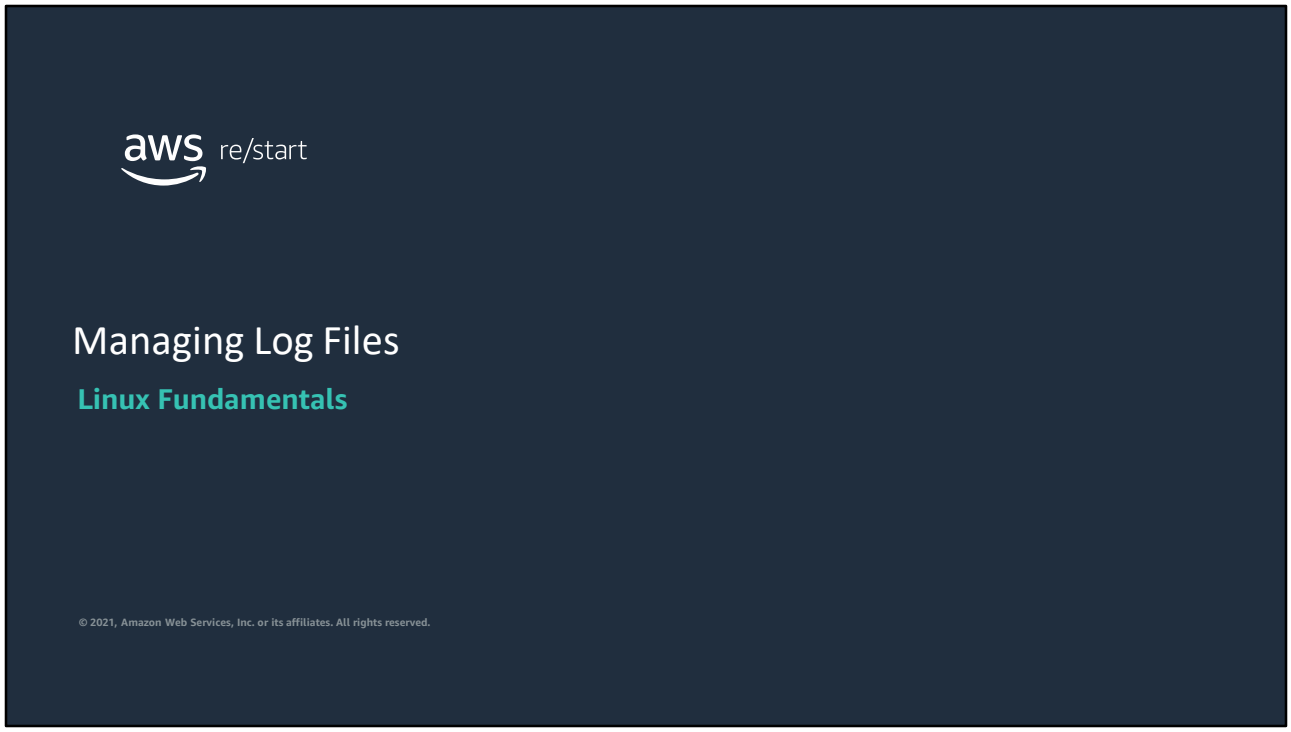


Welcome to Managing Log Files.

What you will learn

At the core of the lesson

You will learn how to:

- Define log files
- Use commands to read different types of messages in a log file
- Recognize the benefits of log rotation

aws re/start

You will learn how to:

- Define log files
- Use commands to read different types of messages in a log file
- Recognize the benefits of log rotation

What is logging?

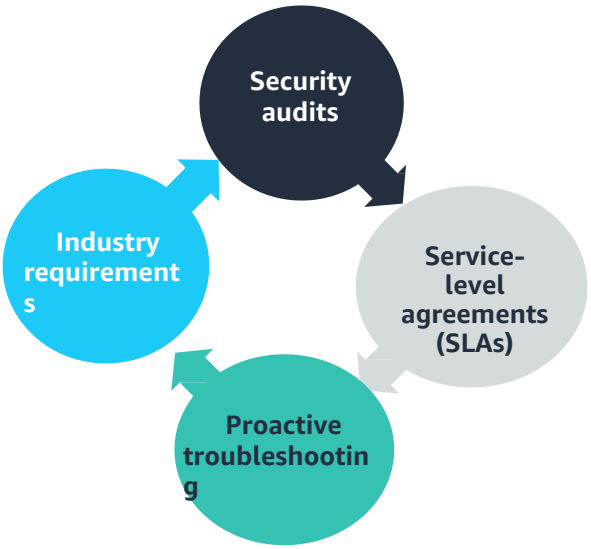
- Logs keep records of events on the system, which helps with auditing.
- The following are types of logs:
  - System logs (system startup information and system shutdown times)
  - Events logs (user login and logout events)
  - Applications logs (startup time, actions, and errors)
  - Services logs

