



Hardening and Securing Linux Services – Unit 3



What is Linux OS

- just like Windows, iOS, and Mac OS, Linux is an operating system(OS). In fact, one of the most popular platforms on the planet, Android, is powered by the Linux operating system.
- the operating system manages the communication between your software and your hardware.

Reference : <https://www.linux.com/what-is-linux/>



Features of Linux OS

- <https://www.javatpoint.com/linux-features>
- <https://www.tecmint.com/list-all-running-services-under-systemd-in-linux/>



Starting services at boot time

How To Check If a Service is Running on Linux

```
sudo systemctl status apache2
```

How to Restart a Service ?

```
sudo systemctl restart SERVICE_NAME
```

```
sudo systemctl restart apache2
```



Starting services at boot time

How to Reload a Service

```
sudo systemctl reload SERVICE_NAME
```

```
sudo systemctl reload apache2
```

How to start a Service ?

```
sudo systemctl start SERVICE_NAME
```

```
sudo systemctl start apache2
```



Starting services at boot time

How to Enable the Service at Boot

```
sudo systemctl enable SERVICE_NAME
```

```
sudo systemctl enable apache2
```

To list only the enabled services at system boot, run:

```
$ sudo systemctl list-unit-files --type=service --  
state=enabled -all
```

OR

```
$ sudo service --status-all
```



Package control in Linux

[**https://packagecontrol.io/browse/popular**](https://packagecontrol.io/browse/popular)

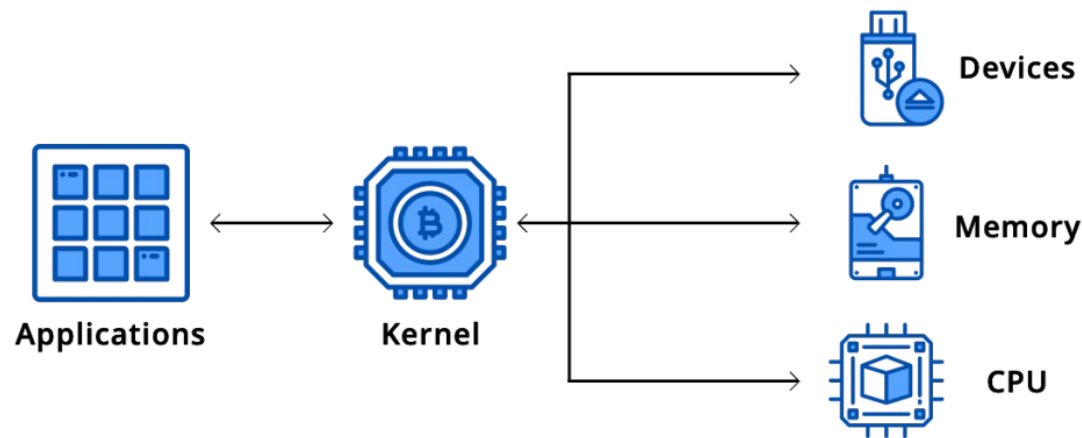
[**https://granneman.com/webdev/editors/sublime-text/packages/how-to-install-and-use-package-control**](https://granneman.com/webdev/editors/sublime-text/packages/how-to-install-and-use-package-control)

[**https://www.atectown.com/install-package-control-in-sublime/**](https://www.atectown.com/install-package-control-in-sublime/)

