CoHEsion

Architecting Banner Infrastructure in AWS

Gabriel Tocci

09-OCT-2019

3:00pm - 4:00pm

EAST TENNESSEE STATE UNIVERSITY

Session Format

- 60 minute time slot
    - Touch on a lot of topics
- Can you see REAL good?
- Q&A anytime
- gabrieltocci.com/talks

What is Cloud?
Cloud Services: Iaas Paas Saas

- Important to be specific
- Compare apples to apples
- Ellucian Cloud? AWS

ETSU AWS Project

- Production Live on AWS 12/2018
- Aging Banner ERP Infrastructure
  - ERP Hosts: RHEL 5, Hardware Replacement, Datacenter Migration
- Timing with Banner 9
  - Server Requisitions
  - Technology Change
- Small Team
  - 2 DBA, 1 manager/director
  - 4 senior programmers, 2 junior programmers
  - 0 system admin (on prem issues only)
  - 0 network engineers (on prem issues only)

Improved Efficiency and Scalability

- Lower overall costs
  - Pay only for what you need
  - Registration Spec'd Hardware
  - No need to estimate capacity at time of purchase
  - Better performance per $$
  - Cap-ex vs. op-ex
- Self-Service Infrastructure
  - Dell, Cisco, Compellant, Who?
  - Faster time to value
- Global resources

Improved Availability and Disaster Recovery

- Single Region - Multi AZ
  - Multiple Instances
  - Load Balanced
- Snapshots
  - AMI's and EBS
- Declarative Infrastructure
  - Infrastructure is easily rebuilt
- Dataguard on premise
  - Worst case scenario

Improved Security

- Secure by Default
- Network Isolation
  - Virtual Private Cloud (VPC)
  - Subnets: Routing tables
  - VPN Gateway
  - Security Groups
    - Ingress/Egress rules
- Monitoring & Logging
  - VPC Flowlogs
  - Cloudtrail

- Encryption Everywhere
  - SSL, S3, EBS
  - KMS
  - Certificate Manager
- 2FA
- SSH Keypair login only
  - Same as On-Prem
  - Puppet
- IAM : Resource Access
  - Roles & Privileges

Why amazon web services™ ?

Magic Quadrant

Figure 1. Magic Quadrant for Public Cloud Storage Services, Worldwide

- Market Leader
- Mature, robust, stable
- Support, Community, Examples
- Well Documented, Best Practices
- AWS CLI
- Terraform Provider
- Manages Services

Compute

Amazon EC2
Amazon Elastic Container Service
Amazon Elastic Container Service for Kubernetes
Amazon Elastic Container Registry
Amazon Lightsail
AWS Batch
AWS Elastic Beanstalk
AWS Fargate
AWS Lambda
AWS Serverless Application Repository
Auto Scaling
Elastic Load Balancing
VMware Cloud on AWS

Storage

Amazon Simple Storage Service (S3)
Amazon Elastic Block Storage (EBS)
Amazon Elastic File System (EFS)
Amazon Glacier
AWS Storage Gateway
AWS Snowball
AWS Snowball Edge
AWS Snowmobile

Database

Amazon Aurora
Amazon RDS
Amazon DynamoDB
Amazon ElastiCache
Amazon Redshift
Amazon Neptune
AWS Database Migration Service

Migration

AWS Migration Hub
AWS Application Discovery Service
AWS Database Migration Service

Networking & Content Delivery

Amazon VPC
Amazon CloudFront
Amazon Route 53
Amazon API Gateway
AWS Direct Connect
Elastic Load Balancing

Developer Tools

AWS CodeStar
AWS CodeCommit
AWS CodeBuild
AWS CodeDeploy
AWS CodePipeline
AWS Cloud9
AWS X-Ray
AWS Tools & SDKs

Management Tools

Amazon CloudWatch
AWS CloudFormation
AWS CloudTrail
AWS Config
AWS OpsWorks
AWS Service Catalog
AWS Systems Manager
AWS Trusted Advisor
AWS Personal Health Dashboard
AWS Command Line Interface
AWS Management Console
AWS Managed Services

Media Services

Amazon Elastic Transcoder
Amazon Kinesis Video Streams
AWS Elemental MediaConvert
AWS Elemental MediaLive
AWS Elemental MediaPackage
AWS Elemental MediaStore

Machine Learning

Amazon SageMaker
Amazon Comprehend
Amazon Lex
Amazon Polly
Amazon Rekognition
Amazon Machine Learning
Amazon Translate
Amazon Transcribe
AWS DeepLens
AWS Deep Learning AMIs
Apache MXNet on AWS
TensorFlow on AWS

Analytics

Amazon Athena
Amazon EMR
Amazon CloudSearch
Amazon Elasticsearch Service
Amazon Kinesis
Amazon Redshift
Amazon QuickSight
AWS Data Pipeline
AWS Glue

