

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

LAMP Stack

Upload a template to Amazon S3

MongoDB-VPC.template

Specify an Amazon S3 template URL

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

AWS-MongoDB-Infrastructure

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

MongoDB-VP...d.template

Specify an Amazon S3 template URL

`https://s3-us-west-2.amazonaws.com/cf-templates-1tpi214b4bfd8-us-west-2/2015248jla-MongoDB-VP`

Select Template

Specify Parameters

Options

Review

Specify Parameters

Specify values or use the default values for the parameters that are associated with your AWS CloudFormation template. [Learn more.](#)

Parameters

BuildBucket	quickstart-reference/mongodb/latest	Main Bucket where the templates/scripts are installed. Do not change.
ClusterReplicaSetCount	3	Number of Replica Set Members. Choose 1 or 3
ClusterShardCount	2	Number of Shards [0,1,2,3]. 0==No Sharding. Set to > 1 for Sharding
ConfigServerInstanceType	t2.micro	Amazon EC2 instance type for the Config Server
Iops	100	Iops of EBS volume when io1 type is chosen. Otherwise ignored
KeyName	skondla_aws_keys_oregon.pem	Name of an existing EC2 KeyPair. MongoDB instances will launch with this KeyPair.
MongoDBVersion	3.0	MongoDB version
NATInstanceType	t2.small	Amazon EC2 instance type for the NAT Instances.
NodeInstanceType	m3.medium	Amazon EC2 instance type for the MongoDB nodes.
PrimaryReplicaSubnet	10.0.2.0/24	Private Subnet where Primary Replica Set will be deployed.
PublicSubnet	10.0.1.0/24	CIDR Block for the Public DMZ Subnet located in the new VPC.
RemoteAccessCIDR	0.0.0.0/0	IP CIDR from where you could SSH into MongoDB cluster via NAT

Specify Parameters

Specify values or use the default values for the parameters that are associated with your AWS CloudFormation template. [Learn more.](#)

Parameters

BuildBucket quickstart-reference/mongodb/latest Main Bucket where the templates/scripts are installed. Do not change.

ClusterReplicaSetCount 3 Number of Replica Set Members. Choose 1 or 3

ClusterShardCount 2 Number of Shards [0,1,2,3]. 0==No Sharding. Set to > 1 for Sharding

ConfigServerInstanceType t2.micro Amazon EC2 instance type for the Config Server

Iops 100 Iops of EBS volume when io1 type is chosen. Otherwise ignored

KeyName Name of an existing EC2 KeyPair. MongoDB instances will launch with this KeyPair.

MongoDBVersion 3.0 MongoDB version

NATInstanceType t2.micro Amazon EC2 instance type for the NAT Instances.

NodeInstanceType t2.micro Amazon EC2 instance type for the MongoDB nodes.

Secondary Replica Subnet 0 Private Subnet of Secondary Replica Set 0 (Applicable only when ClusterReplicaSetCount >= 2)

Secondary Replica Subnet 1 Private Subnet of Secondary Replica Set 1 (Applicable only when ClusterReplicaSetCount == 3)

Shards Per Node Number of Micro Shards Per Node

Volume Size EBS Volume Size (data) to be attached to node in GBs

Volume Type EBS Volume Type (data) to be attached to node in GBs [io1, gp2]

VPC CIDR CIDR Block for the VPC you are creating.

[Cancel](#) [Previous](#) [Next](#)

PrimaryReplicaSubnet	<input type="text" value="10.0.2.0/24"/>	Private Subnet where Primary Replica Set will be deployed.
PublicSubnet	<input type="text" value="10.0.1.0/24"/>	CIDR Block for the Public DMZ Subnet located in the new VPC.
RemoteAccessCIDR	<input type="text" value="0.0.0.0/0"/>	IP CIDR from where you could SSH into MongoDB cluster via NAT
SecondaryReplicaSubnet 0	<input type="text" value="10.0.3.0/24"/>	Private Subnet of Secondary Replica Set 0 (Applicable only when ClusterReplicaSetCount >= 2)
SecondaryReplicaSubnet 1	<input type="text" value="10.0.4.0/24"/>	Private Subnet of Secondary Replica Set 1 (Applicable only when ClusterReplicaSetCount == 3)
ShardsPerNode	<input style="width: 50px;" type="text" value="2"/> ▼	Number of Micro Shards Per Node
VolumeSize	<input type="text" value="8"/>	EBS Volume Size (data) to be attached to node in GBs
VolumeType	<input style="width: 50px;" type="text" value="gp2"/> ▼	EBS Volume Type (data) to be attached to node in GBs [io1, gp2]
VPCCIDR	<input type="text" value="10.0.0.0/16"/>	CIDR Block for the VPC you are creating.

Options

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	Cluster	MongoDB Sharded Cluster	
2	Environment	Lab	
3	Owner	skondla	
4	Description	Set up MongoDB Cluster on AWS	

Options

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	Cluster	MongoDB Sharded Cluster	
2	Environment	Lab	
3	Owner	skondla	
4	Description	Set up MongoDB Cluster on AWS	
5	Purpose	Deploy MongoDB on an existing	

Advanced

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

[Cancel](#) [Previous](#) [Next](#)

Options

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	Environment	Lab	x
2	Purpose	MongoDB Sharded cluster on AWS	x
3	Instance Type Used	t2.micro	x
4	Region	US-West-2(Oregon)	+

► Advanced

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

[Cancel](#) [Previous](#) [Next](#)

▼ Advanced

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

Notification options

No notification

New Amazon SNS topic

Topic	<input type="text"/>
Email	<input type="text"/>

Existing Amazon SNS topic

Timeout 

Minutes

Rollback on failure 

Yes

No

Stack policy 

Enter policy

Upload policy file

No file chosen

[Learn more](#)

[Cancel](#)

[Previous](#)

[Next](#)

Review

Template

Name	AWS-MongoDB-Infrastructure
Template URL	https://s3-us-west-2.amazonaws.com/cf-templates-1tpi214b4bfd8-us-west-2/2015248vm7-MongoDB-VPC_modified.template
Description	(000F) Deploy MongoDB on a New VPC in AWS
Estimate cost	Cost

