AWS Certified SysOps : Associate

Domain 4
Deployment and Provisioning

Eric L. Santelices
Senior Architect
Emerging Technologies
RoundTower Technologies

Twitter: @Digital_Roadies

RoundTower Technologies®

Eric’s Exam tips

- One challenge is knowing the point in time for the question
- The exam is a reading comprehension exam more then it is a technical exam
- Rule out the “way out” questions
- Pick the most right answer
- Get Qwiklabs / Hands on experience
- Read all the FAQ’s
- Read all the White Papers

Domain 4 - Objectives

Domain 4 from the AWS SysOps Administrator Exam Blueprint
15% of Score

Focusing on being able to:

- 4.1 Demonstrate the ability to build the environment to conform with the architected design

- 4.2 Demonstrate the ability to provision cloud resources and manage implementation automation

Elastic Load Balancing

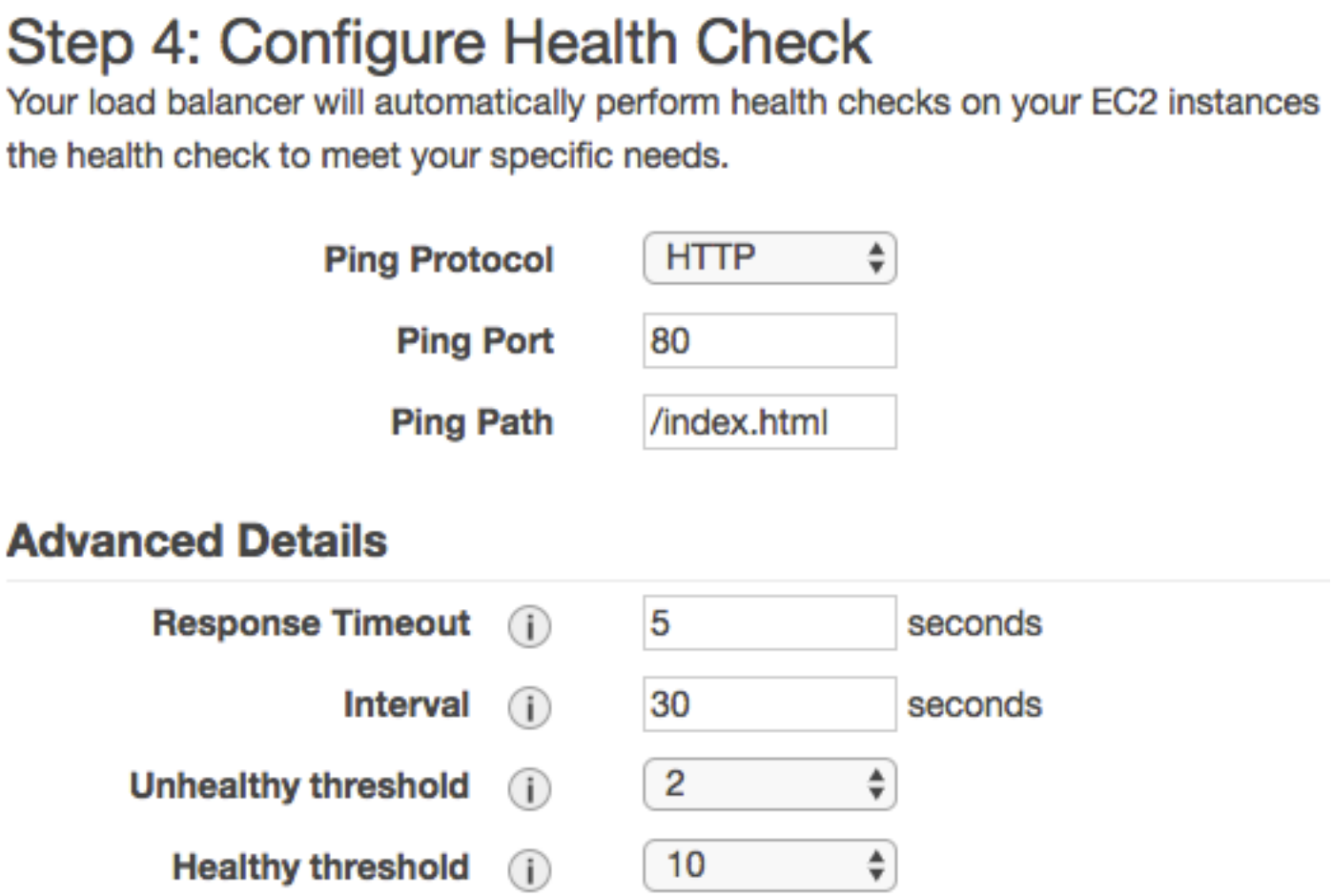
Classic Load Balancing (OSI Layer 4)

A Classic Load Balancer is ideal for simple load balancing of traffic across multiple EC2 instances, while an Application Load Balancer is ideal for microservices or container-based architectures where there is a need to route traffic to multiple services or load balance across multiple ports on the same EC2 instance.

