DevOps Learning Notes

VI Editor & AWS Security Group

In this lecture you are learning LINUX VI Editor & AWS Security Group:

1. Vi-Editor
2. Command Mode vs. Insert Mode
3. Linux Environment Variable
4. PATH environment variable
5. AWS Security Group
6. Network Terminologies - Port, Protocol, Service

VI-EDITOR: is used to add/modify/edit the data in any regular files. Vi-Editor has two operating modes

1. insertion mode
2. command mode

VI-EDITOR begins in command mode, where the cursor movement and text deletion and pasting occur. Insertion mode begins upon entering an insertion (i) or change (a) command.

[ESC] returns the editor to command mode. Most commands execute as soon as you type them except for "colon" commands which execute when you press the return key.

Command mode: This mode enables user to perform administrative tasks such as saving files, executing commands, moving the cursor, cutting (yanking) and pasting lines or words, and finding and replacing. In this mode, whatever user type is interpreted as a command.

Insert mode: This mode enables user to insert text into the file. Everything that's typed in this mode is **interpreted** as input and finally it is put in the file.

The vi always starts in command mode. To enter text, user must be in insert mode. To come in insert mode user simply type **i**. To get out of insert mode, press the **Esc** key, which will put user back into command mode.

1. When you open the vim editor, it will be in the command mode by default.
2. If you are not sure which mode you are in, press the Esc key twice, and then you'll be in command mode. You open a file using vi editor and start type some characters and then come in command mode to understand the difference.
3. The vi is case-sensitive, so you need to pay special attention to capitalization when using commands.
4. Most commands in vi can be prefaced by the number of times you want the action to occur. For example, 2j moves cursor two lines down the cursor location.

Here describing Vi-Editor commands which mostly used to manage the content in a file in Linux.

How to Move the Cursor

- **k –** Move the cursor to top line
- **j –** Move the cursor to the bottom line
- **h –** Move the cursor to the right by one cursor position (from system side)
- **l –** Move the cursor to the left by one cursor position (from system side)

Below table shows very frequently used commands in Vi-Editor:

Insert Mode:

i	To begin insert mode at the cursor position

X

w	To move the cursor forward, word by word
b	To move the cursor backward, word by word
nw	To move the cursor forward to n words (5W)
nb	To move the cursor backward to n words (5B)
u	To undo last change (word)
U	To undo the previous changes (entire line)
Ctrl+R	To redo the changes
yy	To copy a line nyy To copy n lines (5yy or 4yy)
p	To paste line below the cursor position
P	To paste line above the cursor position
dw	To delete the word letter by letter (like Backspace)
x	To delete single character at cursor position
dd	To delete entire line
ndd	To delete n no. of lines from cursor position(5dd)
/	To search a word in the file

Extended Mode: (Colon Mode)

Extended Mode is used for save and quit or save without quit using “Esc” Key with “:”

:w	To Save the changes
:q	To quit (Without saving)
:wq	To save and quit
:w!	To save forcefully
wq!	To save and quit forcefully
:x	To save and quit
:X	To give password to the file and remove password
:20(n)	To go to line no 20 or n
: set nu	To set the line numbers to the file
:set nonu	To Remove the set line numbers

Linux Environment Variable:

Environment variables stores the dynamic system values that can be used within the shell or its child processes. To know the list of environment variables that are set in linux system run the command below.

```
[edwiki@ip-10-8-2-22 ~]$ env
SHELL=/bin/bash
TERM=screen
HISTSIZE=1000
USER=edwiki
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin
PWD=/home/edwiki
HOME=/home/edwiki
LOGNAME=edwiki
[edwiki@ip-10-8-2-22 ~]$
```

Below are the most common environment variables that will be used in Linux systems:

1. PWD – Current working directory.
2. HOME – The user’s home directory location.

X

AWS Security Group:

Security groups are way to control inbound and outbound traffic to EC2 instance. AWS security groups will act as a virtual firewall for EC2 instances, and provide more and additional layer of security for your resources in the AWS cloud.

Inbound and Outbound rule: Security groups can have rules for both inbound and outbound traffic, allowing you to control traffic from both inbound and outbound traffic.

Stateful: Security groups are stateful, this means if we make any changes applied on inbound rule it will be automatically applied to the outbound rule.

Key Notes:

- You can create 60 inbound and outbound rules per security group. This quota can be increased by raising request with AWS Support.
- Security Groups are region specific resources.
- One Security Group can be associated with multiple EC2 Instances that belongs to same region and in same network.
- One EC2 instance can be associated with maximum of 5 Security Groups to manage the rules of multiple applications that run on single EC2 instance.
- When you add, update, or remove rules, your changes are automatically applied to all resources associated with the security group.
- You can create Security Group in the Launch EC2 instance process or can create it explicitly and then associate it with EC2 by selecting the existing Security Group.
- You can specify allow rules, but not deny rules.
- When you first create a security group, it has no inbound rules. Therefore, no inbound traffic is allowed until you add inbound rules to the security group.
- When you first create a security group, it has an outbound rule that allows all outbound traffic from the resource. You can remove the rule and add outbound rules that allow specific outbound traffic only. If your security group has no outbound rules, no outbound traffic is allowed.
- Ensure that you are now allowing the traffic to the source IP range – 0.0.0.0/0 for ports like SSH (22), RDP (3389).

Network Terminologies:

Protocol:

A protocol is a set of rules that must be followed between the client and server to process the data/request. Client ensure the protocol to send the request that can be understand by the server.

Let say SSH client ensure to follow the SSH protocol to send the Login request to the server and in the server side the SSHD daemon ensure to receive the request and understand the request by using the SSH protocol and process the request.

Port: Network port is just a logical number that will be sending throught the client request to let the server program know who is responsible to process this request.

Service: Service is the process that runs on the server side to process the client requrst. Each service is binded with a port number to be responsible to process the requests that reach to that port number.

Below table shows some common services and the associated port assignments in linux systems

Linux Commands:

uptime: To check since how long the system is in running state.

[edwiki@ip-10-8-2-22 ~]$ uptime

 02:14:10 up 26 min, 1 users, load average: 0.08, 0.10, 0.10

X

Comments(0)

Enter your email Subscribe