Security, Identity & Compliance

AWS Identity and Access Management (IAM)
Amazon Cloud Directory
Amazon Cognito
Amazon GuardDuty
Amazon Inspector
Amazon Macie
AWS Certificate Manager
AWS CloudHSM
AWS Directory Service
AWS Key Management Service
AWS Organizations
AWS Single Sign-On

AR & VR

Amazon Sumerian

Application Integration

Amazon MQ
Amazon Simple Queue Service (SQS)
Amazon Simple Notification Service (SNS)
AWS AppSync
AWS Step Functions

Customer Engagement

Amazon Connect
Amazon Pinpoint
Amazon Simple Email Service (SES)

Business Productivity

Alexa for Business
Amazon Chime
Amazon WorkDocs
Amazon WorkMail

Desktop & App Streaming

Amazon WorkSpaces
Amazon AppStream 2.0

Internet of Things

AWS IoT Core
Amazon FreeRTOS
AWS Greengrass
AWS IoT 1-Click
AWS IoT Analytics
AWS IoT Button
AWS IoT Device Defender
AWS IoT Device Management

Game Development

Amazon GameLift
Amazon Lumberyard

ETSU Primary AWS Services

- Virtual Machines
  - EC2 Instances
- Docker container orchestration
  - ECS
- Load balancing
  - ELB: HAProxy, ALB: ECS
- Continuous Delivery
  - S3, ECR
- Networking
  - Route53, VPC
- Serverless functions
  - Lambda
- Databases
  - RDS: MS-Sql

Declarative Infrastructure (Iac)

Whatever the approach, configurations should be:

- Documented
- Repeatable
- Codified
- Automated

Benefits

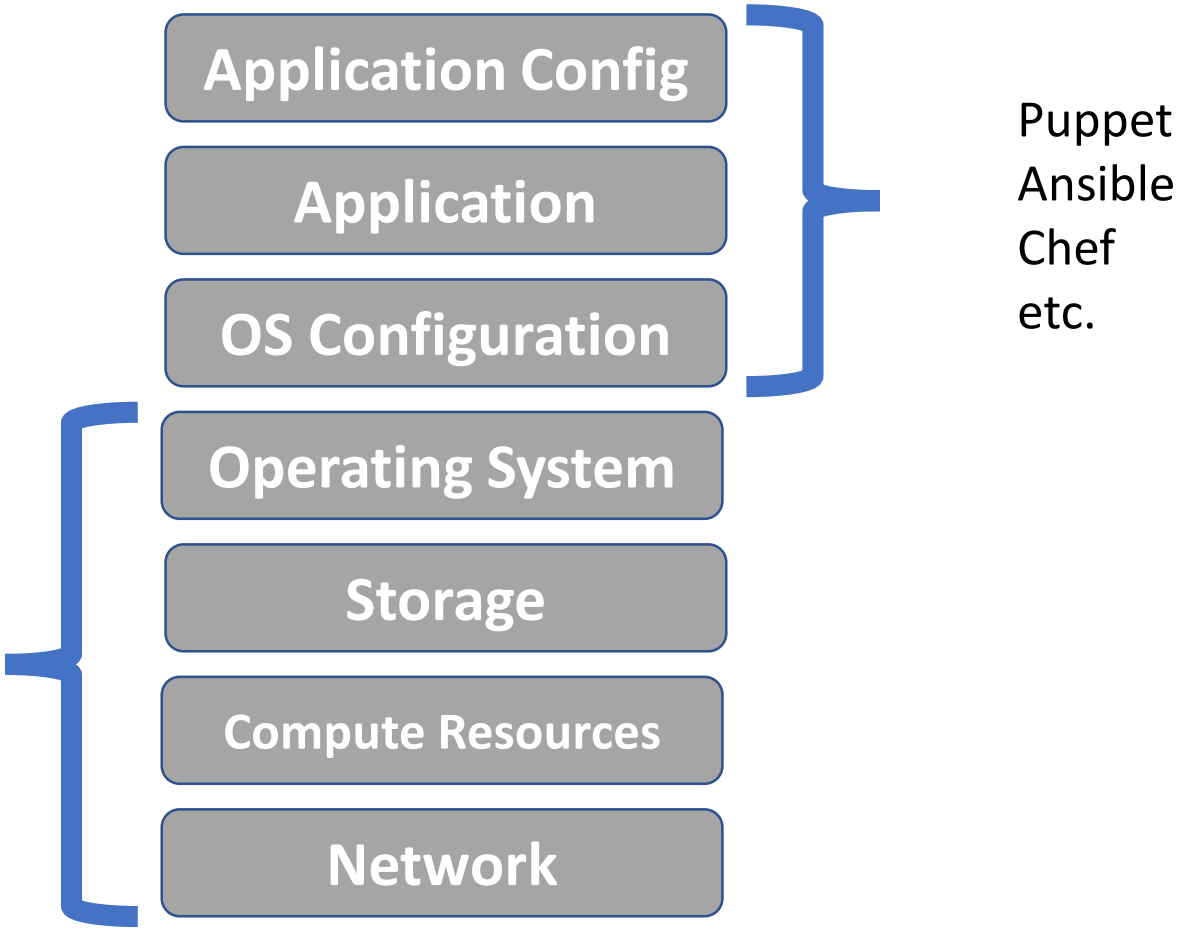
- Manages “drift” of configurations
- Declare intent and interactions of resources
- Auditable infrastructure
- Increases recovery speed
- Reduces go-live errors

