

Avoiding pain when operating in the Cloud

Neil Armitage - Senior Consultant @ Ensono Digital (Amido)

Whoami

- Senior Consultant at Ensono Digital (Amido)
- Engineering Manager running the Skyscanner Cloud Operations Team
- Worked on the Kubernetes implementation @ Skyscanner
- VMWare (vCloudAir DBaaS platform)
- Continuent Inc (MySQL Clustering)
- Before that mainly MySQL/Oracle DBA going back to mainframes in the 1980's

Disclaimer

- Views are my own - not current or past employers
- Focused more on AWS but apply to Azure, GCP, Oracle Cloud
- Examples are current as of Summer 2022 but will date quickly
- Identities have been changed to protect the innocent (or not so innocent)
- I'm a pretty rubbish presenter



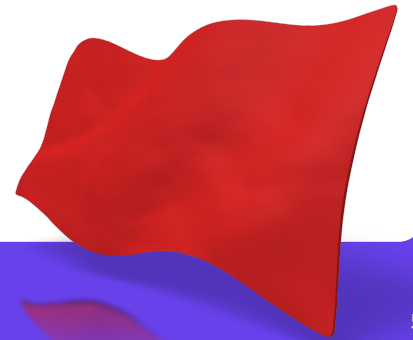
What will I bore you with?

- What is the cloud
- Cost management and how to waste money
- Limits
- Security
- Running Kubernetes in the cloud
- Architecting for Failure



Cloud 'experts'

- I'm not an expert in anything
- If someone claims to be a Cloud expert that should be a red flag
- AWS made over 2000 posts on their what's new feed in 2021
- Having all 12 AWS Certifications does not make you an expert



What is the Cloud?



What is the Cloud?

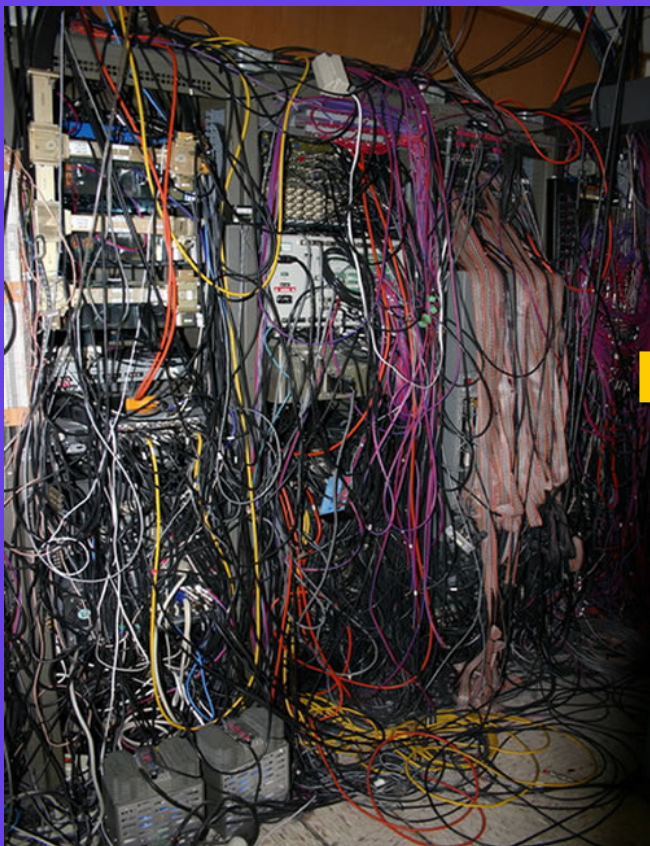
Cloud computing is the on-demand availability of computer system resources, especially data storage (cloud storage) and computing power, without direct active management by the user. Large clouds often have functions distributed over multiple locations, each location being a data center. Cloud computing relies on sharing of resources to achieve coherence and typically using a "pay-as-you-go" model which can help in reducing capital expenses but may also lead to unexpected operating expenses for unaware users.

https://en.wikipedia.org/wiki/Cloud_computing

What is the cloud?

- The cloud is just someone else running your data center for you
- Consolidation gives access to the cost savings of scale
- All provide basic compute then add extra value added services e.g. Database as a service (DBaaS)
- Removes the need to employ or subcontract any form of hardware support
- Capex vs Opex





'Hyperscale' Providers

Company	2021 Revenue	2021 Market Share (%)	2020 Revenue	2020 Market Share (%)	2020-2021 Growth (%)
Amazon	35,380	38.9	26,201	40.8	35.0
Microsoft	19,153	21.1	12,659	19.7	51.3
Alibaba	8,679	9.5	6,117	9.5	41.9
Google	6,436	7.1	3,932	6.1	63.7
Huawei	4,190	4.6	2,681	4.2	56.3
Others	17,056	18.8	12,697	19.8	34.3
Total	90,894	100.0	64,286	100.0	41.4

Other Providers

Wikipedia lists 294 cloud providers (July 22)

