

HO1057 – What's New in SLE 15 SP1

Important Information:

RMT/NTP Server IP Address: 192.168.150.215

openSUSE Leap 15.1 VM IP Address: 192.168.150.226

SLES 15 VM IP Address: 192.168.150.202

Username: geeko

Password: linux

Username: root

Password: linux

Lab 1 – Build and Migrate

1.0 Recreate RMT Certificate Just In Case

1. Ensure the RMT VM is up and running correctly before proceeding.
2. From the lab machine, connect to the RMT VM:

```
$ ssh root@<rmt>
```

3. Because of the lab environment, the RMT certificate needs to be regenerated. First stop the running services:

```
# systemctl stop nginx rmt-server
```

4. Remove previously generated CA and HTTPS certificates:

```
# rm /etc/rmt/ssl/rmt-.*
```

5. Re-run the RMT server configuration, skipping most steps;

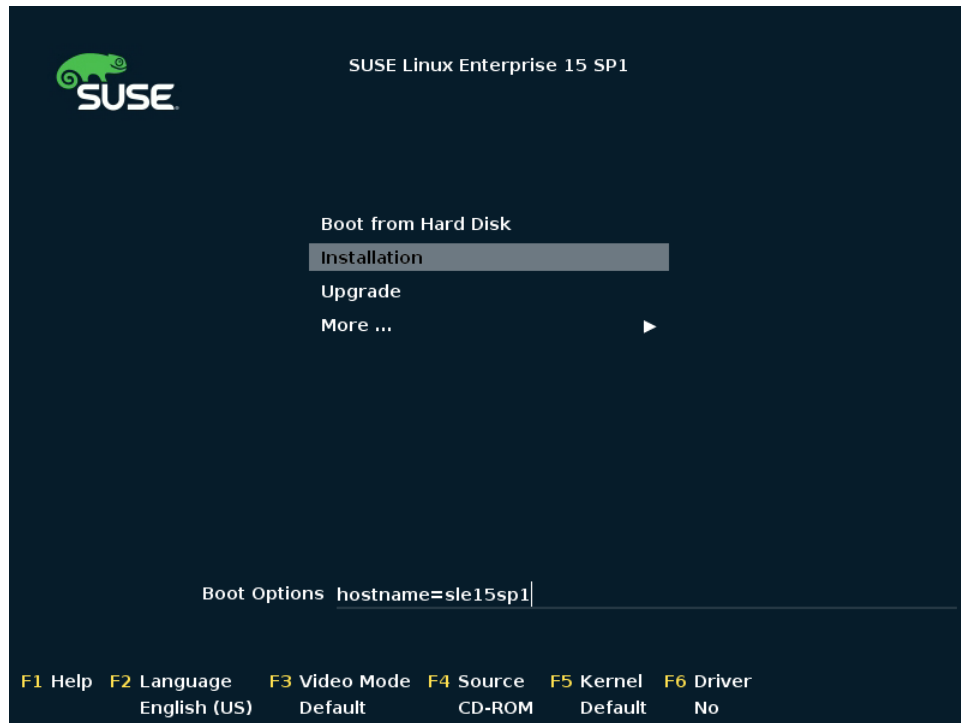
```
# yast rmt
```

6. Enter nothing in the organization credentials and select “Next”. (We don’t want to actually mirror from SUSE.com for this lab.)
7. At the prompt “Configuration written successfully.” select “OK”.
8. Enter nothing for Database Password and select “Next”.
9. At the prompt “Configuration written successfully.” select “OK”.
10. In SSL Certificate Generation select “Next”.
11. Enter “password” for both Password and Confirm Password and select “OK”.
12. On the firewall configuration select “Next”.
13. Select “Next”.
14. Select “Finish”.

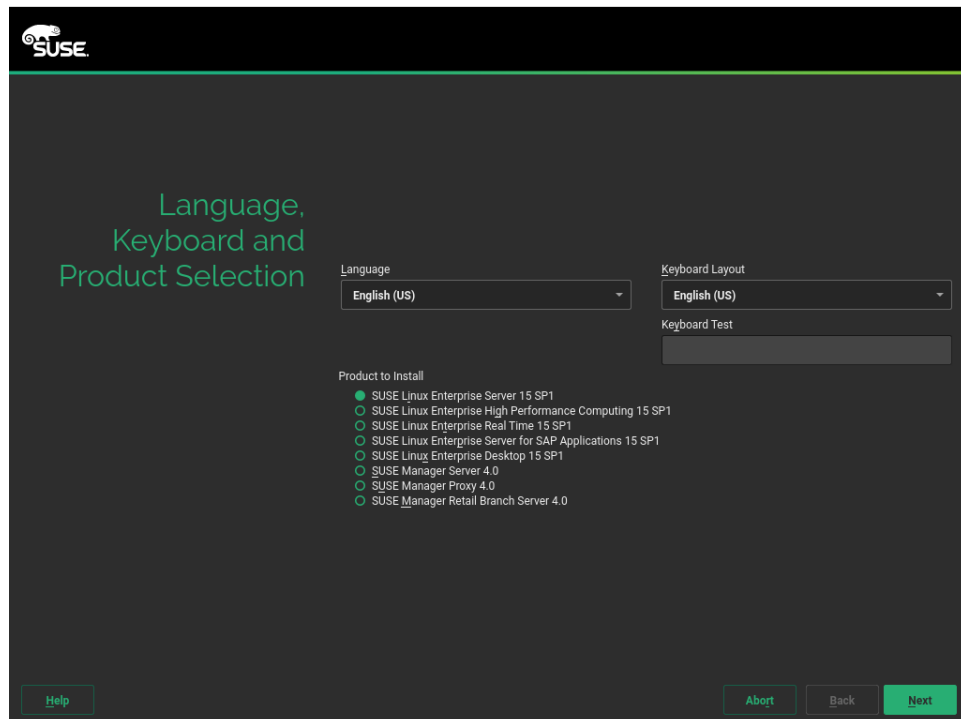
1.1 Build a SLE VM from Installer

15. In virt-manager, boot the sles15sp1 VM from the DVD.

16. Select Installation, enter “hostname=sles15sp1” in Boot Options, and hit Return.



17. Select “SUSE Linux Enterprise Server 15 SP1” and click Next in Language, Keyboard and Product Selection.



18. Check “I Agree to the License Terms” and click Next in License Agreement.

The screenshot shows the SUSE Linux Enterprise Server 15 SP1 License Agreement. On the left, the title "SUSE Linux Enterprise Server 15 SP1 License Agreement" is displayed in green. On the right, the license text is shown in a scrollable area. At the top right, there is a "Language" dropdown menu set to "English (US)". At the bottom, there is a checkbox labeled "I Agree to the License Terms" which is checked. Below the checkbox are three buttons: "Help", "Abort", and "Next".

Language
English (US)

SUSE Linux Enterprise Server 15 SP1 License Agreement

SUSE Linux Enterprise End User License Agreement for the following SUSE products:

- SUSE Linux Enterprise Server 15 SP1
- SUSE Linux Enterprise Server for SAP Applications 15 SP1
- SUSE Linux High Availability Extension 15 SP1
- SUSE Linux Enterprise HPC 15 SP1
- SUSE Linux Enterprise Desktop 15 SP1
- SUSE Linux Enterprise Workstation Extension 15 SP1
- SUSE Linux Enterprise Live Patching 15 SP1
- SUSE Manager Server 4
- SUSE Manager Proxy 4
- SUSE Manager Retail Branch Server 4

PLEASE READ THIS AGREEMENT CAREFULLY. BY PURCHASING, INSTALLING, DOWNLOADING OR OTHERWISE USING THE SOFTWARE (INCLUDING ITS COMPONENTS), YOU AGREE TO THE TERMS OF THIS AGREEMENT. IF YOU DO NOT AGREE WITH THESE TERMS, YOU ARE NOT PERMITTED TO DOWNLOAD, INSTALL OR USE THE SOFTWARE AND YOU SHOULD NOTIFY THE PARTY FROM WHICH YOU PURCHASED THE SOFTWARE TO OBTAIN A REFUND. AN INDIVIDUAL ACTING ON BEHALF OF AN ENTITY REPRESENTS THAT HE OR SHE HAS THE AUTHORITY TO ENTER INTO THIS AGREEMENT ON BEHALF OF THAT ENTITY.

This End User License Agreement ("Agreement") is a legal agreement between You (an entity or a person) and SUSE LLC ("Licensor"). The software products identified in the title of this Agreement for which You have acquired licenses, any media or reproductions (physical or virtual) and accompanying documentation (collectively the "Software") is protected by the copyright laws and treaties of the United States ("U.S.") and other countries and is subject to the terms of this Agreement. If the laws of Your principal place of business require contracts to be in the local language to be enforceable, such local language version may be obtained from Licensor upon written request and shall be deemed to govern Your purchase of licenses to the Software. Any add-on, extension, update, mobile application, module...

☒ I Agree to the License Terms.

Help Abort Back Next

19. Check “Register System via local SMT Server”, enter <https://<rmt>> in Local Registration Server URL, and click Next on Registration.

The screenshot shows the SUSE Linux Enterprise Server 15 SP1 Registration screen. At the top right, there is a "Network Configuration..." button. The main heading is "Registration". Below it, the text "SUSE Linux Enterprise Server 15 SP1" is displayed. A message says "Please select your preferred method of registration." There are three radio button options: "Register System via scc.suse.com", "Register System via local SMT Server" (which is selected), and "Skip Registration". Under the "Register System via local SMT Server" option, there is a field for "Local Registration Server URL" with the value "https://192.168.150.214". At the bottom, there are three buttons: "Help", "Abort", and "Next".

Network Configuration...

Registration

SUSE Linux Enterprise Server 15 SP1

Please select your preferred method of registration.

