Week 01

IT-230

- Course Introduction
- Accessing the Lab Environments
- Linux Command-Line Refresher

IT-230 · Linux Administration

IT-230

Course Introduction

Sean Twiehaus

- Tech Lead Software Engineer
- **RHCSA Certified** (RHEL 7 and 9)
- **RHCE Certified** (RHEL 10)

- 20 years in the IT industry
- 15 years in Ops:
  - Field Service Agent
  - Windows Administrator
  - Linux Systems Administrator
  - Automation Engineer
- 5 years in Dev:
  - Full Stack Software Engineer
  - Tech Lead
- **First Learned Linux in 2014 as SCC**

Introduction (Preview)

What is your name and study program?

(Administration, DevOps, Cybersecurity, Programming)

Sean Twiehaus — Administration

Do you have any experience with Linux outside of class?

Using Linux daily since 2014, professionally and personally (homelab)

Do you plan to take the RHCSA Exam?

Yes

SCC Linux Courses

IT-130 — Intro to Linux ✅

- First half of Red Hat Certified System Administrator (RHCSA) exam preparation.
- Red Hat Academy course: RH124
- Red Hat Enterprise Linux version 9.3

IT-230 — Linux Administration 😎

- Second half of Red Hat Certified System Administrator (RHCSA) exam preparation.
- Red Hat Academy course: RH134
- Red Hat Enterprise Linux version 10.0

Red Hat Enterprise Linux (RHEL) 10

- Released May 20th, 2025
- Red Hat Academy updated its material November 2025
  - Added course sections for new features
    - Flatpaks
    - Image mode
- This course is on **RHEL 10.0**; Intro to Linux was on RHEL 9.3
- **They are about 98% the same**
- When signing up for the RHCSA exam, you can choose version 9 or 10

Red Hat Certified Systems Administrator (RHCSA) Exam

- Normally $500
- Includes one **free** retake
- You will receive a **50% discount** voucher
- Hands-on, task-based

Example RHCSA task

Create users jane and john and make them members of the admins group as a secondary group membership.

RHCSA Cert Guide

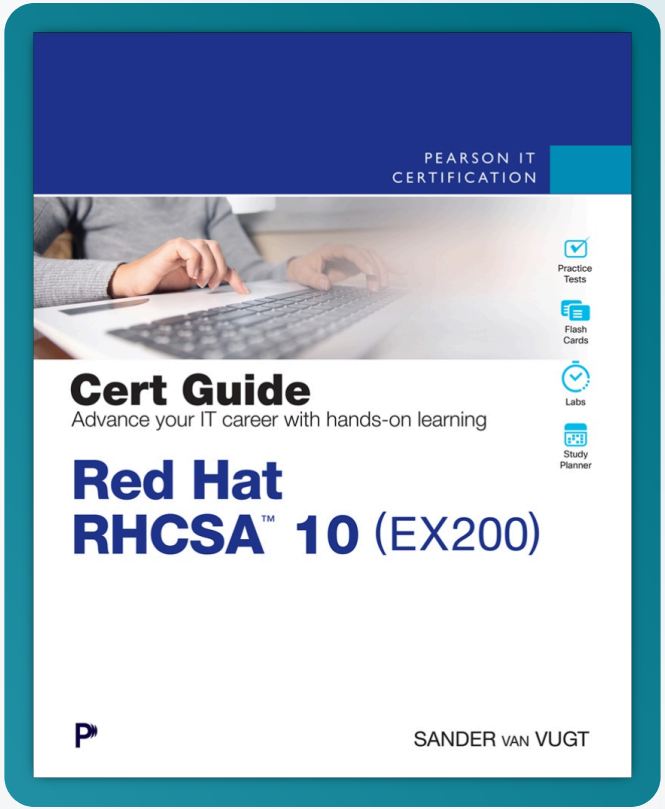
Red Hat RHCSA 10 Cert Guide: EX200

By: Sander van Vugt

Published by Pearson IT Certification

ISBN-10: 0-13-557666-0

ISBN-13: 978-0-13-557666-3



Effortful practice is required to pass the RHCSA Exam

Effortful practice is required to learn

Linux Administration

Four Learning Opportunities Each Week

Before class

1. Read the Red Hat Academy material and practice the Guided Exercises

During class

2. I will show the material (presentations and demonstrations)
3. We will work through in-class exercises

After class

4. Complete graded lab assignments

AI Is a Tool

IT workers are being encouraged to use AI

It isn't prudent to prohibit students from using AI

Realize AI is a tool

Tools can be used positively or negatively

AI Usage

Negative Use

Replaces or diminishes effortful practice

Example

Asking AI to do your homework for you

Positive Use

Enables more effortful practice

Example

Writing a detailed prompt when you are stuck

AI Guidelines

Treat AI like a classmate

Do ask AI for specific help when you are stuck

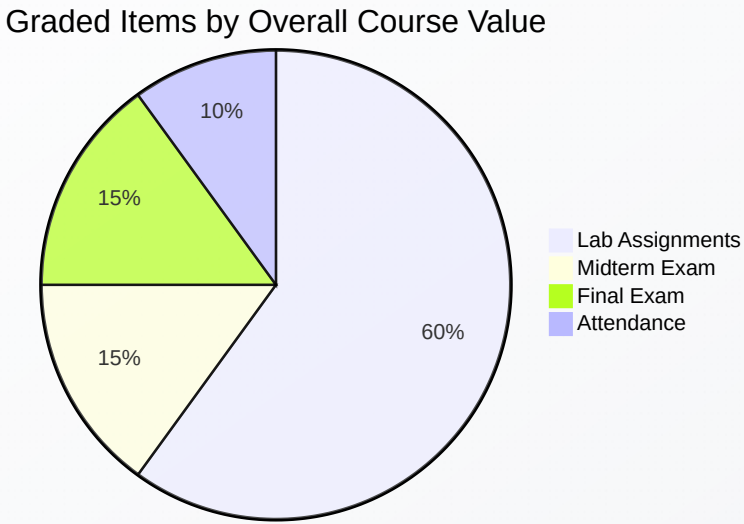
Do **NOT** ask AI do your homework for you

Caution

Use of AI is prohibited on all exams

Grades

Emphasis on hands-on work, just like the RH exams!



