UECS2363 SOFTWARE CONSTRUCTION AND CONFIGURATION

Practical 10 : Jenkins and Travis CI (Part I)

A ‘Feedback’ in DevOps means a developer commit changes to project source files, build, test then get the response of feasibility of the changes. Continuous feedback means a developer will build, test and get feedback on every commit that was done. This will tell whether the commit cause any problem to the project development. In other words, developers will not wait until there are numerous commits then only build, test and get feedback and got confused as to which commit cause the problems, if occurred.

An ‘Integration’ in DevOps include the cycle of deployment. Thus, Continuous Integration (CI) imply that each commit of changes to a project’s source files is built, tested and deployed in order to see whether the commit cause any errors to project development AND deployment/production. Continuous Integration is essential, as it allows collaborating developers to know which commit brings error while being tested as well as being deployed (in real application), as the changes might pass testing however paralyse the system when it is deployed.

There are numerous tools available for CI; among them, we will explore **Jenkins** and **Travis CI** within these two weeks.

Preliminaries.

In order to have CI, following tools are needed:

1. A shared source codes repository (Git, Github, Bitbucket)
2. A continuous integration tool/server (Jenkins, Travis CI, Buildbot, Bamboo, etc)

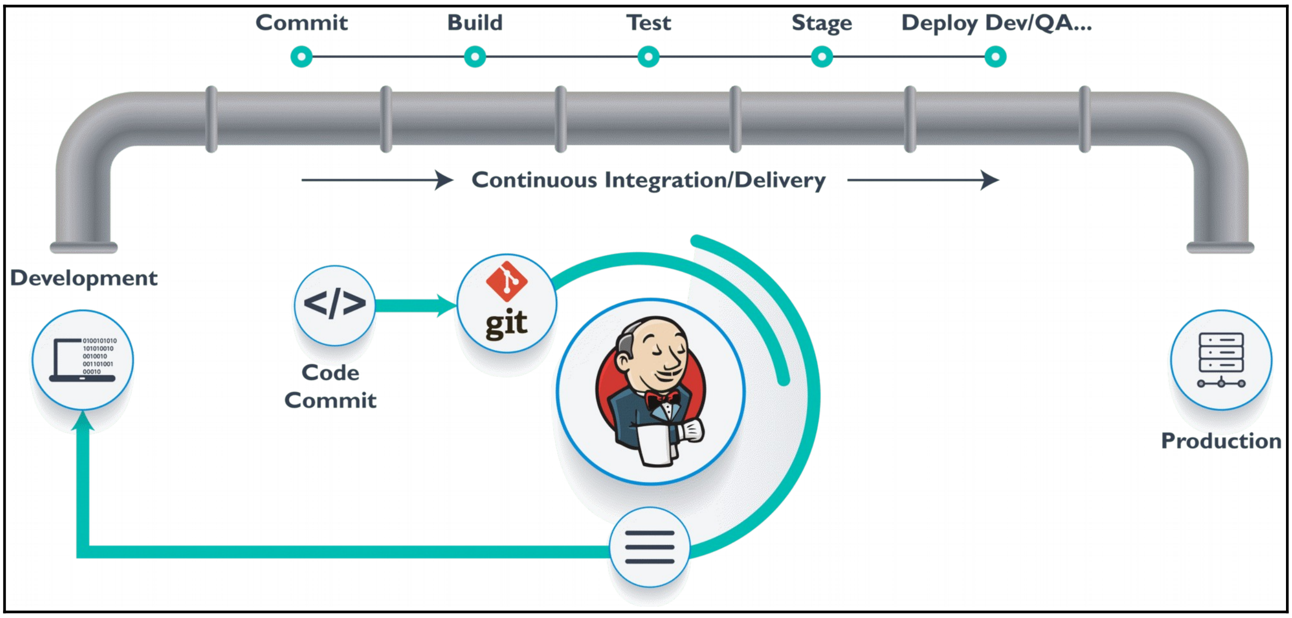


Figure 1: Continuuous Integration Workflow / Implementation.
Source: DevOps Zone www.dzone.com