☐ Register System via scc.suse.com

E-mail Address

Registration Code

☒ Register System via local SMT Server

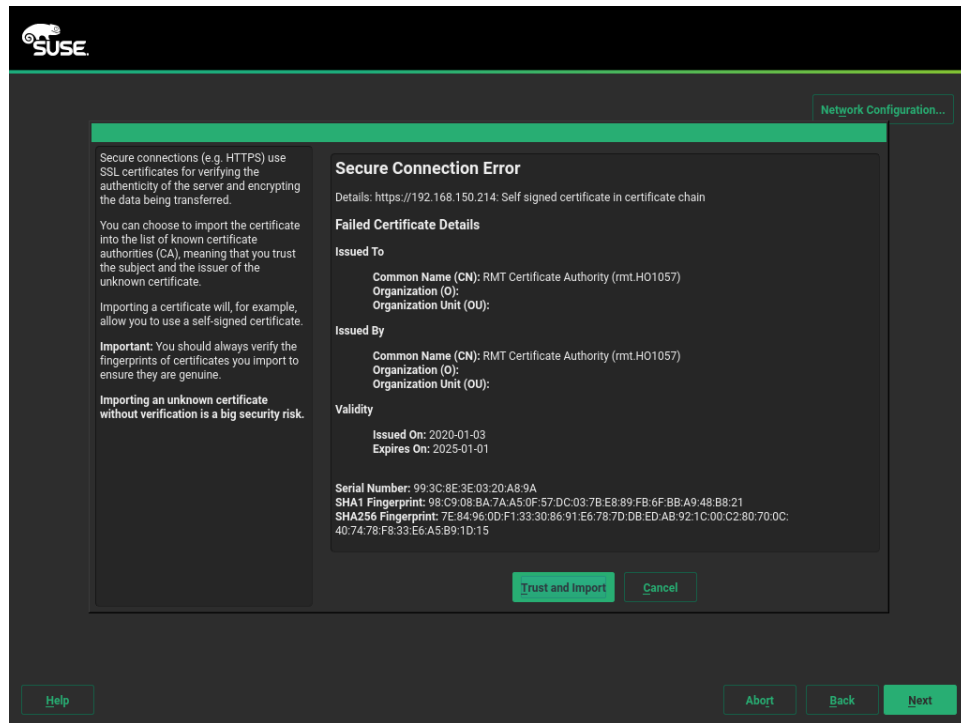
Local Registration Server URL

https://192.168.150.214

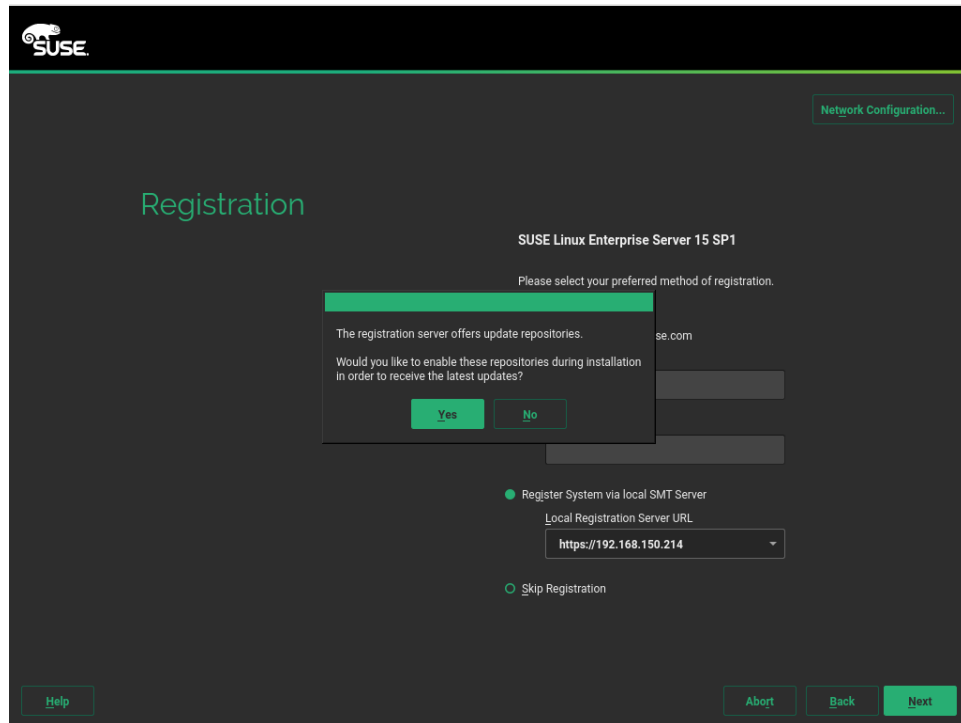
☐ Skip Registration

Help Abort Back Next

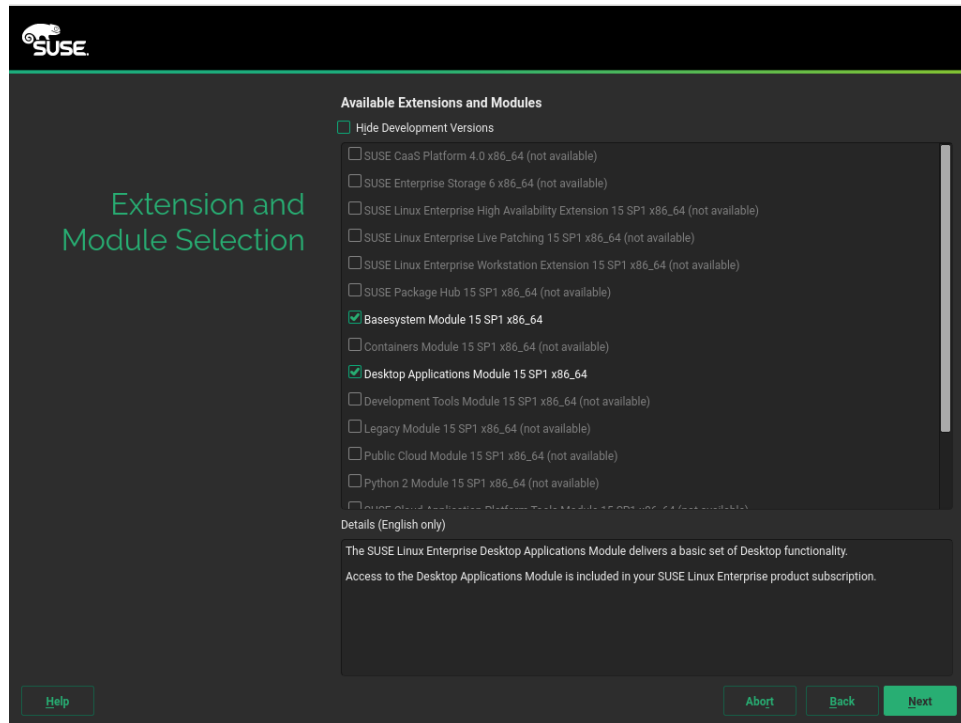
20. Click Trust and Import to accept the self-signed certificate.



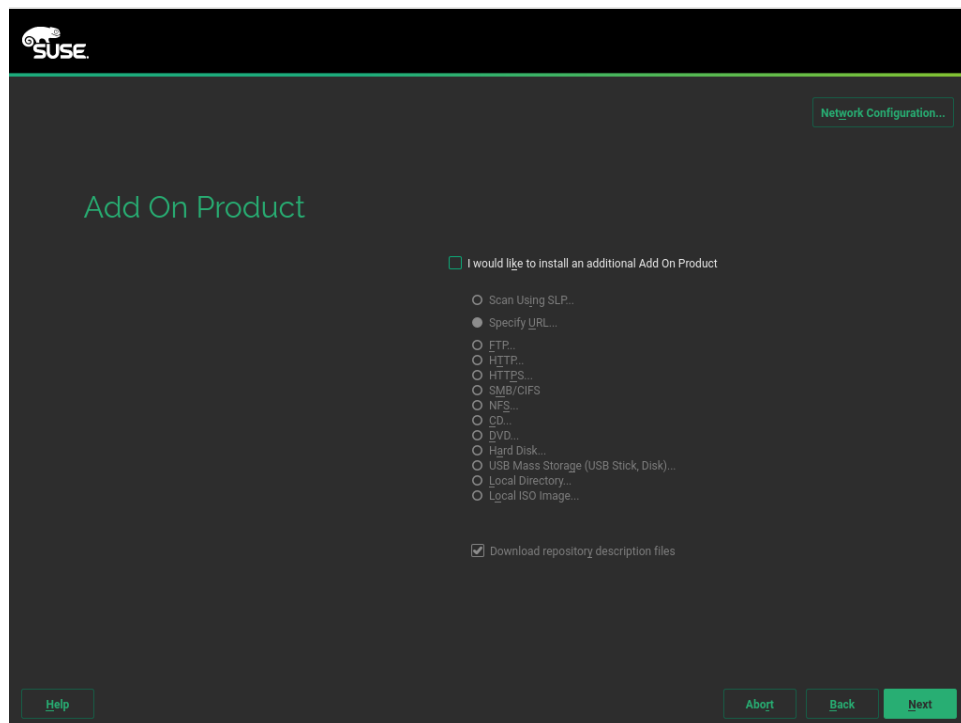
21. Click Yes to enable update repositories.



