

Uni-Hood Report

Appendix A: HOWTOs

Daniel Carnovale

23/04/2017

Contents

Amazon "Free Tier" Account Creation.....	3
Creating and Associating Elastic IP to Instance.....	4
AWS Console for Mobile	5
TortoiseGit for Windows.....	6
OctoDroid.....	7
SSH KeyPair Generation	8
Amazon EC2 Instance Creation	9
Setting up Putty and Encrypted PPK File	10
Stopping and Restarting the EC2 Instance.....	11
Connecting to the Instance	12
Creating / Removing EBS Volumes.....	13
Managing EBS Volumes.....	14
Installing and Configuring Postgres, setting up DB Owner	15
Installing Apache and PHP, and Managing the WebServer	16
Backup/Restore Test Data	17

Amazon “Free Tier” Account Creation

NOTE: You may already have an Amazon “shopping” account, used for purchasing books, DVDs etc. It’s possible to let this account to be the basis for your usage of Amazon Web Services, bearing in mind that any billing for the project will be sent to your personal account. Billing refers to anything beyond Amazon’s “free tier”. If you prefer billing on the project to be kept separate (e.g. Academic / corporate requirements) then best practice is to create a new Amazon account first.

1. Create a new Amazon account if necessary
2. Tie the Amazon account to the Amazon Web Services Free Tier, by clicking “Create a Free Account” here: <https://aws.amazon.com/start-now/>
3. After following the standard prompts, you will be notified that services will be available soon.
4. Once services are available, make sure you are signed into the Web Console from the “Sign In to the Console” link here: <https://aws.amazon.com/console/>
5. Try to use some services, the easiest test is the “S3” service to share very large files privately or publicly

Creating and Associating Elastic IP to Instance

1. Log into the Amazon Web Services Management Console <https://aws.amazon.com/console/>
2. Obtain a new Elastic IP Address by clicking ""Networking & Security" on the left, click on "Elastic IPs" then click "Allocate New Address" (ela-1).
3. Click "Associate Address" filling in the Instance and Private IP fields which should present suggestions upon clicking them. Click "Associate" (ela-2)
4. The elastic IP is now associated with the instance and can always be used even when the instance is restarted

AWS Console for Mobile

1. Download the AWS Console Mobile Application which can be found on the Google Play store: <https://play.google.com/store/apps/details?id=com.amazon.aws.console.mobile> (aws-1)
2. Open on your mobile and sign in as usual
3. You should see a similar screen to (aws-2), from which you should click the “EC2 Instances” row
4. From here, you can manage your instance/s through clicking its row, and the appropriate option. Or simply to review its status (aws-3)

TortoiseGit for Windows

1. Download the application from <https://tortoisegit.org/download/> (tor-1) and make sure you choose the right architecture as highlighted in the screenshot
2. Installation should go smoothly, go with all default installation options
3. If you try to run the application this dialog will show (tor-2) making clear that its usage is as a context menu
4. On github.com, create a new account if you have not already done so and then create a new repository (tor-3)
5. Name it something meaningful, initialise it with a readme (as this is a fresh repo) and continue (tor-4)
6. Click Clone or download, and copy this link for use with TortoiseGit (tor-5)
7. Right click an empty area on a drive in Windows Explorer, and click “Git Clone...” (tor-6)
8. The url and details should already be filled out, so just click “ok” (tor-7)
9. It should be cloned successfully, and ready to have files pushed (tor-8)
10. To test committing and pushing a file, create a new text document in the repository (tor-9)
11. Right-click the Git repo folder, and click “Git Commit -> master...” (tor-10)
12. Type a meaningful message, check All files, select the “Commit & Push” option from the drop down before you click that button. Add the author date if you wish (tor-11)
13. This should complete successfully, and you can also manage the Git Repo on Github.com (tor-12 & tor-13)

OctoDroid

1. Download the OctoDroid for GitHub Mobile Application which can be found on the Google Play store: https://play.google.com/store/apps/details?id=com.gh4a&hl=en_GB (oct-1)
2. Open on your mobile and sign in as usual
3. Select your repository for viewing / modification
4. From here, you can manage your instance/s through clicking its row, and the appropriate option. Or simply to review its status (oct-2)
5. The commit trail here is very useful as an example to quickly check if your latest commit has gone through (oct-3)