 **ABACUSNEXT®**

 **DBGALLERY**

 toast

 OCI

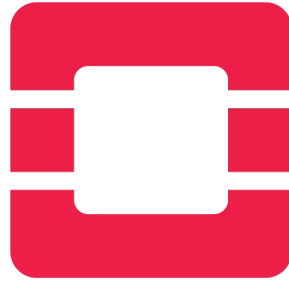
 **VONAGE**

 **Clio**

 **LUNAVI**

zuora

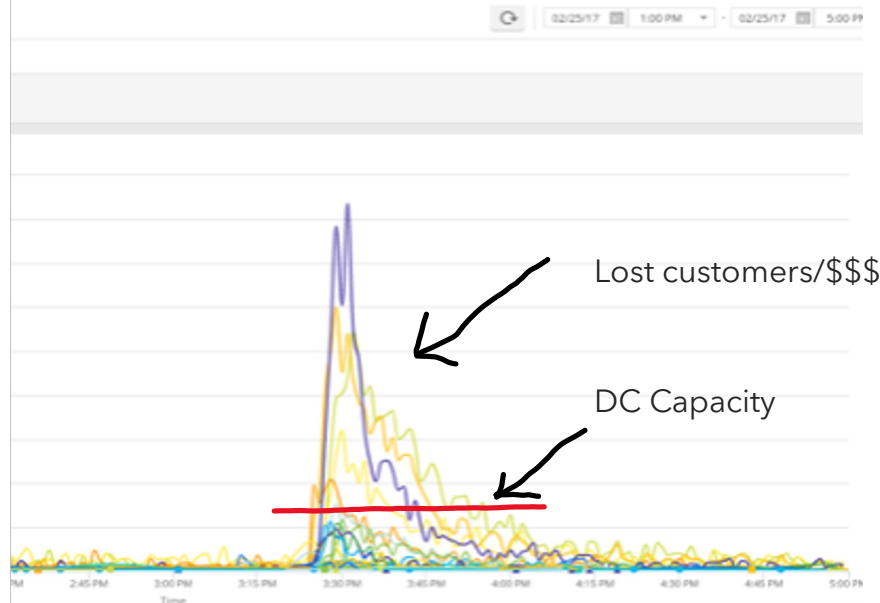
Or maybe your own private one



openstack®

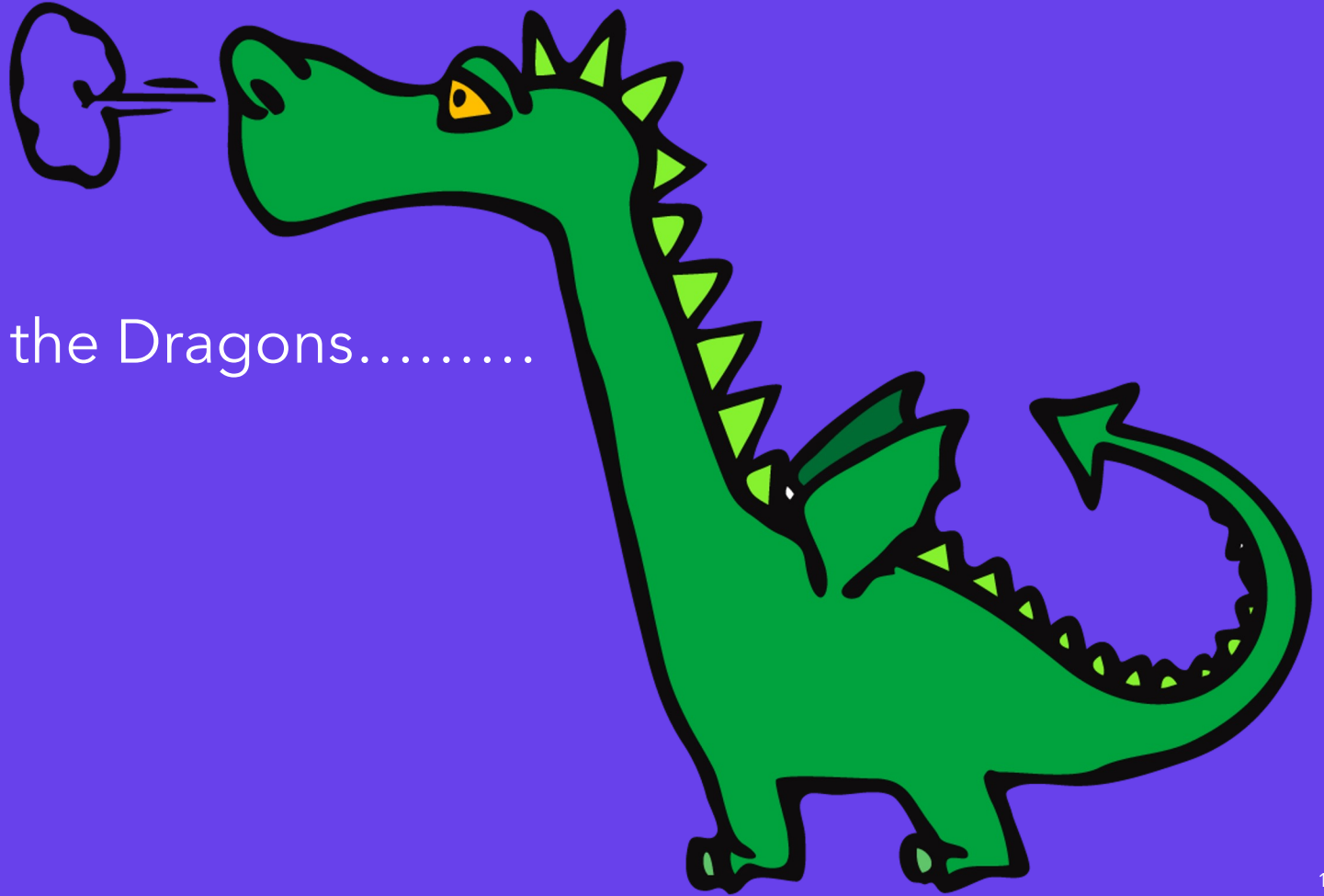
Why move from a Data Centre

- Elastic Capacity
- Scale up and down with load
- "No Limit"



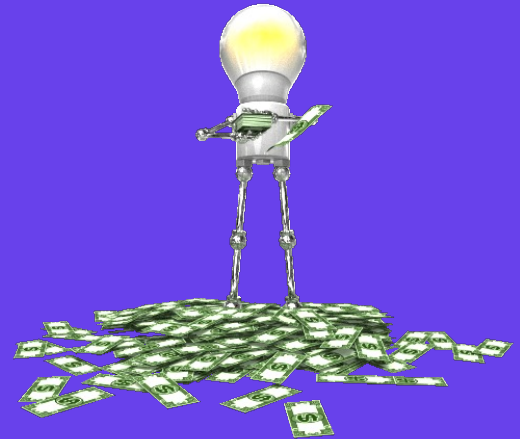
Advantages

- “Instant” availability of compute resources if you have a credit card
- No waiting for servers to be purchased and provisioned
- Want a test environment?
 - Press a button
 - Grab a coffee
 - Play around with the application
 - And forget to tear it down :)



And here be the Dragons.....