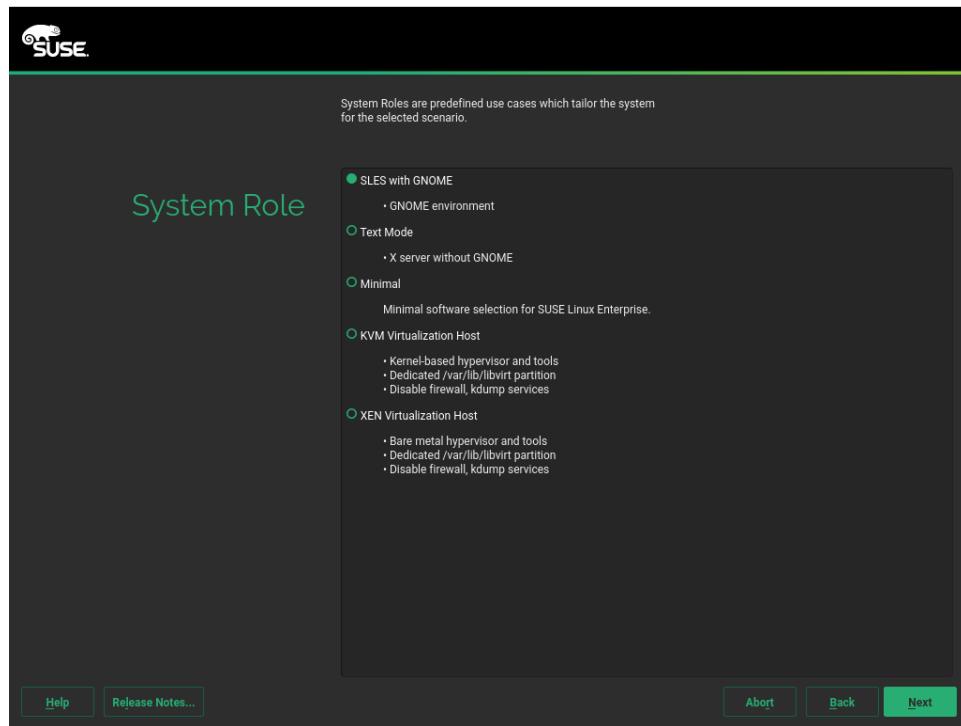
22. Uncheck “Hide Development Versions” to view other extensions and modules, select “Desktop Applications Module”, then click Next in Extension and Module Selection.



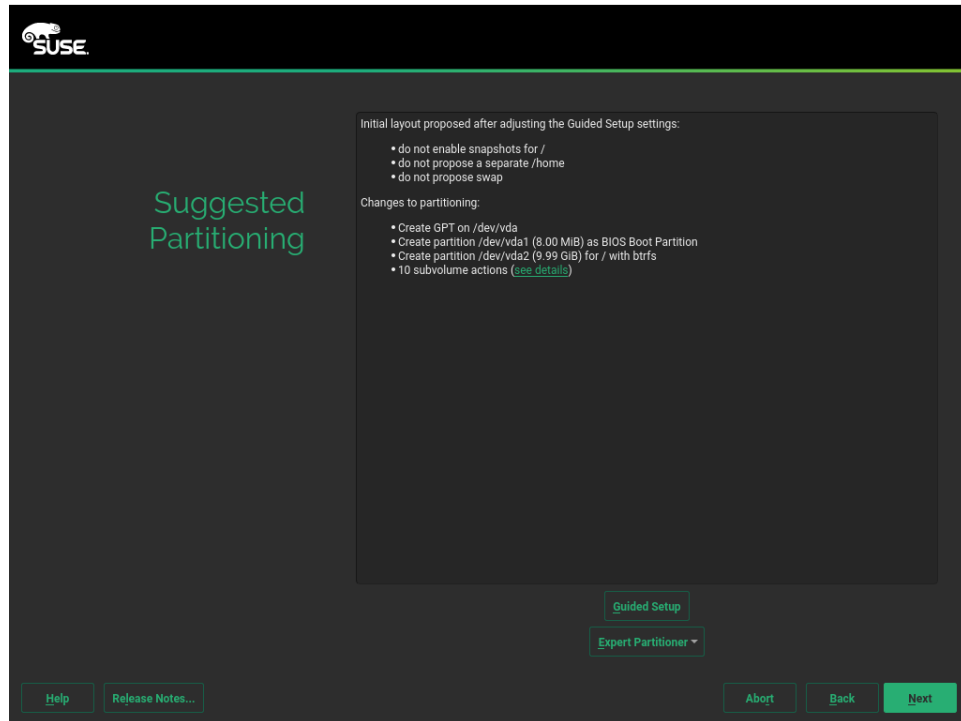
23. Click Next in Add On Product.



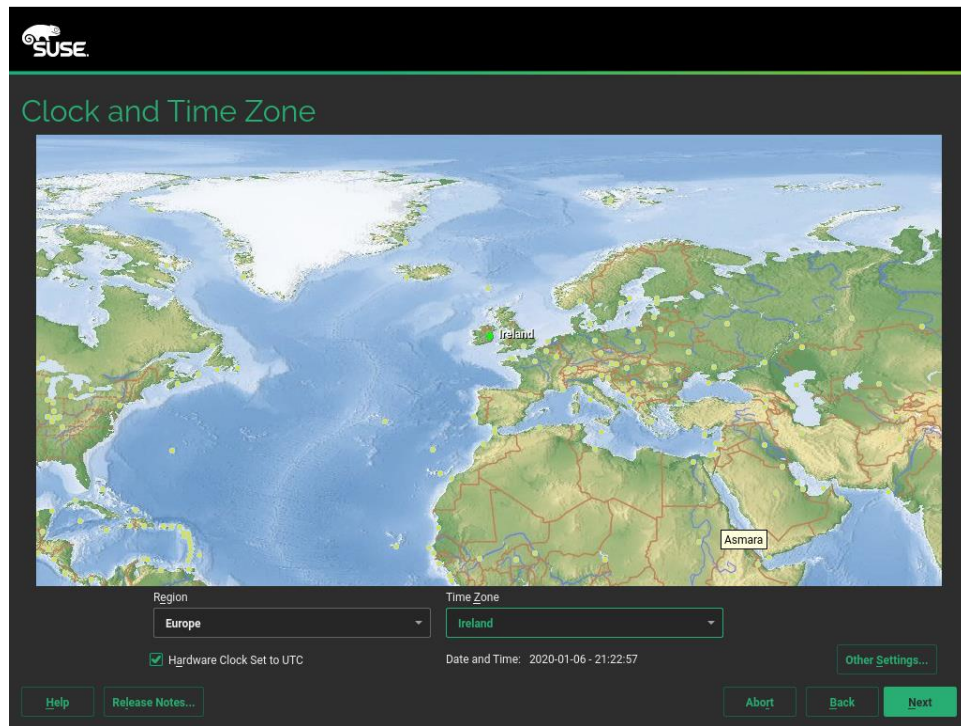
24. Select “SLES with GNOME” and click Next in System Role.



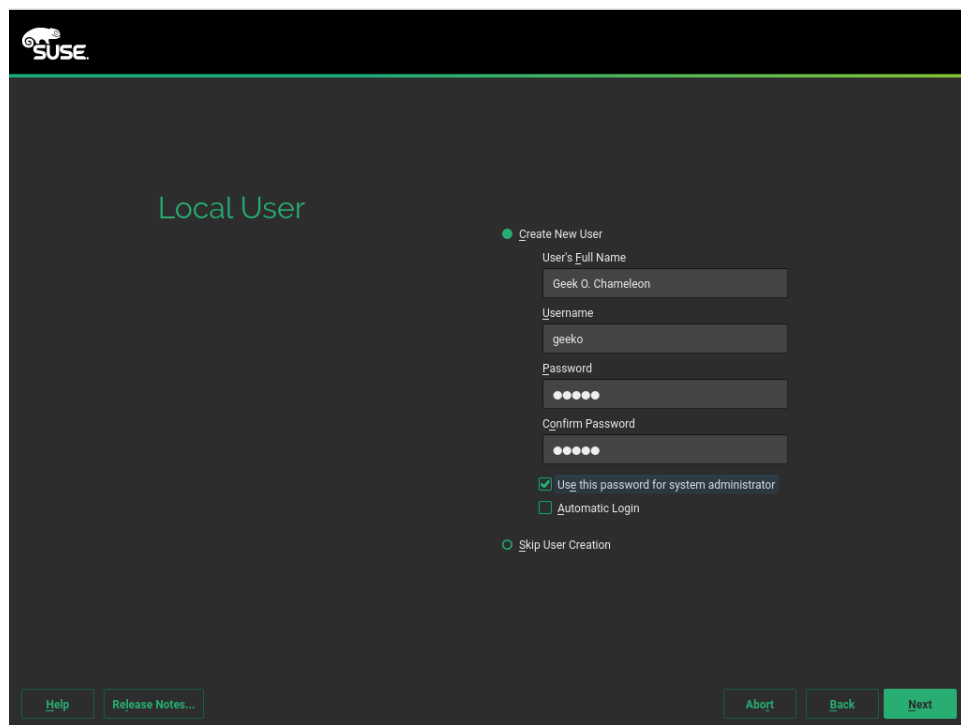
25. Click Next in Suggested Partitioning.



26. Select “Europe” and “Ireland” in Clock and Time Zone, then click Next.



27. Enter “Geek O. Chameleon” for User’s Full Name, “geeko” for Username, “linux” for Password and Confirm Password, check “Use this password for system administrator”, and click Next in Local Users.



28. Click Yes to accept the too-simple password.

Local User

Create New User

User's Full Name

The password is too simple:
it is too short.

You have used only lowercase letters for the password.

Really use this password?

Yes No

☒ Use this password for system administrator

☐ Automatic Login

☐ Skip User Creation

Help Release Notes... Abort Back Next

29. Click Install in Installation Settings. (Don't change anything in the Firewall and SSH section as it will affect the next lab)

Installation Settings

Click a headline to make changes.