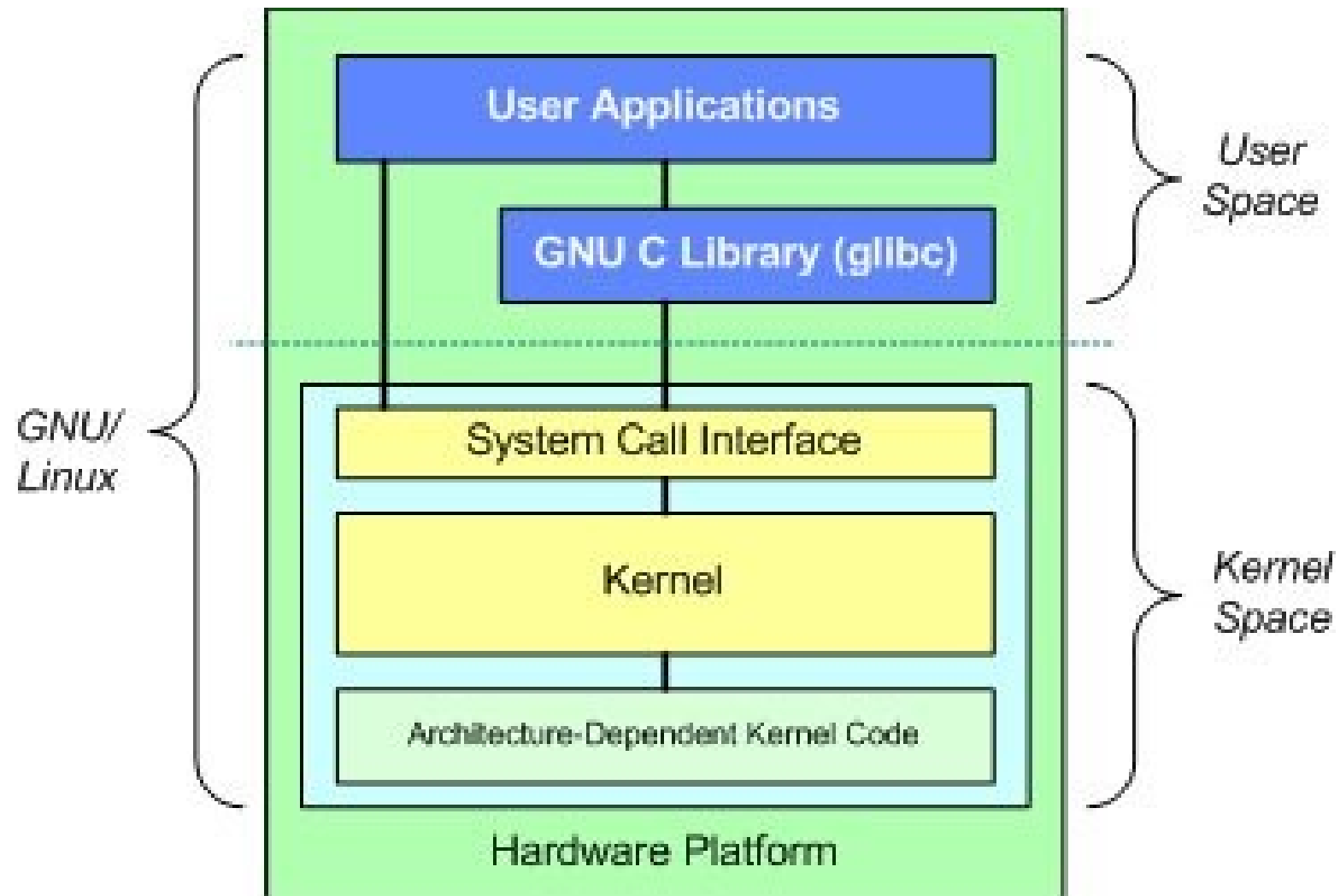
Kernel Security

The kernel - is a major part of the operating system that made it possible to run various processes simultaneously and does not end up crashing the system due to insufficient resources.

How kernel works ?



Kernal Security





How to build Linux kernal

<http://faculty.winthrop.edu/domanm/csci411/Handouts/LinuxKernelIntroduction.pdf>

<https://phoenixnap.com/kb/build-linux-kernel>

<https://www.c-sharpcorner.com/article/create-your-own-kernel/>



What is an open source kernel?

Linux maintains a comprehensive archive on its kernel. Apple has published the kernel types for all of its operating systems for open source access. Microsoft also uses a Linux kernel for the Windows subsystem for Linux.

functions of the kernel

1. Memory management: Regulates how much memory is used in different places.
2. Process management: Determines which processes the CPU can use, as well as when and how long they're used for.
3. Device driver: Intermediates between hardware and processes.
4. System calls and security: Receives service requests from the processes.

When implemented properly, the functions of the kernel are invisible to users. The kernel works in its own setting, the kernel space. On the other hand, files, programs, games, browsers, and everything that the user sees are located in the user space. Interaction between these two use the system call interface (SCI).

Structure of the kernel

A kernel is always built the same way and consists of several layers:

- The deepest layer is the **interface with hardware** (processors, memory, and devices), which manages network controllers and PCI express controllers, for example.
- On top of that is the **memory management**, which entails distributing RAM including the virtual main memory.
- Then comes **process management** (scheduler), which is responsible for time management and makes multitasking possible.
- The next layer contains **device management**.
- The highest layer is the **file system**. That's where processes are assigned to RAM or the hard drive.



Types of kernels

There are, of course, different ways to build a kernel and architectural considerations when building one from scratch. In general, most kernels fall into one of three types: monolithic, microkernel, and hybrid. Linux is a monolithic kernel while OS X (XNU) and Windows 7 use hybrid kernels.

Microkernel

A microkernel takes the approach of only managing what it has to: CPU, memory, and IPC.