Graded Items	Overall Course Value
Lab Assignments	60%
Midterm Exam	15%
Final Exam	15%
Attendance	10%

IT-230 Exams

Very Challenging

Essentially shortened RHCSA practice exams

- Taken in the **SCC Lab** environment
- **Open "book"**: Red Hat Academy website
- **Open notes**: hand-written or hand-typed only

Caution

No other website, material, or resource is allowed

Late Policy

- Assignments are due the **following Tuesday at 11:59 p.m.**
- Late work is accepted for up to two weeks after the due date
  - **20% is deducted from every late assignment**
- Two weeks after the due date, assignments lock
- **Once assignments are locked, submissions are disabled**
- All assignments must be submitted before the final exam

Example

- Today assignments are given
- Next Tuesday at 11:59 p.m. assignments are due
- Two weeks after they are due, assignments lock

10 minute break every 50 minutes.

First break: 6:50 p.m.

Second break: 7:50 p.m.

Cameras On During Class

Except between first and second break

Office Hours

Mondays 7p.m. to 8p.m. central time

Except during holidays and breaks

Between Classes

1. Turn in Lab assignments

2. Read the next chapter and practice the Guided Exercises

Introduction

What is your name and study program?

(Administration, DevOps, Cybersecurity, Programming)

Do you have any experience with Linux outside of class?

Do you plan to take the RHCSA Exam?

IT-230 · Linux Administration

Accessing the Lab Environments

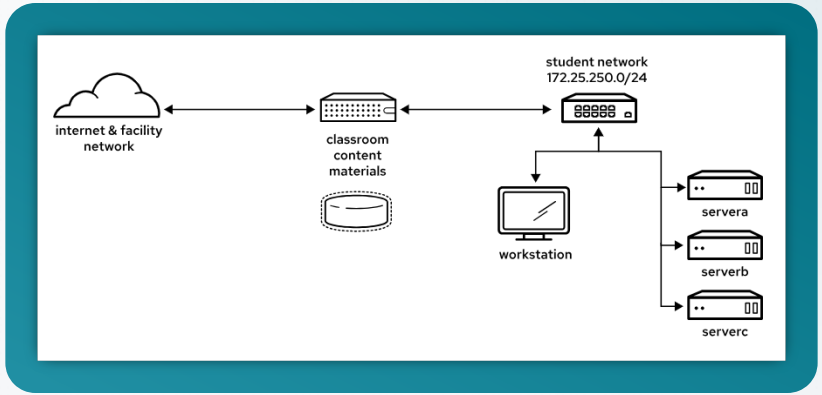
Console, Terminal, Shell

IT-230 Lab Environments

Two lab environments with nearly identical configurations

RHA Lab

- For Guided Exercises and RHA Labs
- Hosted on **Red Hat Academy**
- Uses Linux virtualization



SCC Lab

- For in-class exercises and SCC labs
- Hosted on the **SCC VDI**
- Uses VirtualBox

IT-230 Lab Environments

Admin user

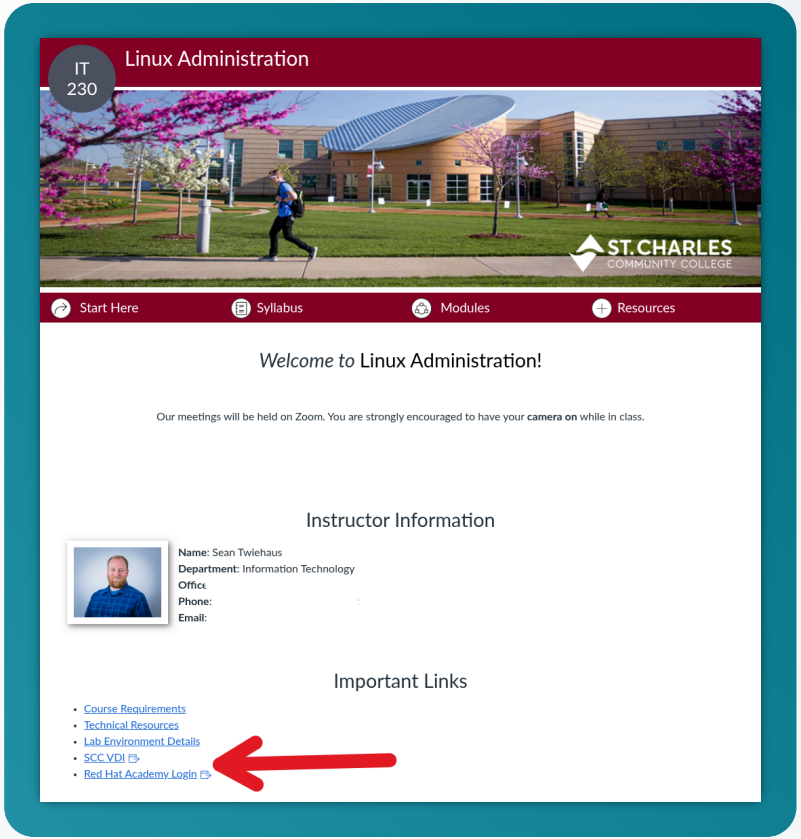
- User: `student`
- Password: `student`
- Allowed to SSH and use `sudo`