Cost Management

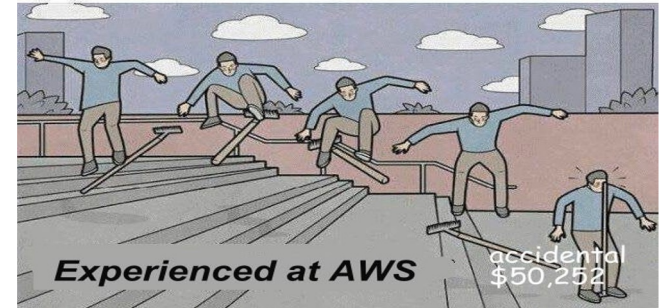


Cost Management

- Controlling cost is hard
- In AWS limiting cost is not available from Day 1
- After 11 years I still get burnt

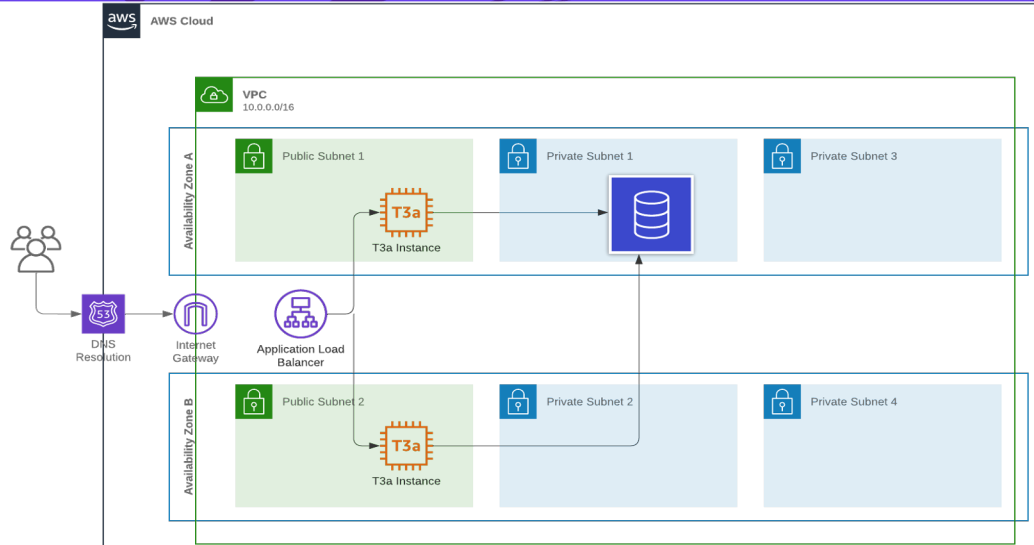


New to AWS



So it starts....

- Developers wanted a simple environment to test in.
- So we created a IAC pipeline to deploy on demand for a Git Branch



Not too expensive

- \$400 to support testing isn't bad
- Management a happy



My Estimate [Edit](#) [🔗](#)

Estimate summary [Info](#)

Upfront cost
0.00 USD

Monthly cost
373.09 USD

Total 12 months cost

4,477.08 USD

Includes upfront cost

Maybe I should have cleaned up

- We never got around to automating the deletion of the branch on PR merge, **we trusted developers to clean up after themselves**
- Cost leaving 50 environments laying around for a year = \$223K
- Management are slightly less happy



Then of course they wanted more

- A couple of API Servers or 3
- Lots of SQS
- NAT Gateways
- Kinesis Streams
- DynamoDB

Estimate summary [Info](#)

Upfront cost
203.40 USD

Monthly cost
1,965.98 USD

Total 12 months cost
23,795.16 USD
Includes upfront cost

ouch

Cost of running 50 environments for a year =

\$1.2m





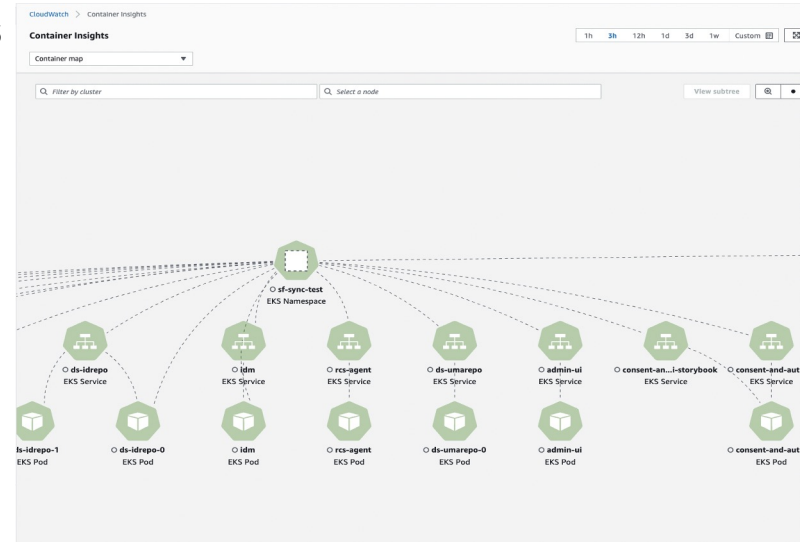
Automate cleaning up

Plenty of tools to use

- Aws-nuke - <https://github.com/rebuy-de/aws-nuke>
- Cloud Custodian - <https://cloudcustodian.io/>
- Keep a close eye on the bill