SSH KeyPair Generation

1. Log into the Amazon Management Console <https://aws.amazon.com/console>
2. Choose EC2 from the high-level list of services under the “Compute” Category (key-1)
3. Click Key Pairs on the left-hand menu
4. Click “Create Key Pair”
5. Choose a simple new Key Pair name, typically named after yourself (key-2)
6. It will automatically download this as a .pem file, used in the later steps

Amazon EC2 Instance Creation

1. Click Launch Instance on the top-left of the Amazon Management Console (EC2).
2. For the purposes of this project, Amazon Linux 64-bit should be selected (indicated as part of the Free-Tier program) (EC2-1).
3. The 1GB t2 micro memory instance type should be selected (EC2-2).
4. Leave Instance Details as default, as any changes here could bring about complications or uncertainty in the future when dealing with network details and such (EC2-3).
5. Change the root storage disk size to 30 gb. This should be more than enough space for the simple needs we require, and is of course also eligible for the Free Tier (EC2-4).
6. At this point you are able to name or “Tag” your instance. There are several other tags available to set here though the only one to be concerned with for our purposes is the Name tag. I have called my instance “vlinux” but any other simple name will also be appropriate (EC2-5).
7. Choose the “Select an existing Security Group” radio button, and then select the only group there (named default). (EC2-6).
8. Proceed to launch the instance and create a new key pair, at the same time assigning it a name. Don’t forget to click Download Key Pair here! You can’t download it anywhere else or call up this dialog box again. (EC2-7 & EC2-8).

Setting up Putty and Encrypted PPK File

1. Download and Install PuTTY from this site:
<http://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html> (put-1) using the full msi Windows installer package for your current system architecture
2. Open PuTTYGen and Load in your .pem file downloaded during “SSH KeyPair Generation” by clicking “Load” highlighted in (put-2)
3. Invent a memorable passphrase and enter it, and again to confirm it and finally generate a .ppk PUTTY Private Key file from the pem file by clicking “Generate” (put-3)
4. Click Save private key and you will be prompted to choose a location to save this key to, I suggest saving it to the same place as the pem file.
5. Add PPK file to the Windows Startup folder by simply dragging it in, the easiest way to access this folder is to paste “%appdata%\Microsoft\Windows\Start Menu\Programs\Startup” into the address bar of Windows Explorer (put-4)
6. Double click this file to test (also executed on startup), enter passphrase
7. PuTTY should now be authenticated with this file, and you can proceed to connect to any EC2 instance which was activated using the SSH keypair corresponding to the .pem file

Stopping and Restarting the EC2 Instance

1. Optionally download and install the AWS Console Mobile app, I have found this is more responsive than desktop in terms of managing the instance state. Link to app on Google Play Store:
https://play.google.com/store/apps/details?id=com.amazon.aws.console.mobile&hl=en_GB
2. Log into your AWS account either on Mobile or Desktop into Management Console. Desktop url: <https://aws.amazon.com/console>
3. Enter the EC2 Instances sub-menu.
4. Select the instance(s) you want to stop.
5. Tap the Stop button, the instance will now perform a Shut Down
6. Similarly, you can tap the Start button if the instance was stopped, in effect having restarted the instance.
7. At any point, call up Instance Settings > Get System Log to verify the state of the instance console

Connecting to the Instance

1. On the instance list on the EC2 page, select your instance.
2. On the description tag below, copy the public IP number (con-1).
3. Open Putty and feel free to save this as a new session: right-click the tray icon and click new session.
4. Paste the ip into the Host Name field, click on Data under the Connection category on the left column and enter ec2-user into the Auto-login username.
5. Click back on the session tab and click save but do not open yet.
6. Right click on the tray icon again and click add key, and select the ppk file you downloaded earlier.
7. Right click on the tray icon once more and click Saved Sessions > (your sessions name). The connection should be established.
8. To perform admin tasks, type “sudo -s”
9. To overview all available packages, whether installed or not installed, type “yum list | more”

Creating / Removing EBS Volumes

1. Launch this instance, and then click Volumes under the ELASTIC BLOCK STORE category on the left column. Click Create Volume on the top-left corner of this window. (EC2-9).
2. For this volume, choose the same availability zone as the current root drive is allocated to. (EC2-10).
3. Right click this new volume, and click Attach Volume. (EC2-11).
4. Click on the instance field, it should show your instance id as the only item in the list that pops up. Note the Device field value here, this is important. (EC2-12).

