St. Francis Institute of Technology, Mumbai-400 103

**Department Of Information Technology**

A.Y. 2023-2024

Class: TE-ITA/B, Semester: V

Subject: **DevOps Lab**

## Experiment – 1: To understand DevOps: Principles, practices and DevOps Engineer role & responsibilities and learn basic Linux commands

1. **Aim:** To prepare case study on DevOps and execute Linux commands
2. **Objectives:** After study of this experiment, the students will be able to
   - Understand the fundamentals of DevOps engineering and be fully proficient with DevOps terminologies, concepts, benefits and deployment options to meet business requirements
   - Demonstrate execution of Linux commands
3. **Outcomes:** After study of this experiment, the students will be able to
   - Understand the importance of DevOps thoroughly
   - Demonstrate the Linux commands
4. **Prerequisite:** Knowledge of software engineering and project management basic principles
5. **Requirements:** Personal Computer, Windows operating system, Ubuntu Operating system or Oracle virtual box or any online terminal, Internet Connection, Microsoft Word.
6. **Pre-Experiment Exercise:**
   **Brief Theory:** Refer shared material
7. **Laboratory Exercise A. Procedure:**
   **a.** Prepare case study based on following:
   - Introduction
   - Need
   - Usage
   - DevOps Lifecycle
   - Advantages and disadvantages

   **b. Enlist following Linux commands with their explanation and execution screenshots**
   1. sudo
   2. Apt -get
   3. Ls
   4. Cd
   5. Ped
   6. Cp
   7. Mv
   8. Rm
   9. Mkdir
   10. History
   11. Df
   12. Du
   13. Free
   14. Uname -a
   15. Top
   16. Man
   17. Info
   18. Passwd
   19. Whatis
   20. Date

21. W
22. Exit
23. Shutdown
24. Head
25. Tail
26. Echo
27. Grep
28. Zip
29. Unzip
30. <command name> -h or <command name> –help

8. **Post-Experiments Exercise A. Extended Theory:**

Nil

**B. Questions:**

- What is Linux and basic components of Linux?
- What are the shells used in Linux?
- What are the top 10 devops tools that are used in the industry today? ● How will you approach when a project needs to implement devops? **C. Conclusion:**
- Write what was performed in the experiment.
- Write the significance of the topic studied in the experiment. **D. References:**
- https://www.edureka.co/blog/interview-questions/linux-interview-questions-for-beginners/
- https://www.softwaretestinghelp.com/devops-interview-question11
- https://techlog360.com/basic-ubuntu-commands-terminal-shortcuts-linux-begin ner/
- https://tutorials.ubuntu.com/tutorial/command-line-for-beginners#0
- https://techlog360.com/basic-ubuntu-commands-terminal-shortcuts-linux-begin ner/
- https://www.edureka.co/blog/top-10-devops-tools/
- https://www.guru99.com/devops-tutorial.html

**Case Study: Implementing DevOps in a Software Development Company "TechSolutions Inc."**

**Introduction:**

In this case study, we will explore the implementation of DevOps practices in a software development company, "TechSolutions Inc." The company specializes in developing innovative software solutions for a diverse range of clients across industries. As TechSolutions Inc. continues to grow, they face challenges in terms of faster delivery, enhanced collaboration, and maintaining high-quality products. To address these challenges, they decided to adopt DevOps practices.

**Need:**

TechSolutions Inc. recognized the need to streamline their software development and deployment processes. Their traditional approach led to siloed teams, manual handoffs, and delays in delivering new features and updates. They needed a solution to improve collaboration between development and operations teams, automate repetitive tasks, and ensure a consistent and reliable deployment pipeline.

**Usage:**

To implement DevOps practices, TechSolutions Inc. focused on the following key areas:

1. **Continuous Integration (CI):** Developers integrated code changes into a shared repository multiple times a day. Automated tests were run to catch bugs early in the development cycle.
2. **Continuous Delivery (CD):** Automated deployment pipelines were set up to ensure that code changes passed through various testing stages and were ready for production deployment at any time.
3. **Infrastructure as Code (IaC):** Infrastructure provisioning and configuration were automated using tools like Terraform and Ansible, ensuring consistency and reducing manual errors.

4. **Monitoring and Feedback Loop:** Real-time monitoring tools were implemented to track application performance and user behavior, providing feedback for further improvements.

