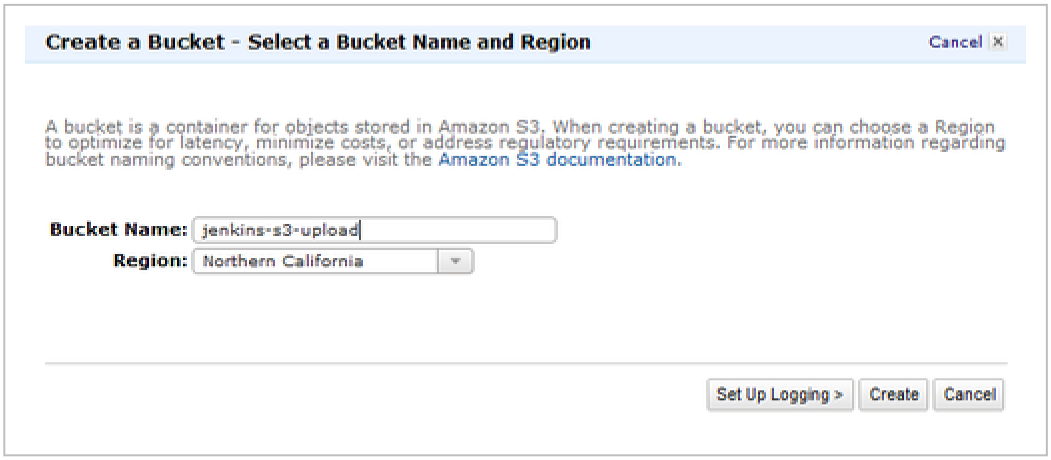
Upload Jenkins Build Artificats to Amazon S3

My previous two blogposts talks about jenkins installation and configuration on Amazon Linuxand jenkins integration with GitHub to trigger automatic build process.

This post focus around uploading build artifacts to amazon s3.

To upload your build artifacts to amazon s3, create a S3 bucket.

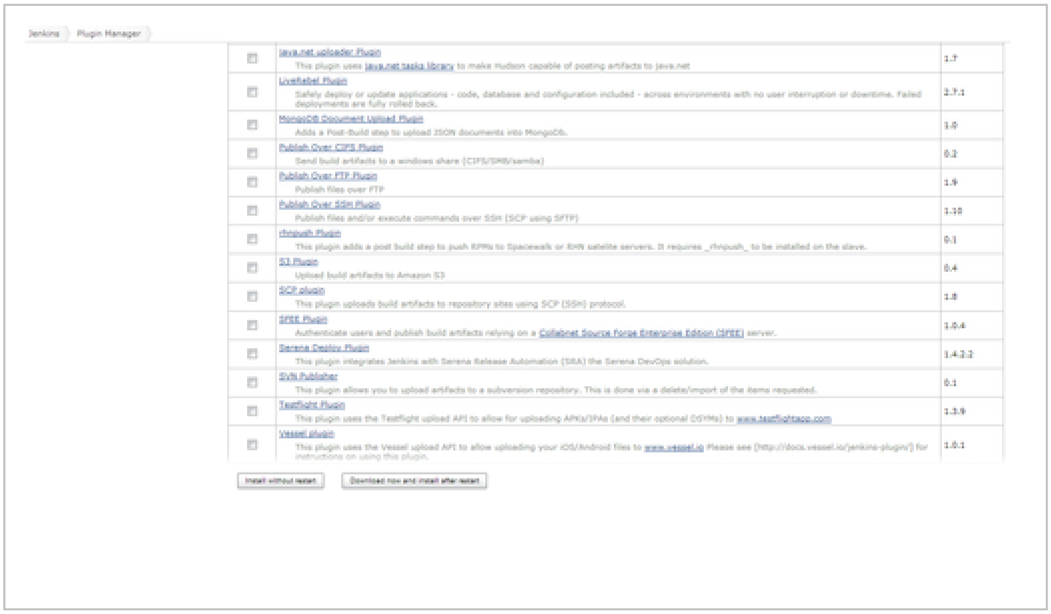


Create a Bucket - Select a Bucket Name and Region

A bucket is a container for objects stored in Amazon S3. When creating a bucket, you can choose a Region to optimize for latency, minimize costs, or address regulatory requirements. For more information regarding bucket naming conventions, please visit the Amazon S3 documentation.

