Exp $
#
SHELL=/bin/sh
PATH=/etc:/bin:/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/local/sbin
#
#minute hour    mday    month   wday    who     command
#
*/5     *       *       *       *       root    /usr/libexec/atrun > /dev/null 2
>&1
#
# Save some entropy so that /dev/random can re-seed on boot.
*/11    *       *       *       *       operator /usr/libexec/save-entropy > /de
v/null 2>&1
#
# Rotate log files only at midnight.
0       0       *       *       *       root    newsyslog > /dev/null 2>&1
#
# Perform daily/weekly/monthly maintenance.
1       3       *       *       *       root    periodic daily
15      4       *       *       6       root    periodic weekly
30      5       1       *       *       root    periodic monthly
```

Fig 3.39: Default crontab file in /etc/crontab

There are other periodic scripts to run weekly and monthly in the crontab file as well.

Any modifications made to the crontab file should be reinitiated with the cron restart command as shown Fig 3.40.

```
root@freenas[~]#
root@freenas[~]#
root@freenas[~]# /etc/rc.d/cron restart
Stopping cron.
Waiting for PIDS: 24334.
Starting cron.
root@freenas[~]#
```

Fig 3.40: Restarting cron

Shutdown down FreeNAS

You can shutdown FreeNAS from the 1-11 console or you can shutdown from the WebGUI interface. At the top right corner, click the 'Power' button.

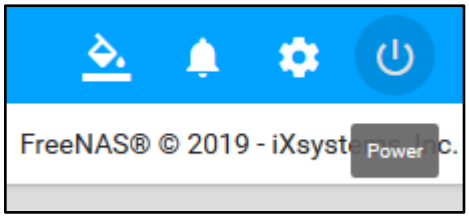


Fig 3.41: Power Button

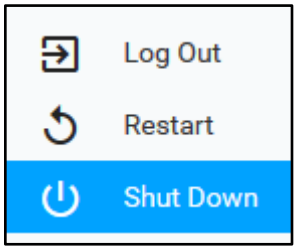


Fig 3.42: Shutdown/Restart/Log Out menu

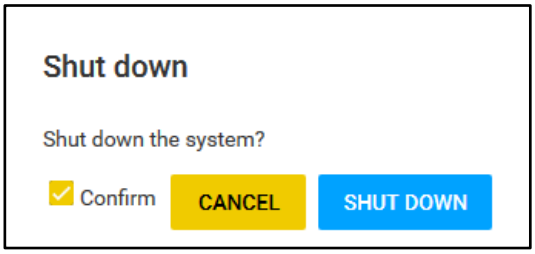


Fig 3.43: Confirm Shutdown Pop Up message

You can also shutdown FreeNAS via the console at the 1-11 menu. Choose '11' to shutdown

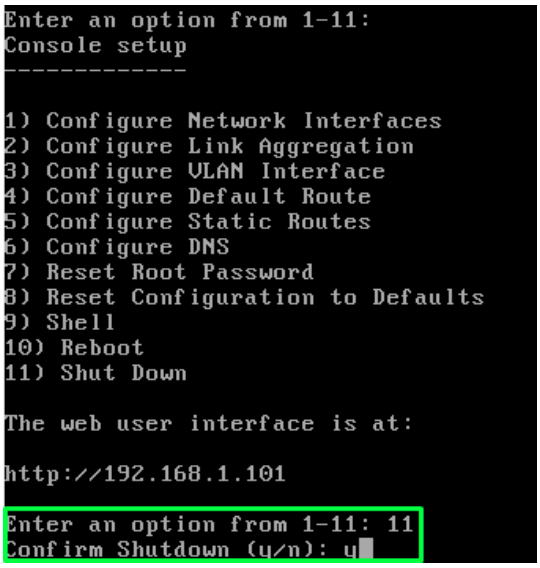


Fig 3.43: Shutdown from the console

CHAPTER 4 (FREENAS NETWORKING)

Basic Networking

Once FreeNAS has booted up in the Virtualbox, the first network interface is usually em0. em0 will pick up an IP address from the DHCP server in the network. It is best to change it to a fixed IP address so that FreeNAS as a file and storage server is known to the NAS clients and the iSCSI initiators.

The IP address can be set at the 1-11 menu of the console in Fig 4.1.

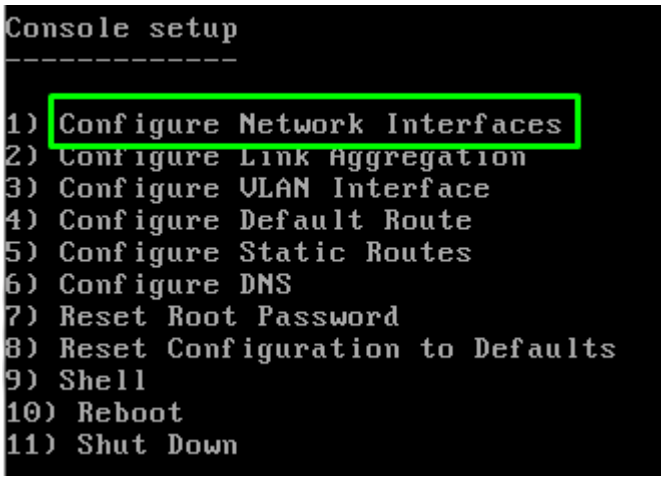


Fig 4.1: 1-11 menu to configure network interfaces

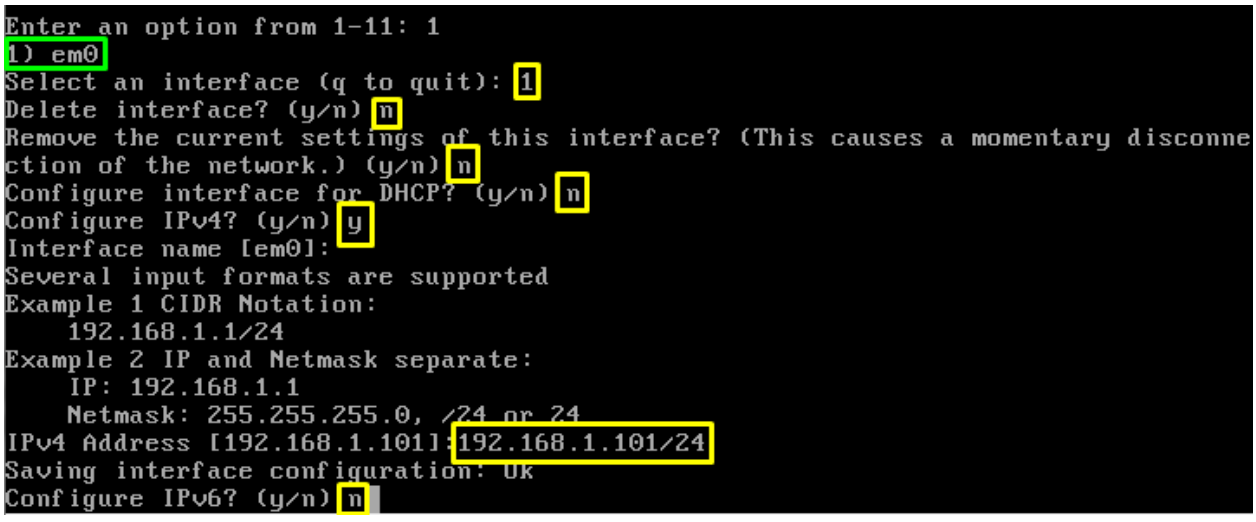


Fig 4.2: Configuring the em0 network interface

Take note of the selections of ‘y’ and ‘n’ in configuring the network interface above in Fig 4.2.

The new network IP address after the network interface configuration shown in Fig 4.3.

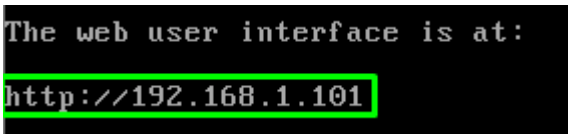


Fig 4.3: Network IP address at the console

Once the initial network interface em0 is setup, the administrator can view, manage and configure the network interface at the FreeNAS WebGUI as shown in Fig 4.4.

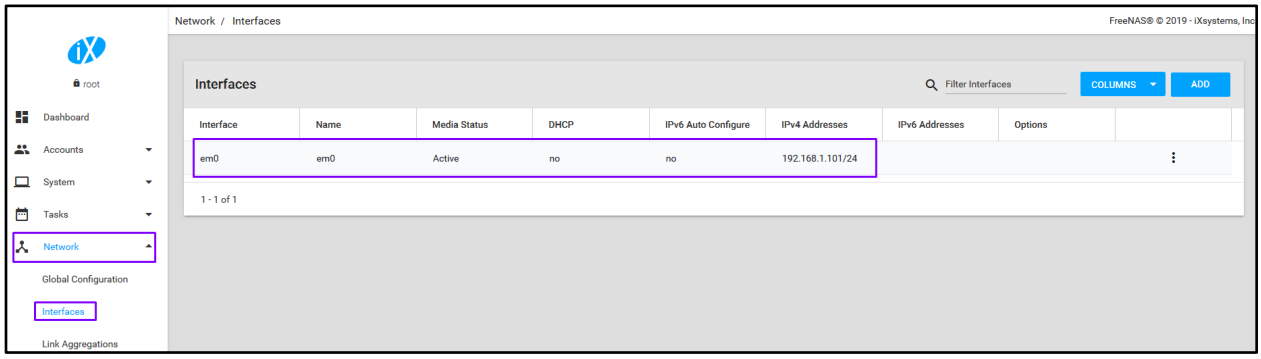


Fig 4.4: Network interface configuration page of the FreeNAS WebGUI

Once the basic networking is setup, we configure a few networking details for the global configuration. These configurations are useful and important for the proper operations for FreeNAS in a network, regardless if it is running on Virtualbox or not.

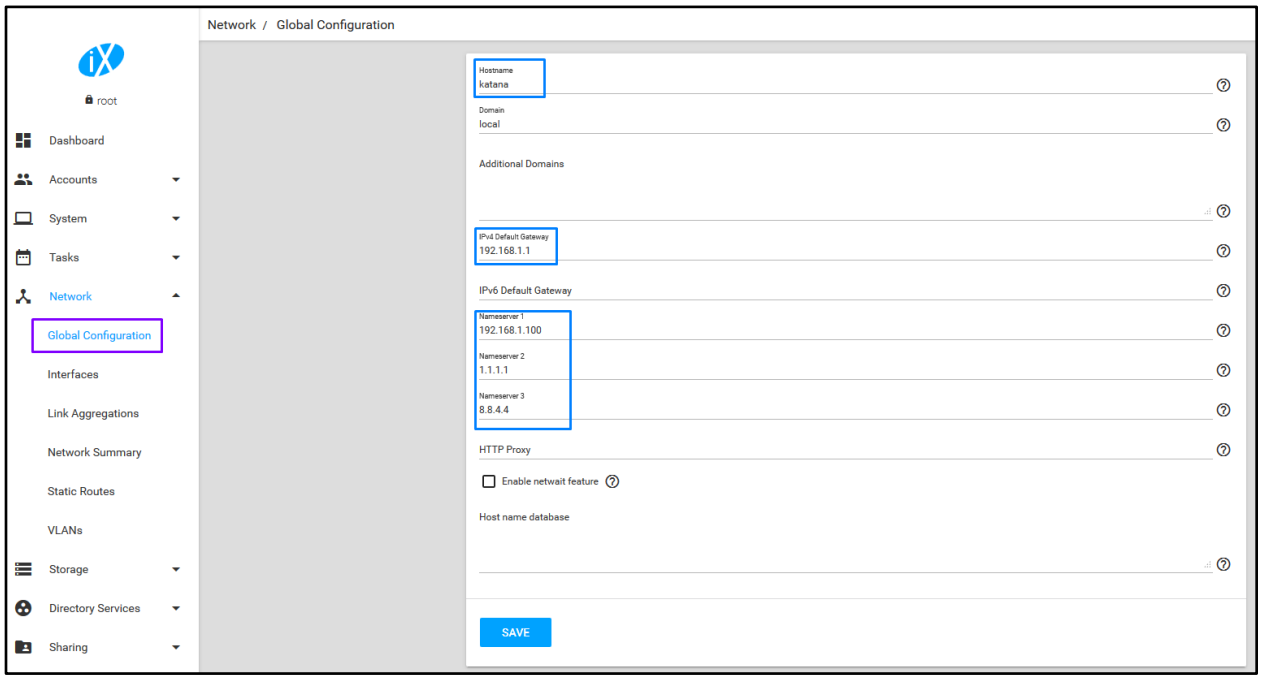


Fig 4.5: Global configuration for networking

These networking details include:

- Hostname
- IPv4 default gateway
- DNS name servers

NIC Bonding for Link Aggregation with LACP

NIC bonding is a common term used to trunk 2 or more ports into a single logical network interface or trunk. The main reason is to improve network resiliency should 1 of the network ports failed and depending on the load balancing method used, to improve network performance by combining both network ports throughput.

The term used in FreeNAS is Link Aggregation and several methods (known as lagg protocol) are supported. Lagg is the default name of the aggregated link; lagg0, lagg1 and so on. The lagg protocols available are shown below:

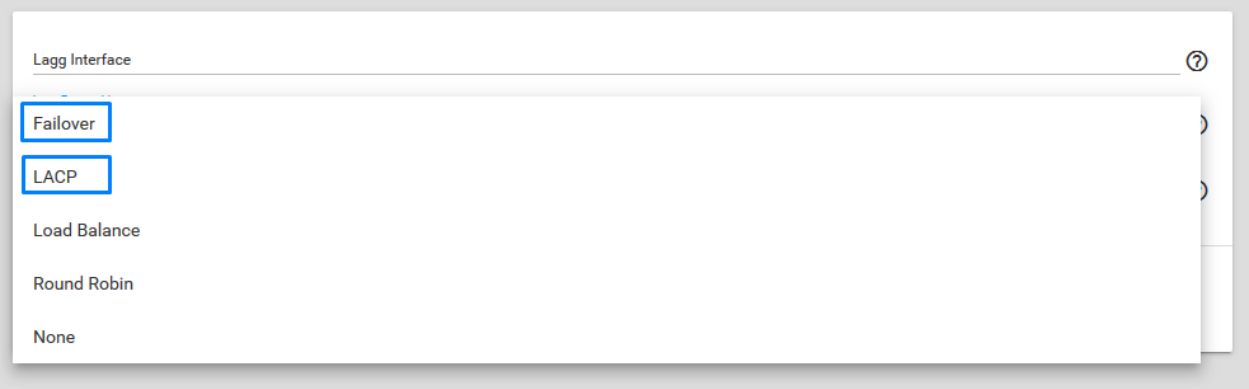


Fig 4.6: Available Link Aggregation protocols

Typically the ‘Failover’ and the ‘LACP’ are the most relevant.

In order to set up and configure link aggregation, a minimum of 2 network ports are required. LACP (Link Aggregation Configuration Protocol) requires a network switch which supports the IEEE 802.3ad, the standard for LACP.

The computer I have has 2 network interfaces - one is the on-board Gigabit Ethernet port and the other is a Wifi dongle. We can simulate a working link aggregated network interface with these 2 network resources.

In Virtualbox, we create 3 network adapters. Here is how they are setup.

1. Adapter 1 - NAT (MAC Address: 080027**CD78AA**)
2. Adapter 2 - Bridged (MAC Address: 080027**102D8A**)
3. Adapter 3 - Bridged (MAC Address: 080027**6317E9**)

It is a good idea to list down each Virtualbox adapter's MAC address. A MAC address is an Ethernet address which is unique for every physical network port globally. It is 48-bit and the first 6 hexadecimal characters denote the manufacturer and the last 6 hexadecimal denotes the unique device itself. I have put the MAC of each Virtualbox adapter above.

These correspond to the em0, em1 and em2 in the FreeNAS and help identify the interfaces when we create the link aggregation interface. We do not want confusion.

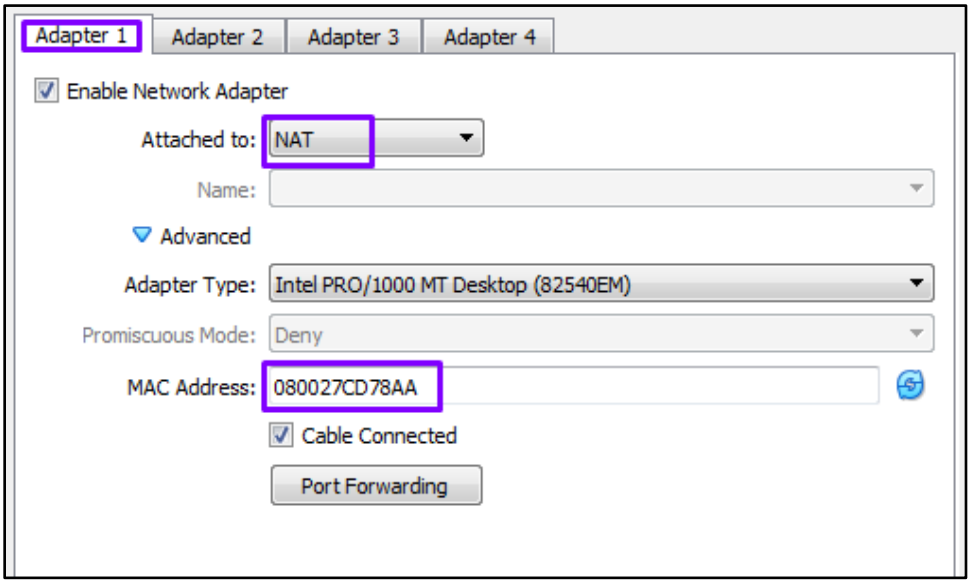


Fig 4.7: Virtualbox Network Adapter 1 (NAT)

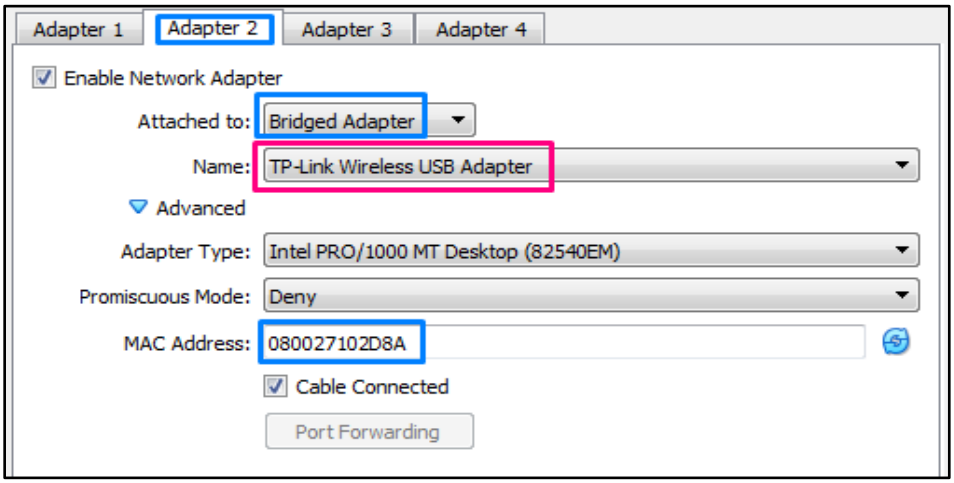


Fig 4.8: Virtualbox Network Adapter 2 (Bridged)

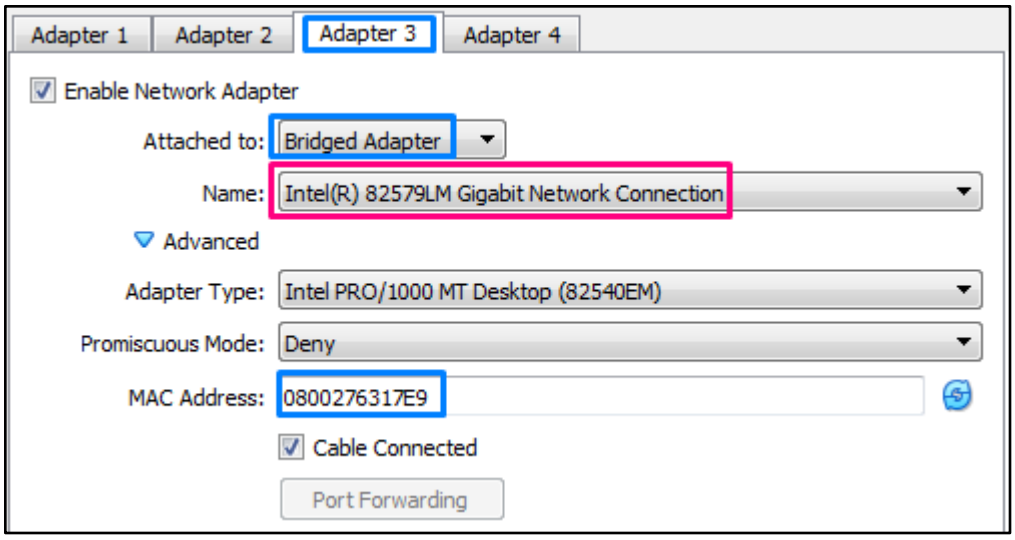


Fig 4.9: Virtualbox Network Adapter 3 (Bridged)

Virtualbox is restarted with the 3 new network adapters. At the 1-11 menu console, we have to remove the earlier network interface which was manually configured. Link aggregation will fail if any network interface to be included in the configuration is manually configured. We have to remove the network interface before setting up link aggregation.