Software

- Product: SUSE Linux Enterprise Server 15 SP1
- Product: BaseSystem Module
- Product: Server Applications Module
- Patterns:
 - Minimal Base System
 - Enhanced Base System
 - AppArmor
 - YaST System Administration
 - X Window System
- Size of Packages to Install: 2 GiB
- Downloading from Remote Repositories: 457.4 MiB

Booting

- Boot Loader Type: GRUB2
- Enable Trusted Boot: no
- Status Location: /dev/vda (MBR)
- Change Location:
 - Install bootcode into MBR (do not install)
 - Do not install boot code into a partition with /boot (install)

Security

- CPU Mitigations: Auto
- Firewall will be enabled (disable)
- SSH service will be enabled (disable)
- SSH port will be blocked (open)

Network Configuration

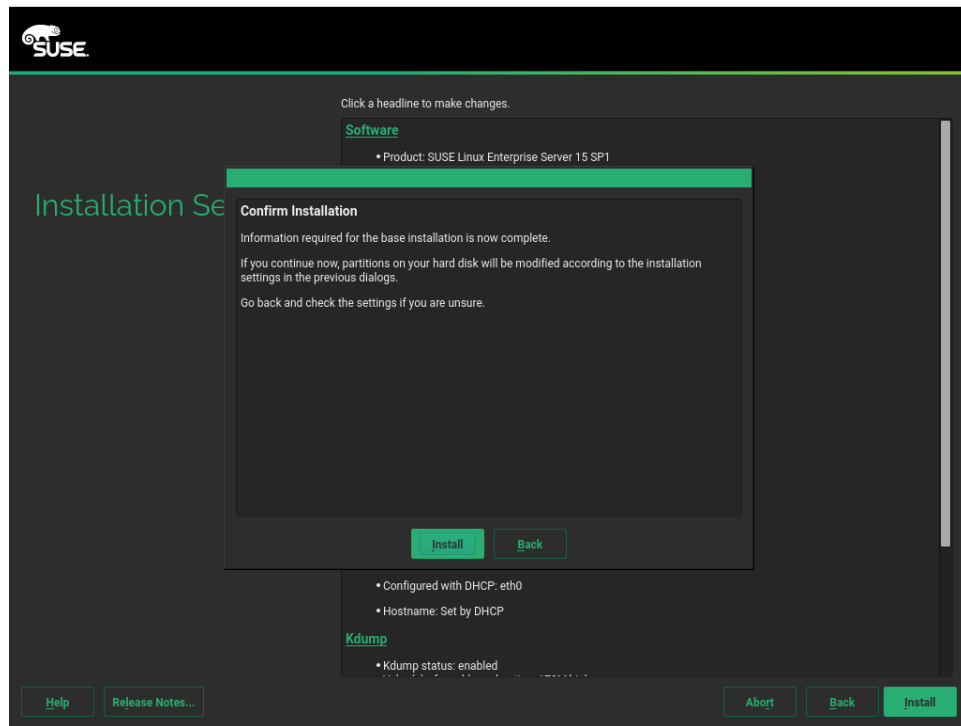
- Configured with DHCP: eth0
- Hostname: Set by DHCP

Kdump

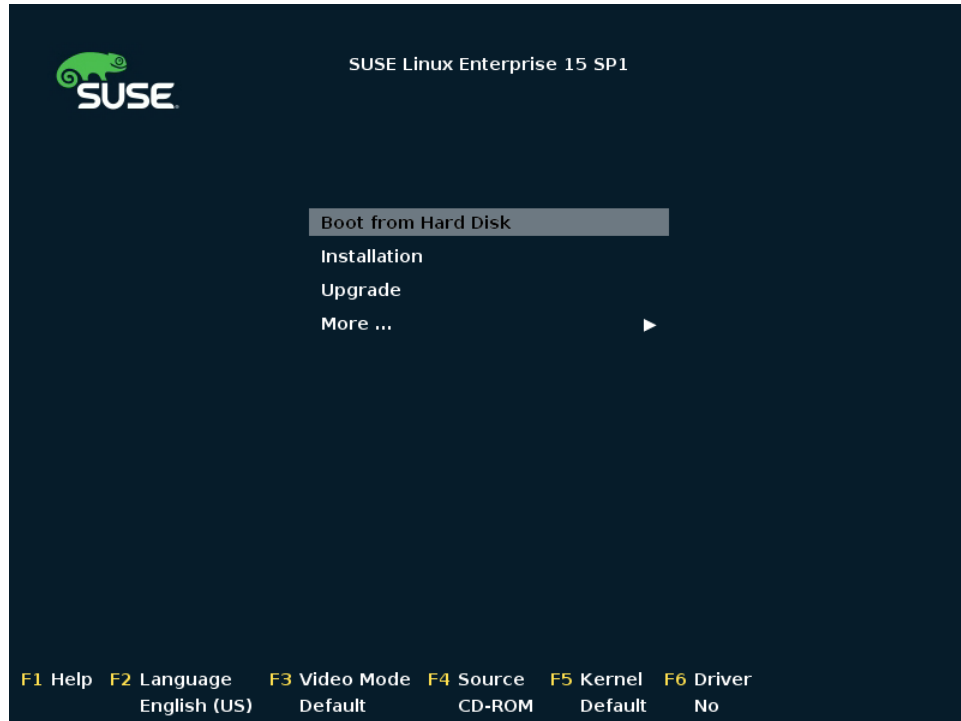
- Kdump status: enabled

Help Release Notes... Abort Back Install

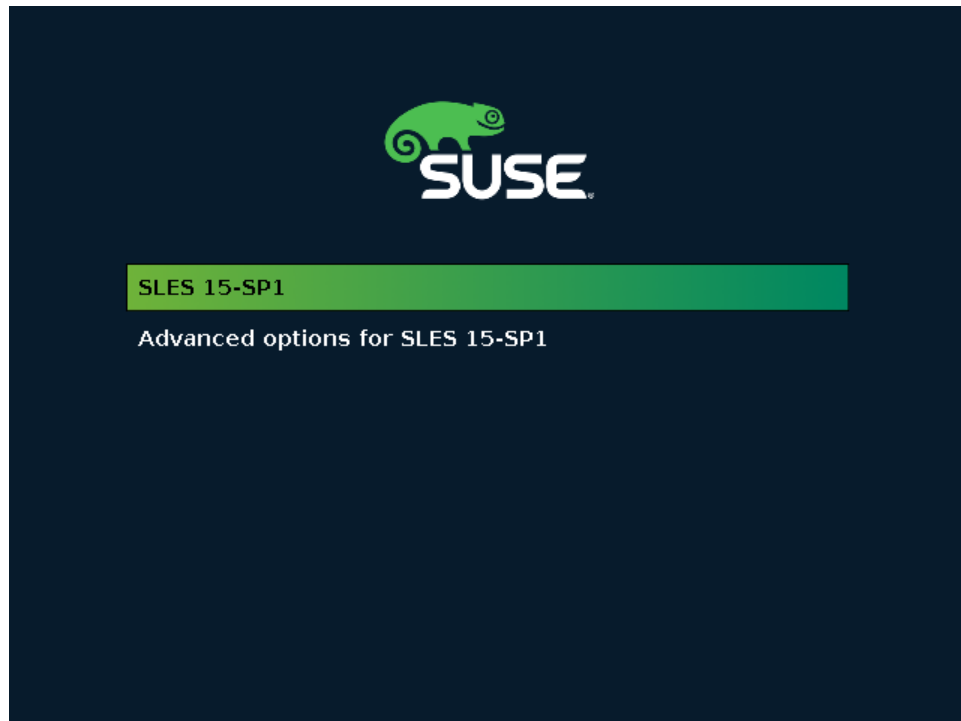
30. Click Install to confirm installation, and wait.



31. When the server reboots up from the DVD, select Boot from Hard Disk.



32. Select SLES 15.



1.2 Migrate openSUSE Leap to SLES 15

https://www.suse.com/documentation/sles-15/book_sle_upgrade/data/sec_upgrade-online_opensuse_to_sle.html

1. In virt-manager, start the opensuse15.1 VM.
2. Open a terminal on your host system and connect to the openSUSE Leap VM:

```
$ ssh root@<leap>
```
3. Check the distro and version (optional)

```
# cat /etc/os-release
```
4. Install SUSEConnect:

```
# zypper in SUSEConnect
```
5. Remove packages that produce file conflict during the migration: **NOT NECESSARY?**

```
# rpm -e --nodeps yast2-branding-openSUSE
```
6. Copy your RMT's root CA:

```
# curl --insecure -o /etc/pki/trust/anchors/rmt.crt  
https://<rmt>/rmt.crt
```
7. Install the new certificate:

```
# update-ca-certificates
```

7. Register with RMT server:

```
# SUSEConnect -u https://<rmt>/ -p SLES/15.1/x86_64
```

8. List and disable all openSUSE repositories on your system:

```
# zypper lr
# zypper rr REPO_IDS
```

Note: REPO_IDS can be stated as a "sequence expression"

Example: {7..16}

9. Add modules needed for installation:

```
# SUSEConnect --list-extensions
# SUSEConnect -p sle-module-basesystem/15.1/x86_64
```

Optional Step: will update gnome and other desktop based applications.

```
# SUSEConnect -p sle-module-desktop-applications/15.1/x86_64
```

10. Migrate installed packages to the SUSE Linux Enterprise Server repositories.

```
# zypper dup --force-resolution
```

Note: If you get a question about proceeding with conflicting files say yes. This is generally just related to branding.

Note: Don't reboot yet.