Kernal Security

Microkernels - have a advantage of portability because they don't have to worry if you change your video card or even your operating system so long as the operating system still tries to access the hardware in the same way.

Microkernels also have a very small footprint, for both memory and install space, and they tend to be more secure because only specific processes run in user mode

Kernal Security

Pros

- Portability
- Small install footprint
- Small memory footprint
- Security

Cons

- Hardware is more abstracted through drivers
- Hardware may react slower because drivers are in user mode
- Processes have to wait in a queue to get information
- Processes can't get access to other processes without waiting



Kernal Security

Monolithic Kernel

Monolithic kernels are the opposite of microkernels because they encompass not only the CPU, memory, and IPC, but they also include things like device drivers, file system management, and system server calls.

Monolithic kernels tend to be better at accessing hardware and multitasking because if a program needs to get information from memory or another process running it has a more direct line to access it and doesn't have to wait in a queue to get things done.

This however can cause problems because the more things that run in supervisor mode, the more things that can bring down your system if one doesn't behave properly.

Kernal Security

Pros

- More direct access to hardware for programs
- Easier for processes to communicate between eachother
- If your device is supported, it should work with no additional installations
- Processes react faster because there isn't a queue for processor time

Cons

- Large install footprint
- Large memory footprint
- Less secure because everything runs in supervisor mode



Kernal Security

Hybrid Kernel

Hybrid kernels have the ability to pick and choose what they want to run in user mode and what they want to run in supervisor mode. Often times things like device drivers and filesystem I/O will be run in user mode while IPC and server calls will be kept in the supervisor mode.

This give the best of both worlds but often will require more work of the hardware manufacturer because all of the driver responsibility is up to them.

It also can have some of the latency problems that is inherent with microkernels.

Kernal Security

Pros

- Developer can pick and choose what runs in user mode and what runs in supervisor mode
- Smaller install footprint than monolithic kernel
- More flexible than other models

Cons

- Can suffer from same process lag as microkernel
- Device drivers need to be managed by user (typically)

Port control and port restriction

What is a port?

A port is a virtual point where network connections start and end. Ports are software-based and managed by a computer's operating system. Each port is associated with a specific process or service. Ports allow computers to easily differentiate between different kinds of traffic: emails go to a different port than webpages, for instance, even though both reach a computer over the same Internet connection.

Types of ports

Physical : <https://winstartechologies.com/types-of-computer-ports/>

Logical: <https://ipcisico.com/lesson/network-ports/>



Port control and port restriction

How to control/restrict Ports ?

<https://study-ccna.com/port-security/>

<https://www.computernetworkingnotes.com/ccna-study-guide/switchport-port-security-explained-with-examples.html>



Configuring and monitoring logs

What are Logs ?

A log file is a file that records either events that occur in an operating system or other software runs, or messages between different users of a communication software. Logging is the act of keeping a log.

Configuring and monitoring logs

Where the logs are generated ?

The screenshot displays the WordPress dashboard for 'Business Bloomer DEV'. The left sidebar shows the 'WooCommerce' menu item. The main content area shows the 'WooCommerce / Status / Logs' breadcrumb trail. Below this, there are tabs for 'System status', 'Tools', 'Logs', and 'Scheduled Actions'. The 'Logs' tab is active, showing a list of log entries. One entry is highlighted with a red box: 'wc-image-regeneration-2020-09-01-db5a3cea3ede752d60036078eec57591.log'. A red arrow points from this entry to the log content below. The log content shows a series of 'INFO Regenerating images for attachment ID: [ID]' messages.

WooCommerce / Status / Logs

System status Tools Logs Scheduled Actions

wc-image-regeneration-2020-09-01-db5a3cea3ede752d60036078eec57591.log Delete log

wc-image-regeneration-2020-09-01-db5a3cea3ede752d60036078eec57591.log View

2020-09-01T16:32:21+00:00 INFO Regenerating images for attachment ID: 186167
2020-09-01T16:32:21+00:00 INFO Regenerating images for attachment ID: 186046
2020-09-01T16:32:22+00:00 INFO Regenerating images for attachment ID: 184664
2020-09-01T16:32:22+00:00 INFO Regenerating images for attachment ID: 184655
2020-09-01T16:32:24+00:00 INFO Regenerating images for attachment ID: 183483
2020-09-01T16:32:24+00:00 INFO Regenerating images for attachment ID: 183482
2020-09-01T16:32:25+00:00 INFO Regenerating images for attachment ID: 183480
2020-09-01T16:32:26+00:00 INFO Regenerating images for attachment ID: 183479
2020-09-01T16:32:26+00:00 INFO Regenerating images for attachment ID: 183442
2020-09-01T16:32:27+00:00 INFO Regenerating images for attachment ID: 183438
2020-09-01T16:32:27+00:00 INFO Regenerating images for attachment ID: 183148
2020-09-01T16:32:27+00:00 INFO Regenerating images for attachment ID: 182925
2020-09-01T16:32:28+00:00 INFO Regenerating images for attachment ID: 182924
2020-09-01T16:32:28+00:00 INFO Regenerating images for attachment ID: 182508
2020-09-01T16:32:28+00:00 INFO Regenerating images for attachment ID: 182507
2020-09-01T16:32:29+00:00 INFO Regenerating images for attachment ID: 182506