Supported Protocols: HTTP, HTTPS, TCP, SSL

TCP Ports: [EC2-VPC] 1-65535; [EC2-Classic] 25, 80, 443, 465, 587, 1024-65535

Elastic Load Balancing



RoundTower Technologies

Elastic Load Balancing

Metrics

- **SurgeQueueLength –** A count of the total number of requests that are pending submission to a registered instance.
- **SpilloverCount –** A count of the total number of requests that are pending submission to a registered instance.
- **Latency –** Measures the time elapsed in seconds after the request leaves the load balancer until the response is received.

Elastic Load Balancing

Elastic Load Balancing has features that support sticky sessions (also known as *session affinity*) using cookies. If the elastic load balancer has sticky sessions enabled, this traffic will be routed to the same back-end instances as the user continues to access your application. When you design your load tests and are using sticky sessions, it will be important to decide how you will test this feature. Consider how sticky sessions can be an issue in both load testing and in the real world.

- **LB generated cookie –** Sessions cookies are managed by the ELB, requires expiration period

- **Application generated cookie –** Session cookies are managed by the Application, requires cookie name

Elastic Load Balancing

Pre-warming of ELB

Amazon ELB is able to handle the vast majority of use cases for our customers without requiring "pre-warming" (configuring the load balancer to have the appropriate level of capacity based on expected traffic). In certain scenarios, such as when flash traffic is expected, or in the case where a load test cannot be configured to gradually increase traffic, we recommend that you AWS to have your load balancer "pre-warmed". We will then configure the load balancer to have the appropriate level of capacity based on the traffic that you expect. We will need to know the start and end dates of your tests or expected flash traffic, the expected request rate per second and the total size of the typical request/response that you will be testing.

Sample Question from Amazon

Your web site is hosted on 10 EC2 instances in 5 regions around the globe with 2 instances per region. How could you configure your site to maintain site availability with minimum downtime if one of the 5 regions was to lose network connectivity for an extended period of time?

A. Create an Elastic Load Balancer to place in front of the EC2 instances. Set an appropriate health check on each ELB.

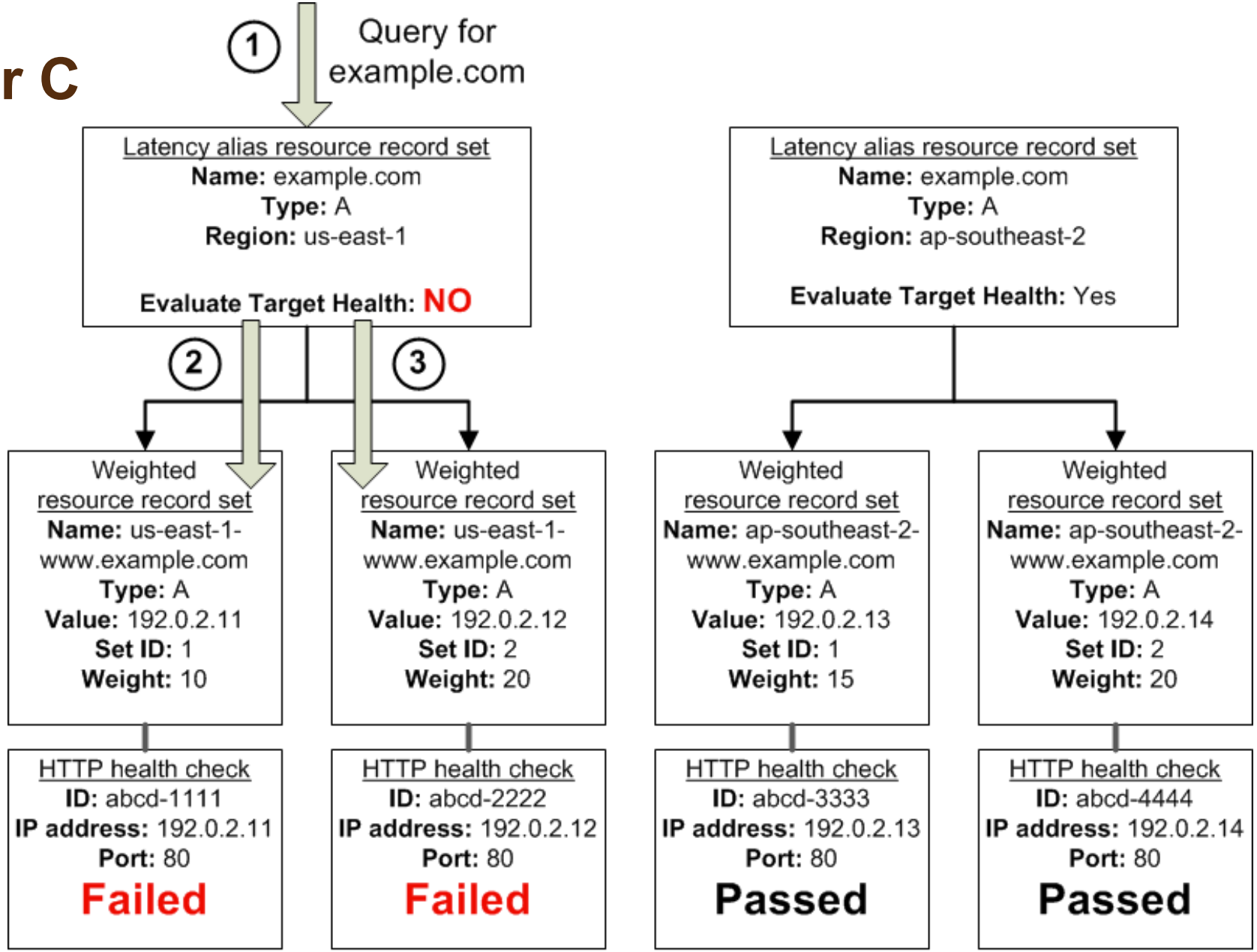
B. Establish VPN Connections between the instances in each region. Rely on BGP to failover in the case of a region wide connectivity outage