Super user

- User: `root`
- Password: `redhat`
- Allowed to SSH

Name	IP	System type	CPU	Memory (MB)	Disks (thin provisioned)
workstation	172.25.250.9	Graphical	2	4096	1 × 20GB (OS)
servera	172.25.250.10	Headless	1	2048	1 × 10GB (OS), 3 × 5GB (blank)
serverb	172.25.250.11	Headless	1	2048	1 × 10GB (OS), 3 × 5GB (blank)
serverc	172.25.250.12	Headless	1	2048	1 × 10GB (OS), 3 × 5GB (blank)

Links to both lab environments on the course home page



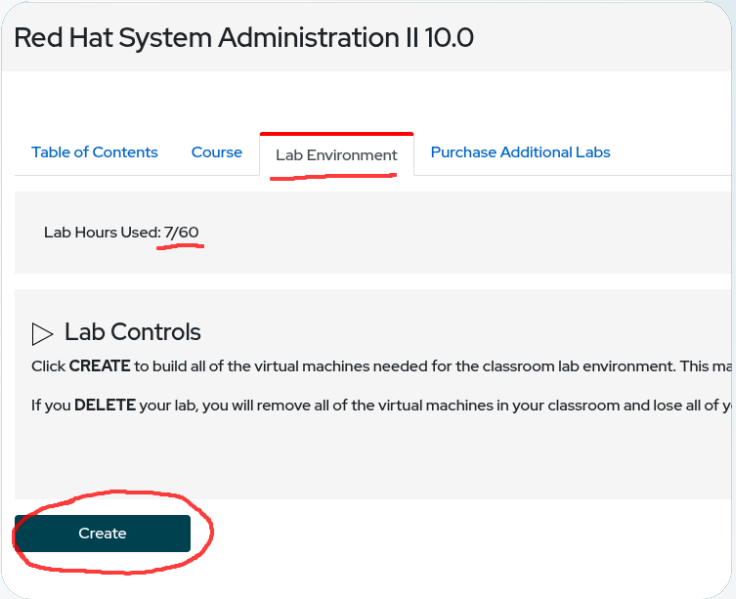
IT-230 · Linux Administration

Accessing the RHA Lab

Create Lab Environment

1. Click the `Lab Environment` tab.
2. Click the `Create` button

> **Warning**
>
> **You have 60 hours of RHA lab time**
>
> **Stop your VMs when you are not using them**



IT-230 · Linux Administration

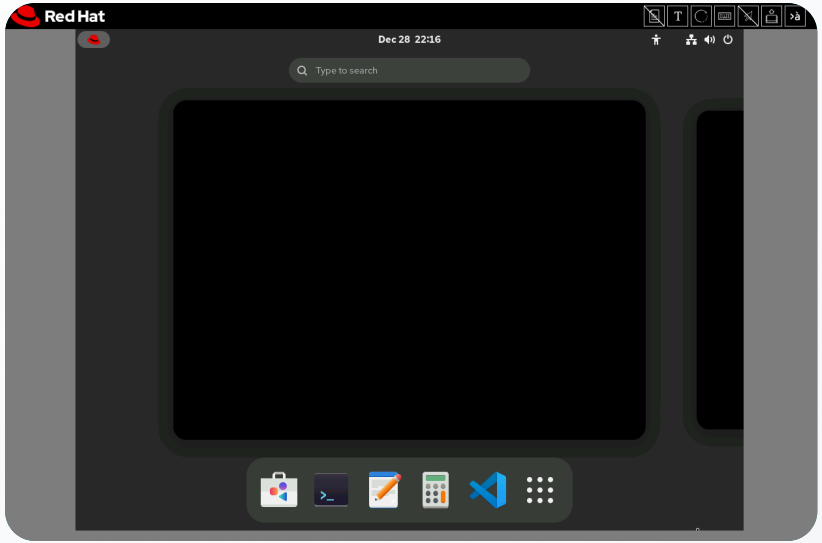
Terminology

Console, Terminal, Shell

Console

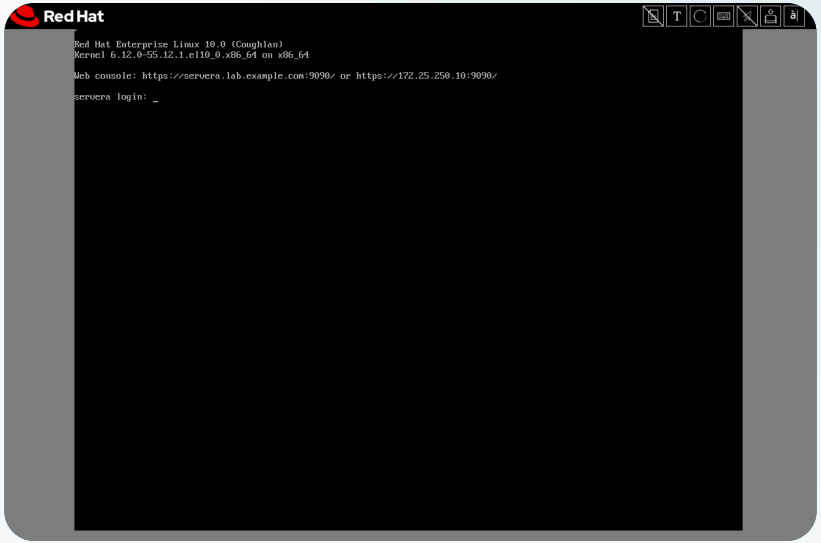
What you see when you look at the screen of a Linux server

