Ibrahim Diabate
Ming Hoi Lam
Matthew Subido
Instruction

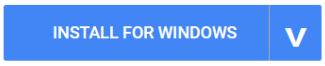
Setup

1. Make sure that Eclipse Oxygen IDE for EE is installed. If not, go to https://www.eclipse.org/downloads/packages/eclipse-ide-java-ee-developers/neon3 to install Eclipse IDE.
2. This instruction is written for use on **Windows Operating System**.
3. In order to properly run this web service, the following components are required:
    a. Google Cloud Tools for Eclipse
    b. Google Cloud SDK
    c. App Engine Java Components

Google Cloud Tool for Eclipse

1. Open Eclipse
2. Go to Help -> Marketplace → Find: “Google Cloud Tools” → click “installed”
3. Restart eclipse

Google Cloud SDK

1. Go to https://cloud.google.com/sdk/
2. Click “Install for Windows” 

3. Follow the 3 steps in the web page to download and setup the sdk

App Engine Java Components

1. After installing Google Cloud SDK, open Google Cloud SDK shell in Windows from the Start Menu.
2. In the shell console, type in `gcloud components install app-engine-java`

Run the Web Service

1. Unzip the source file
2. Open Eclipse and click File → Open Projects from File System…
3. In the Input Source section, choose the web service source file.
4. After Eclipse has detected a project, click Finish.
5. Go to Window → Preferences → Google Cloud Tools, make sure Eclipse has detected the SDK location. If not, manually add the directory that the SDK is installed at the SDK location.
6. Go to Window → Preferences → Java → Installed JREs, make sure jre 1.8 is selected. If jre 1.8 is not shown, manually install Java SE 1.8. Java SE 9 will not work for this web service.

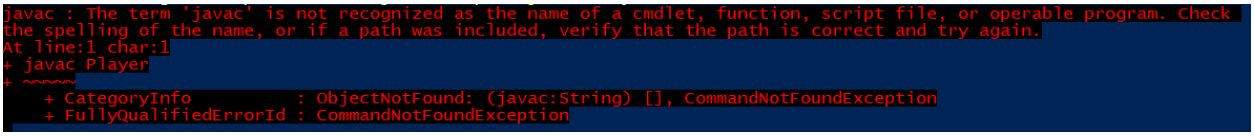
7. In the Project Explorer toolbar, right-click TrustInitializer project → Maven → Update Project…
8. In the Project Explorer panel, right click on the project →Run As → App Engine, then the web service will run.

Test the Web Service

1. Open any web browser (Google Chrome preferred), go to url http://localhost:8080/ or http://localhost:8080/index.html for the login page

Running Player.java

1. Open Command Prompt (cmd) and go to where Player.java is located (using “cd” command”.
2. In the cmd, type in `javac Player.java`. If the followin error exist, type `set “PATH = %PATH%;c:\program files\java\jdk1.8.0_162\bin”`. Or set to where the Java JDK is located. Then run `javac Player.java` again .



3. Type `java Player mySecret myPort myPartnerIP myPartnerPort myRole(A/B) field`. For example, a user has request a share from the web service using **field = 253** and he gets **u = 73**, **v = 78**, **w = 96**. He randomly picks a secret number **10**, and his **port = 1234**, his partner’s **IP = localhost** and **port = 1235**. The user will type in `java Player 10 1234 localhost 1235 A 253`.
4. Then the command line will prompt user to type in all the share that he gets from the web service, which in this case **u = 73**, **v = 78**, **w = 96**.
5. For his partner, he requests his stored share **u = 84**, **v = 55**, **w = 39**.
6. He will follow step 1-4, except his is entering **myPort = 1235**, and **myPartnerPort = 1234**, and another randomly picked number (i.e. **12**). He will type in `java Player 12 1235 localhost 1234 B 253`
7. Then both cmd will start to run and compute the final secret.