



Masterclass

....

AWS CloudFormation

Ian Massingham – Technical Evangelist

 @ianMmmm

Masterclass

A technical deep dive beyond the basics

Help educate you on how to get the best from AWS technologies

Show you how things work and how to get things done

Broaden your knowledge in ~45 mins

AWS CloudFormation

An easy way to create & manage a collection of AWS resources

Allows orderly and predictable provisioning and updating of resources

Allows you to version control your AWS infrastructure

Deploy and update stacks using console, command line or API

Available at no additional charge, you only pay for the resources you create

Transparent and Open

Don't reinvent the wheel

Declarative & Flexible

CloudFormation

No Extra Charge

Customized via Parameters

Integration Ready

CloudFormation – Components & Technology

Template



JSON formatted file

Parameter definition

Resource creation

Configuration actions

CloudFormation



Framework

Stack creation

Stack updates

Error detection and rollback

Stack



Configured AWS services

Comprehensive service support

Service event aware

Customisable

Creating Templates

Using a Template to Create and Manage a Stack

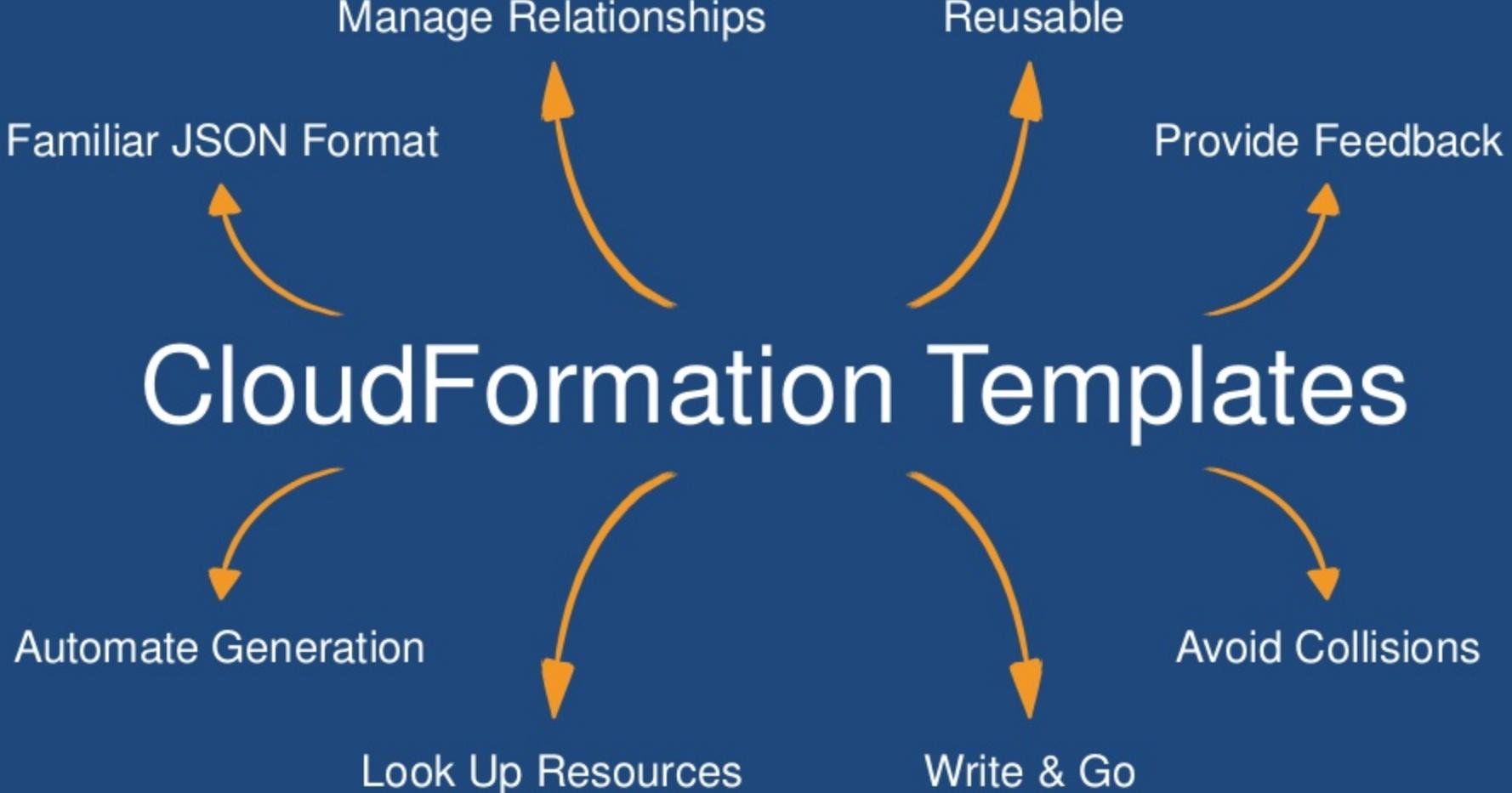
Working with the CloudFormation API

Working with AWS Resources

Bootstrapping Applications and Handling Updates

Creating Templates

CloudFormation Templates



High Level Template Structure

```
{  
    "Description" : "A text description for the template usage",  
    "Parameters": {  
        // A set of inputs used to customize the template per deployment  
    },  
    "Resources" : {  
        // The set of AWS resources and relationships between them  
    },  
    "Outputs" : {  
        // A set of values to be made visible to the stack creator  
    },  
    "AWSTemplateFormatVersion" : "2010-09-09"  
}
```

A Simple Template that creates an EC2 Instance

```
{  
    "Description": "Create an EC2 instance running the latest Amazon Linux AMI.",  
    "Parameters": {  
        "KeyPair": {  
            "Description": "The EC2 Key Pair to allow SSH access to the instance",  
            "Type": "String"  
        }  
    },  
    "Resources": {  
        "Ec2Instance": {  
            "Properties": {  
                "ImageId": "ami-dd925baa",  
                "InstanceType" : "m3.medium",  
                "KeyName": {  
                    "Ref": "KeyPair"  
                }  
            },  
            "Type": "AWS::EC2::Instance"  
        }  
    },  
    "Outputs": {  
        "InstanceId": {  
            "Description": "The InstanceId of the newly created EC2 instance",  
            "Value": {  
                "Ref": "Ec2Instance"  
            }  
        }  
    },  
    "AWSTemplateFormatVersion": "2010-09-09"  
}
```

A Simple Template that creates an EC2 Instance

```
{  
    "Description": "Creates an EC2 instance running the latest Amazon Linux AMI",  
    "Parameters": {  
        "KeyPair": {  
            "Description": "The EC2 Key Pair to allow SSH access to the instance",  
            "Type": "String"  
        }  
    },  
    "Resources": {  
        "Ec2Instance": {  
            "Properties": {  
                "ImageId": "ami-dd925baa",  
                "InstanceType": "m3.medium",  
                "KeyName": {  
                    "Ref": "KeyPair"  
                }  
            },  
            "Type": "AWS::EC2::Instance"  
        }  
    },  
    "Outputs": {  
        "InstanceId": {  
            "Description": "The InstanceId of the newly created EC2 instance",  
            "Value": {  
                "Ref": "Ec2Instance"  
            }  
        }  
    },  
    "AWSTemplateFormatVersion": "2010-09-09"  
}
```



You will be asked to enter values for these parameters when you create your stack

A Simple Template that creates an EC2 Instance

```
{  
    "Description": "Create an EC2 instance running the latest Amazon Linux AMI.",  
    "Parameters": {  
        "KeyPair": {  
            "Description": "The EC2 Key Pair to allow SSH access to the instance",  
            "Type": "String"  
        }  
    },  
    "Resources": {  
        "Ec2Instance": {  
            "Properties": {  
                "ImageId": "ami-dd925baa",  
                "InstanceType" : "m3.medium",  
                "KeyName": {  
                    "Ref": "KeyPair"  
                },  
                "Type": "AWS::EC2::Instance"  
            },  
        },  
        "Outputs": {  
            "InstanceId": {  
                "Description": "The InstanceId of the newly created EC2 instance",  
                "Value": {  
                    "Ref": "Ec2Instance"  
                }  
            }  
        },  
        "AWSTemplateFormatVersion": "2010-09-09"  
    }  
}
```

Includes statically defined properties (ImageID & Instance Type) plus a reference to the KeyPair parameter. ImageID is the AMI specific to the region that you will launch this stack in

A Simple Template that creates an EC2 Instance

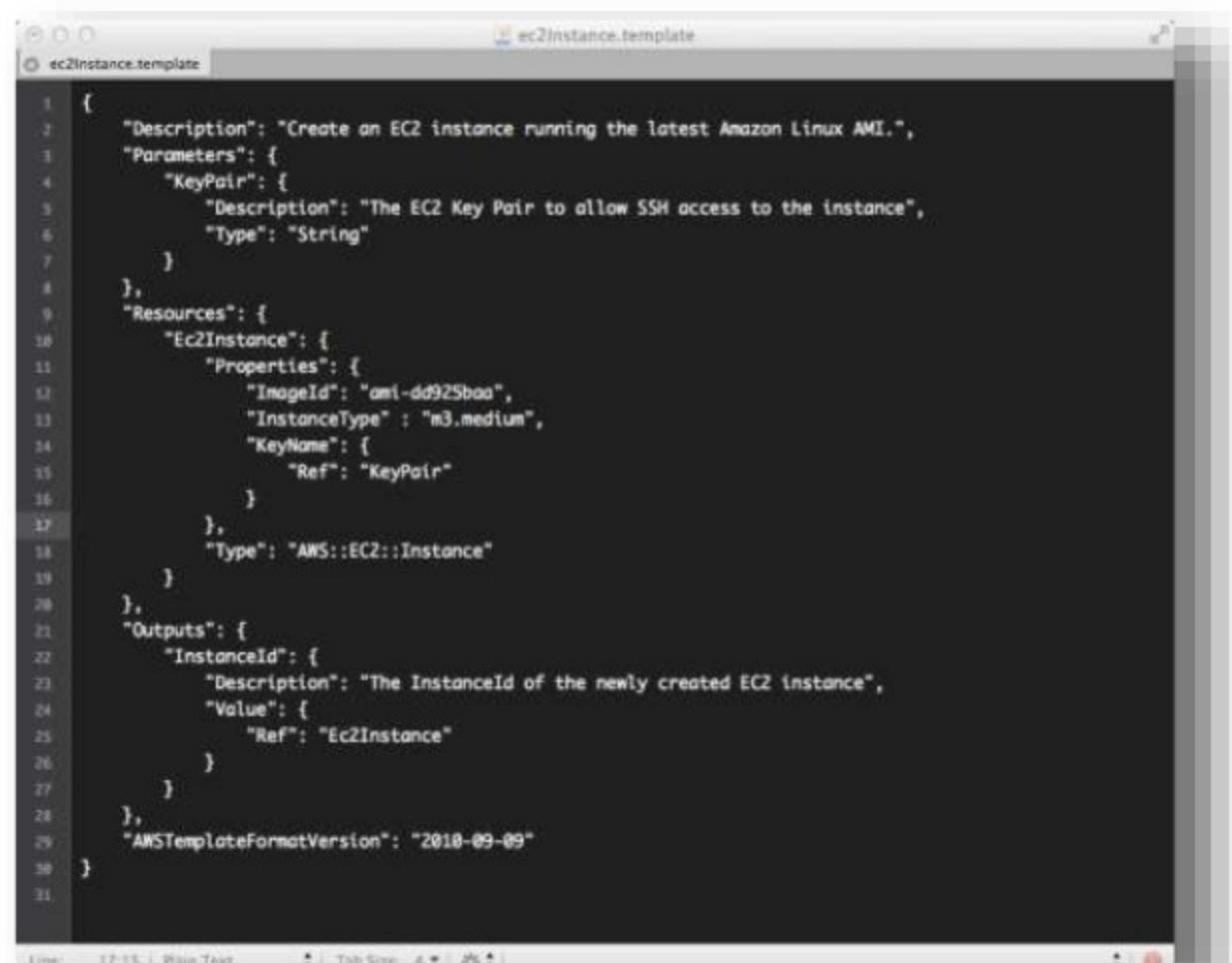
```
{  
    "Description": "Create an EC2 instance running the latest Amazon Linux AMI.",  
    "Parameters": {  
        "KeyPair": {  
            "Description": "The EC2 Key Pair to allow SSH access to the instance",  
            "Type": "String"  
        }  
    },  
    "Resources": {  
        "Ec2Instance": {  
            "Properties": {  
                "ImageId": "ami-dd925baa",  
                "InstanceType" : "m3.medium",  
                "KeyName": {  
                    "Ref": "KeyPair"  
                }  
            },  
            "Type": "AWS::EC2::Instance"  
        }  
    },  
    "Outputs": {  
        "InstanceId": {  
            "Description": "The InstanceId of the newly created EC2 instance",  
            "Value": {  
                "Ref": "Ec2Instance"  
            }  
        }  
    }  
}  
AWS::TemplateFormatVersion : 2010-09-09
```

