

CloudFormation

- Stacks
- Exports
- Stacks
- Stack Detail

AWSMPBASICCloudControllerwithnewVPC

Other Actions

- Create Change Set
- Continue Update Rollback
- Delete Stack
- View/Edit template in Designer

Update Stack

Cancel Update Stack

Stack name:

AWSMPBASICCloudControllerwithnewVPC

Stack ID:

arn:aws:cloudformation:us-east-1:872510204642:stack/AWSMPBASICCloudControllerwithnewVPC/feab1be0-e270-11e6-9ae3-503aca2616fd

Status:

CREATE_FAILED

Status reason:

The following resource(s) failed to create: [InstanceWaitCondition].

IAM Role:

Description:

Outputs

Key	Value	Description	Export Name
No outputs found.			

Resources

Logical ID	Physical ID	Type	Status
0 more resources available to display.			
AttachGateway	AWSMP-Attac-1R9S6S3LYAKS0	AWS::EC2::VPCGatewayAttachment	CREATE_FAILED
Cloudbreak	i-00a0ed85e9b1c3e7d	AWS::EC2::Instance	CREATE_FAILED
CloudbreakRole	AWSMPBASICCloudControllerwithnewVPC-CloudbreakRole-2RXFVM4JGQ8B	AWS::IAM::Role	CREATE_FAILED
CloudbreakSecurityGroup	sg-22f45d5e	AWS::EC2::SecurityGroup	CREATE_FAILED

Logical ID	Physical ID	Type	Status
CredentialRole	AWSMPBASICCloudControllerwithnewVPC-CredentialRole-1VEE7XGB307P4	AWS::IAM::Role	CREATE_FAILED
InstanceWaitCondition	arn:aws:cloudformation:us-east-1:872510204642:stack/AWSMPBASICCloudControllerwithnewVPC/feab1be0-e270-11e6-9ae3-503aca2616fd/InstanceWaitHandle	AWS::CloudFormation::WaitCondition	CREATE_FAILED
InstanceWaitHandle	https://cloudformation-waitcondition-us-east-1.s3.amazonaws.com/arn%3Aaws%3Acloudformation%3Aus-east-1%3A872510204642%3Astack/AWSMPBASICCloudControllerwithnewVPC/feab1be0-e270-11e6-9ae3-503aca2616fd/InstanceWaitHandle?AWSAccessKeyId=AKIAIIT3CWAJMYUTISA&Expires=1485374969&Signature=%2BOSoq4TNvVJQBqaoJVimSO2ddVk%3D	AWS::CloudFormation::WaitConditionHandle	CREATE_FAILED
InternetGateway	igw-89ade3ee	AWS::EC2::InternetGateway	CREATE_FAILED
PublicRoute	AWSMP-Publi-1E60LFPKGRURI	AWS::EC2::Route	CREATE_FAILED
PublicRouteTable	rtb-02c10f7b	AWS::EC2::RouteTable	CREATE_FAILED
PublicSubnet	subnet-7899ae31	AWS::EC2::Subnet	CREATE_FAILED
PublicSubnetRouteTableAssociation	rtbassoc-17f05e6f	AWS::EC2::SubnetRouteTableAssociation	CREATE_FAILED
RootInstanceProfile	AWSMPBASICCloudControllerwithnewVPC-RootInstanceProfile-1G13V7SXOIBAG	AWS::IAM::InstanceProfile	CREATE_FAILED
VPC	vpc-2833bd4e	AWS::EC2::VPC	CREATE_FAILED

No Values Found.

[View More Resources](#)

Events

0 more events available to display

Time	Status	Type	Logical ID	Status reason
2017-01-24 14:21:12 UTC-0600	CREATE_FAILED	AWS::CloudFormation::Stack	AWSMPBASICCloudControllerwithnewVPC	The following resource(s) failed to create: [InstanceWaitCondition].
Physical ID: arn:aws:cloudformation:us-east-1:872510204642:stack/AWSMPBASICCloudControllerwithnewVPC/feab1be0-e270-11e6-9ae3-503aca2616fd				
2017-01-24 14:21:11 UTC-0600	CREATE_FAILED	AWS::CloudFormation::WaitCondition	InstanceWaitCondition	WaitCondition received failed message: 'ERROR: command 'hdc create-credential --credential-name aws-access --role-arn \$CREDENTIAL_ROLE_ARN --ssh-key-url http://169.254.169.254/latest/meta-data/public-keys/0/openssh-key --existing-ssh-key-pair \$KEYPAIR_NAME' exited with status: 1 line: 1' for uniqueId: cbd-init
Physical ID: arn:aws:cloudformation:us-east-1:872510204642:stack/AWSMPBASICCloudControllerwithnewVPC/feab1be0-e270-11e6-9ae3-503aca2616fd/InstanceWaitHandle				
2017-01-24 14:13:13 UTC-0600	CREATE_IN_PROGRESS	AWS::CloudFormation::WaitCondition	InstanceWaitCondition	Resource creation Initiated
Physical ID: arn:aws:cloudformation:us-east-1:872510204642:stack/AWSMPBASICCloudControllerwithnewVPC/feab1be0-e270-11e6-9ae3-503aca2616fd/InstanceWaitHandle				
2017-01-24 14:13:13 UTC-0600	CREATE_IN_PROGRESS	AWS::CloudFormation::WaitCondition	InstanceWaitCondition	
Physical ID:				
2017-01-24 14:13:08 UTC-0600	CREATE_COMPLETE	AWS::EC2::Instance	Cloudbreak	
Physical ID: i-00a0ed85c9b1c3e7d				
2017-01-24 14:12:21 UTC-	CREATE_IN_PROGRESS	AWS::EC2::Instance	Cloudbreak	Resource creation Initiated