aws re/start

Logs are a record of what happened in the operating system or an application.

They are particularly useful if an issue occurs. Administrators, developers, and others can trace which application triggered an error, which user made a wrong action, or which outside host accessed the server.

The importance of logging



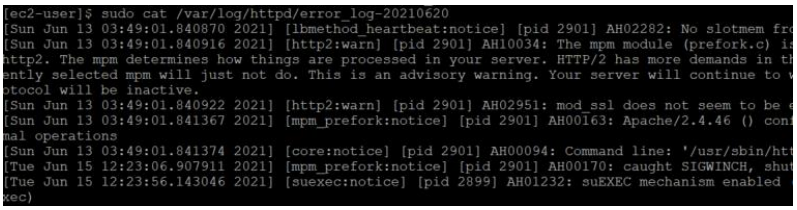
aws re/start

Logging can help troubleshoot issues: What or who caused an error? Did anyone wrongfully access a file, a database, or a server?

Logs are a key to security audits (gathering information about the system) and service-level agreements (troubleshooting must start within *x* hours after an issue occurs).

Example of a log file

sudo cat /var//log/yum.log file lists programs that were installed or updated.



sudo cat /var//log/httpd/error_log file is the log file of the httpd, a web server service.

aws re/start

The first log in the /var/log/YUM.log log file lists programs that were installed or updated. (YUM is a package management utility to install, update, and remove software.)

The second is the log file of the httpd, a web server service.

Logging levels

Severity Level	Identification	Description
0	EMERGENCY	Logs messages when the system becomes unstable
1	ALERT	Logs when immediate action is needed
2	CRITICAL	Logs only messages for critical errors; the system may become unusable
3	ERROR	Logs only messages that indicate non-critical error conditions or more serious messages
4	WARN	Logs only messages that are warnings or more serious messages (usually the default log level on Linux distributions)
5	NOTICE	Logs messages for normal events but of significant importance
6	INFO	Logs all informational messages and more serious messages
7	DEBUG	Logs all debug-level and INFO messages

These logging levels are the logs that the Linux operating system offers. Some programs or libraries may offer fewer logging levels, such as debug, info, warning, and error, with a default logging level set to info.

The logs of the current logging level and higher levels are displayed. For example, when choosing WARN, level 4, the logs of NOTICE, INFO and DEBUG are also displayed.

System logs

The `tail`, `head`, and `less` commands are used to view log file entries. Use `grep` to look for patterns.

```
[ec2-user]$ sudo tail /var/log/yum.log | grep httpd
Jun 10 13:55:49 Installed: httpd-tools-2.4.46-1.amzn2.x86_64
Jun 10 13:55:49 Installed: generic-logos-httpd-18.0.0-4.amzn2.noarch
Jun 10 13:55:49 Installed: httpd-filesystem-2.4.46-1.amzn2.noarch
Jun 10 13:55:50 Installed: httpd-2.4.46-1.amzn2.x86_64
[ec2-user]$
```

```
[ec2-user]$ sudo head -n 5 /var/log/yum.log
May 28 08:35:38 Updated: glibc-minimal-langpack-2.26-45.amzn2.x86_64
May 28 08:35:39 Updated: glibc-common-2.26-45.amzn2.x86_64
May 28 08:35:39 Updated: glibc-2.26-45.amzn2.x86_64
May 28 08:35:40 Updated: libcrypt-2.26-45.amzn2.x86_64
May 28 08:35:40 Updated: python3-pip-20.2.2-1.amzn2.0.2.noarch
[ec2-user]$
```

```
[ec2-user]$ sudo tail -n 5 /var/log/yum.log
Jun 10 13:55:49 Installed: generic-logos-httpd-18.0.0-4.amzn2.noarch
Jun 10 13:55:49 Installed: mailcap-2.1.41-2.amzn2.noarch
Jun 10 13:55:49 Installed: httpd-filesystem-2.4.46-1.amzn2.noarch
Jun 10 13:55:50 Installed: mod_http2-1.15.14-2.amzn2.x86_64
Jun 10 13:55:50 Installed: httpd-2.4.46-1.amzn2.x86_64
[ec2-user]$
```


aws re/start

cat, less, more, tail, and **head** are all commands that are useful to read logs. Using the pipe redirector **|** and **grep** is an efficient way to look for a specific pattern in a log.

