# Cloud Computing

Ryan Tolboom

New Jersey Institute of Technology

# What is it?


- A silly catchphrase with a disheartening origin
- Technically it can be IaaS, SaaS, PaaS
- Is it new?
- Why virtualize? Why not just rent servers?

# What’s good about it?

Redundancy

Scalability

Co-Location

Lower Bandwidth Costs

Lower Building Costs

Lower Staffing Costs

# What’s bad about it?

Recurring Costs

Loss of Physical Control

Security Concerns

# How does it work?

## Virtual Machines

- VirtualBox, QEMU, VMware, AWS Nitro , etc.
- What are they?
- How do they work?
- Why use them?

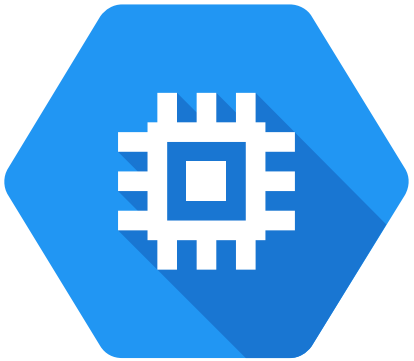
## Containers

- Docker, etc.
- What are they?
- How do they work?
- Why use them?

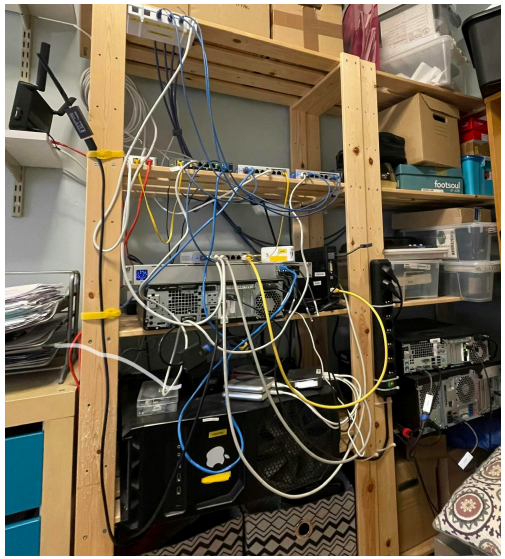
# Who does it?

- Amazon Web Services
- Microsoft Azure
- Google Compute Engine
- Most PaaS and SaaS runs on one of these





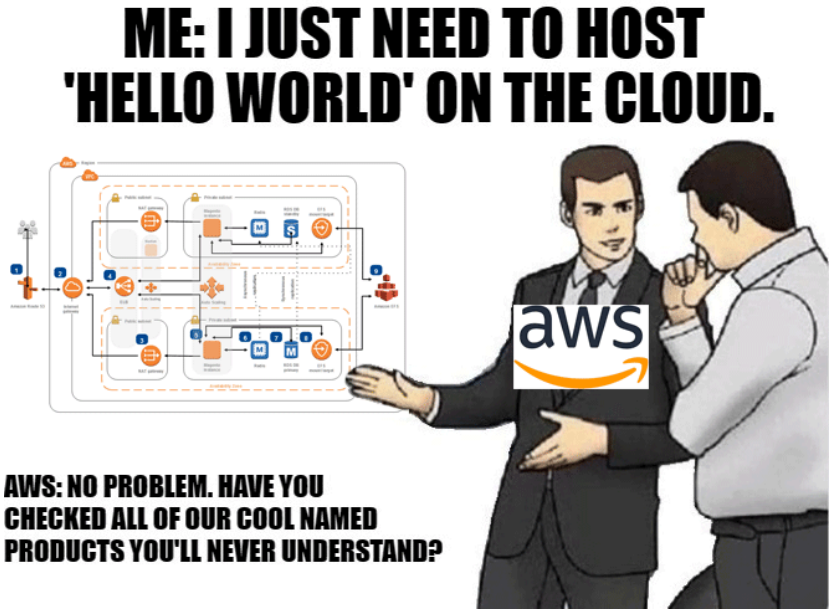
# Want to try it at home?



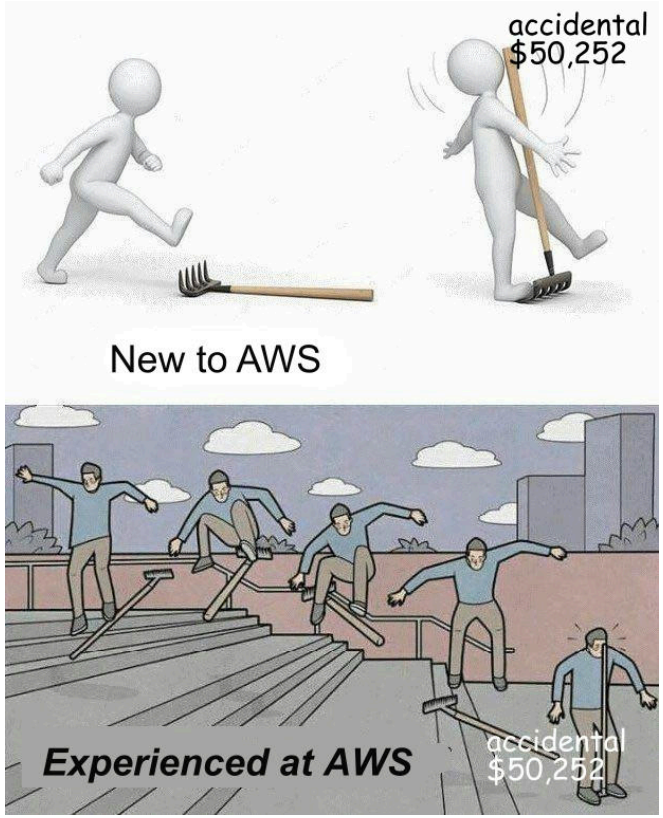
- /r/homelab
- Server hardware is CHEAP (data centers refresh often)
- Virtualization Platforms
  - Proxmox
  - ESXi
  - XenServer
- You will learn a lot
- You may also scare your family members and pets with loud noises and heavy computers!
- What is in the “bare metal” of a virtualization server? Memory and processor cores. The more the merrier.

# What can AWS do for you?

- Network architecture as code (subnets, VPCs, internet gateways, NAT gateways, load balancers, etc).
- Tons of services: S3 storage, RDS database, ECS containers, EC2 compute, IAM identity and access…
- Deploy *quickly*
- Scale *quickly*
- Unique products: spot instances, AMI sharing, mechanical turk 😐



# How do you deploy to AWS?




From least favorite to most favorite:

1. AWS Web Console / Instance Management
2. AWS Command Line and scripts
3. AWS Command Line and Ansible/Vagrant
4. Terraform and Provisioning Scripts
5. Terraform and Ansible

Terraform licensing issue?