ETSU Toolset

- Puppet – R10k
- Terraform
- Docker
- Gitlab-CI

Code =  git

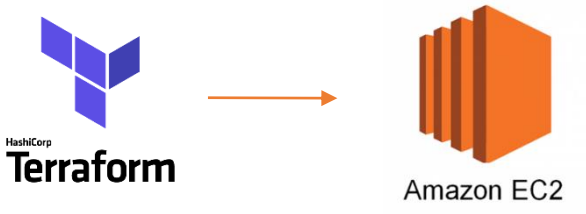
Provisioning vs. Configuration Management



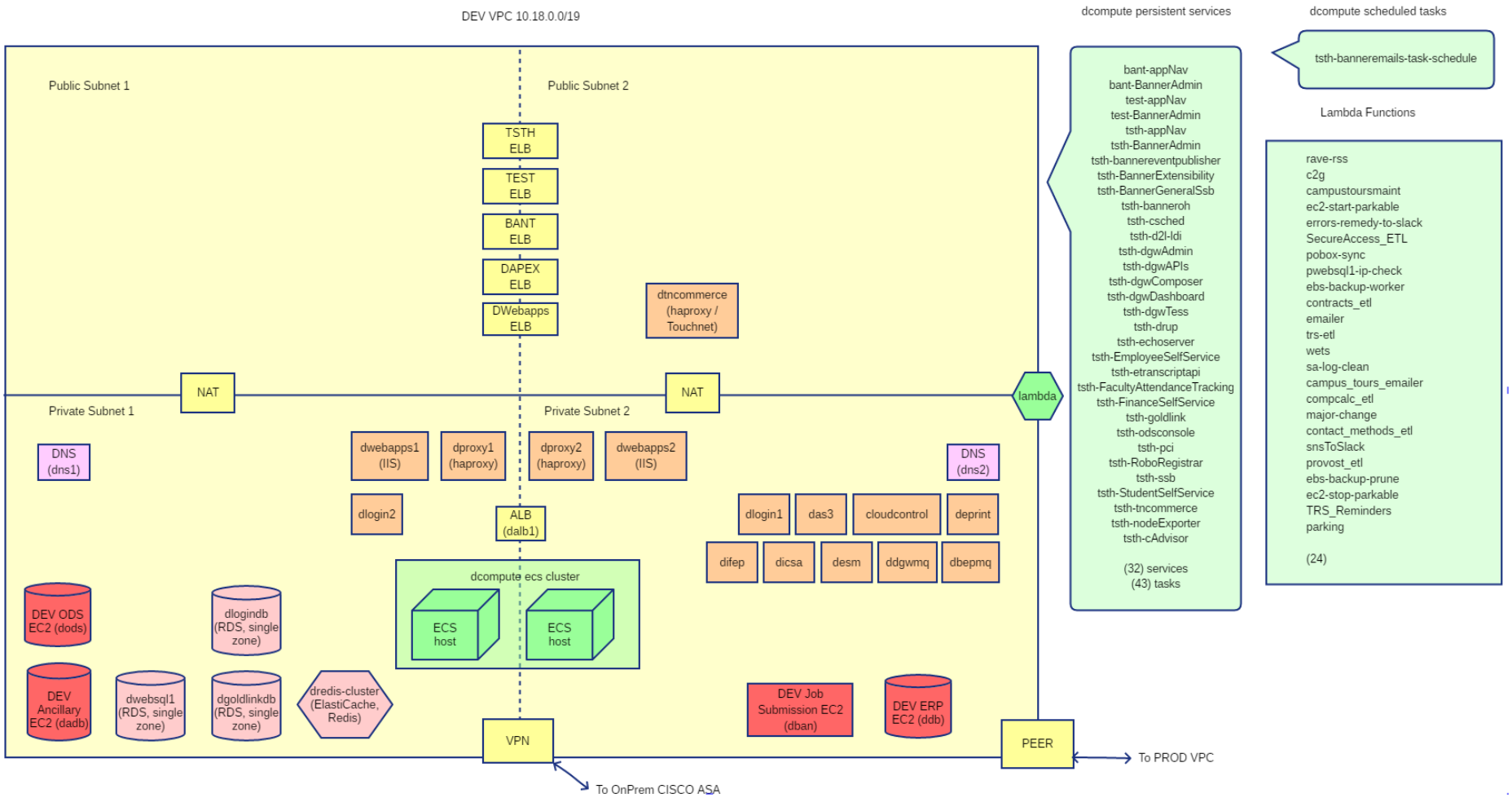
Terraform vs. AWS CloudFormation

- https://www.terraform.io/docs/providers/aws/r/instance.html
- https://docs.aws.amazon.com/en_pv/AWSCloudFormation/latest/UserGuide/aws-properties-ec2-instance.html
- https://git.etsu.edu/aws/terraform
- https://www.terraform.io/docs/providers/index.html