You can also open the files using editors such as vi or gedit.

Using `grep` to search log files

- The `grep` command searches the given file for lines that contain a match to the specified strings or words.
- Add the `grep` command when you look for a specific string of text in log files.
- `grep` is one of the most useful commands in Linux.
- The following are examples:
  - `cat yourlog.log | grep ERROR`
  - `tail -f yourlog.log | grep error`
  - `sudo cat /tmp/log/secure | grep LOGIN > SharedFolders/logins.csv`

```
[ec2-user]$ sudo tail /var/log/secure | grep "invalid user"
Jun 22 14:45:45 ip-172-31-27-186 sshd[1131]: input_userauth_request: invalid user ec3-user [preauth]
Jun 22 14:45:59 ip-172-31-27-186 sshd[1137]: input_userauth_request: invalid user ec4-user [preauth]
[ec2-user]$
```

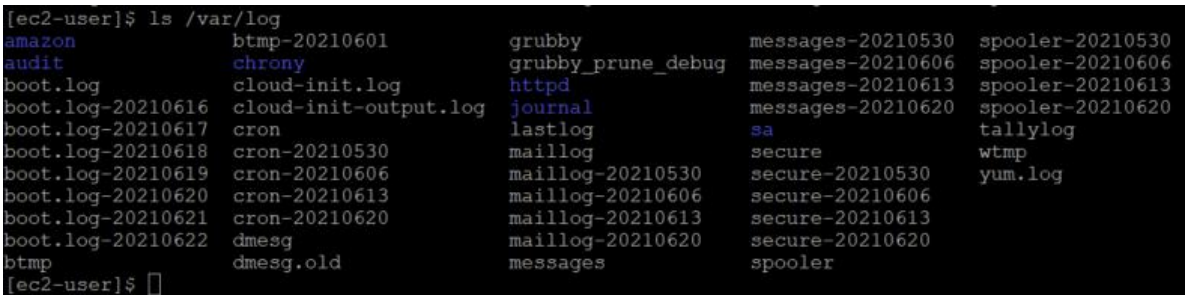

aws re/start

The following are some more details on these errors.

- **cat yourlog.log | grep ERROR**: searches for the ERROR message in the yourlog.log file and displays the matching lines of the file in the console (reads the content of the yourlog.log file and redirects it to the grep command that searches for the word ERROR).
- **tail -f yourlog.log | grep error**: searches the word error in the last 10 lines of the yourlog.log file and displays the matching lines of the file in the console (reads the last 10 lines of the yourlog.log file and redirects it to the grep command that searches for the word error; remember that without any additional option, tail reads the last 10 lines of a file).
- **sudo cat /var/log/secure | grep LOGIN > SharedFolders/logins.csv**: searches the word LOGIN in the file /var/log/secure and writes the matching lines of the file in the file SharedFolders/logins.csv. Sudo is required to access the /var/log/secure file.

Where does Linux store log files?

- Linux and applications normally store log files in the /var/log directory.
- The /var directory is used to store files that might rapidly and unpredictably change in size.



aws re/start

Linux usually stores log files in the /var/log directory.