0600
Physical ID:i-00a0ed85c9b1c3e7d
14:12:20
UTC- CREATE_IN_PROGRESS AWS::EC2::Instance Cloudbreak
0600
Physical ID:
14:12:16
UTC- CREATE_COMPLETE AWS::IAM::InstanceProfile RootInstanceProfile
0600
Physical ID:AWSMPBASICCloudControllerwithnewVPC-RootInstanceProfile-1G13V7SXOIBAG
14:10:30
UTC- CREATE_COMPLETE AWS::EC2::SubnetRouteTableAssociation PublicSubnetRouteTableAssociation
0600
Physical ID:rtbassoc-17f05e6f
14:10:27
UTC- CREATE_COMPLETE AWS::EC2::Route PublicRoute
0600
Physical ID:AWSMP-Publi-1E60LFPKGRURI
14:10:16
UTC- CREATE_IN_PROGRESS AWS::IAM::InstanceProfile RootInstanceProfile Resource creation Initiated
0600
Physical ID:AWSMPBASICCloudControllerwithnewVPC-RootInstanceProfile-1G13V7SXOIBAG
14:10:15
UTC- CREATE_IN_PROGRESS AWS::IAM::InstanceProfile RootInstanceProfile
0600
Physical ID:
14:10:15
UTC- CREATE_COMPLETE AWS::IAM::Role CredentialRole
0600
Physical ID:AWSMPBASICCloudControllerwithnewVPC-CredentialRole-1VEE7XGB307P4
14:10:14
UTC- CREATE_IN_PROGRESS AWS::EC2::SubnetRouteTableAssociation PublicSubnetRouteTableAssociation Resource creation Initiated
0600
Physical ID:rtbassoc-17f05e6f
14:10:14
UTC- CREATE_IN_PROGRESS AWS::EC2::SubnetRouteTableAssociation PublicSubnetRouteTableAssociation
0600
Physical ID:
14:10:13
UTC- CREATE_COMPLETE AWS::EC2::SecurityGroup CloudbreakSecurityGroup
0600
Physical ID:sg-22f45d5e
14:10:12
UTC- CREATE_COMPLETE AWS::IAM::Role CloudbreakRole
0600
Physical ID:AWSMPBASICCloudControllerwithnewVPC-CloudbreakRole-2RXFVM4JGQ8B
14:10:12
UTC- CREATE_IN_PROGRESS AWS::EC2::Route PublicRoute Resource creation Initiated
0600
Physical ID:AWSMP-Publi-1E60LFPKGRURI
14:10:11
UTC- CREATE_IN_PROGRESS AWS::EC2::Route PublicRoute
0600
Physical ID:
14:10:11

UTC- 0600 CREATE_COMPLETE AWS::EC2::Subnet PublicSubnet

Physical ID:subnet-7899ae31

14:10:10
UTC- 0600 CREATE_IN_PROGRESS AWS::EC2::SecurityGroup CloudbreakSecurityGroup Resource creation Initiated

Physical ID:sg-22f45d5e

14:10:08
UTC- 0600 CREATE_COMPLETE AWS::EC2::VPCGatewayAttachment AttachGateway

Physical ID:AWSMP-Attac-1R9S6S3LYAKS0

14:09:54
UTC- 0600 CREATE_IN_PROGRESS AWS::EC2::Subnet PublicSubnet Resource creation Initiated

Physical ID:subnet-7899ae31

14:09:54
UTC- 0600 CREATE_COMPLETE AWS::EC2::RouteTable PublicRouteTable

Physical ID:rtb-02c10f7b

14:09:54
UTC- 0600 CREATE_IN_PROGRESS AWS::EC2::SecurityGroup CloudbreakSecurityGroup

Physical ID:

14:09:53
UTC- 0600 CREATE_IN_PROGRESS AWS::EC2::Subnet PublicSubnet

Physical ID:

14:09:52
UTC- 0600 CREATE_IN_PROGRESS AWS::EC2::VPCGatewayAttachment AttachGateway Resource creation Initiated

Physical ID:AWSMP-Attac-1R9S6S3LYAKS0

14:09:52
UTC- 0600 CREATE_IN_PROGRESS AWS::EC2::RouteTable PublicRouteTable Resource creation Initiated

Physical ID:rtb-02c10f7b

14:09:52
UTC- 0600 CREATE_IN_PROGRESS AWS::EC2::VPCGatewayAttachment AttachGateway

Physical ID:

14:09:52
UTC- 0600 CREATE_IN_PROGRESS AWS::EC2::RouteTable PublicRouteTable

Physical ID:

14:09:49
UTC- 0600 CREATE_COMPLETE AWS::EC2::VPC VPC

Physical ID:vpc-2833bd4e

14:09:45
UTC- 0600 CREATE_COMPLETE AWS::EC2::InternetGateway InternetGateway

Physical ID:igw-89ade3ee

14:09:30
UTC- 0600 CREATE_IN_PROGRESS AWS::EC2::VPC VPC Resource creation Initiated

Physical ID:vpc-2833bd4e

14:09:30
UTC- CREATE_IN_PROGRESS AWS::IAM::Role CloudbreakRole Resource creation Initiated
0600
Physical ID:AWSMPBASICCloudControllerwithnewVPC-CloudbreakRole-2RXFVM4JGQ8B

14:09:30
UTC- CREATE_IN_PROGRESS AWS::EC2::VPC VPC
0600
Physical ID:

14:09:29
UTC- CREATE_IN_PROGRESS AWS::IAM::Role CredentialRole Resource creation Initiated
0600
Physical ID:AWSMPBASICCloudControllerwithnewVPC-CredentialRole-1VEE7XGB307P4

14:09:29
UTC- CREATE_IN_PROGRESS AWS::EC2::InternetGateway InternetGateway Resource creation Initiated
0600
Physical ID:igw-89ade3ee

14:09:29
UTC- CREATE_IN_PROGRESS AWS::IAM::Role CloudbreakRole
0600
Physical ID:

14:09:29
UTC- CREATE_COMPLETE AWS::CloudFormation::WaitConditionHandle InstanceWaitHandle
0600
Physical ID:https://cloudformation-waitcondition-us-east-1.s3.amazonaws.com/arn%3Aaws%3Acloudformation%3Aus-east-1%3A872510204642%3Astack/AWSMPBASICCloudControllerwithnewVPC/feab1be0-e270-11e6-9ae3-503aca2616fd/InstanceWaitHandle?AWSAccessKeyId=AKIAIIT3CWAIMJYUTISA&Expires=1485374969&Signature=%2BOSoq4TNvVJQBqaoJVimSO2ddVk%3D

14:09:29
UTC- CREATE_IN_PROGRESS AWS::CloudFormation::WaitConditionHandle InstanceWaitHandle Resource creation Initiated
0600
Physical ID:https://cloudformation-waitcondition-us-east-1.s3.amazonaws.com/arn%3Aaws%3Acloudformation%3Aus-east-1%3A872510204642%3Astack/AWSMPBASICCloudControllerwithnewVPC/feab1be0-e270-11e6-9ae3-503aca2616fd/InstanceWaitHandle?AWSAccessKeyId=AKIAIIT3CWAIMJYUTISA&Expires=1485374969&Signature=%2BOSoq4TNvVJQBqaoJVimSO2ddVk%3D

14:09:29
UTC- CREATE_IN_PROGRESS AWS::EC2::InternetGateway InternetGateway
0600
Physical ID:

14:09:29
UTC- CREATE_IN_PROGRESS AWS::IAM::Role CredentialRole
0600
Physical ID:

14:09:28
UTC- CREATE_IN_PROGRESS AWS::CloudFormation::WaitConditionHandle InstanceWaitHandle
0600
Physical ID:

14:09:22
UTC- CREATE_IN_PROGRESS AWS::CloudFormation::Stack AWSMPBASICCloudControllerwithnewVPC User Initiated
0600
Physical ID:arn:aws:cloudformation:us-east-1:872510204642:stack/AWSMPBASICCloudControllerwithnewVPC/feab1be0-e270-11e6-9ae3-503aca2616fd

Loading...