Graphical console



`workstation`

Text console

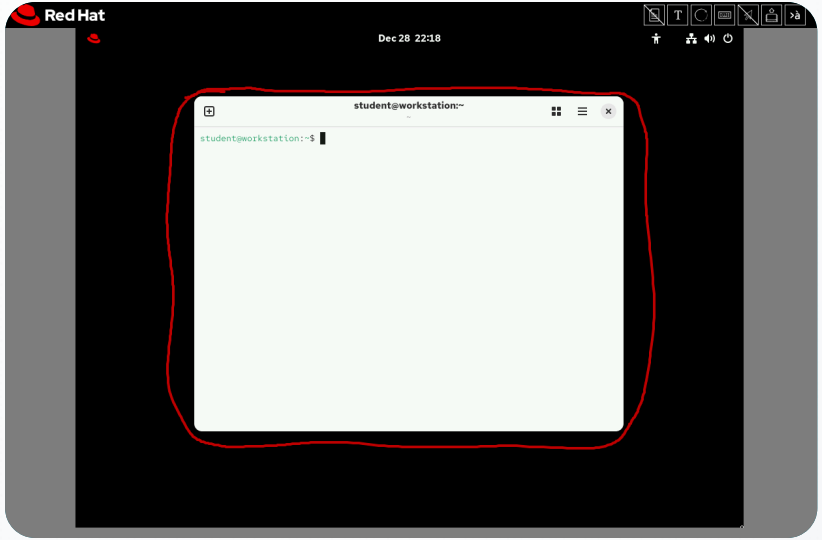


`servera`, `serverb`, `serverc`

Terminal

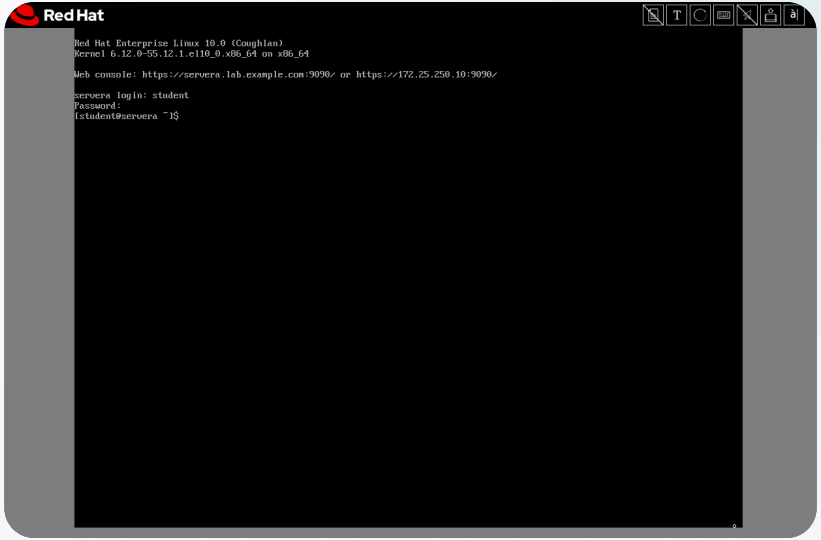
Opened on the console, provides access to shells

Terminal on a graphical console



The terminal is an application

Terminal on a text console

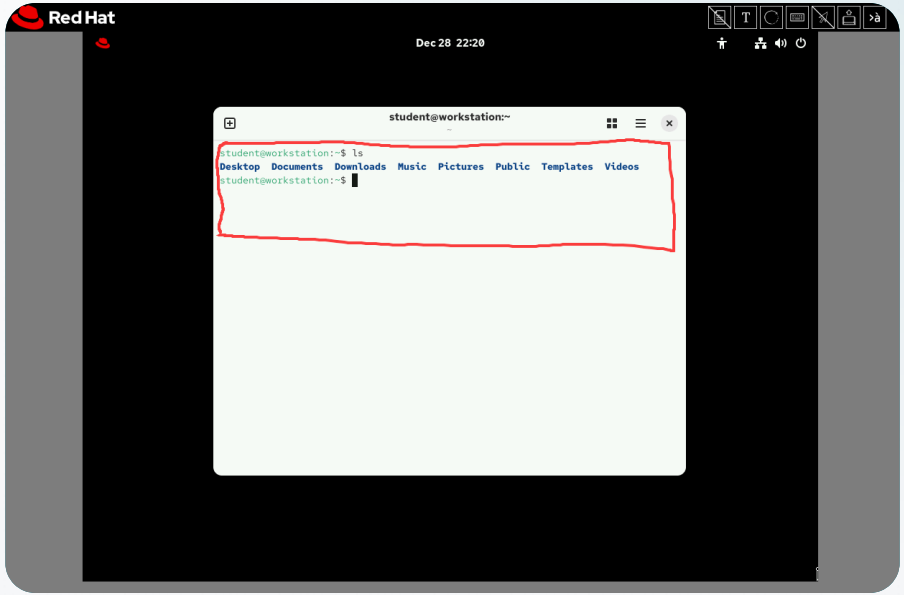


Terminal and console are basically the same.