11. Remove orphaned packages:

```
# zypper rm $(zypper --no-refresh packages --orphaned | gawk
'{print $5}' | tail -n +5)
```

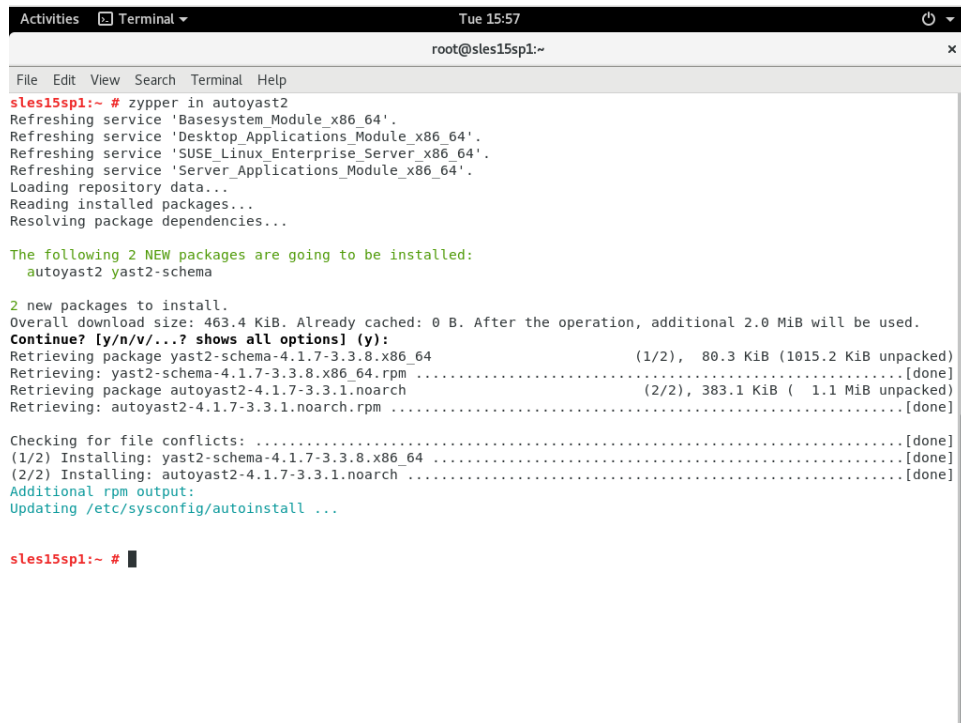
12. Reboot.

1.3 Build an Autoyast file from SLE 15 SP1 VM

Using SLES tools to create a clone of an existing system and then editing the installation script. Use the virt-manager console window for the following steps as ssh is still firewalled off.

1. Log in to the SLES 15 SP1 VM as root.
2. Launch a terminal and install the YaST AutoYast module:

```
# zypper in autoyast2
```



The terminal window shows the command `zypper in autoyast2` being executed. It displays the process of refreshing services, loading repository data, and resolving package dependencies. Two new packages, `autoyast2` and `yast2-schema`, are identified for installation. The terminal shows the download sizes and progress for these packages, along with the installation of `yast2-schema` and `autoyast2`. The command concludes with `Updating /etc/sysconfig/autoinstall ...`.

```
Activities Terminal Tue 15:57
root@sles15sp1:~

File Edit View Search Terminal Help

sles15sp1:~ # zypper in autoyast2
Refreshing service 'Basesystem_Module_x86_64'.
Refreshing service 'Desktop_Applications_Module_x86_64'.
Refreshing service 'SUSE_Linux_Enterprise_Server_x86_64'.
Refreshing service 'Server_Applications_Module_x86_64'.
Loading repository data...
Reading installed packages...
Resolving package dependencies...

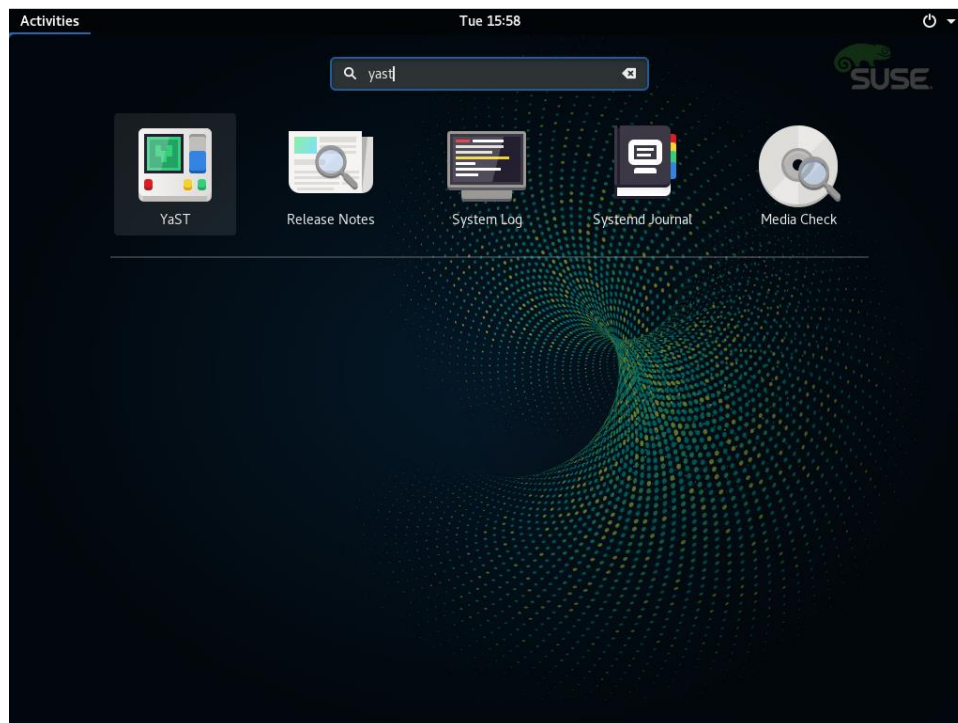
The following 2 NEW packages are going to be installed:
  autoyast2 yast2-schema

2 new packages to install.
Overall download size: 463.4 KiB. Already cached: 0 B. After the operation, additional 2.0 MiB will be used.
Continue? [y/n/v/...? shows all options] (y):
Retrieving package yast2-schema-4.1.7-3.3.8.x86_64 (1/2), 80.3 KiB (1015.2 KiB unpacked)
Retrieving: yast2-schema-4.1.7-3.3.8.x86_64.rpm .....[done]
Retrieving package autoyast2-4.1.7-3.3.1.noarch (2/2), 383.1 KiB ( 1.1 MiB unpacked)
Retrieving: autoyast2-4.1.7-3.3.1.noarch.rpm .....[done]

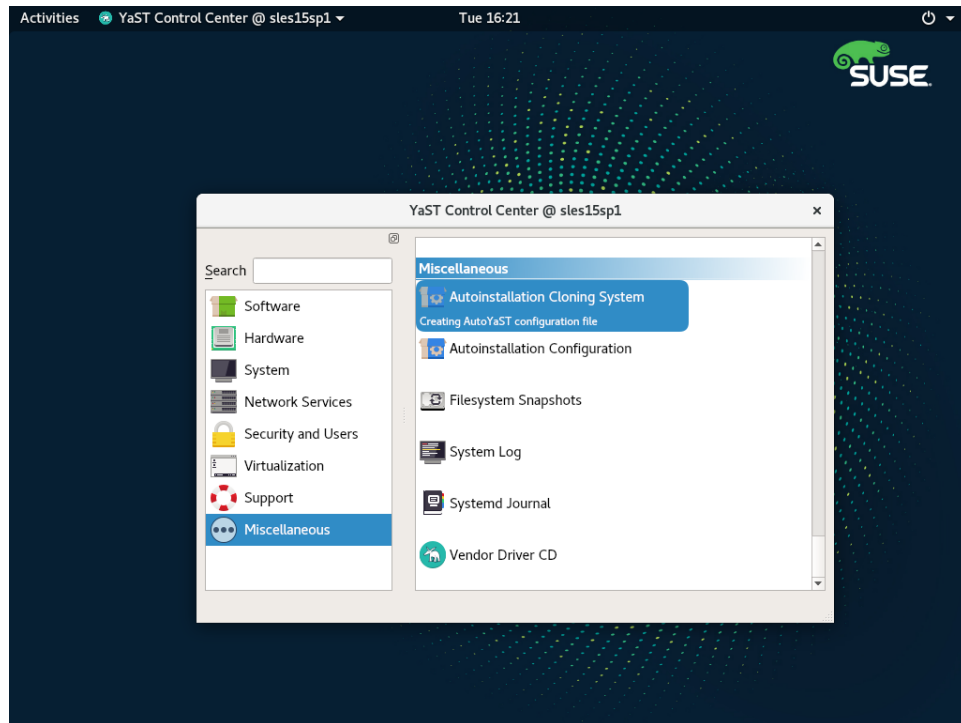
Checking for file conflicts: .....[done]
(1/2) Installing: yast2-schema-4.1.7-3.3.8.x86_64 .....[done]
(2/2) Installing: autoyast2-4.1.7-3.3.1.noarch .....[done]
Additional rpm output:
Updating /etc/sysconfig/autoinstall ...

sles15sp1:~ #
```

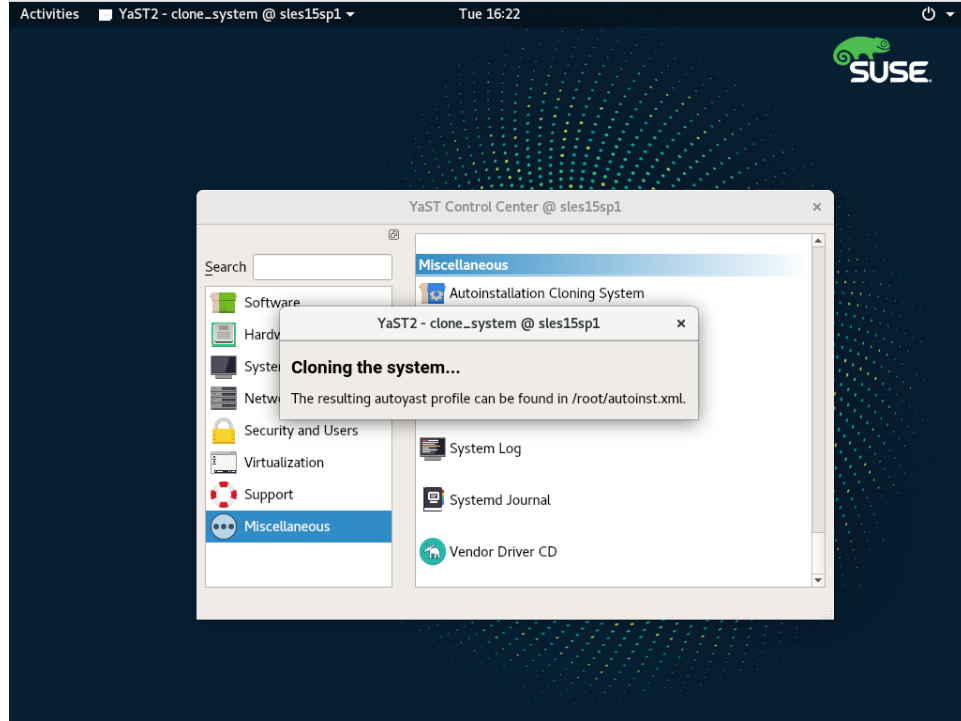
3. Click activities in the top left or hit the “windows key” to access the program search menu and type “yast”, then click the YaST icon:



4. Select the “Miscellaneous” group:



5. Select “Autoinstallation Cloning System”:

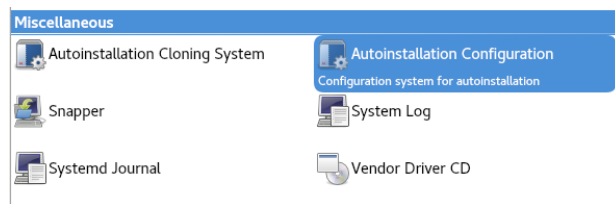


6. In a terminal window review the resulting Autoinstall.xml file in /root:

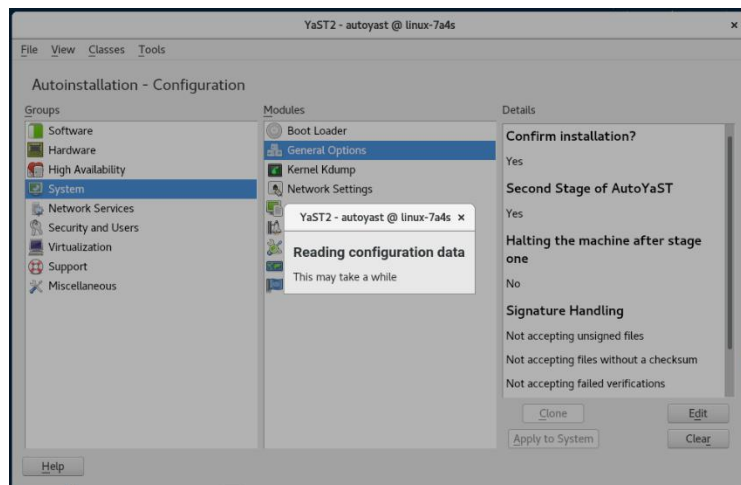
```
# cat /root/autoinst.xml
```

Notice how difficult it is to understand and edit the auto install options in the resulting XML file.

7. Use the YaST editor for Autoyast files. Select “Autoinstallation Configuration” option in YaST:



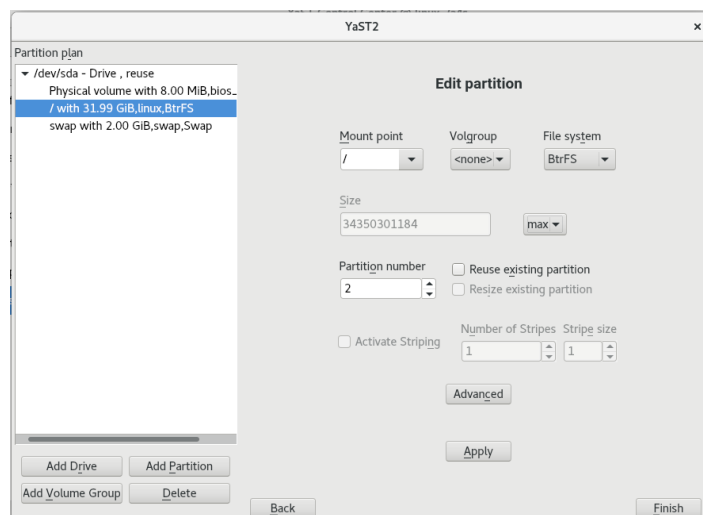
8. Open the autoyast file from the cloning operation earlier. Select File – Open - “/root/autoinstall.xml”:



Once the file has been read into the system you can edit the properties of the cloned Autoyast xml file.

9. Modify Disk Partitions. Select Hardware → Partisi → Edit.

Notice the partitioning settings are directly from the existing machine. Change it to make it more generic and select Max size instead of a specific amount:



Click Apply and Back.

10. Set Time zone

Select System – Date and Time – Edit (Button on the right)

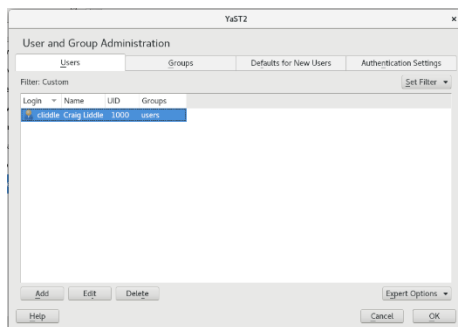
Notice this looks just like it does during the install process. Select a different time zone than you had done in the install:



Click Next

11. Add Users and Groups

Select Security and Users- User and Group Management – Edit

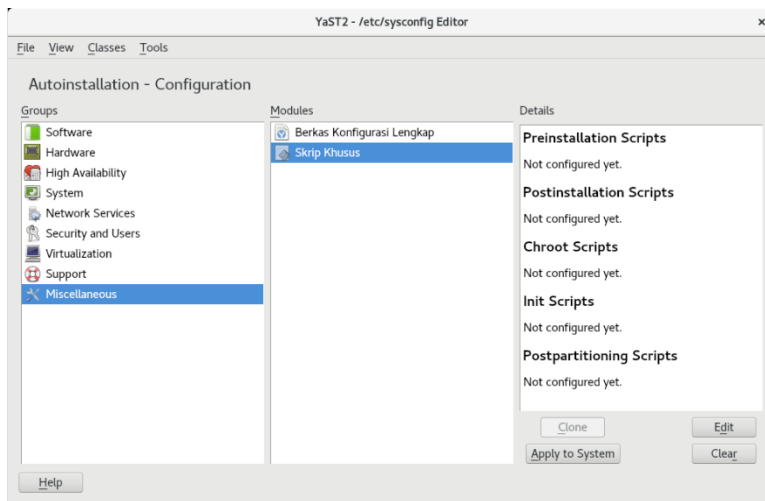


Now you can add the appropriate users and groups for any new installs.

12. Set Pre and Post installation Scripts

Select Miscellaneous → Skrip Khusus

From this interface you can insert scripts into the autoyast provisioning process



13. Save modified autoyast file

Select File → Save As → File name

You now have an autoyast file to apply to an existing machine through this interface or a file you can use to deploy as many machines as you like via Autoyast install.

For more information:

https://www.suse.com/documentation/sles-15/book_autoyast/data/overviewandconcept.html