Select ‘9’ to enter into the command shell. At the prompt

ifconfig -a | more

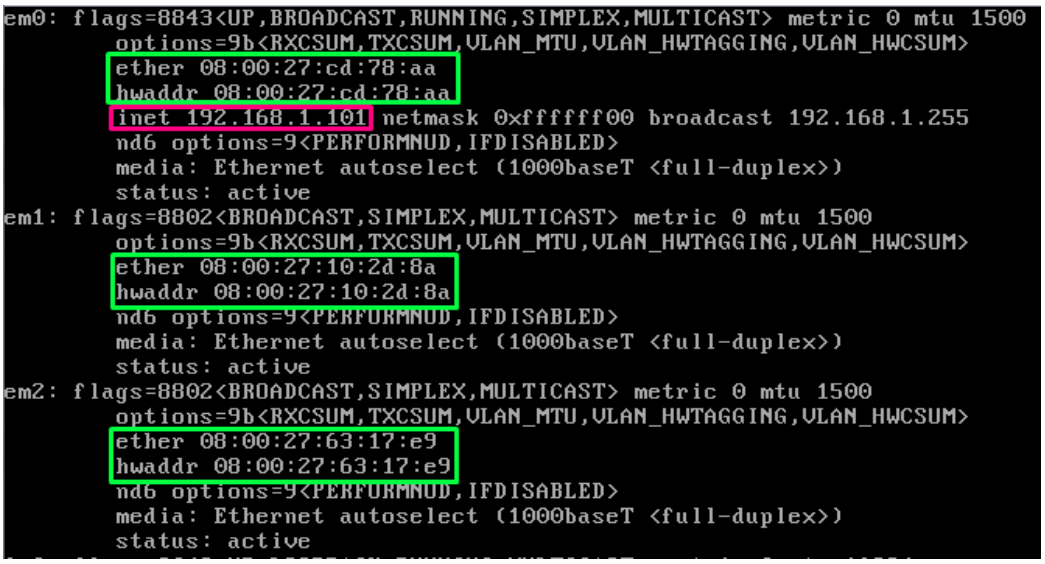


Fig 4.10: View the Ethernet or MAC address of the FreeNAS network interfaces

From Fig 4.10 above, there are 3 interfaces, em0, em1, em2 which corresponds to the respective Virtualbox network adapters 1, 2 and 3. In production environment, the network interface to physical network ports may not be so straightforward. That is why keeping score of the MAC or Ethernet address is useful for us to track the right port to the right interface.

In our case, both em1 and em2 have not been configured and thus, are available to be included in the link aggregation configuration.

Select ‘2’ to ‘Create Link Aggregation’

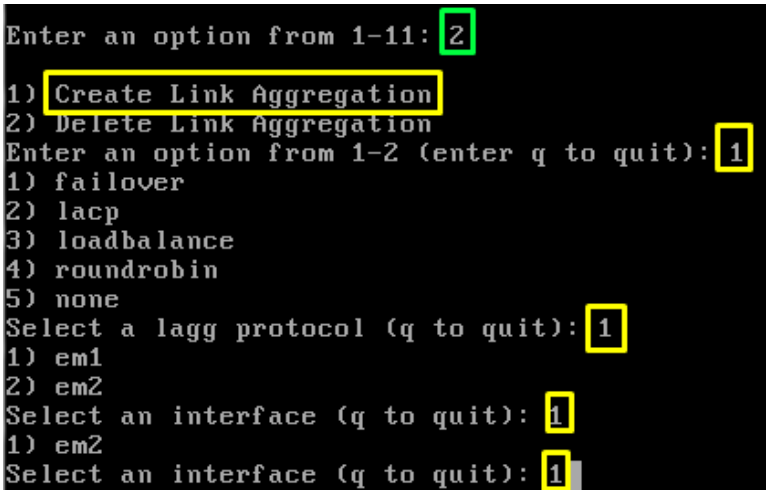


Fig 4.11: Create Link Aggregation

1. Choose ‘1’ to ‘Create Link Aggregation’

2. Choose ‘1’ for ‘failover’ -
3. Choose ‘1’ to include ‘em1’
4. Choose ‘1’ to include ‘em2’

Once the lagg0 is created, select ‘1’ in the 1-11 menu and configure the new interface as per usual, as shown in Fig 4.12 below:

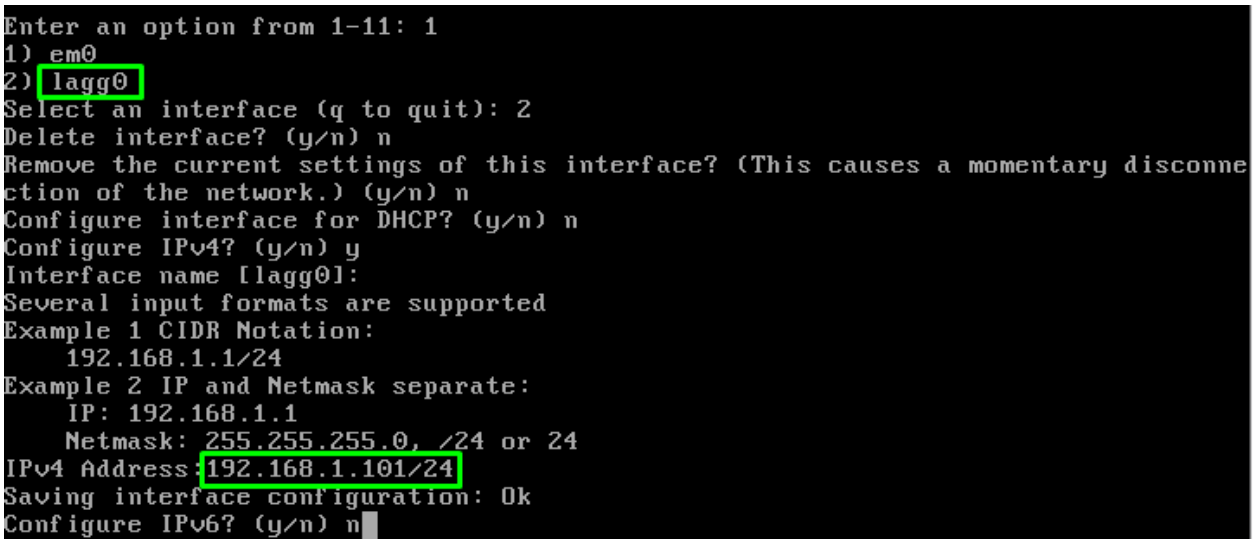


Fig 4.12: Setting the IP address of lagg0

Additional members can be added to the link aggregation after its creation. With the FreeNAS WebGUI, as in Fig 4.13 below:

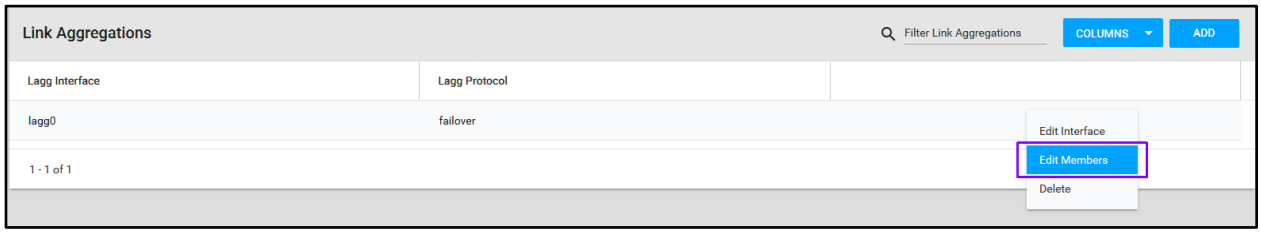


Fig 4.13: Link Aggregation - adding new members

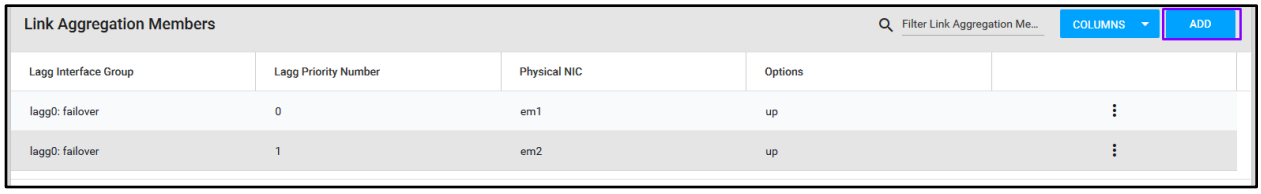


Fig 4.14: Link Aggregation - adding new members

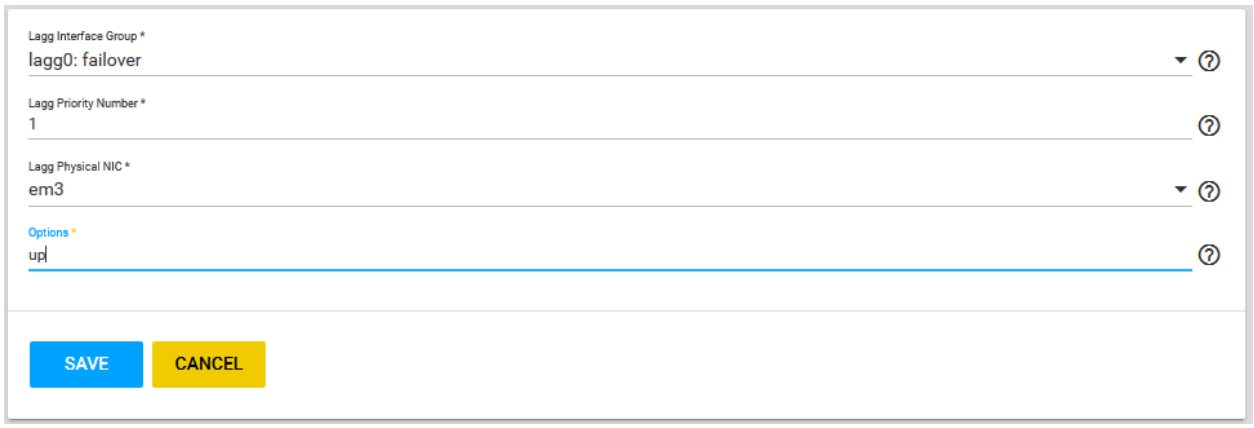
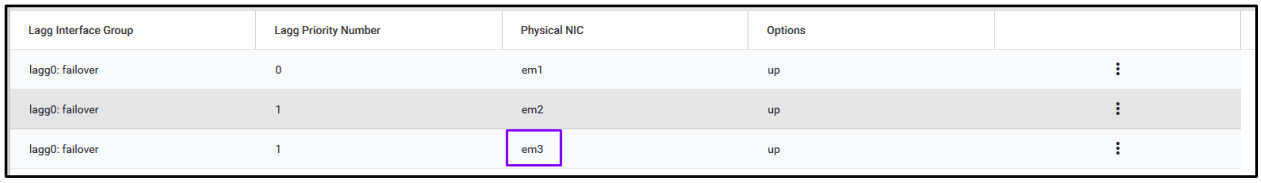


Fig 4.15: Link Aggregation - Adding new members

We select 'lagg0:failover' as the Lagg Interface group with the priority as '1'. We add em3 as the third interface to the lagg0 with the options 'up'.



Lagg Interface Group	Lagg Priority Number	Physical NIC	Options	
lagg0: failover	0	em1	up	⋮
lagg0: failover	1	em2	up	⋮
lagg0: failover	1	em3	up	⋮

Fig 4.16: New member em3 to the lagg0 interface

Once em3 is added, you should be able to ping to the configured IP address of lagg0.

Virtual LAN (VLAN)

The IEEE 802.1q denotes the VLAN standard and FreeNAS supports VLAN. VLANs create segmentation of network links into virtual 'subnetworks'. Each VLAN is a separate broadcast domain that benefits the performance, security and management of each virtual network. This allows FreeNAS to serve different network services and network groups with a large physical network.

To setup VLAN, ‘Add’ a new VLAN as shown in Fig 4.17 below.

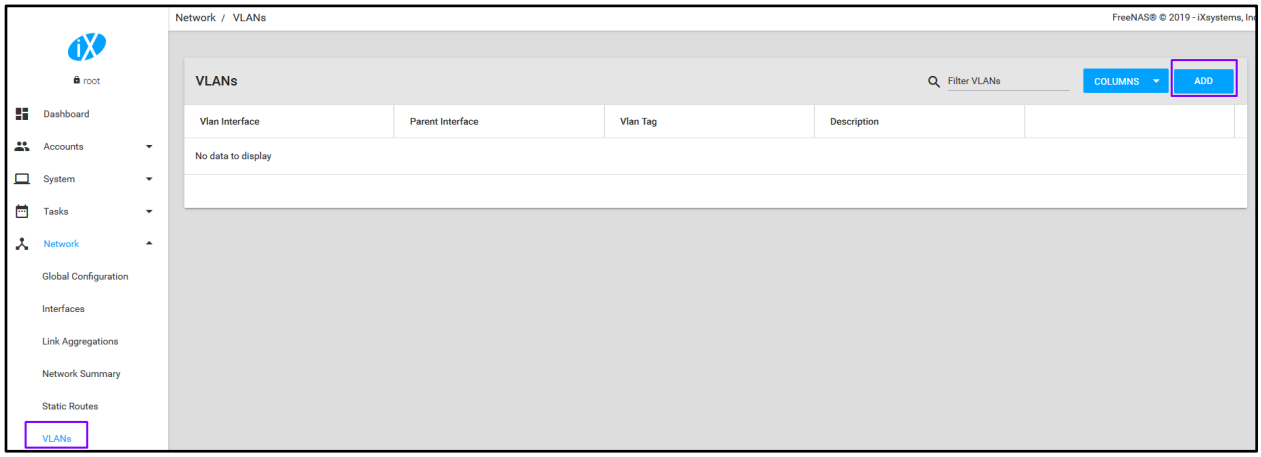


Fig 4.17: Add a new VLAN

Give the ‘Virtual Interface’ a name linked to a ‘Parent Interface’ and a ‘VLAN Tag’. By default, the ‘Priority Code Point’ is ‘Best effort’ delivery.

Fig 4.18: New VLAN20 configuration

The example Fig 4.18 configures ‘vlan20’ under the parent of ‘lagg0’ and a VLAN Tag of ‘20’.

Then we can setup the VLAN20 with network and IP address.

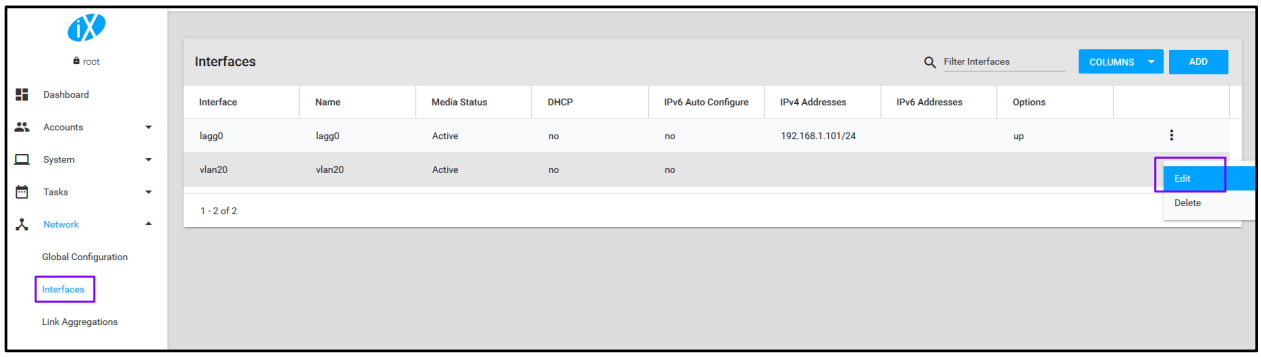


Fig 4.19: Configuring the network for VLAN20

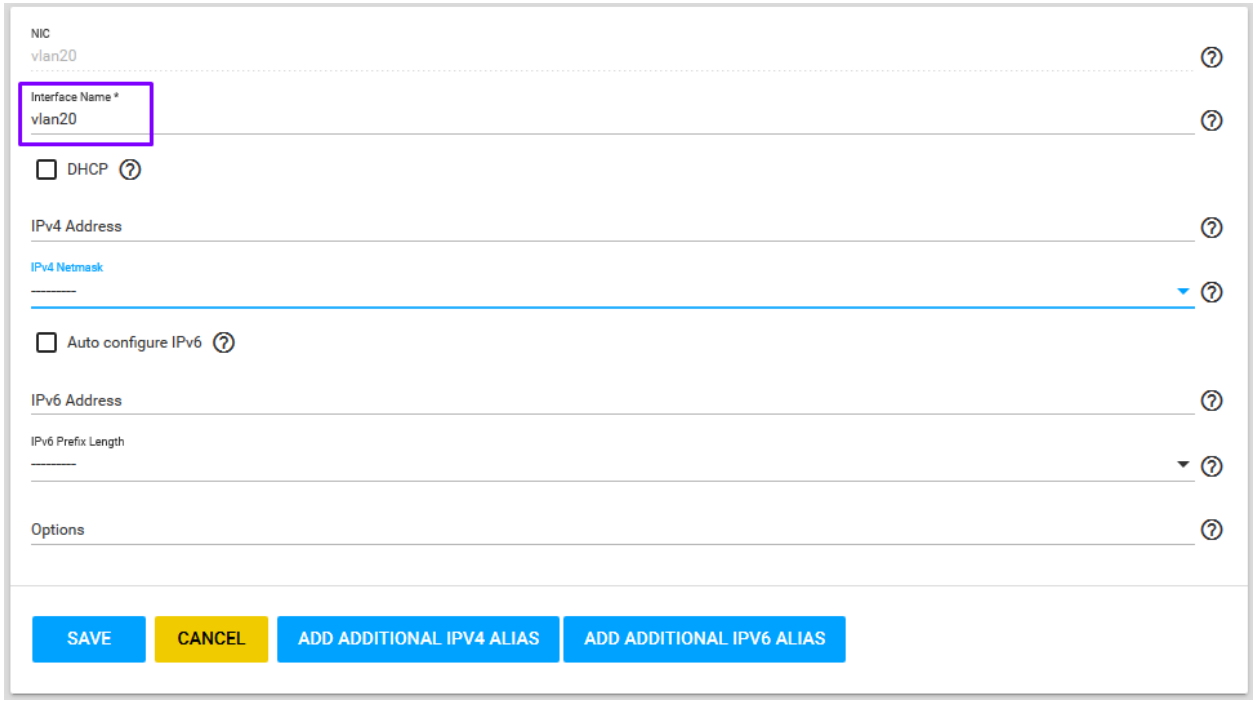


Fig 4.20: Network setting for VLAN20

The VLAN should be configured and ready to be used.

IPMI

IPMI (Intelligent Platform Management Interface) is not a feature in VirtualBox. It is an out-of-band system management interface to monitor and administer the system independently in a data center. It provides remote access to the system CPU, BIOS and other system components, and it also allows system administrators to remotely power off and power on the system.

Since the IPMI interface is not present for Virtualbox, the IPMI interface configuration on the FreeNAS WebGUI does not exist as well. However, if the IPMI configuration does exist, it would look like this in Fig 4.21

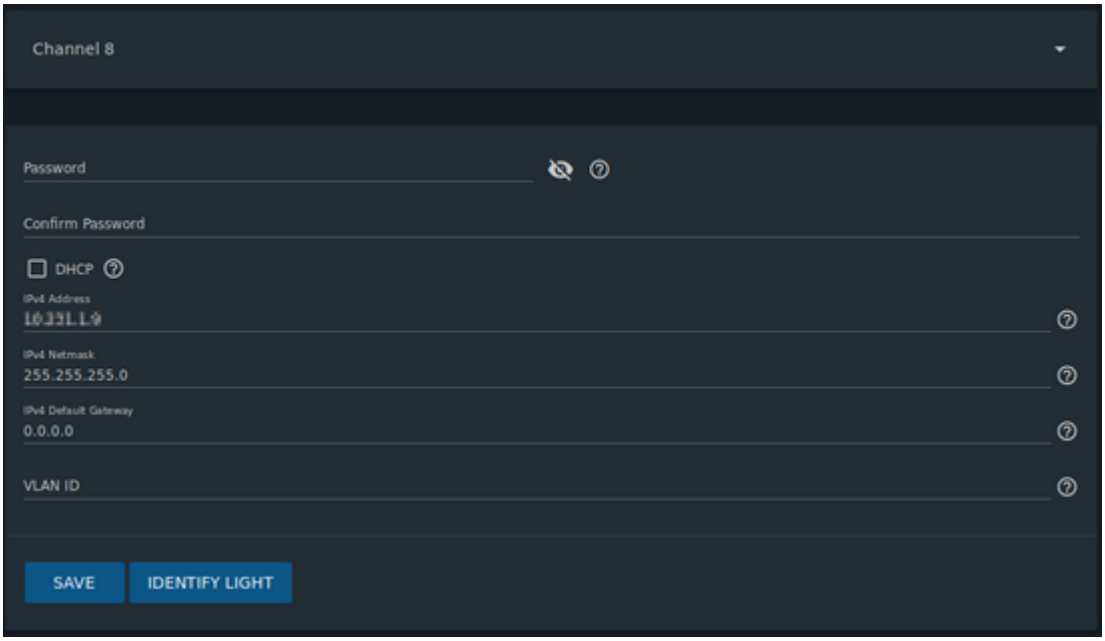


Fig 4.21: IPMI configuration page on FreeNAS

The administrator has to set right channel with a unique password per channel. An IP address is configured for the IPMI channel with the default gateway and a VLAN ID if required.