**DevOps Lifecycle:**

TechSolutions Inc. followed a well-defined DevOps lifecycle:

1. **Plan:**Clear project goals, requirements, and user stories were defined in collaboration with stakeholders, ensuring a shared understanding of the project scope.
2. **Code:** Developers wrote code in line with best practices, utilizing version control to maintain a single source of truth.
3. **Build:** Continuous Integration triggered automated builds and tests, allowing developers to catch and fix issues early in the process.
4. **Test:** Automated testing processes, including unit tests, integration tests, and user acceptance tests, were executed in a controlled environment.
5. **Deploy:** Continuous Delivery pipelines automatically deployed code changes to staging environments, ensuring consistent deployments.
6. **Operate:** The operations team monitored applications in production, addressing any issues promptly and making necessary updates.
7. **Feedback and Learn:** Continuous monitoring and user feedback allowed the team to iterate on the software, making improvements based on real-world usage.

**Advantages:**

The implementation of DevOps at TechSolutions Inc. resulted in several advantages:

1. **Faster Time to Market:**DevOps practices led to quicker development cycles, enabling the company to deliver new features and updates to clients faster.
2. **Improved Collaboration:** The collaboration between development and operations teams improved significantly, reducing communication barriers and fostering a culture of shared responsibility.
3. **Enhanced Quality:** Automated testing and deployment processes helped catch and prevent defects early, resulting in higher-quality software.
4. **Scalability:** Infrastructure automation allowed for seamless scaling of resources based on demand, ensuring optimal performance.

**Disadvantages:**

Despite the benefits, TechSolutions Inc. also faced some challenges:

1. **Initial Learning Curve:** Teams had to adapt to new tools and processes, which required time and effort to become proficient.
2. **Toolchain Complexity:** Managing a complex toolchain for automation and monitoring required ongoing maintenance and expertise.
3. **Resistance to Change:** Some team members were initially resistant to adopting DevOps practices, leading to a need for change management strategies.

**Conclusion:**

By embracing DevOps practices, TechSolutions Inc. successfully transformed their software development and deployment processes. The implementation led to improved collaboration, faster delivery, and higher-quality software. While challenges were encountered along the way, the benefits far outweighed the drawbacks, ultimately positioning the company for continued growth and success in the dynamic software development landscape.

**Output:**

1.**sudo:**sudo is one of the most common Linux commands that lets you perform tasks that require administrative or root permissions

```
onworks@onworks-Standard-PC-i440FX-PIIX-1996:~$ sudo
usage: sudo -h | -K | -k | -V
usage: sudo -v [-AknS] [-g group] [-h host] [-p prompt] [-u user]
usage: sudo -l [-AknS] [-g group] [-h host] [-p prompt] [-U user] [-u user]
            [command]
usage: sudo [-AbEHknPS] [-r role] [-t type] [-C num] [-g group] [-h host] [-p
            prompt] [-T timeout] [-u user] [VAR=value] [-i|-s] [<command>]
usage: sudo -e [-AknS] [-r role] [-t type] [-C num] [-g group] [-h host] [-p
            prompt] [-T timeout] [-u user] file ...
```

2.**Apt -get:**apt-get is a command line tool for handling Advanced Package Tool (APT) libraries in Linux. It lets you retrieve information and bundles from authenticated sources to manage, update, remove, and install software and its dependencies.

3.ls:The ls command lists files and directories within a system. Running it without a flag or parameter will show the current working directory's content.

```
onworks@onworks-Standard-PC-i440FX-PIIX-1996:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  snap  Templates  Videos
```

4.**cd:**To navigate through the Linux files and directories, use the cd command. Depending on your current working directory, it requires either the full path or the directory name.Running this command without an option will take you to the home folder. Keep in mind that only users with sudo privileges can execute it.

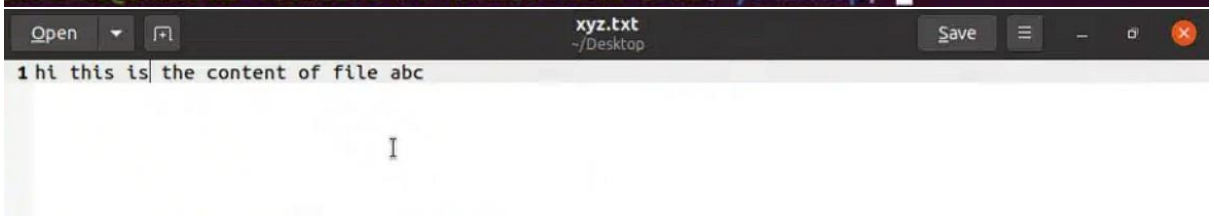
```
onworks@onworks-Standard-PC-i440FX-PIIX-1996:~$ cd Desktop
onworks@onworks-Standard-PC-i440FX-PIIX-1996:~/Desktop$
```