Parameters

BuildBucket	quickstart-reference/mongodb/latest
ClusterReplicaSetCount	3
ClusterShardCount	2
Config ServerInstanceType	t2.micro
 e	
 lops	100
 KeyName	
 MongoDBVersion	3.0
 NATInstanceType	t2.micro
 NodeInstanceType	t2.micro
 PrimaryReplicaSubnet	10.0.2.0/24
 PublicSubnet	10.0.1.0/24
 RemoteAccessCIDR	0.0.0.0/0
SecondaryReplicaSubnet	10.0.3.0/24
 0	
SecondaryReplicaSubnet	10.0.4.0/24
 1	
 ShardsPerNode	2
 VolumeSize	8
 VolumeType	gp2
 VPCCIDR	10.0.0.0/16
Create IAM resources	True

Review

Template

Name	AWS-MongoDB-Infrastructure
Template URL	https://s3-us-west-2.amazonaws.com/cf-templates-1tpi214b4bfd8-us-west-2/2015248jla-MongoDB-VPC.template
Description	(000F) Deploy MongoDB on a New VPC in AWS
Estimate cost	Cost

Parameters

BuildBucket	quickstart-reference/mongodb/latest
ClusterReplicaSetCount	3
ClusterShardCount	2
ConfigServerInstanceType	t2.micro
DocumentSize	100
KeyName	skondla_aws_keys_oregon.pem
MongoDBVersion	3.0
NATInstanceType	t2.small
NodeInstanceType	m3.medium
PrimaryReplicaSubnet	10.0.2.0/24
PublicSubnet	10.0.1.0/24
RemoteAccessCIDR	0.0.0.0/0
SecondaryReplicaSubnet	10.0.3.0/24
ShardReplicaSubnet	10.0.4.0/24
ShardsPerNode	2
VolumeSize	10
VolumeType	gp2
VPCCIDR	10.0.0.0/16
Create IAM resources	False

Options

Tags

Cluster	MongoDB Sharded Cluster
Environment	Lab
Owner	skondla
Description	Set up MongoDB Cluster on AWS
Purpose	Deploy MongoDB on an existing VPC in AWS

Advanced

Notification	
Timeout	none
Rollback on failure	Yes

Capabilities

i The following resource(s) require capabilities: [AWS::IAM::InstanceProfile, AWS::IAM::Role]

This template might include Identity and Access Management (IAM) resources, which can include groups, IAM users, and IAM roles with certain permissions. Ensure that the template you are using is from a trusted source. [Learn more](#).

I acknowledge that this template might cause AWS CloudFormation to create IAM resources.

[Cancel](#) [Previous](#) [Create](#)

Options

Tags

Cluster	MongoDB Sharded Cluster
Environment	Lab
Owner	skondla
Description	Set up MongoDB Cluster on AWS

Advanced

Notification	
Timeout	none
Rollback on failure	Yes

Capabilities



The following resource(s) require capabilities: [AWS::IAM::InstanceProfile, AWS::IAM::Role]

This template might include Identity and Access Management (IAM) resources, which can include groups, IAM users, and IAM roles with certain permissions. Ensure that the template you are using is from a trusted source. [Learn more.](#)

I acknowledge that this template might cause AWS CloudFormation to create IAM resources.

Cancel

Previous

Create



SIMPLE MONTHLY CALCULATOR

AWS pricing helps you reduce costs in multiple ways. [Learn more about AWS pricing](#)

FREE USAGE TIER: New Customers get free usage tier for first 12 months

[Reset All](#)

Services

Estimate of your Monthly Bill (\$ 19.04)

Choose region: US-East / US Standard (Virginia) ▾

Inbound Data

Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides resizable compute capacity in the cloud. It is designed to make web-scale computing instances.

Compute: Amazon EC2 Instances:

	Description	Instances	Usage	Type	Billing Option	Monthly Cost
	Add New Row					

Storage: Amazon EBS Volumes:

	Description	Volumes	Volume Type	Storage	IOPS	Snapshot Storage
	Add New Row					

Elastic IP:

Number of Additional Elastic IPs:

Elastic IP Non-attached Time: Hours/Month ▾

Number of Elastic IP Remaps: Per Month ▾

Data Transfer:

Inter-Region Data Transfer Out: GB/Month ▾

Data Transfer Out: GB/Month ▾

Data Transfer In: GB/Month ▾

VPC Peering Data Transfer: GB/Month ▾

Intra-Region Data Transfer: GB/Month ▾

Public IP/Elastic IP Data Transfer: GB/Month ▾

Elastic Load Balancing:

Number of Elastic LBs:

Total Data Processed by all ELBs: GB/Month ▾

Amazon Services SIMPLE MONTHLY CALCULATOR

AWS pricing helps you reduce costs in multiple ways. [Learn more about AWS pricing](#)

FREE USAGE TIER: New Customers get free usage tier for first 12 months

Services

Estimate of your Monthly Bill (\$ 95.20)

Choose region: US-West-2 (Oregon) ▾

Inbound Data Transfer

 Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides resizable compute capacity in the cloud. It is designed to make web-scale computing easier.

Compute: Amazon EC2 Instances:

	Description	Instances	Usage	Type	Billing Option	Monthly Cost
⊖	ConfigServer2NodeInstance	1	24 Hours/Day ▾	Linux on t2.micro	On-Demand (No Cooldown)	\$ 9.52
⊖	SecondaryReplicaNode00NodeInstance	1	24 Hours/Day ▾	Linux on t2.micro	On-Demand (No Cooldown)	\$ 9.52
⊖	PrimaryReplicaNode10NodeInstance	1	24 Hours/Day ▾	Linux on t2.micro	On-Demand (No Cooldown)	\$ 9.52
⊖	NATInstance	1	24 Hours/Day ▾	Linux on t2.micro	On-Demand (No Cooldown)	\$ 9.52
⊖	PrimaryReplicaNode00NodeInstance	1	24 Hours/Day ▾	Linux on t2.micro	On-Demand (No Cooldown)	\$ 9.52
⊖	SecondaryReplicaNode10NodeInstance	1	24 Hours/Day ▾	Linux on t2.micro	On-Demand (No Cooldown)	\$ 9.52
⊖	SecondaryReplicaNode11NodeInstance	1	24 Hours/Day ▾	Linux on t2.micro	On-Demand (No Cooldown)	\$ 9.52
⊖	ConfigServer0NodeInstance	1	24 Hours/Day ▾	Linux on t2.micro	On-Demand (No Cooldown)	\$ 9.52
⊖	ConfigServer1NodeInstance	1	24 Hours/Day ▾	Linux on t2.micro	On-Demand (No Cooldown)	\$ 9.52
⊖	SecondaryReplicaNode01NodeInstance	1	24 Hours/Day ▾	Linux on t2.micro	On-Demand (No Cooldown)	\$ 9.52
⊕	Add New Row					