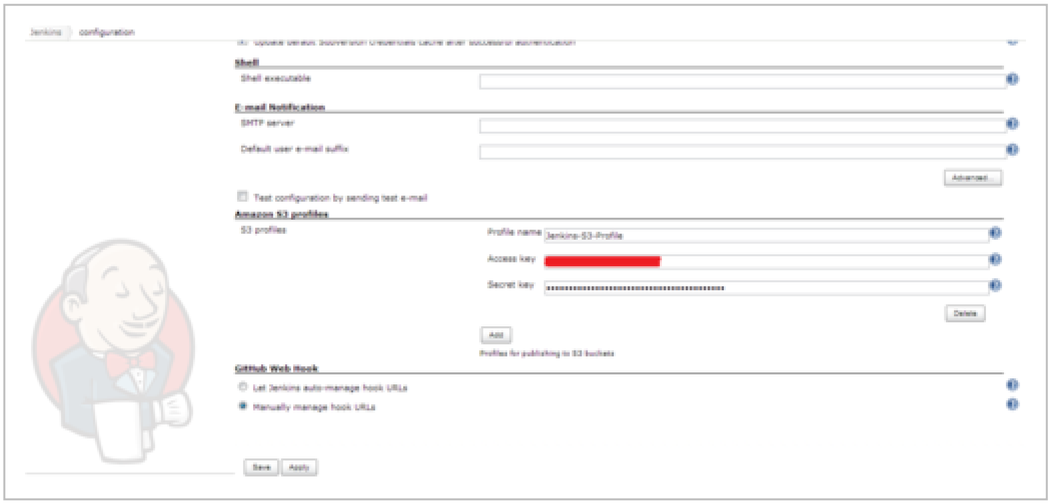
Bucket Name: jenkins-s3-upload
Region: Northern California

Once done, navigate to Jenkins dashboard -> Manage Jenkins -> Manage Plugins and select available tab. Look for “S3 plugin” and install that.



□	java.net uploader Plugin This plugin uses java.net tasks library to make Hudson capable of posting artifacts to java.net	1.7
□	LiveRebel Plugin Safely deploy or update applications - code, database and configuration included - across environments with no user interruption or downtime. Failed deployments are fully rolled back.	2.7.1
□	MongoDB Document Upload Plugin Adds a Post-Build step to upload JSON documents into MongoDb.	1.0
□	Publish Over CIFS Plugin Send build artifacts to a windows share (CIFS/SMB/samba)	0.2
□	Publish Over FTP Plugin Publish files over FTP	1.9
□	Publish Over SSH Plugin Publish files and/or execute commands over SSH (SCP using SFTP)	1.10
□	rhnpush Plugin This plugin adds a post build step to push RPMs to Spacewalk or RHN satellite servers. It requires _rhnpush_ to be installed on the slave.	0.1
□	S3 Plugin Upload build artifacts to Amazon S3	0.4
□	SCP plugin This plugin uploads build artifacts to repository sites using SCP (SSH) protocol.	1.8
□	SFEE Plugin Authenticate users and publish build artifacts relying on a Collabnet Source Forge Enterprise Edition (SFEE) server.	1.0.4
□	Serena Deploy Plugin This plugin integrates Jenkins with Serena Release Automation (SRA) the Serena DevOps solution.	1.4.2.2
□	SVN Publisher This plugin allows you to upload artifacts to a subversion repository. This is done via a delete/import of the items requested.	0.1
□	Testflight Plugin This plugin uses the Testflight upload API to allow for uploading APKs/IPAs (and their optional DSYMs) to www.testflightapp.com	1.3.9
□	Vessel plugin This plugin uses the Vessel upload API to allow uploading your iOS/Android files to www.vessel.io Please see [http://docs.vessel.io/jenkins-plugin/] for instructions on using this plugin.	1.0.1