Terraform



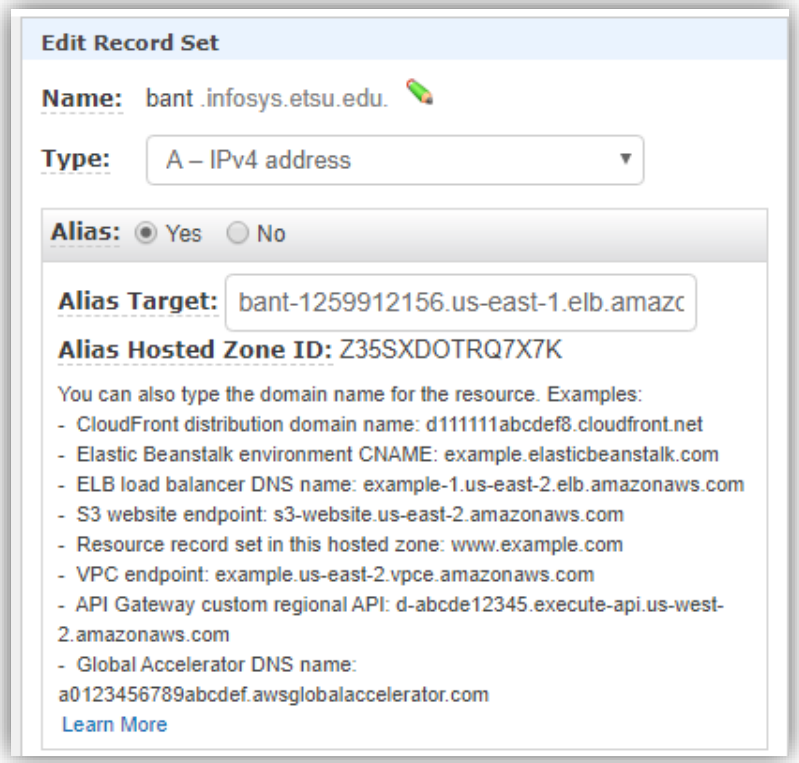

```
resource "aws_instance" "foo" {
        ami = "ami-ae7bfdb8”
        instance_type = "m4.xlarge"
        subnet_id = "${module.vpc.private_subnets[1]}"
        key_name = "linux_ec2"
        private_ip = “123.456.78.9"
        vpc_security_group_ids = ["${aws_security_group.dban.id}"]

        root_block_device{
                volume_type = "gp2"
                volume_size = "100"
                delete_on_termination = "false"
        }
}
```


DEV VPC 10.18.0.0/19
Public Subnet 1
Public Subnet 2
TSTH ELB
TEST ELB
BANT ELB
DAPEX ELB
DWebapps ELB
dtncommerce (haproxy / Touchnet)
NAT
NAT
lambda
Private Subnet 1
Private Subnet 2
DNS (dns1)
dwebapps1 (IIS)
dproxy1 (haproxy)
dproxy2 (haproxy)
dwebapps2 (IIS)
DNS (dns2)
dlogin2
ALB (dalb1)
dlogin1
das3
cloudcontrol
deprint
difep
dicsa
desm
ddgwmq
dbepmq
dcompute ecs cluster
ECS host
ECS host
DEV ODS EC2 (dods)
dlogindb (RDS, single zone)
DEV Ancillary EC2 (dadb)
dwebsql1 (RDS, single zone)
dgoldlinkdb (RDS, single zone)
dredis-cluster (ElastiCache, Redis)
DEV Job Submission EC2 (dban)
DEV ERP EC2 (ddb)
VPN
PEER
To PROD VPC
To OnPrem CISCO ASA
dcompute persistent services
bant-appNav
bant-BannerAdmin
test-appNav
test-BannerAdmin
tsth-appNav
tsth-BannerAdmin
tsth-bannereventpublisher
tsth-BannerExtensibility
tsth-BannerGeneralSsb
tsth-banneroh
tsth-csched
tsth-d2l-ldi
tsth-dgwAdmin
tsth-dgwAPIs
tsth-dgwComposer
tsth-dgwDashboard
tsth-dgwTess
tsth-drup
tsth-echoserver
tsth-EmployeeSelfService
tsth-etranscriptapi
tsth-FacultyAttendanceTracking
tsth-FinanceSelfService
tsth-goldlink
tsth-odsconsole
tsth-pci
tsth-RoboRegistrar
tsth-ssb
tsth-StudentSelfService
tsth-tncommerce
tsth-nodeExporter
tsth-cAdvisor
(32) services
(43) tasks
dcompute scheduled tasks
tsth-banneremails-task-schedule
Lambda Functions
rave-rss
c2g
campustoursmaint
ec2-start-parkable
errors-remedy-to-slack
SecureAccess_ETL
pobox-sync
pwebsql1-ip-check
ebs-backup-worker
contracts_etl
emailer
trs-etl
wets
sa-log-clean
campus_tours_emailer
compcalc_etl
major-change
contact_methods_etl
snsToSlack
provost_etl
ebs-backup-prune
ec2-stop-parkable
TRS_Reminders
parking
(24)