An example to a Supermicro server's IPMI is shown in Fig 4.22 and Fig 4.23 below:

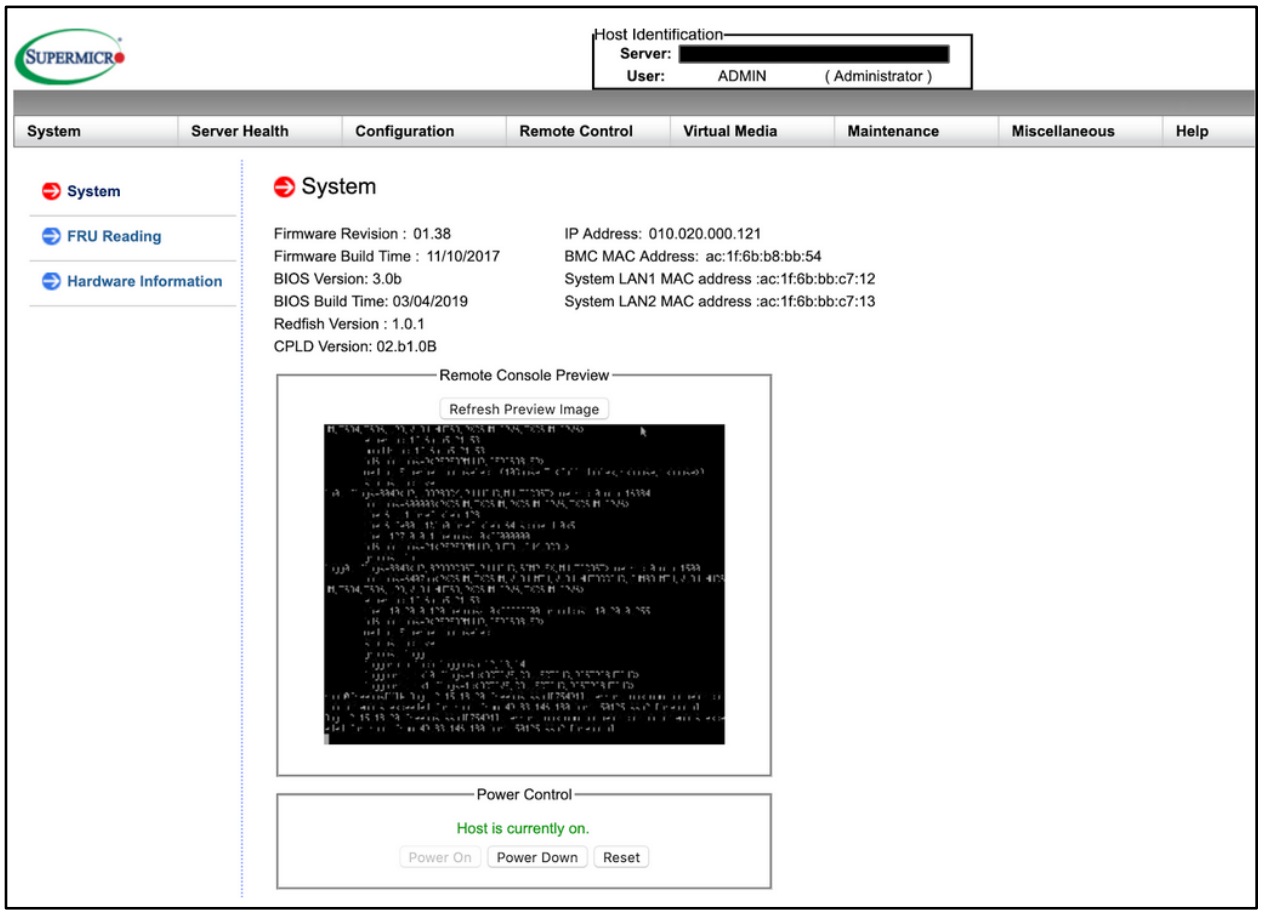


Fig 4.22: Supermicro IPMI Dashboard

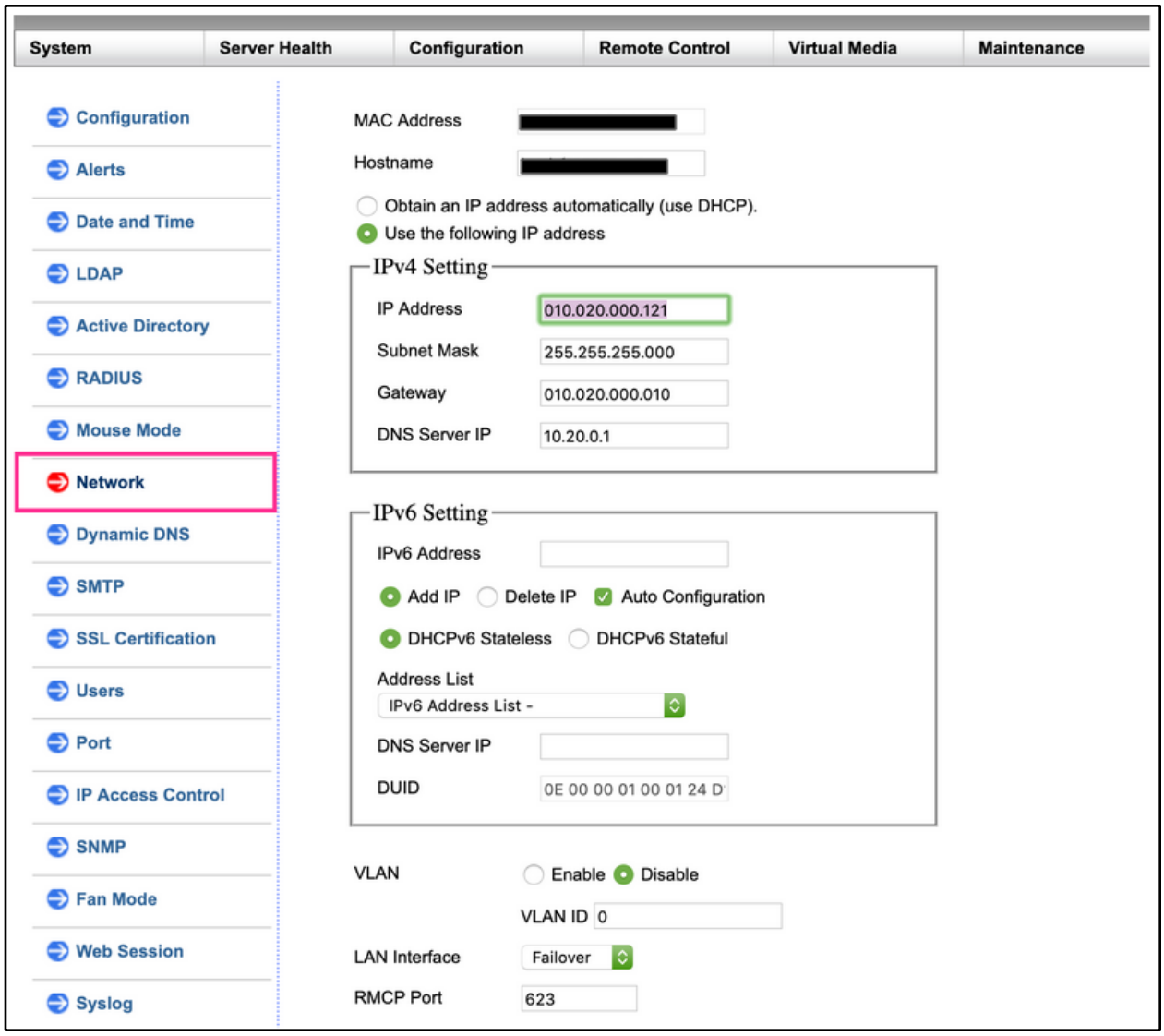


Fig 4.23: IPMI Network Configuration

CHAPTER 5 (FREENAS RAID & STORAGE MANAGEMENT)

FreeNAS Storage Architecture and RAID

To use storage effectively, we must understand the concept of how the ZFS file system is structured. There are several key data structures to know

- zpool
- vdev (virtual device)
- dataset
- zvol

There are other structures but the 4 data structures are the most relevant.

Hard disks or solid state drives are aggregated together to create a vdev (virtual device). A vdev can be configured with different RAID (redundant array of independent disks) levels for 4 main criteria listed below:

- Capacity
- Performance
- Resiliency
- Cost

One or more vdevs are combined to create a zpool. From the zpool, one or more datasets or zvols can be “carved out” to create NAS shares or SAN volumes. The diagram below Fig 5.1 explains the key structures

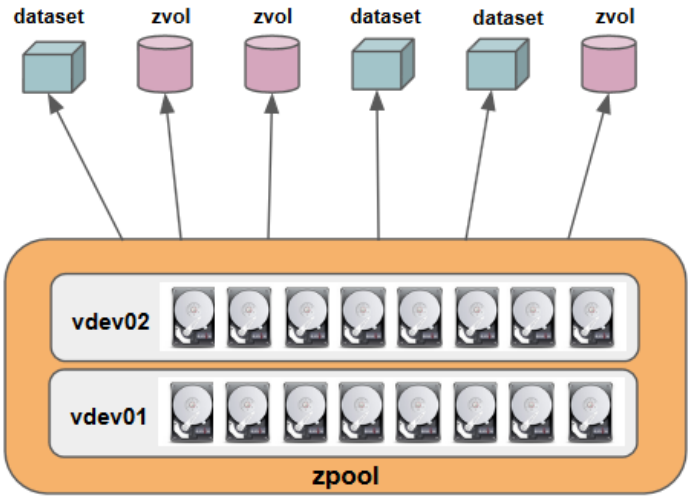


Fig 5.1: zpool, vdev, dataset and zvol structures

ZFS in FreeNAS offers several RAID levels. These are RAID-0 (striped), RAID-1 (mirrored), RAID-Z1, RAID-Z2, and RAID-Z3. RAID-Z1 and RAID-Z2 are equivalent to RAID-5 (distributed single parity) and RAID-6 (distributed dual parity) respectively, but with unique features such as

- Elimination of RAID write hole
- Variable width full stripe write
- Elimination of read-modify-write penalty
- Hierarchical checksum for all data and metadata blocks
- More info from Jeff Bonwick's old blog at https://blogs.oracle.com/bonwick/raid-z-v6. Jeff Bonwick was one of the key architects of the ZFS design

Much has been written about the different RAID levels and how they relate to the 4 criteria of Performance, Capacity, Resiliency and Cost. These will not be covered in this book. A good resource from iXsystems is found from https://static.ixsystems.co/uploads/2018/10/ZFS_Storage_Pool_Layout_White_Paper_WEB.pdf

FreeNAS zpool and disks

To start with storage, 12 disks (d0-d11) have been created in Virtualbox as in Fig 5.2.

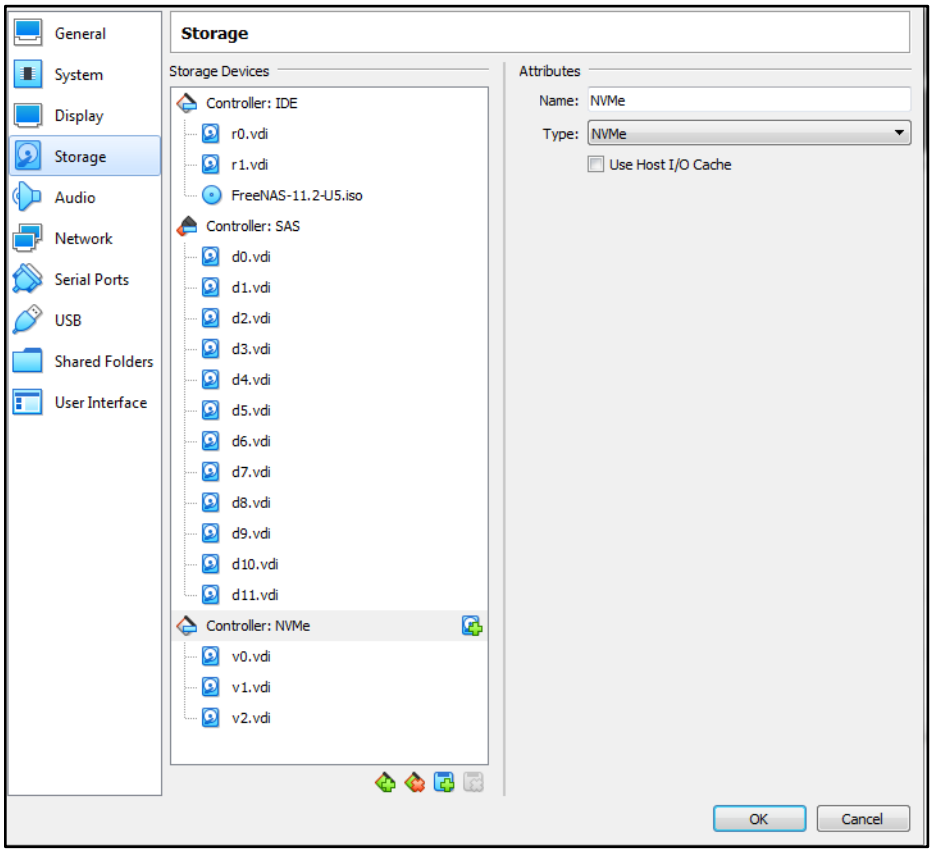


Fig 5.2: 12 disks created in Virtualbox

There are a lot of discussions about the number of disks used in the RAID level in a vdev. There are many different sets of reasoning to select the number of disks, and they depend on the 4 criteria I have listed in the previous section - Performance, Capacity, Resilience and Cost.

Since we are using Virtualbox, the number of disks selected for a chosen RAID level should not matter much, since it is unlikely that FreeNAS is used in a production environment. However, in a real-life production environment, the number of disks matters. In the past, I have chosen one of the well known "formula" to select the number of disks as shown below:

Formula	RAID Type	n=1	n=2	n=3	n=4	n=5	n=6
$2^n + 1$	RAID-Z	3	5	9	17	33	
$2^n + 2$	RAID-Z2	4	6	10	18	34	
$2^n + 3$	RAID-Z3	5	7	11	19	35	

Fig 5.3: RAID level and number of disks per RAID level

The area in **RED** in Fig 5.3 is NOT RECOMMENDED. The reason is, as the number of disks increases, the probability of a disk failure in the vdev is higher, and the vdev health and resiliency is paramount to the storage pool.

In my real life experience, I usually work with Supermicro or Quanta system chassis which has either 8, 12, 16 or 24 3.5" drive slots. The hard disk drives I select are usually 4, 6, 8, 10 or 12TB and I have to consider keeping 1 or 2 hot spare disks to fit into chassis. The above formula works well but you may have to make some compromises for the number of drive slots available in the Supermicro or Quanta chassis.

In the FreeNAS WebGUI, you can view the disks available for FreeNAS storage.

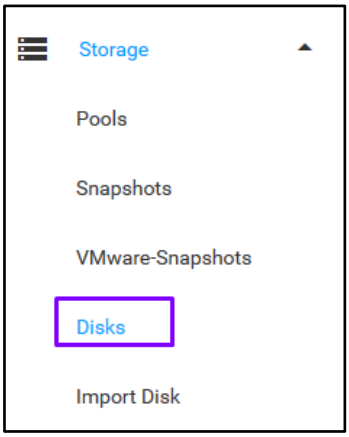
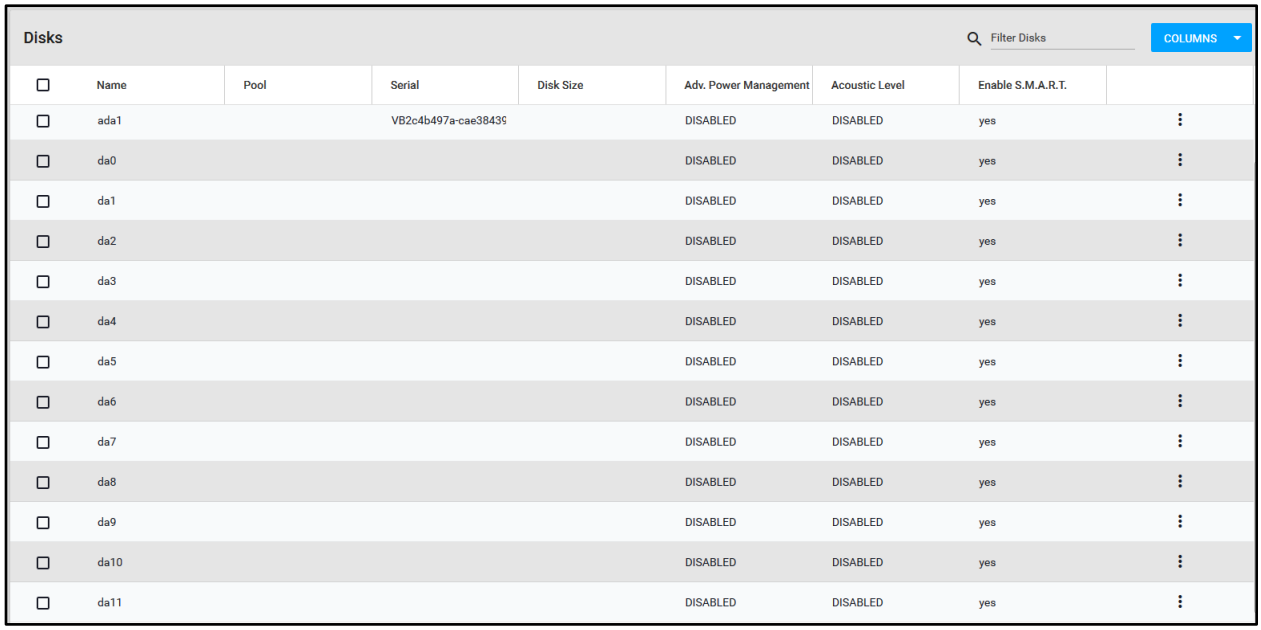


Fig 5.4: FreeNAS WebGUI view disks



Name	Pool	Serial	Disk Size	Adv. Power Management	Acoustic Level	Enable S.M.A.R.T.
ada1		VB2c4b497a-cae8439		DISABLED	DISABLED	yes
da0				DISABLED	DISABLED	yes
da1				DISABLED	DISABLED	yes
da2				DISABLED	DISABLED	yes
da3				DISABLED	DISABLED	yes
da4				DISABLED	DISABLED	yes
da5				DISABLED	DISABLED	yes
da6				DISABLED	DISABLED	yes
da7				DISABLED	DISABLED	yes
da8				DISABLED	DISABLED	yes
da9				DISABLED	DISABLED	yes
da10				DISABLED	DISABLED	yes
da11				DISABLED	DISABLED	yes

Fig 5.5: View all the disks

Select ‘Pool’ on the left navigation bar and select ‘Add’ on the top right corner of the FreeNAS WebGUI.

