Cloud starter

Inspire your team to greater heights in the Cloud.
Audience: Developers, Architects, Testers.

This module is the first part of the Cloud Starter course. It introduces the course and establishes some guidelines on what you need to get started.

Introduction

- Pre-requisites and first steps
- Tools and Clouds
- Provisioning models
- Firewalls, Access and User accounts

This section summarises the module and starts chipping away at the blocks that can inhibit that experimentation and discovery in the Cloud.

First steps

- Starting point
- Not comprehensive

> *Teach a person to ~~fish~~ mess around with Internet examples for hours on end....*

This course is about giving those unfamiliar with the tools a starting point. This is not a ground-up introduction of:

- every single tool/language and

- every single coding statement in

- every single example

Instead it's a good way to get started, build something really quickly and give you the basic grounding you need to mine the Internet for a deeper understanding.

This is intended to be the beginning of your DevOps journey, but I hope the course will also be relevant for those who've got some experience.

Target architecture

- Build something fast

- Build something real
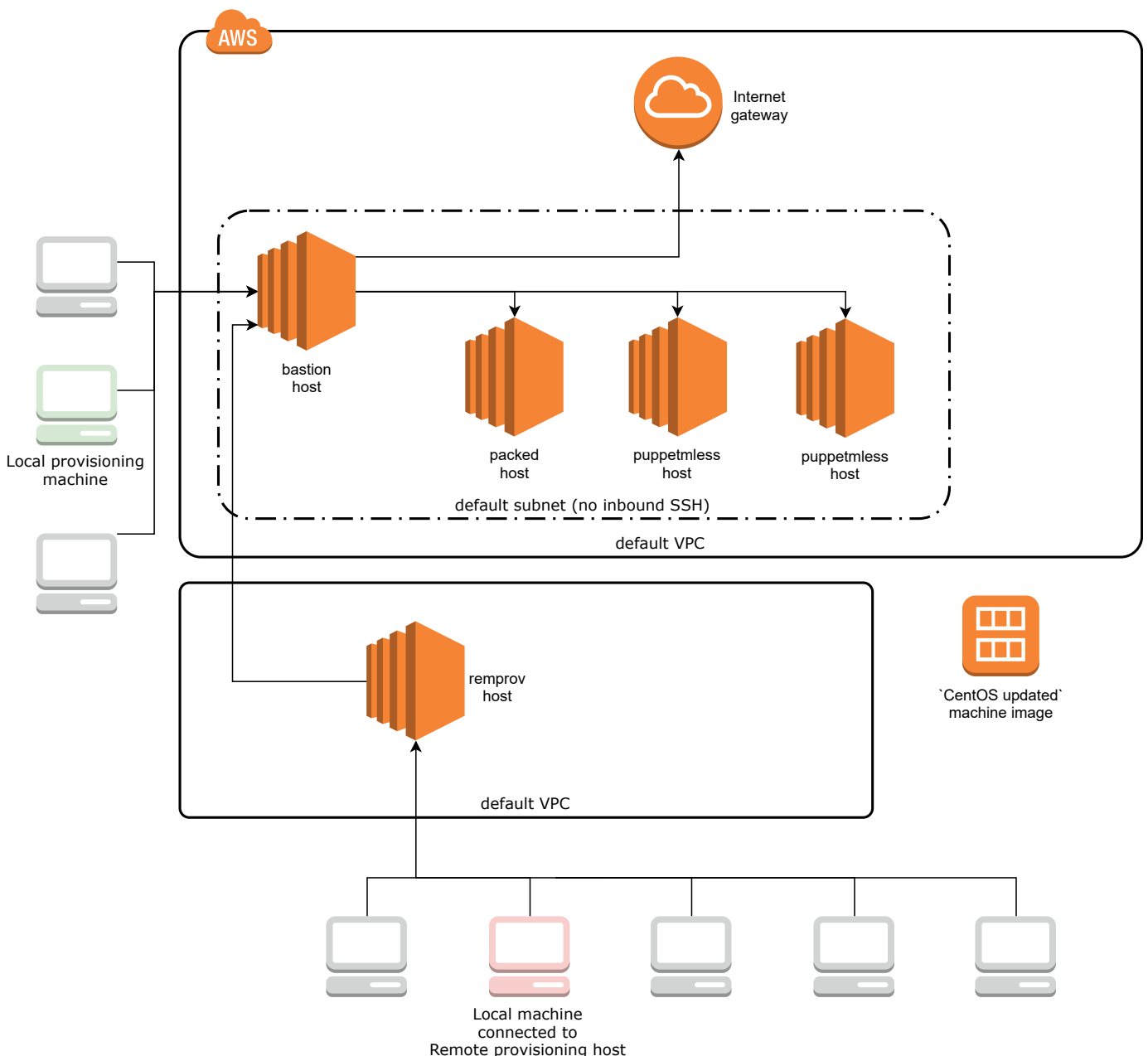
The diagram above shows the basic, secure Cloud environment we will automatically provision as part of this course.