Automatic Monitoring

- Cloud providers provide monitoring solutions
 - AWS container insights
 - Azure monitor
- Or for the rich Datadog



Automatic Monitoring

- Easy to set up

```
2
3 module "cloudwatch_logs" {
4   source = "git::https://github.com/DNXLabs/terraform-aws-eks-cloudwatch-logs.git?ref=0.1.4"
5
6   enabled = true
7
8   cluster_name           = module.eks.cluster_id
9   cluster_identity_oidc_issuer = module.eks.cluster_oidc_issuer_url
10  cluster_identity_oidc_issuer_arn = module.eks.oidc_provider_arn
11  worker_iam_role_name     = module.eks.worker_iam_role_name
12  region                  = var.region
13  settings                 = { "cloudWatch.logRetentionDays" = 7 }
14 }
15
16 module "cloudwatch_metrics" {
17   source = "git::https://github.com/DNXLabs/terraform-aws-eks-cloudwatch-metrics.git?ref=0.1.1"
18
19   cluster_name           = module.eks.cluster_id
20   cluster_identity_oidc_issuer = module.eks.cluster_oidc_issuer_url
21   cluster_identity_oidc_issuer_arn = module.eks.oidc_provider_arn
22   worker_iam_role_name     = module.eks.worker_iam_role_name
23 }
```

But it needs careful monitoring

▼ EU (London)

Amazon CloudWatch

		\$2,516.17
		\$1,005.68
\$0.00 per request - first 1,000,000 requests	2,731.000 Requests	\$0.00
\$0.01 per 1,000 metrics requested using GetMetricData API - EU (London)	992.000 Metrics	\$0.01
\$0.01 per 1,000 requests	9,353.000 Requests	\$0.09
\$0.10 per alarm metric month (standard resolution) - EU (London)	24.458 Alarms	\$2.45
\$0.30 per metric-month for the first 10,000 metrics - EU (London)	3,343.762 Metrics	\$1,003.13

AmazonCloudWatch EUW2-TimedStorage-ByteHrs

		\$2.88
\$0.0315 per GB-mo of log storage - EU (London)	91.327 GB-Mo	\$2.88
First 5GB-mo per month of logs storage is free.	2.635 GB-Mo	\$0.00

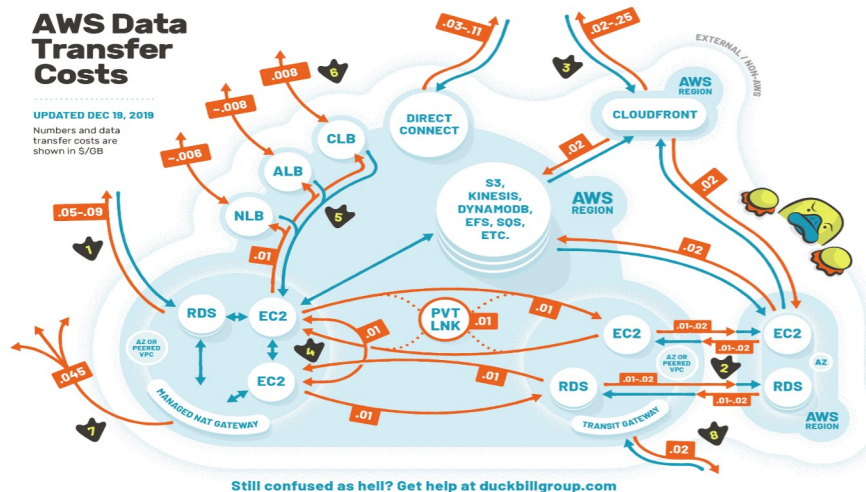
AmazonCloudWatch PutLogEvents

		\$1,507.60
\$0.5985 per GB log data ingested - EU (London)	2,432.026 GB	\$1,455.57
\$0.5985 per GB log data ingested for the first 10TB - EU (London)	86.938 GB	\$52.03

AmazonCloudWatch StartQuery

		\$0.02
\$0.0059 per GB log data scanned by CloudWatch Logs Insights queries - EU (London)	2.599 GB	\$0.02
First 5GB per month of log data scanned by CloudWatch Logs Insights queries is free.	1.578 GB	\$0.00

Data Transfer Costs



- ➡ Inbound traffic is typically free – outbound is not. Some (but not all) internal traffic is **free**.
 - ➡ Outbound traffic costs are shown **per transmission**.
 - 1 Direct outbound data starts at **\$.09/GB** for less than 10TB, and discounts with volume. **First 1GB is free**.
 - 2 Region-to-region traffic is **\$.02/GB** when it exits a region for indicated services except between us-east-1 and us-east-2, where it's **\$.01/GB**. Even data wants to get out of Ohio.
 - 3 Outbound CloudFront prices are highly variable by geography and regional edge cache and start at **\$.085/GB** in US/Canada.
 - 4 Internal traffic via public or elastic IPs incurs **additional fees** in both directions.
 - 5 Cross-AZ EC2 traffic within a region costs as much as region-to-region. ELB-EC2 traffic is **free** except outbound crossing AZs.
 - 6 Elastic Load Balancing: Classic and Network LB is priced per GB. Application LB costs are in LCUs, not S/GB.
 - 7 Traffic via Managed NAT Gateway – regardless of destination – costs an additional **\$.045/GB** on top of other transfer, including internal transfer (S3, Kinesis, etc.).
 - 8 Data processing charges apply for each gigabyte sent to the AWS Transit Gateway – whether from a VPC, Direct Connect or VPN.
- Inspired by Open Guide to AWS's data transfer diagram
github.com/open-guides/og-aws