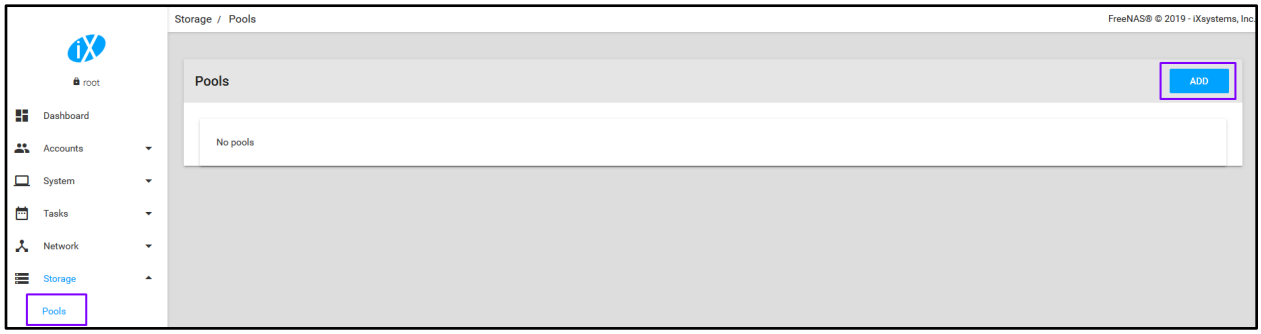


Fig 5.6: Add a zpool

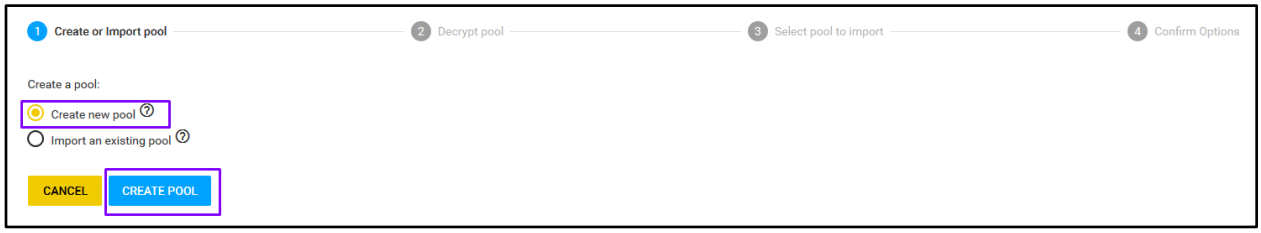


Fig 5.7: Create a new pool

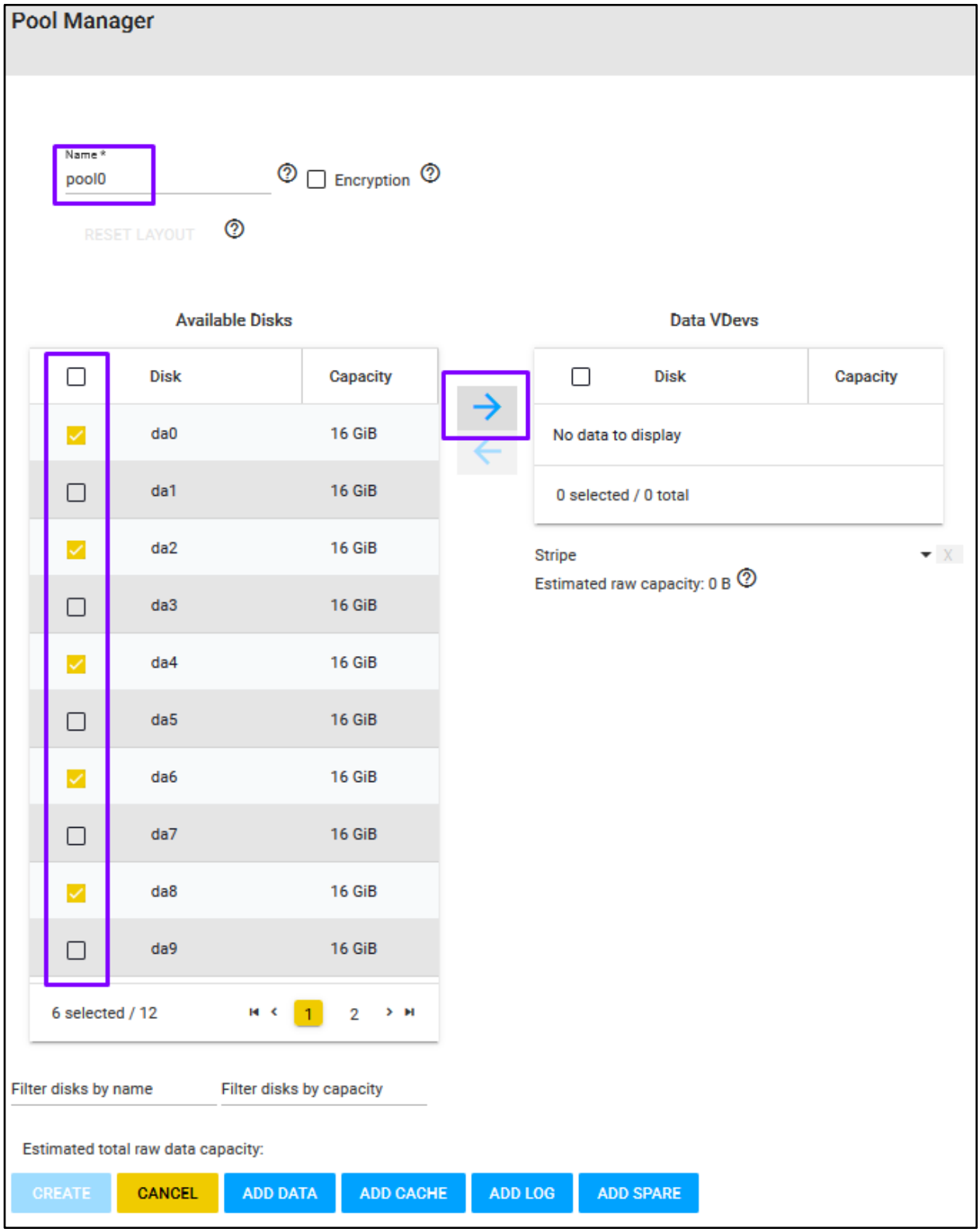


Fig 5.8: Pool Manager

In the Pool Manager page, provide a name for the zpool. In my example in Fig 5.8 above, I have named the zpool ‘pool0’. Select the number of numbers on the left side table and click the Right Arrow → to create the Data vdevs.

I have chosen 6 drives (da0, da2, da4, da6, da8 and da10) with a RAID-Z2 configuration for the first vdev of pool0. Select the drives by checking on the checkboxes and select the RAID level.

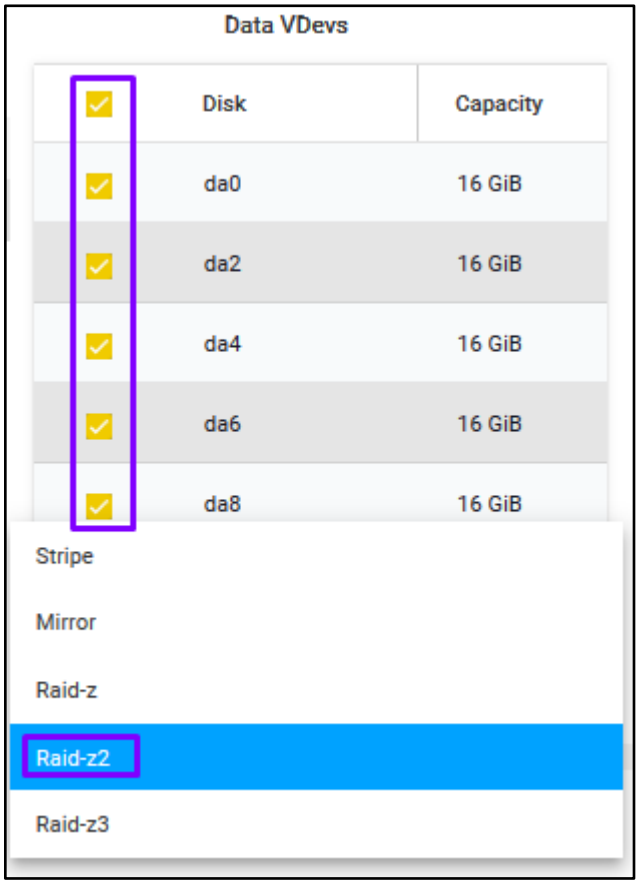


Fig 5.9: Creating the data vdevs

Click on ‘Create’ to create the new zpool as in Fig 5.10 below.

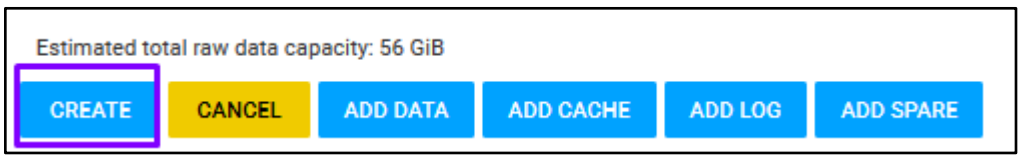


Fig 5.10: Ready to create the new zpool

A pop-up message will appear asking you to confirm the operation.

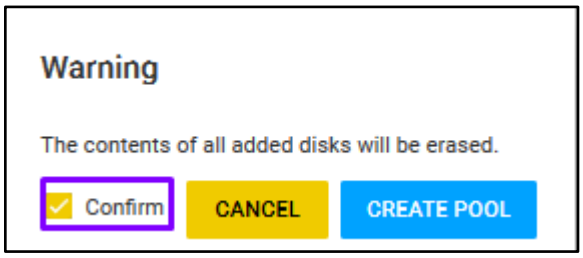


Fig 5.11: Create pool confirmation pop up

A few seconds later, a new zpool, called pool0 is created.

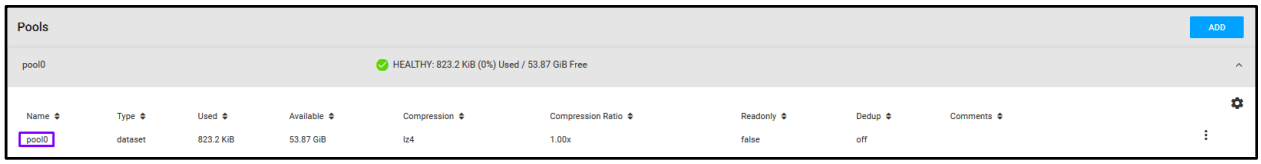


Fig 5.12: A new zpool, pool0 is added

The creation of new datasets for NAS and zvol for SAN will be discussed in another chapter of this book.

SMART

S.M.A.R.T. (Self-Monitoring, Analysis and Reporting Technology) is a monitoring system for hard disk and solid state drives. It is widely used by server and storage vendors to provide status and alert checks for disks, with variation of each supporting vendors. It is extremely useful to provide details of the disks in operations, and also predict potential failure of a disk.

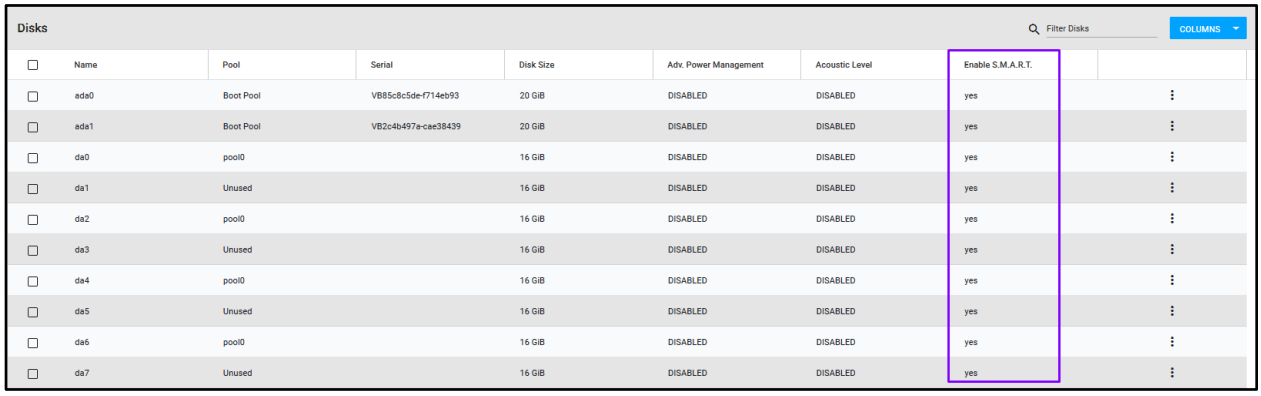


Fig 5.12: FreeNAS showing the S.M.A.R.T. enabled for all the disks

The virtual disks created in Virtualbox and presented to FreeNAS does “support” SMART in a basic manner. The command ‘smartctl’ in FreeNAS, as shown in Fig 5.13, below shows some basic details of disk /dev/da0.

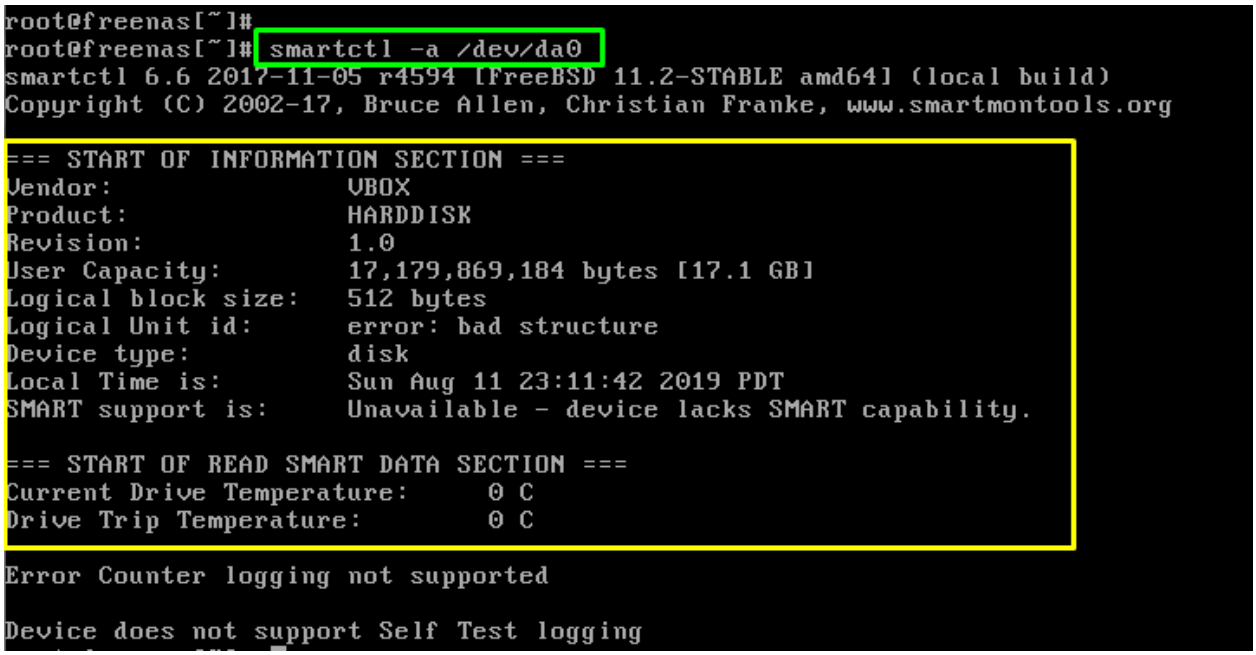


Fig 5.14: The ‘smartctl’ command showing details of disk /dev/da0

However the information provided for the virtual disks in Virtualbox is limited and does not mirror the SMART capability of the disks in the real world. Thus, the SMART capability of the ‘disks’ in FreeNAS on Virtualbox is limited as well.

Hot Spares

Disks will fail. When a disk fails, it affects the health of the vdev. Unless the vdev was configured with RAID-0, the failed disk in a vdev can be replaced logically via an unused, hot spare disk. Hot spares must be presented to the respective zpool in order for them to be made available to all the vdevs in the zpool.

Hot spare disks can be added during the creation of the zpool or they can be added to an existing zpool.

To add hot spares to an existing zpool in operation, select the “Settings” icon on the top right corner. Choose ‘Extend’ as shown in Fig 5.15 below:

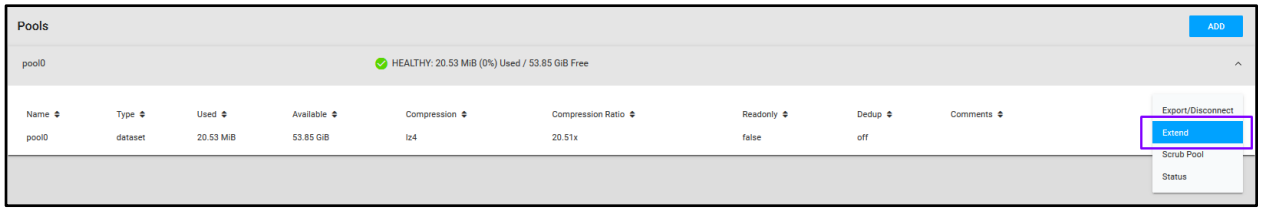


Fig 5.15: Extend hot spares into the zpool

A pop-up message appears to warn of the actions and the effects of extending the zpool. Click ‘Continue’.

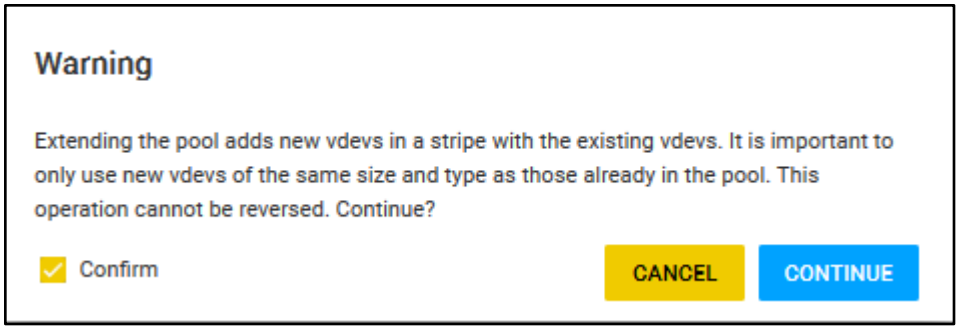


Fig 5.16: Pop-up warning of extending the pool

At the bottom of the page, select ‘Add Spare’.

Fig 5.17: Add spare

A new spare vdev table appears. Select the drives to be added as hot spare disks

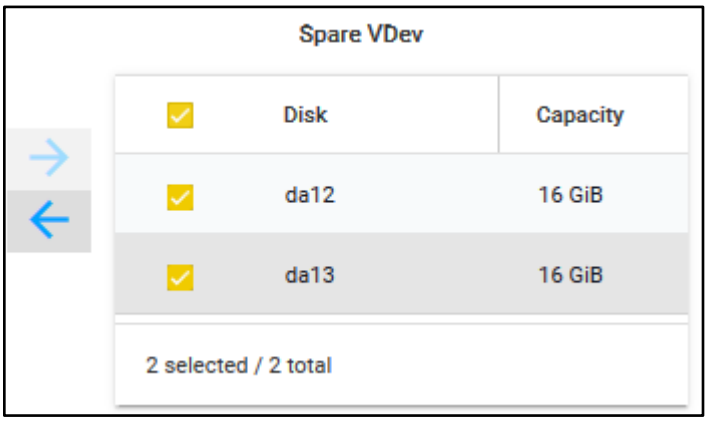


Fig 5.18: Select drives as hot spare disks

Select ‘Extend’ at the bottom of the page.

Fig 5.19: Extend to add the hot spare disks

A pop-up message appears. Click ‘Confirm’. The hot spare disks are added to the existing pool0.

NVMe drives in Virtualbox

In preparing for the next section on storage performance, NVMe drives are added to Virtualbox. NVMe drives has to be incorporated with Oracle VM Virtualbox Extension Pack. Download the Extension Pack from the https://www.virtualbox.org/wiki/Downloads .