The course is experimentation-driven, meaning that you'll get little from it unless you're firing up the examples and pinging the servers and systems we create.

What is DevOps?

- Not a person

- Not a programming language

- Not a fad

- It's a collaboration

- ...and tools that aid that

It's a contraction of Development and Operations, essentially an idea about how the two fundamental activities of web service delivery have to work together.

In a grim and not so distant past, a 'Dev' team wrote some code, then hurled it over the fence for the 'Ops' team to run it. A lot of the rich understanding of how the application was designed to function was lost. Frequently third-line support ended up reverting to the Dev team anyway.

Increasingly today, we rely on a more pragmatic "you build it, you run it" mentality. This empowers developers to really think about the likely issues they'll encounter once in Production.

This means that as coders, we have a greater responsibility for building and maintaining the *environment* in which our application will perform, as well as the application itself.

This 'DevOps' collaboration has precipitated a new set of tools - the subject of this course - which facilitate that.

Pre-requisites

- Laptop with install (root/Administrator) access

- Account with major Cloud Provider

- Time

To get the most out of this course, you need to be able to run and customise the examples.

- Essential

    - Basic coding knowledge.

    - A local machine with an SSH client and a web browser (required for remote provisioning).

- Preferred

    - A local machine with root/Administrator access and 100MB free space (required for local provisioning).

    - Familiarity with an IDE such as IntelliJ IDEA.

Tools

- Ever improving

- Generalised methodology

- Alternatives available

These tools are likely to change. Their evolution, even over Internet-timescales, has been short and storied. Many came before and many new tools will follow, but hopefully the skills you adopt in learning these will give you the confidence to repeat the process when there are new, better alternatives available in the future.

Equally, this is a developing course, so future iterations will try to keep pace with their evolution.

The tools, specifically

- Provisioning (Terraform)

- Machine images (Packer)

- Configuration management (Puppet/Ansible)

- Containerisation (Docker)

- Container orchestration (Kubernetes)

Clouds

- AWS first

- Alternatives available

The course is based on Amazon Web Services because at the time of writing, AWS is the largest and most established Cloud provider.

While many of the examples mention AWS, the tools mentioned are multi-platform and can be equally applied to Microsoft Azure, Google Cloud or half a dozen others providers.

Provisioning models

- Local development

- Remote production

- Remote provisioning (remprov)

When developing application code, there are pros and cons to all development models.

Local development places all the source code on your local desktop/laptop computer. Access is typically fast (low latency) and high-bandwidth (direct GUI). If the application can be run locally, testing can benefit from that same fast/high-bandwidth access.

While 'local' is a great place for application development, it's not the best place to host production solutions. Always on hosts, reliable power supplies and dedicated network connections make servers a better place to host production services.

When developing infrastructure-as-code (IAC), the same applies but to a lesser extent. Since most provisioning involves the client talking to a Cloud Provider (AWS/Google Cloud/Microsoft Azure), the advantage of local development is lessened.

Remprov starter

- Public ready-to-go AMI

- Pre-installed with all the tools

The `workstream-remprov` AMI comes pre-packaged with all the content you need to get started.

Firewalls

- Zero inbound port access

- SSH outbound access

- 22 -> 443 workaround

Remote provisioning requires no inbound port access, but you will need to be able to SSH out to your remprov machine.

That outbound access tends to be on port 22. Some firewalls may block outbound access on port 22. A special version of the remprov AMI is available on request with SSH running on a non-standard port (443).

Access and user accounts

- Cloud account

- User account

- Scope (IAM permissions)

Open-source code

- Available on GitHub (https://github.com/lightenna/).

This course introduces the DevOps Workstream (https://github.com/lightenna/devops-workstream) set of open-source examples in Terraform, Packer, Ansible and Puppet.

All the Markdown source code for this presentation (https://github.com/cleverlight/lightenna/blob/master/_includes/presentations/2018-09-21-cloud-starter.md) is available on GitHub.

For more information, please contact Alex Stanhope on +442071835698. ©2018 Lightenna Ltd