Source : <https://www.lastweekinaws.com/blog/understanding-data-transfer-in-aws/>

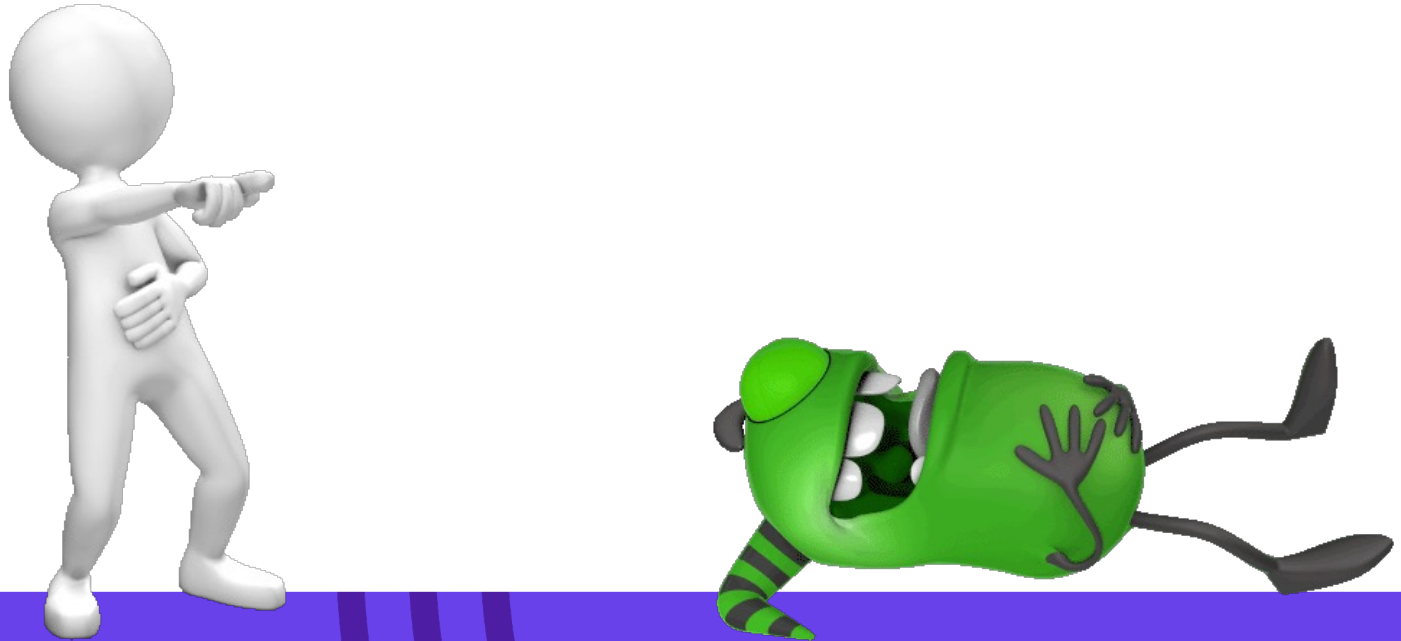
Data Transfer Costs

- Pay for data between regions
- Pay for data between availability
- Pay for data from your app to a C



Scalability

The cloud is infinitely scalable



Resources are finite



Neil Armitage 11:43 AM

oops



Launching a new EC2 instance. Status Reason: We currently do not have sufficient c5.9xlarge capacity in the Availability Zone you requested (eu-west-1a). Our system will be working on provisioning additional capacity. You can currently get c5.9xlarge capacity by not specifying an Availability Zone in your request or choosing eu-west-1b. Launching EC2 instance failed.



1

But it's the cloud I can have as many IP's as I want!

(a well-respected QA Engineer @ VMware)

Plan for unavailability

- Run Auto scaling groups with multiple instance types
- Don't assume you will get what you want
- Our Kubernetes clusters run with at least 5 different types over multiple AZ's

C4	AZ1	AZ2	AZ3	On-Demand
8XL	\$0.28	\$0.27	\$0.29	\$1.76
4XL	\$0.21	\$0.19	\$0.16	\$0.88
2XL	\$0.08	\$0.07	\$0.08	\$0.44
XL	\$0.04	\$0.05	\$0.04	\$0.22
L	\$0.01	\$0.01	\$0.02	\$0.11

Cloud API's

- Generally, every action on a Cloud Platform is via an API.
- The console, CLI or an SDK all use the API's.
- Can be complex to understand and inconsistent.
- They have limits and throttles to protect the platform for everyone



API Limits

Nope CA still bein throttled:

```
W0210 06:42:26.738192      1 default_retryer.go:43]  
Retries are being throttled (Throttling: Rate exceeded  
status code: 400, request id: 0799e781-2cff-11e9-  
b2ce-7545453e6506). Total retries: 0
```

API Limits

- Each account has a limit on it's API usage
- Limits are not published _(ツ)_/
- Limits seem to change
- One bad script can kill all the API calls in the account
- Kubernetes software is really good at this (e.g. cluster-autoscaler)

API Limits

- It's quite hard to find the cause
- Work with TAM's
- 'Splunk' like analysis of Cloudwatch logs
- Education

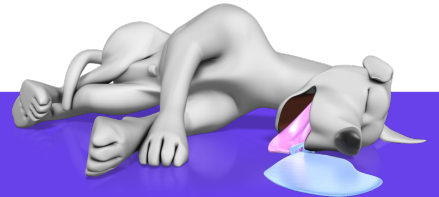
Account limits stop you hurting yourself

- These all can be raised but it needs to be done via a support ticket and can take time..... so Plan ahead
- Each region/account needs a separate ticket - can be automated (Cloud Custodian)
- also don't ask for too big a change as support have to refer big raises internally



Spot Instances

- Makes use of unused capacity
- 2 Mins warning and they can all disappear
- You **can not** monitor spot prices to predict this
- Use lots of instance types, lots of AZ's, lots of Regions.....
- Can save tons of money



Security



Help

Common

enson

aws r/aws · Posted by u/pic sh 9 days ago

aws r/aws · Posted by u/letsgodaledo 7 months ago

general aws Thanks to all of the "My account was hacked!" posts here, I finally setup MFA on all of my accounts

395 upvotes 86 comments 2 awards

aws r/aws · Posted by u/CarlTheGreaser 3 months ago

discussion My old account was hacked, left large billing and I was told to update the payment method by aws team to continue the review even if I want to suspend my account.