Storage: Amazon EBS Volumes:

	Description	Volumes	Volume Type	Storage	IOPS	Snapshot Storage	
	EBS volumes	10	Provisioned IOPS (SSD) ▾	8 GB	100	0 GB-month of Storage	▼
	Add New Row						

Elastic IP:

Number of Additional Elastic IPs:

Elastic IP Non-attached Time: Hours/Month ▾

Number of Elastic IP Remaps: Per Month ▾

Data Transfer:

Inter-Region Data Transfer Out: GB/Month ▾

Data Transfer Out: GB/Month ▾

Data Transfer In: GB/Month ▾

VPC Peering Data Transfer: GB/Month ▾

Intra-Region Data Transfer: GB/Month ▾

Public IP/Elastic IP Data Transfer: GB/Month ▾

Elastic Load Balancing:

Number of Elastic LBs:

Total Data Processed by all ELBs: GB/Month ▾

Compute: Amazon EC2 Instances:

	Description	Instances	Usage	Type	Billing Option	Monthly Cost
1	ConfigServer2NodeInstance	1	24 Hours/Day	Linux on t2.micro	On-Demand (No Contract)	\$ 9.52
2	SecondaryReplicaNode00NodeInst	1	24 Hours/Day	Linux on t2.micro		
3	PrimaryReplicaNode10NodeInsta	1	24 Hours/Day	Linux on t2.micro		
4	NATInstance	1	24 Hours/Day	Linux on t2.micro		
5	PrimaryReplicaNode00NodeInsta	1	24 Hours/Day	Linux on t2.micro		
6	SecondaryReplicaNode10NodeInst	1	24 Hours/Day	Linux on t2.micro		
7	SecondaryReplicaNode11NodeInst	1	24 Hours/Day	Linux on t2.micro		
8	ConfigServer0NodeInstance	1	24 Hours/Day	Linux on t2.micro		
9	ConfigServer1NodeInstance	1	24 Hours/Day	Linux on t2.micro		
10	SecondaryReplicaNode01NodeInst	1	24 Hours/Day	Linux on t2.micro	On-Demand (No Contract)	\$ 9.52

Select Billing Option

Instance Type: t2.micro

Operating System: Linux

Usage: 24 Hours/Day

Per Instance Prices & Projected Costs (all in USD)

Select	Name	Upfront Price	Effective Hourly Cost	Effective Monthly Cost	1 Year Cost	3 Year Cost
<input checked="" type="radio"/>	On-Demand (No Contract)	---	0.013	9.52	114.24	342.72
<input type="radio"/>	1 Yr No Upfront Reserved	0.00	0.009	6.57	78.84	236.52
<input type="radio"/>	1 Yr Partial Upfront Reserved	51.00	0.009	6.44	77.28	231.84
<input type="radio"/>	1 Yr All Upfront Reserved	75.00	0.009	6.25	75.00	225.00
<input type="radio"/>	3 Yr Partial Upfront Reserved	109.00	0.006	4.49	---	161.56
<input type="radio"/>	3 Yr All Upfront Reserved	151.00	0.006	4.20	---	151.00

[Close](#)

Services

Estimate of your Monthly Bill (\$ 192.41)

Estimate of Your Monthly Bill

Show First Month's Bill (include all one-time fees, if any)

Below you will see an estimate of your monthly bill. Expand each line item to see cost breakout of each service. To save this bill and input values, click on 'Save and Share' button. To remove the service from the estimate, jump back to the service and clear the specific service's form.

Save and Share

	Amazon EC2 Service (US-West-2)	\$ 191.60
	AWS Data Transfer In	\$ 0.00
	AWS Data Transfer Out	\$ 0.81
	AWS Support (Basic)	\$ 0.00
Total Monthly Payment:		\$ 192.41

Select Instance Type

Operating System

- Linux Red Hat Enterprise Linux SUSE Linux Enterprise Server
 Windows Windows and Web SQL Server Windows and Std. SQL Server
- EBS-Optimized

Select	Name	vCPU	Memory (GiB)	Instance Storage (GB)	I/O	EBS Opt.	On-Demand Hourly Cost	Reserved Effective Hourly Cost (Savings %) *
<input type="radio"/>	t1.micro	1	0.6	--	Very Low	--	\$0.020	\$0.008 (59%)
<input checked="" type="radio"/>	t2.micro	1	1.0	--	Low	--	\$0.013	\$0.006 (56%)
<input type="radio"/>	t2.small	1	2.0	--	Low	--	\$0.026	\$0.012 (56%)
<input type="radio"/>	t2.medium	2	4.0	--	Low	--	\$0.052	\$0.023 (56%)
<input type="radio"/>	t2.large	2	8.0	--	Low	--	\$0.104	\$0.046 (56%)
<input type="radio"/>	m4.large	2	8.0	--	Moderate	Yes	\$0.126	\$0.047 (63%)
<input type="radio"/>	m4.xlarge	4	16.0	--	High	Yes	\$0.252	\$0.094 (63%)
<input type="radio"/>	m4.2xlarge	8	32.0	--	High	Yes	\$0.504	\$0.188 (63%)
<input type="radio"/>	m4.4xlarge	16	64.0	--	High	Yes	\$1.008	\$0.376 (63%)
<input type="radio"/>	m4.10xlarge	40	160.0	--	Very High	Yes	\$2.520	\$0.940 (63%)
<input type="radio"/>	m3.medium	1	3.75	SSD 1 x 4	Moderate	--	\$0.067	\$0.026 (61%)
<input type="radio"/>	m3.large	2	7.5	SSD 1 x 32	Moderate	--	\$0.133	\$0.052 (61%)
<input type="radio"/>	m3.xlarge	4	15.0	SSD 2 x 40	High	Yes	\$0.266	\$0.105 (61%)
<input type="radio"/>	m3.2xlarge	8	30.0	SSD 2 x 80	High	Yes	\$0.532	\$0.209 (61%)
<input type="radio"/>	c4.large	2	3.7	--	Moderate	Yes	\$0.110	\$0.043 (61%)
<input type="radio"/>	c4.xlarge	4	7.5	--	Moderate	Yes	\$0.220	\$0.086 (61%)
<input type="radio"/>	c4.2xlarge	8	15.0	--	High	Yes	\$0.441	\$0.172 (61%)
<input type="radio"/>	c4.4xlarge	16	30.0	--	High	Yes	\$0.882	\$0.344 (61%)
<input type="radio"/>	c4.8xlarge	36	60.0	--	Very High	Yes	\$1.763	\$0.687 (61%)
<input type="radio"/>	c3.large	2	3.7	SSD 2 x 16	Moderate	--	\$0.105	\$0.039 (63%)
<input type="radio"/>	c3.xlarge	4	7.5	SSD 2 x 40	Moderate	Yes	\$0.210	\$0.079 (63%)
<input type="radio"/>	c3.2xlarge	8	15.0	SSD 2 x 80	High	Yes	\$0.420	\$0.157 (63%)
<input type="radio"/>	c3.4xlarge	16	30.0	SSD 2 x 160	High	Yes	\$0.840	\$0.315 (63%)
<input type="radio"/>	c3.8xlarge	32	60.0	SSD 2 x 320	Very High	--	\$1.680	\$0.628 (63%)
<input type="radio"/>	g2.2xlarge	8	15.0	SSD 1 x 60	High	Yes	\$0.650	\$0.282 (57%)
<input type="radio"/>	g2.8xlarge	32	60.0	SSD 2 x 120	Very High	--	\$2.600	\$1.128 (57%)
<input type="radio"/>	r3.large	2	15.2	SSD 1 x 32	Moderate	--	\$0.175	\$0.061 (65%)
<input type="radio"/>	r3.xlarge	4	30.5	SSD 1 x 80	Moderate	Yes	\$0.350	\$0.123 (65%)
<input type="radio"/>	r3.2xlarge	8	61.0	SSD 1 x 160	High	Yes	\$0.700	\$0.246 (65%)
<input type="radio"/>	r3.4xlarge	16	122.0	SSD 1 x 320	High	Yes	\$1.400	\$0.491 (65%)
<input type="radio"/>	r3.8xlarge	32	244.0	SSD 2 x 320	Very High	--	\$2.800	\$0.982 (65%)

