Developer Setup

Goals

This module describes how to setup a developer environment. You will use this environment for the Lab exercises.

At the end of this module you will be able to:

- Install and Configure a single server development cluster in
  - AWS
  - your own computer
- Manage your development cluster


Server Installation

Server options

- Install on your own server.
  http://www.aerospike.com/docs/operations/install/
  - Windows
  - OSX
  - Linux
- Amazon Web Server instance (AWS)

Own Computer: Windows

Follow the instructions at:

- http://www.aerospike.com/docs/operations/install/vagrant/win/

Create an Aerospike Virtual Machine

1. Install Git for windows
2. Create an Aerospike Virtual Machine
3. Initialize the Aerospike Virtual Machine
4. Start Aerospike and the AMC

Own Computer: OSX

Follow the instructions at:

- http://www.aerospike.com/docs/operations/install/vagrant/mac/

Create an Aerospike Virtual Machine

1. Create an Aerospike Directory
2. Initialize the Aerospike Virtual Machine
3. Start Aerospike and AMC

Own computer: Linux

To install the software you must have root/sudo privileges

- Download the server software

```
wget -O aerospike-server.tgz http://www.aerospike.com/download/server/latest/artifact/el6
```

- Download the Aerospike Monitoring software

```
wget -O aerospike-amc.rpm http://www.aerospike.com/download/amc/latest/artifact/el6
```

- Install the AMC
  - `sudo rpm -ivh aerospike-amc.rpm`
- Install the server
  - `tar xvf aerospike-server.tgz`
  - `cd aerospike-server-community-<version>`
  - `sudo ./asinstall`

The URLs for the latest version remain static, so these URLs will get the latest version of the both the database and the Aerospike Management Console.

It is NOT required to install the AMC as the same node as the Aerospike Server, but it is useful to install it here for the class.

Install on your own Computer

Follow the instructions at this URL to install Aerospike on your own Machine

- http://www.aerospike.com/download/server/latest/

Install

- Aerospike server
- Aerospike tools
- Aerospike Management Console

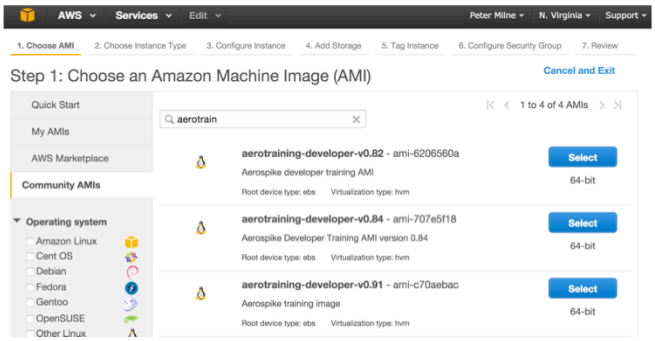
AWS Aerospike Server

For this course, Aerospike has prepared an Amazon EC2 instance for your use.

AMI: "**`aerotraining-developer-vX.XX`**"

This instance has the following pre-loaded:

- Aerospike Java client
- Aerospike Python client
- Aerospike C Aerospike data loader
- Oracle JDK 1.7
- maven
- lua
- git



You may not need all of these, but they are there for your convenience.

How To Log Into The AWS Instance

You will use the IP address for your own AWS instance to use with the training module.

Log into the server:

- Mac/Linux

  `ssh aerotraining@<ip_address>`

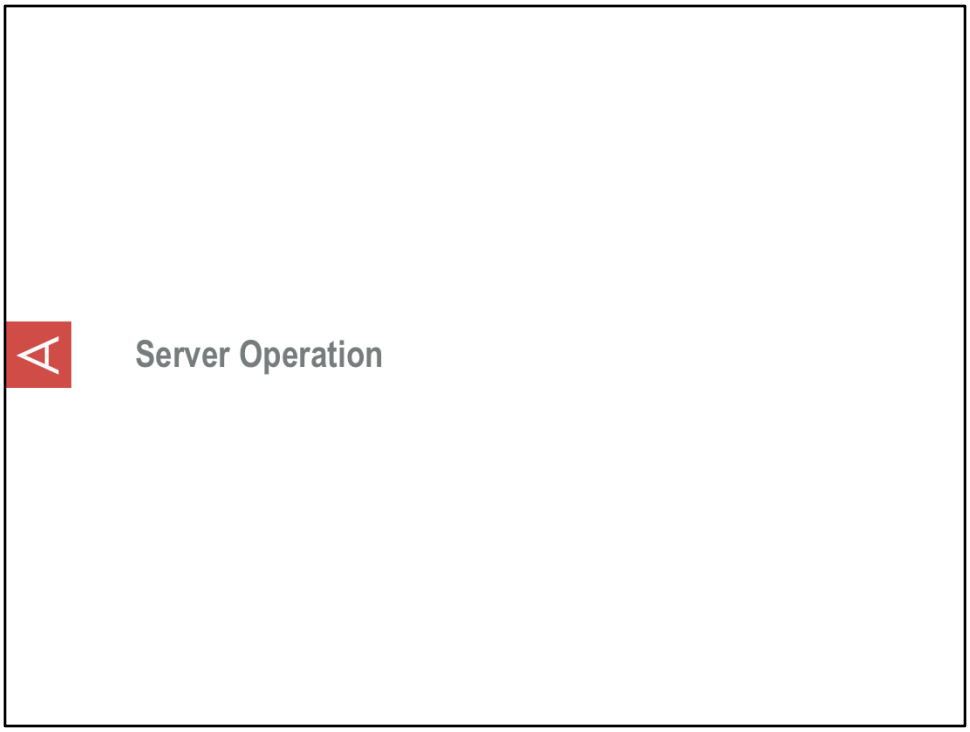
- Windows PC
  - You can use a tool such as putty to ssh login to the server:
  - http://www.chiark.greenend.org.uk/~sgtatham/putty/

The username/password is aerotraining/aerotraining.

You each should have your own IP address to log into your own instance. You can use whichever tool you are comfortable with to log in. You will not need to have a good knowledge of Linux for this class.

Server IP Address

Write down your server IP address


Server Operation

Starting And Stopping Aerospike Server

Controlling the server requires you to be root or have sudo privileges, which the aerotraining user has.

Start server

```
sudo service aerospike start
```

Check on server status

```
sudo service aerospike status
```

Stop server

```
sudo service aerospike stop
```

Restart server

```
sudo service aerospike restart
```

Starting and Stopping the AMC Server

The Aerospike Management Console (AMC) is used to see how the server is doing.

Start server

```
sudo service amc start
```

Check on server status

```
sudo service amc status
```

Stop server

```
sudo service amc stop
```

Restart server

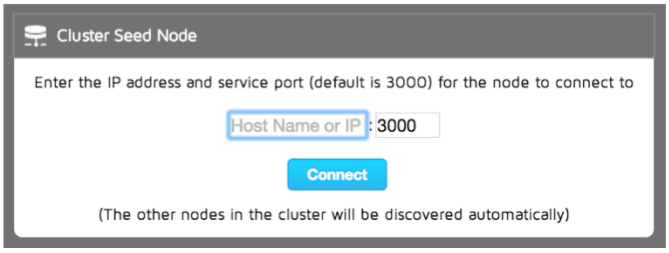
```
sudo service amc restart
```

Aerospike Management Console

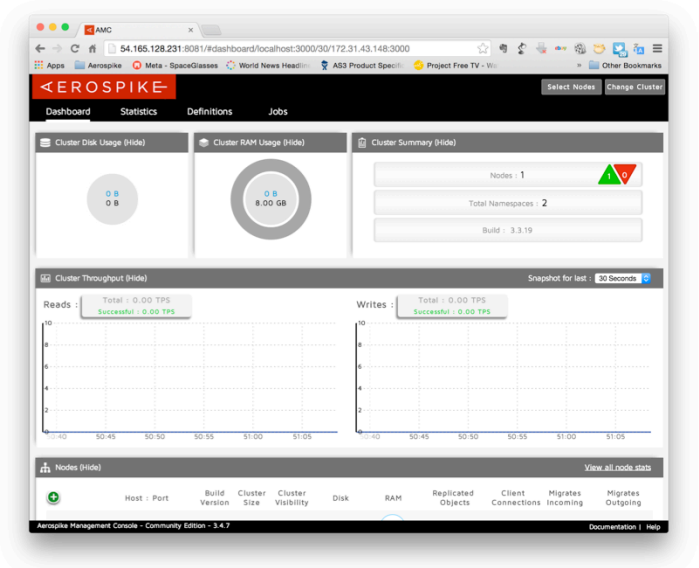
Connect to the AMC using

- http://<your ip address>:8081

You will be prompted for a node address in the cluster, enter your IP address:



AMC


AMC
54.165.128.231:8081/#dashboard/localhost:3000/30/172.31.43.148:3000
AEROSPIKE
Select Nodes
Change Cluster
Dashboard
Statistics
Definitions
Jobs
Cluster Disk Usage (Hide)
Cluster RAM Usage (Hide)
Cluster Summary (Hide)
Nodes : 1
Total Namespaces : 2
Build : 3.3.19
Cluster Throughput (Hide)
Snapshot for last : 30 Seconds
Reads :
Writes :
Nodes (Hide)
View all node stats
Host : Port
Build Version
Cluster Size
Cluster Visibility
Disk
RAM
Replicated Objects
Client Connections
Migrates Incoming
Migrates Outgoing
Aerospike Management Console - Community Edition - 3.4.7
Documentation | Help

Logs

It is always helpful to be able to look at the logs of the server. You MUST have root/sudo privileges to see the logs in the default locations:

```
/var/log/aerospike/aerospike.log
/var/log/aerospike/udf.log
```

It is often useful to keep a window open with a tail of the log:

```
sudo tail -f /var/log/aerospike/aerospike.log
```

UDF logging

Log entries are used to debug User Defined Functions (UDFs). To log UDF entries in a separate log file, do the following:

Edit the Aerospike configuration file:

`/etc/aerospike/aerospike.conf`

Locate the "logging" stanza and modify it to look like this:

```
logging {
    file /var/log/aerospike/aerospike.log {
        context any warning
     }
     file /var/log/aerospike/udf.log {
              context any critical
              context udf debug
   }
}
```

Restart Aerospike (`sudo service aerospike restart`)

If you are not comfortable with using a tool such as vi to edit the file, simply copy the example file by using:

```
sudo cp -f ~/aerospike.conf /etc/aerospike
```

Development Environment

You will need to setup you language development environment to include Aerospike:

- Java
- C#
- Go
- PHP

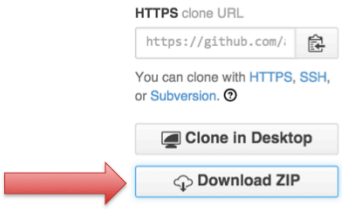
The details of each environment are in the next slides – follow the instructions for you language

C# Client is one of the several client libraries that can be used to access Aerospike's database from an application. For a full list of client libraries, please visit http://www.aerospike.com/develop/

Labs on GitHub

The lab Exercises, Answers and student Notes are stored on GitHub. Clone the GitHub repository with the following command:

- `git clone https://github.com/aerospike-edu/student-workbook.git`



C# Client is one of the several client libraries that can be used to access Aerospike's database from an application. For a full list of client libraries, please visit http://www.aerospike.com/develop/