0 upvotes 5 comments 0 awards

aws r/aws · Posted by u/grey_sky 6 months ago

discussion AWS Account Hacked - Racked up \$20,000 of Fraud Charges - MFA was enabled

56 upvotes 73 comments 0 awards

aws r/aws · Posted by u/TheBrain511 4 months ago

security My account was hacked and i have dedicated host running on ec2 tab

7 upvotes 17 comments 0 awards

aws r/aws · Posted by u/Heater2000 3 months ago

billing AWS Acc hacked and charged

0 upvotes 4 comments 0 awards

aws r/aws · Posted by u/utkarsh0708 4 months ago

billing My account got hacked and suspended for not paying the bills, when I haven't even used it. INDIA

0 upvotes 10 comments 0 awards

aws r/aws · Posted by u/offkeyharmony 9 months ago

discussion Got hacked and found a 30k bill. Please turn on MFA if/when you start

1800 worth of compute usage

ould take?

maker Charges

roof!

help please

gin, and I got \$3k+ bill

was changed)

nged

what can I do?

I should have read the instructions.

- The account hasn't been 'hacked'
- The front door has probably been left open
- Either
 - Poor root password
 - No MFA
 - Credentials shared on GitHub

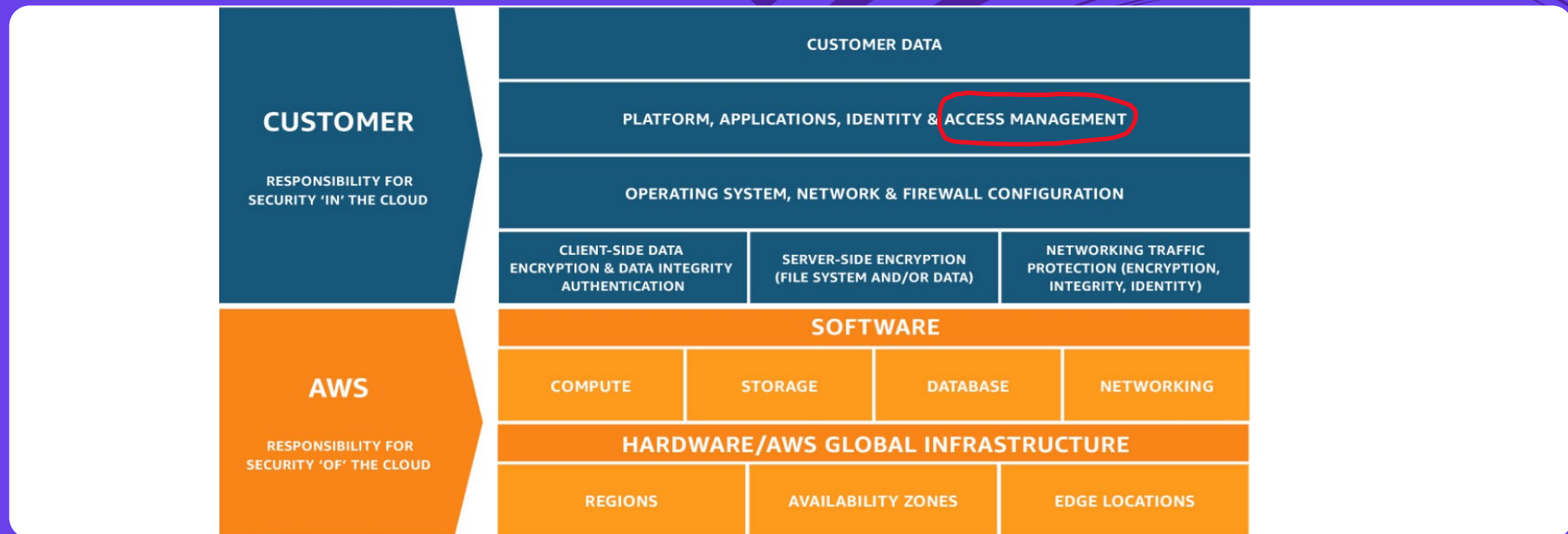


But it's not my problem!

- It's not AWS fault, you signed up for a service and didn't follow good guidelines.
- AWS could help by enforcing MFA etc but it would hinder larger users
- You are responsible for the bill's, but AWS can help
- If you left the keys in a car and the doors open would you blame Ford?



Shared Responsibility Model



AWS Free Tier != no cost

- Free Tier is not free, only certain services are free.
- You can still run up huge bills by not being careful.
- Running a EC2 and RDS can rack a bill of a bill of > \$50k a year
- Use sandbox services - agloudguru
- Use tools like aws-nuke



Be careful with Keys

- Secure access keys, do you really need them?
- Reduce the ways into the Account
- Consider using SSO and external identity provider (google/AD)
- IAM roles everywhere
- 2FA
- Don't commit keys to GitHub