Configuring and monitoring logs

```
/**
 * @snippet      Log Failed Orders @ WC Status
 * @how-to       Get CustomizeWoo.com FREE
 * @author       Rodolfo Melogli
 * @testedwith   WooCommerce 4.4
 * @donate $9    https://businessbloomer.com/bloomer-armada/
 */

add_action( 'woocommerce_before_thankyou', 'bbloomer_log_failed_orders_wc_status' );

function bbloomer_log_failed_orders_wc_status( $order_id ) {

    // GET ORDER FROM ORDER ID @ THANK YOU PAGE
    $order = wc_get_order( $order_id );

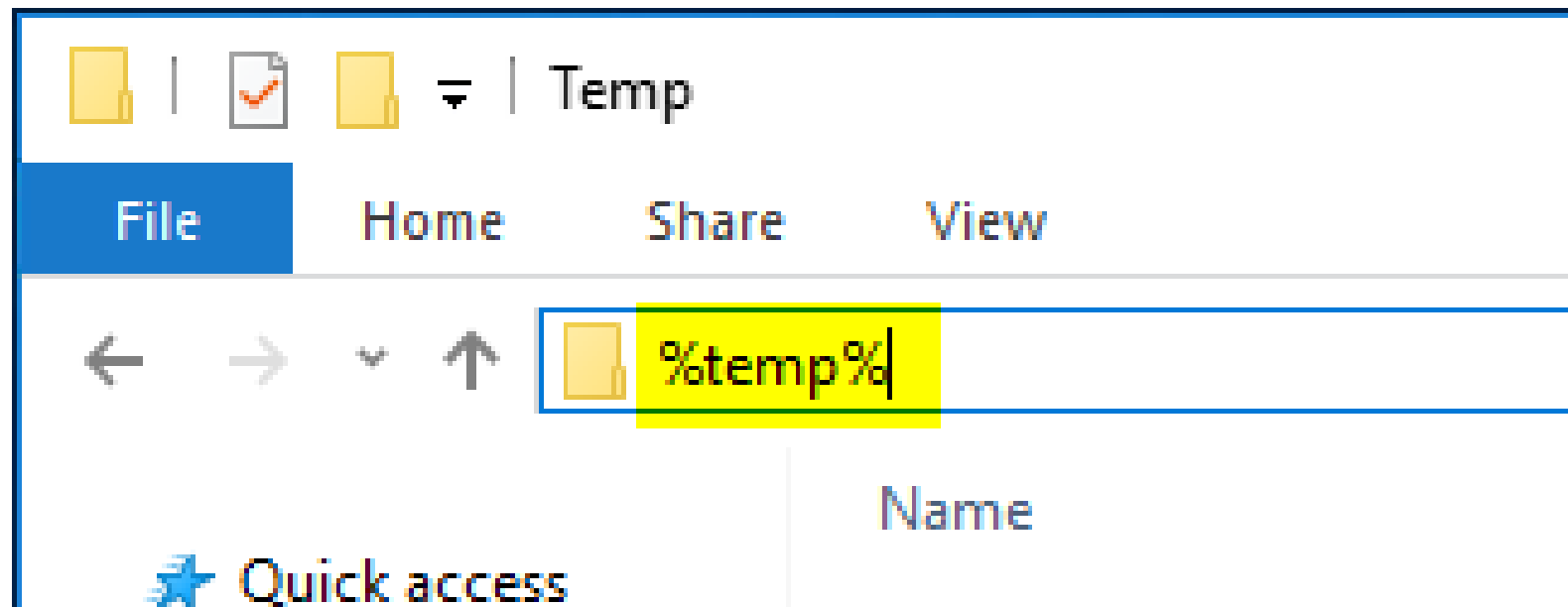
    // EXIT IF ORDER HAS NOT FAILED
    if ( ! $order->has_status( 'failed' ) ) return;

    // LOAD THE WC LOGGER
    $logger = wc_get_logger();

    // LOG THE FAILED ORDER TO CUSTOM "failed-orders" LOG
    $logger->info( wc_print_r( $order, true ), array( 'source' => 'failed-orders' ) );

}
```