DNS to ELB: Host based routing

Route 53 entry for each banner environment

- test
  - test.Infosys.etsu.edu
- bant
  - bant.infosys.etsu.edu
- prod
  - banner.infosys.etsu.edu



Name	DNS name	State
banner	banner-1516881703.us-east-1.elb.amazonaws.com	
bant	bant-1259912156.us-east-1.elb.amazonaws.com	
dalb1	internal-dalb1-1675399197.us-east-1.elb.amazonaws.com	active

Load balancer: bant

Description | **Instances** | Health check | Listeners | Monitoring | Tags | Migration

Connection Draining: Enabled, 400 seconds (Edit)

Edit Instances

Instance ID	Name	Availability Zone
i-07cc6d9cc4b1e4712	dproxy1	us-east-1a
i-058d3ccde71878f31	dproxy2	us-east-1b

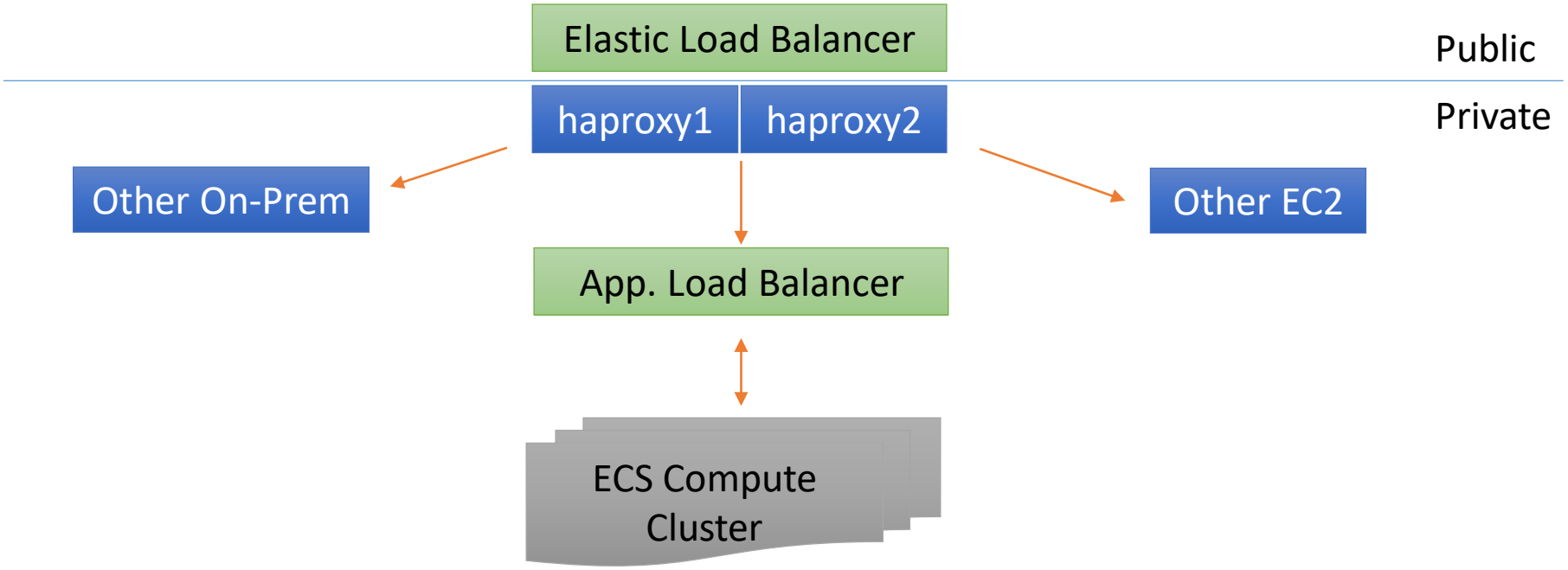
Description | Instances | Health check | **Listeners** | Monitoring | Tags | Migration

The following listeners are currently configured for this load balancer:

Load Balancer Protocol	Load Balancer Port	Instance Protocol	Instance Port	Cipher	SSL Certificate
HTTP	80	HTTP	80	N/A	N/A
HTTPS	443	HTTP	80	Change	02b414d1-f01e-47fe-96a4-d295e054ed60 (ACM) Change

elb **haproxy**

https://**banner**.infosys.etsu.edu/**applicationNavigator**



AWS Elastic Container Service

- Managed Cloud Service (Paas)
- Docker Cluster Service
- AWS Integrations
  - Elastic Container Registry (ECR)
  - Access Management (IAM)
  - Logging & Alerting (Cloudwatch)
  - Load Balancer (ALB)
- All Banner 9 Applications
- 24 Banner Services in Production