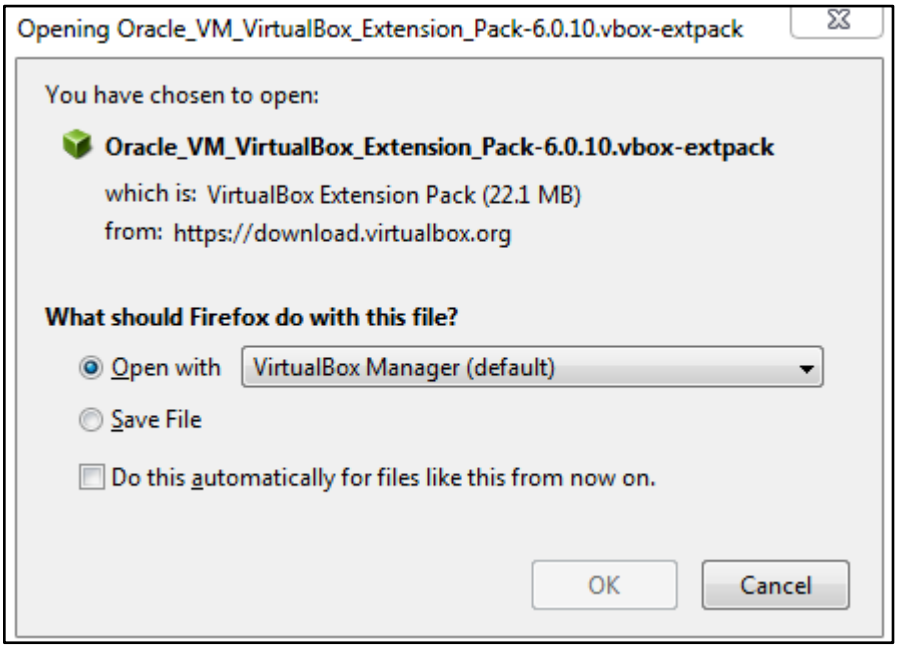


Fig 5.20: Download and opening Oracle VM Virtualbox Extension Pack

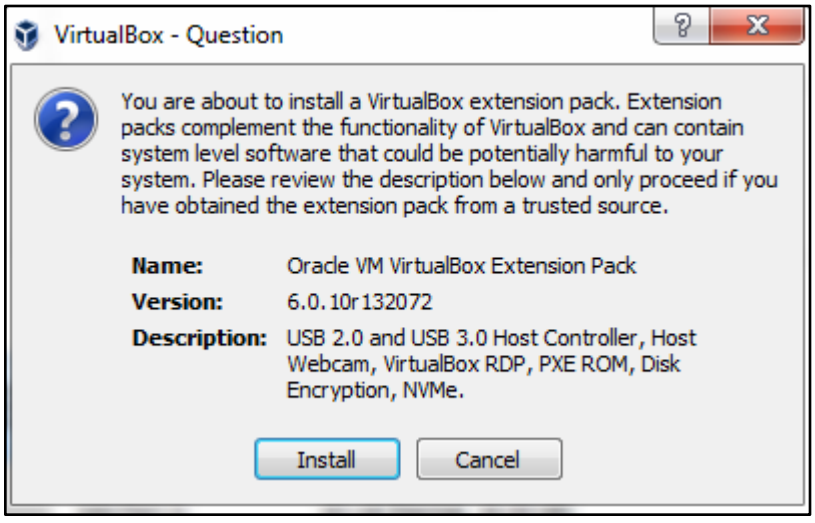


Fig 5.21: Installing Virtualbox Extension Pack

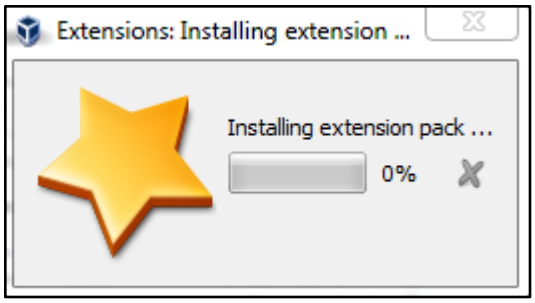


Fig 5.22: The installation progress

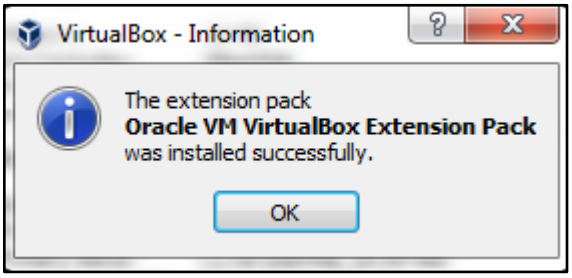


Fig 5.23: Installation completed

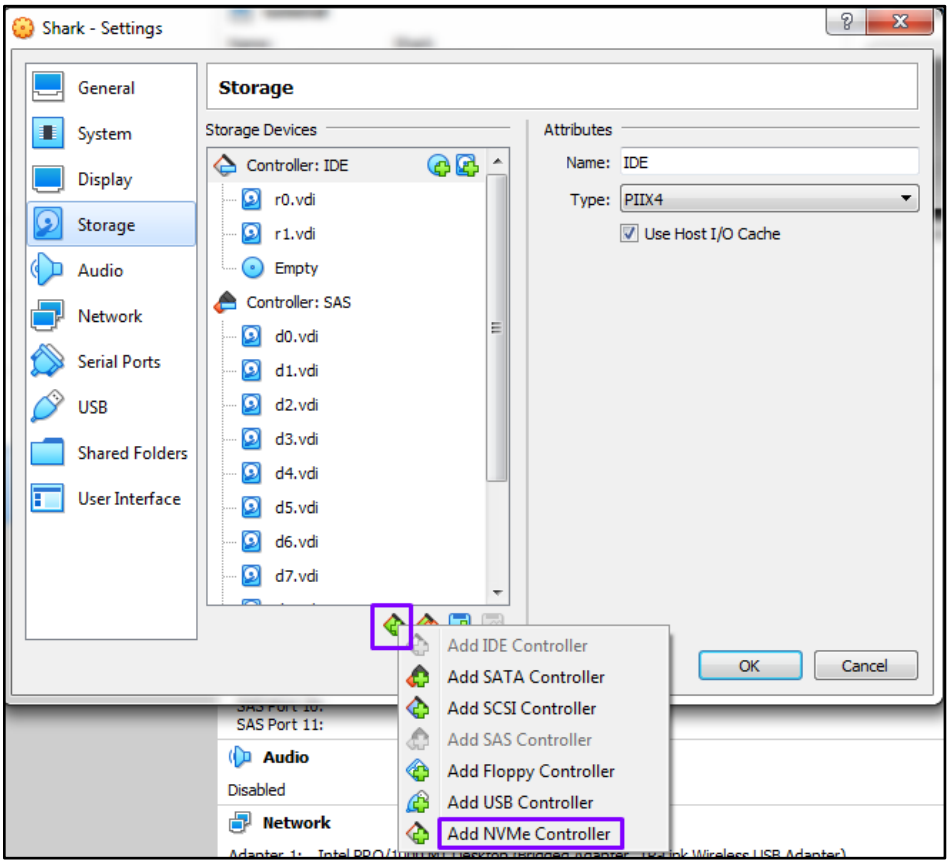


Fig 5.24: Adding a new NVMe Controller to Virtualbox

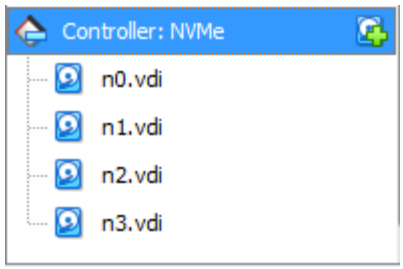


Fig 5.25: Adding 4 NVMe drives to Virtualbox

Log and Cache Disks

The ZFS file system has 2 structures - log and cache - to improve the I/O performance of the zpool. The use of the log and the cache is dependent on the type of applications and workloads and it is best to understand the characteristics and the I/O patterns of these applications and workloads before deciding if a ZFS log or a ZFS cache should be added.

In a very simplistic manner, ZFS cache is the write cache, in which it is the ZIL (ZFS Intent Log). ZFS log is the read cache or also known as L2ARC (Level 2 Adaptive Replacement Caching). It would be very wrong to assume that they should behave like a typical write cache or a read cache. It is important to read and fully understand the ZFS primer (URL: https://www.ixsystems.com/documentation/freenas/11.2-U5/zfsprimer.html#zfs-primer) and the use of both cache and log respectively. Applying them incorrectly can have negative performance effects on the applications and workloads.

For the sake of brevity, I am adding both the ZFS cache and ZFS log simultaneously using the NVMe drives in the previous section of this chapter. In real-life production, you usually do not add a read cache (ZFS log) or a write cache (ZFS cache) simultaneously to an existing zpool.

It is important to segment the zpool(s) according to applications and workloads. Therefore, it is usually good practice to add a ZFS log to a zpool which handles read I/O intensive workloads and separately, for a zpool which handles write I/O intensive (synchronous writes) workloads. Refer to the ZFS primer for guidance.

Click ‘Add Cache’ and ‘Add Log’.

Fig 5.26: Adding ZFS cache and ZFS log

The “Cache vdev” and “Log vdev” tables appears. Select the NVMe drives to the respective vdevs. In the example, Fig 5.27 below, ‘nvd0’ and ‘nvd2’ are added to the “Cache vdev” and ‘nvd1’ and ‘nvd3’ are added to the “Log vdev”.

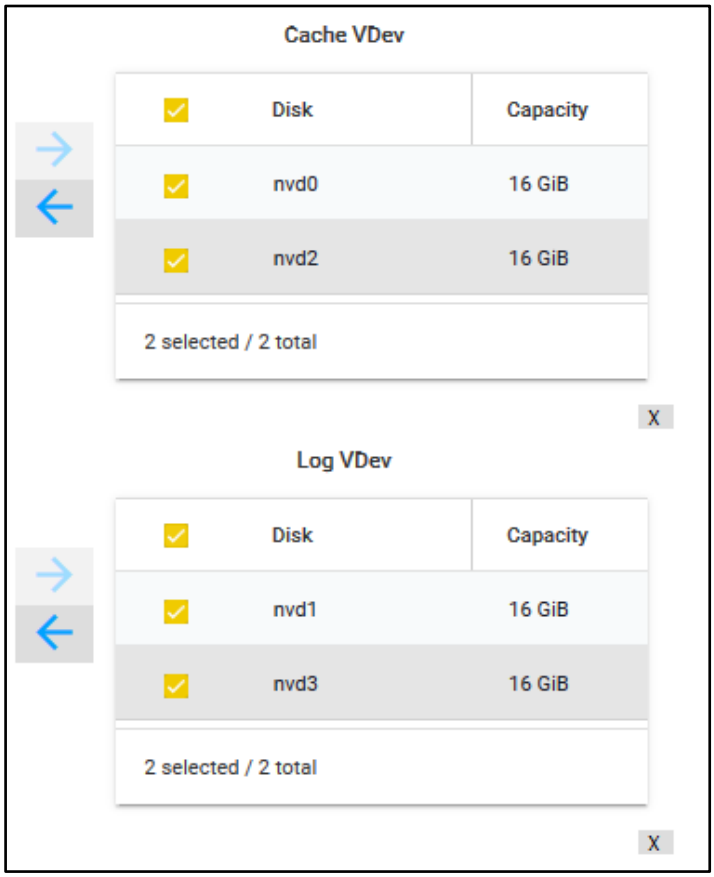


Fig 5.27: Selecting NVMe drives for Cache vdev and Log vdev respectively

Select ‘Extend’ at the bottom of the page to add the Cache and the Log.

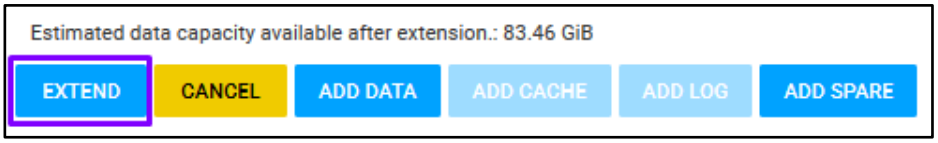


Fig 5.28: Extend the Cache and the Log

A warning pop-up appears. ‘Confirm’ to extend the pool.

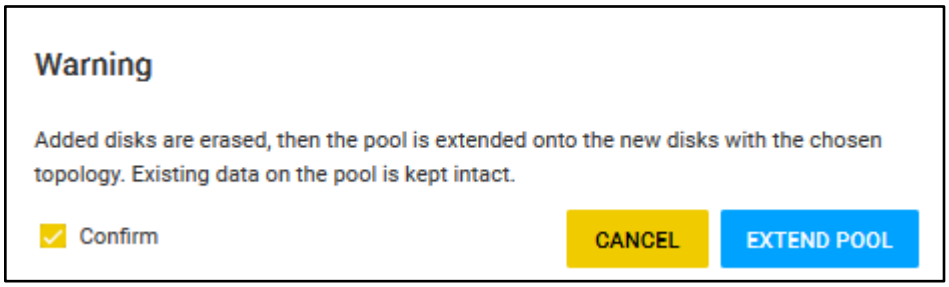


Fig 5.29: Extending the pool.

Increasing the zpool capacity

Besides increasing performance using ZFS cache and ZFS log, the zpool capacity can be increased in size to serve the applications and workloads.

The existing vdev(s) in the zpool does not allow the addition of disk drives. New vdev(s) has to be added to the existing zpool.

Select the drives to be added to create a new Data vdev.

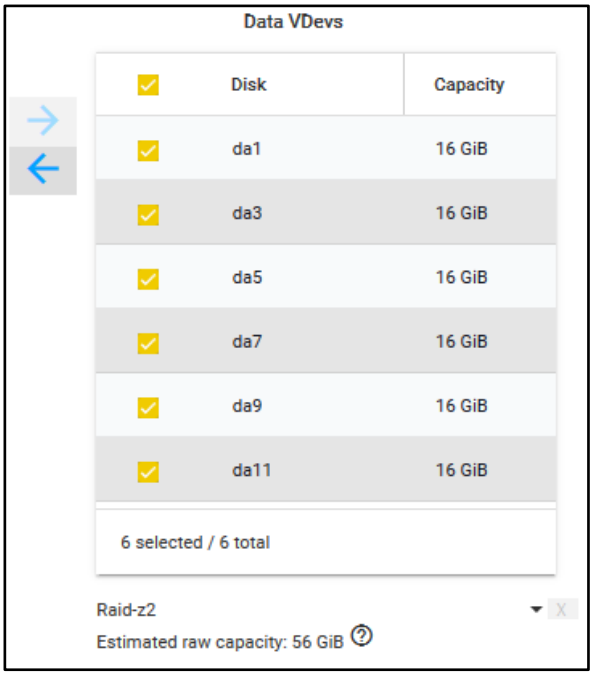


Fig 5.30: Selecting and adding new drives to Data vdev

Click ‘Extend’ to add the new vdev to the existing zpool.

Fig 5.31: Extend to increase the size of the zpool

‘Confirm’ in the pop-up message to execute the extension.

Best practice for vdev health

The health of the vdev is critical for the good operation of the zpool. A vdev failure would result in an inoperable zpool, and consequently lead to data inaccessibility and data loss.

Adding a Data vdev with the different number of disks to the existing vdev or setting a different RAID level to the existing vdev could create bad effects to the zpool.

FreeNAS would provide a warning to alert the system administrator of these unfavourable practices as shown in Fig 5.32 below.

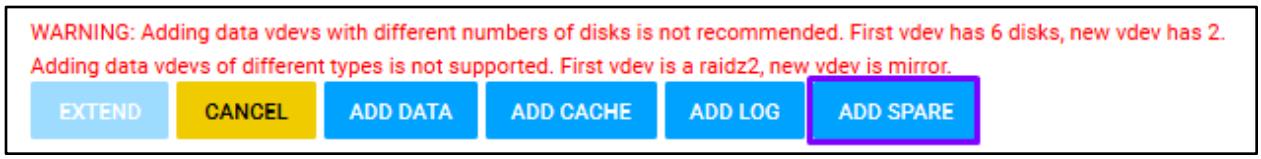


Fig 5.32: A warning alerting the inconsistent configuration of adding a new vdev.

Take note of the Warning message. The health of the vdev in a zpool is important. The best practice is to make sure that the number of disks and the RAID level in the new vdev is the same, and the capacity should be consistent with the existing vdev. In real-life production, the discipline to keep a standardized and consistent vdev configuration is important to ensure good vdev health.

CHAPTER 6 (FILE SHARING IN WINDOWS FOR WORKGROUP)

File Sharing

NAS or Network Attached Storage is one of the key functions of FreeNAS. In the Windows environment, the file sharing protocol is SMB (Server Message Block). SMB is also known as CIFS (Common Internet File System), although the CIFS name was only used briefly in SMB version 1.x.

File sharing in FreeNAS version 11.2-U5 is based on Samba version 4.9.9. You can easily verify the version by running the 'smbstatus' command at the shell prompt, as shown in Fig 6.1 below.

Fig 6.1: Checking the Samba version on FreeNAS

There are several versions of the SMB protocol from version 1 to version 3.1.1. The minimum SMB version is version 2.0. SMB version 1.0 is deprecated and disabled by default by FreeNAS version 11.2-U5 for security reasons. Older Windows clients like Windows XP and Windows Server 2003 has to work with SMB version 1.0.

SMB version 3.x is supported by FreeNAS and works with default sessions. Windows 10 and Windows Server 2016 support SMB version 3.x.

Shared folders from FreeNAS will appear as Network Drives on the Windows Client.

SMB Service

If Windows file sharing is configured for the first time, the SMB service on FreeNAS must be started. Slide the button to enable the service.

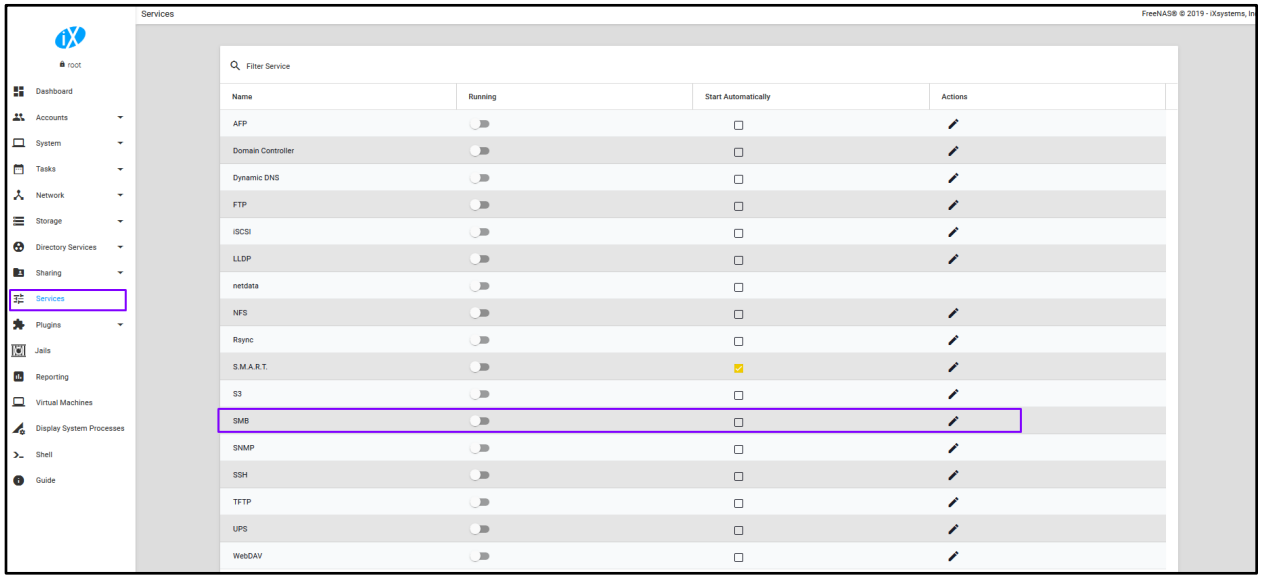


Fig 6.2: SMB service

Make sure that it starts automatically so that the SMB service will start up after a reboot.

Fig 6.3: Start SMB service

User and Group

In a Windows for Workgroup (WFW) setting, user authentication and permissions (generally known as Access Control Entries) are part of the Access Control List (ACL), and is executed at the local level. This means that the process of authentication and the assignment of access control are performed at the Windows client PC, and not at a centralized server like an Active Directory Domain Controller.

Therefore, in order to preserve a basic level of security and permissions for the FreeNAS shares for Windows, a user owner or a group owner is assigned to the FreeNAS dataset to be shared. This user owner is a separate and a different entity from the user logon of the Windows client.

User Private Group

The convention of User Private Group (UPG) was introduced by RedHat Linux and its related distributions where a private primary group is assigned to the newly created user. The only member of this primary group is the created user. The user is then assigned to other auxiliary secondary groups, and the respective auxiliary group is associated to various applications and services. There is no UPG convention practised in FreeBSD or FreeNAS.

The User Private Group convention does not improve security or changes the way Linux or any Unix variants behave but I find this practice useful to improve management, and use groups to control permissions and access.

During the creation of a new user in FreeNAS, the default is checked to create a 'New Primary Group' where the user is the only member of this Primary Group, as shown below in Fig 6.4.

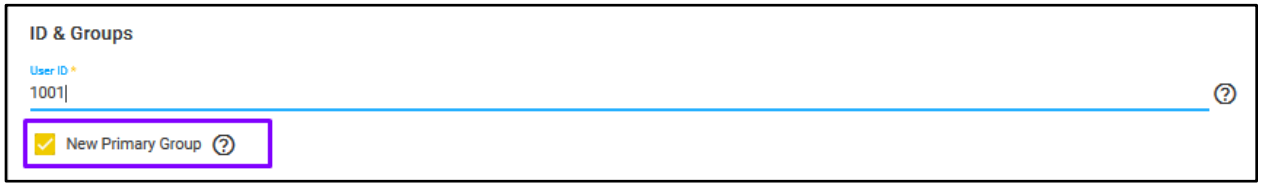


Fig 6.4: The checkbox for New Primary Group is checked by default

In our example, we created userA and we can observe that userA's Primary Group is also userA as shown in Fig 6.5.

