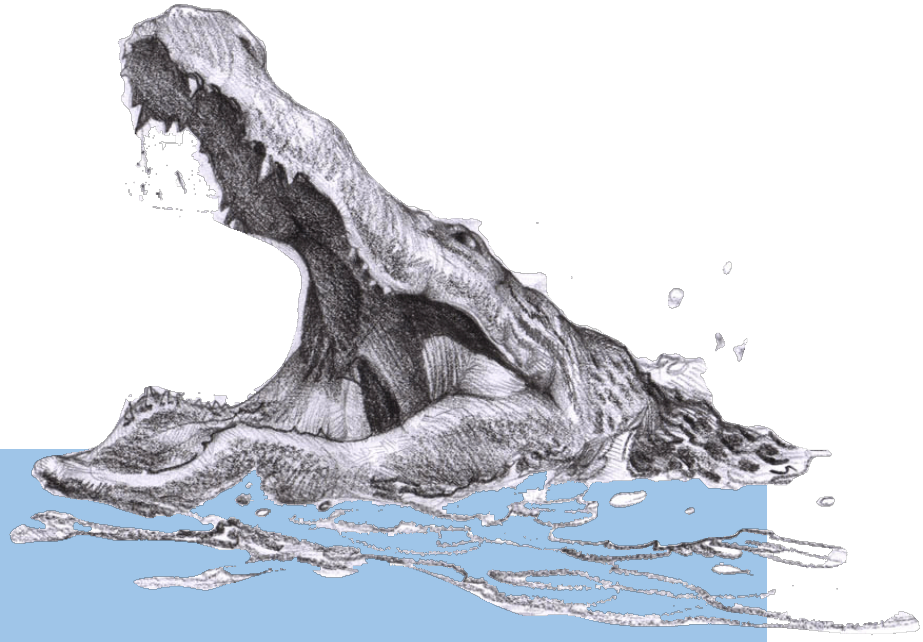

Unix Admin
Project Installation Guide

Martin Ting
Zachary Sells
Michael Brevard
Keshav Dasu
Caleb Martinez

Introduction

This document covers the details of the technology used in the project as well as instructions for installation and running. To skip to the instructions to set up the project, go to page 2.

Environment

The project uses **Vagrant** as a configuration management tool and shell scripts to provision the virtual machines. Each virtual machine is utilizing the "chef/centos-6.6" box from the vagrantcloud repository.

Packages

The packages that each virtual machine requires and installs are listed in the following table:

VM Name	Java(wget)	HTTPD(yum)	Tomcat(yum)	HAProxy(yum)	MySQL (yum)
apache					
webapp					
db					

Configuring packages

The configurations needed for each VM are listed in the following table:

VM Name	Configuration changes
apache	1. Configure /etc/httpd/conf/httpd.conf 2. Configure HAProxy /etc/haproxy/haproxy.cfg
webapp	1. Configure /etc/httpd/conf/httpd.conf 2. set java environment variables and update path 3. install necessary Java libraries 4. deploy/redeploy webapp WAR file
db	1. set-up admin username/password via mysql command 2. populate the database

Note: These are automated with the provision scripts.

Instructions

To set up the entire stack of VMs, go to the directory with the Vagrantfile in it (In this case, it should be in the <**projectroot**>/**extra_credit**/ directory. A simple vagrant up command will set up all the VMs. To run automated testing, run one of the <projectroot>/Testing/tests/test[n].sh scripts to run a single test case.

Scripts

This section will cover each script that was included in the project deliverables and what it does. These scripts have been run or will be run by the Vagrant provision scripts.

filename	Description
*bootstrap.sh	Provision scripts used by Vagrant for initial VM set-up
confighaproxy.sh	Script that generates HAProxy configuration files Usage: ./confighaproxy.sh <tomcat-ip1> <tomcat-ip2> ⋯ <tomcat-ipn>
configjava.sh	Script to configure Java and set Java environment variables & path*^
generateData.py	Python script for generating data to populate the data with
importDataMySQL.sh	Shell script used by db server to populate database *^
logparse.pl	parsing script for analyzing our webapp log data
upcon.sh	concurrent vagrant up *
provisioncon.sh	concurrent vagrant provision --reload *
deploy.sh	developer script for creating the war file from source and putting it in the right place for deployment. *

* Works only in development project directory tree due to hard links and folders that are differently named. Will not work in deliverable directory structure.
^ expected to be run on VM

Validation

On the host machine, visit **http://127.0.0.1:8087/CS183WebApplication/?auto=off. The web application should display with no errors.**