Important log files

Log File	Description
/var/log/syslog	Stores system information
/var/log/secure	Stores authentication information for Red Hat-derived distributions
/var/log/kern	Stores Linux kernel information
/var/log/boot.log	Stores startup messages
/var/log/maillog	Stores mail messages
/var/log/daemon.log	Stores information about running background services
/var/log/auth.log	Stores authentication information for Debian-derived distributions
/var/log/cron.log	Stores cron messages for scheduled tasks
/var/log/httpd	Stores Apache information for Red Hat-derived distributions

Other important log files include the following:

- /var/log/YUM: Stores YUM installer information for Red Hat-derived distributions
- /var/log/apache2/access.log: Stores Apache authentication information for Debian-derived distributions
- /var/log/lastlog: Stores information about successful logins to the host

The `lastlog` command

Reports recent login information for the system

Can report all logins or login information for a specific user

All users

```
Username         Port     From             Latest
ec2-user         pts/0    72-21-198-64.ama Wed Jun 23 08:10:16 +0000 2021
mmajor           pts/0                     Tue Jun 22 09:31:07 +0000 2021
jdoe             pts/0                     Mon Jun 21 08:25:22 +0000 2021
```

Specific user

```
Username         Port     From             Latest
ec2-user         pts/0    72-21-198-64.ama Wed Jun 23 08:10:16 +0000 2021
```


aws re/start

The **lastlog** command retrieves user information from the **/var/log/lastlog** file and outputs it in the console.

lastlog –u ec2-user displays information of the ec2-user only.

lastlog –t 1 displays login information more recent than 1 day ago.

Use **man lastlog** for more options.

Log rotation

- Servers typically run large applications.
  - Servers often log every request.
  - This logging leads to bulky log files.
- Log rotation can help with the following in regard to bulky logs:
  - It is a way to limit the total size of the logs that are retained.
  - It still helps analysis of recent events.
- Log rotation is an automated process that is used in system administration where dated log files are archived.

Log rotation is not activated by default.

With the **logrotate** utility, you can compress, rename, or clean up the log files.

You can activate log rotation according to the log file size: if the log file is more than a specific size, it will be renamed xxxxx.log-20210612 (if it is renamed on June 21, 2021). The default date format is **yyyymmdd**.

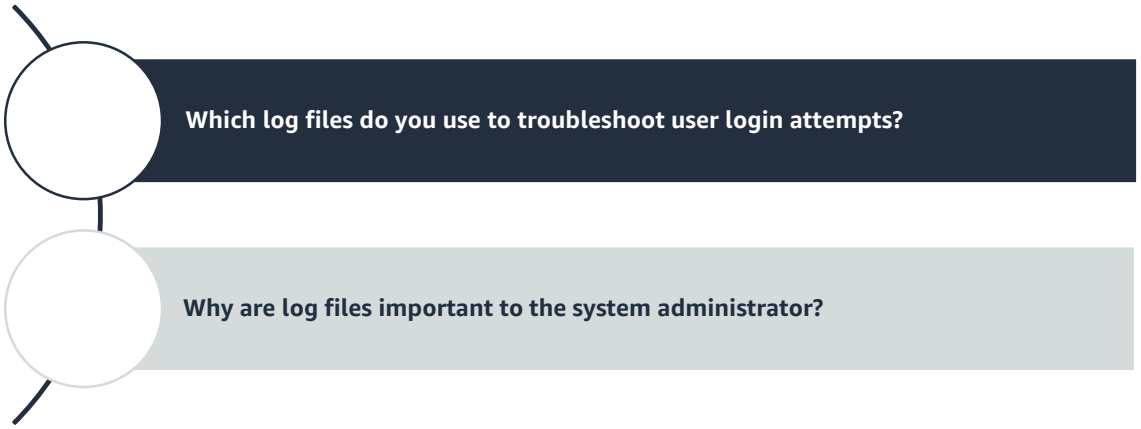
It can also be activated on a regular basis (weekly, daily or monthly).

A maximum number of log files to keep can be set: if the maximum number of logs is reached, the logs are erased, moved, or emailed.

You can also compress log files.

You can individually tailor log rotation for different kinds of logs.

Checkpoint questions



aws re/start

1. You use either **/var/log/auth.log** or **/var/log/secure** depending on whether the system is Debian-derived or Red Hat-derived.
2. Log files are running records of what is occurring on a system. These records include authentication and also logs from programs, status of services, and many other server events.

Key takeaways

- You use logging to record events that are happening with the system.
- Log files can become large. To keep the logs manageable, use log rotation to routinely save files.
- You can control the amount of detail in the logs with logging levels.

aws re/start

Some key takeaways from this lesson include:

- You use logging to record events that are happening with the system.
- Log files can become large. To keep the logs manageable, use log rotation to routinely save files.
- You can control the amount of detail in the logs with logging levels.

Thank you

aws re/start