Log Viewer Setup and Usage

Follow these steps to set up a secure user with restricted access to view Laravel logs.

Steps to Set Up

1. Create a New User

Run the following command:

```
sudo adduser <user-name>
```

Replace `<user-name>` with the desired username.

2. Create the Log Viewer Script

Create a script file:

```
sudo nano /usr/local/bin/logviewer.sh
```

Add this content:

```
#!/bin/bash

echo "Select the log file to view:"
echo "1) Laravel Log"
echo "2) Worker Log"
echo "3) Exit"

read -p "Enter your choice: " choice

case $choice in
    1)
        echo "Showing Laravel Log..."
        tail -f /var/www/<your-laravel-project>/storage/logs/laravel.log
        ;;
    2)
        echo "Showing Worker Log..."
        tail -f /var/www/<your-laravel-project>/storage/logs/worker.log
        ;;
    3)
        echo "Exiting..."
        exit 0
        ;;
    *)
        echo "Invalid choice. Exiting."
        exit 1
        ;;
esac
```

In Case of Scenario 2:

If your setup generates multiple log files daily and you need to access the latest file, use the following script:

Add this content

```
#!/bin/bash

echo "Select the log file to view:"

echo "1) Laravel Log"

echo "2) Worker Log"

echo "3) Exit"


read -p "Enter your choice: " choice

case $choice in

    1)

        # Find the latest Laravel log file

        latest_laravel_log=$(ls -t /var/www/<your-laravel-project>/storage/logs/laravel-*.log | head -n 1)


        if [ -z "$latest_laravel_log" ]; then

            echo "No Laravel log files found."

            exit 1

        fi


        echo "Showing Latest Laravel Log: $latest_laravel_log"

        tail -f "$latest_laravel_log"

        ;;

    2)

        echo "Showing Worker Log..."

        tail -f /var/www/<your-laravel-project>/storage/logs/worker.log

        ;;

    3)

        echo "Exiting..."

        exit 0

        ;;

    *)
```

```
    echo "Invalid choice. Exiting."
    exit 1
    ;;
esac
```

Make it executable:

```
sudo chmod +x /usr/local/bin/logviewer.sh
```

3. Restrict User Access to the Script

Open the SSH configuration file:

```
sudo nano /etc/ssh/sshd_config
```

Add this section:

```
Match User <user-name>
    ForceCommand /usr/local/bin/logviewer.sh
```

Replace `<user-name>` with the username created earlier.

Restart SSH service:

```
sudo systemctl restart sshd
```

4. Generate SSH Key for the User

Switch to the new user:

```
sudo su - <user-name>
```

Generate an SSH key:

```
ssh-keygen -t rsa -b 4096
```

The public key is saved in `~/.ssh/id_rsa.pub`.

5. Add Developer's Public Key

Add the developer's public key to `authorized_keys`:

```
echo "<developer-public-key>" >> ~/.ssh/authorized_keys
```

Set permissions:

```
chmod 700 ~/.ssh
chmod 600 ~/.ssh/authorized_keys
```

6. Test the Setup

From the developer’s machine, connect to the server:

```
ssh <user-name>@<server-ip>
```

You will see this prompt:

```
Select the log file to view:
1) Laravel Log
2) Worker Log
3) Exit
```

Choose an option to view the logs.

Notes:

- Ensure the script has the correct paths to log files.
- Test the setup before using it in production.
- To exit, type `3` at the prompt.