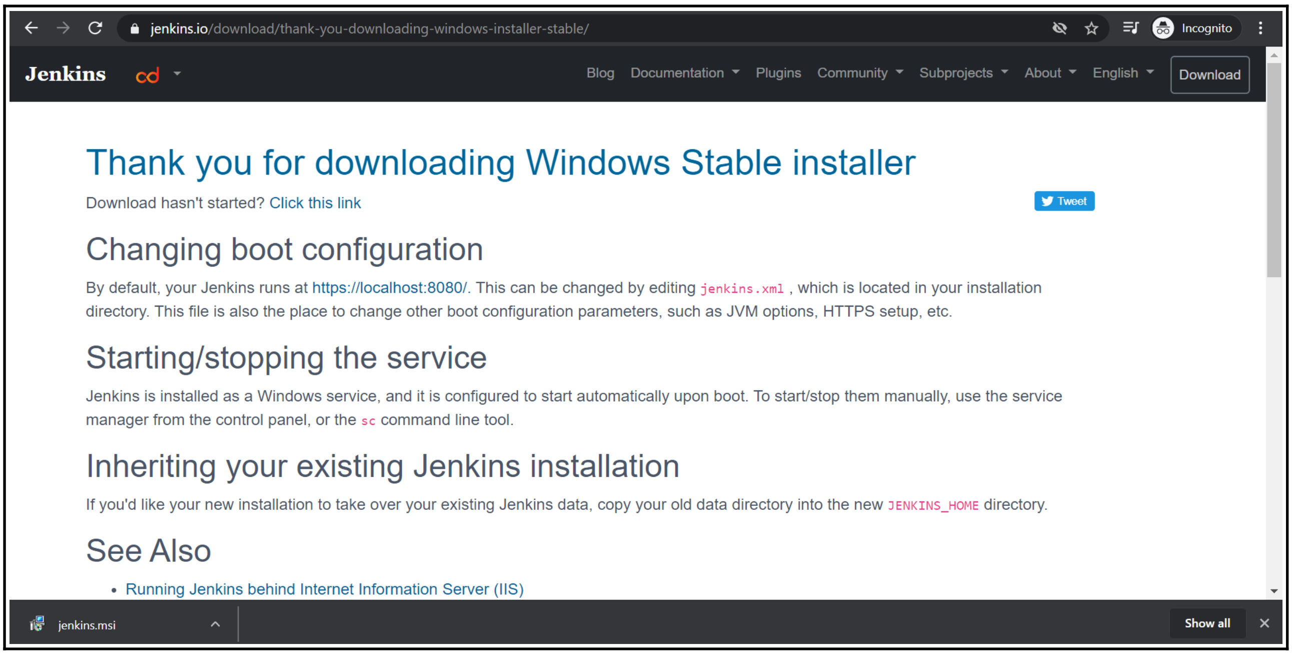
The workflow of CI goes as illustrated in Figure 1; developer commit a change in source codes in the shared cource codes repository, then CI server pull the updated project and build, then return build, test and deployment results.

Jenkins Setup.

Firstly, let's download and install Jenkins at this URL https://www.jenkins.io/download/.
There are three ways of installing Jenkins:

1. Windows / Linux/Unix services
2. Docker integration
3. Generic war file (jdk install using CLI)
4. Install on any Java based web servers – Apache Tomcat (older way of running Jenkins)

We will explore on the first way for Jenkins which is to install using an msi executable to install on Windows using the default installation.



Thank you for downloading Windows Stable installer

Download hasn't started? Click this link

Changing boot configuration

By default, your Jenkins runs at https://localhost:8080/. This can be changed by editing `jenkins.xml` , which is located in your installation directory. This file is also the place to change other boot configuration parameters, such as JVM options, HTTPS setup, etc.

Starting/stopping the service

Jenkins is installed as a Windows service, and it is configured to start automatically upon boot. To start/stop them manually, use the service manager from the control panel, or the `sc` command line tool.

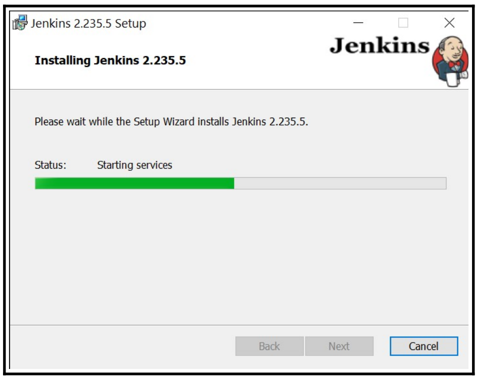
Inheriting your existing Jenkins installation

If you'd like your new installation to take over your existing Jenkins data, copy your old data directory into the new `JENKINS_HOME` directory.

See Also

- Running Jenkins behind Internet Information Server (IIS)

Figure 2: Jenkins msi file download.