Managing EBS Volumes

1. You will first need to make a new file system with the location of the drive (sdf in my case, also displayed in the details of the drive when clicked on in the AWS window).

In the terminal, type “mkfs /dev/sdf” where sdf is the device name field from step 12.

Make a new directory which maps to this drive, I called mine “bigdata”.

- mkdir /mnt/bigdata
- mount /dev/sdf /mnt/bigdata

Check disk space using the “df” command.

Verify that the 100gb extra EBS volume has been mounted on the directory you specified above. You will notice this in the same output from the “df” command, as it also displays the size of each drive as well as the directory they are mounted on.(EC2-16).

2. To confirm the drive can be written to, a simple test can be performed. You can change the working directory to the one which the drive is mounted on, list its contents, write the current date to a new file called “x”, output the contents of this file and then list the contents of the folder. The screenshot outlines how to do this. (EC2-17).

Installing and Configuring Postgres, setting up DB Owner

1. `sudo -s` # to become root
2. `yum install -y postgresql8-server`
3. `service postgresql initdb`
4. `chkconfig postgresql on`
5. `su - postgres`
6. `cd /var/lib/pgsql/data`
7. # edit the file `pg_hba.conf`
8. # duplicate the line with "local all all ident"
9. # the first line should have its method set to md5
10. # comment out the entries whose connection type is "host"
11. `ctrl+d` # to log out of postgres user
12. `service postgresql start`
13. `su - postgres`
14. `psql` # this opens up the PostgreSQL command line SQL interpreter
15. `alter user postgres with password 'x';` # supply your own secure password here in place of x
16. `create user unihood with password 'unihood';` # similarly here
17. `create database unihood;`
18. `grant all privileges on database unihood to unihood;`
19. `ctrl+d` # to log out of postgres user
20. `ctrl+d` # again to log out of root
21. # you should now be back to ec2-user
22. `psql -U unihood unihood` # supply the suitable password for the unihood database user (first 'unihood' is username and second 'unihood' is database)
23. # if you see a prompt similar to `unihood=>` you now have a valid postgres database owner

Screenshots: pak-1 shows the result of a `yum-list` command listing `postgreSQL92`. Note that we now recommend falling back to `postgreSQL8` for compatibility with `phpPgAdmin`. pak-2 shows a typical output from the `initDB` step

Installing Apache and PHP, and Managing the WebServer

1. Ensure PostgreSQL is installed and running, type “service postgresql status” to report the current status of the postgresql service
2. `yum install -y phpPgAdmin` # this will install phpPgAdmin, php itself and the Apache webserver as well as many required library packages.
3. Once installed, the httpd service (i.e. Apache) must be started in order to use the management console. Type “service httpd start” to start it. You can verify it is running through observing the output of “service httpd status”.
4. Go to the URL “(instance ip)/phpPgAdmin”. You should be presented with the management console.
5. Click on the left “PostgreSQL” and log in with the DB user you set up previously.
6. The database should be listed, expand it and proceed to: Schemas > public > Tables categories. Tables can be managed here for the web app.

Backup/Restore Test Data

1. Click Export on the top-right of the phpPgAdmin management console.
2. Select Structure and data, Format : SQL and the Download option.
3. Click Export, and save the dump file somewhere safe.
4. To restore, DROP THE DATABASE FIRST (bak-1) and re-create, select the database from the tree controller on the left, click the SQL tab at the top, then "Choose file" and then locate and open the dump file you saved. (bak-2)
5. Not only will test data be restored, but the whole database including tables, their structure, foreign keys and so on.