5.**Pwd:**Use the pwd command to find the path of your current working directory. Simply entering pwd will return the full current path – a path of all the directories that starts with a forward slash (/).

```
onworks@onworks-Standard-PC-i440FX-PIIX-1996:~/Desktop$ pwd
/home/onworks/Desktop
```

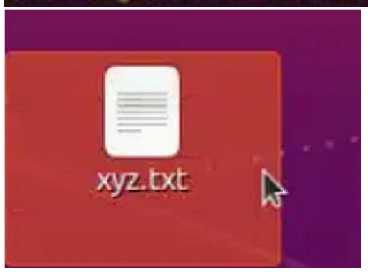
6.**cp:**Use the cp command to copy files or directories and their content.

```
onworks@onworks-Standard-PC-i440FX-PIIX-1996:~/Desktop$ cp abc.txt xyz.txt
onworks@onworks-Standard-PC-i440FX-PIIX-1996:~/Desktop$
```



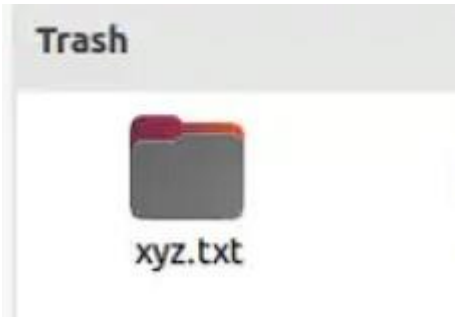

7.**mv:**The primary use of the mv command is to move and rename files and directories. Additionally, it doesn't produce an output upon execution.

```
onworks@onworks-Standard-PC-i440FX-PIIX-1996:~/Desktop$ mv abc.txt xyz.txt
onworks@onworks-Standard-PC-i440FX-PIIX-1996:~/Desktop$
```




8.**rm:**The rm command is used to delete files within a directory. Make sure that the user performing this command has write permissions.Remember the directory's location as this will remove the file(s) and you can't undo it.

```
onworks@onworks-Standard-PC-i440FX-PIIX-1996:~/Desktop$ rm xyz.txt
onworks@onworks-Standard-PC-i440FX-PIIX-1996:~/Desktop$
```




9.**mkdir:**Use the mkdir command to create one or multiple directories at once and set permissions for each of them. The user executing this command must have the privilege to make a new folder in the parent directory, or they may receive a permission denied error.

```
onworks@onworks-Standard-PC-i440FX-PIIX-1996:~/Desktop$ mkdir slavi
onworks@onworks-Standard-PC-i440FX-PIIX-1996:~/Desktop$ cd slavi
onworks@onworks-Standard-PC-i440FX-PIIX-1996:~/Desktop/slavi$
```

10.**history:**With history, the system will list up to 500 previously executed commands, allowing you to reuse them without re-entering. Keep in mind that only users with sudo privileges can execute this command.

```
onworks@onworks-Standard-PC-i440FX-PIIX-1996:~/Desktop/slavi$ history
    1  cd /etc
    2  cd default
    3  sudo vi grub
    4  sudo grub-upgrade
    5  sudo grub-update
    6  sudo update-grub
    7  sudo su -
    8  sudo
    9  Apt -get
   10  ls
   11  cd Desktop
   12  pwd
   13  cp abc.txt xyz.txt
   14  mv abc.txt xyz.txt
   15  mkdir slavi
   16  cd slavi
   17  history
```

11.**df:**Use the df command to report the system's disk space usage, shown in percentage and kilobyte (KB).

```
onworks@onworks-Standard-PC-i440FX-PIIX-1996:~/Desktop/slavi$ df
Filesystem     1K-blocks    Used Available Use% Mounted on
udev             1486440       0   1486440   0% /dev
tmpfs             302496    8648    293848   3% /run
/dev/sda1       29846488 4935792  23371524  18% /
tmpfs            1512464     172   1512292   1% /dev/shm
tmpfs               5120       4      5116   1% /run/lock
tmpfs            1512464       0   1512464   0% /sys/fs/cgroup
tmpfs             302496      68    302428   1% /run/user/1000
```

12.**du:**If you want to check how much space a file or a directory takes up, use the du command. You can run this command to identify which part of the system uses the storage excessively.

```
onworks@onworks-Standard-PC-i440FX-PIIX-1996:~/Desktop/slavi$ du
8       .
```