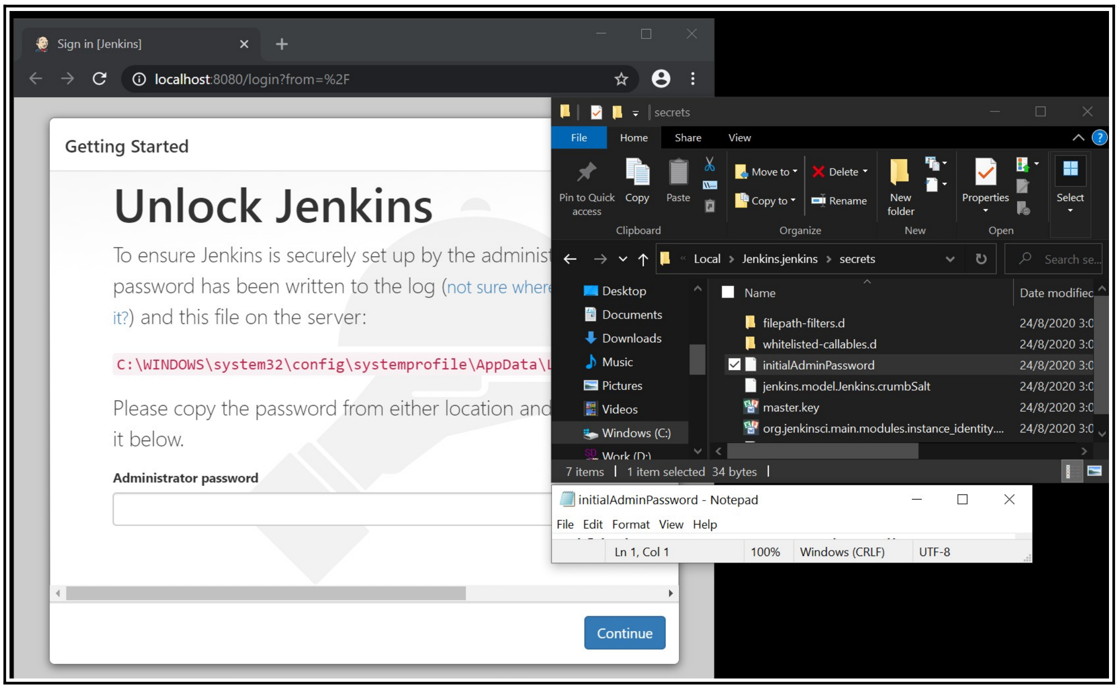


Installing Jenkins 2.235.5

Please wait while the Setup Wizard installs Jenkins 2.235.5.

Status: Starting services

Figure 4: Jenkins installation yet to complete.



Getting Started

Unlock Jenkins

To ensure Jenkins is securely set up by the adminis
password has been written to the log (not sure where
it?) and this file on the server:

`C:\WINDOWS\system32\config\systemprofile\AppData\L`

Please copy the password from either location and
it below.

Administrator password

Name	Date modified
filepath-filters.d	24/8/2020 3:0
whitelisted-callables.d	24/8/2020 3:0
☑ initialAdminPassword	24/8/2020 3:0
jenkins.model.Jenkins.crumbSalt	24/8/2020 3:0
master.key	24/8/2020 3:0
org.jenkinsci.main.modules.instance_identity....	24/8/2020 3:0