Configuring and monitoring logs



Reference: <https://www.loggly.com/ultimate-guide/windows-logging-basics/>

parsing and filtering logs with grep, awk and sed

grep = global regular expression print

grep (global regular expression print) will search input files for a search string, and print the lines that match it. Beginning at the first line in the file, grep copies a line into a buffer, compares it against the search string, and if the comparison passes, prints the line to the screen. Grep will repeat this process until the file runs out of lines. Notice that nowhere in this process does grep store lines, change lines, or search only a part of a line.

parsing and filtering logs with grep, awk and sed

grep eg.

Please cut & paste the following data and save to a file called 'a_file':

```
boot  
book  
booze  
machine  
boots  
bungie  
bark  
aardvark  
broken$stuff  
robots
```

parsing and filtering logs with grep, awk and sed

grep eg.

```
grep "boo" a_file
```

In this example, grep would loop through every line of the file "a_file" and print out every line that contains the word 'boo':

```
boot  
book  
booze  
boots
```

Reference : <https://geekflare.com/grep-command-examples/>

parsing and filtering logs with grep, awk and sed

AWK

A text pattern scanning and processing language, created by Aho, Weinberger & Kernighan (hence the name).

awk is not just a command. It is a scripting language that can be used from both terminal and awk file. It supports the variable, conditional statement, array, loops etc. like other scripting languages. It can read any file content line by line and separate the fields or columns based on a specific delimiter. It also supports regular expression for searching particular string in the text content or file and takes actions if any match founds.

Ex: https://linuxhint.com/20_awk_examples/

parsing and filtering logs with grep, awk and sed

AWK (eg)