C. Create a Route 53 Latency Based Routing Record Set that resolves to an Elastic Load Balancer in each region. Set an appropriate health check on each ELB.

D. Create a Route 53 Latency Based Routing Record Set that resolves to Elastic Load Balancers in each region and has the Evaluate Target Health flag set to true.

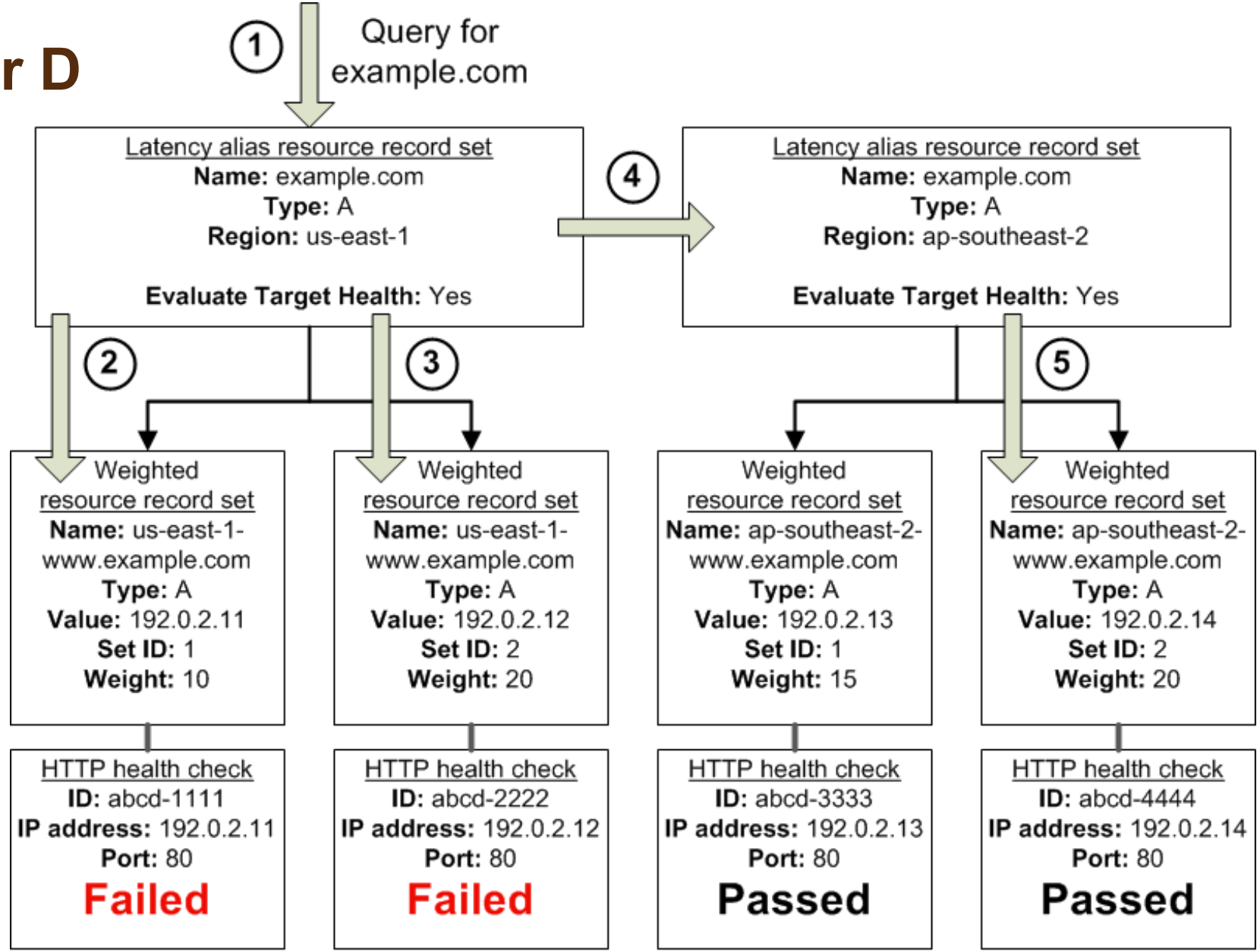
Question from AWS Exam Sample:
http://awstrainingandcertification.s3.amazonaws.com/production/AWS_certified_sysops_associate_examsample.pdf