These outputs will be returned once the template has completed execution



Creating & Managing Stacks

Using a template to create and manage a stack



The screenshot shows a code editor window with the title bar "ec2Instance.template". The main content area displays the following AWS CloudFormation template:

```
1 {
2     "Description": "Create an EC2 instance running the latest Amazon Linux AMI.",
3     "Parameters": {
4         "KeyPair": {
5             "Description": "The EC2 Key Pair to allow SSH access to the instance",
6             "Type": "String"
7         }
8     },
9     "Resources": {
10        "Ec2Instance": {
11            "Properties": {
12                "ImageId": "ami-dd925baa",
13                "InstanceType": "m3.medium",
14                "KeyName": {
15                    "Ref": "KeyPair"
16                }
17            },
18            "Type": "AWS::EC2::Instance"
19        }
20    },
21    "Outputs": {
22        "InstanceId": {
23            "Description": "The InstanceId of the newly created EC2 instance",
24            "Value": {
25                "Ref": "Ec2Instance"
26            }
27        }
28    },
29    "AWSTemplateFormatVersion": "2010-09-09"
30}
31
```

The template defines a single resource, "Ec2Instance", which creates an EC2 instance with the specified properties. It also defines an output, "InstanceId", which refers to the "Ref" of the "Ec2Instance" resource. The template is versioned at "2010-09-09".

Using a template to create and manage a stack

AWS Management Console

Services VPC EC2 S3 RDS CloudFront Edit iammas@iammas-aws Ireland Help

Amazon Web Services

Compute & Networking Database Analytics

- Direct Connect
- EC2
- Route 53
- VPC
- WorkSpaces
- DynamoDB
- ElastiCache
- RDS
- Redshift

Storage & Content Delivery Deployment & Management

- CloudFront
- Glacier
- S3
- Storage Gateway
- CloudFormation
- CloudTrail
- CloudWatch
- Elastic Beanstalk
- IAM
- OpenShift

App Services

- AppStream
- CloudSearch
- Elastic Transcoder
- SES
- SNS
- SQS
- SWF

Additional Resources

Getting Started See our documentation to get started and learn more about how to use our services.

Trusted Advisor Best practice recommendations to save money, improve fault tolerance, increase performance, and close security gaps.

Service Health All services operating normally. Updated: Jun 29 2014 12:46:50 (GMT+0100)

Service Health Dashboard

Set Start Page

Console Home

AWS Marketplace Find & buy software, launch with 1-Click and pay by the hour.

© 2006 - 2014, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Feedback

Using a template to create and manage a stack

The screenshot shows the CloudFormation Management Console interface. At the top, there's a navigation bar with tabs for Services (selected), VPC, EC2, S3, RDS, CloudFront, Edit, and Help. The user is signed in as ianmas@ianmas-aws and is located in Ireland. A search bar at the top right says "Showing 0 stacks". Below the navigation, there are two main sections: "Create a Stack" and "Create a Template from your Existing Resources".

Create a Stack

AWS CloudFormation allows you to quickly and easily deploy your infrastructure resources and applications on AWS. You can use one of the templates we provide to get started quickly with applications like WordPress or Drupal, one of the many sample templates or create your own template.

You do not currently have any stacks. Click the "Create New Stack" button below to create a new AWS CloudFormation Stack.

Create New Stack

Create a Template from your Existing Resources

If you already have AWS resources running, the CloudFormer tool can create a template from your existing resources. This means you can capture and redeploy applications you already have running.

To do this, click Launch CloudFormer and create an AWS CloudFormation stack that runs the CloudFormer tool. After the stack creation is complete, navigate to the CloudFormer URL available on the Outputs tab.

Launch CloudFormer

At the bottom of the page, there are footer links for Privacy Policy and Terms of Use, and a Feedback button. There are also standard browser control buttons (back, forward, search) at the very bottom.

Using a template to create and manage a stack

Create A New Stack

Services VPC EC2 S3 RDS CloudFront Edit Ianmac @ ianmacs-aws Ireland Help

Select Template

Specify Parameters Options Review

Select Template

Specify a stack name and then select the template that describes the stack that you want to create.

Stack

An AWS CloudFormation stack is a collection of related resources that you provision and update as a single unit.

Name

Template

A template is a JSON-formatted text file that describes your stack's resources and their properties. AWS CloudFormation stores the stack's template in an Amazon S3 bucket. [Learn more](#).

Source Select a sample template
 Upload a template to Amazon S3 No file selected.
 Specify an Amazon S3 template URL

© 2008 - 2014, Amazon Web Services, Inc. or its affiliates. All rights reserved. [Privacy Policy](#) [Terms of Use](#) [Feedback](#)

Using a template to create and manage a stack

Create A New Stack

Services VPC EC2 S3 RDS CloudFront Edit Ianmas @ ianmas-aws Ireland Help

Select Template

Specify Parameters Options Review

Select Template

Specify a stack name and then select the template that describes the stack that you want to create.

Stack

An AWS CloudFormation stack is a collection of related resources that you provision and update as a single unit.

Name

Template

A template is a JSON-formatted text file that describes your stack's resources and their properties. AWS CloudFormation stores the stack's template in an Amazon S3 bucket. [Learn more](#).

Source Select a sample template

Upload a template to Amazon S3 No file selected.

Specify an Amazon S3 template URL

© 2006 – 2014, Amazon Web Services, Inc. or its affiliates. All rights reserved. [Privacy Policy](#) [Terms of Use](#) [Feedback](#)

Using a template to create and manage a stack

Create A New Stack

Services VPC EC2 S3 RDS CloudFront Edit Ianmas @ ianmas-aws Ireland Help

Select Template

Specify Parameters Options Review

Select Template

Specify a stack name and then select the template that describes the stack that you want to create.

Stack

An AWS CloudFormation stack is a collection of related resources that you provision and update as a single unit.

Name

Template

A template is a JSON-formatted text file that describes your stack's resources and their properties. AWS CloudFormation stores the stack's template in an Amazon S3 bucket. [Learn more](#).