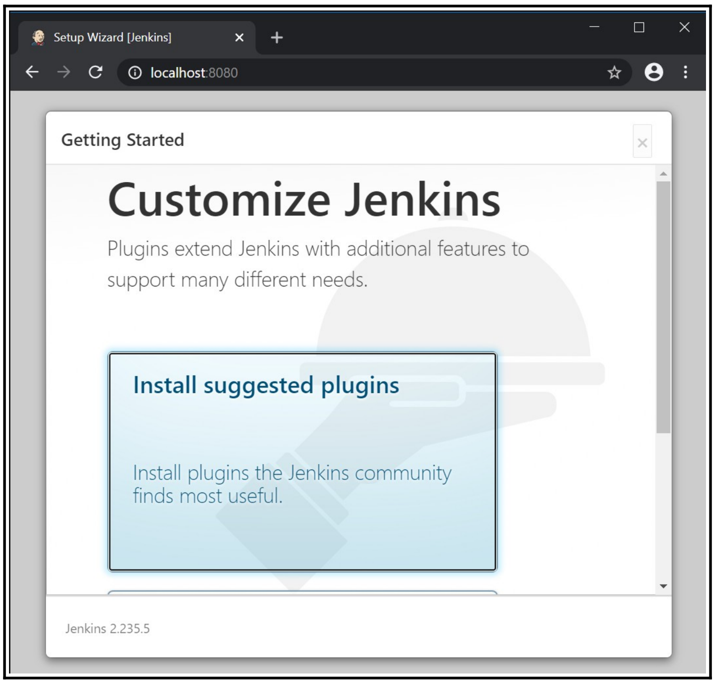
7 items | 1 item selected 34 bytes |

initialAdminPassword - Notepad

Ln 1, Col 1 | 100% | Windows (CRLF) | UTF-8

Figure 5: Complete Jenkins installation by Unlocking Jenkins with instruction provided.

Note: For future login; save the initial password at somewhere you can find easily or change the Jenkins login in the Jenkins installation steps.



Getting Started

Customize Jenkins

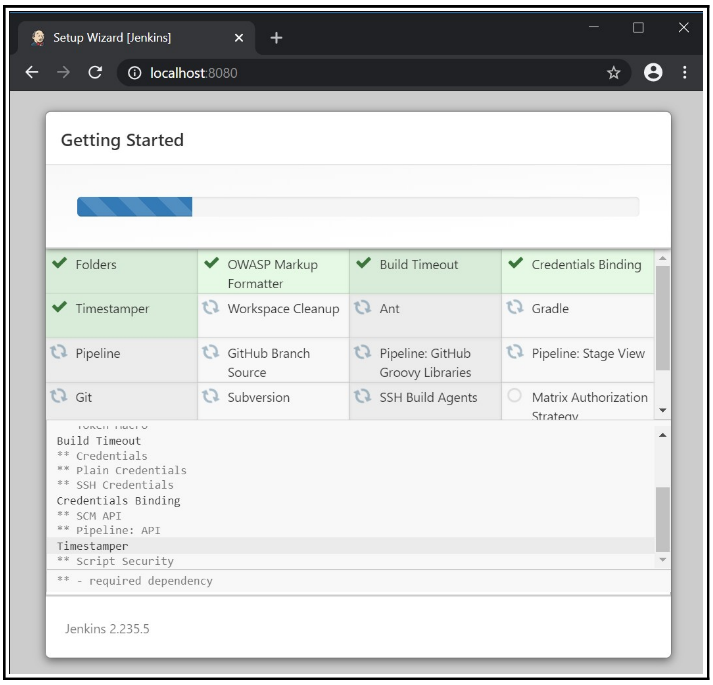
Plugins extend Jenkins with additional features to support many different needs.

Install suggested plugins

Install plugins the Jenkins community finds most useful.

Jenkins 2.235.5

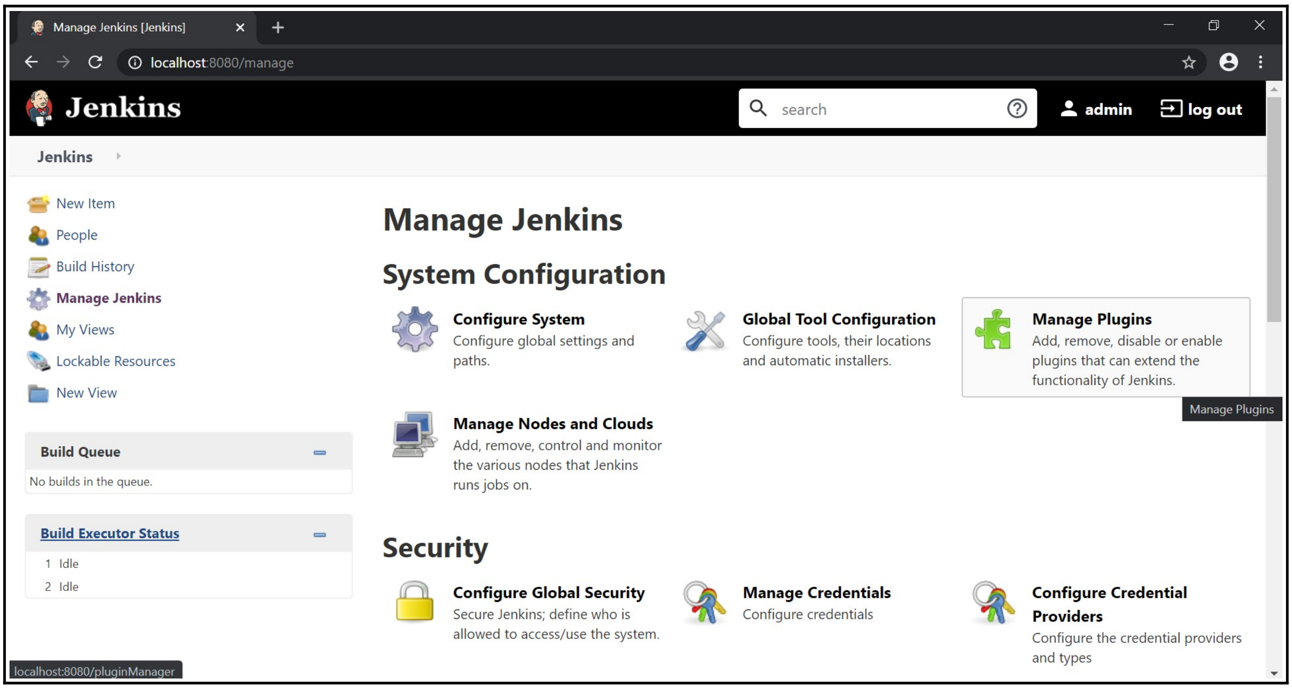
Figure 6: Configure Jenkins by installation of suggested plugins.



```
Build Timeout
** Credentials
** Plain Credentials
** SSH Credentials
Credentials Binding
** SCM API
** Pipeline: API
Timestamper
** Script Security
** - required dependency
```