Answer C



RoundTower Technologies

Answer D



RoundTower Technologies

OpsWorks

OpsWorks for Chef Automate vs Stacks

AWS OpsWorks for Chef Automate lets you create AWS-managed Chef servers that include **Chef Automate** premium features, and use the Chef DK and other Chef tooling to manage them.

AWS OpsWorks Stacks, the original service, provides a simple and flexible way to create and manage stacks and applications. AWS OpsWorks Stacks lets you deploy and monitor applications in your stacks.

OpsWorks Stacks

AWS OpsWorks Stacks benefits

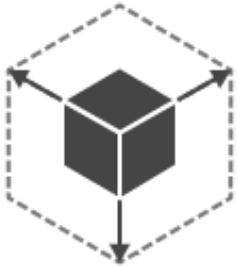
Support any application

AWS OpsWorks Stacks supports a wide variety of architectures, from simple web applications to highly complex custom applications running on Linux or Windows.

Configuration as code

AWS OpsWorks Stacks lets you define and maintain configurations for your entire environment in code and lets you provision your instances with Chef.

Automation to run at scale



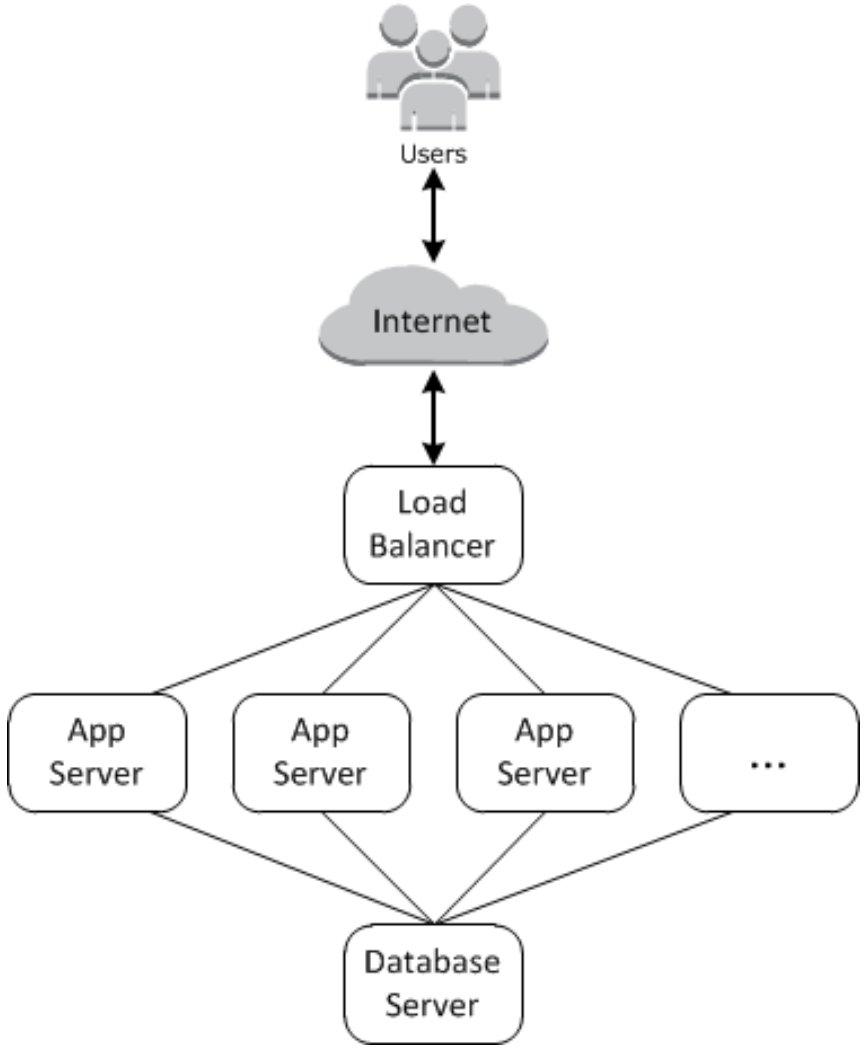
AWS OpsWorks Stacks enables you to efficiently manage your applications over their lifetime, including support for automatic instance scaling and auto healing.