[View more events](#)

Template

[View/Edit template in Designer](#)

View original template View processed template

```
{
  "AWSTemplateFormatVersion": "2010-09-09",
  "Description": "Deploys Hortonworks Data Cloud for AWS into a new VPC. --AWSMP::571fb43d-99f6-4182-8166-61c477473f09::9635be39-350e-40b5-98b9-f0db258f3674",
  "Parameters": {

    "KeyName": {
      "Description": "Name of an existing EC2 KeyPair to enable SSH access to the instance",
      "Type": "AWS::EC2::KeyPair::KeyName",
      "MinLength": "1",
      "ConstraintDescription": "must be the name of an existing EC2 KeyPair."
    },
    "RemoteLocation": {
      "Description": "Allow connections from this address range. Must be a valid CIDR IP. Learn more at http://tinyurl.com/hdcloud-1-8-0-cidr-2.",
      "ConstraintDescription": "must be a single IP address or an IP address range in CIDR notation (More information at http://tinyurl.com/hdcloud-1-8-0-cidr-2.",
      "Type": "String",
      "AllowedPattern": "^(\\[[0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]\\]\\.)\\{3\\}(\\[0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]\\]\\|\\|\\(\\[0-9]|[1-2][0-9]|3[0-2])\\)$"
    },
    "EmailAddress": {
      "Description": "Username for the Admin account. Must be a valid email address.",
      "Type": "String",
      "MinLength": "5",
      "MaxLength": "100",
      "ConstraintDescription": "must be a valid e-mail address",
      "AllowedPattern": "^[a-z0-9!#$%&'*/=?^_`{|}~-]+(?:\\.\\[a-z0-9!#$%&'*/=?^_`{|}~-]+)*\\(?(?:\\x01-\\x08\\|\\x0b\\|\\x0c\\|\\x0e-\\x1f\\|\\x21\\|\\x23-\\x5b\\|\\x5d-\\x7f\\|\\\\\\\\\\|\\x01-\\x09\\|\\x0b\\|\\x0c\\|\\x0e-\\x7f\\])*\\)$"
    },
    "AdminPassword": {
      "Description": "Password for Admin account. Must be at least 8 characters containing letters, numbers and symbols.",
      "Type": "String",
      "NoEcho": "true",
      "MinLength": "8",
      "MaxLength": "50",
      "ConstraintDescription": "must be at least 8 characters containing letters, numbers and symbols",
      "AllowedPattern": "^(?=.*\\d)(?=.*[a-z])(?!.*\\s){8,50}$"
    },
    "InstanceType": {
      "Description": "EC2 instance type to use for the cloud controller.",
      "Type": "String",
      "Default": "m4.large",
      "AllowedValues": [ "c3.xlarge", "c3.2xlarge", "c3.4xlarge", "c3.8xlarge", "c4.large", "c4.xlarge", "c4.2xlarge", "c4.4xlarge", "c4.8xlarge", "m3.xlarge", "m3.2xlarge", "m4.large", "m4.xlarge", "m4.2xlarge" ]
    },
    "ProductTelemetry": {
      "Description": "By Opting-In, you read and agree to the SmartSense terms at http://tinyurl.com/hdcloud-1-8-0-smartsense-1.",
      "Type": "String",
      "Default": "I Do Not Opt In to SmartSense Telemetry",
      "AllowedValues": [ "I Have Read and Opt In to SmartSense Telemetry", "I Do Not Opt In to SmartSense Telemetry" ]
    }
  },
  "Conditions": {
    "AllowProductTelemetry": { "Fn::Equals": [ {"Ref": "ProductTelemetry"}, "I Have Read and Opt In to SmartSense Telemetry" ] }
  },
  "Metadata": {
    "AWS::CloudFormation::Interface": {
      "ParameterGroups": [
        {
          "Label": { "default": "General Configuration" },
          "Parameters": [ "InstanceType", "EmailAddress", "AdminPassword", "ProductTelemetry" ]
        },
        {
          "Label": { "default": "Security Configuration" },
          "Parameters": [ "KeyName", "RemoteLocation" ]
        }
      ]
    },
    "ParameterLabels": {
      "InstanceType": { "default": "Controller Instance Type" },
    }
  }
}
```

```

    "EmailAddress" : { "default" : "Email Address" },
    "AdminPassword" : { "default" : "Admin Password" },
    "ProductTelemetry" : { "default" : "SmartSense" },
    "KeyName" : { "default" : "SSH Key Name" },
    "RemoteLocation" : { "default" : "Remote Access" }
  }
},
"Mappings": {
  "AWSRegionAMI": { "ap-northeast-1":{"ami":"ami-7d53f61c"}, "ap-southeast-1":{"ami":"ami-2cf0564f"}, "ap-southeast-2":{"ami":"ami-c1d9eba2"}, "eu-central-1":{"ami":"ami-d6619fb9"}, "eu-west-1":{"ami":"ami-c
},
"Resources" : {

  "VPC" : {
    "Type" : "AWS::EC2::VPC",
    "Properties" : {
      "CidrBlock" : "10.0.0.0/16",
      "EnableDnsSupport" : "true",
      "EnableDnsHostnames" : "true",
      "Tags" : [
        { "Key" : "Application", "Value" : { "Ref" : "AWS::StackId" } }
      ]
    }
  },

  "PublicSubnet" : {
    "Type" : "AWS::EC2::Subnet",
    "Properties" : {
      "MapPublicIpOnLaunch" : true,
      "VpcId" : { "Ref" : "VPC" },
      "CidrBlock" : "10.0.0.0/24",
      "Tags" : [
        { "Key" : "Application", "Value" : { "Ref" : "AWS::StackId" } }
      ]
    }
  },

  "InternetGateway" : {
    "Type" : "AWS::EC2::InternetGateway",
    "Properties" : {
      "Tags" : [
        { "Key" : "Application", "Value" : { "Ref" : "AWS::StackId" } }
      ]
    }
  },

  "AttachGateway" : {
    "Type" : "AWS::EC2::VPCGatewayAttachment",
    "Properties" : {
      "VpcId" : { "Ref" : "VPC" },
      "InternetGatewayId" : { "Ref" : "InternetGateway" }
    }
  },

  "PublicRouteTable" : {
    "Type" : "AWS::EC2::RouteTable",
    "Properties" : {
      "VpcId" : { "Ref" : "VPC" },
      "Tags" : [
        { "Key" : "Application", "Value" : { "Ref" : "AWS::StackId" } }
      ]
    }
  },

  "PublicRoute" : {
    "Type" : "AWS::EC2::Route",
    "DependsOn" : [ "PublicRouteTable", "AttachGateway" ],
    "Properties" : {
      "RouteTableId" : { "Ref" : "PublicRouteTable" },
      "DestinationCidrBlock" : "0.0.0.0/0",