Figure 7: Jenkins plugins installation.

Upon successful installation, Jenkins will be accesible at the localhost port where it was initially set. Login will always be ‘admin’ with initial password if further setup for Jenkins account is skipped in the customization step. Jenkins plugins can be easily managed through ‘Manage Jenkins’ menu on left navigation pane as illustrated in Figure 8.



No builds in the queue.

1 Idle
2 Idle

Manage Jenkins

System Configuration

Configure System Configure global settings and paths.

Global Tool Configuration Configure tools, their locations and automatic installers.

Manage Plugins Add, remove, disable or enable plugins that can extend the functionality of Jenkins.

Manage Nodes and Clouds Add, remove, control and monitor the various nodes that Jenkins runs jobs on.

Security

Configure Global Security Secure Jenkins; define who is allowed to access/use the system.

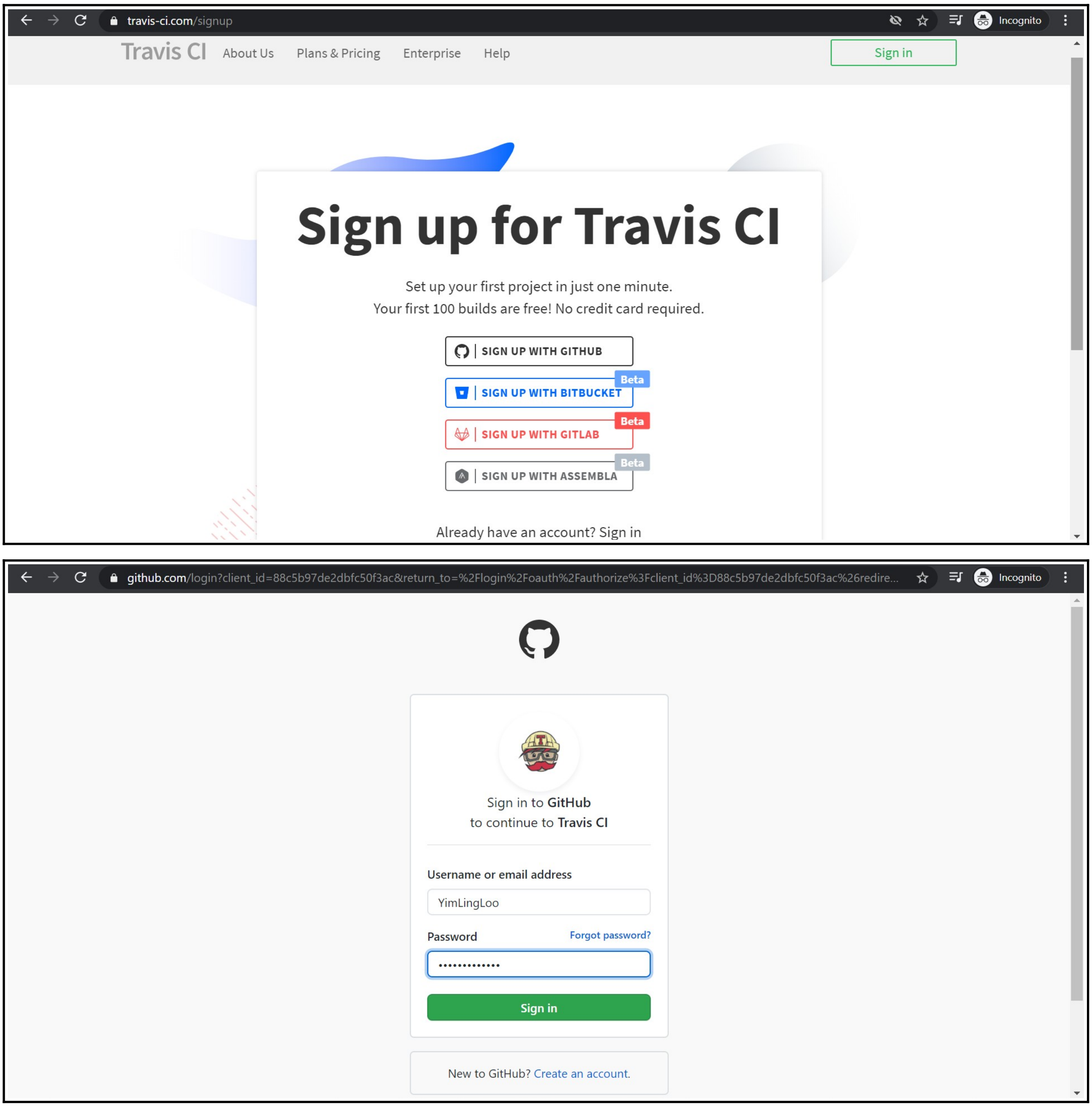
Manage Credentials Configure credentials