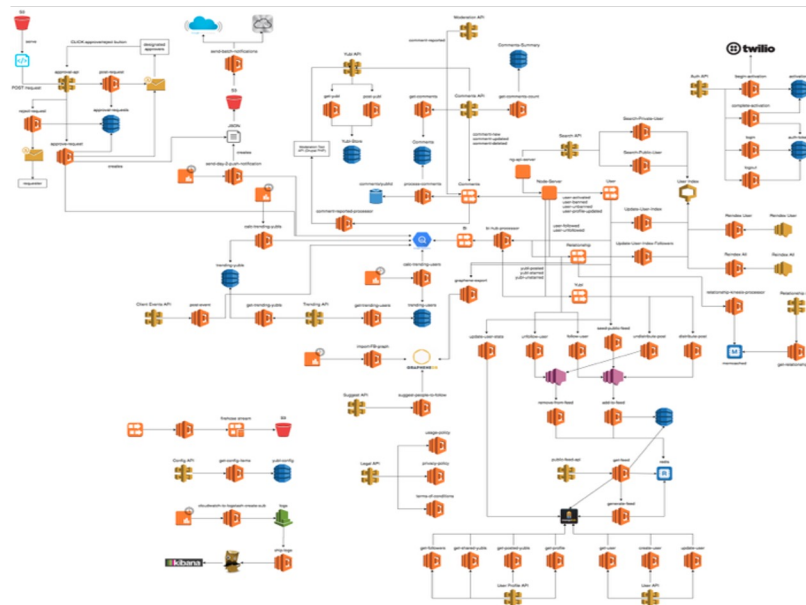


Architecting for failure



Just because you can - doesn't mean it's right

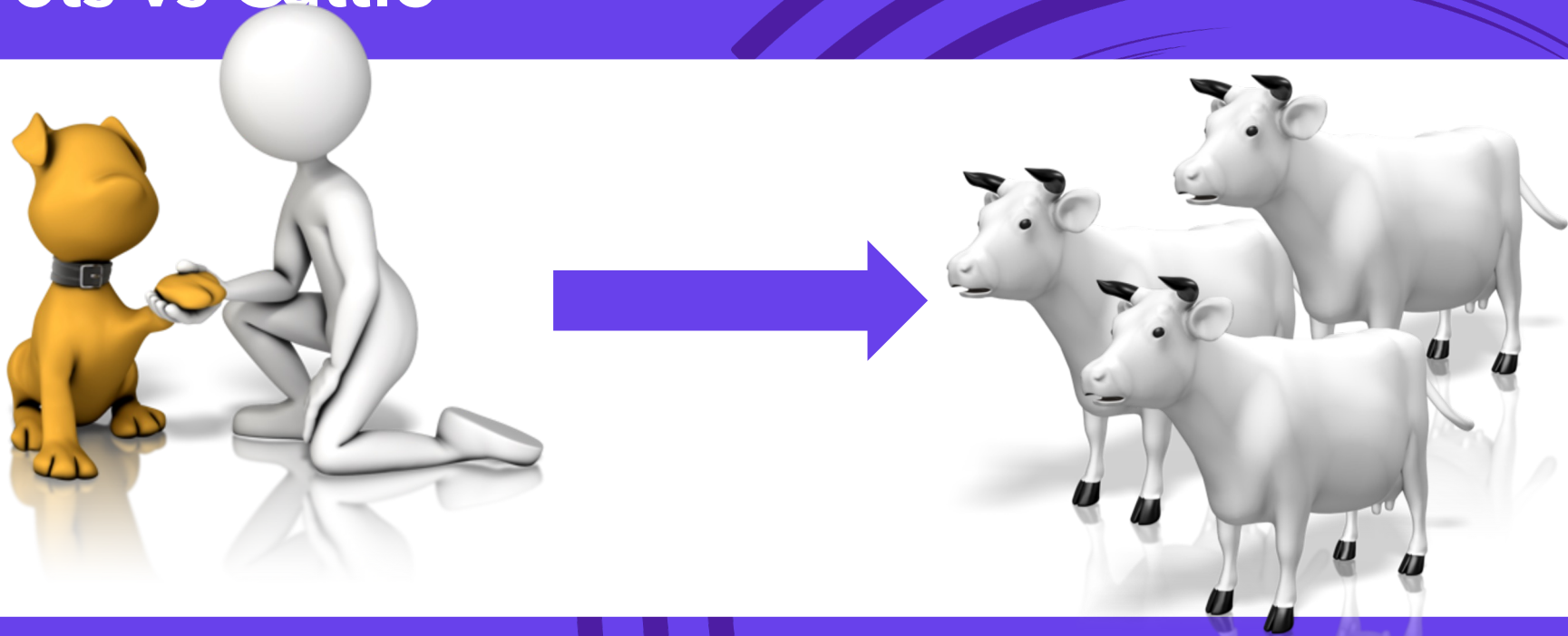
- AWS provides over 200 Services, GCP 100 +
- You don't have to use all of them
- Simple is good and easy to maintain
- Good rule to follow - Can you fix something at 2am with a hangover (or still drunk)



Servers will die

- Underlying hosts will die or be retired
- Can just happen randomly
- Every host should be replaceable with no manual effort
- In theory no ssh access ever needed
- Serverless and products like Fargate remove any server management

Pets vs Cattle



SSL Certificate Expiry

- In AWS certs can be validated by either email or DNS record
- Email is the quick and easy method
- But in 12 months you need to both see the email and click a link
- DNS validation is initially harder but you never need to do anything again

Select Regions carefully



- And us-east-2 is now starting to show some of the same problems
- AWS run some core services out of us-east-1 e.g. IAM

Burstable instances

- Tx Series in AWS, Bx instances in Azure
- Handles workloads that are not consistent
- When the CPU is not in use you can earn credits
- The instances are generally cheaper
- But they can run out of credits leaving the hosts underperforming until more credits are accrued



clickops

- It's very easy to spin up a resource via the console
- The problem is when you are asked to deploy something again you must remember what you did
- Spend a bit more time deploying with cli tools, Terraform, CDK, crossplane
- Add it to a Gitops workflow
- It seems like a lot of work, but it will pay off in the end