Resource organization

AWS OpsWorks Stacks lets you model and visualize your application using concepts such as stacks, layers, and apps. You can also manage your users and resource access on all your instances using AWS IAM.

RoundTower Technologies

OpsWorks Stacks



Multi Tiered

Code Deployment

Package Installations

Database Connection Strings

Automation to Scale on Time or Load

Permissions and Policy Management

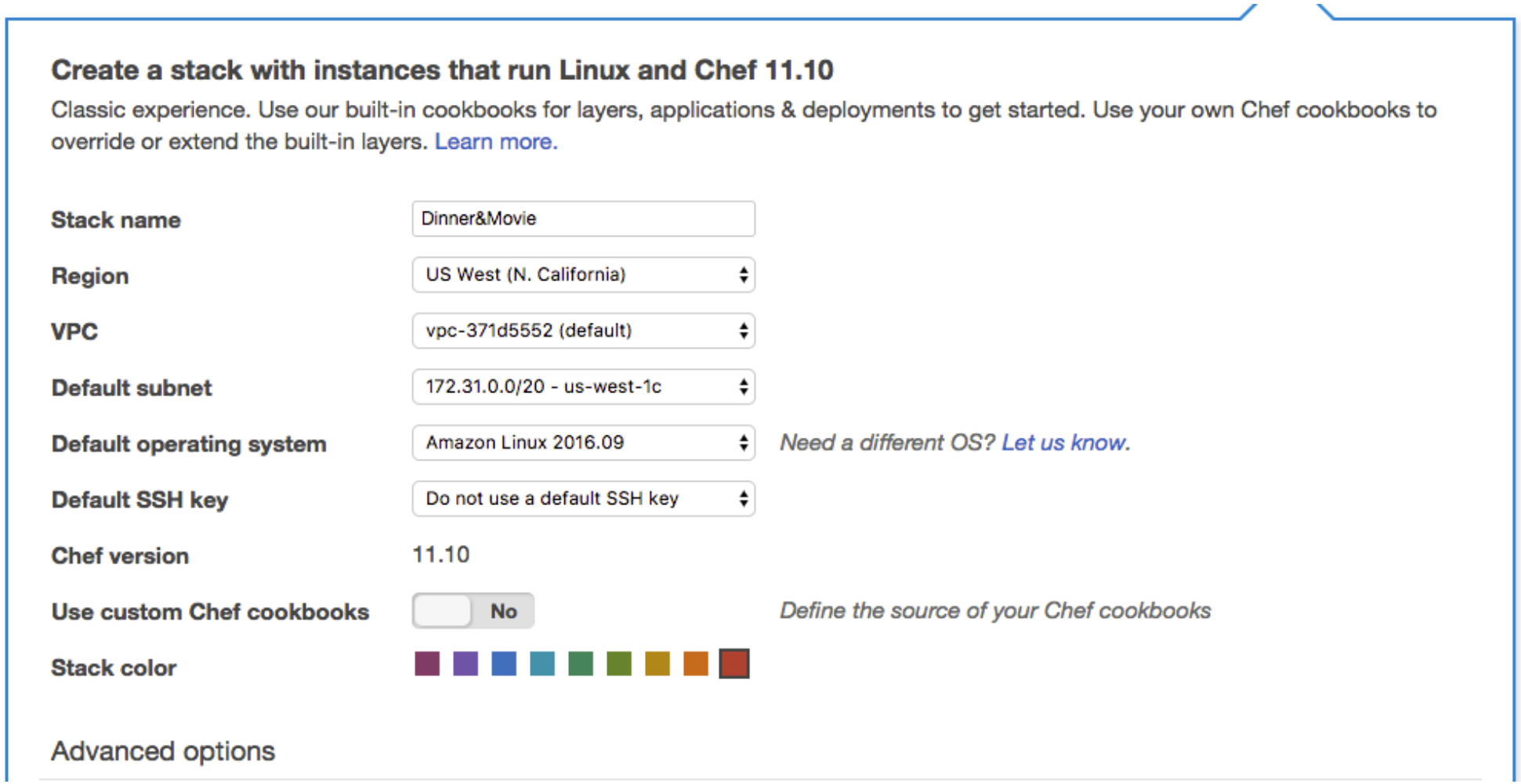
Based on Chef - uses chef-client

Stacks – is the entire thing on left (load balancer, App Server, Database Server)

Layers – is a single group of things on left (App Server or Load Balancer)

RoundTower Technologies

OpsWorks Stacks



RoundTower Technologies

Advanced options

Default root device type
- ○ EBS backed
- ● Instance store

IAM role New IAM role

Default IAM instance profile S3_access

API endpoint region NEW
- ● us-west-1 REGIONAL
- ○ us-east-1 CLASSIC

Hostname theme Layer Dependent

OpsWorks Agent version 3444 (Apr 1st 2017)

Custom JSON Optional

Enter custom JSON that is passed to your Chef recipes for all instances in your stack. You can use this to override and customize built-in recipes or pass variables to your own recipes. Learn more.