Configure Credential Providers Configure the credential providers and types

Figure 8: Jenkins plugins management.

Travis CI Setup.

Go to https://travis-ci.com/ and click in 'Sign up' button. Select the sign up with GitHub as shown in Figure 9.



travis-ci.com/signup

Sign up for Travis CI

Set up your first project in just one minute.
Your first 100 builds are free! No credit card required.

Already have an account? Sign in

github.com/login?client_id=88c5b97de2dbfc50f3ac&return_to=%2Flogin%2Foauth%2Fauthorize%3Fclient_id%3D88c5b97de2dbfc50f3ac%26redire...

Sign in to GitHub
to continue to Travis CI

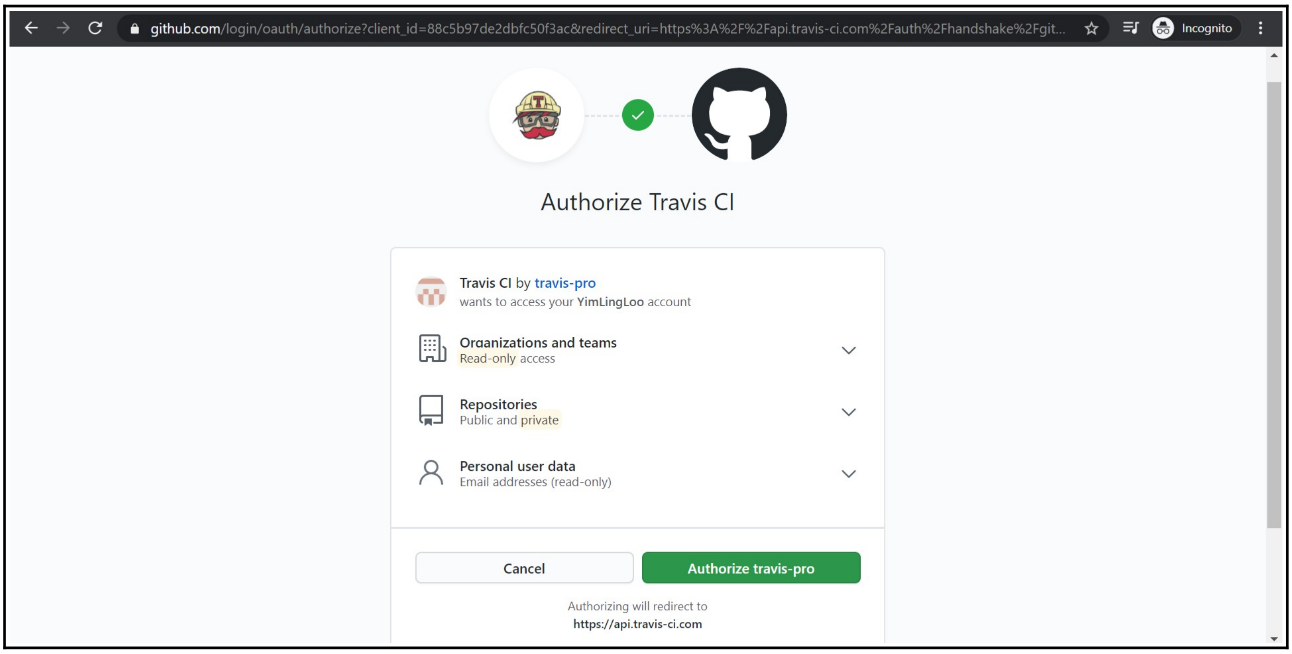
Username or email address: YimLingLoo