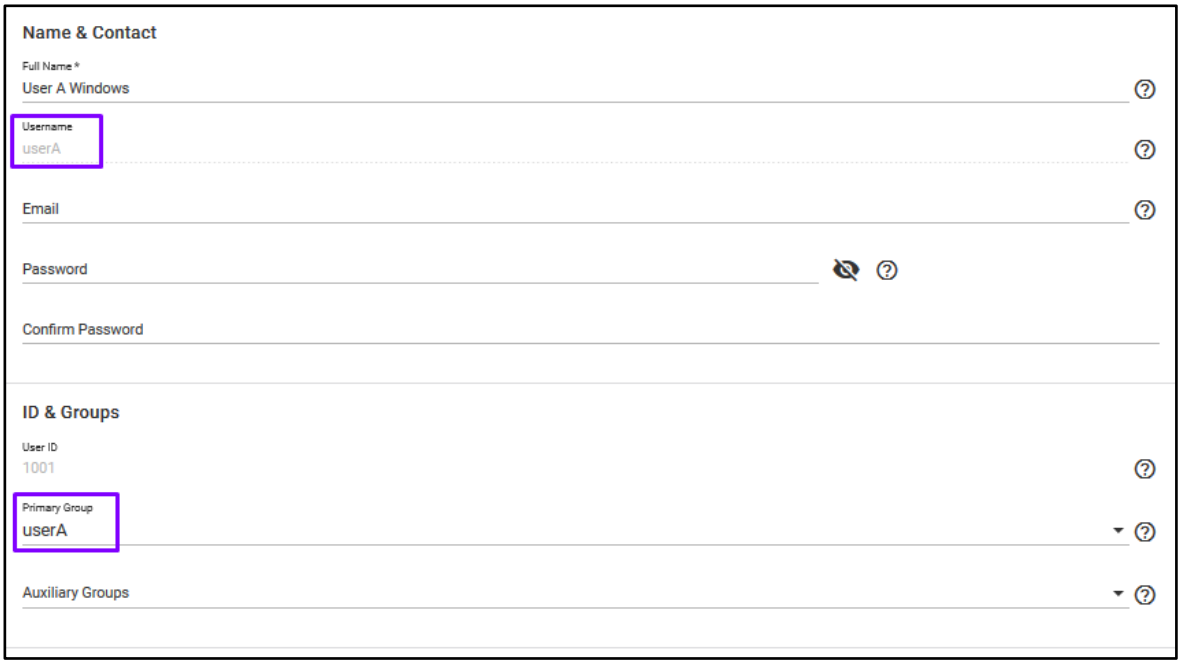


Fig 6.5: Primary Group of userA is userA

In this chapter, we will test using 2 types of dataset owners, which will be shared to the Windows client. One is with the owner who is a user, and one is with the owner who is a group. The table, Tab 6.1, below helps to identify the owners in the examples used in this chapter

username	Primary group	Auxiliary Group	Dataset/share owner	Client OS
userA	userA	none	win01	Windows
userB	userB	none	win02	Windows
userC	userC	none	mac01	MacOS
userD	userD	none	mac02	MacOS
userW	userW	groupW	win98	Windows
userX	userX	groupX	win99	Windows
userY	userY	groupY	mac98	MacOS
userZ	userZ	groupZ	mac99	MacOS

Tab 6.1: Users, Groups and datasets

It may seem odd that I have created more users to do one exercise but as you will observe later that having multiple use and group owners (in yellow) creates a problem in Windows in a workgroup environment.

To create users - userA, userB, userC, userD - the FreeNAS WebGUI presents the screenshot in Fig 6.6 and 6.7 below:

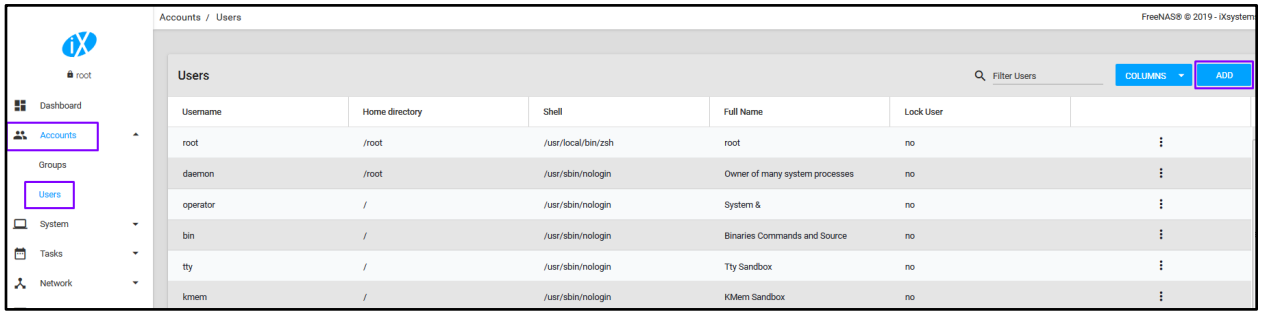


Fig 6.6: Add new user

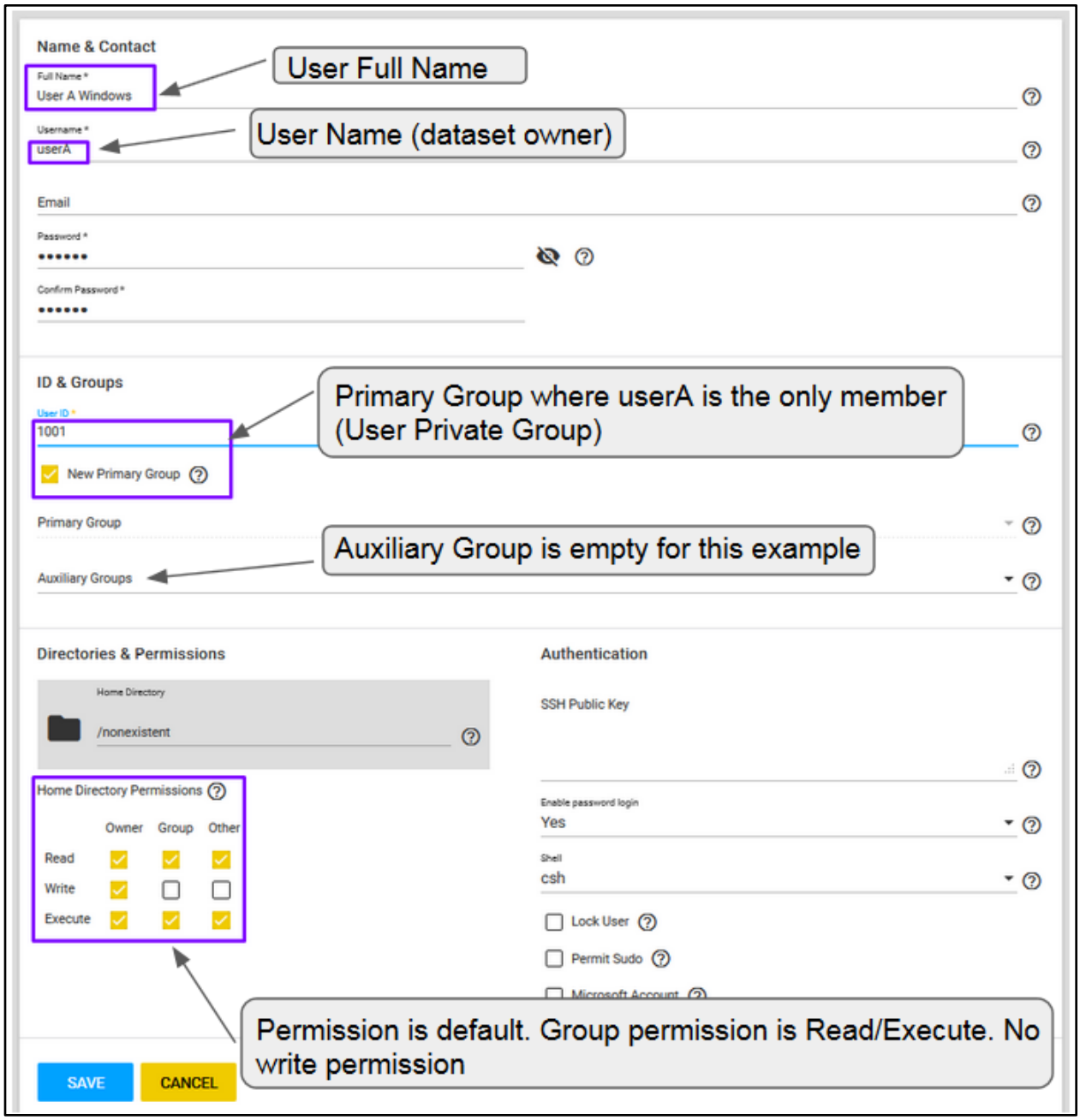


Fig 6.7: Adding new user details with permission setting

To create groups - groupW, groupX, groupY, groupZ - see screenshots below in Fig 6.8 and 6.9.

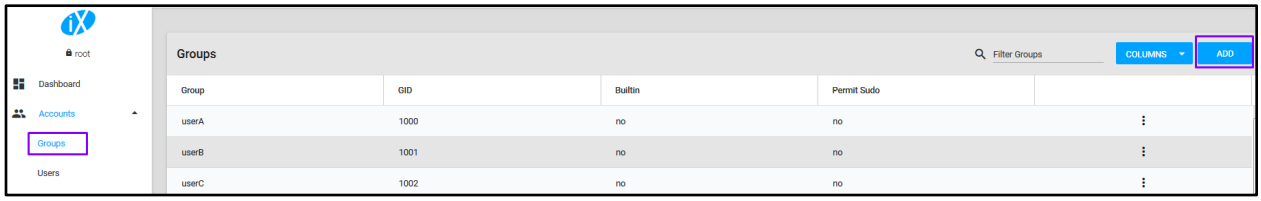


Fig 6.8: Add new group

Fig 6.9: New group details

All the created groups - groupW, groupX, groupY, groupZ - shown below in Fig 6.10.

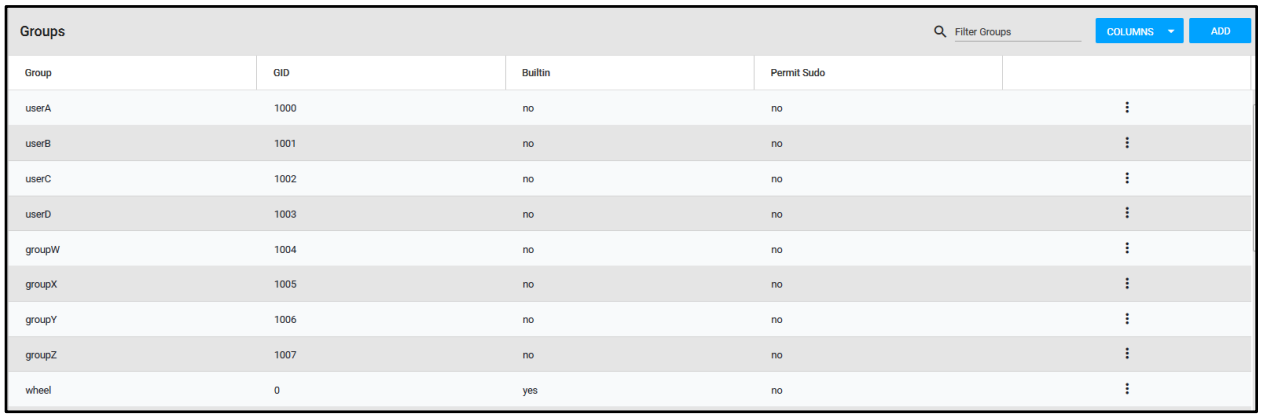


Fig 6.10: All the created groups

The users - userW, userX, userY, userZ - are assigned to groups - groupW, groupX, groupY, groupZ - respectively as their auxiliary group. They will retain their own respective Primary Group.

Create ‘userW’ with Primary Group ‘userW’ and with Auxiliary Group ‘groupW’ as in Fig 6.11

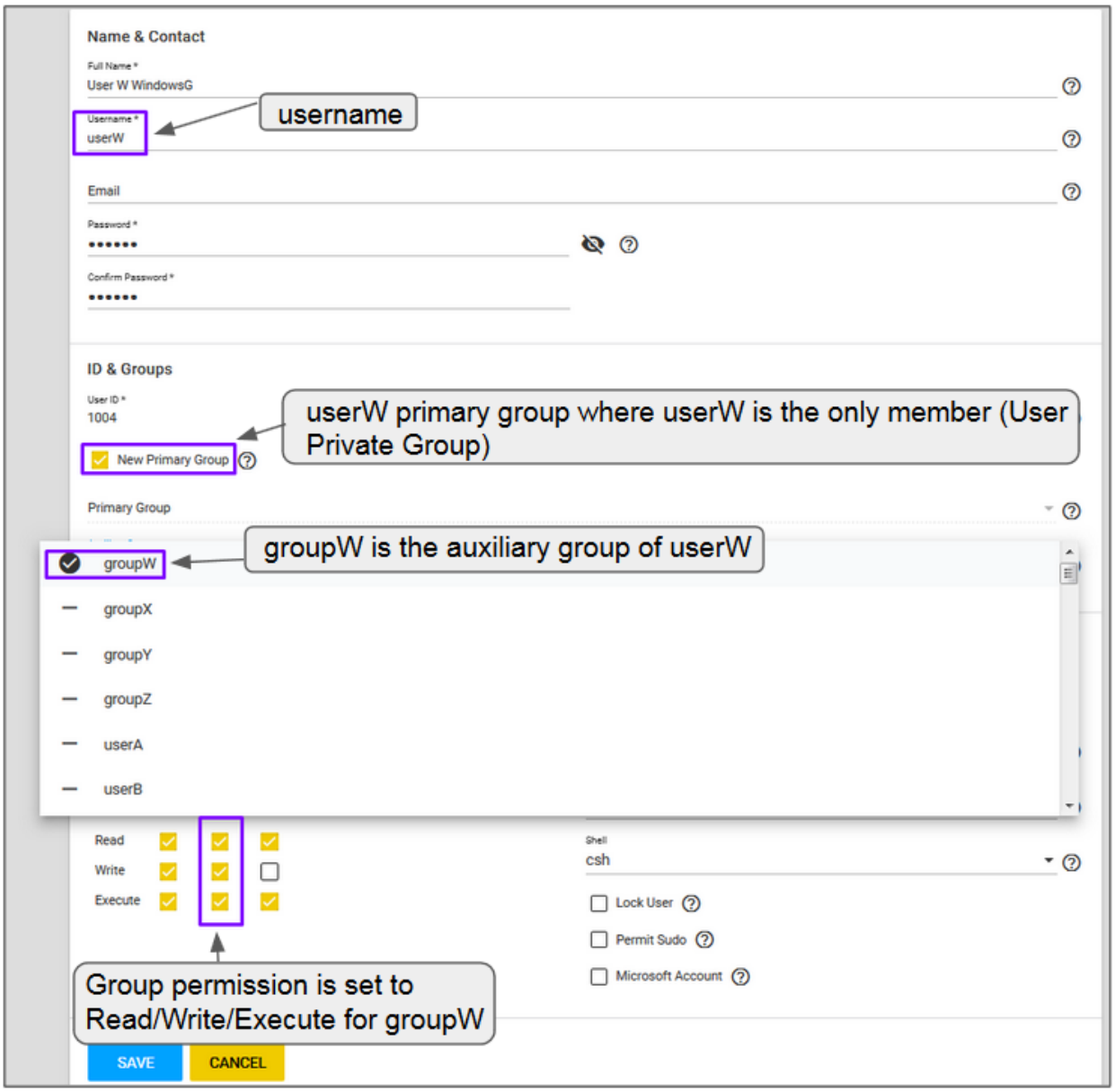


Fig 6.11: Creating ‘userW’ with Auxiliary Group ‘groupW’

After all the users - userA, userB, userC, userD, userW, userX, userY, userZ - have been created, see Fig 6.11 below.

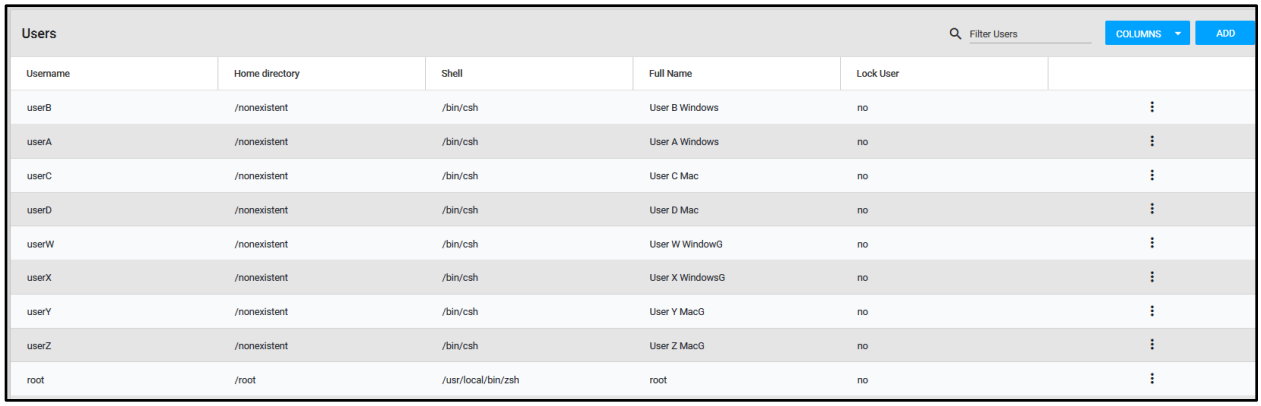


Fig 6.11: Displays all the created users

Creating a Dataset

A dataset is a data structure in the form of a file system directory “carved” from a zpool, an aggregated storage pool comprising of set of vdevs. The concept of a storage pool and vdevs are discussed in the previous Chapter 5.

Eventually, the dataset is shared as a Windows share to the SMB clients.

Following the plan in Tab 6.1, we create 4 user owned datasets - win01, win02, mac01, mac02 and 4 group owned datasets - win98, win99, mac98, mac99.

Under Storage > Pools > Add Dataset

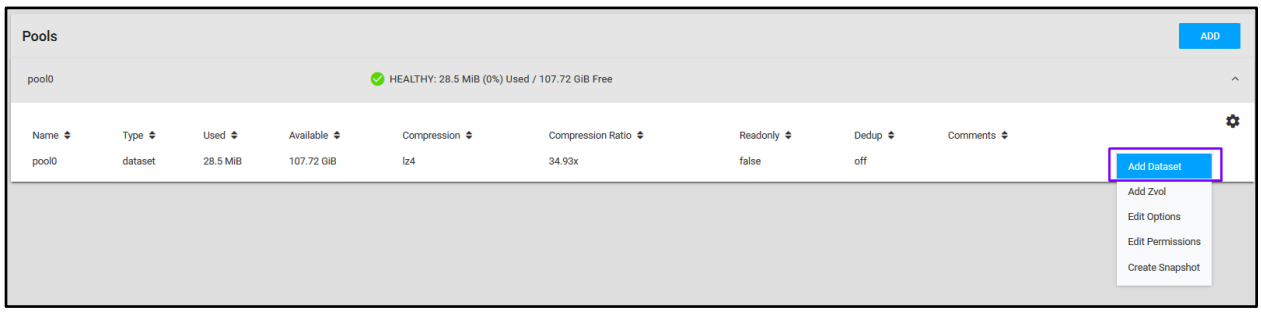


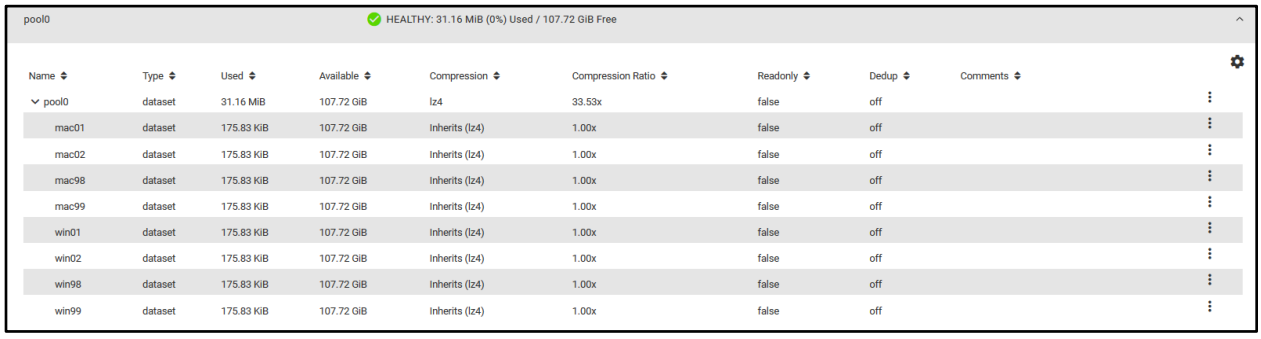
Fig 6.12: Add new dataset

Fill in the details of the dataset as shown in Fig 6.13 below.

Fig 6.13: Adding the details to the dataset

Provide a name to the dataset as well as the Share Type. In this example, the Share Type is Windows for all the created datasets since they are shared through the SMB protocol.

All the datasets have been created as shown below in Fig 6.14.