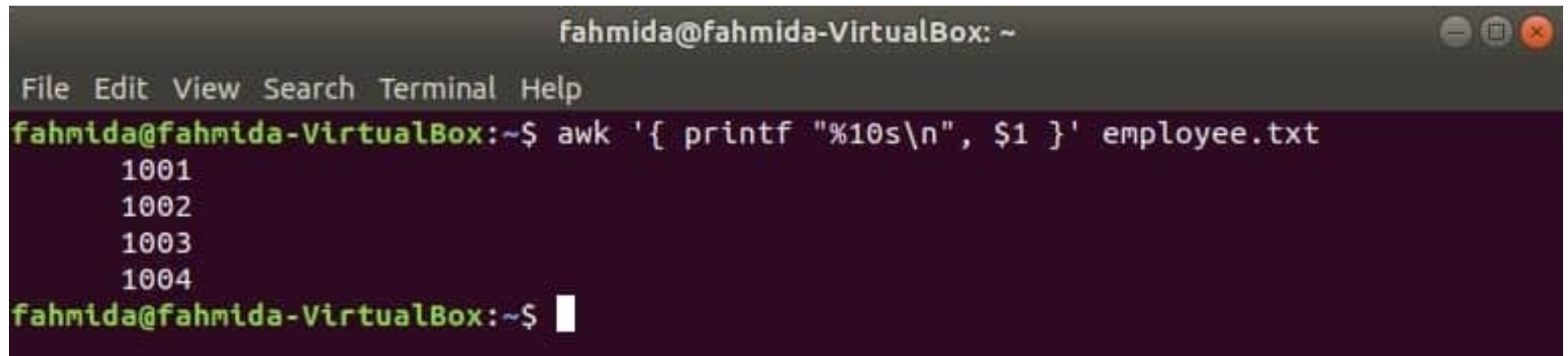
employee.txt

1001 John sena 40000

1002 Jafar Iqbal 60000

1003 Meher Nigar 30000

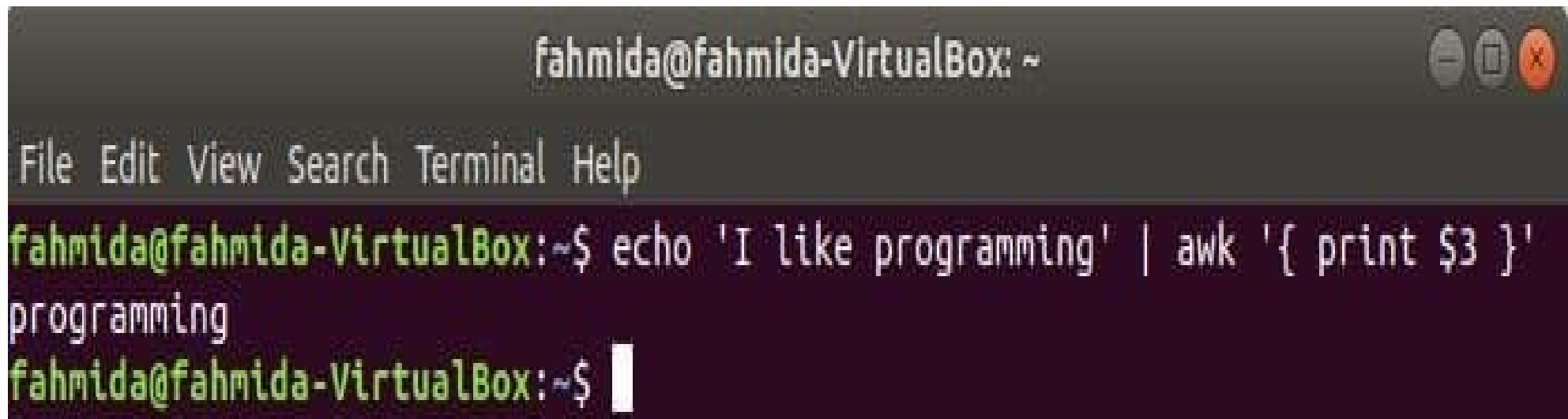
1004 Jonny Liver 70000

A terminal window titled 'fahmida@fahmida-VirtualBox: ~' with a menu bar (File, Edit, View, Search, Terminal, Help). The prompt is 'fahmida@fahmida-VirtualBox:~\$'. The command 'awk '{ printf "%10s\n", \$1 }' employee.txt' is entered. The output shows the first column of the file: '1001', '1002', '1003', and '1004'. The prompt is now 'fahmida@fahmida-VirtualBox:~\$' with a cursor.

```
fahmida@fahmida-VirtualBox: ~  
File Edit View Search Terminal Help  
fahmida@fahmida-VirtualBox:~$ awk '{ printf "%10s\n", $1 }' employee.txt  
1001  
1002  
1003  
1004  
fahmida@fahmida-VirtualBox:~$
```

parsing and filtering logs with grep, awk and sed

AWK (eg)

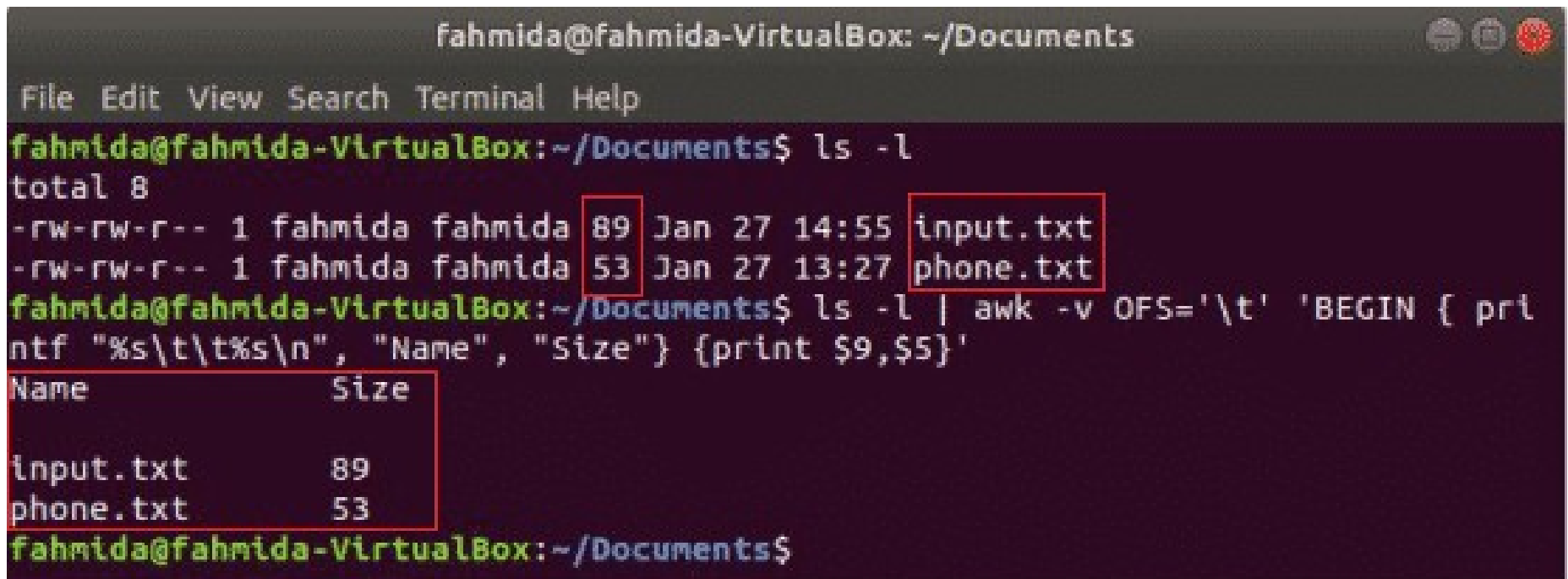
A terminal window titled 'fahmida@fahmida-VirtualBox: ~' with standard window controls. The menu bar includes 'File', 'Edit', 'View', 'Search', 'Terminal', and 'Help'. The terminal shows a command being executed: 'fahmida@fahmida-VirtualBox:~\$ echo 'I like programming' | awk '{ print \$3 }''. The output of the command is 'programming', which appears on the line immediately following the command. The prompt 'fahmida@fahmida-VirtualBox:~\$' is shown again on the next line with a cursor.

```
fahmida@fahmida-VirtualBox: ~  
File Edit View Search Terminal Help  
fahmida@fahmida-VirtualBox:~$ echo 'I like programming' | awk '{ print $3 }'  
programming  
fahmida@fahmida-VirtualBox:~$
```


