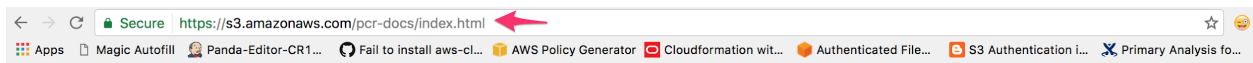


## Bootstrapping using CloudFormation :: Init



### CloudFormation Init Metadata :)

This is a public index.html page, that we will copy to our EC2 Instance through cloudformation



```

"Resources" : {
  "EC2Instance" : {
    "Type" : "AWS::EC2::Instance",
    "Metadata" : { ...
  },
  "Properties" : {
    "InstanceType" : { "Ref" : "InstanceType" },
    "SecurityGroups" : [ { "Ref" : "InstanceSecurityGroup" } ],
    "KeyName" : { "Ref" : "KeyName" },
    "ImageId" : { "Fn::FindInMap" : [ "AWSRegionArch2AMI", { "Ref" : "AWS::Region" },
    { "Fn::FindInMap" : [ "AWSInstanceType2Arch", { "Ref" : "InstanceType" }, "Arch" ] } ] },
    "UserData" : { "Fn::Base64" :
      { "Fn::Join" : [ "", [
        "#!/bin/bash -xe\n",
        "yum update -y aws-cfn-bootstrap\n",
        "# Install the files and packages from the metadata\n",
        "/opt/aws/bin/cfn-init -v ",
        "  --stack ", { "Ref" : "AWS::StackName" },
        "  --resource EC2Instance ",
        "  --configsets InstallAndRun ",
        "  --region ", { "Ref" : "AWS::Region" }, "\n",
        "# Signal the status from cfn-init\n",
        "/opt/aws/bin/cfn-signal -e $? ",
        "  --stack ", { "Ref" : "AWS::StackName" },
        "  --resource EC2Instance ",
        "  --region ", { "Ref" : "AWS::Region" }, "\n"
      ] ] }
    }
  }
}

```

Note the path of helper scripts `/opt/aws/bin`

aws

Services

Resource Groups

S3

RDS

EC2

DynamoDB

IAM

CloudFormation

AWS SysOps

N. Virginia

Support

CloudFormation

Stacks

Create Stack

Actions

Design template

Filter: Active

By Stack Name

Showing 2 stacks

Stack Name	Created Time	Status	Description
<input checked="" type="checkbox"/> EC2-cfn-init	2018-08-16 17:02:56 UTC+0550	CREATE_COMPLETE	AWS CloudFormation Sample Template EC2InstanceWithSecurityGroupSample: Create an Amazon EC2 ins...
<input type="checkbox"/> EC2-UserData	2018-08-16 14:30:59 UTC+0550	CREATE_COMPLETE	AWS CloudFormation Sample Template EC2InstanceWithSecurityGroupSample: Create an Amazon EC2 ins...

Overview

Outputs

Resources

Events

Template

Parameters

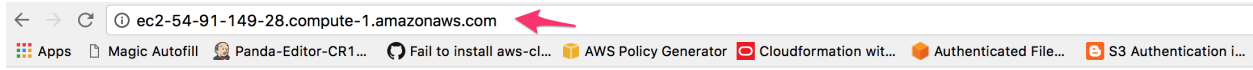
Tags

Stack Policy

Change Sets

Rollback Triggers

Key	Value	Description	Export Name
InstanceId	i-0d3fb00af289fbcce	InstanceId of the newly created EC2 instance	
PublicIP	54.91.149.28	Public IP address of the newly created EC2 instance	
AZ	us-east-1c	Availability Zone of the newly created EC2 instance	
PublicDNS	ec2-54-91-149-28.compute-1.amazonaws.com	Public DNSName of the newly created EC2 instance	



## CloudFormation Init Metadata :)

Even though our Instance was launched successfully and all the bootstrapping was done, we still have one issue in our CloudFormation stack and that is Stack reaches to CREATE\_COMPLETE state just after launching the Instance, it doesn't wait for bootstrapping of the Instance. And although we have added the cfn-signal to signal the last exit code (SUCCESS or FAILED - zero or non zero) to CloudFormation, the problem is CF stack reaches CREATE\_COMPLETE before we signal the SUCCESS or FAILED with cfn-signal.