Availability of Resources

- Don't expect resources to be available
- Not everything in all regions
- Some services are restricted to certain customers
- GPU shortage

Lift and Shift

- “Lift and Shift” is where existing on-premise servers are moved into the cloud
- Can be seen as a quick win with plans to re-architect in the future (which never happens)
- Often drags legacy problems into the cloud (I’ve seen windows hosts in the cloud running VMware agents)
- Re-architect were possible, invest some time.
- Sometimes it’s the only option

Kubernetes in the cloud

Managed service vs build it yourself

Build it yourself

- Deploy infrastructure, API hosts, etcd hosts
- Install and configure Kubernetes software
- Test patches and upgrades
- Provide 24x7 support for the cluster
- Great way of learning technical skill but requires considerable resource
- Good for specialized deployments

Managed Service

- Pay AWS, Google or Microsoft about \$60/months
- Concentrate on building and running the business-critical applications



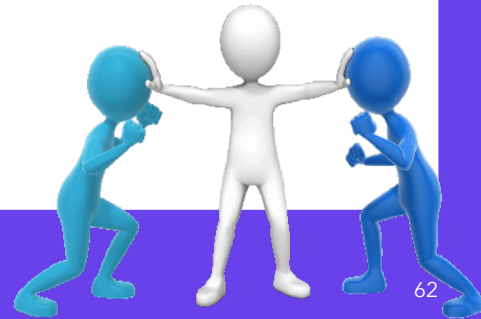
Careful with Subnetting

- Kubernetes can use lots to IP addresses
- Can be hard to add more (certainly in AWS)
- Consider using IPv6

☐	Name ▾	Subnet ID ▾	State ▾	IPv4 CIDR ▾	Available IPv4 addresses ▾	Availability Zone
☐	eks-vpc-dev-privat...	subnet-0acacd581194363e8	✔ Available	10.0.3.0/24	38	eu-west-2c
☐	eks-vpc-dev-privat...	subnet-058f715049339c4e6	✔ Available	10.0.1.0/24	33	eu-west-2a
☐	eks-vpc-dev-privat...	subnet-0bb6d098b74851787	✔ Available	10.0.2.0/24	8	eu-west-2b

Kubernetes fighting with cloud systems

- Kubernetes cluster Auto-scaler tries to manage nodes
- AWS Auto Scaling Group (ASG) tries to keep nodes balanced across AZ's
- The 2 start to fight against each other, nodes constantly churning
- Create ASG per AZ
- Use cloud specific auto-scaler (karpenter)



PVC's and Cloud Storage

- By default, EKS uses EBS for Persistent Claim Volumes (PVC's)
- EBS Volumes are in 1 Availability Zone and can't move
- In case of an AZ outage your Pod can move but the data will be stuck
- Either plan for the problem - hold data in multiple AZ's
- Consider using EFS



Sustainability



Carbon Usage

- AWS Graviton instance up to 60% less energy and cheaper
- A 2018 study found that using the Microsoft Azure cloud platform can be up to 93 percent more energy efficient and up to 98 percent more carbon efficient than on-premises solutions.
- AWS 100% renewable energy by 2025
- Google is carbon neutral today, but aiming higher: our goal is to run on carbon-free energy, 24/7, at all of our data centers by 2030.

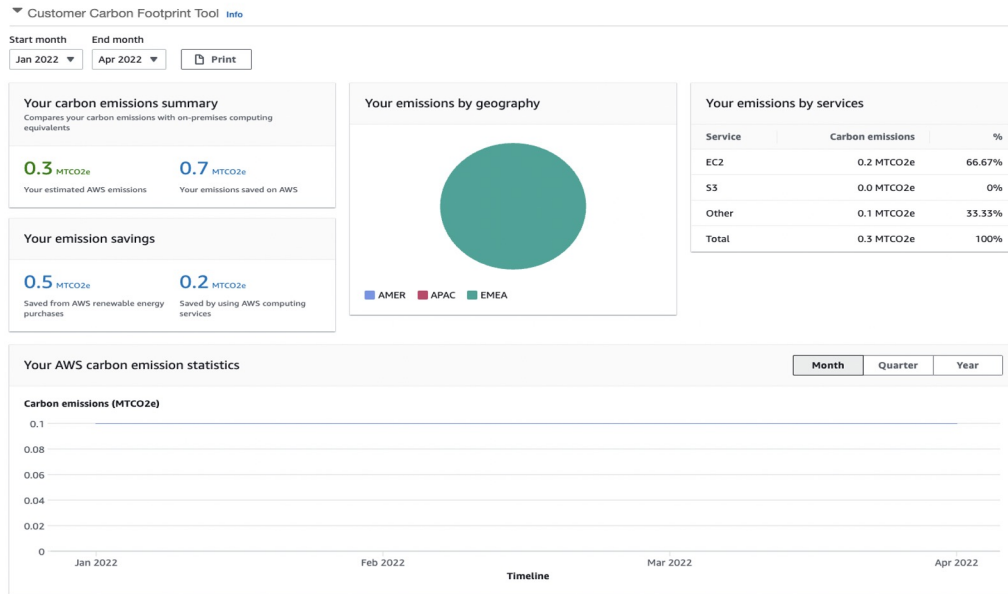


Consider 'Green' Options

- If there are wasted compute resources, you are wasting energy and generating carbon
- Do you really need that spare capacity
- Good for the planet, good for the company and good for employees
- AWS Sustainability Pillar



AWS Carbon Footprint Tool



Wrap Up

Summary

- Wasting money is bad.
- Saving the company money can mean more to spend on you..
- You will make mistakes learn from them and share them.
- Try not to make the same mistake twice.

Contact

- Neil.Armitage@ensono.com
- www.neilarmitage.com
- www.linkedin.com/in/neil-armitage-5750885