Security

Use OpsWorks security groups Yes

Cancel **Add stack**

OpsWorks Stacks

Add layer

Layer type Node.js App Server

The Node.js Application Server layer is a blueprint for instances that function as JavaScript application servers. Learn more.

Node.js version 0.12.18

Elastic Load Balancer No ELBs have been created in your vpc-371d5552 in us-west-1. To add an ELB go to the EC2 console.

Need further support? Let us know.

Cancel **Add layer**

RoundTower Technologies

OpsWorks Stacks

Add layer

OpsWorks | ECS | RDS

Layer type HAProxy

An HAProxy layer is a blueprint for instances that expose a single IP address to represent a set of application servers. It receives incoming requests, distributes them across the application server instances, and returns responses to the caller. Learn more.

HAProxy statistics Yes

Statistics URL	/haproxy?stats
Statistics user name	opsworks
Statistics password	ox5m6lliic
Health check URL	/
Health check method	OPTIONS

Need further support? Let us know.

Cancel **Add layer**

OpsWorks Stacks

Layers

Add layer

HAProxy

Settings Recipes Network EBS Volumes Security

Delete

Instances

2

Node.js App Server

Settings Recipes Network EBS Volumes Security

Delete

Instances

3

RDS: dinnermovie

Details

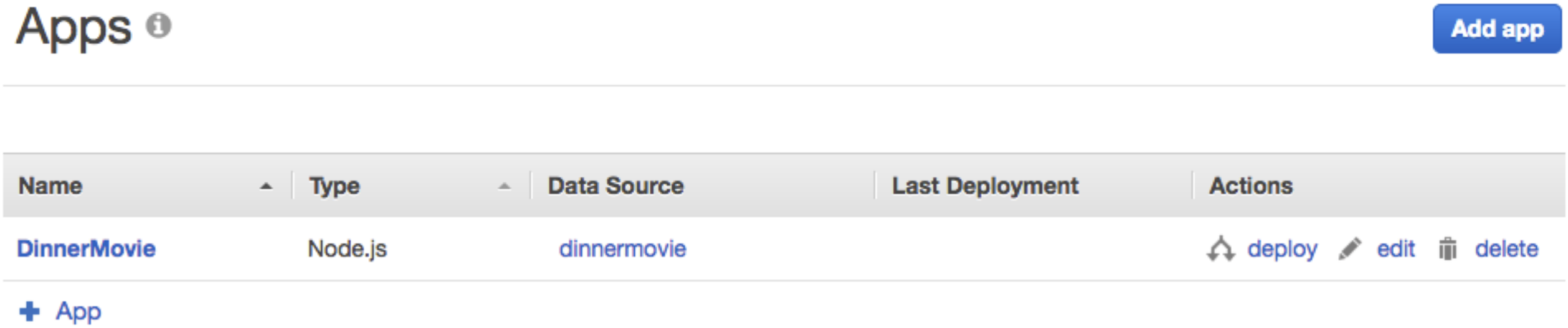
Apps

1

+ Layer

RoundTower Technologies

OpsWorks Stacks



RoundTower Technologies

OpsWorks | Stacks > Dinner-Movie

- Stack
- Layers
- Instances
  - Time-based
  - Load-based
- Apps
- Deployments
- Monitoring
- Resources
- Permissions

Stacks

Users

My Settings

Dinner-Movie REGIONAL

Run Command | Stack Settings | Delete Stack

A stack represents a collection of EC2 instances and related AWS resources that have a common purpose and that you want to manage collectively. Within a stack, you use layers to define the configuration of your instances and use apps to specify the code you want to deploy. Learn more.

Layers 2

HAProxy

Node.js App Server

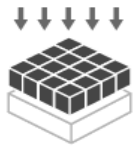
Instances 5

0	0	0	5	0
online	setting up	shutting down	stopped	error

Apps 1

DinnerMovie deploy

Deployments and Commands



You can deploy the code from your repository to the appropriate server or run commands on some or all instances in your stack.

Deploy an app or **run a command**

Resources

Volumes 0

RDS DB Instances 1

Elastic IPs 0

Monitoring

AWS OpsWorks uses Amazon CloudWatch to provide thirteen custom metrics with detailed monitoring for each instance in the stack.

Show monitoring

Permissions

Permissions specify how imported IAM users can access this stack. To import users, go to the **Users** page.