13.**free:**The free command gives information about used and unused memory usage and swap memory of a system. By default, it displays memory in kb (kilobytes).

```
onworks@onworks-Standard-PC-i440FX-PIIX-1996:~/Desktop/slavi$ free
              total        used        free      shared  buff/cache   available
Mem:        3024932      520788      435236       12688     2068908     2139344
Swap:        998396           0      998396
```

14.**uname -a:**The uname or unix name command will print detailed information about your Linux system and hardware. This includes the machine name, operating system, and kernel.

```
onworks@onworks-Standard-PC-i440FX-PIIX-1996:~/Desktop/slavi$ uname -a
Linux onworks-Standard-PC-i440FX-PIIX-1996 4.15.0-50-generic #54~16.04.1-Ubuntu
SMP Wed May 8 15:50:20 UTC 2019 i686 i686 i686 GNU/Linux
```

15.**top:**The top command in Linux Terminal will display all the running processes and a dynamic real-time view of the current system. It sums up the resource utilization, from CPU to memory usage.The top command can also help you identify and terminate a process that may use too many system resources.

```
top - 20:09:40 up 13 min,  1 user,  load average: 0,08, 0,17, 0,18
Tasks: 163 total,   1 running, 127 sleeping,   0 stopped,   0 zombie
%Cpu(s):  0,7 us,  0,2 sy,  0,0 ni, 99,0 id,  0,0 wa,  0,0 hi,  0,0 si,  0,2 st
KiB Mem :  3024932 total,   435056 free,   520700 used,  2069176 buff/cache
KiB Swap:   998396 total,   998396 free,        0 used.  2139380 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S  %CPU %MEM     TIME+ COMMAND
  812 root      20   0  228968  63968  29440 S   0,7  2,1   0:08.85 Xorg
 1601 onworks   20   0  425376 212288  75904 S   0,7  7,0   0:25.47 compiz
 6883 onworks   20   0  120492  31528  26996 S   0,7  1,0   0:00.63 gnome-term+
    1 root      20   0   25416   5396   3756 S   0,0  0,2   0:02.38 systemd
    2 root      20   0       0      0      0 S   0,0  0,0   0:00.00 kthreadd
    4 root       0 -20       0      0      0 I   0,0  0,0   0:00.00 kworker/0:+
    5 root      20   0       0      0      0 I   0,0  0,0   0:00.51 kworker/u4+
    6 root       0 -20       0      0      0 I   0,0  0,0   0:00.00 mm_percpu_+
    7 root      20   0       0      0      0 S   0,0  0,0   0:00.08 ksoftirqd/0
    8 root      20   0       0      0      0 I   0,0  0,0   0:00.11 rcu_sched
    9 root      20   0       0      0      0 I   0,0  0,0   0:00.00 rcu_bh
   10 root      rt   0       0      0      0 S   0,0  0,0   0:00.01 migration/0
   11 root      rt   0       0      0      0 S   0,0  0,0   0:00.00 watchdog/0
   12 root      20   0       0      0      0 S   0,0  0,0   0:00.00 cpuhp/0
   13 root      20   0       0      0      0 S   0,0  0,0   0:00.00 cpuhp/1
   14 root      rt   0       0      0      0 S   0,0  0,0   0:00.00 watchdog/1
   15 root      rt   0       0      0      0 S   0,0  0,0   0:00.01 migration/1
```

16.**man:**The man command provides a user manual of any commands or utilities you can run in Terminal, including the name, description, and options.

```
Hi this is the content of file abc

Manual page xyz.txt line 1/2 (END) (press h for help or q to quit)
```

17.**Info:**info command reads documentation in the info format. It will give detailed information for a command when compared with the man page. The pages are made using the texinfo tools because of which it can link with other pages, create menus and easy navigation.

```
File: dir,      Node: Top,      This is the top of the INFO tree.

This is the Info main menu (aka directory node).
A few useful Info commands:

  'q' quits;
  '?' lists all Info commands;
  'h' starts the Info tutorial;
  'mTexinfo RET' visits the Texinfo manual, etc.

* Menu:

Basics
* Common options: (coreutils)Common options.
* Coreutils: (coreutils).       Core GNU (file, text, shell) utilities.
* Date input formats: (coreutils)Date input formats.
* Ed: (ed).                     The GNU line editor
* File permissions: (coreutils)File permissions.
                                Access modes.
* Finding files: (find).        Operating on files matching certain criteria.

C++ libraries
-----Info: (dir)Top, 254 lines --Top-----------------------------------------
Welcome to Info version 6.1.  Type H for help, h for tutorial.
```