```

```

    "GatewayId" : { "Ref" : "InternetGateway" }
  }
},
"PublicSubnetRouteTableAssociation" : {
  "Type" : "AWS::EC2::SubnetRouteTableAssociation",
  "Properties" : {
    "SubnetId" : { "Ref" : "PublicSubnet" },
    "RouteTableId" : { "Ref" : "PublicRouteTable" }
  }
},

"CloudbreakRole": {
  "Type": "AWS::IAM::Role",
  "Properties": {
    "AssumeRolePolicyDocument": {
      "Version" : "2012-10-17",
      "Statement": [ {
        "Effect": "Allow",
        "Principal": {
          "Service": [ "ec2.amazonaws.com" ]
        },
        "Action": [ "sts:AssumeRole" ]
      } ]
    },
    "Path": "/",
    "Policies": [ {
      "PolicyName": "root",
      "PolicyDocument": {
        "Version" : "2012-10-17",
        "Statement": [
          {
            "Effect": "Allow",
            "Action": [ "sts:AssumeRole" ],
            "Resource": "*"
          },
          {
            "Effect": "Allow",
            "Action": [ "ec2:DescribeKeyPairs",
              "ec2:DescribeInternetGateways",
              "ec2:DescribeVpcs",
              "ec2:DescribeSubnets",
              "cloudformation:DescribeStackResource",
              "iam:ListRoles" ],
            "Resource": "*"
          }
        ]
      }
    } ]
  }
},
"RootInstanceProfile": {
  "Type": "AWS::IAM::InstanceProfile",
  "Properties": {
    "Path": "/",
    "Roles": [ {
      "Ref": "CloudbreakRole"
    } ]
  }
},

"CredentialRole": {
  "Type": "AWS::IAM::Role",
  "Properties": {
    "AssumeRolePolicyDocument": {
      "Version" : "2012-10-17",
      "Statement": [ {
        "Effect": "Allow",
        "Principal": {
          "AWS": { "Fn::Join" : [ "", [ "arn:aws:iam::", { "Ref" : "AWS::AccountId" }, ":root" ] ] }
        },
        "Action": [ "sts:AssumeRole" ],
        "Condition": {

```



```

    "StringEquals": {
      "sts:ExternalId": "provision-ambari"
    }
  }
},
"Path": "/",
"Policies": [ {
  "PolicyName": "root",
  "PolicyDocument": {
    "Version": "2012-10-17",
    "Statement": [ {
      "Effect": "Allow",
      "Action": [ "ec2:DeleteRoute",
        "ec2:DeleteRouteTable",
        "ec2:DeleteSecurityGroup",
        "ec2:StartInstances",
        "ec2:StopInstances",
        "ec2:TerminateInstances" ],
      "Resource": "*"
    }
  ], {
    "Effect": "Allow",
    "Action": [ "autoscaling:CreateAutoScalingGroup",
      "autoscaling:CreateLaunchConfiguration",
      "autoscaling:DeleteAutoScalingGroup",
      "autoscaling:DeleteLaunchConfiguration",
      "autoscaling:DescribeAutoScalingGroups",
      "autoscaling:DescribeLaunchConfigurations",
      "autoscaling:DescribeScalingActivities",
      "autoscaling:DetachInstances",
      "autoscaling:ResumeProcesses",
      "autoscaling:SuspendProcesses",
      "autoscaling:UpdateAutoScalingGroup",
      "cloudformation:CreateStack",
      "cloudformation:DeleteStack",
      "cloudformation:DescribeStackEvents",
      "cloudformation:DescribeStackResource",
      "cloudformation:DescribeStacks",
      "iam:ListRolePolicies",
      "iam:GetRolePolicy",
      "iam:CreateInstanceProfile",
      "iam:AddRoleToInstanceProfile",
      "iam:ListAttachedRolePolicies",
      "iam:PutRolePolicy",
      "iam:PassRole",
      "iam:CreateRole",
      "iam:DeleteRolePolicy",
      "iam:DeleteRole",
      "iam:RemoveRoleFromInstanceProfile",
      "iam:DeleteInstanceProfile",
      "ec2:AllocateAddress",
      "ec2:AssociateAddress",
      "ec2:AssociateRouteTable",
      "ec2:AuthorizeSecurityGroupIngress",
      "ec2:CreateRoute",
      "ec2:CreateRouteTable",
      "ec2:CreateSecurityGroup",
      "ec2:CreateSubnet",
      "ec2:CreateTags",
      "ec2:DeleteSubnet",
      "ec2:DisassociateAddress",
      "ec2:DisassociateRouteTable",
      "ec2:ModifySubnetAttribute",
      "ec2:ReleaseAddress",
      "ec2:DescribeAddresses",
      "ec2:DescribeImages",
      "ec2:DescribeInstanceStatus",
      "ec2:DescribeInstances",
      "ec2:DescribeInternetGateways",
      "ec2:DescribeKeyPairs",
      "ec2:DescribeRouteTables",
      "ec2:DescribeSecurityGroups",
      "ec2:DescribeSubnets",

```

```

        "ec2:DescribeVpcs",
        "ec2:DescribeVpcAttribute" ],
    "Resource": "*"
  } ]
} ]
},
"Cloudbreak": {
  "Type": "AWS::EC2::Instance",
  "DependsOn" : [ "PublicSubnetRouteTableAssociation", "PublicRoute", "CredentialRole" ],
  "Metadata" : {
    "profile" : { "Fn::Join" : [ "", [
      "export CREDENTIAL_ROLE_ARN=", { "Fn::GetAtt" : [ "CredentialRole", "Arn" ] }, "'\n",
      "export KEYPAIR_NAME=", { "Ref" : "KeyName" }, "\n",
      "export PRODUCT_TELEMETRY=", { "Ref" : "ProductTelemetry" }, "'\n",
      "export WAIT_HANDLE_URL=", { "Ref" : "InstanceWaitHandle" }, "'\n",
      "export VPC_ID=", { "Ref" : "VPC" }, "'\n",

      "export IGW_ID=", { "Ref" : "InternetGateway" }, "'\n",

      "export CBD_VERSION=1.8.0\n",
      "export BUILD_CLUSTER=false\n",
      "export EXISTING_VPC=\n",

      "export GA=true\n",

      "export RDS_URL=", "\n",
      "\n",
      ""
    ] ] },
    "cbdprofile" : { "Fn::Join" : [ "", [
      "# START-CF-TEMPLATE\n",

      "export CB_BLUEPRINT_DEFAULTS='Data Science: Apache Spark 1.6, Apache Zeppelin 0.6.0=hdp25-data-science;EDW-ETL: Apache Hive 1.2.1, Apache Spark 1.6=hdp25-etl-edw;EDW-ETL: Apache Hive 1.2.1, Apaci

      "export PUBLIC_IP=$(curl -4fs 169.254.169.254/latest/meta-data/public-hostname)", "\n",
      "export COMPOSE_HTTP_TIMEOUT=240", "\n",
      "export UAA_DEFAULT_USER_EMAIL=", { "Ref" : "EmailAddress" }, "", "\n",
      "export UAA_DEFAULT_USER_PW=", { "Ref" : "AdminPassword" }, "", "\n",
      "export CB_AWS_DEFAULT_CF_TAG=", { "Ref" : "AWS::StackId" }, "", "\n",
      "export ULU_HWX_CLOUD_DEFAULT_CREDENTIAL=aws-access", "\n",
      "export ULU_HWX_CLOUD_DEFAULT_REGION=", { "Ref" : "AWS::Region" }, "\n",
      "export CB_PLATFORM_DEFAULT_REGIONS=AWS:", { "Ref" : "AWS::Region" }, "\n",
      "export ULU_HWX_CLOUD_DEFAULT_VPC_ID=", { "Ref" : "VPC" }, "", "\n",
      "export ULU_HWX_CLOUD_DEFAULT_SSH_KEY=", { "Ref" : "KeyName" }, "\n",
      "export DOCKER_IMAGE_CLOUDBREAK_WEB=hortonworks/cloud-web", "\n",
      "export DOCKER_IMAGE_CLOUDBREAK_AUTH=hortonworks/cloud-auth", "\n",
      "export AWS_ACCOUNT_ID=", { "Ref" : "AWS::AccountId" }, "", "\n",
      "export CLOUDBREAK_TELEMETRY_MAIL_ADDRESS=aws-marketplace@hortonworks.com", "\n",
      "export HWX_CLOUD_TYPE=advanced", "\n",
      "export HWX_CLOUD_TEMPLATE_VERSION=1.8.0", "\n",
      "export AWS_AMI_ID=$(curl -4fs 169.254.169.254/latest/meta-data/ami-id)", "\n",
      "export AWS_INSTANCE_ID=$(curl -4fs 169.254.169.254/latest/meta-data/instance-id)", "\n",
      "export CB_SMARTSENSE_CONFIGURE=", { "Fn::If" : [ "AllowProductTelemetry", "true", "false" ] }, "\n",

      "export HWX_DISPLAY_TERMS_AND_CONDITIONS=false", "\n",
      "export HWX_HCC_AVAILABLE=true", "\n",

      "export CB_IMAGE_CATALOG_URL=https://s3-eu-west-1.amazonaws.com/cloudbreak-info/marketplace-cb-image-catalog.json", "\n",
      "export ULU_HWX_CLOUD_REGISTRATION_URL=https://service.hdccloud.hortonworks.com", "\n",

      "export ULU_HWX_CLOUD_DEFAULT_IGW_ID=", { "Ref" : "InternetGateway" }, "", "\n",