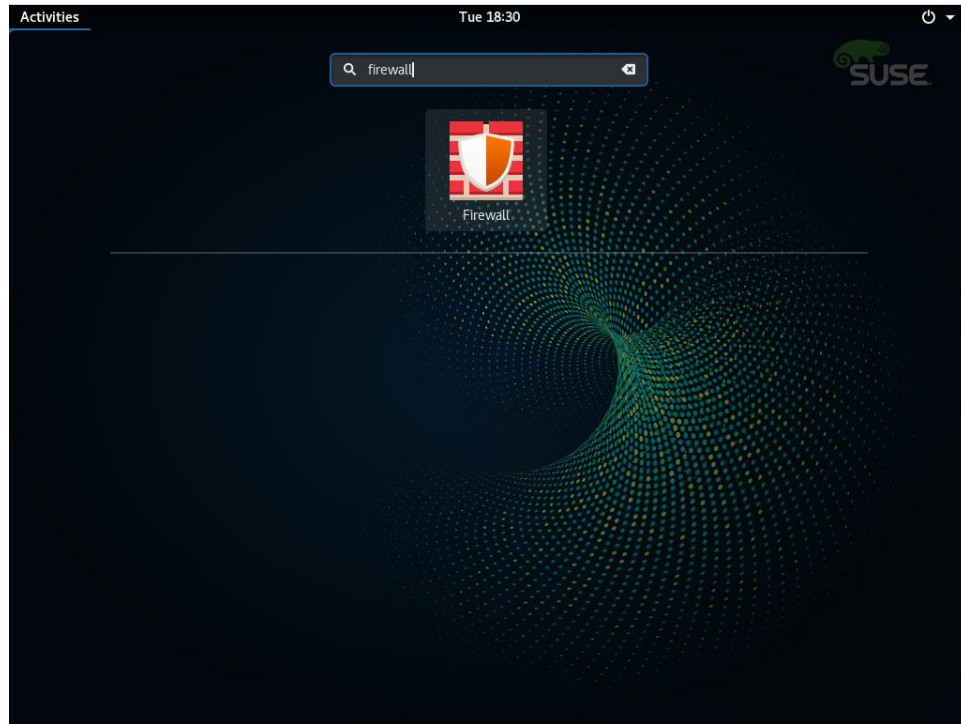
Lab 2 – Firewall

2.1 Use YaST to open SSH port for public zone

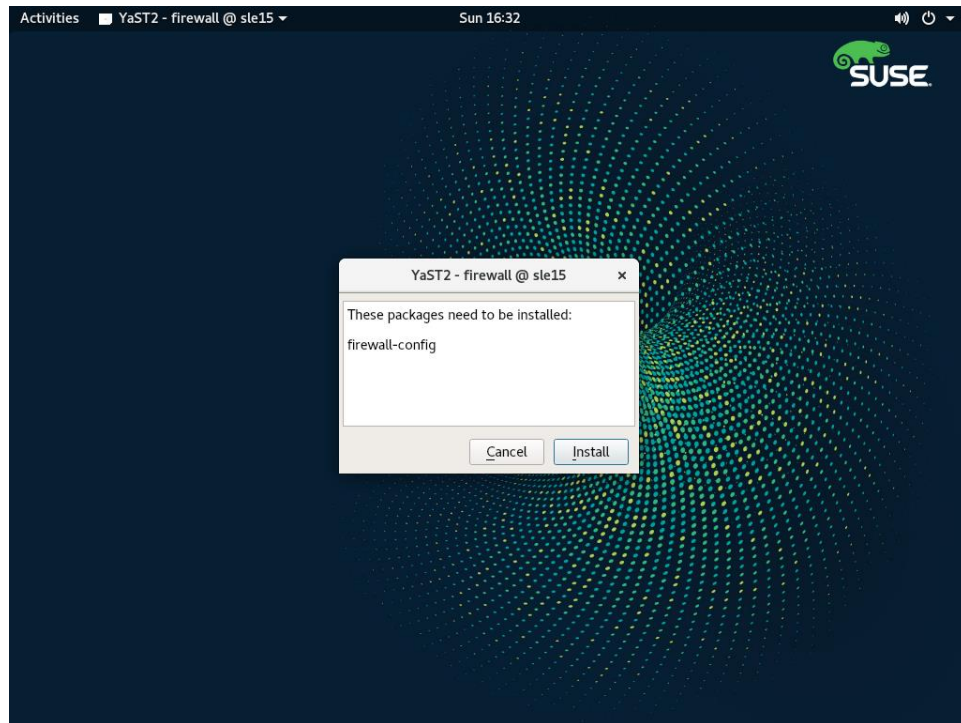
1. Attempt to ssh to the SLE 15 SP` VM from laptop (this should fail):

```
$ ssh root@<sles15sp1>
```

2. Connect to the VM in virt-manager and log in as geeko.
3. Select “Activities” and type “firewall”:

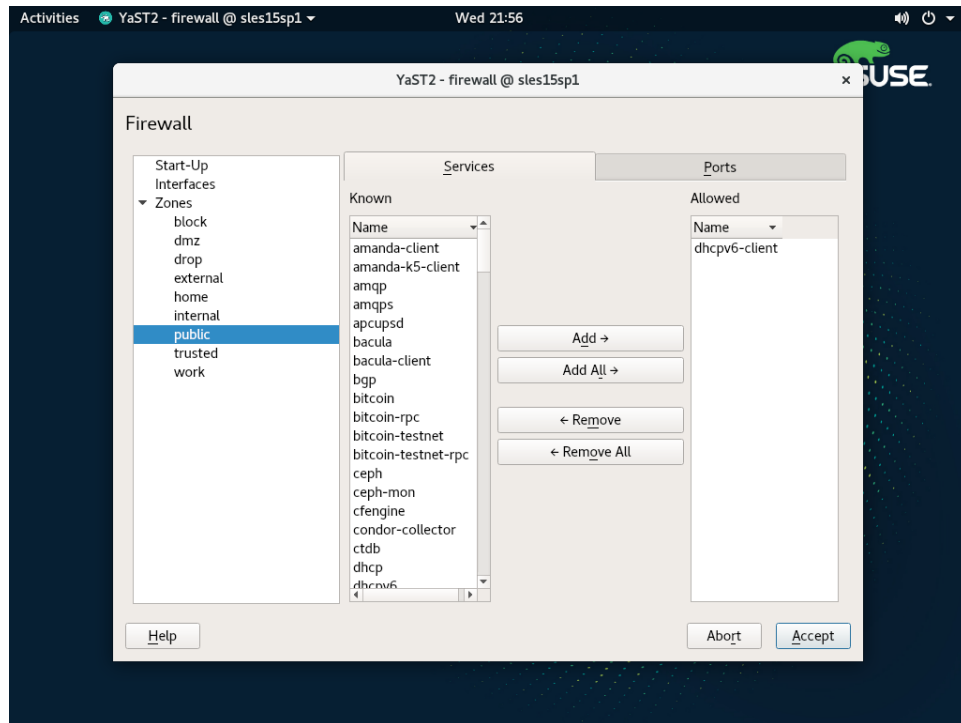


4. Click the Firewall icon and enter the root password.
5. Click Install to install the firewall-config package.

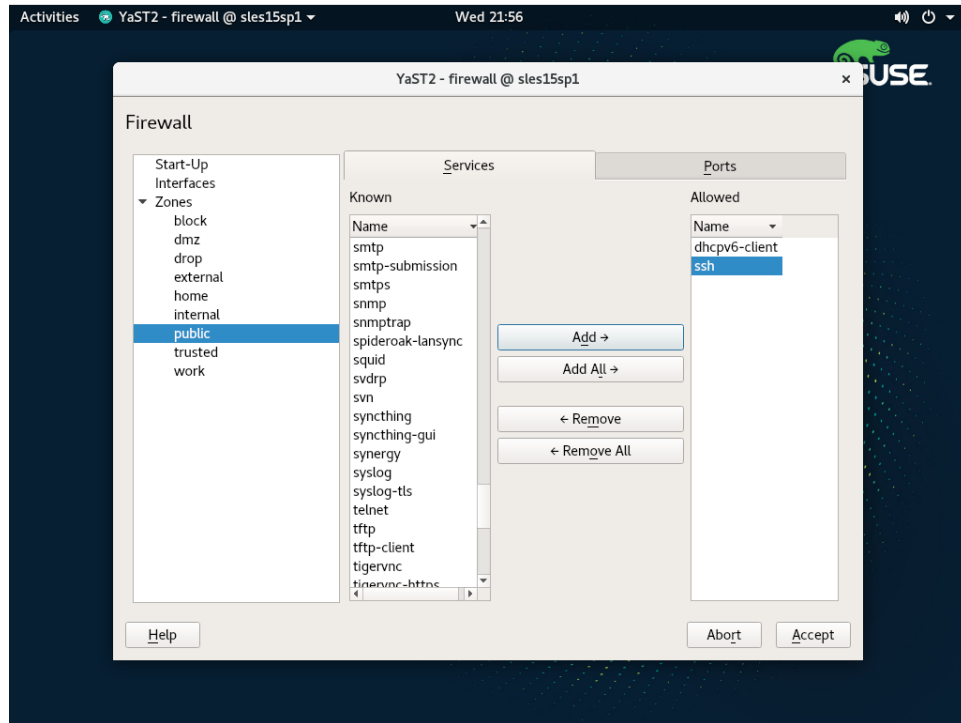


Note: The changes below only affect “runtime” and don’t persist through a reboot. You can click the drop down to change that, but you still need to change the runtime settings for immediate change OR restart the firewalld service under the options menu.

6. Select Zones -> public.



7. Select "ssh", then click Add ->.



8. Click Accept to make the changes take effect.
9. Attempt to ssh to the SLE 15 SP` VM from laptop

```
$ ssh root@<sles15sp1>
```

2.2 Use the command line to open HTTP and HTTPS ports for public zone

1. View all zones:

```
$ firewall-cmd --get-zones
```

2. View the current configuration for the default zone

```
$ firewall-cmd --list-all
```

3. View the current configuration for the trusted zone

```
$ firewall-cmd --zone=trusted --list-all
```

4. View the current configuration for the public zone

```
$ firewall-cmd --zone=public --list-all
```

5. Open HTTP port to public internet:

```
$ firewall-cmd --zone=public --add-service http --add-service  
https
```

6. Make the changes permanent:

```
$ firewall-cmd --permanent --zone=public --add-service http --  
add-service https  
$ firewall-cmd --reload
```

Lab 3 – Salt, nginx, and chrony

3.0 Setup a hosts file (optional)

This shouldn't be strictly necessary, but it is good practice to base salt on hostnames rather than IP addresses. Since we don't have a DNS setup here it would need to be done through `/etc/hosts`. At a minimum we need to include our rmt server and the sles15sp1 VM we just installed. You may also pull in other VMs from previous labs if you want to have more to test with. We will leave that up to you.

1. Edit the hosts file

```
$ sudo vi /etc/hosts
```

2. Include lines similar to the following, this can vary depending on your IP address and hostname

```
192.168.150.215 rmt.HO1057 rmt salt
```

```
192.168.150.202 sles15sp1.HO1057 sles15sp1
```

3. Copy the file to all machines you plan to test with, example below:

```
$ scp /etc/hosts root@<sles15sp1>:/etc/hosts
```

3.1 Register the new system to the Salt master

1. Connect to all VMs you plan to include if not already

```
$ ssh root@<rmt>
```

```
$ ssh root@<sles15sp1>
```

```
...
```

2. Check what is installed already

```
$ zypper se salt
```

You will see that the salt master is already installed on the rmt server, but nothing on the others.

3. Install salt minions

```
$ zypper in salt-minion
```

Complete for each system in the lab

4. Start the Salt minion:

```
$ systemctl start salt-minion
```

Complete for each system in the lab

5. Enable the minions so they persist through reboot

```
$ systemctl enable salt-minion
```

Complete for each system in the lab

6. Verify the minions are requesting access

```
$ salt-key
```

7. Accept the Salt minion on the **rmt server**

```
$ salt-key -A
```

8. Ping the Salt minion

```
$ salt '*' test.ping
```

Note: It may take a minute or so before this command will work as there is some initialization taking place after the keys are accepted.

9. Create salt filesrv

```
$ vi /etc/salt/master
```

Search for the line in the file that includes '/srv/salt'. In vi you can type a '/' to open the search pane and type the folder in there and hit enter. By default this should take you to line 661. Uncomment the following section:

```
file_roots:
  base:
    - /srv/salt/
```

Ensure there are no leading spaces in the lines. The first line should have no spaces, the next has 2, then 4.