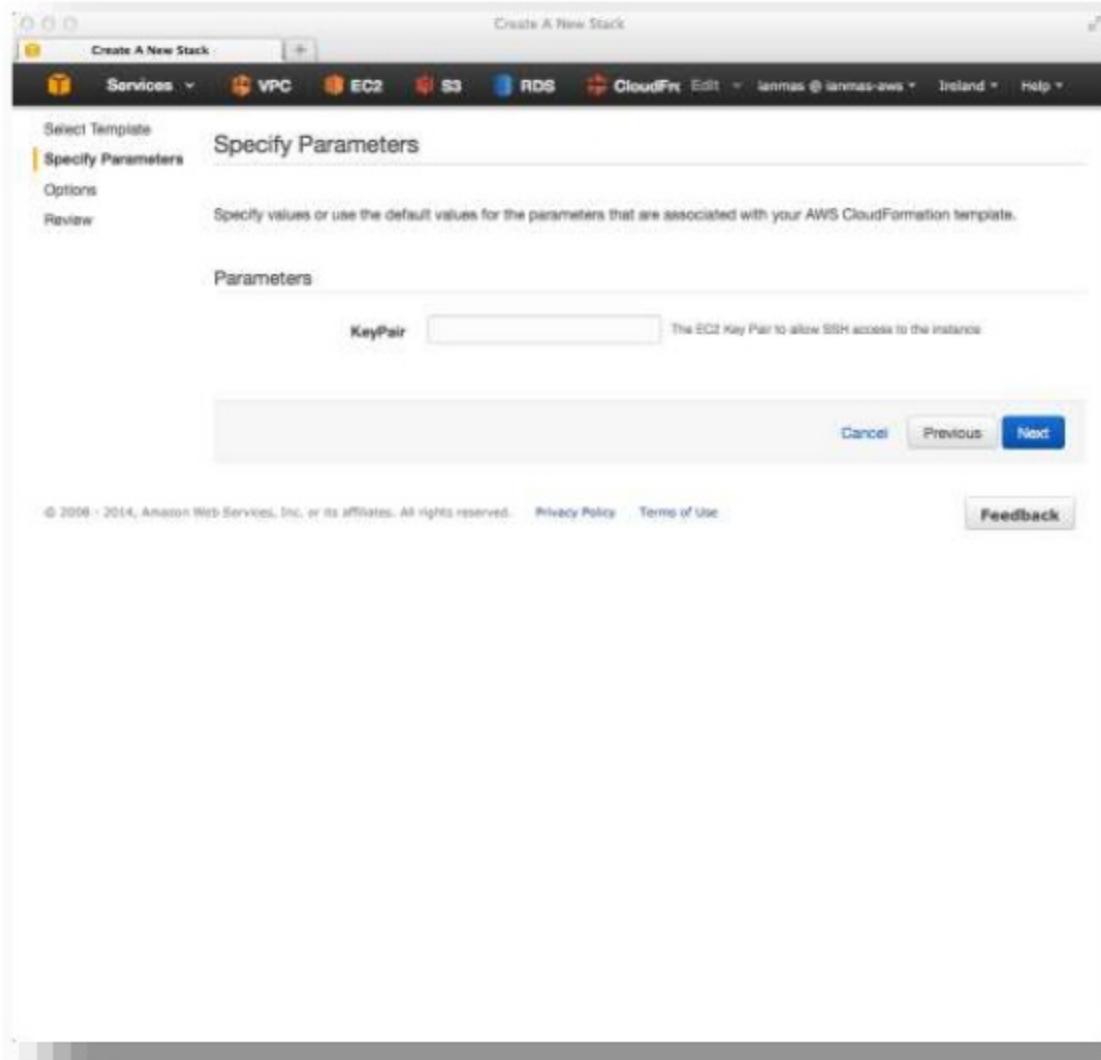
Source Select a sample template

Upload a template to Amazon S3

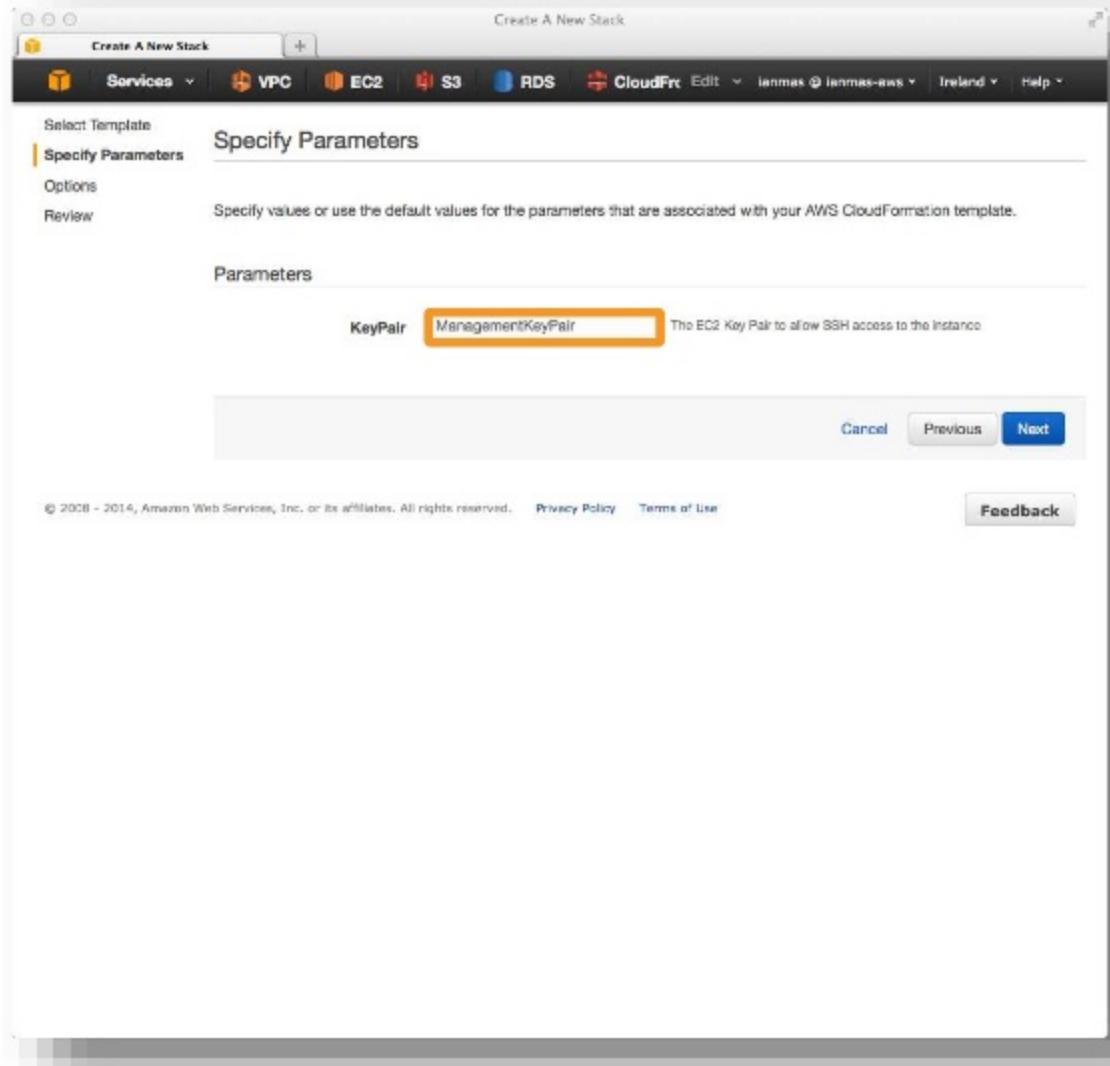
Specify an Amazon S3 template URL

© 2006 – 2014, Amazon Web Services, Inc. or its affiliates. All rights reserved. [Privacy Policy](#) [Terms of Use](#) [Feedback](#)

Using a template to create and manage a stack



Using a template to create and manage a stack



Using a template to create and manage a stack

Create A New Stack

Services VPC EC2 S3 RDS CloudFront Edit Ianmas @ ianmas-aws Ireland Help

Select Template Options

Specify Parameters Options Review

Tags

You can specify tags (key-value pairs) for resources in your stack. You can add up to 10 unique key-value pairs for each stack. [Learn more.](#)

Key (127 characters maximum)	Value (255 characters maximum)
1 name	StackName

Advanced

You can set additional options for your stack, like notification options and a stack policy. [Learn more.](#)

Cancel Previous Next

© 2006 - 2014, Amazon Web Services, Inc. or its affiliates. All rights reserved. [Privacy Policy](#) [Terms of Use](#) [Feedback](#)

Using a template to create and manage a stack

Create A New Stack

Services VPC EC2 S3 RDS CloudFront Edit Ianmas @ ianmas-aws Ireland Help

Select Template Review

Options Review

Template

Name	ec2instanceDemoStack
Template URL	https://s3.amazonaws.com/cf-templates-1/theizyvndtx-eu-west-1/20141740-ec2instance.template
Description	Create an EC2 instance running the newest Amazon Linux AMI.
Estimate cost	Link is not available

Parameters

KeyPair	ManagementKeyPair
Create IAM resources	False

Options

Tags

name	StackName
------	-----------

Advanced

Notification	
Timeout	none
Rollback on failure	Yes

Cancel Previous Create

Feedback

© 2006 - 2014, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Using a template to create and manage a stack

CloudFormation Management Console

Services VPC EC2 S3 RDS CloudFront Edit Ianmas @ Ianmas-AWS Ireland Help

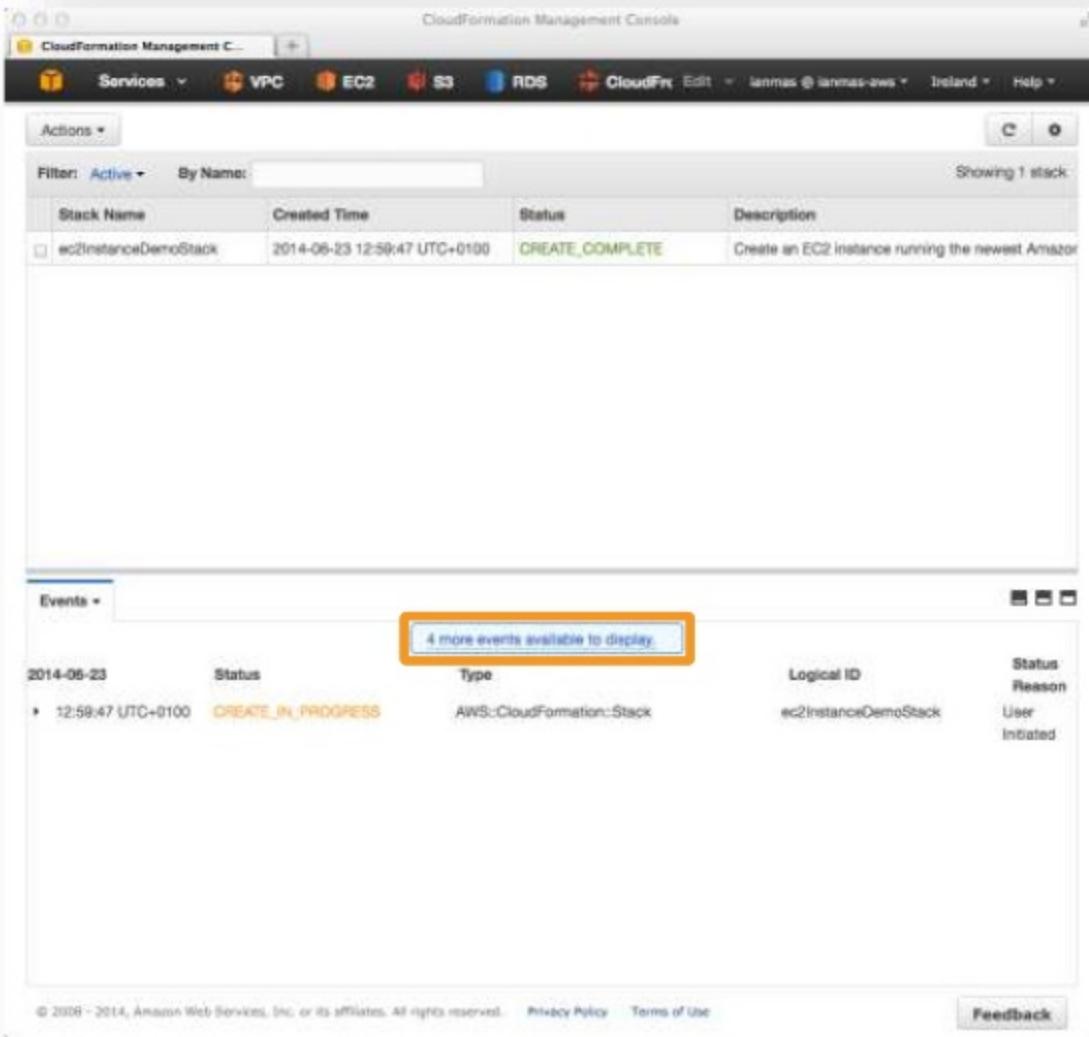
Actions Filter: Active By Name: Showing 1 stack

Stack Name	Created Time	Status	Description
ec2InstanceDemoStack	2014-06-23 12:59:47 UTC+0100	CREATE_COMPLETE	Create an EC2 instance running the newest Amazon Linux AMI.

Events 4 more events available to display.

2014-06-23	Status	Type	Logical ID	Status Reason
12:59:47 UTC+0100	CREATE_IN_PROGRESS	AWS::CloudFormation::Stack	ec2InstanceDemoStack	User Initiated

© 2006 - 2014, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Feedback



Using a template to create and manage a stack

CloudFormation Management Console

Services VPC EC2 S3 RDS CloudFront Edit Ianmas @ Ianmas-AWS Ireland Help

Actions Filter: Active By Name: Showing 1 stack

Stack Name	Created Time	Status	Description
ec2InstanceDemoStack	2014-06-23 12:59:47 UTC+0100	CREATE_COMPLETE	Create an EC2 instance running the newest Amazon Linux AMI.

Events

Date	Status	Type	Logical ID	Status Reason
2014-06-23	CREATE_COMPLETE	AWS::CloudFormation::Stack	ec2InstanceDemoStack	
	CREATE_COMPLETE	AWS::EC2::Instance	Ec2Instance	
	CREATE_IN_PROGRESS	AWS::EC2::Instance	Ec2Instance	Resource creation initiated
	CREATE_IN_PROGRESS	AWS::EC2::Instance	Ec2Instance	
	CREATE_IN_PROGRESS	AWS::CloudFormation::Stack	ec2InstanceDemoStack	User initiated

© 2006 - 2014, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Feedback

Using a template to create and manage a stack

CloudFormation Management Console

Services VPC EC2 S3 RDS CloudFront Edit Ianmas @ Ianmas-AWS Ireland Help

Actions Filter: Active By Name: Showing 1 stack

Stack Name	Created Time	Status	Description
ec2InstanceDemoStack	2014-06-23 12:59:47 UTC+0100	CREATE_COMPLETE	Create an EC2 instance running the newest Amazon Linux AMI.