```

```

    "export ULU_SUBSCRIBE_TO_NOTIFICATIONS=false", "\n",
    "export CB_DEFAULT_SUBSCRIPTION_ADDRESS=http://uluwatu.service.consul:3000/notifications", "\n",
    "export CB_AWS_DEFAULT_INBOUND_SECURITY_GROUP=", { "Ref" : "CloudbreakSecurityGroup" }, "\n",
    "export CB_AWS_VPC=", { "Ref" : "VPC" }, "", "\n",
    "# END-CF-TEMPLATE\n"
  ]]]
},
"Properties": {
  "IamInstanceProfile": { "Ref" : "RootInstanceProfile" },
  "ImageId" : { "Fn::FindInMap" : [ "AWSRegionAMI", { "Ref" : "AWS::Region" }, "ami" ] },

  "SubnetId" : { "Ref" : "PublicSubnet" },
  "SecurityGroupIds" : [ { "Ref" : "CloudbreakSecurityGroup" } ],

  "InstanceType" : { "Ref" : "InstanceType" },
  "KeyName" : { "Ref" : "KeyName" },
  "Tags" : [
    { "Key" : "Name", "Value" : { "Fn::Join" : [ "", [ { "Ref" : "AWS::StackName" }, "-cbd" ] ] } }
  ],
  "UserData" : "IyEvYm1uL2Jhc2gKCjogJHtUUKFDRTO9MX0KOiAke0RFQ1VH0j0xfQo6ICR7REVQTE9ZRVJfTE9DQVRJT046PSIvdmFyL2xpYi9jbG9lZGJyZWFrLWRLcGxveW11bnQiQ06ICR7QVdTX0JJT19MT0NBVE1PTjo9Ii9vcHQvYXdlL2Jpb:
},
"InstanceWaitHandle" : {
  "Type" : "AWS::CloudFormation::WaitConditionHandle"
},
"InstanceWaitCondition" : {
  "Type" : "AWS::CloudFormation::WaitCondition",
  "DependsOn" : "Cloudbreak",
  "Properties" : {
    "Handle" : { "Ref" : "InstanceWaitHandle" },
    "Timeout" : 3600
  }
},
"CloudbreakSecurityGroup" : {
  "Type" : "AWS::EC2::SecurityGroup",
  "Properties" : {
    "VpcId" : { "Ref" : "VPC" },
    "GroupDescription" : "Enable HTTP access to Cloud UI and SSH access from a remote location",
    "SecurityGroupIngress" : [
      { "IpProtocol" : "tcp", "FromPort" : "80", "ToPort" : "80", "CidrIp" : { "Ref" : "Remotelocation" } },
      { "IpProtocol" : "tcp", "FromPort" : "443", "ToPort" : "443", "CidrIp" : { "Ref" : "Remotelocation" } },
      { "IpProtocol" : "tcp", "FromPort" : "22", "ToPort" : "22", "CidrIp" : { "Ref" : "Remotelocation" } }
    ]
  }
},
"Outputs" : {
  "CloudURL" : {
    "Value" : { "Fn::Join" : [ "", [ "https://", { "Fn::GetAtt" : [ "Cloudbreak", "PublicDnsName" ] ] ] } },
    "Description" : "URL to the Cloud UI. Login using the Admin account information you provided during Create Stack."
  },
  "SshAccess" : {
    "Value" : { "Fn::Join" : [ "", [ "ssh -i ", { "Ref" : "KeyName" }, ".pem cloudbreak@", { "Fn::GetAtt" : [ "Cloudbreak", "PublicIp" ] ] ] } },
    "Description" : "Access the instance via ssh"
  }
}
}
}