Once done, go back to Manage Jenkins and select “Configure System” and look for “Amazon S3 Profiles”. Please provide a profile name, access key and secret access key for your AWS account. Don’t provide ROOT AWS User credentials. Create an IAM user will S3 permissions.



Jenkins › configuration

Shell

Shell executable

E-mail Notification

SMTP server

Default user e-mail suffix

□ Test configuration by sending test e-mail

Amazon S3 profiles

S3 profiles

Profile name: Jenkins-S3-Profile

Access key: [redacted]

Secret key: ••

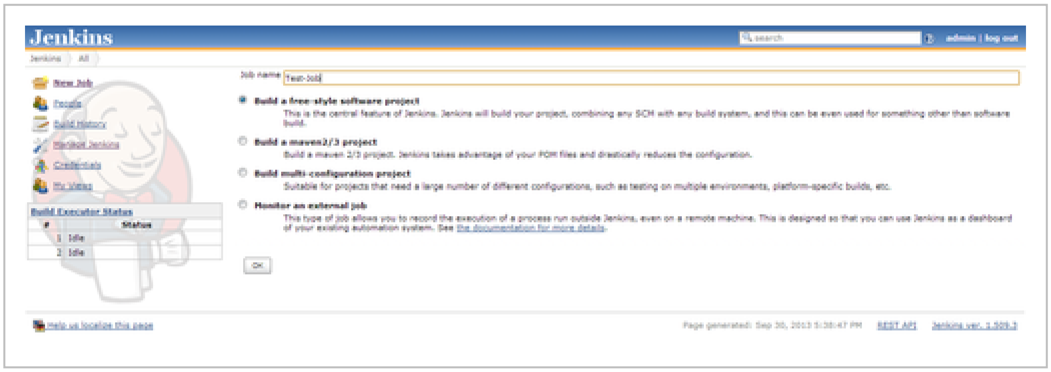
Profiles for publishing to S3 buckets

GitHub Web Hook

○ Let Jenkins auto-manage hook URLs

● Manually manage hook URLs

Once done, create a job under “Build a free-style software project”.



Build Executor Status	
#	Status
1	Idle
2	Idle

Job name: Test-Job

● **Build a free-style software project**
This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.

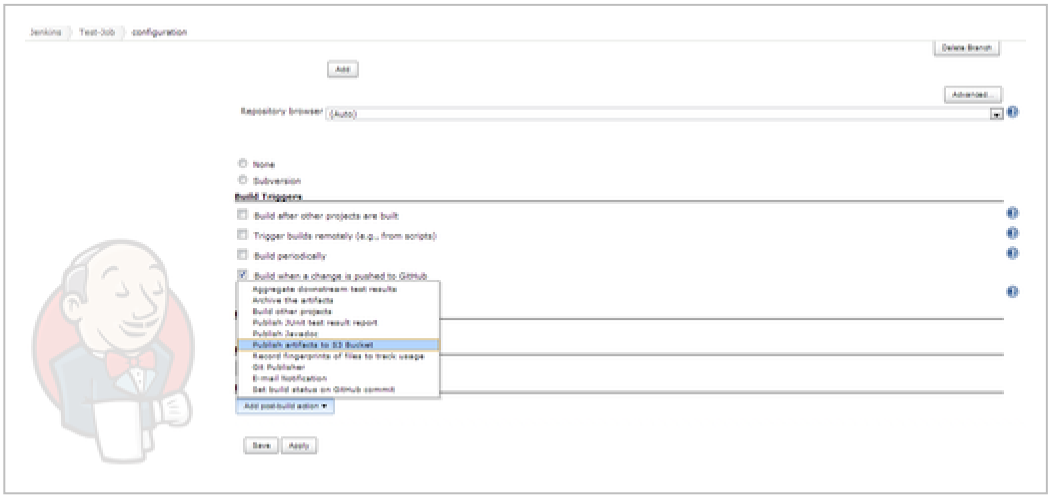
○ **Build a maven2/3 project**
Build a maven 2/3 project. Jenkins takes advantage of your POM files and drastically reduces the configuration.