Shell (Bash, Zsh, Fish…)

The interpreter running inside the terminal

- The shell interprets your commands
- Bash is the default shell on RHEL



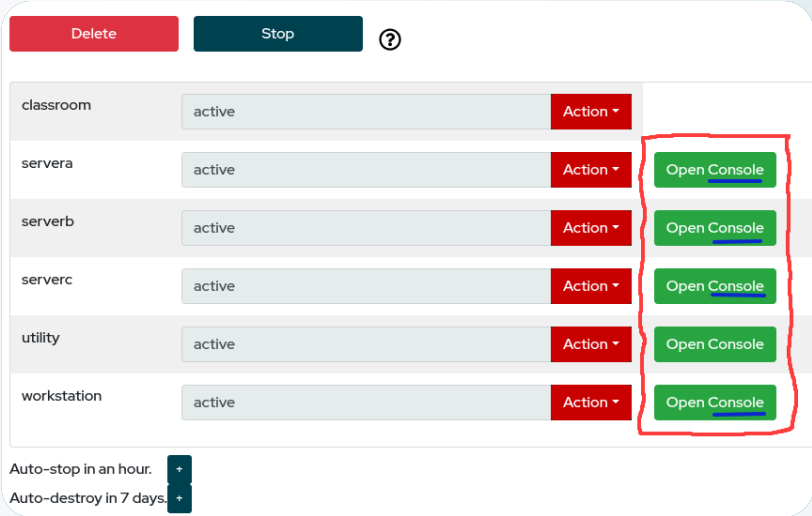
Opening RHA Lab Consoles

Click the `Open Console` button next to the machine you want to access.

The environment will autostop after an hour

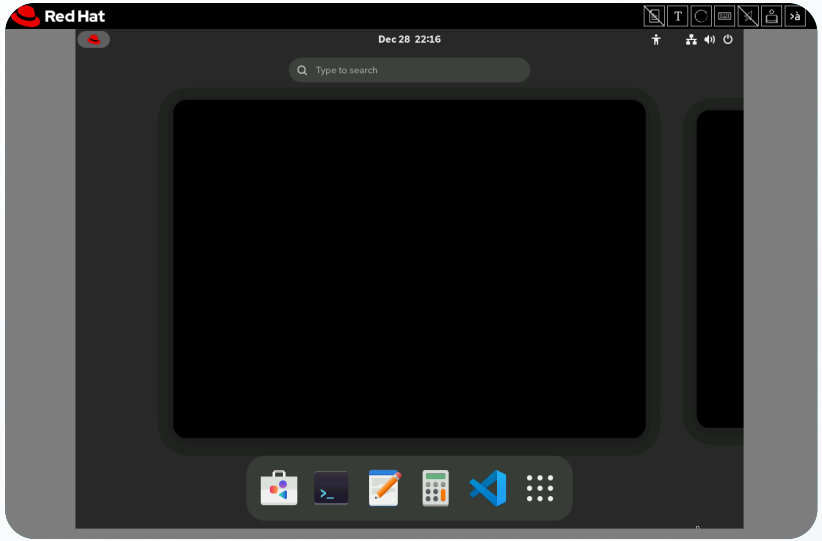
The environment will delete after 7 days

> **Note**
>
> Use the **+** button to extend, if needed.

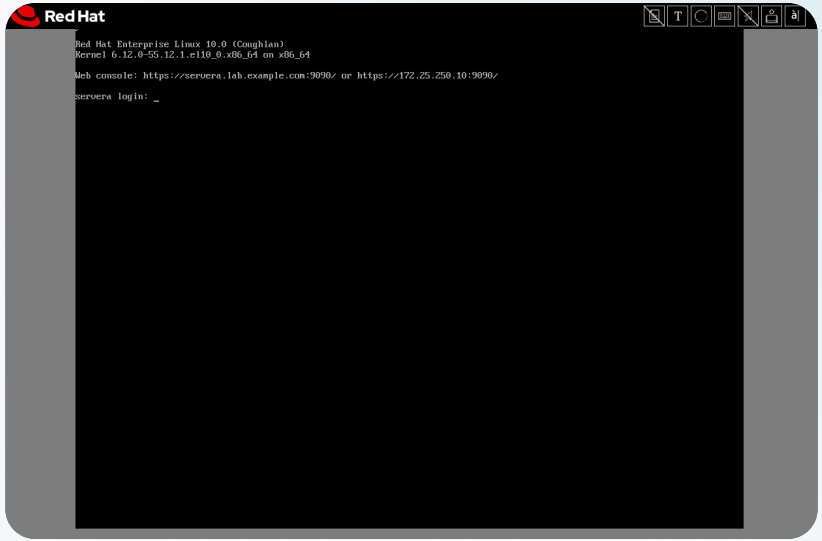


Console — RHA Lab

Graphical console

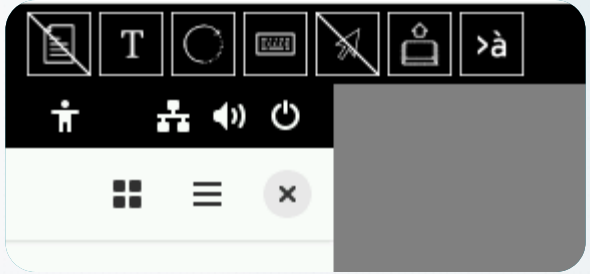


Text console



Console Controls

1. Enable Host Paste
   - Allows you to use Ctrl+V to paste text
2. Open Text Dialogue
   - Opens a text input you can paste text into
3. Ctrl+Alt+Delete
4. On-screen keyboard