parsing and filtering logs with grep, awk and sed

AWK (eg)

```
$ ls -l  
$ ls -l | awk -v OFS='\t' 'BEGIN { printf "%s\t%s\n", "Name", "Size"} {print $9,$5}'
```



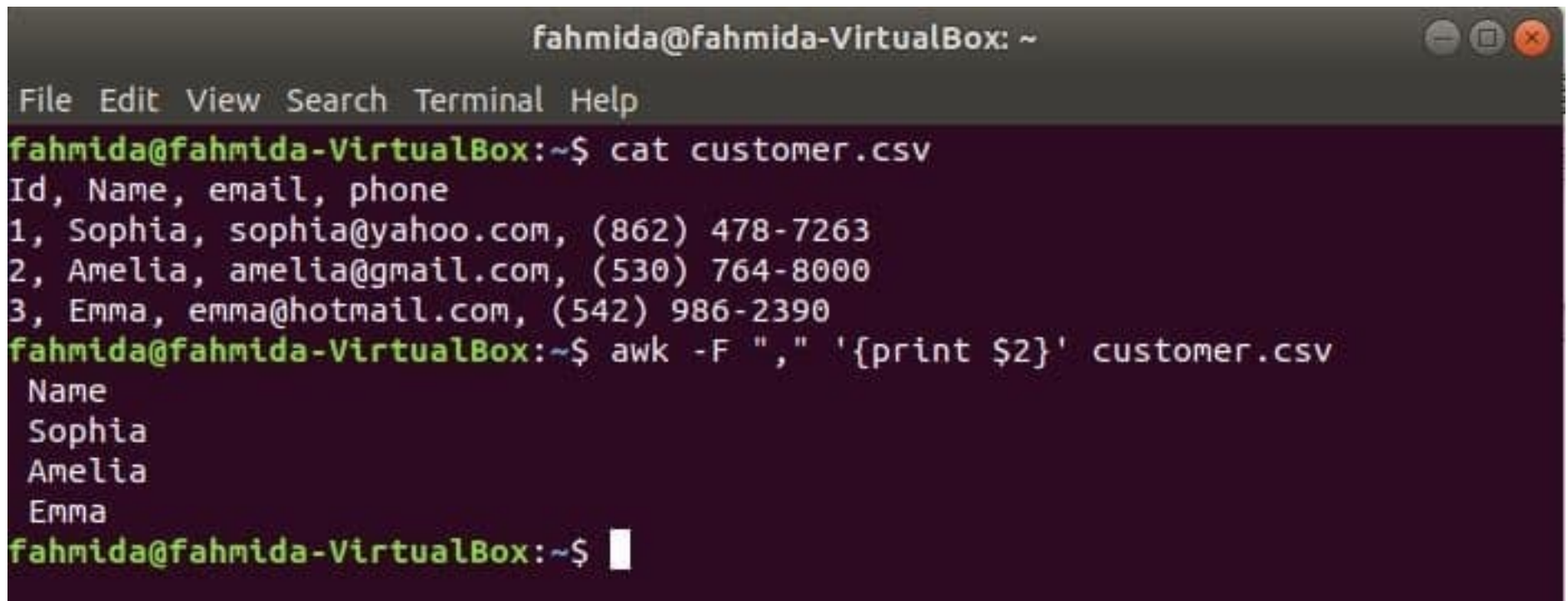
A terminal window titled 'fahmida@fahmida-VirtualBox: ~/Documents' showing the execution of the command `ls -l | awk -v OFS='\t' 'BEGIN { printf "%s\t%s\n", "Name", "Size"} {print $9,$5}'`. The output is a table with two columns: 'Name' and 'Size'. The first two lines of the table are highlighted with red boxes.

