Barcelona GSE Data Science Center

Introduction to Computing

September, 2018

Barcelona GSE Data Science Center

Goals

This module will provide the tools you need to start being productive using Linux servers.

You will:

- Launch a cloud virtual machine
- Connect via SSH.
- Use the command line
- Write a shell script
- Install programs
- Use Docker

TCP

TCP is a protocol, a “transport layer” representation that abstracts the two-way communication between two computers. It ensures all the “packets” get delivered in the proper order.

HTTP, the protocol of the web, is built on top of TCP.

SSH, “secure shell”, is also built on top of TCP .

SSH can be used to login to another computer and control it via a “shell”, particularly via the a CLI shell.

Barcelona GSE Data Science Center

Launching a Server

Let's launch a virtual machine on AWS.

We will use SSH to login and control the machine via a CLI.

We will run a Jupyter notebook on this server as well.

The Jupyter notebook creates a web server, which we access over HTTP.

So we will create two TCP "channels" to communicate with our virtual machine both via SSH and via HTTP.

Key pairs

The first TCP channel we will create is to SSH into the server.

Logging in via SSH requires a password or RSA keypair.

For AWS instances, you must use a keypair!

A keypair consists of a private key and a public key.

You can create this keypair, or AWS can create this keypair.

ssh-keygen & ssh-agent

```
eval "$(ssh-agent -s)"
ssh-add [path-to-pem-file]
```

Launching a Server

Demo:

(create a key)

(configure: specs, volumes, security groups)

(connect via SSH)

Shell

There are many shells in the world.

Luckily, most use similar commands.

“Bash” is a very common shell that is available in most Linux distros, as well as in Mac OSX.

All shell CLI’s come with builtin commands. For example:

```
which
pwd
cd
```

Practise:

(Navigating and tab-complete (creating, moving, copying, viewing files)

`touch`

`mv`

`cp`

`cat`

`head`

`tail`

Barcelona GSE Data Science Center

$HOME

Home contains per-user settings.

It’s alias’d to ~ and available in environment variable: $HOME

Environment Variables

These are just variables, but they live in the “process” (the shell, or whatever is started from the shell)

$HOME is an example of an environment variable.

You can create environment variables, which can be a way to pass information to your program.

```
echo $USER
```

Barcelona GSE Data Science Center

Practise

If you want more practise with shell commands and basic programs:

OverTheWire

Programs

What if you want to create a program?

```
echo "date > foo" > write-date-to-foo.sh
chmod +x write-date-to-foo.sh
./write-date-to-foo.sh
```

You've created a program.

Barcelona GSE Data Science Center

Executables

Shell scripting is one way to create programs.

The other way is to create an executable (a.k.a. binary)!

An executable is machine code: instructions that your computer can run.

Machine code is created by compiling source code.

Source code is for humans, machine code is for machines.

Compilers are translators, human > machine!

Shell as a program

Shells, such as bash, are executables that runs on your computer!

```
stdin >  bash  > stdout
       ---^v---
       computer
```

Compiling from source

Demo:

(Compile CPython from source)

https://github.com/python/cpython

Downloading binaries

Demo:

(Download pre-compiled binary of Julia)

https://julialang.org/downloads/platform.html

(Symbolic linking and $PATH)

Using package managers

Demo:

(Use apt to install r-base)

Reproducible environments

Setting up a server requires lots of programs.

Every version of a program is different: software is alive!

How can you ensure your code will run on a new server? Or someone else's computer?

Cloud Images

Cloud providers usually provide a way to make an “image” of your server.

AWS calls these AMI (Amazon Machine Image).

This can be a great solution if you do all your work on AWS servers.

Barcelona GSE Data Science Center

Docker

Docker is the name of a popular containerization software.

Containers are like little mini computers that run inside your computer.

If you've heard of virtual machines, it's like that, but without the CPU overhead.

Docker

Demo:

(Install Docker)

(Pulling images)

(–name, -d, –rm, -v, -p)