```

Loading...

```

{
  "AWSTemplateFormatVersion": "2010-09-09",

```

```

"Description": "Deploys Hortonworks Data Cloud for AWS into a new VPC. --AWSMP::571fb43d-99f6-4182-8166-61c477473f09::9635be39-350e-40b5-98b9-f0db258f3674",
"Parameters": {

  "KeyName": {
    "Description": "Name of an existing EC2 KeyPair to enable SSH access to the instance",
    "Type": "AWS::EC2::KeyPair::KeyName",
    "MinLength": "1",
    "ConstraintDescription": "must be the name of an existing EC2 KeyPair."
  },
  "RemoteLocation": {
    "Description": "Allow connections from this address range. Must be a valid CIDR IP. Learn more at http://tinyurl.com/hdcloud-1-8-0-cidr-2.",
    "ConstraintDescription": "must be a single IP address or an IP address range in CIDR notation (More information at http://tinyurl.com/hdcloud-1-8-0-cidr-2.",
    "Type": "String",
    "AllowedPattern": "^(?!(?:[0-9]{1,3}[.]{1,4}){4}|(?:[0-9]{1,3}[.]{1,4}){3}|(?:[0-9]{1,3}[.]{1,4}){2}|(?:[0-9]{1,3}[.]{1,4}){1}|(?:[0-9]{1,3}[.]{1,4}){0})$)"
  },
  "EmailAddress" : {
    "Description" : "Username for the Admin account. Must be a valid email address.",
    "Type" : "String",
    "MinLength" : "5",
    "MaxLength" : "100",
    "ConstraintDescription" : "must be a valid e-mail address",
    "AllowedPattern" : "^(?=[a-z0-9!#$%&'*/+=?^_`{|}~-]+(?:\\.[a-z0-9!#$%&'*/+=?^_`{|}~-]+)*|"(?:[\\x01-\\x08\\x0b\\x0c\\x0e-\\x1f\\x21\\x23-\\x5b\\x5d-\\x7f]|\\\\"[\\x01-\\x09\\x0b\\x0c\\x0e-\\x7f])*\\")$)"
  },
  "AdminPassword" : {
    "Description" : "Password for Admin account. Must be at least 8 characters containing letters, numbers and symbols.",
    "Type" : "String",
    "NoEcho": "true",
    "MinLength" : "8",
    "MaxLength" : "50",
    "ConstraintDescription" : "must be at least 8 characters containing letters, numbers and symbols",
    "AllowedPattern" : "^(?=.*\\d)(?=.*[a-z])(?!.*\\s){8,50}$"
  },
  "InstanceType" : {
    "Description" : "EC2 instance type to use for the cloud controller.",
    "Type" : "String",
    "Default" : "m4.large",
    "AllowedValues" : [ "c3.xlarge", "c3.2xlarge", "c3.4xlarge", "c3.8xlarge", "c4.large", "c4.xlarge", "c4.2xlarge", "c4.4xlarge", "c4.8xlarge", "m3.xlarge", "m3.2xlarge", "m4.large", "m4.xlarge", "m4.2" ]
  },
  "ProductTelemetry" : {
    "Description" : "By Opting-In, you read and agree to the SmartSense terms at http://tinyurl.com/hdcloud-1-8-0-smartsense-1.",
    "Type" : "String",
    "Default" : "I Do Not Opt In to SmartSense Telemetry",
    "AllowedValues" : [ "I Have Read and Opt In to SmartSense Telemetry", "I Do Not Opt In to SmartSense Telemetry" ]
  }
},

"Conditions" : {
  "AllowProductTelemetry" : { "Fn::Equals" : [{"Ref" : "ProductTelemetry"}, "I Have Read and Opt In to SmartSense Telemetry"] }
},

"Metadata" : {
  "AWS::CloudFormation::Interface" : {
    "ParameterGroups" : [
      {
        "Label" : { "default" : "General Configuration" },
        "Parameters" : [ "InstanceType", "EmailAddress", "AdminPassword", "ProductTelemetry" ]
      },
      {
        "Label" : { "default" : "Security Configuration" },
        "Parameters" : [ "KeyName", "RemoteLocation" ]
      }
    ]
  },
  "ParameterLabels" : {
    "InstanceType" : { "default" : "Controller Instance Type" },
    "EmailAddress" : { "default" : "Email Address" },
    "AdminPassword" : { "default" : "Admin Password" },
    "ProductTelemetry" : { "default" : "SmartSense" },
    "KeyName" : { "default" : "SSH Key Name" },
    "RemoteLocation" : { "default" : "Remote Access" }
  }
}

```

```

    }
  },
  "Mappings": {
    "AWSRegionAMI": { "ap-northeast-1":{"ami":"ami-7d53f61c"}, "ap-southeast-1":{"ami":"ami-2cf0564f"}, "ap-southeast-2":{"ami":"ami-c1d9eba2"}, "eu-central-1":{"ami":"ami-d6619fb9"}, "eu-west-1":{"ami":"ami-c
  },
  "Resources" : {

    "VPC" : {
      "Type" : "AWS::EC2::VPC",
      "Properties" : {
        "CidrBlock" : "10.0.0.0/16",
        "EnableDnsSupport" : "true",
        "EnableDnsHostnames" : "true",
        "Tags" : [
          { "Key" : "Application", "Value" : { "Ref" : "AWS::StackId" } }
        ]
      }
    },

    "PublicSubnet" : {
      "Type" : "AWS::EC2::Subnet",
      "Properties" : {
        "MapPublicIpOnLaunch" : true,
        "VpcId" : { "Ref" : "VPC" },
        "CidrBlock" : "10.0.0.0/24",
        "Tags" : [
          { "Key" : "Application", "Value" : { "Ref" : "AWS::StackId" } }
        ]
      }
    },

    "InternetGateway" : {
      "Type" : "AWS::EC2::InternetGateway",
      "Properties" : {
        "Tags" : [
          { "Key" : "Application", "Value" : { "Ref" : "AWS::StackId" } }
        ]
      }
    },

    "AttachGateway" : {
      "Type" : "AWS::EC2::VPCGatewayAttachment",
      "Properties" : {
        "VpcId" : { "Ref" : "VPC" },
        "InternetGatewayId" : { "Ref" : "InternetGateway" }
      }
    },

    "PublicRouteTable" : {
      "Type" : "AWS::EC2::RouteTable",
      "Properties" : {
        "VpcId" : { "Ref" : "VPC" },
        "Tags" : [
          { "Key" : "Application", "Value" : { "Ref" : "AWS::StackId" } }
        ]
      }
    },

    "PublicRoute" : {
      "Type" : "AWS::EC2::Route",
      "DependsOn" : [ "PublicRouteTable", "AttachGateway" ],
      "Properties" : {
        "RouteTableId" : { "Ref" : "PublicRouteTable" },
        "DestinationCidrBlock" : "0.0.0.0/0",
        "GatewayId" : { "Ref" : "InternetGateway" }
      }
    },

    "PublicSubnetRouteTableAssociation" : {

```

```

    "Type" : "AWS::EC2::SubnetRouteTableAssociation",
    "Properties" : {
      "SubnetId" : { "Ref" : "PublicSubnet" },
      "RouteTableId" : { "Ref" : "PublicRouteTable" }
    }
  },
  "CloudbreakRole": {
    "Type": "AWS::IAM::Role",
    "Properties": {
      "AssumeRolePolicyDocument": {
        "Version" : "2012-10-17",
        "Statement": [ {
          "Effect": "Allow",
          "Principal": {
            "Service": [ "ec2.amazonaws.com" ]
          },
          "Action": [ "sts:AssumeRole" ]
        } ]
      },
      "Path": "/",
      "Policies": [ {
        "PolicyName": "root",
        "PolicyDocument": {
          "Version" : "2012-10-17",
          "Statement": [
            {
              "Effect": "Allow",
              "Action": [ "sts:AssumeRole" ],
              "Resource": "*"
            },
            {
              "Effect": "Allow",
              "Action": [ "ec2:DescribeKeyPairs",
                "ec2:DescribeInternetGateways",
                "ec2:DescribeVpcs",
                "ec2:DescribeSubnets",
                "cloudformation:DescribeStackResource",
                "iam:ListRoles" ],
              "Resource": "*"
            }
          ]
        }
      } ]
    }
  },
  "RootInstanceProfile": {
    "Type": "AWS::IAM::InstanceProfile",
    "Properties": {
      "Path": "/",
      "Roles": [ {
        "Ref": "CloudbreakRole"
      } ]
    }
  },
  "CredentialRole": {
    "Type": "AWS::IAM::Role",
    "Properties": {
      "AssumeRolePolicyDocument": {
        "Version" : "2012-10-17",
        "Statement": [ {
          "Effect": "Allow",
          "Principal": {
            "AWS": { "Fn::Join" : [ "", [ "arn:aws:iam::", { "Ref" : "AWS::AccountId" }, ":root" ] ] }
          },
          "Action": [ "sts:AssumeRole" ],
          "Condition": {
            "StringEquals": {
              "sts:ExternalId": "provision-ambari"
            }
          }
        } ]
      }
    }
  }
}