3.2 Configure nginx web server using Salt

1. Review the nginx.sls state file

```
$ cat /srv/salt/webserver/nginx.sls
```

This installs the package, turns on the service and enables it so it turns on during boot. The watch command will check for differences in the conf file when the state is run and recopy it if it isn't the same. This can be done as a daily task. We then also insert a index file for the web server and define the managed conf file.

```
pkg_nginx:
  pkg.installed:
    - name: nginx

srv_nginx:
  service.running:
    - name: nginx
    - enable: True
    - watch:
      - pkg: nginx
      - file: /etc/nginx/nginx.conf
```

```
push_index:
  file.managed:
    - name: /srv/www/htdocs/index.html
    - source: salt://webserver/index.html

cfg_nginx:
  file.managed:
    - name: /etc/nginx/nginx.conf
    - source: salt://webserver/nginx.conf
```

2. Apply salt state to start nginx and load index.html (use the rmt terminal session)

```
$ salt 'sles15sp1*' state.apply webserver.nginx
```

Note: The wildcard is just so we don't need to type the FQDN, the definition of "webserver.nginx" is based on the files we moved into the /srv/salt/ directory which we previously set at the root of the salt fileserver.

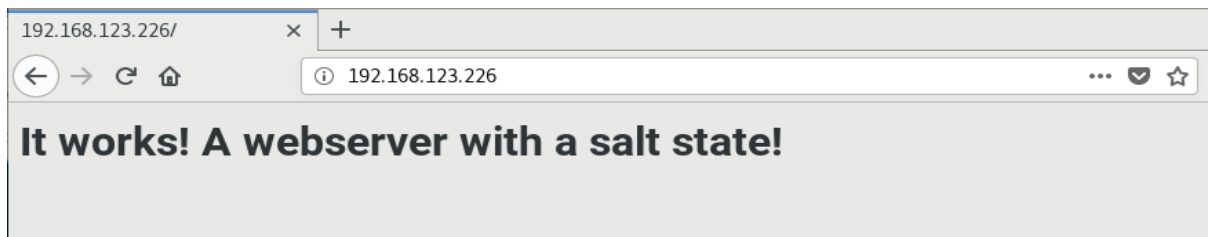
3. Verify the web server is running

```
$ systemctl status nginx
```

Note: The status may not update immediately and take a minute.

4. Open in Firefox

On the host system open a browser and go to `http://<sles15sp1>`



3.3 Configure NTP server using chrony

1. Review the ntp state file

```
$ cat /srv/salt/chrony/ntp.sls
```

This will install chrony (if not already), turn it on, enable it, and then manage the config file.

```
pkg_ntp:
  pkg.installed:
    - name: chrony

srv_ntp:
  service.running:
    - name: chronyd
    - enable: True
    - watch:
```

```
- pkg: chrony
- file: /etc/chrony.conf

cfg_ntp:
  file.managed:
    - name: /etc/chrony.conf
    - source: salt://chrony/chrony.conf
    - user: root
    - group: chrony
    - mode: 640
```

2. Review the top part of the chrony.conf file

```
$ vi /srv/salt/chrony/chrony.conf
```

If the IP address in the “pool” setting at the top of the file doesn’t match the IP of your rmt server, then edit the file to fix it.

3. Apply salt state to add NTP server setting

```
$ salt 'sles15sp1*' state.apply chrony.ntp
```

4. Verify the setting was changed

```
$ vi /etc/chrony.conf
```

```
$ chronyc sources
```

Note: chronyc is a command line tool for interacting with chrony settings and is added here as just an optional step you can take as an alternative to see that the NTP setting is updated. If you get a message that it cannot talk to the daemon then the chrony service isn’t started.

3.4 Understanding highstate

1. Review the top.sls file

```
$ cat /srv/salt/top.sls
```

This file is very basic and just outlines what states to apply to each system by hostname. This can get much more complex however and be organized by other settings aside from hostname. You can also define groups of rules that are outside ‘base’ such as ‘dev’ or ‘prod’

```
base:
  '*':
    - chrony.ntp
  'sles15sp1*':
    - webserver.nginx
```

2. Revert all states we have done so far

```
$ salt 'sles15sp1*' state.apply webserver.nonginx
```

```
$ salt 'sles15sp1*' state.apply chrony.nontp
```

3. Apply the highstate to all systems

```
$ salt '*' state.highstate
```


Lab 4 – Repository Management Tool

4.1 View RMT Status

1. Connect to the RMT VM via SSH:

```
$ ssh root@<rmt>
```

2. View the RMT timers

```
# systemctl status rmt-server-sync.timer rmt-server-mirror.timer
```

3. View the RMT services:

```
# systemctl status rmt-server rmt-server-sync rmt-server-mirror
```

4.2 View products and repositories

1. View mirrored products:

```
# rmt-cli products list
```

2. View all available products:

```
# rmt-cli products list --all
```

3. View mirrored repositories:

```
# rmt-cli repos list
```

4. View all available repositories:

```
# rmt-cli repos list --all
```

Lab 5 – Software Management

5.1 Use zypper to search for a package

1. Connect to the RMT VM via SSH:

```
$ ssh root@<rmt>
```

Note: we are using the RMT server because it is connected to SUSE Customer Center. Global package search does not yet work with RMT as the installation source.

2. Search for gcc-objc in currently assigned repositories:

```
# zypper se gcc-objc
```

3. When asked “Do you want to run 'zypper search-packages' now?
[yes/no/always/never] (no):” enter yes.

4. You can also search for gcc-objc across all available repositories directly from the command line:

```
# zypper search-packages gcc-objc
```

5.2 Use SUSE Customer Center to search for a package

1. Browse to the following URL in Firefox:

```
https://scc.suse.com/packages
```

2. Fill out the fields as shown and click “Search”:

The screenshot shows the SUSE Customer Center web interface for searching packages. The browser address bar shows the URL: `scc.suse.com/packages?name=SUSE%20Linux%20Enterprise%20Server&version=15.1&arch=x86_64`. The page title is "SUSE Customer Center" with "Sign in" and "English" links. The main heading is "Packages" with the subtitle "Find packages by product". A feedback box asks for input on the package search. The "Select your product" section has a "Name" dropdown menu with options: "SUSE CaaS Platform", "SUSE Linux Enterprise Desktop", "SUSE Linux Enterprise High Performance Computing", "SUSE Linux Enterprise Real Time", "SUSE Linux Enterprise Server" (selected), "SUSE Linux Enterprise Server for SAP Applications", and "SUSE Manager Server". Below this are "Version" and "Architecture" dropdown menus. The "Version" menu shows "15 SP2", "15 SP1" (selected), "15", "12 SP5", "12 SP4", "12 SP3", and "12 SP2". The "Architecture" menu shows "aarch64", "ppc64le", "s390x", and "x86_64" (selected). The "Search packages" section shows the search criteria: "in SUSE Linux Enterprise Server 15 SP1 x86_64". There are two input fields: "Matching name" with the text "gcc-objc" and "In module" with the value "(all modules)". A green "Search" button is at the bottom.

1. Click on “gcc-objc” to view additional details about the package:

The screenshot shows the SUSE Customer Center interface. A modal window titled "gcc-objc" is open, displaying the following details:

- Version: 7-3.3.22
- License: GPL-3.0+
- Architecture: x86_64
- Repository: SLE-Module-DevTools15-SP1-Pool
- Filename: gcc-objc-7-3.3.22.x86_64.rpm

Below the details, there is a description: "The system GNU Objective C Compiler. The system GNU Objective C Compiler." and a link to "Sign in to download packages".

The modal also indicates that the package can be found in the "Development Tools Module 15 SP1 x86_64". It provides instructions on how to install the package:

To install the package on your system, first activate this module on your registered system:

```
SUSEConnect -p sle-module-development-tools/15.1/x86_64
```

After this module is activated on your system, you can install the package:

```
zypper install gcc-objc-7-3.3.22
```

Below the modal, the main page shows "1 package" with name containing "gcc-objc" in SUSE Linux Enterprise Server 15 SP1 x86_64. It includes a table with the following data:

| Name | Version | License | Module |
|--------------------------|----------|----------|---|
| gcc-objc | 7-3.3.22 | GPL-3.0+ | Development Tools Module 15 SP1 |

At the bottom of the page, there is a footer with copyright information and links to various resources.

Lab 6 – Extra Credit

6.1 Migrate SLES 12 SP4 to SLES 15

https://www.suse.com/documentation/sles-15/singlehtml/book_sle_upgrade/book_sle_upgrade.html#cha.upgrade-offline

1. In virt-manager, boot the sles12sp4 VM from the DVD.
2. Select Boot from Hard Disk and hit Return
3. Select "SLES 12-SP4"
4. SSH into the VM

```
$ ssh root@<sles12sp4>
```

5. Remove any existing registrations

```
# SUSEConnect --cleanup
```

6. Reboot the VM from DVD
7. Select Upgrade, enter "media_upgrade=1" in Boot Options, and hit Return
8. Click Next in "Language and Keyboard Selection"
9. Click Next in "Select for Update"
10. Check "I Agree to the License Terms" and click Next in the License Agreement
11. Click Next in Previously Used Repositories
12. Click OK in Media Base Upgrade
13. Select DVD... in Add On Product and click Next
14. In VM console, select View -> Details
15. Select SATA CDROM1
16. In Source path select "SLE-15-Packages-x86_64-GM-DVD1.iso"
17. Click Apply
18. In VM console, select View -> Console
19. Click Continue
20. Select Basesystem-Module 15-0 and SLES15 15-0 in Extension and Module Selection
21. Click Next
22. Click Next in Add-On Product Installation
23. Click Update in Installation Settings
24. Click Start Update in Confirm Update
25. Change the DVD as requested following steps 14-17