Advanced Options

Show

* assumes 100% usage and Reserved Instance paid all upfront (more billing options available)

Close

[Create Stack](#)[Update Stack](#)[Delete Stack](#)Filter: Active ▾ By Name:

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](#)

Create StackUpdate StackDelete Stack

C G

Filter: Active ▾ By Name: Showing 1 stack

Stack Name	Created Time	Status	Description
AWS-MongoDB-Infrastructure	2015-09-05 19:31:59 UTC-0400	CREATE_IN_PROGRESS	(000F) Deploy MongoDB on a New VPC in AWS

OverviewOutputsResourcesEventsTemplateParametersTagsStack Policy

Select a stack



AWS

Services

Edit

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

Key Pairs

Network Interfaces

Launch Instance

Connect

Actions

Filter by tags and attributes or search by keyword

	Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS
	ConfigServer2	i-db74ff00	t2.micro	us-west-2c	green running	hourglass Initializing	None	yellow green
	PrimaryRepli...	i-2ecd54e8	t2.micro	us-west-2a	green running	hourglass Initializing	None	yellow green
	ConfigServer0	i-cbcd540d	t2.micro	us-west-2a	green running	hourglass Initializing	None	yellow green
	SecondaryR...	i-91367954	t2.micro	us-west-2b	green running	hourglass Initializing	None	yellow green
	PrimaryRepli...	i-2fcd54e9	t2.micro	us-west-2a	green running	hourglass Initializing	None	yellow green
	SecondaryR...	i-dc74ff07	t2.micro	us-west-2c	green running	hourglass Initializing	None	yellow green
	SecondaryR...	i-5175fe8a	t2.micro	us-west-2c	green running	hourglass Initializing	None	yellow green
	ConfigServer1	i-c8357a0d	t2.micro	us-west-2b	green running	hourglass Initializing	None	yellow green
	SecondaryR...	i-95367950	t2.micro	us-west-2b	green running	hourglass Initializing	None	yellow green
	NAT Instanc...	i-1aca53dc	t2.micro	us-west-2a	green running	hourglass Initializing	None	yellow green ec2-52-24-142-213.us...
		i-95755f62	t2.micro	us-west-2b	red stopped		None	yellow green
		i-92755f65	t2.micro	us-west-2b	red stopped		None	yellow green

...

EC2 Dashboard

Events

Tags

Reports

Limits

INSTANCES

Instances

Spot Requests

Reserved Instances

IMAGES

AMIs

Bundle Tasks

Launch Instance

Connect

Actions ▾

Filter by tags and attributes or search by keyword

<input type="checkbox"/>	Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	
<input type="checkbox"/>	NAT Instanc...	i-1aca53dc	t2.micro	us-west-2a	green circle running	 Initializing	None	
<input type="checkbox"/>		i-92755f65	t2.micro	us-west-2b	red circle stopped		None	
<input type="checkbox"/>		i-95755f62	t2.micro	us-west-2b	red circle stopped		None	

AWS Services Edit

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

Launch Instance

Connect

Actions

Filter by tags and attributes or search by keyword

<input type="checkbox"/>	Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	
<input type="checkbox"/>	ConfigServer2	i-db74ff00	t2.micro	us-west-2c	green running	green 2/2 checks passed	None	
<input type="checkbox"/>	PrimaryReplicaNode00	i-2ecd54e8	t2.micro	us-west-2a	green running	green 2/2 checks passed	None	
<input type="checkbox"/>	ConfigServer0	i-cbcd540d	t2.micro	us-west-2a	green running	green 2/2 checks passed	None	
<input type="checkbox"/>	SecondaryReplicaNod...	i-91367954	t2.micro	us-west-2b	green running	green 2/2 checks passed	None	
<input type="checkbox"/>	PrimaryReplicaNode10	i-2fcd54e9	t2.micro	us-west-2a	green running	green 2/2 checks passed	None	
<input type="checkbox"/>	SecondaryReplicaNod...	i-dc74ff07	t2.micro	us-west-2c	green running	green 2/2 checks passed	None	
<input type="checkbox"/>	SecondaryReplicaNod...	i-5175fe8a	t2.micro	us-west-2c	green running	green 2/2 checks passed	None	
<input type="checkbox"/>	ConfigServer1	i-c8357a0d	t2.micro	us-west-2b	green running	green 2/2 checks passed	None	
<input type="checkbox"/>	SecondaryReplicaNod...	i-95367950	t2.micro	us-west-2b	green running	green 2/2 checks passed	None	
<input type="checkbox"/>	NAT Instance (Public ...)	i-1aca53dc	t2.micro	us-west-2a	green running	green 2/2 checks passed	None	
<input type="checkbox"/>		i-95755f62	t2.micro	us-west-2b	red stopped		None	
<input type="checkbox"/>		i-92755f65	t2.micro	us-west-2b	red stopped		None	