Overview Outputs Resources Events Template Parameters Tags Stack Policy

Status	Type	Logical ID	Status Reason
CREATE_COMPLETE	AWS::CloudFormation::Stack	ec2InstanceDemoStack	
CREATE_COMPLETE	AWS::EC2::Instance	Ec2Instance	
CREATE_IN_PROGRESS	AWS::EC2::Instance	Ec2Instance	Resource creation initiated
CREATE_IN_PROGRESS	AWS::EC2::Instance	Ec2Instance	User initiated
CREATE_IN_PROGRESS	AWS::CloudFormation::Stack	ec2InstanceDemoStack	User initiated

© 2006 - 2014, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Feedback

Using a template to create and manage a stack

CloudFormation Management Console

Services VPC EC2 S3 RDS CloudFront Edit Ianmas @ Ianmas-AWS Ireland Help

Actions

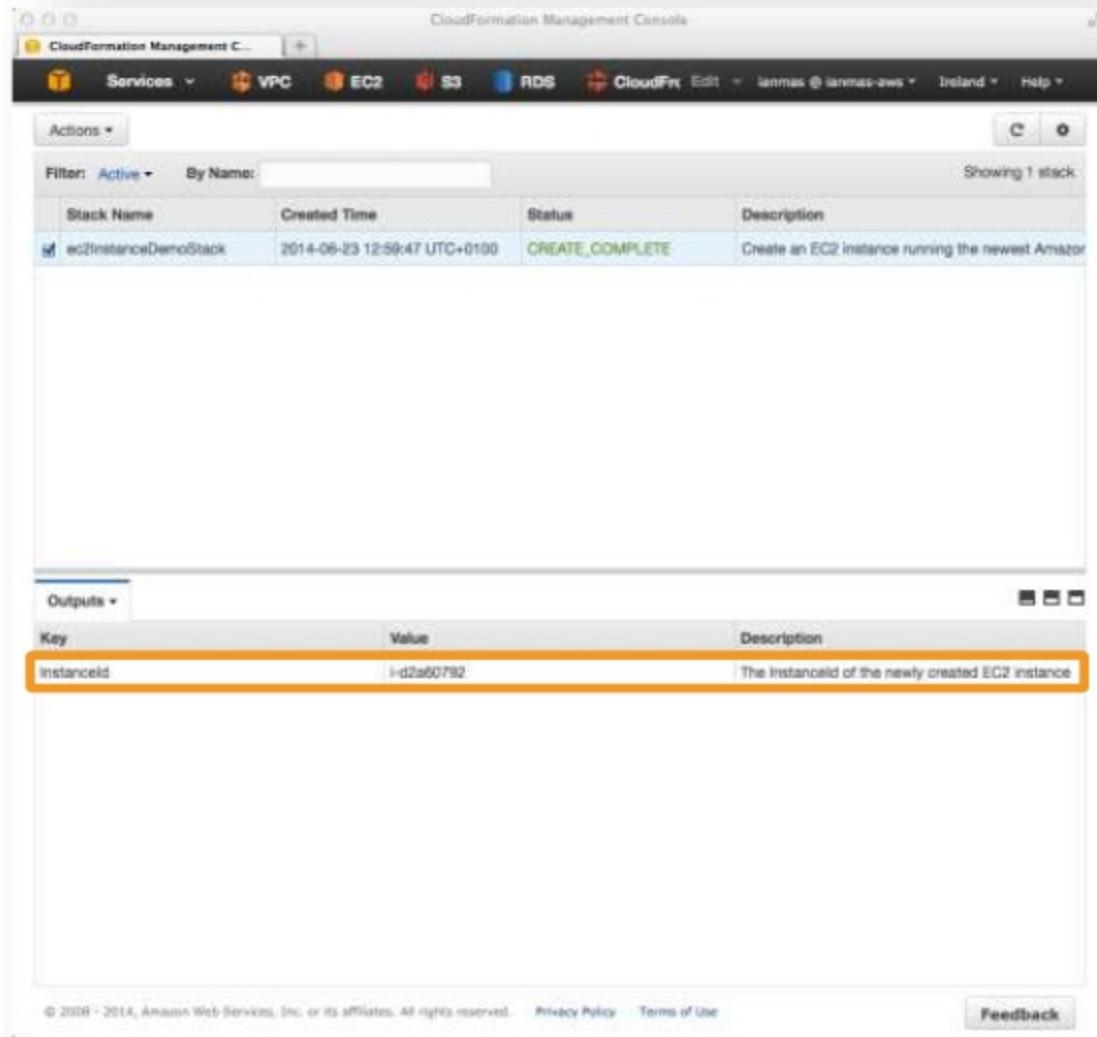
Filter: Active By Name: Showing 1 stack

Stack Name	Created Time	Status	Description
ec2InstanceDemoStack	2014-06-23 12:59:47 UTC+0100	CREATE_COMPLETE	Create an EC2 instance running the newest Amazon Linux AMI.

Outputs

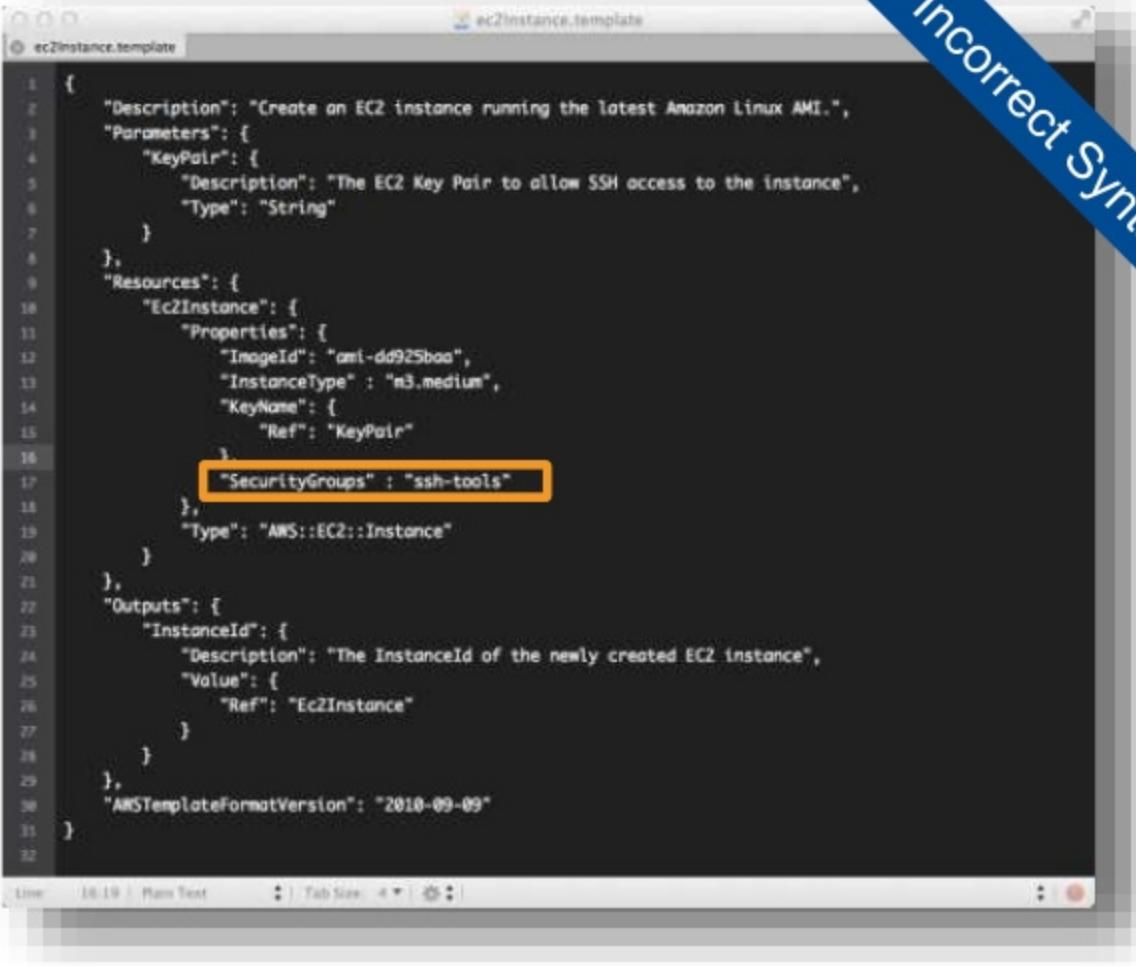
Key	Value	Description
InstanceId	i-d2a60792	The InstanceId of the newly created EC2 instance

© 2006 - 2014, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Feedback



Incorrect Syntax

Using a template to create and manage a stack



```
ec2Instance.template
1 {
  "Description": "Create an EC2 instance running the latest Amazon Linux AMI.",
  "Parameters": {
    "KeyPair": {
      "Description": "The EC2 Key Pair to allow SSH access to the instance",
      "Type": "String"
    }
  },
  "Resources": {
    "Ec2Instance": {
      "Properties": {
        "ImageId": "ami-dd925baa",
        "InstanceType": "m3.medium",
        "KeyName": {
          "Ref": "KeyPair"
        },
        "SecurityGroups" : "ssh-tools" // Incorrect syntax
      },
      "Type": "AWS::EC2::Instance"
    }
  },
  "Outputs": {
    "InstanceId": {
      "Description": "The InstanceId of the newly created EC2 instance",
      "Value": {
        "Ref": "Ec2Instance"
      }
    }
  },
  "AWSTemplateFormatVersion": "2010-09-09"
}
```

Using a template to create and manage a stack

CloudFormation Management Console

Actions ▾

Create Stack Name: Showing 1 stack

Update Stack

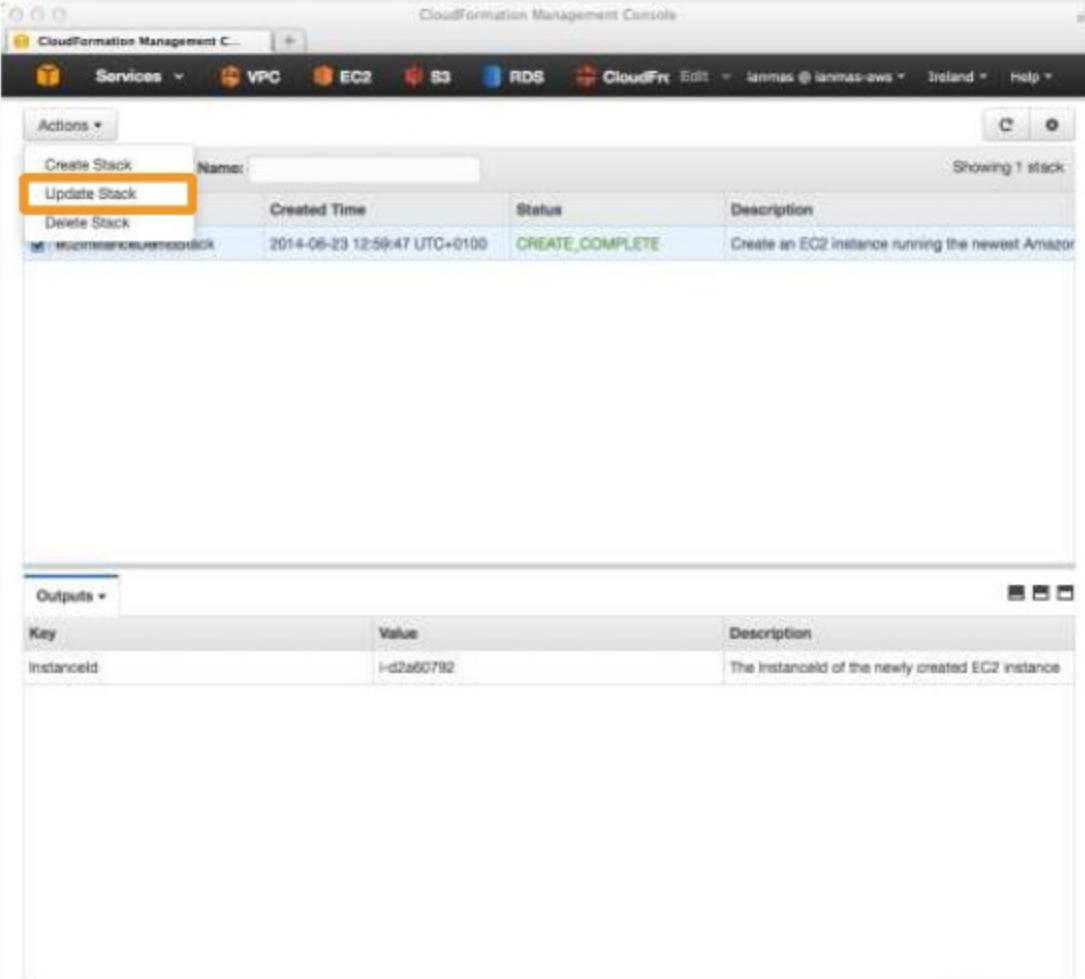
Delete Stack

	Created Time	Status	Description
M: myfirstcloudformationstack	2014-06-23 12:50:47 UTC+0100	CREATE_COMPLETE	Create an EC2 instance running the newest Amazon Linux AMI.

Outputs ▾

Key	Value	Description
InstanceId	i-d2a60792	The InstanceId of the newly created EC2 instance.