C# Client Setup - Download

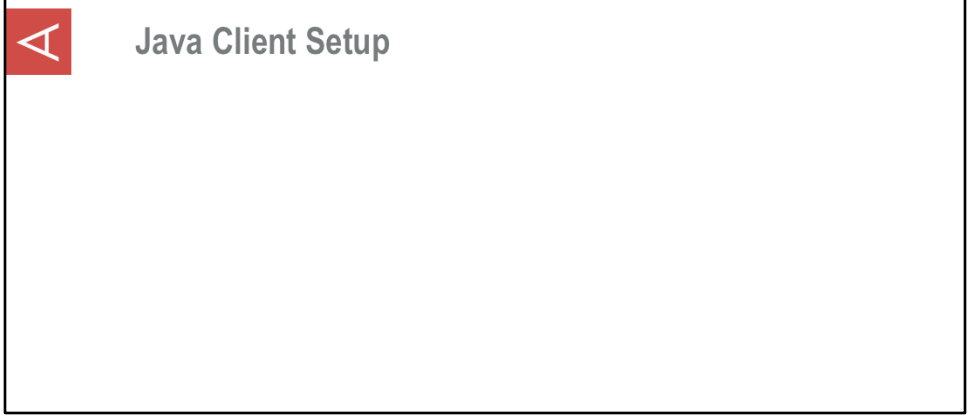
NuGet

- The compiled library is also available via NuGet. To install, run the following command in the Package Manager Console:

```
PM> Install-Package Aerospike.Client
```

Download (Alternative)

- Visit this link to download the latest version – http://www.aerospike.com/download/client/csharp/latest


Java Client Setup

Java Client Tools - Install

- Download and install Maven:
  - http://maven.apache.org/download.cgi#Installation
- Optional Eclipse Installation:
  - Download and install Eclipse JDT – Kepler or better
    - http://www.eclipse.org/jdt/
  - Install m2e plugin (maven):
    - Add an Update site for m2eclipse
    - http://download.eclipse.org/technology/m2e/releases
    - Install Maven Integration for Eclipse
  - Install Aerospike Developer Tools plugin:
    - Add an Update site for the Aerospike Developer Tools
    - https://github.com/aerospike/eclipse-tools/raw/master/aerospike-site
    - Install Aerospike Developer Tools

Java Client Setup – Maven or download

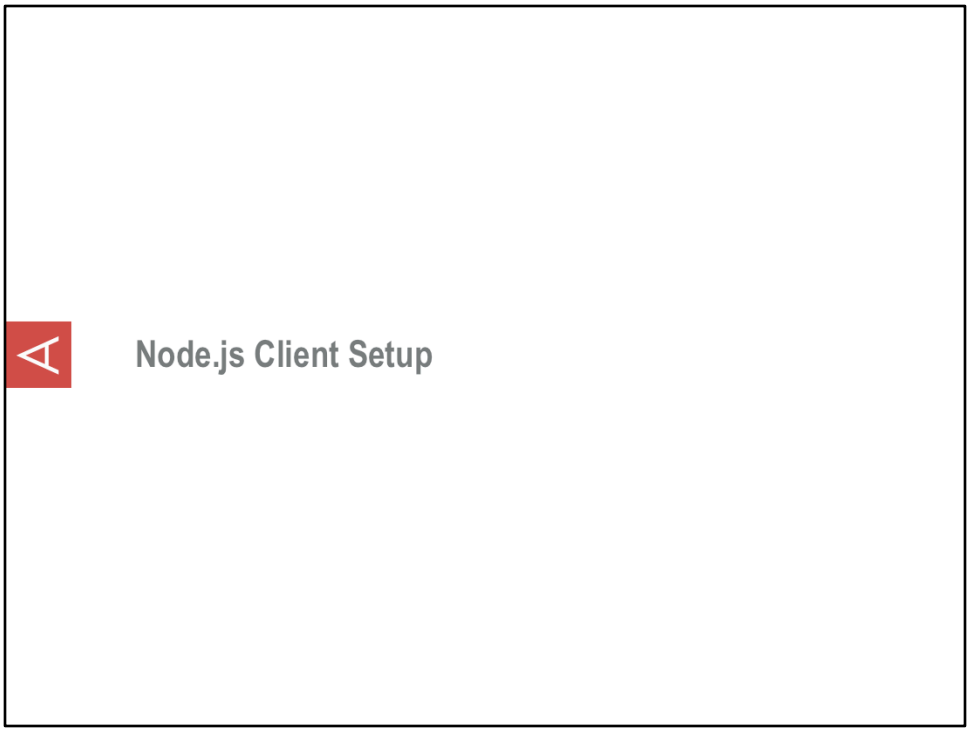
Maven

```
<dependencies>
   <dependency>
     <groupId>com.aerospike</groupId>
     <artifactId>aerospike-client</artifactId>
     <version>3.1.XX</version>
   </dependency>
 </dependencies>
```

Download (Alternative)

- Visit this link to download the latest version – http://www.aerospike.com/download/client/java/latest
- Unzip/untar files into local folder. The package contains source code for the Java client, comprehensive examples and a Benchmark tool

Note: For this class we will use **Maven** to build the project and manage dependencies.