Manage permissions

RoundTower Technologies

OpsWorks Stacks: Clean up

Filter: All security groups | Search Security Groups and t

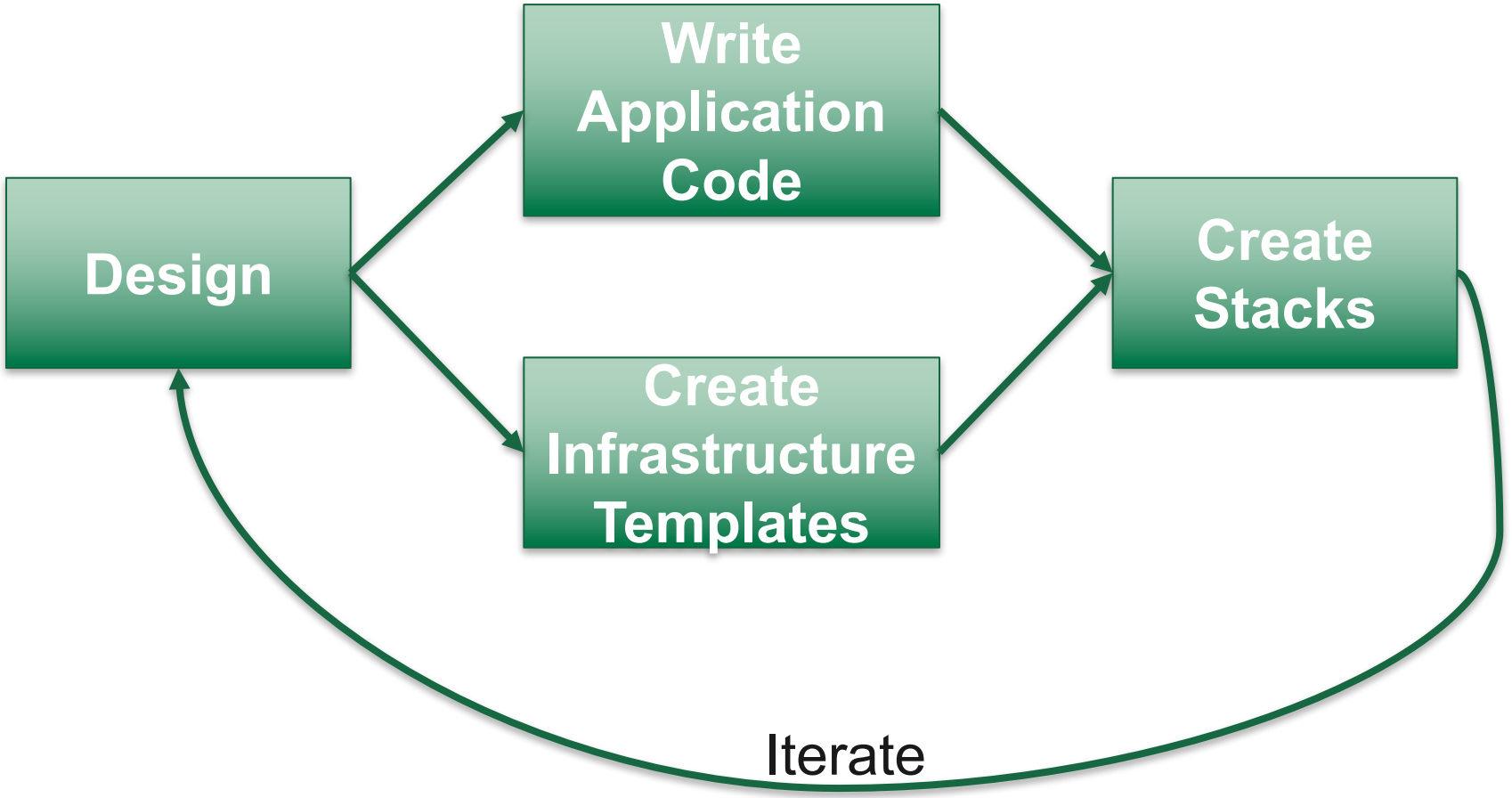
Name tag	Group ID	Group Name	VPC	Description
	sg-60d6a507	AWS-OpsWorks-AWS...	vpc-371d5552	AWS Flow Ruby server - do not change or delete
	sg-4ad0a32d	AWS-OpsWorks-Blan...	vpc-371d5552	AWS OpsWorks blank server - do not change or delete
	sg-48d0a32f	AWS-OpsWorks-Cust...	vpc-371d5552	AWS OpsWorks custom server - do not change or delete
	sg-c6d2a1a1	AWS-OpsWorks-DB-...	vpc-371d5552	AWS OpsWorks database master server - do not change or delete
	sg-00d6a567	AWS-OpsWorks-Defa...	vpc-371d5552	AWS OpsWorks Default server - do not change or delete
	sg-f8d7a49f	AWS-OpsWorks-ECS...	vpc-371d5552	AWS OpsWorks ECS cluster - do not change or delete
	sg-f9d7a49e	AWS-OpsWorks-Java...	vpc-371d5552	AWS OpsWorks Java-App server - do not change or delete
	sg-bad5a6dd	AWS-OpsWorks-LB-S...	vpc-371d5552	AWS OpsWorks load balancer - do not change or delete
	sg-8bd0a3ec	AWS-OpsWorks-Mem...	vpc-371d5552	AWS OpsWorks Memcached server - do not change or delete
	sg-62d6a505	AWS-OpsWorks-Moni...	vpc-371d5552	AWS OpsWorks Monitoring Ganglia server - do not change or delete
	sg-61d6a506	AWS-OpsWorks-node...	vpc-371d5552	AWS OpsWorks nodejs-App server - do not change or delete
	sg-66d6a501	AWS-OpsWorks-PHP...	vpc-371d5552	AWS OpsWorks PHP-App server - do not change or delete
	sg-27d7a440	AWS-OpsWorks-Rails...	vpc-371d5552	AWS OpsWorks Rails-App server - do not change or delete
	sg-7ed1a219	AWS-OpsWorks-RDP...	vpc-371d5552	AWS OpsWorks RDP server
	sg-7fd5a618	AWS-OpsWorks-Web...	vpc-371d5552	AWS OpsWorks Web server - do not change or delete
	sg-8a44e0ee	default	vpc-371d5552	default VPC security group
	sg-b6d6a5d1	rds-launch-wizard	vpc-371d5552	Created from the RDS Management Console

