**MANAGE JENKINS**

## WHAT DID WE LEARN ?

* Jenkins has a dedicated Management page with many sections, providing:
  + System, Tool configurations
  + Security options
  + Metrics, Logs, Configuration Item overview
  + Lifecycle Management utilities
* Jenkins can be managed with Apache Groovy Code
  + Helps integrating with Configuration Management
  + Automate large scale Jenkins deployment

**MANAGE JENKINS CONSOLE**



## NOTIFICATIONS ABOUT UPDATES

* The top of the screen may contain "Monitors" that alert you when a new version of the Jenkins software or a security update is available.
* You should apply all updates as soon as possible, according to your processes (staging environment, validation, backup/restore). We will cover this later.
  + New features, bug fixes and security updates
  + Read the changelog for information about what is in the package
  + Click the link to download and install the system

## GENERAL SYSTEM-WIDE CONFIGURATIONS

## CONFIGURE SYSTEM

## GLOBAL TOOL CONFIGURATION

## MANAGE PLUGINS

## MANAGE NODES & CLOUD

## 1] CONFIGURE SYSTEM

## HOME DIRECTORY & SYSTEMS MESSAGE

## JENKINS URL

## JENKINS LOCATION

## QUIET PERIOD

## TIMESTAMP

## #EXECUTOR

## EMAIL NOTIFICATION

## GITHUB API USAGE

## GLOBAL PIPELINE LIBRARIES

## GLOBAL PROPERTIES [ENV VARAIBLE]

**1.1] HOME DIRECTORY**

Jenkins stores information about your build server configuration, your build jobs, build artifacts, user accounts, and other useful information, as well as any plugins you may have installed. The Jenkins home directory format is backward compatible across versions, so you can freely update or reinstall your Jenkins executable without affecting your Jenkins home directory.

* Default home directory in redhat **[/var/lib/Jenkins**]
* Using the default path is not recommended. Couple of reasons
  + /var is operating system related file system.
  + Every time application team do maintenance, root will be needed, so extra work for system admins.
  + Storage
* To change this location, modify the value of the **JENKINS\_HOME** environment variable.

**1] Stop the Jenkins server**

* **systemctl stop Jenkins**

**2] Move the default home directory to new home**

* **mkdir –p /home/jenkins\_home**
* **chown –R jenkins:Jenkins jenkins\_home**
* **cp -pr /var/lib/jenkins /home/Jenkins\_home**
* **usermod -d /home/jenkins jenkins**

**3] Update the new Jenkins path in /etc/sysconfig/jenkins**

* **vim /etc/sysconfig/jenkins**

**JENKINS\_HOME=”/home/Jenkins\_home”**

**4] Start the Jenkins server**

* **sudo systemctl start Jenkins**

**Note: When we update the Jenkins home directory it will asks us new admin secret can be found under new Jenkins home /home/jenkins/secrets/initialAdminPassword.**

* **Set JENKINS\_HOME as a global parameter**

**Vim /etc/profile**

**Export JENKINS\_HOME=”/apps/jenkins”**

**1.2] SYSTEM MESSAGE**

**Text that is displayed at the top of your Jenkins home page**

* Can include HTML tags
* Use this page to display the name of the server and a short description of its purpose
* May include a pointer to a Wiki page and information about who to contact
* Display announcements about scheduled down time or other information of interest to all Jenkins users

## 1.3] EXECUTORS

* What is an ‘Executor’

1. Unit of task execution on a computer
2. Defines upper bound of concurrent build execution

* Typically, you set this to # of CPU cores Unless your builds are highly parallel in itself
* This configures the number of executors on the master
* Set this to 0 to prevent builds from running on the master
* Watch out for memory requirement

## 1.4] QUIET PERIOD

* Waits for this many seconds before actually starting the build
* When a build is scheduled into the queue with quiet period, the build will sit in the queue until the quiet period expires. If during this period, additional attempts are made to put the same build in the queue, the quiet period resets to its initial value.
* CI server should wait for the burst to finish before attempting a build. This is said to reduce the chance of having broken build, and it is also sometimes useful in reducing the average turn-around time for builds that take longer.
* Avoid setting quiet period longer than the polling interval.
* The quiet period introduces a delay so the builds are not occur quite so continuously.