Name	Type	Used	Available	Compression	Compression Ratio	Readonly	Dedup	Comments
pool0	dataset	31.16 MiB	107.72 GiB	lz4	33.53x	false	off	
mac01	dataset	175.83 KiB	107.72 GiB	Inherits (lz4)	1.00x	false	off	
mac02	dataset	175.83 KiB	107.72 GiB	Inherits (lz4)	1.00x	false	off	
mac98	dataset	175.83 KiB	107.72 GiB	Inherits (lz4)	1.00x	false	off	
mac99	dataset	175.83 KiB	107.72 GiB	Inherits (lz4)	1.00x	false	off	
win01	dataset	175.83 KiB	107.72 GiB	Inherits (lz4)	1.00x	false	off	
win02	dataset	175.83 KiB	107.72 GiB	Inherits (lz4)	1.00x	false	off	
win98	dataset	175.83 KiB	107.72 GiB	Inherits (lz4)	1.00x	false	off	
win99	dataset	175.83 KiB	107.72 GiB	Inherits (lz4)	1.00x	false	off	

Fig 6.14: All the created datasets

To configure the Permissions of each dataset, the action is shown in Fig 6.15

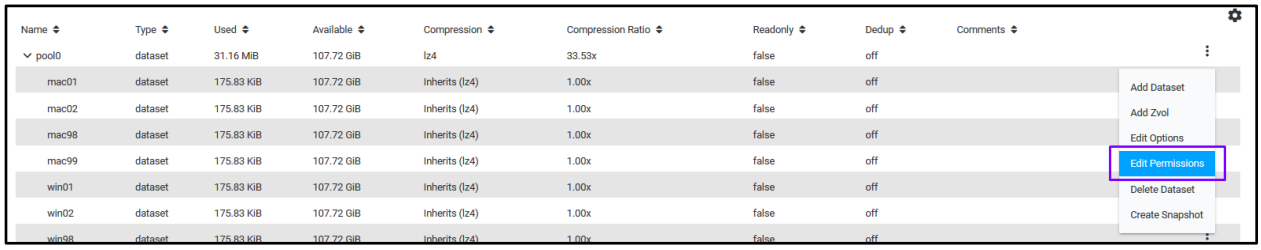


Fig 6.15: Edit permission of the dataset

For the user owned datasets, in Fig 6.16

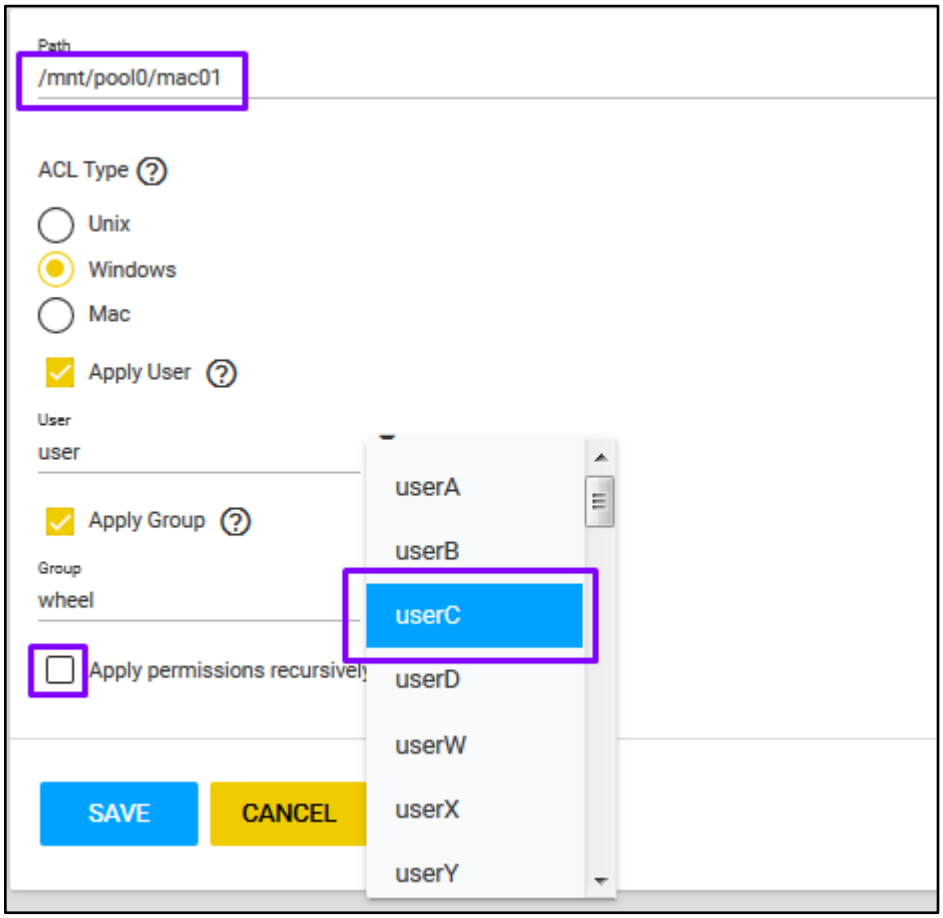


Fig 6.16: Setting the ownership of the dataset

Make user the right owner corresponds to the right path of the dataset. In the example, /mnt/pool0/mac01 is owned by userC.

Note that checking the ‘Apply permission recursively’ check box will have a pop-up message as in Fig 6.17. Applying recursive permission can have effects if there are

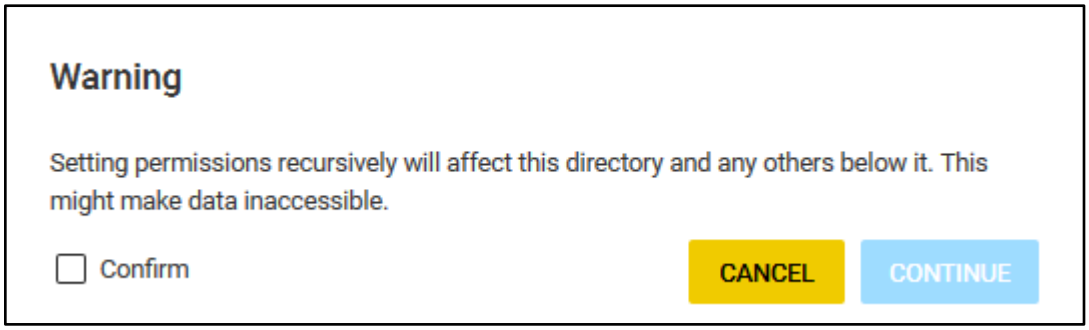


Fig 6.17: Setting permission recursively pop-up message

If set, the permission settings of the parent dataset will be inherited by the child datasets, i.e. sub directories of the main dataset path.

Also note that you can set a quota - a fixed size - for the dataset. Click on the ‘Advanced Mode’ button as in Fig 6.18 below.

Fig 6.18: Dataset Advanced Mode

Fig 6.19: Setting a fixed size (quota) for the dataset

If the quota is not fixed, then the dataset will assume the available storage capacity of the entire storage pool and competes with the other datasets for the capacity. It is good practice to set a quota for the datasets to be shared, because there will be a capacity limit allocated to each Windows network drive.

For the group owned datasets, the dataset win99 is owned by groupX as in Fig 6.20

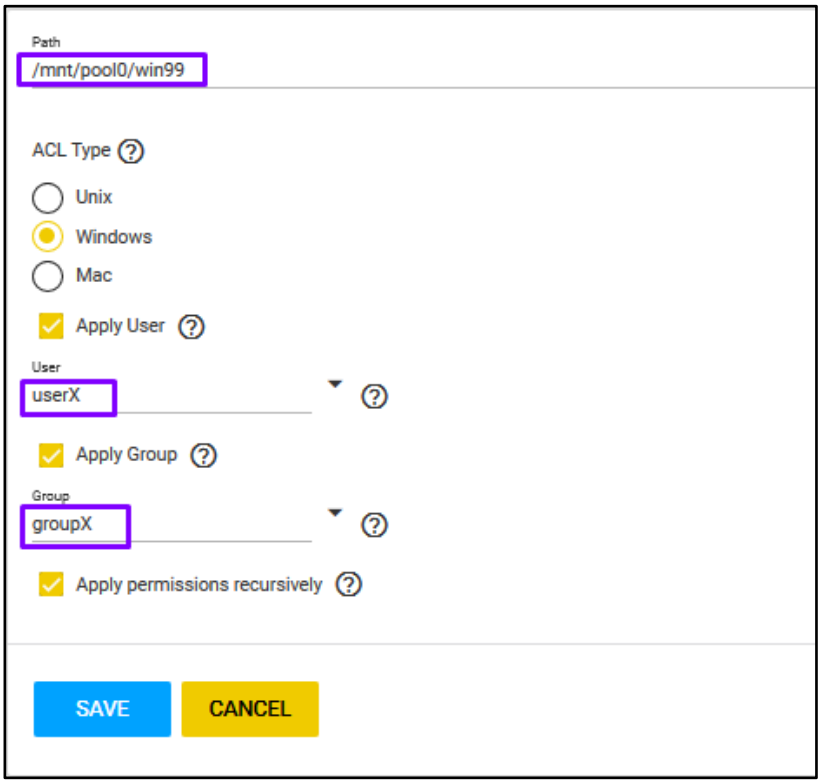


Fig 6.20: Setting group owner for dataset win99

This group ownership practice allows multiple users to be grouped into a specific group which is assigned to the dataset and share. The advantage of this grouping is to simplify permission management without creating one user per dataset and share.

Sharing for Windows Client

Once the dataset has been assigned a permission list, either via user ownership or group ownership, the dataset is shared via the SMB protocol, with a Share Name.

Fig 6.21: Adding Windows Share

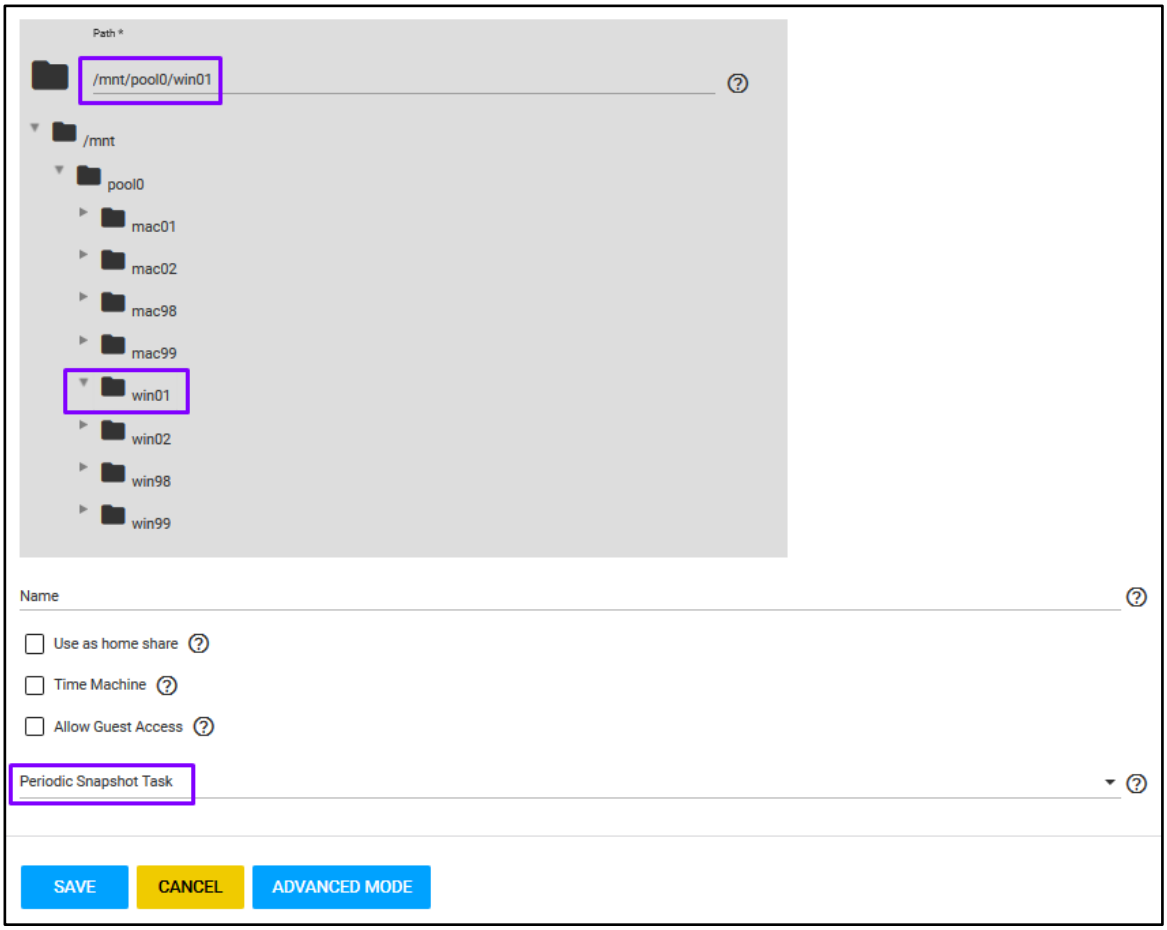


Fig 6.22: Setting the dataset path and the Share Name

Select the mounted dataset path and provide a Share Name for the FreeNAS Windows Share. If the dataset was configured with a Periodic Snapshot, choose the Periodic Snapshot policy. Periodic Snapshot is discussed in another chapter of this book.

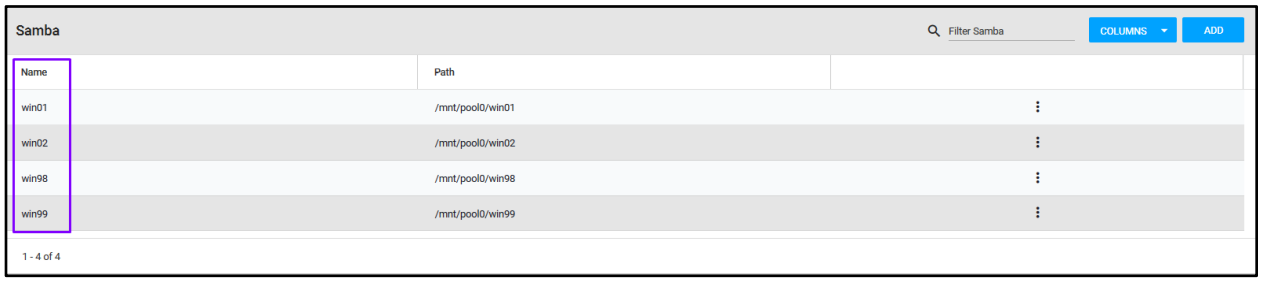


Fig 6.23: Share Names created with respective paths to the datasets

Mapping shares at the Windows Client

Always make sure that the FreeNAS system is reachable via the network before mapping the share to the Windows network drive. A simple ping test helps.

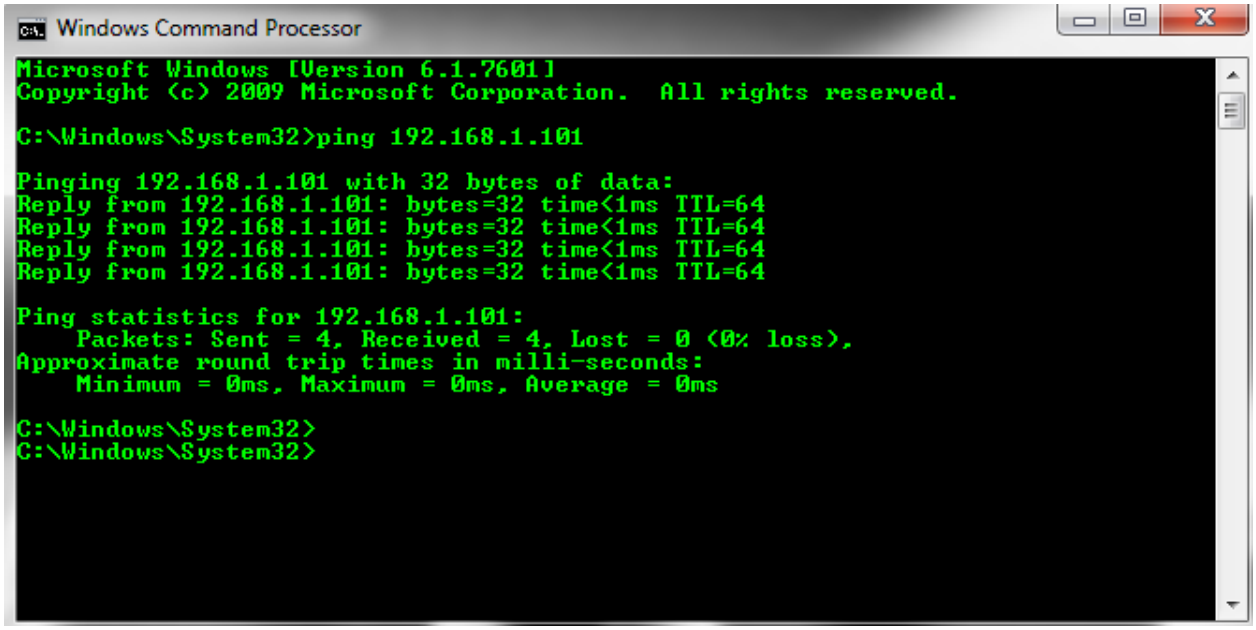


Fig 6.24: Testing the FreeNAS system is reachable via the network

At the Windows File Explorer, select ‘Computer’ and click ‘Map Network Drive’ at the top, as shown below in Fig 6.25.

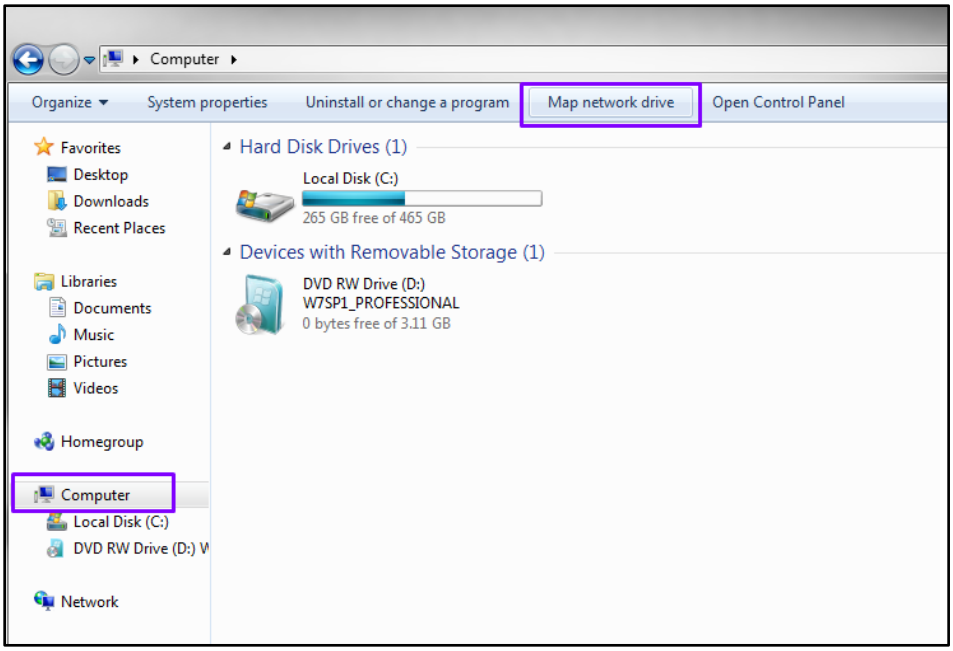


Fig 6.25: Map network drive

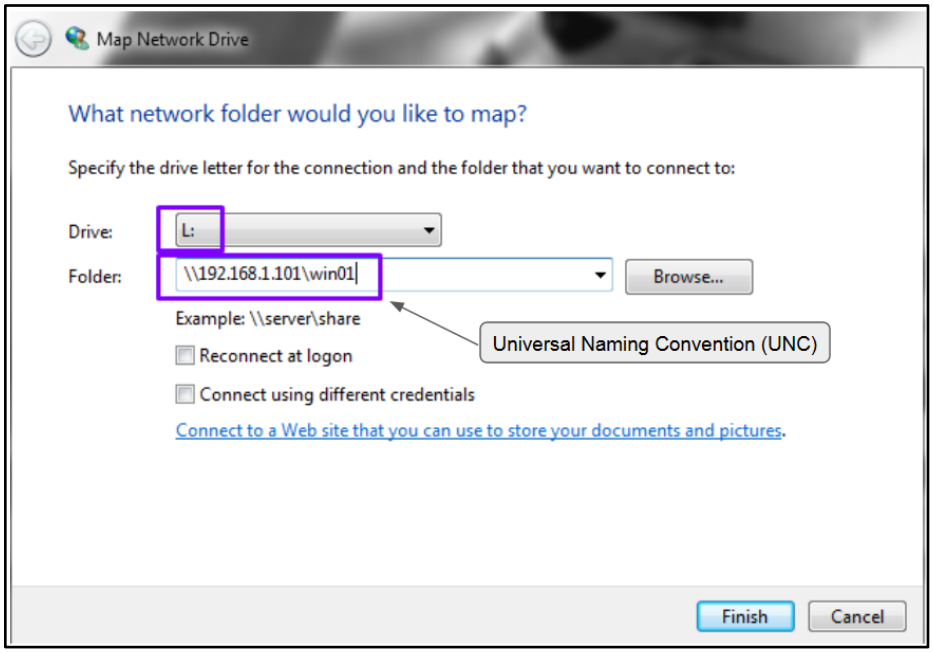


Fig 6.26: Assigning L: as network drive with UNC

A network drive, L: is assigned to the FreeNAS Share through the UNC (Universal Naming Convention) address. The syntax is “\\servername\sharename” as shown in Fig 6.26.

A pop-up message appear for the Windows Client to input username and password to map the network drive.