AWS

Services

Edit

Create Stack

Update Stack

Delete Stack

Filter:

Active

By Name:

	Stack Name	Created Time	Status	Description
<input type="checkbox"/>	AWS-MongoDB-Infrastructure	2015-09-05 19:31:59 UTC-0400	CREATE_COMPLETE	(000F) Deploy MongoDB on a New VPC in AWS

Overview

Outputs

Resources

Events

Template

Parameters

Tags

Stack Policy

Select a stack

[Create Stack](#)[Update Stack](#)[Delete Stack](#)Filter: Active ▾ By Name:

	Stack Name	Created Time	Status	Description
<input checked="" type="checkbox"/>	AWS-MongoDB-Infrastructure	2015-09-05 19:31:59 UTC-0400	CREATE_COMPLETE	(000F) Deploy MongoDB on a New VPC in AWS

Overview	Outputs	Resources	Events	Template	Parameters	Tags	Stack Policy	
Logical ID	Physical ID						Type	Status
AssociateInterfaceNAT	eipassoc-f2711396						AWS::EC2::EIPAssociation	CREATE_COMPLETE
AttachGateway	AWS-M-Attac-1TRG10MTUU3AC						AWS::EC2::VPGatewayAttachment	CREATE_COMPLETE
ConfigServer0NodeIAMProfile	AWS-MongoDB-Infrastructure-ConfigServer0NodeIAMProfile-T5CJXAJL2AVC						AWS::IAM::InstanceProfile	CREATE_COMPLETE
ConfigServer0NodeIAMRole	AWS-MongoDB-Infrastructur-ConfigServer0NodeIAMRole-321782O2BKEM						AWS::IAM::Role	CREATE_COMPLETE
ConfigServer0NodeInstance	i-cbcd540d						AWS::EC2::Instance	CREATE_COMPLETE
ConfigServer0NodeInterface	eni-0d4a7244						AWS::EC2::NetworkInterface	CREATE_COMPLETE
ConfigServer0NodeSecurityGroup	sg-80c77ce4						AWS::EC2::SecurityGroup	CREATE_COMPLETE
ConfigServer0WaitForNodeInstall	arn:aws:cloudformation:us-west-2:518599548208:stack/AWS-MongoDB-Infrastructure/4ddc6cd0-5426-11e5-99b1-50fa5e6e7c0a/ConfigServer0WaitForNodeInstallHandle						AWS::CloudFormation::WaitCondition	CREATE_COMPLETE
ConfigServer0WaitForNodeInstallWaitHandle	https://cloudformation-waitcondition-us-west-2.s3-us-west-2.amazonaws.com/arn%3AAws%3Acloudformation%3Aus-west-2%3A518599548208%3Astack/AWS-MongoDB-Infrastructure/4ddc6cd0-5426-11e5-99b1-50fa5e6e7c0a/ConfigServer0WaitForNodeInstallWaitHandle?AWSAccessKeyId=AKIAIY5RM4DUXMU25KNA&Expires=1441582331&Signature=Ads%2BrYiX5aglGENUpCzmMxyQ0FY%3D						AWS::CloudFormation::WaitConditionHandle	CREATE_COMPLETE
ConfigServer1NodeIAMProfile	AWS-MongoDB-Infrastructure-ConfigServer1NodeIAMProfile-1OG5GRCDY7A18						AWS::IAM::InstanceProfile	CREATE_COMPLETE
ConfigServer1NodeIAMRole	AWS-MongoDB-Infrastructur-ConfigServer1NodeIAMRole-EE7RSKR7JA3Q						AWS::IAM::Role	CREATE_COMPLETE
ConfigServer1NodeInstance	i-c8357a0d						AWS::EC2::Instance	CREATE_COMPLETE
ConfigServer1NodeInterface	eni-f9189f9f						AWS::EC2::NetworkInterface	CREATE_COMPLETE
ConfigServer1NodeSecurityGroup	sg-88c77cec						AWS::EC2::SecurityGroup	CREATE_COMPLETE

[Create Stack](#)[Update Stack](#)[Delete Stack](#)Filter: Active ▾ By Name:

	Stack Name	Created Time	Status	Description
<input checked="" type="checkbox"/>	AWS-MongoDB-Infrastructure	2015-09-05 19:31:59 UTC-0400	CREATE_COMPLETE	(000F) Deploy MongoDB on a New VPC in AWS

[Overview](#) [Outputs](#) [Resources](#) [Events](#) [Template](#) [Parameters](#) [Tags](#) [Stack Policy](#)

Logical ID	Physical ID	Type	Status
AssociateInterfaceNAT	eipassoc-f2711396	AWS::EC2::EIPAssociation	CREATE_COMPLETE
AttachGateway	AWS-M-Attac-1TRG10MTUU3AC	AWS::EC2::VPGatewayAttachment	CREATE_COMPLETE
ConfigServer0NodeIAMProfile	AWS-MongoDB-Infrastructure-ConfigServer0NodeIAMProfile-T5CJXAJL2AVC	AWS::IAM::InstanceProfile	CREATE_COMPLETE
ConfigServer0NodeIAMRole	AWS-MongoDB-Infrastructur-ConfigServer0NodeIAMRole-321782O2BKEM	AWS::IAM::Role	CREATE_COMPLETE
ConfigServer0NodeInstance	i-cbcd540d	AWS::EC2::Instance	CREATE_COMPLETE
ConfigServer0NodeInterface	eni-0d4a7244	AWS::EC2::NetworkInterface	CREATE_COMPLETE
ConfigServer0NodeSecurityGroup	sg-80c77ce4	AWS::EC2::SecurityGroup	CREATE_COMPLETE
ConfigServer0WaitForNodeInstall	arn:aws:cloudformation:us-west-2:518599548208:stack/AWS-MongoDB-Infrastruktur/4ddc6cd0-5426-11e5-99b1-50fa5e6e7c0a/ConfigServer0WaitForNodeInstallWaitHandle	AWS::CloudFormation::WaitCondition	CREATE_COMPLETE
ConfigServer0WaitForNodeInstallWaitHandle	https://cloudformation-waitcondition-us-west-2.s3-us-west-2.amazonaws.com/arn%3AAws%3Acloudformation%3Aus-west-2%3A518599548208%3Astack/AWS-MongoDB-Infrastructure/4ddc6cd0-5426-11e5-99b1-50fa5e6e7c0a/ConfigServer0WaitForNodeInstallWaitHandle?AWSAccessKeyId=AKIAIY5RM4DUXMU25KNA&Expires=1441582331&Signature=Ads%2BrYiX5aglGENUpCzmMxyQ0FY%3D	AWS::CloudFormation::WaitConditionHandle	CREATE_COMPLETE
ConfigServer1NodeIAMProfile	AWS-MongoDB-Infrastructure-ConfigServer1NodeIAMProfile-1OG5GRCDY7A18	AWS::IAM::InstanceProfile	CREATE_COMPLETE
ConfigServer1NodeIAMRole	AWS-MongoDB-Infrastructur-ConfigServer1NodeIAMRole-EE7RSKR7JA3Q	AWS::IAM::Role	CREATE_COMPLETE
ConfigServer1NodeInstance	i-c8357a0d	AWS::EC2::Instance	CREATE_COMPLETE
ConfigServer1NodeInterface	eni-f9189f9f	AWS::EC2::NetworkInterface	CREATE_COMPLETE
ConfigServer1NodeSecurityGroup	sg-88c77cec	AWS::EC2::SecurityGroup	CREATE_COMPLETE
ConfigServer1WaitForNodeInstall	arn:aws:cloudformation:us-west-2:518599548208:stack/AWS-MongoDB-Infrastruktur/4ddc6cd0-5426-11e5-99b1-50fa5e6e7c0a/ConfigServer1WaitForNodeInstallWaitHandle	AWS::CloudFormation::WaitCondition	CREATE_COMPLETE