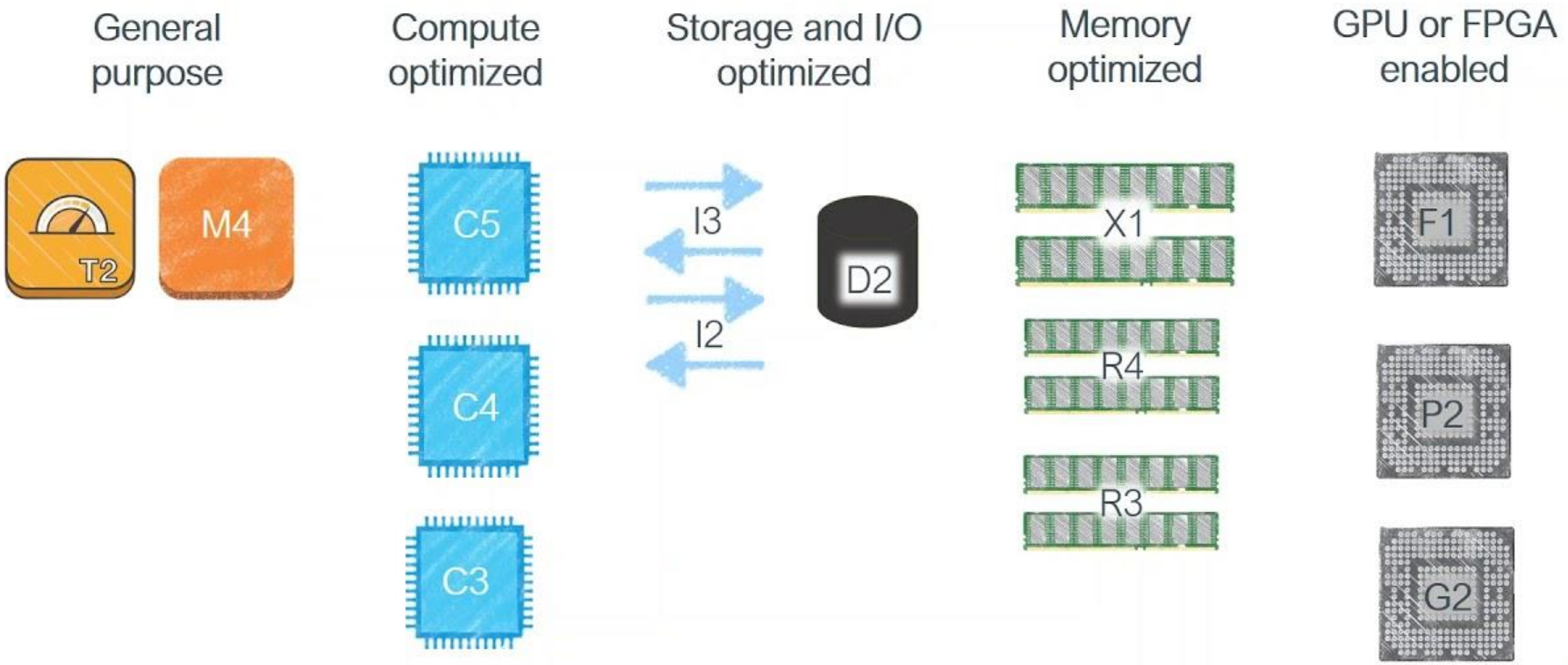
Banner Applications on EC2

- Oracle: Banner, ODS, Degreeworks, UC4
- Jobsub / UC4 / Runner / DGW Classic Server
- ESM
- BDM

- Informattica Agent (salesforce / Targetx)
- eInvoice

- EIS
- Rabbitmq

EC2: Instance Types



Relational Database Service (RDS)

- Non-Oracle
- Aurora
  - EIS : MySQL
  - .NET : Postgres
- MSSQL

Aurora (AWS database)

- Automatic, continuous, incremental
- Point-in-time restore
- No performance hit during backups
- 35 day retention

ORACLE®

Benefits

- VM performance
- S3 RMAN Backups
- AMI Snapshots
- EBS Snapshots

Limitations

- SYS/SYSTEM Locked
- OS: DIRECTORIES, etc
- DBA limitations
  - Not available: alter database, alter system, create any directory, drop any directory, grant any privilege, grant any role
- No shell access

Lambda

ec2-stop-parkable.py 780 Bytes

```python
import boto3
import collections
import datetime

ec2 = boto3.client('ec2')

def lambda_handler(event, context):
    reservations = ec2.describe_instances(
        Filters=[
            {'Name': 'tag-key', 'Values': ['parkable', 'Parkable']},
        ]
    ).get(
        'Reservations', []
    )

    instances = sum(
        [
            [i for i in r['Instances']]
            for r in reservations
        ], [])

    print "Found %d instances that need shutting down" % len(instances)

    for instance in instances:
        try:
            response = ec2.stop_instances(InstanceIds=[instance['InstanceId']], DryRun=False)
            print(response)
            print 'Stopped instance: ' + instance['InstanceId']
        except ClientError as e:
            print(e)
```