```

```

},
"Path": "/",
"Policies": [ {
  "PolicyName": "root",
  "PolicyDocument": {
    "Version" : "2012-10-17",
    "Statement": [ {
      "Effect": "Allow",
      "Action": [ "ec2:DeleteRoute",
                  "ec2:DeleteRouteTable",
                  "ec2:DeleteSecurityGroup",
                  "ec2:StartInstances",
                  "ec2:StopInstances",
                  "ec2:TerminateInstances" ],
      "Resource": "*"
    }
  ],
  "Resource": "*"
}, {
  "Effect": "Allow",
  "Action": [ "autoscaling:CreateAutoScalingGroup",
              "autoscaling:CreateLaunchConfiguration",
              "autoscaling:DeleteAutoScalingGroup",
              "autoscaling:DeleteLaunchConfiguration",
              "autoscaling:DescribeAutoScalingGroups",
              "autoscaling:DescribeLaunchConfigurations",
              "autoscaling:DescribeScalingActivities",
              "autoscaling:DetachInstances",
              "autoscaling:ResumeProcesses",
              "autoscaling:SuspendProcesses",
              "autoscaling:UpdateAutoScalingGroup",
              "cloudformation:CreateStack",
              "cloudformation:DeleteStack",
              "cloudformation:DescribeStackEvents",
              "cloudformation:DescribeStackResource",
              "cloudformation:DescribeStacks",
              "iam:ListRolePolicies",
              "iam:GetRolePolicy",
              "iam:CreateInstanceProfile",
              "iam:AddRoleToInstanceProfile",
              "iam:ListAttachedRolePolicies",
              "iam:PutRolePolicy",
              "iam:PassRole",
              "iam:CreateRole",
              "iam:DeleteRolePolicy",
              "iam:DeleteRole",
              "iam:RemoveRoleFromInstanceProfile",
              "iam:DeleteInstanceProfile",
              "ec2:AllocateAddress",
              "ec2:AssociateAddress",
              "ec2:AssociateRouteTable",
              "ec2:AuthorizeSecurityGroupIngress",
              "ec2:CreateRoute",
              "ec2:CreateRouteTable",
              "ec2:CreateSecurityGroup",
              "ec2:CreateSubnet",
              "ec2:CreateTags",
              "ec2:DeleteSubnet",
              "ec2:DisassociateAddress",
              "ec2:DisassociateRouteTable",
              "ec2:ModifySubnetAttribute",
              "ec2:ReleaseAddress",
              "ec2:DescribeAddresses",
              "ec2:DescribeImages",
              "ec2:DescribeInstanceStatus",
              "ec2:DescribeInstances",
              "ec2:DescribeInternetGateways",
              "ec2:DescribeKeyPairs",
              "ec2:DescribeRouteTables",
              "ec2:DescribeSecurityGroups",
              "ec2:DescribeSubnets",
              "ec2:DescribeVpcs",
              "ec2:DescribeVpcAttribute" ],
  "Resource": "*"
} ]
}

```

```

    } ]
  }
},
"Cloudbreak": {
  "Type": "AWS::EC2::Instance",
  "DependsOn": [ "PublicSubnetRouteTableAssociation", "PublicRoute", "CredentialRole" ],
  "Metadata": {
    "profile": { "Fn::Join": [ "", [
      "export CREDENTIAL_ROLE_ARN=", { "Fn::GetAtt": [ "CredentialRole", "Arn" ] }, "'\n",
      "export KEYPAIR_NAME=", { "Ref": "KeyName" }, "\n",
      "export PRODUCT_TELEMETRY=", { "Ref": "ProductTelemetry" }, "'\n",
      "export WAIT_HANDLE_URL=", { "Ref": "InstanceWaitHandle" }, "'\n",
      "export VPC_ID=", { "Ref": "VPC" }, "'\n",

      "export IGW_ID=", { "Ref": "InternetGateway" }, "'\n",

      "export CBD_VERSION=1.8.0\n",
      "export BUILD_CLUSTER=false\n",
      "export EXISTING_VPC=\n",

      "export GA=true\n",

      "export RDS_URL=", "\n",
      "\n",
      ""
    ] ] },
    "cbdprofile": { "Fn::Join": [ "", [
      "# START-CF-TEMPLATE\n",

      "export CB_BLUEPRINT_DEFAULTS='Data Science: Apache Spark 1.6, Apache Zeppelin 0.6.0=hdp25-data-science;EDW-ETL: Apache Hive 1.2.1, Apache Spark 1.6=hdp25-etl-edw;EDW-ETL: Apache Hive 1.2.1, Apac

      "export PUBLIC_IP=$(curl -4fs 169.254.169.254/latest/meta-data/public-hostname)", "\n",
      "export COMPOSE_HTTP_TIMEOUT=240", "\n",
      "export UAA_DEFAULT_USER_EMAIL=", { "Ref": "EmailAddress" }, "'", "\n",
      "export UAA_DEFAULT_USER_PW=", { "Ref": "AdminPassword" }, "'", "\n",
      "export CB_AWS_DEFAULT_CF_TAG=", { "Ref": "AWS::StackId" }, "'", "\n",
      "export ULU_HWX_CLOUD_DEFAULT_CREDENTIAL=aws-access", "\n",
      "export ULU_HWX_CLOUD_DEFAULT_REGION=", { "Ref": "AWS::Region" }, "\n",
      "export CB_PLATFORM_DEFAULT_REGIONS=AWS:", { "Ref": "AWS::Region" }, "\n",
      "export ULU_HWX_CLOUD_DEFAULT_VPC_ID=", { "Ref": "VPC" }, "'", "\n",
      "export ULU_HWX_CLOUD_DEFAULT_SSH_KEY=", { "Ref": "KeyName" }, "\n",
      "export DOCKER_IMAGE_CLOUDBREAK_WEB=hortonworks/cloud-web", "\n",
      "export DOCKER_IMAGE_CLOUDBREAK_AUTH=hortonworks/cloud-auth", "\n",
      "export AWS_ACCOUNT_ID=", { "Ref": "AWS::AccountId" }, "'", "\n",
      "export CLOUDBREAK_TELEMETRY_MAIL_ADDRESS=aws-marketplace@hortonworks.com", "\n",
      "export HWX_CLOUD_TYPE=advanced", "\n",
      "export HWX_CLOUD_TEMPLATE_VERSION=1.8.0", "\n",
      "export AWS_AMI_ID=$(curl -4fs 169.254.169.254/latest/meta-data/ami-id)", "\n",
      "export AWS_INSTANCE_ID=$(curl -4fs 169.254.169.254/latest/meta-data/instance-id)", "\n",
      "export CB_SMARTSENSE_CONFIGURE=", { "Fn::If": [ "AllowProductTelemetry", "true", "false" ] }, "\n",

      "export HWX_DISPLAY_TERMS_AND_CONDITIONS=false", "\n",
      "export HWX_HCC_AVAILABLE=true", "\n",

      "export CB_IMAGE_CATALOG_URL=https://s3-eu-west-1.amazonaws.com/cloudbreak-info/marketplace-cb-image-catalog.json", "\n",
      "export ULU_HWX_CLOUD_REGISTRATION_URL=https://service.hdcloud.hortonworks.com", "\n",

      "export ULU_HWX_CLOUD_DEFAULT_IGW_ID=", { "Ref": "InternetGateway" }, "'", "\n",

      "export ULU_SUBSCRIBE_TO_NOTIFICATIONS=false", "\n",
      "export CB_DEFAULT_SUBSCRIPTION_ADDRESS=http://uluwatu.service.consul:3000/notifications", "\n",
      "export CB_AWS_DEFAULT_INBOUND_SECURITY_GROUP=", { "Ref": "CloudbreakSecurityGroup" }, "\n",
      "export CB_AWS_VPC=", { "Ref": "VPC" }, "'", "\n",
      "# END-CF-TEMPLATE\n"
    ] ] }
  }
}