[Create Stack](#)[Update Stack](#)[Delete Stack](#)Filter: Active ▾ By Name:

Stack Name	Created Time	Status	Description
AWS-MongoDB-Infrastructure	2015-09-05 19:31:59 UTC-0400	CREATE_COMPLETE	(000F) Deploy MongoDB on a New VPC in AWS

Overview	Outputs	Resources	Events	Template	Parameters	Tags	Stack Policy
ConfigServer2WaitForNodeInstallWaitHandle		https://cloudformation-waitcondition-us-west-2.s3-us-west-2.amazonaws.com/arn%3Aaws%3Acloudformation%3Aus-west-2%3A518599548208%3Astack/AWS-MongoDB-Infrastructure/4ddc6cd0-5426-11e5-99b1-50fa5e6e7c0a/ConfigServer2WaitForNodeInstallWaitHandle?AWSAccessKeyId=AKIAIY5RM4DUXMU25KNA&Expires=1441582331&Signature=RvAacZbS5LCB0RaHYbceXaa%2FQvg%3D			AWS::CloudFormation::WaitConditionHandle		CREATE_COMPLETE
DMZRouteTable		rtb-2f84c24a			AWS::EC2::RouteTable		CREATE_COMPLETE
DMZSubnet		subnet-db3758ac			AWS::EC2::Subnet		CREATE_COMPLETE
InboundPublicNetworkAclEntry		AWS-M-Inbou-1PB30XRVR2S7D			AWS::EC2::NetworkAclEntry		CREATE_COMPLETE
InternetGateway		igw-f14ecd94			AWS::EC2::InternetGateway		CREATE_COMPLETE
NATEIP		52.24.142.213			AWS::EC2::EIP		CREATE_COMPLETE
NATInstance		i-1aca53dc			AWS::EC2::Instance		CREATE_COMPLETE
NATInterface		eni-774a723e			AWS::EC2::NetworkInterface		CREATE_COMPLETE
NATSecurityGroup		sg-89c77ced			AWS::EC2::SecurityGroup		CREATE_COMPLETE
OutBoundPublicNetworkAclEntry		AWS-M-OutBo-FZI41KZ8A7KM			AWS::EC2::NetworkAclEntry		CREATE_COMPLETE
PrimaryInboundPublicNetworkAclEntry		AWS-M-Prima-1NUIHEKFSZ9TB			AWS::EC2::NetworkAclEntry		CREATE_COMPLETE
PrimaryNodeRoute		AWS-M-Prima-10K6TGPVC299A			AWS::EC2::Route		CREATE_COMPLETE
PrimaryNodeRouteTable		rtb-2d84c248			AWS::EC2::RouteTable		CREATE_COMPLETE
PrimaryNodeSubnet		subnet-da3758ad			AWS::EC2::Subnet		CREATE_COMPLETE
PrimaryNodeSubnetRouteTableAssociation		rtbassoc-722a7517			AWS::EC2::SubnetRouteTableAssociation		CREATE_COMPLETE
PrimaryOutBoundPublicNetworkAclEntry		AWS-M-Prima-JZAL9PII7Y0O			AWS::EC2::NetworkAclEntry		CREATE_COMPLETE
PrimaryPublicNetworkAcl		acl-4d671828			AWS::EC2::NetworkAcl		CREATE_COMPLETE

[Create Stack](#)[Update Stack](#)[Delete Stack](#)Filter: Active ▾ By Name:

Stack Name	Created Time	Status	Description
<input checked="" type="checkbox"/> AWS-MongoDB-Infrastructure	2015-09-05 19:31:59 UTC-0400	CREATE_COMPLETE	(000F) Deploy MongoDB on a New VPC in AWS