AMIs and Snapshots

- Snap AMIs quarterly or major upgrades
- Snap AMIs for install baselines
- Snapshots nightly
- Automated Backup and Prune via Lambda
- Migrate to Amazon Data Lifecycle Manager
- Pay for storage:
  - only deltas

ebs-backup-worker.py 1.83 KB

```python
import boto3
import collections
import datetime

ec = boto3.client('ec2')

def lambda_handler(event, context):
    reservations = ec.describe_instances(
        Filters=[
            {'Name': 'tag-key', 'Values': ['backup', 'Backup']},
        ]
    ).get(
        'Reservations', []
    )

    instances = sum(
        [
            [i for i in r['Instances']]
            for r in reservations
        ], [])

    print "Found %d instances that need backing up" % len(instances)

    to_tag = collections.defaultdict(list)

    for instance in instances:
        try:
            retention_days = [
                int(t.get('Value')) for t in instance['Tags']
                if t['Key'] == 'Retention'][0]
        except IndexError:
            retention_days = 7

        for dev in instance['BlockDeviceMappings']:
            if dev.get('Ebs', None) is None:
                continue
            vol_id = dev['Ebs']['VolumeId']
            print "Found EBS volume %s on instance %s" % (
                vol_id, instance['InstanceId'])

            snap = ec.create_snapshot(
                VolumeId=vol_id,
            )

            to_tag[retention_days].append(snap['SnapshotId'])
```

DR - Personal Health Dashboard

Event log

Add filter

Event	Status	Region/AZ	Start time	Last update time	Affected resources
EBS operational issue	Closed	us-east-1	August 31, 2019 at 7:33:00 AM UTC-4	September 3, 2019 at 12:26:08 AM UT…	1 entity
EC2 operational issue	Closed	us-east-1b	August 31, 2019 at 7:33:00 AM UTC-4	September 1, 2019 at 6:00:27 PM UTC-4	1 entity
EC2 operational issue	Closed	us-east-1	August 31, 2019 at 7:33:00 AM UTC-4	September 1, 2019 at 6:00:12 PM UTC-4	1 entity
RDS operational issue	Closed	us-east-1	August 31, 2019 at 10:16:06 AM UTC-4	August 31, 2019 at 10:23:14 PM UTC-4	-
WorkSpaces operational issue	Closed	us-east-1	August 31, 2019 at 9:55:52 AM UTC-4	August 31, 2019 at 5:02:19 PM UTC-4	-
EC2 operational issue	Closed	us-east-1	August 31, 2019 at 9:22:45 AM UTC-4	August 31, 2019 at 4:30:48 PM UTC-4	-

EBS operational issue

Close

Details | Affected resources

Add filter

Resource ID / ARN
vol-072ded7cf4be61f2f

S3: Simple Storage Service

ETSU Usage

- Rman backups
- Logs for pretty much every application
- Aws-cli on prem logs
- Ecs configs
  - Staged apps ready for deployment in docker cluster on base images
- Public file hosting
  - adminpages css
  - Appnav background images
- Temporary Storage
  - ETL

Features

- Tiered Storage Classes
- Lifecycle Policies
  - Durability
  - Cost
  - Retrieval time
- AWS Managed: Scalable
- Encryption
- Security concerns
  - Private by default (now)
- Security Policies
- IAM Roles
- Access Logging
- Versioning
- Analytics

Storage Gateway

- File Gateway
- NFS Mount
- S3 backend
- Cache

TouchNet

- IP Address must be whitelisted by TouchNet
- IP address may change for LB
- TouchNet will not honor that change
- EC2 Instance of HAProxy in public subnet with static ip
- Routes touchnet traffic (based on route)
  - SSB (alb -> ecs)
  - Pci interface (alb -> ecs)
  - UPay .NET web server

Monitoring

- Prometheus
- Alerts Manager / Slack
- Graphana
- Enterprise Manager


HAProxy
Today so far
Refresh every 5m
Basic General Info
Total responses by HTTP code
40
30
20
10
0
-10
-20
00:00
02:00
04:00
06:00
08:00
10:00
12:00
14:00
min max avg current
Frontend 1xx 0 0 0 0
Frontend 2xx 0 34 11 18
Frontend 3xx 0 3 0 0
Frontend 4xx 0 2 1 1
Frontend 5xx 0 0 0 0
Frontend other 0 0 0 0
Backend 1xx 0 0 0 0
Backend 2xx 0 15 2 1
Backend 3xx 0 0 0 0
Backend 4xx 0 0 0 0
Backend 5xx 0 0 0 0
Backend other 0 0 0 0
Current total of incoming / outgoing bytes
977 KiB
0 B
-977 KiB
-1.9 MiB
-2.9 MiB
-3.8 MiB
00:00
02:00
04:00
06:00
08:00
10:00
12:00
14:00
Total number of connections
2e-15
1e-15
0
-1e-15
-2e-15