© 2006 - 2014, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Feedback



Using a template to create and manage a stack

The screenshot shows the AWS CloudFormation 'Update Stack' interface. The top navigation bar includes 'Services' (with a dropdown arrow), 'VPC', 'EC2', 'S3', 'RDS', 'CloudFront', 'Edit', and user information ('Jammes @ jammes-aws', 'Ireland', 'Help'). The main left sidebar has tabs: 'Select Template' (highlighted in yellow), 'Specify Parameters', 'Options', and 'Review'. The 'Select Template' tab displays the following content:

Select Template

To update an existing stack, provide a template that specifies the changes for the resources and properties that you want to update. AWS CloudFormation updates only the resources that have changed. [Learn more.](#)

Stack

An AWS CloudFormation stack is a collection of related resources that you provision and update as a single unit.

Name

Template

A template is a JSON-formatted text file that describes your stack's resources and their properties. AWS CloudFormation stores the stack's template in an Amazon S3 bucket. [Learn more.](#)

Source

Select a sample template
 Upload a template to Amazon S3
Browse...
 Specify an Amazon S3 template URL
 Use existing template

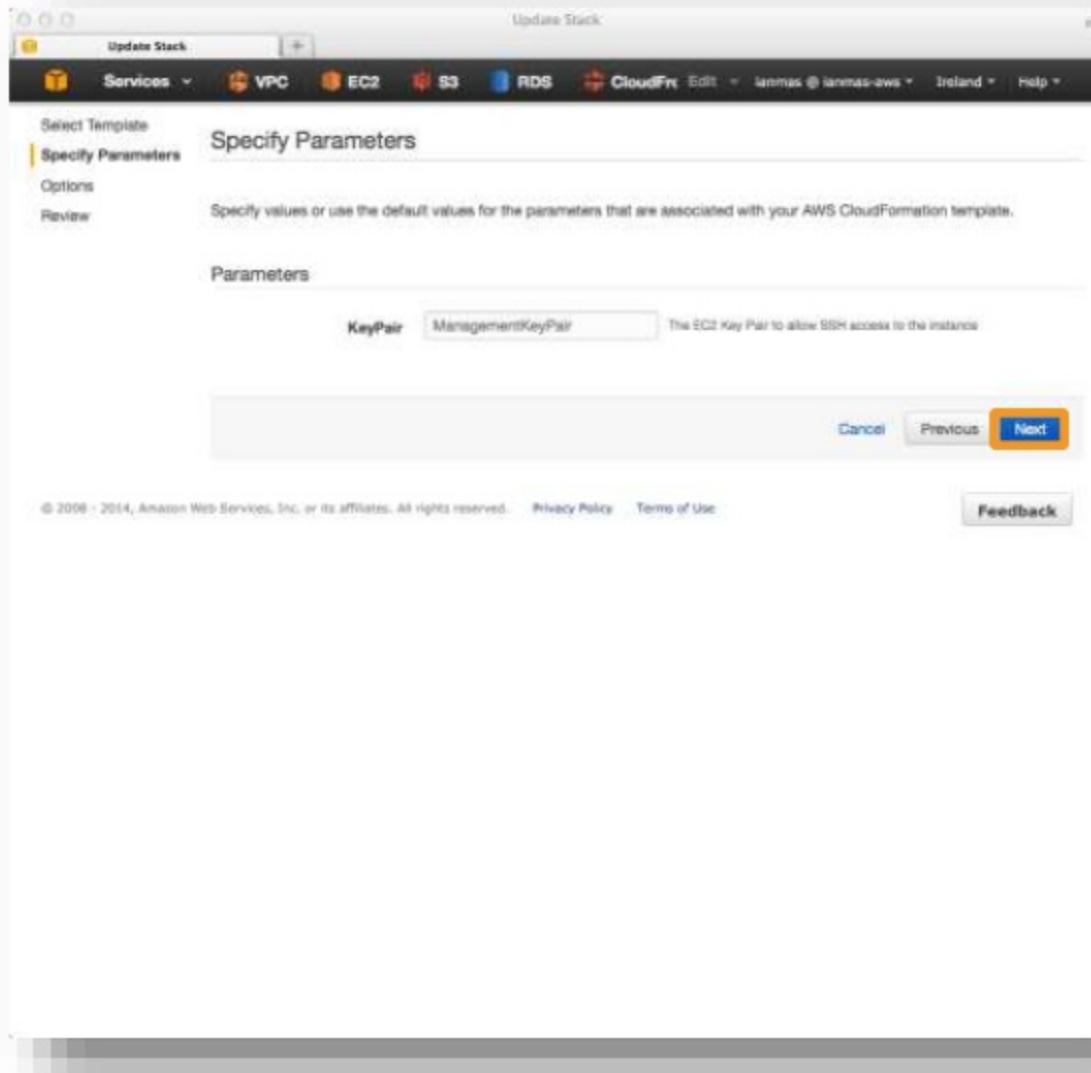
Next Step

Cancel **Next Step**

© 2008 - 2014, Amazon Web Services, Inc. or its affiliates. All rights reserved. [Privacy Policy](#) [Terms of Use](#)

Feedback

Using a template to create and manage a stack



Using a template to create and manage a stack

Provides Feedback

CloudFormation Management Console

Services VPC EC2 S3 RDS CloudFront Edit Jammes @ jammies-aws 16

Actions

Filter: Active By Name: Showing 1 result

Stack Name	Created Time	Status	Description
ec2InstanceDemoStack	2014-06-23 12:58:48 UTC+0100	UPDATE_ROLLBACK_COMPLETE	Create an EC2 instance running the newest Amazon Linux AMI.

Events

Time	Type	Resource	Stack	Message
13:37:55 UTC+0100	UPDATE_ROLLBACK_COMPLETE	AWS::CloudFormation::Stack	ec2InstanceDemoStack	
13:37:54 UTC+0100	UPDATE_ROLLBACK_COMPLETE	AWS::CloudFormation::Stack	ec2InstanceDemoStack	TE_CLEANUP_IN_PROGRESS
13:37:52 UTC+0100	UPDATE_COMPLETE	AWS::EC2::Instance	Ec2Instance	
13:37:27 UTC+0100	UPDATE_ROLLBACK_IN_PROGRESS	AWS::CloudFormation::Stack	ec2InstanceDemoStack	The following resource(s) failed to update: (Ec2Instance). RESS
13:37:26 UTC+0100	UPDATE_FAILED	AWS::EC2::Instance	Ec2Instance	Value of property SecurityGroups must be of type List of String

© 2006 - 2014, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Feedback

Using a template to create and manage a stack

Provides Feedback

CloudFormation Management Console

Services VPC EC2 S3 RDS CloudFront Edit Jammes @ jammies-aws 16

Actions Filter: Active By Name: Showing 1 result

Stack Name	Created Time	Status	Description
ec2InstanceDemoStack	2014-06-23 12:58:48 UTC+0100	UPDATE_ROLLBACK_COMPLETE	Create an EC2 instance running the newest Amazon Linux AMI. The following resource(s) failed to update: (Ec2Instance).

Events

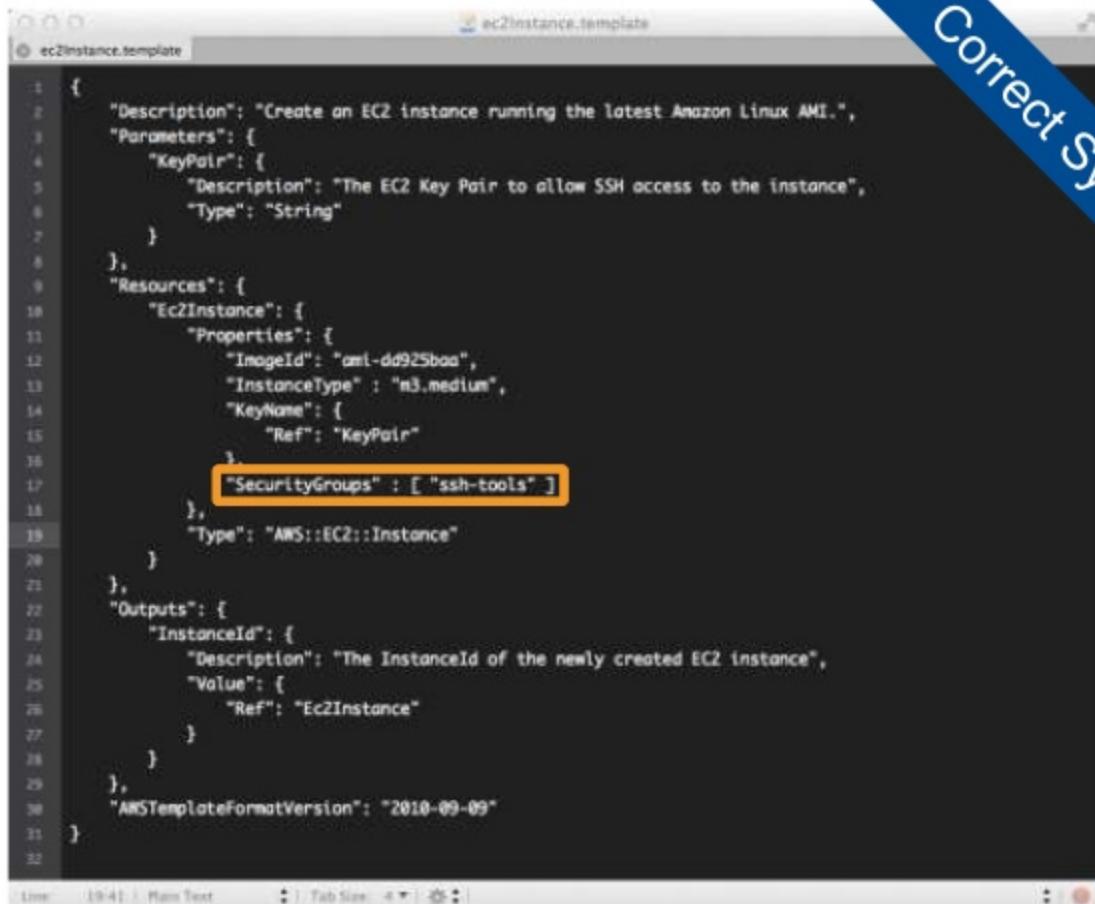
Time	Type	Resource	Stack
13:37:55 UTC+0100	UPDATE_ROLLBACK_COMPLETE	AWS::CloudFormation::Stack	ec2InstanceDemoStack
13:37:54 UTC+0100	UPDATE_ROLLBACK_COMPLETE	AWS::CloudFormation::Stack	ec2InstanceDemoStack
13:37:52 UTC+0100	UPDATE_COMPLETE	AWS::EC2::Instance	Ec2Instance
13:37:27 UTC+0100	UPDATE_ROLLBACK_IN_PROGRESS	AWS::CloudFormation::Stack	ec2InstanceDemoStack
13:37:26 UTC+0100	UPDATE_FAILED	AWS::EC2::Instance	Ec2Instance

The following resource(s) failed to update: (Ec2Instance).
Value of property SecurityGroups must be of type List of String

Feedback

Correct Syntax

Using a template to create and manage a stack



```
ec2Instance.template
1 {
2     "Description": "Create an EC2 instance running the latest Amazon Linux AMI.",
3     "Parameters": {
4         "KeyPair": {
5             "Description": "The EC2 Key Pair to allow SSH access to the instance",
6             "Type": "String"
7         }
8     },
9     "Resources": {
10        "Ec2Instance": {
11            "Properties": {
12                "ImageId": "ami-dd925baa",
13                "InstanceType": "m3.medium",
14                "KeyName": {
15                    "Ref": "KeyPair"
16                },
17                "SecurityGroups": [ "ssh-tools" ]
18            },
19            "Type": "AWS::EC2::Instance"
20        }
21    },
22    "Outputs": {
23        "InstanceId": {
24            "Description": "The InstanceId of the newly created EC2 instance",
25            "Value": {
26                "Ref": "Ec2Instance"
27            }
28        }
29    },
30    "AWSTemplateFormatVersion": "2010-09-09"
31 }
32 }
```

