Continuous Integration: Jenkins

Machine Learning in Production / AI Engineering - Recitation 6

Outline

- Continuous Integration (CI)
- CI Pipeline
- Demo
- CI Pipeline Qualities
- Testing ML and CI Pipeline

Continuous Integration

- A sequence of stages through which the system has to go through before it can be deployed; usually followed by continuous deployment stages
- Flow
  - Code commit triggers a new pipeline run
  - Pipeline executes
  - If the CI pipeline passes, CD pipeline starts
- Main goal is to reduce the time taken from code commit to deployment (with CD)
- Another goal is to automate activities (or reduce manual effort as much as possible)

CI Pipeline

- Defined set of stages which run in an automated fashion once triggered
- Pipeline stages:
  - Checkout code → Set up environment → Build code → Static checks → Unit tests → Integration tests → Packaging the software → …
- For machine learning, you may have more stages such as:
  - Data quality check, offline model evaluation, data collection, data cleaning/preprocessing, model serialization, telemetry data collection, etc.
- CI/CD tools: Jenkins, TravisCI, GitHub Actions, etc.

Demo

- Goals:
  - Look at some starter code and initial setup of a CI pipeline for a sample ML system
  - Save you some time (hopefully) in setting up your CI pipeline for Milestone 2
- Contents
  - Sample codebase (https://github.com/cmu-seai/jenkins-demo)
  - Demo-server (http://128.2.205.112:8080/)
  - Jenkins installation and GitHub integration
  - Jenkins pipeline structure
  - Jenkins coverage and plot plugins

Getting Started

Unlock Jenkins

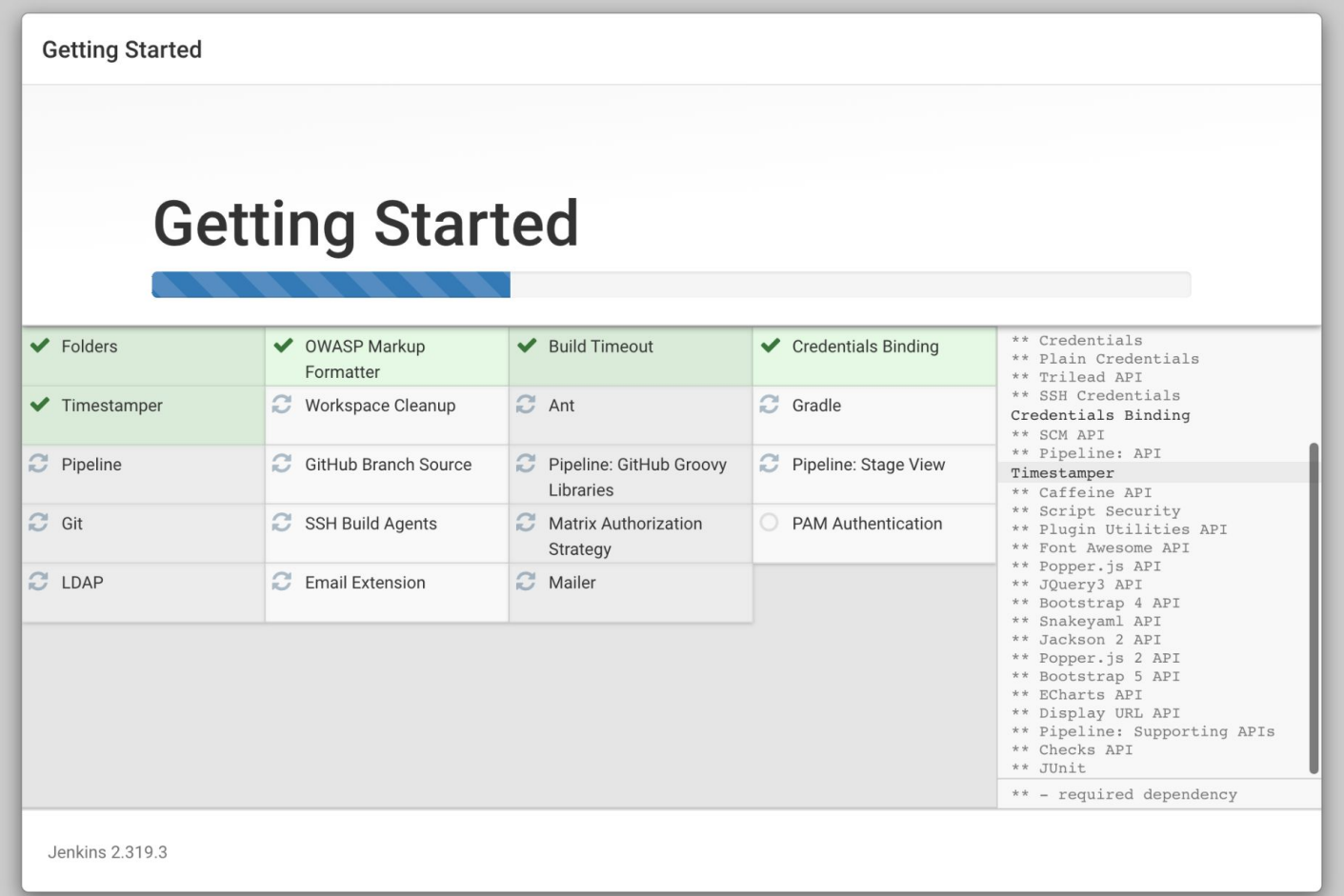
To ensure Jenkins is securely set up by the administrator, a password has been written to the log (**not sure where to find it?**) and this file on the server:

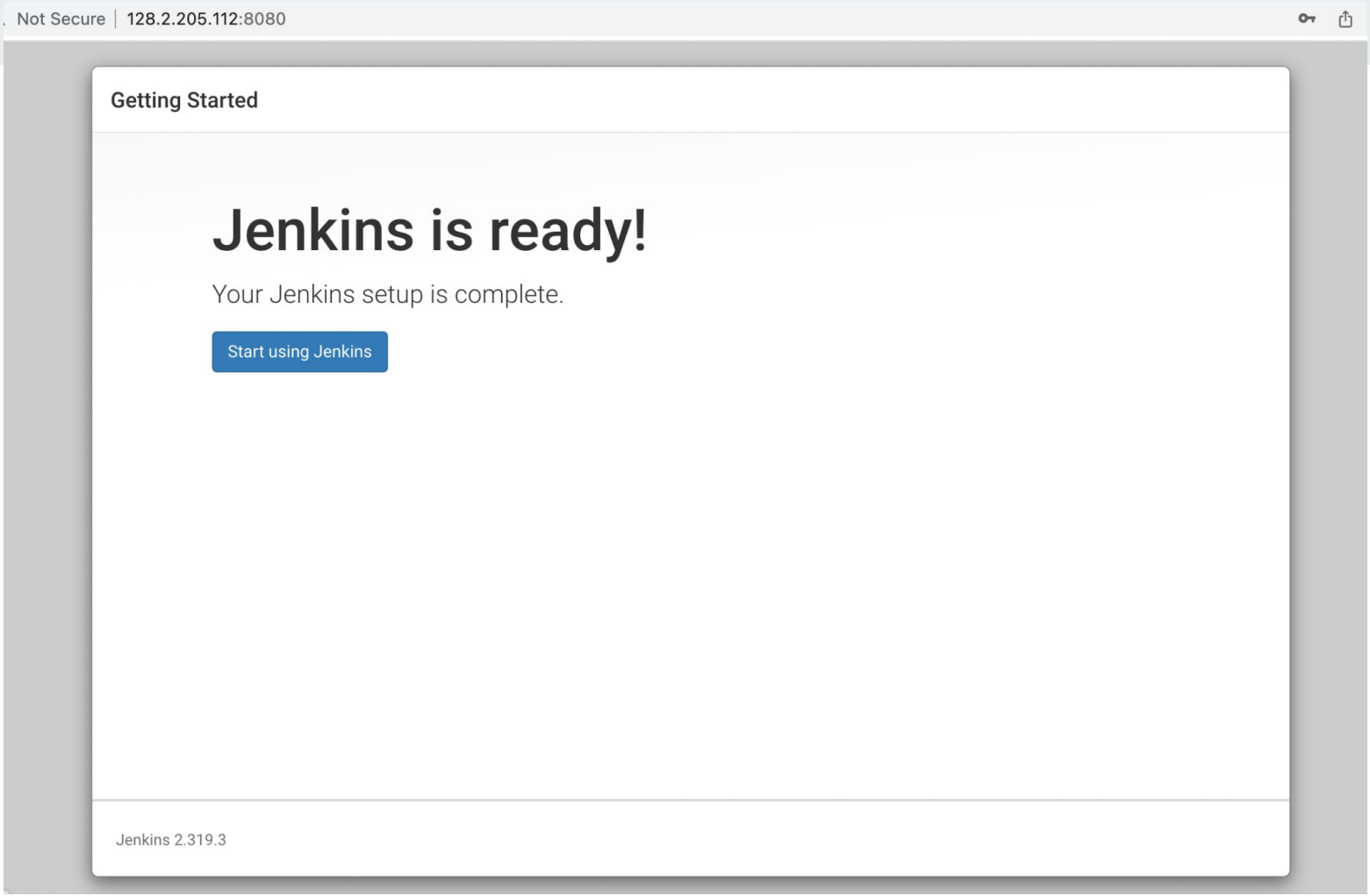
`/var/lib/jenkins/secrets/initialAdminPassword`

Please copy the password from either location and paste it below.

Administrator password

Continue




Not Secure | 128.2.205.112:8080
Getting Started
Jenkins is ready!
Your Jenkins setup is complete.
Start using Jenkins
Jenkins 2.319.3

CI Pipeline Qualities

- Repeatable [consistent results across runs; consecutive runs are independent]
- Fault-tolerant [fail gracefully if any stage fails, ie. system remains operational]
- Correct [performs what is expected of it given some inputs]
- Robust [should be able to handle noise in any inputs the pipeline expects]
- Testable [stages of the pipeline should be independently testable]
- Traceable [should be possible to trace any error to its source quickly]
- Performant [should be possible to move through the pipeline quickly]

Testing ML & CI Pipelines

- Unit tests for independent stages of the machine learning pipeline (automated)
  - Adequacy can be measured in terms of statement/branch coverage, etc.
  - Can use equivalence classes, boundary value analysis, etc. to identify test cases
- Integration tests for APIs (automated + manual)
  - Adequacy can be measured in terms of statement/branch coverage, etc.
  - Can use equivalence classes, boundary value analysis, etc. to identify test cases
  - Mock dependencies
- Manual blackbox tests for the CI pipeline
  - Adequacy can be measured in terms of use cases, nodes in activity/flow diagrams, etc.

Links

- Install Jenkins [https://www.jenkins.io/doc/book/installing/linux/]
- Jenkins plugins [https://plugins.jenkins.io/plot/, https://plugins.jenkins.io/cobertura/]
- Git to Jenkins integration [https://www.blazemeter.com/blog/how-to-integrate-your-github-repository-to-your-jenkins-project]
- Creating a pipeline in Jenkins [https://www.jenkins.io/doc/pipeline/tour/hello-world/]
- Example codebase [https://github.com/cmu-seai/jenkins-demo]
- http://128.2.205.112:8080, Username: admin, password: seaidemo