Exercise: Accessing RHA Lab Environment

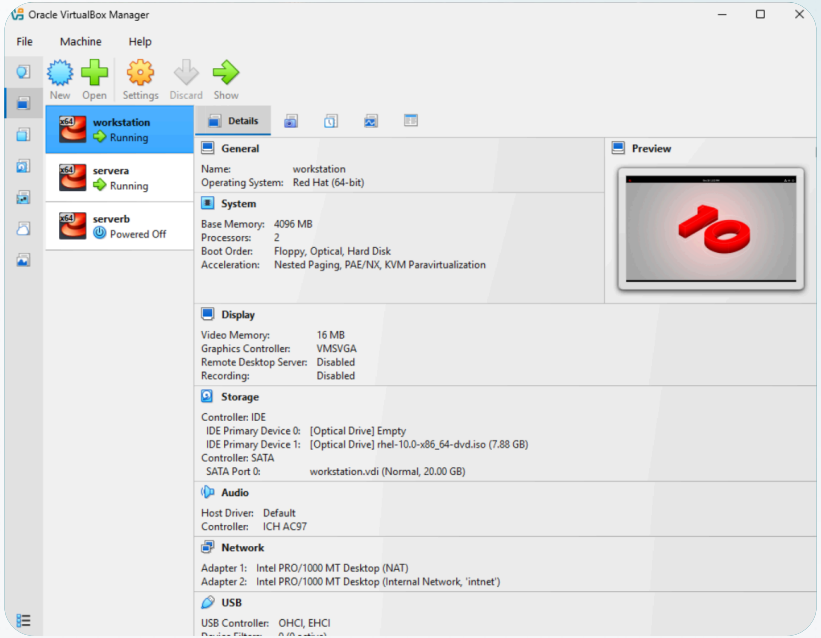
IT-230 · Linux Administration

Accessing the SCC Lab

SCC Lab

Hosted on SCC's Virtual Desktop Infrastructure (VDI)

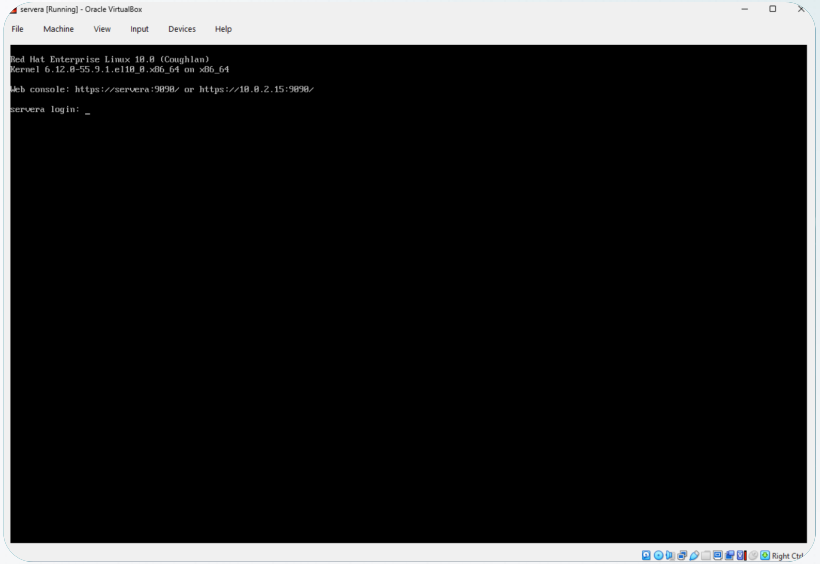
- Data is **NOT** persisted
- VM changes are **NOT** persisted
- Kicked if inactive
- Sessions held open for an hour
- After that, progress will be wiped



Console — SCC Lab

Graphical console

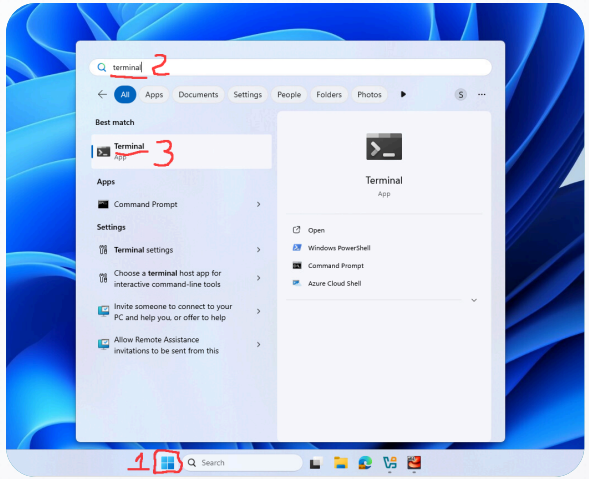
Text console

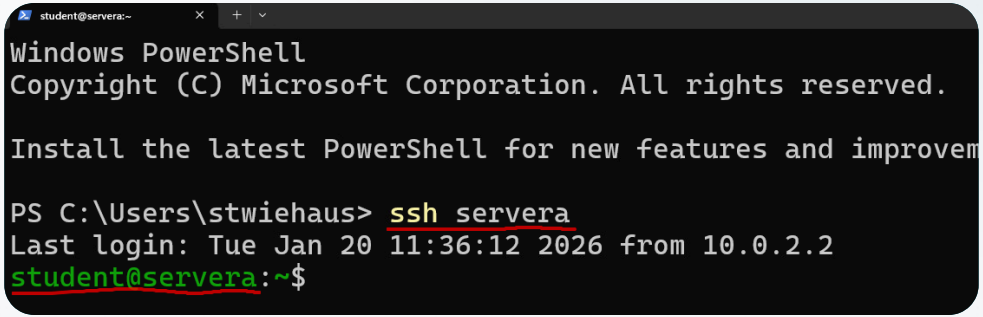


SSH — SCC Lab

SCC lab allows SSH from Windows Terminal in the VDI

1. Start the virtual machine(s) you want to access using VirtualBox.
2. Open the **Windows Terminal** application and type `ssh {VM_NAME}`