### EC2 System Log for Error on signaling

```
--> Finished Dependency Resolution
Dependencies Resolved

=====
Package      Arch      Version      Repository    Size
=====
Updating:
aws-cfn-bootstrap  noarch    1.4-30.21.amzn1  amzn-updates  872 k

Transaction Summary
=====
Upgrade 1 Package

Total download size: 872 k
Downloading packages:
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Updating   : aws-cfn-bootstrap-1.4-30.21.amzn1.noarch                1/2
  Cleanup   : aws-cfn-bootstrap-1.4-27.19.amzn1.noarch                2/2
  Verifying  : aws-cfn-bootstrap-1.4-30.21.amzn1.noarch                1/2
  Verifying  : aws-cfn-bootstrap-1.4-27.19.amzn1.noarch                2/2

Updated:
  aws-cfn-bootstrap.noarch 0:1.4-30.21.amzn1

Complete!
+ /opt/aws/bin/cfn-init -v --stack EC2-cfn-init --resource EC2Instance --configsets InstallAndRun --region us-east-1
+ /opt/aws/bin/cfn-signal -e 0 --stack EC2-cfn-init --resource EC2Instance --region us-east-1
ValidationErrors: Stack arn:aws:cloudformation:us-east-1:472821263165:stack/EC2-cfn-init/1f00fc80-a148-11e8-9779-500c28b3e2d2 is in CREATE_COMPLETE state and cannot be signaled
Aug 16 11:34:04 cloud-init[2680]: util.py[WARNING]: Failed running /var/lib/cloud/instance/scripts/part-001 [1]
ci-info: +Authorized keys from /home/ec2-user/.ssh/authorized_keys for user ec2-user+
```

CF marks the stack CREATE\_COMPLETE just after launching the EC2 Instance and we are signaling the last exit code when it have already reached to CREATE\_COMPLETE, thats why we get the error when signaling with cfn-signal

To solve this issue, we can use CreationPolicy on EC2 Instance so that CloudFormation proceeds with stack creation only after EC2 Instance's bootstrapping is completed. A CreationPolicy instructs CloudFormation to wait on an instance until CloudFormation receives the specified number of signals.

```

"Resources" : {
  "EC2Instance" : {
    "Type" : "AWS::EC2::Instance",
    "Metadata" : { ...
  },
  "Properties" : {
    "InstanceType" : { "Ref" : "InstanceType" },
    "SecurityGroups" : [ { "Ref" : "InstanceSecurityGroup" } ],
    "KeyName" : { "Ref" : "KeyName" },
    "ImageId" : { "Fn::FindInMap" : [ "AWSRegionArch2AMI", { "Ref" : "AWS::Region" },
      { "Fn::FindInMap" : [ "AWSInstanceType2Arch", { "Ref" : "InstanceType" }, "Arch" ] ] } ],
    "UserData" : { "Fn::Base64" :
      { "Fn::Join" : [ "", [
        "#!/bin/bash -xe\n",
        "yum update -y aws-cfn-bootstrap\n",
        "# Install the files and packages from the metadata\n",
        "/opt/aws/bin/cfn-init -v ",
        "    --stack ", { "Ref" : "AWS::StackName" },
        "    --resource EC2Instance ",
        "    --configsets InstallAndRun ",
        "    --region ", { "Ref" : "AWS::Region" }, "\n",
        "# Signal the status from cfn-init\n",
        "/opt/aws/bin/cfn-signal -e $? ",
        "    --stack ", { "Ref" : "AWS::StackName" },
        "    --resource EC2Instance ",
        "    --region ", { "Ref" : "AWS::Region" }, "\n"
      ] ] }
    }
  },
  "CreationPolicy": {
    "ResourceSignal": {
      "Count": "1",
      "Timeout": "PT5M"
    }
  }
},

```

With CreationPolicy attached with the EC2 Instance CloudFormation waits on the EC2 instance until CloudFormation receives 1 signal within five minutes.