Name	Size
input.txt	89
phone.txt	53

parsing and filtering logs with grep, awk and sed

AWK (eg)

```
$ cat customer.csv  
$ awk -F "," '{print $2}' customer.csv
```

A terminal window titled 'fahmida@fahmida-VirtualBox: ~' with a menu bar (File, Edit, View, Search, Terminal, Help). The terminal shows the execution of two commands. The first command is 'cat customer.csv', which outputs a CSV file with four columns: Id, Name, email, and phone. The second command is 'awk -F "," '{print \$2}' customer.csv', which outputs the names from the CSV file: Sophia, Amelia, and Emma.

```
fahmida@fahmida-VirtualBox: ~  
File Edit View Search Terminal Help  
fahmida@fahmida-VirtualBox:~$ cat customer.csv  
Id, Name, email, phone  
1, Sophia, sophia@yahoo.com, (862) 478-7263  
2, Amelia, amelia@gmail.com, (530) 764-8000  
3, Emma, emma@hotmail.com, (542) 986-2390  
fahmida@fahmida-VirtualBox:~$ awk -F "," '{print $2}' customer.csv  
Name  
Sophia  
Amelia  
Emma  
fahmida@fahmida-VirtualBox:~$
```



Log Aggregation and SIEM

What is log aggregation?

Log aggregation is the process of collecting, standardizing, and consolidating log data from across an IT environment in order to facilitate streamlined log analysis. Without log aggregation, developers would have to manually organize, prepare, and search through log data from numerous sources in order to extract useful information from it.

Log aggregation is one of the early stages in the log management process, and it can be accomplished through several different methods:

Log Aggregation and SIEM

File replication: Log files can be copied to a central location using tools like rsync or cron. While these commands can relocate log files, they do not support real-time monitoring.

Syslog: Logs can be directed to a central location using a syslog daemon. This is a relatively simple method of aggregating logs, but it can be difficult to scale.

Automated pipelines: Log processing pipelines incorporate syslog daemons or agents to systematically ship, parse, and index logs on a continuous basis.



Log Aggregation and SIEM

Why is log aggregation important?

Search, filter, and group logs

Troubleshoot in response to production incidents

Collaborate with others across the organization

Perform real-time monitoring



Log Aggregation and SIEM

What to look for when choosing a log aggregation tool

A robust pipeline library

Correlation with other telemetry

Security monitoring

Granular control over indexing and storage

Live tailing



Log Aggregation and SIEM

SIEM: Security Information and Event Management

SIEM falls within the computer security field, and it includes both products and software that help companies manage security events and secure information.

Experts describe SIEM as greater than the sum of its parts. Indeed, SIEM comprises many security technologies, and implementing SIEM makes each individual security component more effective. In effect, SIEM is the singular way to view and analyze all of your network activity.



Log Aggregation and SIEM

SIEM: Benefits

the goal of SIEM is security – and it is only as good as the data it accesses. But advantages of a SIEM approach are its real-time analysis and connecting disparate systems in order to unify the information in one console.

Log Aggregation and SIEM

SIEMs and log management have a lot in common. For starters, they both:

- Enable real-time collection, storage and search of log data across operating systems, security devices, network infrastructure, systems and applications
- Report on operational and compliance performance
- Require dedicated personnel to manage the software, figure out the type of information that needs to be collected, set up log transfer, set up storage workflows, frequently fine-tune the settings, deploy and configure changes, deal with numerous alerts, and sort out false positives

Log Aggregation and SIEM

However, they differ in critical ways as well:

- SIEM logging combines event logs with contextual information about users, assets, threats and vulnerabilities and compares them using algorithms, rules and statistics. Log management provides no analysis of log data; it's up to the security analyst to interpret it and determine whether or not the threat is real. It's difficult to spot aberrant activity and interpret data if you're not yet aware of a problem with a specific account or file.
- SIEM tools provide real-time and historical threat analysis based on log data. They also send alerts whenever a potential security threat is detected, and prioritize the threats according to importance, making it easier for security professionals to tackle issues systematically. Log management does not have features like automated alerting and threat notifications, and so LMS applications are typically not ideal tools to drive decision making.
- Once collected by a SIEM, logs are converted into a uniform format and organized into categories, helping to ensure consistency across all log data. LMS applications do not convert log data across disparate sources into a unified format, resulting in inconsistency and variability across the collected data.