Password: ••••••••••••••

New to GitHub? Create an account.

Figure 9: Sign up for Travis CI using Github.

Then, accept the Authorization of Travis CI pro as shown in Figure 10. You'll be redirected to GitHub.



github.com/login/oauth/authorize?client_id=88c5b97de2dbfc50f3ac&redirect_uri=https%3A%2F%2Fapi.travis-ci.com%2Fauth%2Fhandshake%2Fgit...

Authorize Travis CI

Travis CI by travis-pro
wants to access your YimLingLoo account

Organizations and teams
Read-only access

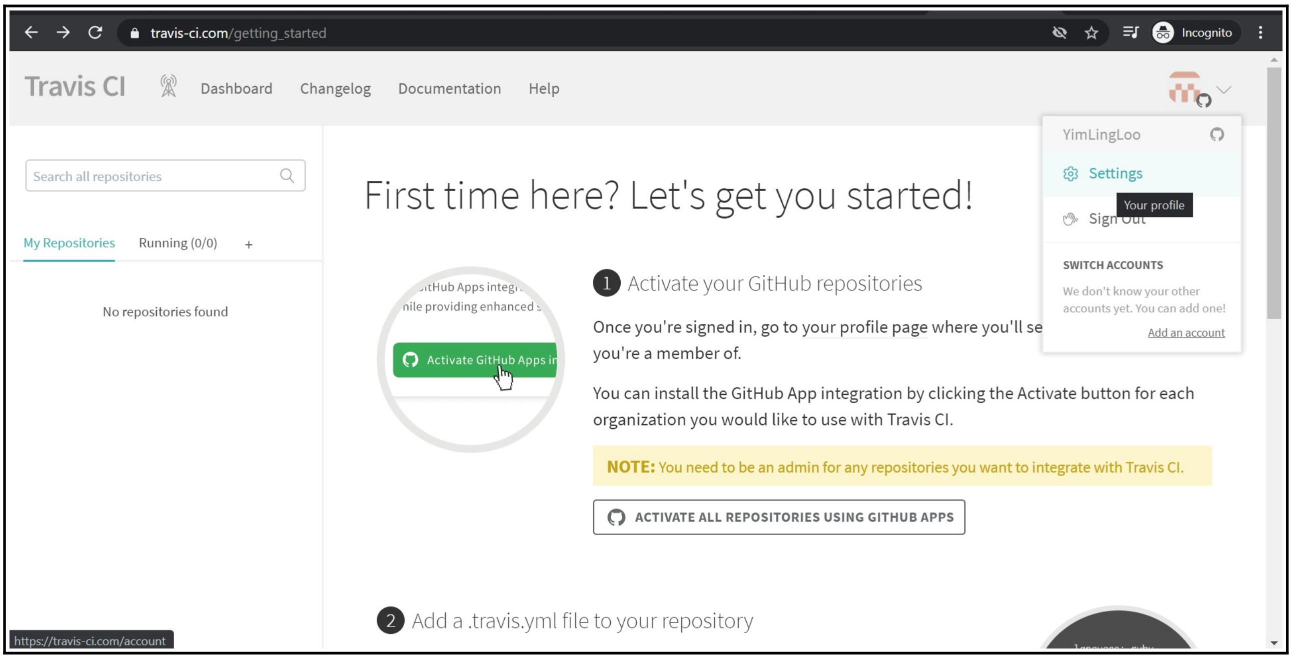
Repositories
Public and private

Personal user data
Email addresses (read-only)

Authorizing will redirect to
https://api.travis-ci.com

Figure 10: Authorize Travis CI in Github.

