Configuring EC2 and WebLogic 12c

# Tools and Setup

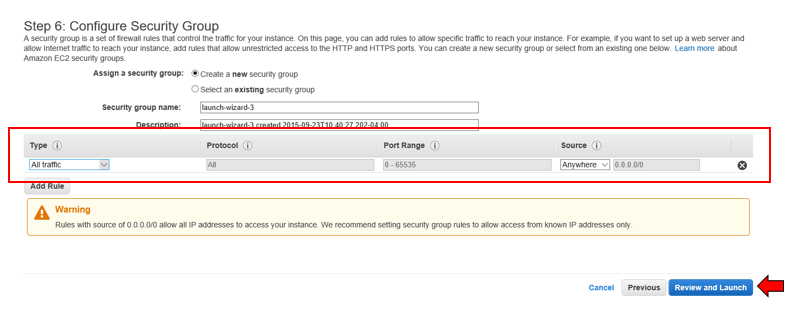
1. Create an account with Amazon Web Services (AWS).
2. Download Putty, PuttyGen, and PSCP at <http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html>

# Create an EC2 Linux Server

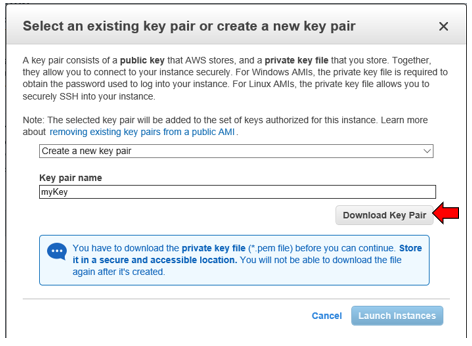
1. Make an EC2 Server Instance by click **Launch Instance** in EC2 Dashboard.



1. Select Amazon Linux AMI.
2. (Optional) Click **next** through configuration pages until step 3 “Configure Instance.”
3. (Required) On step 6 “Configure Security Group,” use a Rule that allows All Traffic on all ports from all IP addresses. Then, click **Review and Launch**. Then, click **Launch** on the next page.
   1. **This is unsecure**. If you will keep this instance running a longer time, please pick My IP as the source—then any traffic outside the office will be disallowed.



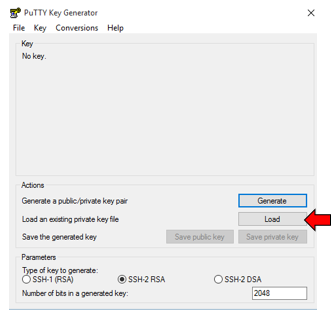
1. Create a new Public/Private Key pair for your instance. This allows us to use SSH to connect to instance.



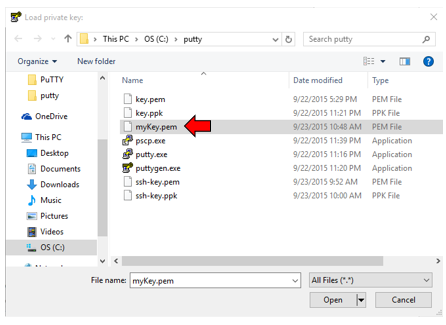
1. **Launch** your instance!

# Connecting to EC2 Server via SSH

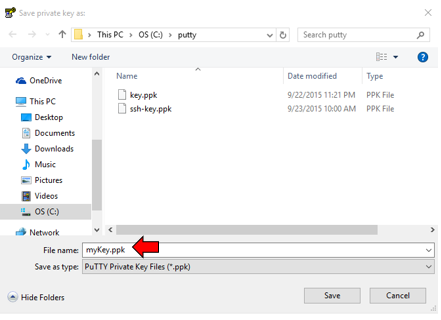
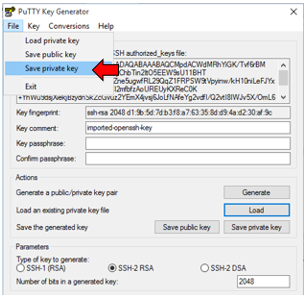
1. PuTTY cannot use PEM files. So, use PuTTYGen to convert it to a PPK file. Click Load.