You can see the SUCCESS signal event under CF stack events

Create Stack

Actions

Design template

C

Filter: Active

By Stack Name

Showing 4 stacks

	Stack Name	Created Time	Status	Description
<input checked="" type="checkbox"/>	EC2-cfn-init-Creation-Policy	2018-08-17 16:04:59 UTC+0550	CREATE_COMPLETE	AWS CloudFormation Sample Template EC2InstanceWithSecurityGroupSample: Create an Amazon EC2 ins...
<input type="checkbox"/>	EC2-cfn-no-signal	2018-08-16 22:59:39 UTC+0550	CREATE_COMPLETE	AWS CloudFormation Sample Template EC2InstanceWithSecurityGroupSample: Create an Amazon EC2 ins...
<input type="checkbox"/>	EC2-cfn-init	2018-08-16 17:02:56 UTC+0550	CREATE_COMPLETE	AWS CloudFormation Sample Template EC2InstanceWithSecurityGroupSample: Create an Amazon EC2 ins...
<input type="checkbox"/>	EC2-UserData	2018-08-16 14:30:59 UTC+0550	CREATE_COMPLETE	AWS CloudFormation Sample Template EC2InstanceWithSecurityGroupSample: Create an Amazon EC2 ins...

Overview

Outputs

Resources

Events

Template

Parameters

Tags

Stack Policy

Change Sets

Rollback Triggers

Filter by: Status

Search events

2018-08-17	Status	Type	Logical ID	Status Reason
16:06:31 UTC+0550	CREATE_COMPLETE	AWS::CloudFormation::Stack	EC2-cfn-init-Creation-Policy	
16:06:28 UTC+0550	CREATE_COMPLETE	AWS::EC2::Instance	EC2Instance	
16:06:27 UTC+0550	CREATE_IN_PROGRESS	AWS::EC2::Instance	EC2Instance	Received SUCCESS signal with Uniqueid i-061ffc437a0535a7
16:05:13 UTC+0550	CREATE_IN_PROGRESS	AWS::EC2::Instance	EC2Instance	Resource creation Initiated
16:05:12 UTC+0550	CREATE_IN_PROGRESS	AWS::EC2::Instance	EC2Instance	
16:05:10 UTC+0550	CREATE_COMPLETE	AWS::EC2::SecurityGroup	InstanceSecurityGroup	
16:05:08 UTC+0550	CREATE_IN_PROGRESS	AWS::EC2::SecurityGroup	InstanceSecurityGroup	Resource creation Initiated
16:05:03 UTC+0550	CREATE_IN_PROGRESS	AWS::EC2::SecurityGroup	InstanceSecurityGroup	
16:04:59 UTC+0550	CREATE_IN_PROGRESS	AWS::CloudFormation::Stack	EC2-cfn-init-Creation-Policy	User Initiated

CloudFormation Stacks

Create Stack Actions Design template

Filter: Active By Stack Name Showing 4 stacks

Stack Name	Created Time	Status	Description
<input checked="" type="checkbox"/> EC2-cfn-init-Create-Policy	2018-08-17 16:04:59 UTC+0550	CREATE_COMPLETE	AWS CloudFormation Sample Template EC2InstanceWithSecurityGroupSample: Create an Amazon EC2 Ins...
<input type="checkbox"/> EC2-cfn-no-signal	2018-08-16 22:59:39 UTC+0550	CREATE_COMPLETE	AWS CloudFormation Sample Template EC2InstanceWithSecurityGroupSample: Create an Amazon EC2 Ins...
<input type="checkbox"/> EC2-cfn-init	2018-08-16 17:02:56 UTC+0550	CREATE_COMPLETE	AWS CloudFormation Sample Template EC2InstanceWithSecurityGroupSample: Create an Amazon EC2 Ins...
<input type="checkbox"/> EC2-UserData	2018-08-16 14:30:59 UTC+0550	CREATE_COMPLETE	AWS CloudFormation Sample Template EC2InstanceWithSecurityGroupSample: Create an Amazon EC2 Ins...

Overview Outputs Resources Events Template Parameters Tags Stack Policy Change Sets Rollback Triggers

Key	Value	Description	Export Name
InstanceId	i-061f0cc437a0535a7	InstanceId of the newly created EC2 instance	
PublicIP	18.206.120.33	Public IP address of the newly created EC2 instance	
AZ	us-east-1c	Availability Zone of the newly created EC2 instance	
PublicDNS	ec2-18-206-120-33.compute-1.amazonaws.com	Public DNSName of the newly created EC2 instance	

← → ↺ 18.206.120.33

Apps Magic Autofill Panda-Editor-CR1... Fail to install aws-cl... AWS Policy Generator Cloudformation wit... Authenticated File... S3 Authentication i...