Using a template to create and manage a stack

CloudFormation Management Console

Services VPC EC2 S3 RDS CloudFront Edit Janmas @ janmas-aws Ireland Help

Actions

Filter: Active By Name: Showing 1 stack

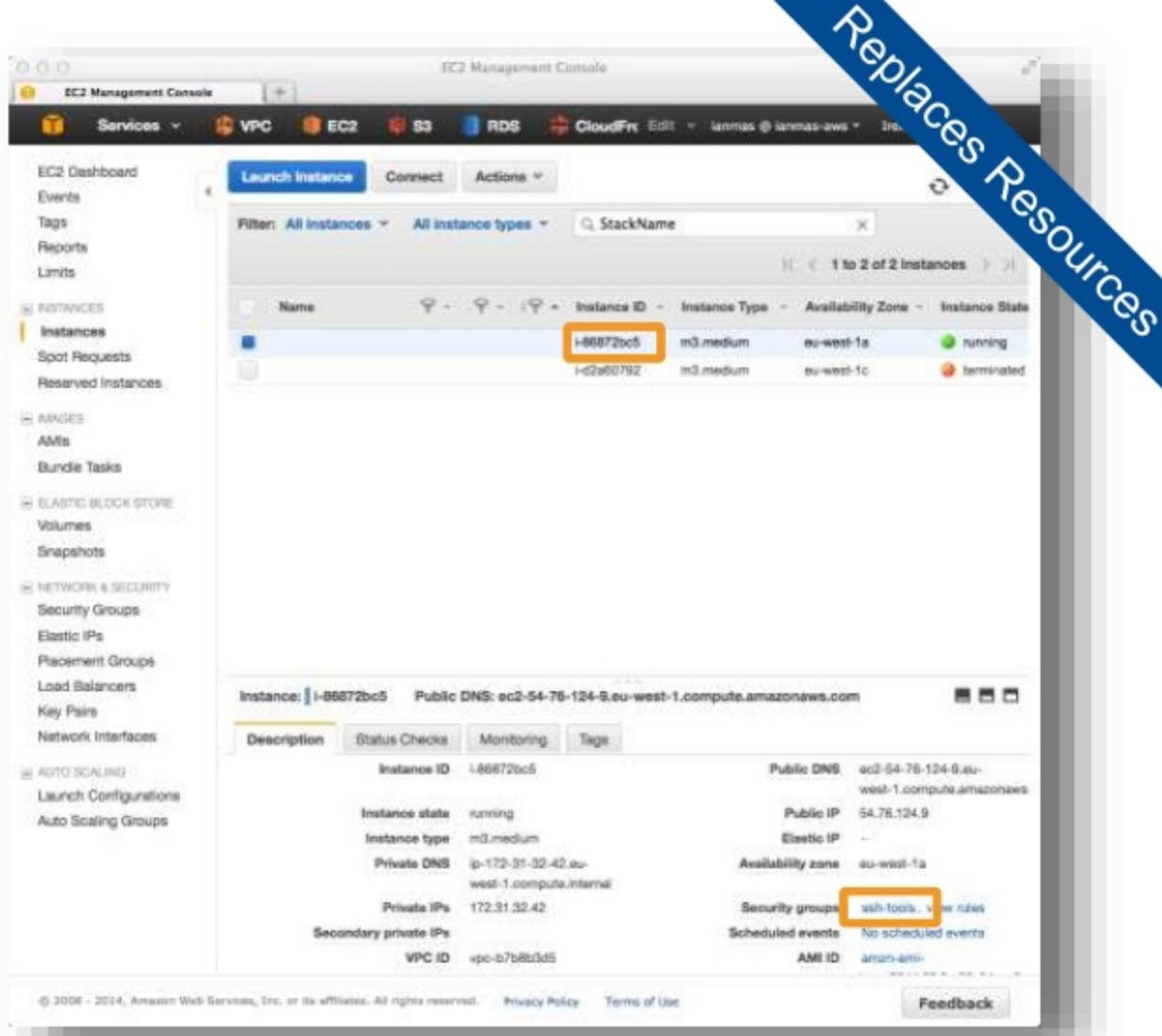
Stack Name	Created Time	Status	Description
ec2InstanceDemoStack	2014-06-23 12:50:48 UTC+0100	UPDATE_COMPLETE	Create an EC2 instance running the latest Amazon Linux AMI.

Events

Date	Status	Type	Logical ID	Status Reason
2014-06-23	13:45:53 UTC+0100	UPDATE_COMPLETE	AWS:CloudFormation:Stack	ec2InstanceDemoStack
	13:45:53 UTC+0100	DELETE_COMPLETE	AWS:EC2:Instance	Ec2Instance
	13:45:27 UTC+0100	DELETE_IN_PROGRESS	AWS:EC2:Instance	Ec2Instance
	13:45:25 UTC+0100	UPDATE_COMPLETE_CLEANUP_IN_PROGRESS	AWS:CloudFormation:Stack	ec2InstanceDemoStack
	13:45:23 UTC+0100	UPDATE_COMPLETE	AWS:EC2:Instance	Ec2Instance
	13:44:35 UTC+0100	UPDATE_IN_PROGRESS	AWS:EC2:Instance	Ec2Instance
	13:44:34 UTC+0100	UPDATE_IN_PROGRESS	AWS:EC2:Instance	Ec2Instance
				Resource creation initiated.
				Requested update requires the creation of a new physical resource; hence creating one.
	13:44:27 UTC+0100	UPDATE_IN_PROGRESS	AWS:CloudFormation:Stack	ec2InstanceDemoStack
	13:37:55 UTC+0100	UPDATE_ROLLBACK_COMPLETE	AWS:CloudFormation:Stack	ec2InstanceDemoStack
	13:37:54 UTC+0100	UPDATE_ROLLBACK_COMPLETE_CLEANUP_IN_PROGRESS	AWS:CloudFormation:Stack	ec2InstanceDemoStack
	13:37:52 UTC+0100	UPDATE_COMPLETE	AWS:EC2:Instance	Ec2Instance

© 2006 - 2014, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Feedback

Using a template to create and manage a stack



Using a template to create and manage a stack

CloudFormation Management Console

Actions: Create Stack, Update Stack, Delete Stack

Name: ec2instanceDemoStack

Showing 1 stack

	Created Time	Status	Description
ec2instanceDemoStack	2014-06-23 12:50:48 UTC+0100	UPDATE_COMPLETE	Create an EC2 instance running the latest Amazon Linux AMI.

Events:

Date	Status	Type	Logical ID	Status Reason
2014-06-23	UPDATE_COMPLETE	AWS::CloudFormation::Stack	ec2instanceDemoStack	
	DELETE_COMPLETE	AWS::EC2::Instance	Ec2Instance	
	DELETE_IN_PROGRESS	AWS::EC2::Instance	Ec2Instance	
	UPDATE_COMPLETE_CLEANUP_IN_PROGRESS	AWS::CloudFormation::Stack	ec2instanceDemoStack	
	UPDATE_COMPLETE	AWS::EC2::Instance	Ec2Instance	
	UPDATE_IN_PROGRESS	AWS::EC2::Instance	Ec2Instance	Resource creation initiated
	UPDATE_IN_PROGRESS	AWS::EC2::Instance	Ec2Instance	Requested update requires the creation of a new physical resource; hence creating one.
	UPDATE_IN_PROGRESS	AWS::CloudFormation::Stack	ec2instanceDemoStack	User Initiated
	UPDATE_ROLLBACK_COMPLETE	AWS::CloudFormation::Stack	ec2instanceDemoStack	
	UPDATE_ROLLBACK_COMPLETE_CLEANUP_IN_PROGRESS	AWS::CloudFormation::Stack	ec2instanceDemoStack	
	UPDATE_COMPLETE	AWS::EC2::Instance	Ec2Instance	

© 2006 - 2014, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Feedback

Using a template to create and manage a stack

CloudFormation Management Console

Actions ▾

Filter: Active ▾ By Name: Showing 0 stacks

Stack Name	Created Time	Status	Description
------------	--------------	--------	-------------

Events ▾

2014-06-23	Status	Type	Logical ID	Status Reason
13:53:48 UTC+0100	DELETE_COMPLETE	AWS::CloudFormation::Stack	ec2instanceDemoStack	
13:53:45 UTC+0100	DELETE_COMPLETE	AWS::EC2::Instance	Ec2Instance	
13:53:17 UTC+0100	DELETE_IN_PROGRESS	AWS::EC2::Instance	Ec2Instance	
13:52:55 UTC+0100	DELETE_IN_PROGRESS	AWS::CloudFormation::Stack	ec2instanceDemoStack	User Initiated
13:45:53 UTC+0100	UPDATE_COMPLETE	AWS::CloudFormation::Stack	ec2instanceDemoStack	
13:45:52 UTC+0100	DELETE_COMPLETE	AWS::EC2::Instance	Ec2Instance	
13:45:27 UTC+0100	DELETE_IN_PROGRESS	AWS::EC2::Instance	Ec2Instance	
13:45:25 UTC+0100	UPDATE_COMPLETE_CLEANUP_IN_PROGRESS	AWS::CloudFormation::Stack	ec2instanceDemoStack	
13:45:23 UTC+0100	UPDATE_COMPLETE	AWS::EC2::Instance	Ec2Instance	
13:44:35 UTC+0100	UPDATE_IN_PROGRESS	AWS::EC2::Instance	Ec2Instance	Resource creation initiated
13:44:34 UTC+0100	UPDATE_IN_PROGRESS	AWS::EC2::Instance	Ec2Instance	Requested update requires the creation of a new physical resource; hence creating one.
13:44:27 UTC+0100	UPDATE_IN_PROGRESS	AWS::CloudFormation::Stack	ec2instanceDemoStack	User Initiated
13:43:56 UTC+0100	CREATE_IN_PROGRESS	AWS::CloudFormation::Stack	ec2instanceDemoStack	

© 2006 - 2014, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Feedback

Using a template to create and manage a stack

Cleans Up Resources

The screenshot shows the EC2 Management Console interface. On the left, there's a navigation sidebar with links like EC2 Dashboard, Events, Tags, Reports, Limits, Instances, Images, Elastic Block Store, Network & Security, and Auto Scaling. The main area has tabs for Launch Instance, Connect, and Actions. A search bar at the top right says "StackName". Below it, a table lists two instances: one named "i-86872bc5" which is "terminated" and another named "i-02a60792" which is also "terminated". At the bottom, a detailed view for instance "i-86872bc5" shows its configuration: Instance ID: i-86872bc5, Instance state: terminated, Instance type: m3.medium, Public DNS: -, Private DNS: -, Private IP: -, Secondary private IPs: -, VPC ID: -, Public DNS: -, Public IP: -, Elastic IP: -, Availability zone: eu-west-1a, Security groups: -, Scheduled events: -, AMI ID: amzn-ami-hvm-2014.09.2.x86_64-gp2 (ami-dc925baa).

EC2 Management Console

Services VPC EC2 S3 RDS CloudFront Edit Jammas @ jammas-aws

EC2 Dashboard

Events

Tags

Reports

Limits

Instances

Instances

Spot Requests

Reserved Instances

Images

AMIs

Bundle Tasks

Elastic Block Store

Volumes

Snapshots

Network & Security

Security Groups

Elastic IPs

Placement Groups

Load Balancers

Key Pairs

Network Interfaces

Auto Scaling

Launch Configurations

Auto Scaling Groups

Launch Instance Connect Actions

Filter: All instances All instance types StackName

Name Instance ID Instance Type Availability Zone Instance State

i-86872bc5	i-86872bc5	m3.medium	eu-west-1a	terminated
i-02a60792	i-02a60792	m3.medium	eu-west-1c	terminated

Instance: i-86872bc5 Public DNS: -

Description	Status Checks	Monitoring	Tags
Instance ID: i-86872bc5			Public DNS: -
Instance state: terminated			Public IP: -
Instance type: m3.medium			Elastic IP: -
Private DNS: -			Availability zone: eu-west-1a
Private IP: -			Security groups: -
Secondary private IPs: -			Scheduled events: -
VPC ID: -			AMI ID: amzn-ami-hvm-2014.09.2.x86_64-gp2 (ami-dc925baa)

© 2006 - 2014, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Feedback

Using a template to create and manage a stack via the AWS CLI

```
aws cloudformation create-stack  
  --stack-name ec2InstanceCmdLineDemo  
  --template-url https://s3-eu-west-1.amazonaws.com/cf-templates-  
1fhelryvrdrbr-eu-west-1/2014174d0r-ec2Instance.template  
  --parameters ParameterKey=KeyPair,ParameterValue=ManagementKeyPair
```

Returns the details of the created stack, in the output format of your choice

```
arn:aws:cloudformation:eu-west-  
1:554625704737:stack/ec2InstanceCmdLineDemo/42cc6150-fad7-11e3-8f4d-  
5017e1aef4e7
```

Using a template to create and manage a stack via the AWS CLI

Screenshot of the CloudFormation Management Console showing a stack named "ec2instanceCmdLineDemo" in the "CREATE_COMPLETE" state.