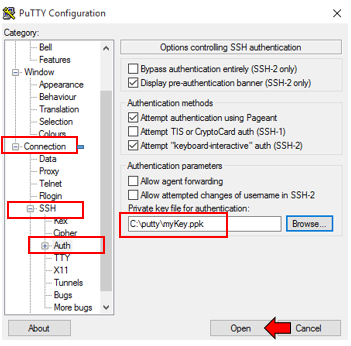
1. Load your newly created PEM key file



1. Save it as a PPK file (use the same name, just change file extension). Ignore any pop-ups along the way.

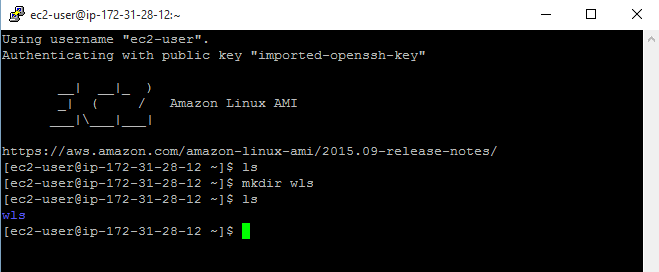


1. Open putty.exe. Select your key in Connection > SSH > Auth. Click Open to connect to EC2 Linux Shell via SSH.



# Installing WebLogic 12c on EC2 Server

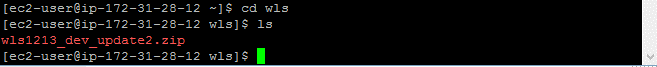
1. Linux Shell: Make a folder for WebLogic.



1. Windows CMD: Move the WebLogic ZIP install up to the EC2 server.



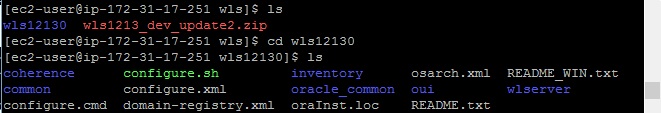
SUCCESS!



1. Then, unzip the WebLogic installation.



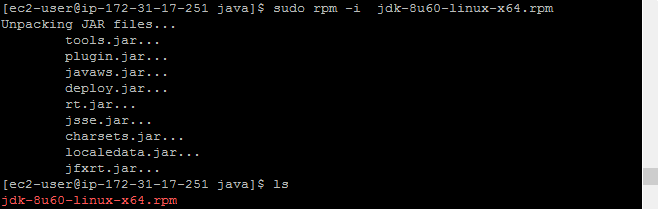
SUCCESS!



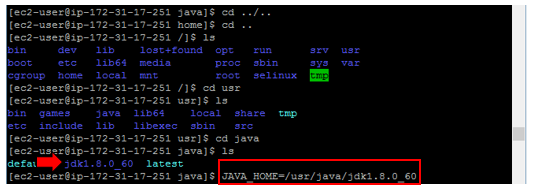
1. Windows CMD: Upload Java Linux install to the EC2 Server. **Use only 64-bit install!**



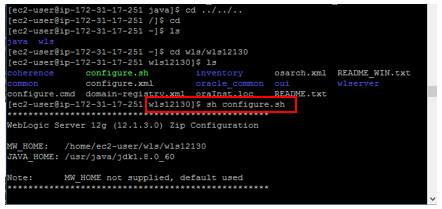
1. Install Java 8 on Linux with rpm command. You must use sudo, since you are not logged in as root.



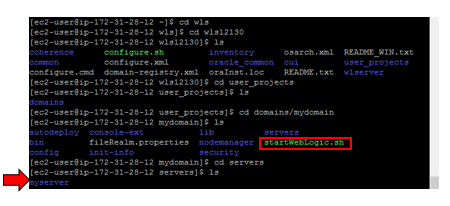
1. Set JAVA\_HOME environment variable. Usually this is /usr/java/jdk{version}. Find the folder to ensure it installed. If this doesn’t work, try:
   1. export JAVA\_HOME=/usr/java/jdk{version}



1. Move to WebLogic folder and configure WebLogic:



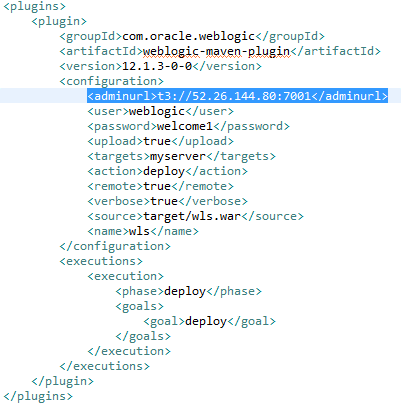
1. WebLogic should start, but if you ever need to start it again, move to your domain and run startWebLogic. You can also see all of your configured servers.



1. Test your WebLogic installation by visiting http://{yourElasticIP}:7001/console

# Configuring Maven Applications for Jenkins Deployment

1. Configure your project’s POM file to use Oracle-Maven-Sync to point to your EC2 server. For more details on configuring your local machine to use WebLogic Maven plugin, consult further documentation.



1. You can now use Jenkins to build your project and deploy to EC2 Server!! ☺