○ **Build multi-configuration project**
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.

○ **Monitor an external job**
This type of job allows you to record the execution of a process run outside Jenkins, even on a remote machine. This is designed so that you can use Jenkins as a dashboard of your existing automation system. See the documentation for more details.

Page generated: Sep 30, 2013 5:38:47 PM REST API Jenkins ver. 1.509.3

Navigate down to “Post-Build Actions” and click on “Add Post-Build Action” and select “Publish Artifacts to S3 Bucket”.



Jenkins › Test-Job › configuration

Delete Branch
Add
Advanced...

Repository browser: (Auto)

○ None

○ Subversion

Build Triggers

□ Build after other projects are built

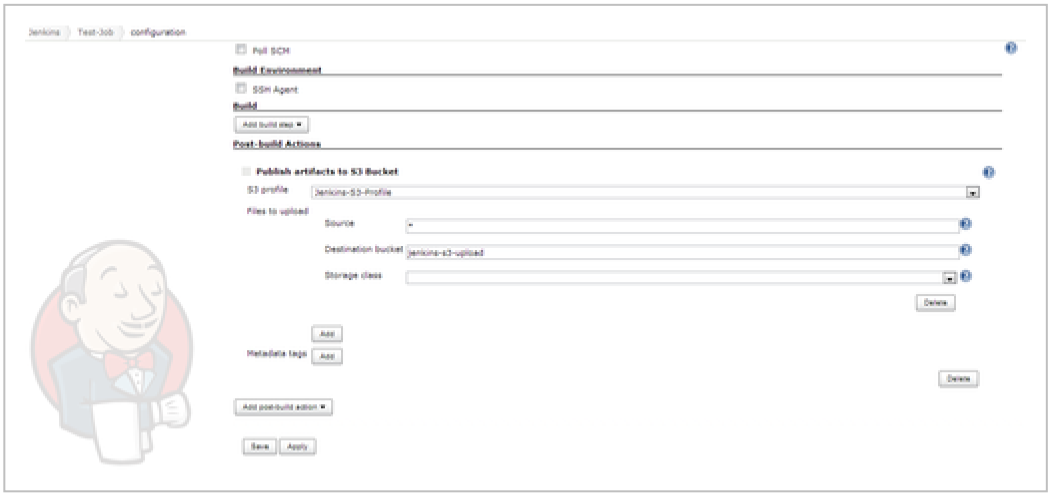
□ Trigger builds remotely (e.g., from scripts)

□ Build periodically

☑ Build when a change is pushed to GitHub

Aggregate downstream test results
Archive the artifacts
Build other projects
Publish JUnit test result report
Publish Javadoc
Publish artifacts to S3 Bucket
Record fingerprints of files to track usage
Git Publisher
E-mail Notification
Set build status on GitHub commit
Add post-build action ▾
Save
Apply

Then, select your S3 profile, pass source of the files to upload (if you want to upload all files, please use “ * ” else provide valid path), destination bucket name.



Jenkins › Test-Job › configuration

□ Poll SCM

Build Environment

□ SSH Agent

Build

Post-build Actions

□ Publish artifacts to S3 Bucket

S3 profile: Jenkins-S3-Profile

Files to upload

Source: *

Destination bucket: jenkins-s3-upload

Storage class:

Metadata tags

Now, click on save and you are done !!

All your build artifacts will get uploaded to Amazon S3 bucket.