Infrastructure as a Service

Infrastructure as a Service

- Earliest and most popular form of cloud service

History

1. Early 2000's : Virtual private servers (VPS)
   - Mainly used for web-hosting
   - Restricted functionality and heavily sandboxed
2. 2006: Amazon EC2 (Elastic Compute Cloud) launches
3. Full-featured *hardware virtual machines* offered at low cost

Hardware Virtualization

- Modern IaaS platforms provide **Virtual Machines**
- Want to provide the illusion of a virtual CPU, virtual memory, virtual I/O devices to VMs
- Make virtual things appear and behave like physical things
- Job of hypervisor is to safely virtualize resources.
- Performance of virtual resources should match physical resources!

IaaS VMs

- IaaS provides users with VMs
- Also refered to as “instances” (EC2) (instance of a virtual server)

Basic Instance Lifecycle

1. Select and configure instance
2. Run/Launch
3. Connect via ssh to run scripts/applications
4. Stop/terminate when done
5. Some clouds: explicit delete after termination

Accessing IaaS resources

1. Cloud-provider command-line API : gcloud, aws, ec2-api, ..
2. Often a wrapper around an http api (in case of gcloud)
3. Web-based console
4. Language libraries (boto for aws, etc.)

SSH Setup:

1. For each instance, can specify the credentials that will be used to access it over SSH
2. Gcloud uses the user credentials by default

GCP API

- gcloud auth list/login
- gcloud config set account
- gcloud compute config-ssh
- Register ssh key-pair for access to VM
- gcloud compute machine-types
- gcloud compute instances info

Instance Attributes

1. Instance-name , cloud-assigned instance-id
2. Geographical region where VM is to be launched
3. “Type” or size of VM (Amount of CPUs, memory, etc).
   - `n1-standard-2` : 2 vCPUs, `n1-highcpu-X`, `n1-highmem-X`
   - Small “micro” VMs with 1 vCPU to multi terabyte ram VMs
4. Operating System and boot-disk image
5. Networking setup: IP addresses, firewalls, etc.
6. Other metadata

Who's gonna pay for it?

- Every student should have one GCP $50 coupon
- GCP also provides $300 free credits for first time use, but needs a credit-card.
- Not charged if usage is less than $300.

Caution!

- Running Virtual Machines costs money
- Often low (few cents an hour), but costs quickly add up if you "forget" to stop a VM.
- Or launch 100 VMs due to a bad automation script

Instance Pricing

- Same VM is available with multiple pricing options
- **On-demand:** Default. Pay per minute
- **Preemptible:** Pay per minute, can be *preempted* at any time (more on this later in the course)
- **Reserved:** Meant for long-term use (months/years)
- GCP gives sustained use discounts
- All storage, networking charges are added on top of the instance cost.

Networking

- By default, a single public IP address will be allocated
- Connect to server on that IP (using SSH etc)
- These are IPv4 addresses, so rare and valuable
- Additional public addresses are charged
- Cap on maximum free public IPs per account (50?)

Storage and virtual disks

- Conventional storage in the form of virtual disks
- Can specify size and speed
- Can easily copy and clone disks
- **Ephemeral Storage:** Changes are lost after VM is terminated
- **Persistent Storage:** Changes are persisted even after VM is terminated.
- Persistent storage is charged on a per-gigabyte-hour basis
- Ephemeral storage can be used for boot-disks etc.
- Advantage: “Clean slate” on every boot.
- Downside: All changes (file edits, configurations, software packages installed) are lost

Boot Scripts

Cloud-init

- Running configuration scripts on boot is very convenient to initialize an instance
- Cloud-init is a new standard adopted by distributions and cloud platforms
- During instance creation: specify init script as a file or text string

Managing state inside VMs

State: Code, dependencies, libraries, config files, data, parameters, etc...

- One extreme: Boot default OS image, and initialize everything, everytime. Pull from some common repository, etc. Useful when software changes frequently.
- Sometimes hard to determine what state will not change and what frequently changes.
- Known libraries/packages can be downloaded ahead of time and a new disk image created.
- But managing and keeping track of disk images non-trivial. Naming issues: img-new-43, etc.
- Post-boot configuration: copy setup script (using scp), then ssh and run the script, and then start the service. Many things can go wrong: different test/dev environments, files not checked into the repo, etc.