

Automating Deployment

Hosting Django & React
Applications

OWNER

K.Kamal
Hanesha

CONTRIBUTORS

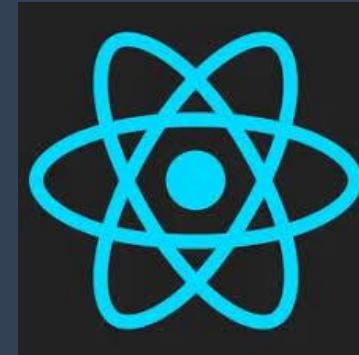
Alan P G
Khavij VB

The Mission

Goal: Deploy a full-stack web application with a Django backend and a React frontend without manual headaches.

Solution: A robust shell script that uses standard Linux commands to:

- Locate project files automatically.
- Configure the server environment.
- Verify dependencies and network status.
- Update system configurations securely.



| File Operations

> **cat** (Concatenate)

Reads data from files and outputs their content to the terminal. It's the standard way to view a file quickly.

> **echo** (Display Text)

Prints text to the screen. When combined with `>>`, it appends text to a file.

> **read** (User Input)

Pauses the script to take input from the keyboard and assigns it to a variable.



Search & Navigate

> **find**

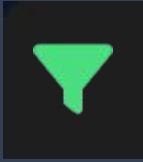
The ultimate search tool. It recursively hunts for files based on names, types, or sizes within a directory hierarchy.

> **dirname**

A path manipulation utility. It strips the filename from a full path, returning just the directory portion. Essential for relative linking.



Text Processing: The Filters



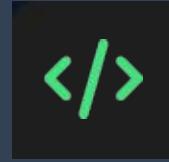
grep

Global Regular Expression Print. It filters input lines, outputting only those that match a specific pattern.



sed

Stream Editor. A powerful tool for parsing and transforming text, often used for find-and-replace operations.



awk

A complete scanning and processing language. It excels at extracting specific columns of data from structured output.

Text Processing: The Slicers

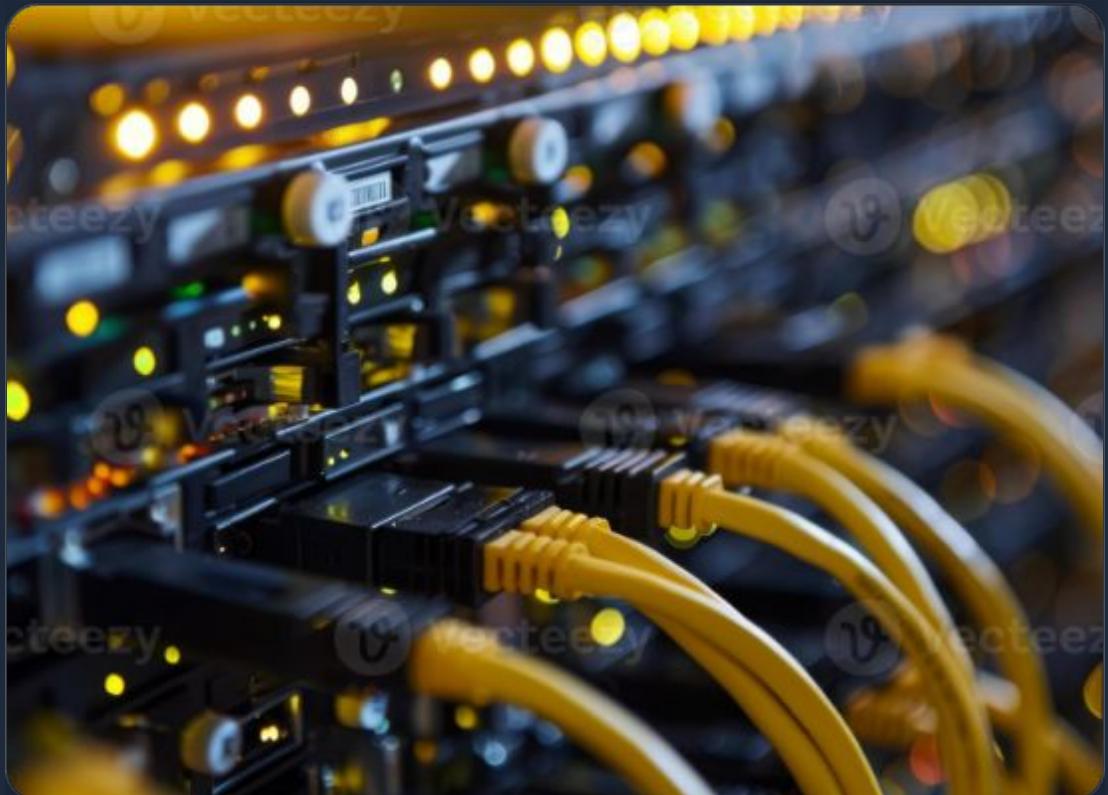
cut

Removes sections from each line of a file. It is typically used with a delimiter (like a comma or colon) to extract specific fields.

head & tail

These commands output the very beginning or the very end of a file. Useful for checking log files or headers without opening the entire document.

| System & Network Info



> **ip addr show**

Displays detailed information about network interfaces, used here to find the server's IP.

> **nmcli**

The NetworkManager Command Line Interface. Used to report network connectivity status.

> **dpkg -l**

Lists installed packages on Debian-based systems. Used to verify if specific software is present.

Logic & Control Flow

- > **set -e**
A crucial safety switch. It forces the script to exit immediately if any command fails, preventing cascading errors.
- > **if [...]; then**
The building block of logic. Checks conditions (like if a file exists) to decide which code to run.
- > **export**
Makes a variable available to child processes and the environment, not just the current script.



Privileges & Output

> **sudo** (SuperUser Do)

Executes a command with administrative (root) privileges. Essential for modifying system configuration files.

> **tee**

Reads from standard input and writes to both standard output *and* files simultaneously. When combined with sudo, it allows writing to protected files.



Thank You!

Questions



Project by **K.Kamal Hanesha**