Exercise: Accessing the SCC Lab Environment

IT-230 · Linux Administration

Bash Prompt

The Bash Prompt is…

text displayed by the shell before you type a command

```
student@workstation:~$
```

Bash Prompt

```
student@workstation:/etc$ ls -l
```

The command you typed

Bash Prompt

student@workstation:/etc

```
1 student@workstation:/etc$ ls -l
```

Part	Meaning
`student`	The user you are logged in as
`workstation`	The host you are logged into
`/etc`	Your current working directory
`$`	Indicates a non-root user
`ls -l`	The command you typed

IT-230 · Linux Administration

Output Redirection

Redirecting and Appending

Output Redirection Operator >

Redirects Standard Output (stdout) to a file

student@workstation:~

```
student@workstation:~$ ls -l
-rw-r--r--. 1 student student 18 Aug 16 09:14 notes.txt
-rw-r--r--. 1 student student 27 Aug 16 09:14 servers.txt
student@workstation:~$ ls -l > long_listing.txt
student@workstation:~$ ls -l
-rw-r--r--. 1 student student 18 Aug 16 09:14 notes.txt
-rw-r--r--. 1 student student 27 Aug 16 09:14 servers.txt
-rw-r--r--. 1 student student 31 Aug 16 10:22 long_listing.txt
student@workstation:~$ cat long_listing.txt
-rw-r--r--. 1 student student 18 Aug 16 09:14 notes.txt
-rw-r--r--. 1 student student 27 Aug 16 09:14 servers.txt
student@workstation:~$
```

A New Way to Lose Your Work!

The Scenario

1. You spend all week writing `thesis.txt` (10,000 words).
2. You decide to add one final thought.
3. You type: `echo "The End" > thesis.txt`

The Result

Congratulations!

Your 10,000-word thesis has been successfully **overwritten**

Your file now **contains exactly two words:** `The End`.

What you meant to use was the append operator `>>`.

Appending with `>>`

Use `>>` to append output to the end of an existing file.

```
ls -l /tmp > listing.txt
ls -l /etc >> listing.txt
```

`listing.txt` contains the `/tmp` listing followed by the `/etc` listing.

Comparison

Using >

```
student@workstation:~$ echo "Hello Class" > greet1.txt
student@workstation:~$ echo "Hello Class" > greet1.txt
student@workstation:~$ echo "Hello Class" > greet1.txt
student@workstation:~$ cat greet1.txt
Hello Class
student@workstation:~$
```

Using >>

```
student@workstation:~$ echo "Hello Class" >> greet2.txt
student@workstation:~$ echo "Hello Class" >> greet2.txt
student@workstation:~$ echo "Hello Class" >> greet2.txt
student@workstation:~$ cat greet2.txt
Hello Class
Hello Class
Hello Class
student@workstation:~$
```

Redirect

What will this output to the screen?

student@workstation:inventories

```
student@workstation:inventories$ ls > inventories
student@workstation:inventories$ cat inventories
2022-11-inventory.csv
2022-12-inventory.csv
2023-01-inventory.csv
2023-02-inventory.csv
2023-03-inventory.csv
2023-04-inventory.csv
2023-05-inventory.csv
2023-06-inventory.csv
2023-07-inventory.csv
2023-08-inventory.csv
...
student@workstation:inventories$
```

The output is redirected to a file. Nothing is shown on screen.

The Null Device

What will this command output?

student@workstation:/inventories

```
1 student@workstation:/inventories$ ls > /dev/null
2 student@workstation:/inventories$
```

`/dev/null` is like a black hole.

No information can escape

The command produces nothing on the screen.

HANDS-ON EXERCISE

Exercise: Output Redirection vs Append

GOAL

Compare replacing a file's contents with appending new lines

ENVIRONMENT

Host: `servera`

WORKFLOW

1. Create a file in Vim and verify its contents
2. Redirect output with `>` and inspect what remains
3. Create a second file and add lines with `>>`
4. Verify that appending preserves the existing content

Exercise: Output Redirection vs Append

```
##################################################################################################
    Exercise: Output Redirection (>) vs Append (>>)
        Requirements
          Host: servera, logged in as student
          Start with no evil_plan.md or evil_plan-v2.md in your home directory
          Compare replacing a file with adding to its existing contents
        Steps
          1. Create and verify a file in Vim
          2. Observe how > overwrites the file
          3. Build a new file with >> and verify the appended lines
          4. Remove the exercise files
##################################################################################################
[student@servera ~]$
```

00:00

Written exercise

IT-230 · Linux Administration

Pipe Operator

Use output from one command as input to another command

grep - Global, Regular Expression, Print

Originally created by Ken Thompson in **1973**.

Used to **search text** and print matching lines.

student@workstation:~

```
student@workstation:~$ grep nologin /etc/passwd
bin:x:1:1:bin:/bin:/usr/sbin/nologin
daemon:x:2:2:daemon:/sbin:/usr/sbin/nologin
adm:x:3:4:adm:/var/adm:/usr/sbin/nologin
mail:x:8:12:mail:/var/spool/mail:/usr/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/usr/sbin/nologin
nobody:x:65534:65534:Kernel Overflow User:/:/usr/sbin/nologin
dbus:x:81:81:System message bus:/:/sbin/nologin
systemd-oom:x:997:996:systemd Userspace OOM Killer:/:/sbin/nologin
polkitd:x:114:114:User for polkitd:/:/sbin/nologin
sssd:x:996:995:User for sssd:/run/sssd/:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/usr/share/empty.sshd:/usr/sbin/nologin
chrony:x:995:994:chrony system user:/var/lib/chrony:/sbin/nologin
systemd-coredump:x:993:993:systemd Core Dumper:/:/usr/sbin/nologin
```

`wc` - Word Count

Prints line, word, and byte counts.