Stack Name	Created Time	Status	Description
ec2instanceCmdLineDemo	2014-06-23 14:06:59 UTC+0100	CREATE_COMPLETE	Create an EC2 instance running the latest Amazon Linux AMI.

The "Events" tab shows the following log entries:

Date	Status	Type	Logical ID	Status Reason
2014-06-23	CREATE_COMPLETE	AWS::CloudFormation::Stack	ec2instanceCmdLineDemo	
+ 14:07:54 UTC+0100	CREATE_COMPLETE	AWS::EC2::Instance	Ec2Instance	
+ 14:07:52 UTC+0100	CREATE_IN_PROGRESS	AWS::EC2::Instance	Ec2Instance	Resource creation initiated
+ 14:07:03 UTC+0100	CREATE_IN_PROGRESS	AWS::EC2::Instance	Ec2Instance	
+ 14:06:59 UTC+0100	CREATE_IN_PROGRESS	AWS::CloudFormation::Stack	ec2instanceCmdLineDemo	User initiated

Other AWS CLI actions for CloudFormation

cancel-update-stack

get-stack-policy

create-stack

get-template

delete-stack

list-stack-resources

describe-stack-events

list-stacks

describe-stack-resource

set-stack-policy

describe-stack-resources

update-stack

describe-stacks

validate-template

As usual, you can get more details via the AWS CLI

```
$ aws cloudformation update-stack help
```

Help via the AWS CLI

```
$ aws cloudformation update-stack help
```

SYNOPSIS

```
update-stack
  --stack-name <value>
  [--template-body <value>]
  [--template-url <value>]
  [--use-previous-template | --no-use-previous-template]
  [--stack-policy-during-update-body <value>]
  [--stack-policy-during-update-url <value>]
  [--parameters <value>]
  [--capabilities <value>]
  [--stack-policy-body <value>]
  [--stack-policy-url <value>]
  [--notification-arns <value>]
```

```
$ aws cloudformation update-stack help
```

SYNOPSIS

```
update-stack
--stack-name <value>
[--template-body <value>]
[--template-url <value>]
[--use-previous-template | --no-use-previous-template]
[--stack-policy-during-update-body <value>]
[--stack-policy-during-update-url <value>]
[--parameters <value>]
[--capabilities <value>]
[--stack-policy-body <value>]
[--stack-policy-url <value>]
[--notification-arns <value>]
```

Working with AWS Resources

Working with AWS Resources

Designed to use your existing experience with AWS.

Each resource has a set of parameters with names that are identical to the names used to create the resources through their native API.

Working with AWS Resources

```
"myVolume" : {  
    "Type" : "AWS::EC2::Volume",  
    "Properties" : {  
        "Size" : "10",  
        "SnapshotId" : "snap-7b8fd361",  
        "AvailabilityZone" : "eu-west-1a"  
    }  
}
```

This example defines an Amazon EBS Volume with a logical name 'myVolume'. Its type is "AWS::EC2::Volume"

If you've used EBS previously, the properties should look very familiar

Working with AWS Resources

```
"InstanceSecurityGroup" : {  
    "Type" : "AWS::EC2::SecurityGroup",  
    "Properties" : {  
        "GroupDescription" : "Enable SSH access via port 22",  
        "SecurityGroupIngress" : [ {  
            "IpProtocol" : "tcp",  
            "FromPort" : "22",  
            "ToPort" : "22",  
            "CidrIp" : "0.0.0.0/0"  
        } ]  
    }  
}
```

Creating a Security Group resource

Working with AWS Resources

Supported AWS Services:

- Auto Scaling
- Amazon CloudFront
- AWS CloudWatch
- Amazon DynamoDB
- Amazon EC2
- Amazon ElastiCache
- AWS Elastic Beanstalk
- AWS Elastic Load Balancing
- AWS Identity and Access Management
- Amazon RDS
- Amazon Redshift
- Amazon Route 53
- Amazon S3
- Amazon SimpleDB
- Amazon SNS
- Amazon SQS
- Amazon VPC

Referencing Properties of Another Resource

```
{ "Resources" : {  
    "Ec2Instance" : {  
        "Type" : "AWS::EC2::Instance",  
        "Properties" : {  
            "SecurityGroups" : [ { "Ref" : "InstanceSecurityGroup" } ],  
            "KeyName" : "mykey",  
            "ImageId" : "ami-7a11e213"  
        }  
    },  
    "InstanceSecurityGroup" : {  
        "Type" : "AWS::EC2::SecurityGroup",  
        "Properties" : {  
            "GroupDescription" : "Enable SSH access via port 22",  
        "SecurityGroupIngress" : [ {  
                "IpProtocol" : "tcp",  
                "FromPort" : "22",  
                "ToPort" : "22",  
                "CidrIp" :"0.0.0.0/0" } ]  
        }  
    }  
}
```

```
{ "Resources" : {  
    "Ec2Instance" : {  
        "Type" : "AWS::EC2::Instance",  
        "Properties" : {  
            "SecurityGroups" : [ { "Ref" : "InstanceSecurityGroup" }, , "MyExistingSG"  
        ],  
        "KeyName" : "mykey",  
        "ImageId" : "ami-7a11e213" }  
    },  
    "InstanceSecurityGroup" : {  
        "Type" : "AWS::EC2::SecurityGroup",  
        "Properties" : {  
            "GroupDescription" : "Enable SSH access via port 22",  
        "SecurityGroupIngress" : [ {  
                "IpProtocol" : "tcp",  
                "FromPort" : "22",  
                "ToPort" : "22",  
                "CidrIp" :"0.0.0.0/0" } ]  
        }  
    }  
}
```

Referencing Input Parameters

Input Parameters

```
{  
  "Parameters" : {  
    "KeyPair" : {  
      "Description" : "The EC2 Key Pair to allow SSH access to the instance",  
      "Type" : "String"  
    },  
    "Resources" : {  
      "Ec2Instance" : {  
        "Type" : "AWS::EC2::Instance",  
        "Properties" : {  
          "SecurityGroups" : [ { "Ref" : "InstanceSecurityGroup" }],  
          "KeyName" : { "Ref" : "KeyPair"},  
          "ImageId" : ""  
        },  
        ...  
      } } }
```

Input
Parameters

Create A New Stack

Services VPC EC2 S3 RDS CloudFront Edit lanmas @ lanmas-aws Ireland Help

Select Template

Specify Parameters

Options

Review

Specify values or use the default values for the parameters that are associated with your AWS CloudFormation template.

Parameters

KeyPair ManagementKeyPair The EC2 Key Pair to allow SSH access to the instance

Cancel Previous Next

© 2008 - 2014, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Feedback

```
"WordPressUser": {  
    "Default": "admin",  
    "Description" : "The WordPress database admin account username",  
    "Type": "String",  
    "MinLength": "1",  
    "MaxLength": "16",  
    "AllowedPattern" : "[a-zA-Z][a-zA-Z0-9]*"  
},
```

Validate your input parameters with :

Maxlength, MinLength, MaxValue, MinValue,
AllowedPattern, AllowedValues

Conditional Values

```
{"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" ]  
            }  
        }  
    } } }
```

Other intrinsic functions and pseudo parameters

Intrinsic functions

Fn::Base64

Fn::FindInMap

Fn::GetAtt

Fn::GetAZs

Fn::Join

Fn::Select

Ref

Pseudo parameters

AWS::NotificationARNs

AWS::Region

AWS::StackId

AWS::StackName

Working with non-AWS Resources

Defining custom resources allows you to include non-AWS resources in a CloudFormation stack

More on Custom Resources in 'AWS CloudFormation under the Hood' from re:Invent 2013: <http://youtu.be/ZhGMaw67Yu0>
AWS CloudFormation Custom Resource Walkthrough documentation:

docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/crpg-walkthrough.html

Bootstrapping Applications and Handling Updates

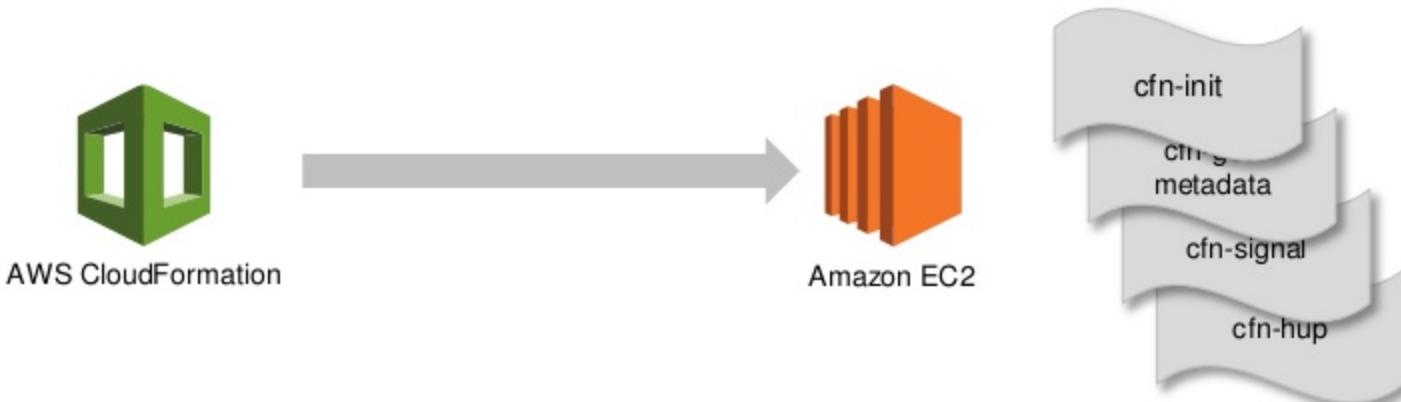
Bootstrapping Applications & Handling Updates

Option 1: Continue to use EC2 UserData, which is available as a property of AWS::EC2::Instance resources

```
"Resources" : {
    "Ec2Instance" : {
        "Type" : "AWS::EC2::Instance",
        "Properties" : {
            "KeyName" : { "Ref" : "KeyName" },
            "SecurityGroups" : [ { "Ref" : "InstanceSecurityGroup" } ],
            "ImageId" : { "Fn::FindInMap" : [ "RegionMap", { "Ref" : "AWS::Region" }, "AMI" ] },
            "UserData" : { "Fn::Base64" : { "Fn::Join" : [ "", [
                "#!/bin/bash -ex",
                "yum -y install gcc-c++ make",
                "yum -y install mysql-devel sqlite-devel",
                "yum -y install ruby-rdoc rubygems ruby-mysql ruby-devel",
                "gem install --no-ri --no-rdoc rails",
                "gem install --no-ri --no-rdoc mysql",
                "gem install --no-ri --no-rdoc sqlite3",
                "rails new myapp",
                "cd myapp",
                "rails server -d",
                "curl -X PUT -H 'Content-Type: application/json' --data-binary '{\"Status\": \"SUCCESS\", \"Reason\": \"The application myapp is ready\"}'"
            ] ] }
        }
    }
}
```

Bootstrapping Applications & Handling Updates

Option 2: AWS CloudFormation provides helper scripts for deployment within your EC2 instances



Metadata Key — `AWS::CloudFormation::Init`

Cfn-init reads this metadata key and installs the packages listed in this key (e.g., httpd, mysql, and php). Cfn-init also retrieves and expands files listed as sources.