In order to activate Travis CI authority of managing CI for your repos in Github, click on your Github profile icon and select ‘Settings’ as shown in Figure 11. Select ‘Activate’ to integrate Travis CI using Github Apps.



travis-ci.com/getting_started

First time here? Let's get you started!

1 Activate your GitHub repositories

Once you're signed in, go to your profile page where you'll se
you're a member of.

You can install the GitHub App integration by clicking the Activate button for each organization you would like to use with Travis CI.

NOTE: You need to be an admin for any repositories you want to integrate with Travis CI.

2 Add a .travis.yml file to your repository

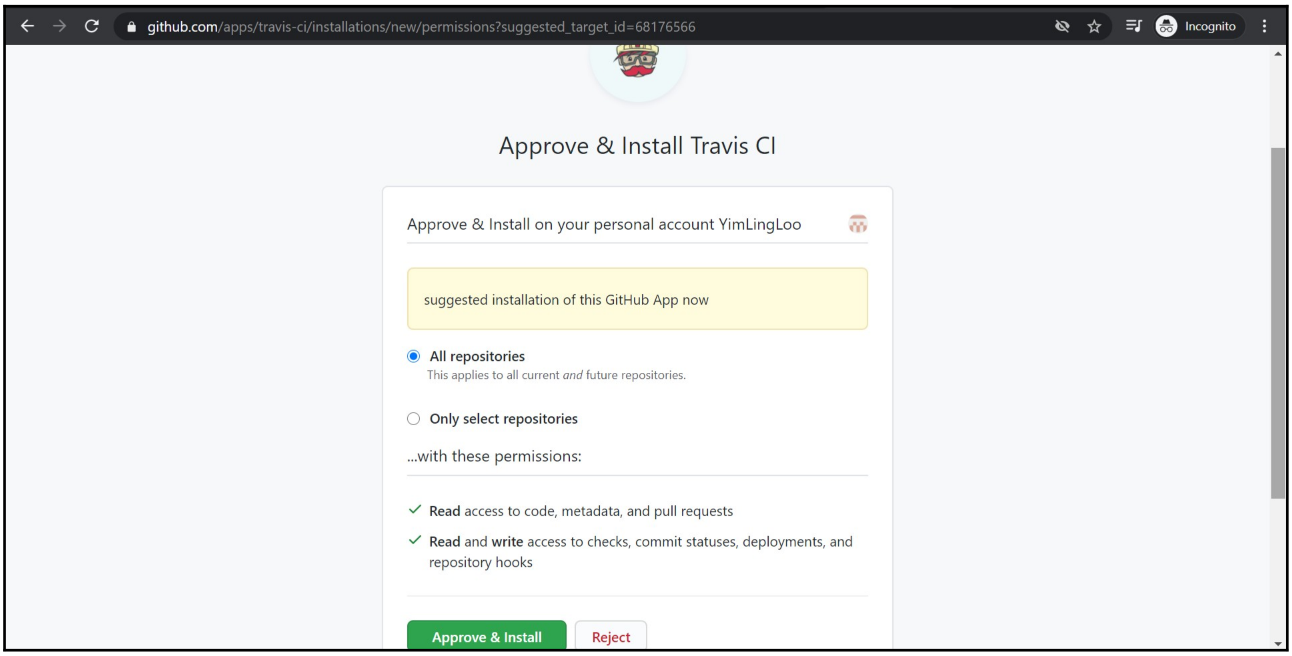
No repositories found

We don't know your other accounts yet. You can add one!

https://travis-ci.com/account

Figure 11: Activate Github Apps integration for Travis CI in Github repo.

Upon choosing ‘Activate’, by choosing ‘All repositories’ then ‘Approve and Install’ as shown in Figure 12 will complete the setup of Travis CI.



github.com/apps/travis-ci/installations/new/permissions?suggested_target_id=68176566

Approve & Install Travis CI

Approve & Install on your personal account YimLingLoo

suggested installation of this GitHub App now

● **All repositories**
This applies to all current *and* future repositories.

○ **Only select repositories**

...with these permissions:

✓ **Read** access to code, metadata, and pull requests

✓ **Read** and **write** access to checks, commit statuses, deployments, and repository hooks

Figure 12: Complete Travis CI setup in Github.