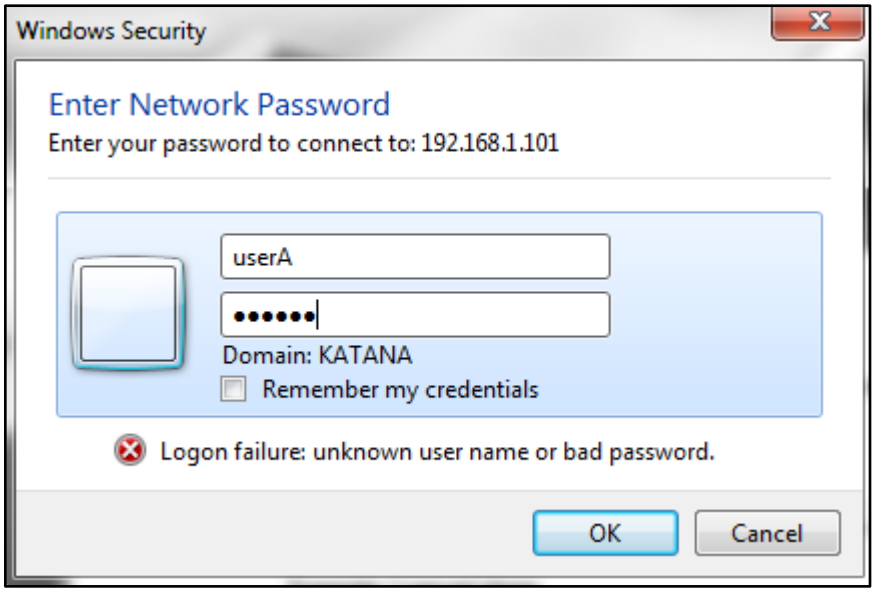


Fig 6.27: Username and password to map the FreeNAS share

In the above example, userA is the owner of FreeNAS share 'win01' and it is mapped to the L: network drive.

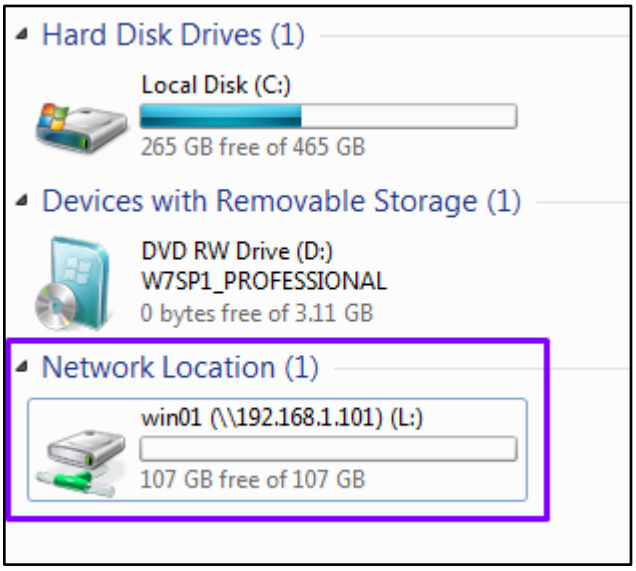


Fig 6.28: L: network drive mapped

The multiple user credentials in Window client problem

When you want to map another share from the same FreeNAS system, you are likely to encounter the problem of permission to write or mapping. An error such as

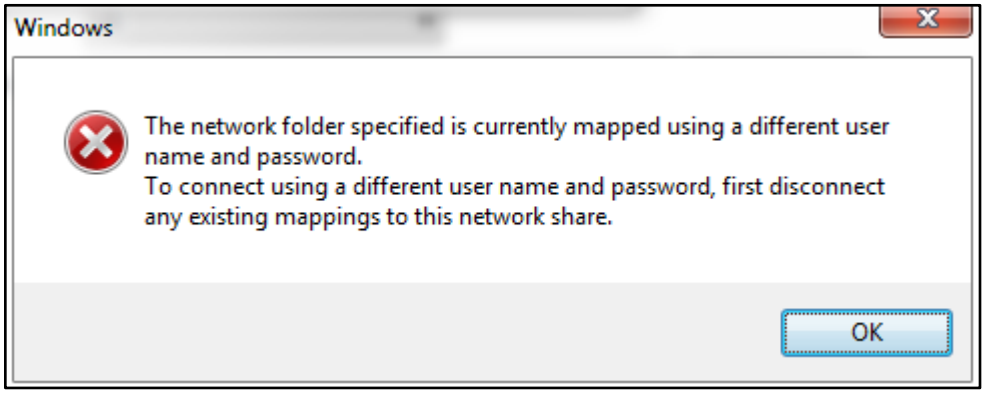


Fig 6.29: Pop-up error when mapping another share with a different user credential

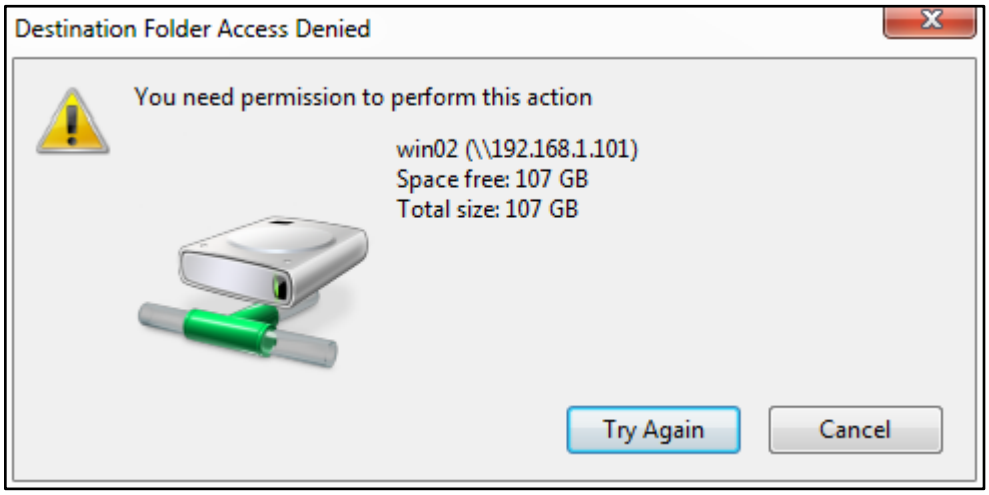


Fig 6.30: A second share is mapped but does not have permission to create a new folder or file

This is a normal Windows client behaviour in a Workgroup environment. In a non-Active Directory setting, the Windows Client rejects a second user credentials. It only allows one user credential per SMB map for security reasons.

Therefore the second network drive mapping works using the first user credentials (because it is cached) of the Windows client but it does not give the write permission to create a folder or file in the second mapped drive.

To overcome this problem, we have to “trick” Windows that there is a “different” FreeNAS system in the network. We create a hostname alias in the C:\windows\system32\drivers\etc\hosts file.

Open this file with Notepad and add in a hostname alias as such

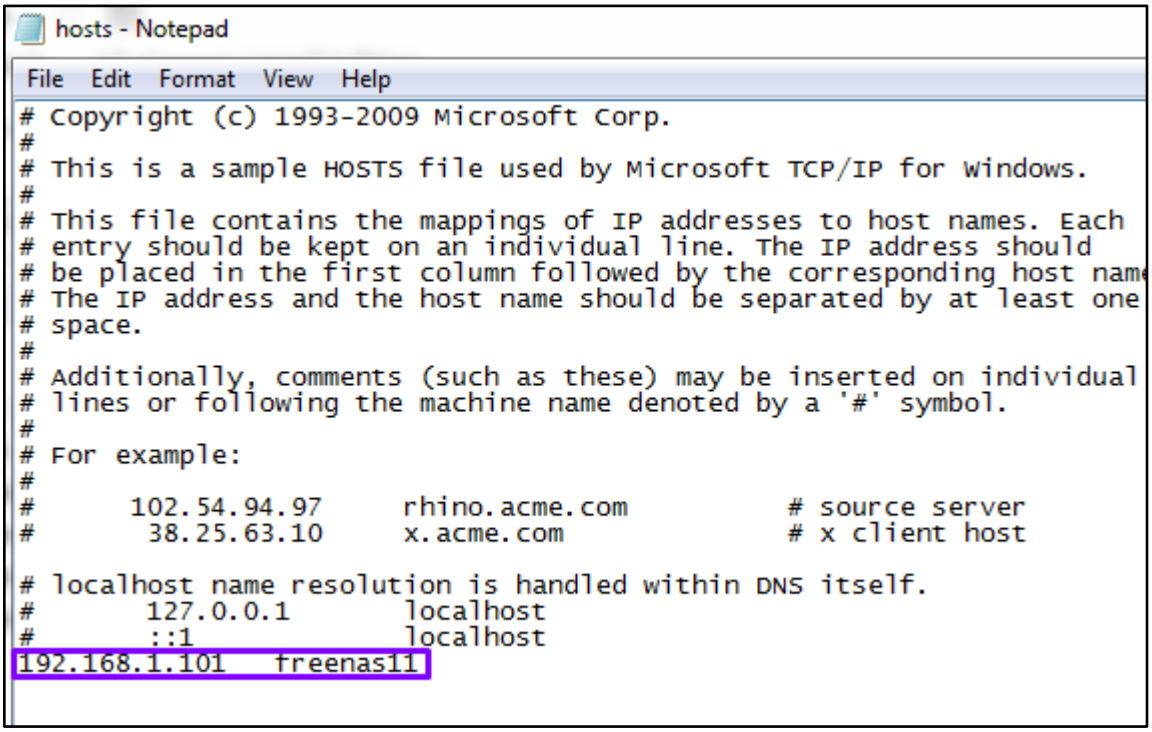


Fig 6.31: Editing the HOSTS file

In the Fig 6.31, the hostname ‘freenas11’ is added.

IMPORTANT: When saving the HOSTS file in Notepad, ensure that it is NOT saved as a .TXT file. Save with the “All Files” type.

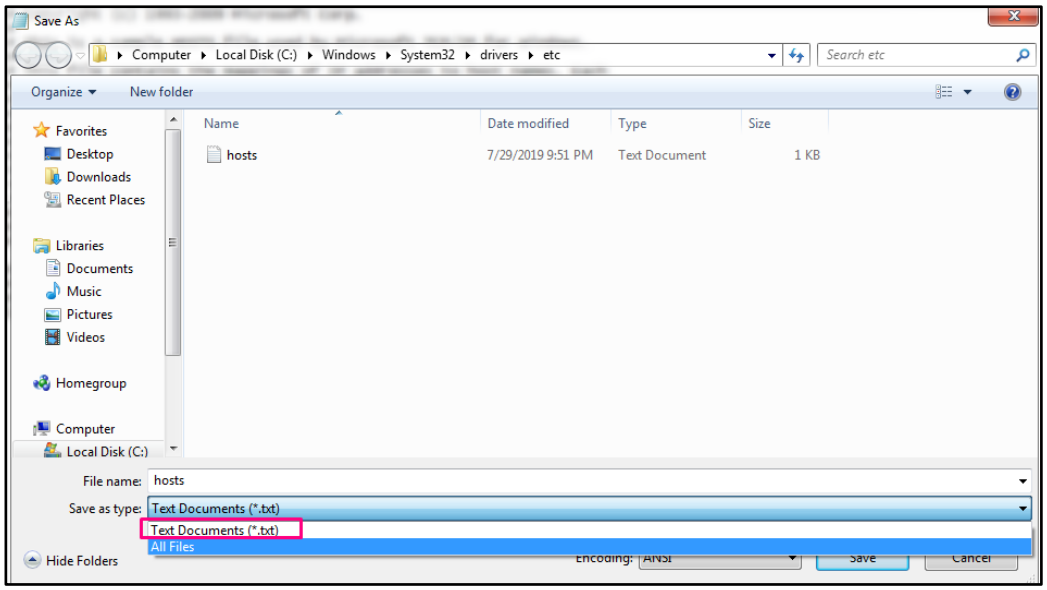


Fig 6.32: Do not save as .TXT file

Mapping the second share with a different user credentials (after tricking the Windows Client) is shown below:

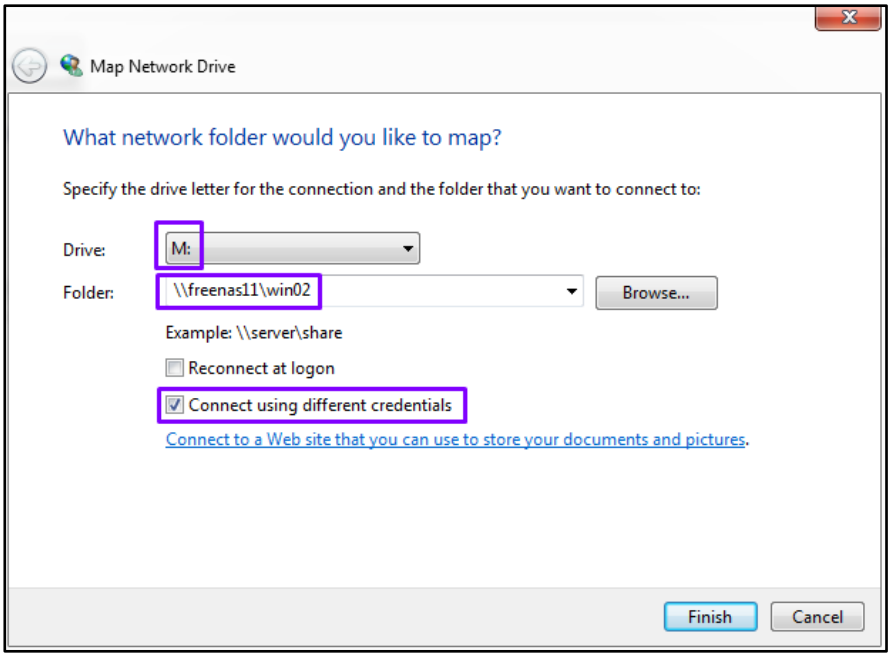


Fig 6.33: Mapping with a new set of session credentials

In the Fig 6.33 above, the FreeNAS second share from the same system is linked to the UNC of \\freenas11\win02 instead of the IP address of 192.168.1.101. The Windows Client will assume that the second share is from a different FreeNAS system.

Check the box “Connect using different credentials”. Logon with userB and you get 2 mapped drives, L: and M: respectively, with write permissions.

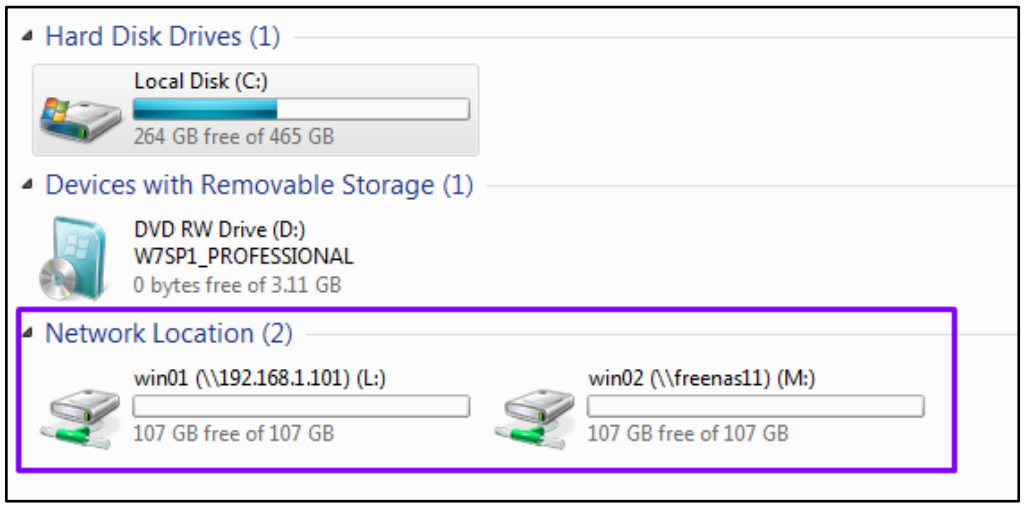


Fig 6.34: Multiple network drives on the Windows Client

Mounting Remote Drives in MacOS (SMB Protocol)

MacOS mounts the FreeNAS SMB shares via the Finder. At the top menu, select “Connect to Server ...”.

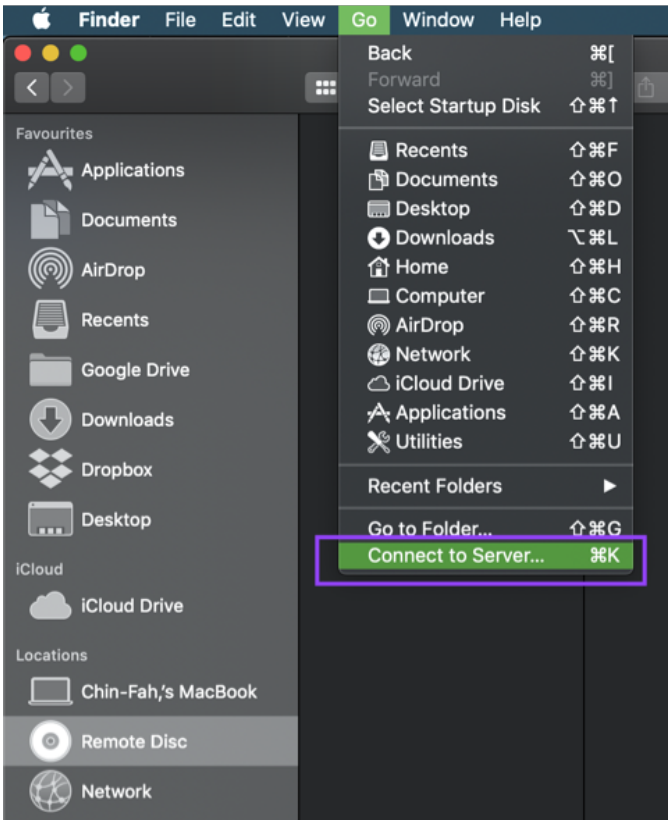


Fig 6.35: Connect to FreeNAS system

Type ‘smb://192.168.1.101/mac01’ to mount the SMB share from FreeNAS as in Fig 6.36.

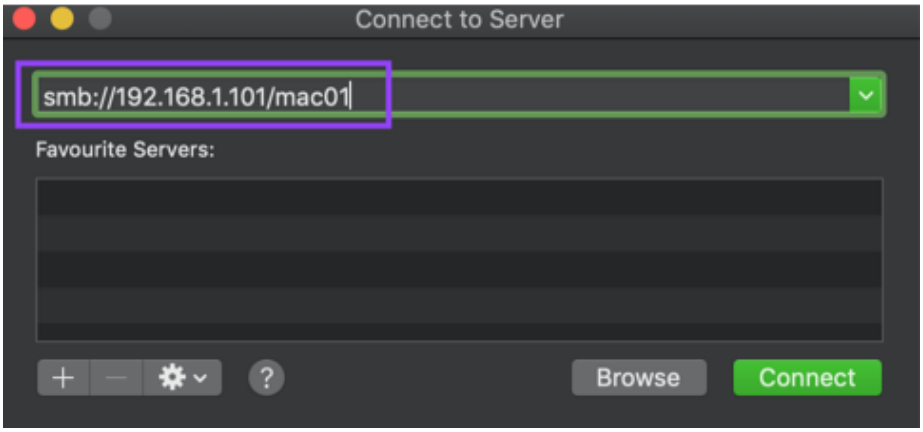


Fig 6.36: Mounting the FreeNAS SMB share

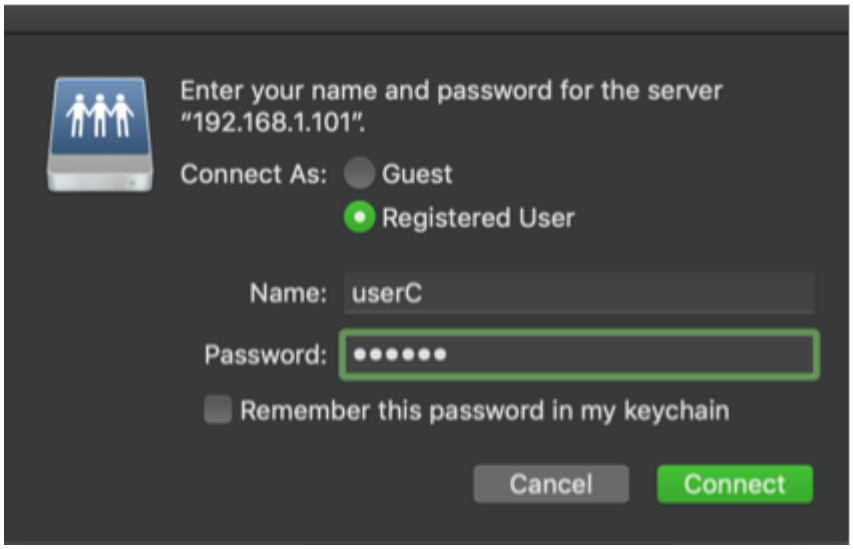


Fig 6.37: Login with the user credentials

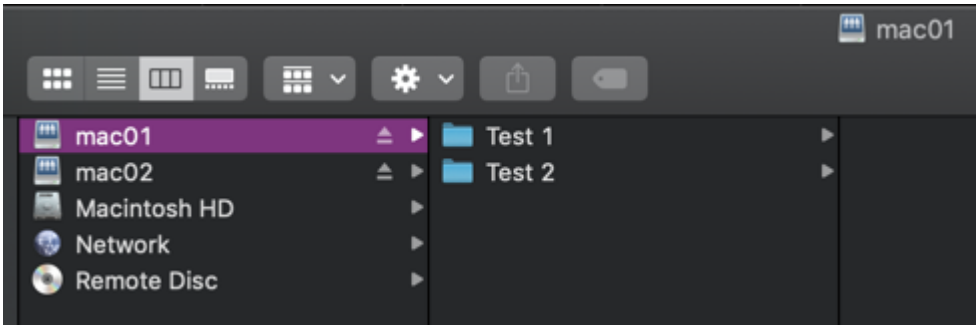


Fig 6.38: The mounted “drives” from FreeNAS SMB shares