18.**passwd:**passwd command in Linux is used to change the user account passwords. The root user reserves the privilege to change the password for any user on the system, while a normal user can only change the account password for his or her own account.

```
onworks@onworks-Standard-PC-i440FX-PIIX-1996:~/Desktop/slavi$ passwd
Changing password for onworks.
(current) UNIX password:
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
```

19.**whatis:**The whatis command is used to get brief information about Linux commands or functions. It displays the manual page description in a single line of the command that passes with the whatis command. It searches for the strings that have been passed with it from its index databases.

```
onworks@onworks-Standard-PC-i440FX-PIIX-1996:~/Desktop/slavi$ whatis ls
ls (1)               - list directory contents
```

20.**date:**The date command is one of the most basic commands in Linux. To display the current date and time, simply type "date" at the command prompt and press enter.

```
onworks@onworks-Standard-PC-i440FX-PIIX-1996:~/Desktop/slavi$ date
Do 10. Aug 20:12:33 CEST 2023
```

21.**w:**w displays information about the users currently on the machine, and their processes.

```
onworks@onworks-Standard-PC-i440FX-PIIX-1996:~/Desktop/slavi$ w
 20:12:44 up 16 min,  1 user,  load average: 0,00, 0,09, 0,14
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
onworks  tty7     :0               30Aug19 1441days 10.58s  0.15s /sbin/upstart
-
```

22.**exit:**exit command in linux is used to exit the shell where it is currently running.

23.**shutdown:**This will perform a system shutdown in a proper way. You can also specify a timer (in seconds), instead of the word "now", for example: shutdown -h -t 30. This will bring the computer down in 30 seconds.

24.**head:**The head command allows you to view the first ten lines of a text. Adding an option lets you change the number of lines shown.

```
onworks@onworks-Standard-PC-i440FX-PIIX-1996:~/Desktop/slavi$ cat xyz.txt
Hi this is the content of file abc
```

25.**tail:**The tail command displays the last ten lines of a file. It allows users to check whether a file has new data or to read error messages.

```
onworks@onworks-Standard-PC-i440FX-PIIX-1996:~/Desktop/slavi$ tail xyz.txt
Hi this is the content of file abc
```

26.**echo:**The echo command is a built-in utility that displays a line of text or string using the standard output.

```
onworks@onworks-Standard-PC-i440FX-PIIX-1996:~/Desktop/slavi$ echo Hello World!
Hello World!
```

27.**grep:**The grep filter searches a file for a particular pattern of characters, and displays all lines that contain that pattern. The pattern that is searched in the file is referred to as the regular expression (grep stands for global search for regular expression and print out).

```
onworks@onworks-Standard-PC-i440FX-PIIX-1996:~/Desktop/slavi$ grep -i "is" xyz.t
xt
Hi this is the content of file abc
```

28.**zip:**Use the zip command to compress your files into a ZIP file, a universal format commonly used on Linux. It can automatically choose the best compression ratio.

```
onworks@onworks-Standard-PC-i440FX-PIIX-1996:~/Desktop/slavi$ zip terror xyz.txt
  adding: xyz.txt (stored 0%)
```

29.**Unzip:**So, to unzip a file called archive.zip in the current directory, enter: unzip archive.zip

```
onworks@onworks-Standard-PC-i440FX-PIIX-1996:~/Desktop/slavi$ unzip terror xyz.t
xt
Archive:  terror.zip
replace xyz.txt? [y]es, [n]o, [A]ll, [N]one, [r]ename: y
 extracting: xyz.txt
```

30.**df -h:**(df -h) will show the file system disk space statistics in “human-readable” format, means it gives the details in bytes, megabytes, and gigabyte.

```
onworks@onworks-Standard-PC-i440FX-PIIX-1996:~/Desktop/slavi$ df -h
Filesystem      Size  Used Avail Use% Mounted on
udev            1,5G     0  1,5G   0% /dev
tmpfs           296M  8,5M  287M   3% /run
/dev/sda1        29G  4,8G   23G  18% /
tmpfs           1,5G  172K  1,5G   1% /dev/shm
tmpfs           5,0M  4,0K  5,0M   1% /run/lock
tmpfs           1,5G     0  1,5G   0% /sys/fs/cgroup
tmpfs           296M   64K  296M   1% /run/user/1000
```