Bootstrapping Applications & Handling Updates

```
"Resources" : {  
    "WebServer": {  
        "Type": "AWS::EC2::Instance",  
        "Metadata" : {  
            "Comment1" : "Configure the bootstrap helpers to install the Apache Web Server and PHP",  
            "Comment2" : "The website content is downloaded from the CloudFormationPHPSample.zip file",  
        },  
        "AWS::CloudFormation::Init" : {  
            "config" : {  
                "packages" : {  
                    "yum" : {  
                        "mysql"      : [],  
                        "mysql-server" : [],  
                        "mysql-libs"   : [],  
                        "httpd"       : [],  
                        "php"         : [],  
                        "php-mysql"   : []  
                    }  
                },  
                "sources" : {  
                    "/var/www/html" : "https://s3.amazonaws.com/cloudformation-examples/CloudFormationPHPSample.zip"  
                }  
            }  
        }  
    }  
}
```

Bootstrapping Applications & Handling Updates

The `UserData` key allows you to execute shell commands.

This template issues two shell commands: the first command installs the AWS CloudFormation helper scripts; the second executes the `cfn-init` script.

```
"Properties": {
    "ImageId" : { "Fn::FindInMap" : [ "AWSRegionArch2AMI", { "Ref" : "AWS::Region" },
                                      { "Fn::FindInMap" : [ "AWSInstanceType2Arch", { "Ref" : "InstanceType" }, "Arch" ] } ] },
    "InstanceType" : { "Ref" : "InstanceType" },
    "SecurityGroups" : [ { "Ref" : "WebServerSecurityGroup" } ],
    "KeyName" : { "Ref" : "KeyName" },
    "UserData" : { "Fn::Base64" : { "Fn::Join" : [ "", [
        "#!/bin/bash -v\n",
        "yum update -y aws-cfn-bootstrap\n",

        "# Install packages\n",
        "/opt/aws/bin/cfn-init -s ", { "Ref" : "AWS::StackName" }, " -r WebServer ",
        " --region ", { "Ref" : "AWS::Region" }, " || error_exit 'Failed to run cfn-init'\n"
    ]]} }
},
```

Bootstrapping Applications & Handling Updates

Creating files on
Instance Filesystems

The files key allows you to write files to the instance filesystem

```
"files" : {  
    "/tmp/setup.mysql" : {  
        "content" : { "Fn::Join" : [ "", [  
            "CREATE DATABASE ", { "Ref" : "DBName" }, ";\\n",  
            "GRANT ALL ON ", { "Ref" : "DBName" }, ".* TO '", { "Ref" : "DBUsername" }, "'@localhost  
IDENTIFIED BY '", { "Ref" : "DBPassword" }, "';\\n"  
        ]]},  
        "mode" : "000644",  
        "owner" : "root",  
        "group" : "root"  
    }  
}
```

Bootstrapping Applications & Handling Updates

Controlling Services

The services key allows you ensures that the services are not only running when cfn-init finishes (ensureRunning is set to true); but that they are also restarted upon reboot (enabled is set to true).

```
"services" : {  
    "sysvinit" : {  
        "mysqld" : {  
            "enabled"      : "true",  
            "ensureRunning" : "true"  
        },  
        "httpd" : {  
            "enabled"      : "true",  
            "ensureRunning" : "true"  
        }  
    }  
}
```

More on Deploying Applications with AWS CloudFormation:

docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/deploying.applications.html

Bootstrapping Applications & Handling Updates

Yes!

All that functionality is available for
your Windows instances too!

Bootstrapping AWS CloudFormation Windows Stacks:

docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/cfn-windows-stacks-bootstrapping.html

Bootstrapping Applications & Handling Updates

AWS CloudFormation Articles and Tutorials

Create Free Account

AWS CloudFormation gives developers and systems administrators an easy way to create a collection of related AWS resources and provision them in an orderly and predictable fashion. The following articles and documents provide guidance on building templates and using the various AWS CloudFormation features to provision your AWS resources.

What about Chef?

and/or

What about Puppet?

Bootstrapping Applications via AWS CloudFormation

AWS CloudFormation gives you an easy way to create the set of resources such as Amazon EC2 instance, Amazon RDS database instances and Elastic Load Balancers needed to run your application. The template describes what resources you need and AWS CloudFormation takes care of how provisioning the resources in an orderly and predictable fashion, handling and recovering from any failures or issues. While AWS CloudFormation takes care of provisioning all the resources, it raises the obvious question of how your application software is deployed, configured and executed on the Amazon EC2 instances. There are many options, each of which has implications on how quickly your application is ready and how flexible you need to be in terms of deploying new versions of the software.

[Read on... ▾](#)

Integrating AWS CloudFormation with Opscode Chef

AWS CloudFormation can help you to configure and/or install your application as well as how to bootstrap deployment and management tools that you may already use in your environment. Chef is an open source infrastructure automation solution from Opscode, written in Ruby, that allows you to automate the configuration of your systems and the applications that sit on top of it. AWS CloudFormation and Chef can be used together to automate your entire deployment and management processes, from your AWS resources through to your application artifacts.

[Read on... ▾](#)

Integrating AWS CloudFormation with Puppet

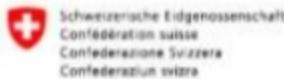
AWS CloudFormation can help you to configure and/or install your application as well as how to bootstrap deployment and management tools that you may already use in your environment. Puppet is an open source platform for provisioning, configuring and patching applications and operating system components. AWS CloudFormation and Puppet can be used together to automate your entire deployment and management processes, from your AWS resources through to your application artifacts.

[Read on... ▾](#)

Find out more here: aws.amazon.com/cloudformation/aws-cloudformation-articles-and-tutorials/

Some other things...

Who is using CloudFormation?



Find out more about these and other AWS case studies: aws.amazon.com/solutions/case-studies/



Zero to Sixty: AWS CloudFormation

Chetan Dandikar, Senior Product Manager – AWS CloudFormation

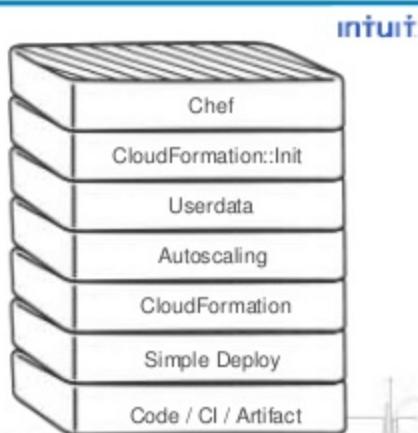
Caben Brinkley, Software Developer – Intuit

November 13, 2013



© 2013 Amazon.com, Inc. or its affiliates. All rights reserved. May not be copied, modified, or distributed in whole or in part without the express consent of Amazon.com, Inc.

Bootstrapping



Intuit's CloudFormation Story



```
"run_chef": {  
    "commands": {  
        "run_chef": {  
            "command": "/usr/bin/chef-solo \  
-c /var/chef/config/solo.rb \  
-o ", { "Ref", "Role" }  
        }  
    }  
}
```

Resources to learn more

Getting Started with AWS CloudFormation:

docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/GettingStarted.html

AWS CloudFormation Templates & Samples:

aws.amazon.com/cloudformation/aws-cloudformation-templates/

aws.amazon.com/cloudformation/aws-cloudformation-templates/aws-cloudformation-templates-eu-west-1/

AWS cfncluster HPC deployment framework:

github.com/awslabs/cfncluster/

Summary

Transparent and Open

Don't reinvent the wheel

Declarative & Flexible

CloudFormation

No Extra Charge

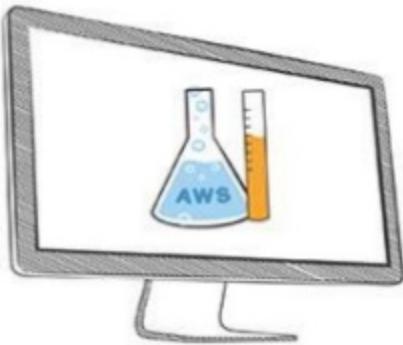
Customized via Parameters

Integration Ready

Find out more:
aws.amazon.com/cloudformation

AWS Training & Certification

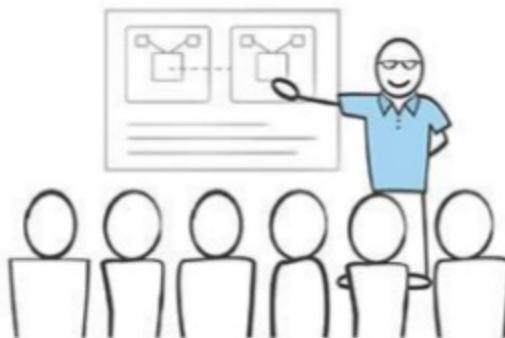
Self-Paced Labs



Try products, gain new skills, and get hands-on practice working with AWS technologies

[aws.amazon.com/training/
self-paced-labs](https://aws.amazon.com/training/self-paced-labs)

Training



Skill up and gain confidence to design, develop, deploy and manage your applications on AWS

aws.amazon.com/training

Certification



Demonstrate your skills, knowledge, and expertise with the AWS platform

aws.amazon.com/certification

Follow us for more
events & webinars



Ian Massingham – Technical Evangelist

 @IanMmmm



@AWS_UK for local AWS events & news



@AWScloud for Global AWS News and Announcements

We typically see customers start by trying our services

All Products

Compute & Networking

Storage

Database

Application Services

Development & Management

AWS Marketplace Software

FAQ »

Find answers to common questions
about the AWS Free Tier.



Amazon EC2 »

Web service that provides resizable compute capacity in the cloud.



Amazon S3 »

Highly-scalable, reliable, and low-latency data storage.



Amazon RDS »

Managed MySQL, Oracle and SQL Server databases.



Amazon CloudWatch »

Monitoring for AWS cloud resources and applications.



AWS Data Pipeline »

Orchestration for data-driven workflows.



Amazon DynamoDB »

Fully managed NoSQL database service with seamless scalability.



Amazon EBS »

Highly available, highly reliable, predictable storage volumes.



Amazon ELB »

Web service that provides scalability and high availability.



Amazon ElastiCache »

Managed scale-out caching.



Amazon SNS »

Web service to set up, operate, and send notifications from the cloud.



Amazon Elastic Transcoder »

Convert your media files easily, at low cost and at scale.



Amazon SWF »

Workflow service for building scalable, resilient applications.



AWS Marketplace »

Partner software pre-configured to run on AWS.

Get started now at : aws.amazon.com/getting-started



Design your application for the AWS Cloud

AWS Reference Architectures

The flexibility of AWS allows you to design your application architectures the way you like. AWS Reference Architecture Datasheets provide you with the architectural guidance you need in order to build an application that takes full advantage of the AWS cloud. Each datasheet includes a visual representation of the architecture and basic description of how each service is used.



Web Application Hosting
Build highly-scalable and reliable web or mobile-web applications ([PDF](#))



Content and Media Serving
Build highly reliable systems that serve massive amounts of content and media ([PDF](#))



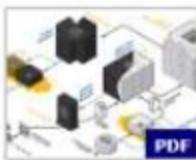
Batch Processing
Build auto-scalable batch processing systems like video processing pipelines ([PDF](#))



Fault tolerance and High Availability
Build systems that quickly failover to new instances in an event of failure ([PDF](#))



Large Scale Processing and Huge Data sets
Build high-performance computing systems that involve Big Data ([PDF](#))



Ad Serving
Build highly-scalable online ad serving solutions ([PDF](#))



Disaster Recovery for Local Applications
Build cost-effective Disaster Recovery solutions for on-premises applications ([PDF](#))



File Synchronization
Build simple file synchronization service ([PDF](#))

More details on the AWS Architecture Center at : aws.amazon.com/architecture