Node.js Client is one of the several client libraries that can be used to access Aerospike's database from an application. For a full list of client libraries, please visit http://www.aerospike.com/develop/

Node.js Client - Compatibility

- Compatibility:
  - Node.js
    - Versions v0.10.x and v0.12.x
  - OS
    - Mac OS X, CentOS/RHEL 6.x, Debian 6+, Ubuntu 12.04, Ubuntu 14.04
    - **IMP**: Aerospike Node.js Client currently does not have support for Windows
  - Mac OS X:
    - 10.8 or greater
    - Xcode 5 or greater
    - Xcode Dev Tools

Node.js Client - Setup

- Install Node.js:
  - To install the latest stable version, visit http://nodejs.org

Note: To easily manage your existing Node.js installations and switch between versions, use version manager such as **nvm**. To install, visit https://www.npmjs.com/package/nvm. Once installed, you may use it to install different version of Node.js using command ***nvm install <version>*** or to switch to a different version you already have installed, use command ***nvm use <version>***


PHP Client Setup

PHP Client Setup - Download

- **Download** the latest source from the GitHub repository: https://github.com/aerospike/aerospike-client-php/releases/latest
- Unzip the files to a local folder. The package contains the **source code** for building the PHP extension.
- Alternatively you can **use Composer** (https://getcomposer.org/): composer require aerospike/aerospike-client-php "*"

PHP Client Setup - OS X

- You need command line tools to compile the extension:
  xcode-select --install
- You will use Homebrew to install the prerequisites. Please install brew:
  http://brew.sh/

```
brew update && brew doctor
brew install automake
brew install openssl
brew install lua
```

PHP Client Setup - RedHat Variants

- RedHat Linux and variants such as CentOS use the **yum** package manager.

```
sudo yum groupinstall "Development Tools"
sudo yum install openssl-devel
sudo yum install lua-devel
# on Fedora 20+ use compat-lua-devel-5.1.5
sudo yum install php-devel php-pear
```

PHP Client Setup - Debian & Ubuntu

- Debian and Ubuntu Linux use the apt package manager.

```
sudo apt-get install build-essential autoconf
sudo apt-get install libssl-dev
sudo apt-get install liblua5.1-dev
sudo apt-get install php5-dev php-pear
```

PHP Client Setup - Build

- A build script gets the latest version of the C client and compiles the source:

```
cd src/aerospike
./build.sh
```

- After compilation, the build script will provide further instructions. Follow them.
- One instruction is for creating a PHP config (.ini) for the new module. Example:

```
extension=aerospike.so
aerospike.udf.lua_system_path=/usr/local/aerospike/client-php/sys-lua
aerospike.udf.lua_user_path=/usr/local/aerospike/client-php/usr-lua
```

PHP Client Setup - Install

- Once the build script compiles the extension you need to install it:
  sudo make install
- Confirm that the module loads for your PHP interpreter:
  php -m | grep aerospike

PHP Documentation

- The API documentation for the client is in the GitHub repo:
- https://github.com/aerospike/aerospike-client-php/blob/master/doc/aerospike.md

- On our site we have a 'Quick Guide':
- http://www.aerospike.com/docs/client/php/usage/


Go Client Setup

Go Client install

Prerequisites

- Go version 1.2+
- The latest stable version of Go is at: http://golang.org/dl/

Installation

- Add the Go client in your GOPATH:

```
go get github.com/aerospike/aerospike-client-go
```

- To update the Go client:

```
go get -u github.com/aerospike/aerospike-client-go
```

Go Client Source Code

Source code (Alternative)

- The source code is available on GitHub at https://github.com/aerospike/aerospike-client-go

NOTE: Be sure to set your GOPATH correctly

Summary

You have learned how to:

- Install and Configure a single server development cluster in AWS
- Manage the development cluster


AEROSPIKE