## 1.5] TIME STAMPER

* Scroll down the page to the "Timestamper" area
* Use these fields to define the format used to display time stamp information

## 1.6] JENKINS LOCATION

* Replace the localhost:8080 string with the actual URL that people use to access Jenkins
* Define the email address to which email about general Jenkins issues are sent
* This can be an individual’s email address or an email alias that contains multiple email addresses

**2] GLOBAL TOOLS CONFIGURATION**

Use the "Global Tools Configuration" page to configure the tools used for Pipeline development:

## WHAT TOOLS ARE CONFIGURED?

* Some tools are listed here by default; others only if the appropriate plugins are installed
  + - JDK and other languages
    - Build tools: Maven, Gradle, Ant and others
    - SCM: Git, Mecurial and others
    - Containers: Docker and Kubernetes (when installed)

**3] MANAGE PLUGINS**

* Jenkins uses plugins to provide much of its functionality

**4] MANAGE NODES & CLOUD**

* Use the "Manage Nodes" screen to create, configure and view nodes:
* A **node** is a server where Jenkins runs jobs on **executors**
* The **agent** is the tool that manages the **executors** on a remote **node**, on behalf of Jenkins.
* The Jenkins master also runs on a **node**.

## SECURITY

## GLOBAL SECURITY

## MANAGE CREDENTIALS

## CONFIGURE CREDENTIAL PROVIDERS

## MANAGE USERS

## IN-PROCESS SCRIPT APPROVAL

## 1] CONFIGURE GLOBAL SECURITY

* The "Configure Global Security" screen is used to:
  + Define how users are authenticated and what they are authorized to do
  + Control other security settings
* Authentication [LDAP]
* Authorization [RBAC]
* API token
* SSH server

## 2] MANAGE USERS

* The "Manage Users" screen is used to add user to the Jenkins user database and lists all users who are in that database

## 3] CONFIGURE CREDENTIALS

* Jenkins credentials control access to third-party sites and applications such as artifact repositories and cloud-based storage systems.
* Use the "Configure Credentials" screen to manage credentials:

## STATUS INFORMATION

## SYSTEM INFORMATION

## SYSTEM LOG

## LOAD STATISTICS

## ABOUT JENKINS

## 1] LOAD STATISTICS [MONITORING NODE USAGE]

* Use the "Load Statistics" page to monitor node utilization for your Jenkins instance
* The "Load Statistics" reports the following information about the nodes configured on your Jenkins instance:
  + - Number of online executors
    - Number of busy executors
    - Number of available executors
    - Queue length (number of jobs that are waiting for an available executor)
* Use this information to determine when you need to add more nodes and/or executors to your instance to improve throughput

**2] SYSTEM INFORMATION**

* **System Properties** that can be used as arguments to the command line used to start Jenkins
* **Environment Variables** with current values
* List of **Plugins** installed on the system

**3] SYSTEM LOG**

* The "System Log" page includes a link to the log about activities on the Jenkins instance itself
* Note that this information is independent of the build logs that are presented for each job or Pipeline that runs

**4] ABOUT JENKINS**

* The "About Jenkins" page shows the current release of Jenkins on your system plus information about licenses for all components:

**TROUBLESHOOTING**

**MANAGE OLD DATA**

**TOOLS & ACTIONS**

* **RELOAD CONFIGURATION FROM DISK**
* **jenkins cli**
* **script console**
* **prepare for shutdown**

**1] RELOAD CONFIGURATION FROM DISK**

* Click on "Reload Configuration from Disk" to refresh the Jenkins configuration files and directory structure without restarting Jenkins
* Use this when you modify Jenkins or its environment outside of the UI
  + For example, when you edit config files from the command line or move jobs between folders using the command line
* Not necessary when you modify the configuration from the UI or restart Jenkins after modifying the configuration
* Read configuration again from $JENKINS\_HOME without Jenkins restart

## 2] SCRIPT CONSOLE

* The Script Console allows you to type in and execute an arbitrary Groovy script on the server.
* Apache Groovy is the foundation of the DSL used for Jenkins Pipelines
* This is useful for troubleshooting and diagnostics
* Scripts execute in a Groovy Sandbox that limits the internal APIs that are accessible
* With the Script Security plugin, administrators can use these results to manage which unsafe methods (if any) are allowed in the Jenkins environment

**3] JENKINS CLI**

* The Jenkins Command Line Interface (CLI) provides commands that perform functions that are usually executed using the UI
* These can be used in scripts that automate administrative tasks
* Can be accessed in either of the following ways:
* Over SSH
* With the Jenkins CLI client, which is a .jar file distributed with Jenkins
* See [Command Line Interface](https://jenkins.io/doc/book/managing/cli/) in the Jenkins Handbook for more information

**4] PREPARE FOR SHUTDOWN**

* Prevents new jobs from being started
* Waits for running jobs to complete (clean job termination)