Filter events

Time (UTC -04:00)	Message
2018-09-13	
09:08:54	13-Sep-2018 13:08:54.751 INFO [main] org.apache.coyote.AbstractProtocol.init Initializing ProtocolHandler ["http-apr-8080"]
09:08:54	13-Sep-2018 13:08:54.768 INFO [main] org.apache.coyote.AbstractProtocol.init Initializing ProtocolHandler ["ajp-apr-8009"]
09:08:54	13-Sep-2018 13:08:54.773 INFO [main] org.apache.catalina.startup.Catalina.load Initialization processed in 1229 ms
09:09:00	13-Sep-2018 13:09:00.083 INFO [main] org.apache.catalina.core.StandardService.startInternal Starting service Catalina
09:09:00	13-Sep-2018 13:09:00.083 INFO [main] org.apache.catalina.core.StandardEngine.startInternal Starting Servlet Engine: Apache Tomcat/8.0
09:09:00	13-Sep-2018 13:09:00.120 INFO [localhost-startStop-1] org.apache.catalina.startup.HostConfig.deployWAR Deploying web application arc
09:09:21	13-Sep-2018 13:09:21.145 INFO [localhost-startStop-1] org.apache.jasper.servlet.TldScanner.scanJars At least one JAR was scanned for
	13-Sep-2018 13:09:21.145 INFO [localhost-startStop-1] org.apache.jasper.servlet.TldScanner.scanJars At least one JAR was scanned for TLDs yet contained no TLDs. were scanned but no TLDs were found in them. Skipping unneeded JARs during scanning can improve startup time and JSP compilation time.
09:09:21	configuration: classpath:bannerHelp_configuration.groovy
09:09:21	configuration: file:/root/.grails/bannerHelp_configuration.groovy
09:09:21	log4j:WARN No appenders could be found for logger (org.codehaus.groovy.grails.commons.cfg.ConfigurationHelper).
09:09:21	log4j:WARN Please initialize the log4j system properly.
09:09:21	log4j:WARN See http://logging.apache.org/log4j/1.2/faq.html#noconfig for more info.
09:09:24	configuration: classpath:bannerHelp_configuration.groovy
09:09:24	configuration: file:/root/.grails/bannerHelp_configuration.groovy
09:09:37	13-Sep-2018 13:09:37.308 INFO [localhost-startStop-1] org.apache.catalina.startup.HostConfig.deployWAR Deployment of web applicatio
09:09:37	13-Sep-2018 13:09:37.315 INFO [localhost-startStop-1] org.apache.catalina.startup.HostConfig.deployWAR Deploying web application arc
09:09:37	13-Sep-2018 13:09:37.321 WARNING [localhost-startStop-1] org.apache.catalina.startup.SetContextPropertiesRule.begin [SetContextPro
09:09:38	13-Sep-2018 13:09:38.036 INFO [localhost-startStop-1] org.apache.jasper.servlet.TldScanner.scanJars At least one JAR was scanned for
09:09:38	13-Sep-2018 13:09:38.045 INFO [localhost-startStop-1] org.apache.catalina.startup.HostConfig.deployWAR Deployment of web applicatio
09:09:38	13-Sep-2018 13:09:38.046 INFO [localhost-startStop-1] org.apache.catalina.startup.HostConfig.deployWAR Deploying web application arc
09:10:08	13-Sep-2018 13:10:08.206 INFO [localhost-startStop-1] org.apache.jasper.servlet.TldScanner.scanJars At least one JAR was scanned for
09:10:27	13-Sep-2018 13:10:27.993 INFO [localhost-startStop-1] org.apache.catalina.startup.HostConfig.deployWAR Deployment of web applicatio
09:10:28	13-Sep-2018 13:10:28.001 INFO [main] org.apache.coyote.AbstractProtocol.start Starting ProtocolHandler ["http-apr-8080"]
09:10:28	13-Sep-2018 13:10:28.017 INFO [main] org.apache.coyote.AbstractProtocol.start Starting ProtocolHandler ["ajp-apr-8009"]
09:10:28	13-Sep-2018 13:10:28.028 INFO [main] org.apache.catalina.startup.Catalina.start Server startup in 93254 ms

On-Prem

- Evisions MAPS
- Intellicheck
- INB
- Dataguard
- FSAAtlas

Summary

- Take inventory of current resources – network, hardware, software, personnel
- Get Comfortable with AWS services
  - Free Tier
- Learn how to use the AWS cost estimator
- Use Terraform and git
- AWS Well Architected tool

Whats Next?

- Containerize more apps
- Puppetize more vm configuration
- Create standby in separate region
- Increase usage of CI/CD
- Increase monitoring via Prometheus and alerting
- RDS: UC4P, ADBP, RCAT, DGWP
- Greenfield
  - AWS Fargate

Questions?

Resources

- http://www.gabrieltocci.com/talks
- BannerInTheCloud: AWS Group
  - https://bannerinthecloud.slack.com