## CloudFormation Init Metadata :)

```
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Updating      : aws-cfn-bootstrap-1.4-30.21.amzn1.noarch                1/2
  Cleanup      : aws-cfn-bootstrap-1.4-27.19.amzn1.noarch                2/2
  Verifying    : aws-cfn-bootstrap-1.4-30.21.amzn1.noarch                1/2
  Verifying    : aws-cfn-bootstrap-1.4-27.19.amzn1.noarch                2/2

Updated:
  aws-cfn-bootstrap.noarch 0:1.4-30.21.amzn1

Complete!
+ /opt/aws/bin/cfn-init -v --stack EC2-cfn-init-Create-Policy --resource EC2Instance --configsets InstallAndRun --region us-east-1
+ /opt/aws/bin/cfn-signal -e 0 --stack EC2-cfn-init-Create-Policy --resource EC2Instance --region us-east-1
ci-info: +Authorized keys from /home/ec2-user/.ssh/authorized_keys for user ec2-user+
ci-info: Keytype      Fingerprint (md5)      Options  Comment
ci-info: ssh-rsa      0f:b5:3f:10:88:a7:70:85:f3:f0:e9:53:ac:4e:47:1f      -      sysops
ec2: #####
ec2: -----BEGIN SSH HOST KEY FINGERPRINTS-----
ec2: 1024 SHA256:HQnW0Ufm4nSL561M/5M0EyLev/KMu+wLEsTtGcPkrk root@ip-172-31-90-241 (DSA)
ec2: 256  SHA256:su1zaoYcKXGN62VfzFuRjVTKTQ6CKWx7N+xxQpA3Y root@ip-172-31-90-241 (ECDSA)
ec2: 256  SHA256:D0PPtltHCq7ihLfUPL49Et1sYe0fWpWCIHWTXGhGneg no comment (ED25519)
ec2: 2048  SHA256:qgS3W2B8mYeDfW9sAcPjvMf/JqJnE4RzCwaZ5UHaZA root@ip-172-31-90-241 (RSA)
ec2: -----END SSH HOST KEY FINGERPRINTS-----
ec2: #####
ec2: -----BEGIN SSH HOST KEY KEYS-----
ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlldHAyNTYAAAAIbmlldHAyNTYAAABBBIBHpSo80z0F3xvItce3usTmknPCqazbfarH9zKYITiCT5sCA5wLWp6aA9WPrzAvCZSLM+32vejByNGWhjZ3
EPc= root@ip-172-31-90-241
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIFNxUo8L739IyiL70r/v+1rHp7XgnERKDNipqydX3Mj5
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQChPOxZughW3jobjjIE269ATA02u1nLNB6Ush+R08CJmdT09izxv/4qfUuWc6EDQNPfJg+K00u61mtwdMvy1F9pbn5e4Vij5mMrYfiez3RTnkMyvfdoH
U09uQYeyYIm6uwEJ1fRgEYyykMPQtdiKGP7C/oaLR/fgMeMyjQif2cbgdr8B+N2D15Vg5r5f5XYiHBWYTD/254jTTf9H5f3Bh6zC1W77jdjo4k2UbkIxxvpRnXnHoXzHBxoZvz4oa9bbSrgtV9ah190K19
gd8nLu1Fzh/cr0txXU5tNcE41Mg7M0Pc+Cr3bEwnb7qWBWE2CJHR1UQmpNzQ0ZqQ/u9 root@ip-172-31-90-241
-----END SSH HOST KEY KEYS-----
Cloud-init v. 0.7.6 finished at Fri, 17 Aug 2018 10:36:19 +0000. Datasource DataSourceEc2. Up 33.59 seconds

Amazon Linux AMI release 2017.09
Kernel 4.9.76-3.78.amzn1.x86_64 on an x86_64

ip-172-31-90-241 login:      2018-08-17T10:38:45.000Z
```

Note that after attaching CreationPolicy we don't get the error while singling and also CF waits for the signal before proceeding with the stack

<https://aws.amazon.com/blogs/devops/use-a-creationpolicy-to-wait-for-on-instance-configurations/>