CloudFormation

Q: What is AWS CloudFormation?

AWS CloudFormation is a service that gives developers and businesses an easy way to create a collection of related AWS resources and provision them in an orderly and predictable fashion.

CloudFormation : Infrastructure as Code



RoundTower Technologies

CloudFormation

Q: How is AWS CloudFormation different from AWS Elastic Beanstalk?

These services are designed to complement each other. AWS Elastic Beanstalk provides an environment to easily deploy and run applications in the cloud. It is integrated with developer tools and provides a one-stop experience for you to manage the lifecycle of your applications. AWS CloudFormation is a convenient provisioning mechanism for a broad range of AWS resources. It supports the infrastructure needs of many different types of applications such as existing enterprise applications, legacy applications, applications built using a variety of AWS resources and container-based solutions (including those built using AWS Elastic Beanstalk).

AWS CloudFormation supports Elastic Beanstalk application environments as one of the AWS resource types. This allows you, for example, to create and manage an AWS Elastic Beanstalk–hosted application along with an RDS database to store the application data. In addition to RDS instances, any other supported AWS resource can be added to the group as well.

CloudFormation

- **Stack** - CloudFormation unit of grouping for infrastructure
- **Template** - a JSON document given to CloudFormation with instructions on how to act and what to create. Note: A template can be used to create and/or update a stack
- **Stack Policy** - IAM style policy statement which governs what can be changed and by who; policy's cannot be removed but can be updated after created

Anatomy of CloudFormation

- **Parameters** - allow the passing of variables into a template
- **Mappings** - allow processing of hash's (array's of key value pairs) by the cfnTemplate (like w2k AMI by region)
- **Resources** - where your actual resources are declared
- **Outputs** - results from the template

CloudFormation: S3 Bucket

```
{
        "Resources"  :  {
                "HelloBucket"  :  {
                        "Type" : "AWS::S3::Bucket”
                }
        }
}
```

CloudFormation: S3 Bucket w/Properties

```
{
        "Resources"  :  {
                "HelloBucket"  :  {
                        "Type" : "AWS::S3::Bucket”,
                        “Properties” : {
                                “AccessControl” : “PublicRead”
                        }
                }
        }
}
```


RoundTower Technologies

CloudFormation: Mappings

```
{   "Parameters" : {
      "KeyName" : {
        "Description" : "Name of an  existing EC2 KeyPair to enable SSH access to the instance",
        "Type" : "String”
      }
   },
   "Mappings" : {
     "RegionMap" : {
       "us-east-1" : { "AMI" : "ami-76f0061f” },
       "us-west-1" : { "AMI" : "ami-655a0a20” },
       "eu-west-1" : { "AMI" : "ami-7fd4e10b” },
       "ap-southeast-1" : { "AMI" : "ami-72621c20” },
       "ap-northeast-1" : { "AMI" : "ami-8e08a38f” }
     }
   },
   "Resources" : {
     "Ec2Instance" : {
       "Type" : "AWS::EC2::Instance",
       "Properties" : {
         "KeyName" : { "Ref" : "KeyName" },
         "ImageId" : { "Fn::FindInMap" : [ "RegionMap", { "Ref" : "AWS::Region" }, "AMI" ]},
         "UserData" : { "Fn::Base64" : "80" }
       }     }   }}
```


RoundTower Technologies

Summary

- Remember timeframe for Questions on exam
- Rule out the “way out” questions
- Get hands on with Free Tier
- Read AWS WhitePapers
- Read AWS CloudFormation FAQ
- Read AWS OpsWorks FAQ
- Read AWS Classic Load Balancer FAQ

Thank You

RoundTower Technologies®