```



```

    ]}]
  },
  "Properties": {
    "IamInstanceProfile": { "Ref" : "RootInstanceProfile" },
    "ImageId" : { "Fn::FindInMap" : [ "AWSRegionAMI", { "Ref" : "AWS::Region" }, "ami" ] },

    "SubnetId" : { "Ref" : "PublicSubnet" },
    "SecurityGroupIds" : [ { "Ref" : "CloudbreakSecurityGroup" } ],

    "InstanceType" : { "Ref" : "InstanceType" },
    "KeyName" : { "Ref" : "KeyName" },
    "Tags" : [
      { "Key" : "Name", "Value" : { "Fn::Join" : [ "", [ { "Ref" : "AWS::StackName" }, "-cbd" ] ] } }
    ],
    "UserData" : "IyEvYm1uL2Jhc2gKCjogJHtUUkFDRT09Mx0K0iAke0RFQlVHOj0xfQo6ICR7REVQTE9ZRVJfTE9DQVRJT046PSIvdmFyL2xpYi9jbG91ZGJyZWFrLWR1cGxveW11bnQifQo6ICR7QVdTX0J1T19MT0NBVE1PTjo9Ii9vcHQvYXZlL2Jpb:
  },
  "InstanceWaitHandle" : {
    "Type" : "AWS::CloudFormation::WaitConditionHandle"
  },
  "InstanceWaitCondition" : {
    "Type" : "AWS::CloudFormation::WaitCondition",
    "DependsOn" : "Cloudbreak",
    "Properties" : {
      "Handle" : { "Ref" : "InstanceWaitHandle" },
      "Timeout" : 36000
    }
  },
  "CloudbreakSecurityGroup" : {
    "Type" : "AWS::EC2::SecurityGroup",
    "Properties" : {
      "VpcId" : { "Ref" : "VPC" },
      "GroupDescription" : "Enable HTTP access to Cloud UI and SSH access from a remote location",
      "SecurityGroupIngress" : [
        { "IpProtocol" : "tcp", "FromPort" : "80", "ToPort" : "80", "CidrIp" : { "Ref" : "RemoteLocation" } },
        { "IpProtocol" : "tcp", "FromPort" : "443", "ToPort" : "443", "CidrIp" : { "Ref" : "RemoteLocation" } },
        { "IpProtocol" : "tcp", "FromPort" : "22", "ToPort" : "22", "CidrIp" : { "Ref" : "RemoteLocation" } }
      ]
    }
  },
  "Outputs" : {
    "CloudURL" : {
      "Value" : { "Fn::Join" : [ "", [ "https://", { "Fn::GetAtt" : [ "Cloudbreak", "PublicDnsName" ] } ] ] },
      "Description" : "URL to the Cloud UI. Login using the Admin account information you provided during Create Stack."
    },
    "SshAccess" : {
      "Value" : { "Fn::Join" : [ "", [ "ssh -i ", { "Ref" : "KeyName" }, ".pem cloudbreak@", { "Fn::GetAtt" : [ "Cloudbreak", "PublicIp" ] } ] ] },
      "Description" : "Access the instance via ssh"
    }
  }
}

```

Parameters

Key	Value
AdminPassword	****
EmailAddress	at46@yahoo.com
InstanceType	c3.xlarge
KeyName	nVirginia-keypair
ProductTelemetry	I Do Not Opt In to SmartSense Telemetry

Key	Value
RemoteLocation	0.0.0.0/24
No Values Found.	

Tags

Key	Value
No Values Found.	

Policy

No stack policy set.

Loading...

Change Sets

The stack was updated with change set at .

Name	Created time	Status	Description
There are no executable change sets. Learn more about change sets or create one .			There are no executable change sets. Click here to create one.
			There are no executable change sets. Your stack must be in an updatable state to use change sets.

[View more change sets](#)

Delete Stack

Are you sure you want to delete this stack?

Deleting a stack deletes all stack resources.

Are you sure you want to try deleting this stack again?

If the stack deletion failed because AWS CloudFormation couldn't delete one or more of the following resources, you can choose to retain them. AWS CloudFormation deletes the stack, skipping the retained resources. Choose the resources to retain:

Logical ID **Physical ID**

No Values Found.

[Cancel](#) [Yes, Delete](#)

Cancel Update

Are you sure you want to cancel the update of this stack?

If you cancel the update, CloudFormation will attempt to rollback your stack's resources to their previous state.

[Cancel](#) [Yes, Cancel Update](#)

Continue Update Rollback

Are you sure you want to continue rolling back of this stack?

Note: You might need to manually fix the error to complete the rollback. [Learn more](#)

Advanced

If you've tried troubleshooting the error and still can't roll back the stack, you can tell CloudFormation to skip the resources that it can't roll back.

To skip resources, type a list of comma-separated logical resource IDs. Include only the resources that are blocking the rollback. [Learn more](#)

Warning

After the rollback completes, skipped resources will be out of sync with the stack template, and you'll need to sync them.

Logical_Id or
StackName.Logical_Id

[Cancel](#)

Loading...

[Feedback](#)

[English](#)

© 2008 - 2017, Amazon Web Services, Inc. or its affiliates. All rights reserved.

[Privacy Policy](#)

[Terms of Use](#)