The `-l` option prints how many lines.

student@workstation:~

```
student@workstation:~$ wc -l /etc/passwd
55 /etc/passwd
```

Use `grep` and `wc` Together with a Pipe

student@workstation:~

```
1  student@workstation:~$ grep nologin /etc/passwd | wc -l
2  50
```

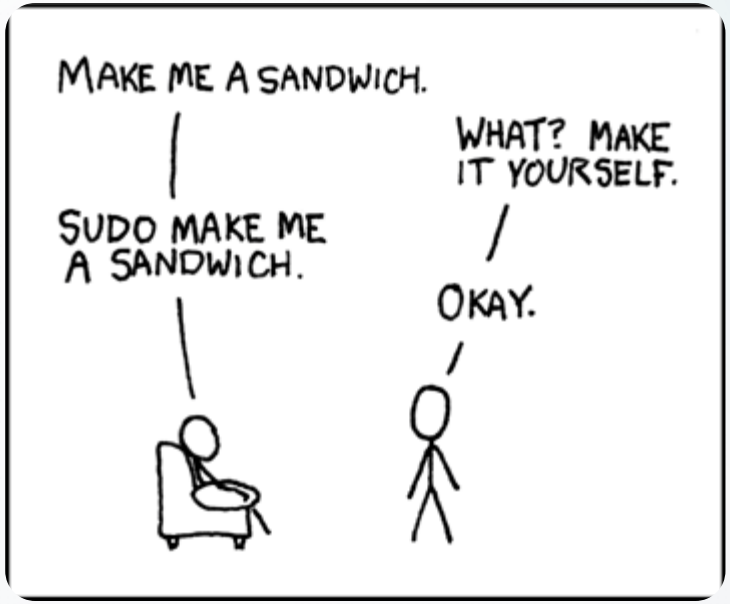
Pipe Operator

What will this command output?

student@workstation:inventories

```
student@workstation:inventories$ ls -l | grep 2023
total 0
-rw-r--r--. 1 sean sean 0 Aug  8 07:33 2023-01-inventory.csv
-rw-r--r--. 1 sean sean 0 Aug  8 07:33 2023-02-inventory.csv
-rw-r--r--. 1 sean sean 0 Aug  8 07:33 2023-03-inventory.csv
-rw-r--r--. 1 sean sean 0 Aug  8 07:33 2023-04-inventory.csv
-rw-r--r--. 1 sean sean 0 Aug  8 07:33 2023-05-inventory.csv
-rw-r--r--. 1 sean sean 0 Aug  8 07:33 2023-06-inventory.csv
-rw-r--r--. 1 sean sean 0 Aug  8 07:33 2023-07-inventory.csv
-rw-r--r--. 1 sean sean 0 Aug  8 07:33 2023-08-inventory.csv
-rw-r--r--. 1 sean sean 0 Aug  8 07:33 2023-09-inventory.csv
-rw-r--r--. 1 sean sean 0 Aug  8 07:33 2023-10-inventory.csv
-rw-r--r--. 1 sean sean 0 Aug  8 07:33 2023-11-inventory.csv
-rw-r--r--. 1 sean sean 0 Aug  8 07:33 2023-12-inventory.csv
```

Sudo



xkcd #149, "Sandwich" — used under CC BY-NC 2.5

`sudo` runs a command as a different user.

If no user is specified, it runs as the `root` user.

Accessing a Sensitive File as a Regular User

Logged in as the `student` user.

Try to access `/var/log/secure`

student@servera:~

```
student@servera:~$ ls -l /var/log/secure
-rw-------. 1 root root 1646 Jan  6 12:52 /var/log/secure
student@servera:~$ cat /var/log/secure
cat: /var/log/secure: Permission denied
student@servera:~$ sudo cat /var/log/secure
Jan  6 12:26:47 servera sshd-session[3269]: Accepted publickey for student from 172.25.250.1 port 45098 ssh2:...
Jan  6 12:26:47 servera (systemd)[3274]: pam_unix(systemd-user:session): session opened for user student(uid=1000)...
Jan  6 12:26:47 servera sshd-session[3269]: pam_unix(sshd:session): session opened for user student(uid=1000) by...
...
student@servera:~$
```

Without Using `sudo`

If I wanted to view the file without using `sudo`

student@workstation:~

```
student@servera:~$ su - root
root@servera:~# cat /var/log/secure
Jan  6 12:26:47 servera sshd-session[3269]: Accepted publickey for student from 172.25.250.1 port 45098 ssh2:...
Jan  6 12:26:47 servera (systemd)[3274]: pam_unix(systemd-user:session): session opened for user student(uid=1000)...
Jan  6 12:26:47 servera sshd-session[3269]: pam_unix(sshd:session): session opened for user student(uid=1000) by...
...
root@servera:~# exit
student@servera:~$
```

Open a Shell as the `root` User

Easily "log in" as a different user using `sudo -i`

Uses **YOUR** password.

student@workstation:~

```
student@workstation:~$ sudo -i
root@workstation:~#
```

No user was specified, so it opened a shell as the **root** user.

Open a Shell as a Specified User

To specify a different user, use the `-u` option.

Open an interactive login shell as the **bob** user.

student@workstation:~

```
student@workstation:~$ sudo -iu bob
bob@workstation:~$
```