Overview	Outputs	Resources	Events	Template	Parameters	Tags	Stack Policy
		PrimaryReplicaNode00NodeIAMProfile	AWS-MongoDB-Infrastructure-PrimaryReplicaNode00NodeIAMProfile-IR7SXZV21BOC		AWS::IAM::InstanceProfile		CREATE_COMPLETE
		PrimaryReplicaNode00NodeIAMRole	AWS-MongoDB-Infrastructure-PrimaryReplicaNode00Node-FM38ARZD1GUP		AWS::IAM::Role		CREATE_COMPLETE
		PrimaryReplicaNode00NodeInstanceGP2	i-2ecd54e8		AWS::EC2::Instance		CREATE_COMPLETE
		PrimaryReplicaNode00NodeInterface	eni-094a7240		AWS::EC2::NetworkInterface		CREATE_COMPLETE
		PrimaryReplicaNode00NodeSecurityGr...	sg-81c77ce5		AWS::EC2::SecurityGroup		CREATE_COMPLETE
		PrimaryReplicaNode00WaitForNodeInst...	arn:aws:cloudformation:us-west-2:518599548208:stack/AWS-MongoDB-Infrastructure/4ddc6cd0-5426-11e5-99b1-50fa5e6e7c0a/PrimaryReplicaNode00WaitForNodeInstallHandle		AWS::CloudFormation::WaitCondition		CREATE_COMPLETE
		PrimaryReplicaNode00WaitForNodeIns...	https://cloudformation-waitcondition-us-west-2.s3-us-west-2.amazonaws.com/arn%3AAws%3Acloudformation%3Aus-west-2%3A518599548208%3Astack/AWS-MongoDB-Infrastructure/4ddc6cd0-5426-11e5-99b1-50fa5e6e7c0a/PrimaryReplicaNode00WaitForNodeInstallHandle?AWSAccessKeyId=AKIAIY5RM4DUXMU25KNA&Expires=1441582331&Signature=fTIP%2BqkC5TVUFjup%2F3%2F3ie5q6k%3D		AWS::CloudFormation::WaitConditionHandle		CREATE_COMPLETE
		PrimaryReplicaNode10NodeIAMProfile	AWS-MongoDB-Infrastructure-PrimaryReplicaNode10NodeIAMProfile-1LXKS63MPCQFE		AWS::IAM::InstanceProfile		CREATE_COMPLETE
		PrimaryReplicaNode10NodeIAMRole	AWS-MongoDB-Infrastructure-PrimaryReplicaNode10Node-139UODA06Z8ON		AWS::IAM::Role		CREATE_COMPLETE
		PrimaryReplicaNode10NodeInstanceGP2	i-2fdc54e9		AWS::EC2::Instance		CREATE_COMPLETE
		PrimaryReplicaNode10NodeInterface	eni-0a4a7243		AWS::EC2::NetworkInterface		CREATE_COMPLETE
		PrimaryReplicaNode10NodeSecurityGr...	sg-8fc77ceb		AWS::EC2::SecurityGroup		CREATE_COMPLETE
		PrimaryReplicaNode10WaitForNodeIns...	arn:aws:cloudformation:us-west-2:518599548208:stack/AWS-MongoDB-Infrastructure/4ddc6cd0-5426-11e5-99b1-50fa5e6e7c0a/PrimaryReplicaNode10WaitForNodeInstallHandle		AWS::CloudFormation::WaitCondition		CREATE_COMPLETE
			https://cloudformation-waitcondition-us-west-2.s3-us-west-2.amazonaws.com/arn%3A				

[Create Stack](#) [Update Stack](#) [Delete Stack](#)

Filter: Active ▾ By Name: <input type="text"/>				
	Stack Name	Created Time	Status	Description
<input checked="" type="checkbox"/>	AWS-MongoDB-Infrastructure	2015-09-05 19:31:59 UTC-0400	CREATE_COMPLETE	(000F) Deploy MongoDB on a New VPC in AWS

	Overview	Outputs	Resources	Events	Template	Parameters	Tags	Stack Policy	
SecondaryReplicaNode10NodeInterface			eni-fd189f9b					AWS::EC2::NetworkInterface	CREATE_COMPLETE
SecondaryReplicaNode10NodeSecurity...			sg-8cc77ce8					AWS::EC2::SecurityGroup	CREATE_COMPLETE
SecondaryReplicaNode10WaitForNode...			arn:aws:cloudformation:us-west-2:518599548208:stack/AWS-MongoDB-Infrastructure/4ddc6cd0-5426-11e5-99b1-50fa5e6e7c0a/SecondaryReplicaNode10WaitForNodeInstallWaitHandle					AWS::CloudFormation::WaitCondition	CREATE_COMPLETE
SecondaryReplicaNode10WaitForNode...			https://cloudformation-waitcondition-us-west-2.s3-us-west-2.amazonaws.com/arn%3AAws%3Acloudformation%3Aus-west-2%3A518599548208%3Astack/AWS-MongoDB-Infrastructure/4ddc6cd0-5426-11e5-99b1-50fa5e6e7c0a/SecondaryReplicaNode10WaitForNodeInstallWaitHandle?AWSAccessKeyId=AKIAIY5RM4DUXMU25KNA&Expires=1441582331&Signature=j31P7yYtEoXYEX5D5R69KRzz1d4%3D					AWS::CloudFormation::WaitConditionHandle	CREATE_COMPLETE
SecondaryReplicaNode11NodeIAMProfile			AWS-MongoDB-Infrastructure-SecondaryReplicaNode11NodeIAMProfile-A508U878XN2U					AWS::IAM::InstanceProfile	CREATE_COMPLETE
SecondaryReplicaNode11NodeIAMRole			AWS-MongoDB-Infrastructure-SecondaryReplicaNode11No-1W4PE2LQE6YI1					AWS::IAM::Role	CREATE_COMPLETE
SecondaryReplicaNode11NodeInstanc...			i-5175fe8a					AWS::EC2::Instance	CREATE_COMPLETE
SecondaryReplicaNode11NodeInterface			eni-ee4cf5b5					AWS::EC2::NetworkInterface	CREATE_COMPLETE
SecondaryReplicaNode11NodeSecurity...			sg-8dc77ce9					AWS::EC2::SecurityGroup	CREATE_COMPLETE
SecondaryReplicaNode11WaitForNode...			arn:aws:cloudformation:us-west-2:518599548208:stack/AWS-MongoDB-Infrastructure/4ddc6cd0-5426-11e5-99b1-50fa5e6e7c0a/SecondaryReplicaNode11WaitForNodeInstallWaitHandle					AWS::CloudFormation::WaitCondition	CREATE_COMPLETE
SecondaryReplicaNode11WaitForNode...			https://cloudformation-waitcondition-us-west-2.s3-us-west-2.amazonaws.com/arn%3AAws%3Acloudformation%3Aus-west-2%3A518599548208%3Astack/AWS-MongoDB-Infrastructure/4ddc6cd0-5426-11e5-99b1-50fa5e6e7c0a/SecondaryReplicaNode11WaitForNodeInstallWaitHandle?AWSAccessKeyId=AKIAIY5RM4DUXMU25KNA&Expires=1441582330&Signature=d8TTs%2FTwraa745iEHneqBQvUcDw%3D					AWS::CloudFormation::WaitConditionHandle	CREATE_COMPLETE
VPC			vpc-d69fe7b3					AWS::EC2::VPC	CREATE_COMPLETE

[EC2 Dashboard](#)[Events](#)[Tags](#)[Reports](#)[Limits](#)[Launch Instance](#)[Connect](#)[Actions ▾](#) Filter by tags and attributes or search by keyword

	Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks
		i-95755f62	t2.micro	us-west-2b	stopped	
		i-92755f65	t2.micro	us-west-2b	stopped	
	ConfigServer0	i-cbcd540d	t2.micro	us-west-2a	running	2/2 checks passed
	ConfigServer1	i-c8357a0d	t2.micro	us-west-2b	running	2/2 checks passed
	ConfigServer2	i-db74ff00	t2.micro	us-west-2c	running	2/2 checks passed
	NAT Instance (Public Subnet)	i-1aca53dc	t2.micro	us-west-2a	running	2/2 checks passed
	PrimaryReplicaNode00	i-2ecd54e8	t2.micro	us-west-2a	running	2/2 checks passed
	PrimaryReplicaNode10	i-fcd54e9	t2.micro	us-west-2a	running	2/2 checks passed
	SecondaryReplicaNode00	i-95367950	t2.micro	us-west-2b	running	2/2 checks passed
	SecondaryReplicaNode01	i-dc74ff07	t2.micro	us-west-2c	running	2/2 checks passed
	SecondaryReplicaNode10	i-91367954	t2.micro	us-west-2b	running	2/2 checks passed
	SecondaryReplicaNode11	i-5175fe8a	t2.micro	us-west-2c	running	2/2 checks passed

Instance: i-db74ff00 (ConfigServer2) Private IP: 10.0.4.189

[Description](#) [Status Checks](#) [Monitoring](#) [Tags](#)

Instance ID: i-db74ff00

Instance state: running

Instance type: t2.micro

Private DNS: ip-10-0-4-189.us-west-2.compute.internal

Private IPs: 10.0.4.189

Secondary private IPs

VPC ID: vpc-d69fe7b3

Subnet ID: subnet-367bfa6f

Network interfaces: eth0

Source/dest. check: True

	NAT Instance (Public Subnet)	i-1aca53dc	t2.micro	us-west-2a	running	2/2 checks passed	None		ec2-52-24-142-213.us-west-2.compute.amazonaws.com	52.24.142.213	skondla_aws_...	disabled
	PrimaryReplicaNode00	i-2ecd54e8	t2.micro	us-west-2a	running	2/2 checks passed	None				skondla_aws_...	
	PrimaryReplicaNode10	i-2fcdf4e9	t2.micro	us-west-2a	running	2/2 checks passed	None				skondla_aws_...	
	SecondaryReplicaNode00	i-95367950	t2.micro	us-west-2b	running	2/2 checks passed	None				skondla_aws_...	
	SecondaryReplicaNode01	i-dc74ff07	t2.micro	us-west-2c	running	2/2 checks passed	None				skondla_aws_...	
	SecondaryReplicaNode10	i-91367954	t2.micro	us-west-2b	running	2/2 checks passed	None				skondla_aws_...	
	SecondaryReplicaNode11	i-5175fe8a	t2.micro	us-west-2c	running	2/2 checks passed	None				skondla_aws_...	

Instance: i-1aca53dc (NAT Instance (Public Subnet)) Elastic IP: 52.24.142.213



Description Status Checks Monitoring Tags

Instance ID	i-1aca53dc	Public DNS	ec2-52-24-142-213.us-west-2.compute.amazonaws.com
Instance state	running	Public IP	52.24.142.213
Instance type	t2.micro	Elastic IP	52.24.142.213
Private DNS	ip-10-0-1-144.us-west-2.compute.internal	Availability zone	us-west-2a
Private IPs	10.0.1.144	Security groups	AWS-MongoDB-Infrastructure-NATSecurityGroup-41LQD8H4UUPC. view rules
Secondary private IPs		Scheduled events	No scheduled events
VPC ID	vpc-d69fe7b3	AMI ID	amzn-ami-vpc-nat-hvm-2015.03.0.x86_64-gp2 (ami-69ae8259)
Subnet ID	subnet-db3758ac	Platform	-
Network interfaces	eth0	IAM role	-

```
[2015-09-05 20:04.34] /drives/c/Users/Sudhe_000/Downloads  
[Sudhe_000.rose-win8] ➤ ssh -i skondla_aws_keys_oregonpem.pem ec2-user@  
X11 forwarding request failed on channel 0  
Last login: Sun Sep  6 00:03:39 2015 from c-50-180-143-247.hsd1.ga.comcast.net
```

```
__|__|_)  
__|(|_| /  Amazon Linux AMI  
__|\_\_|_||
```

```
https://aws.amazon.com/amazon-linux-ami/2015.03-release-notes/  
35 package(s) needed for security, out of 70 available  
Run "sudo yum update" to apply all updates.  
[ec2-user@ip-10-0-1-144 ~]$
```

```
[2015-09-05 20:04.34] /drives/c/Users/Sudhe_000/Downloads
[Sudhe_000.rose-win8] > ssh -i skondla_aws_keys_oregonpem.pem ec2-user@52.24.142.213
X11 forwarding request failed on channel 0
Last login: Sun Sep  6 00:03:39 2015 from c-50-180-143-247.hsd1.ga.comcast.net

  _|_(_|_)_
  ||(   /   Amazon Linux AMI
  __| \__|__|_

https://aws.amazon.com/amazon-linux-ami/2015.03-release-notes/
35 package(s) needed for security, out of 70 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-10-0-1-144 ~]$
```

Connect To Your Instance



I would like to connect with

- A standalone SSH client
- A Java SSH Client directly from my browser (Java required)

To access your instance:

1. Open an SSH client. ([find out how to connect using PuTTY](#))
2. Locate your private key file (skondla_aws_keys_oregon.pem.pem). The wizard automatically detects the key you used to launch the instance.
3. Your key must not be publicly viewable for SSH to work. Use [this command](#) if needed:

```
chmod 400 skondla_aws_keys_oregon.pem.pem
```

4. Connect to your instance using its Elastic IP:

52.24.142.213

Example:

```
ssh -i "skondla_aws_keys_oregon.pem.pem" ec2-user@52.24.142.213
```

Please note that in most cases the username above will be correct, however please ensure that you read your AMI usage instructions to ensure that the AMI owner has not changed the default AMI username.

If you need any assistance connecting to your instance, please see our [connection documentation](#).

Close

Connect To Your Instance



I would like to connect with

- A standalone SSH client
- A Java SSH Client directly from my browser (Java required)

Enter the required information in the fields below to connect to your instance. AWS automatically detects the key pair name, and Elastic IP for your instance. You need to enter the location and name of the .pem file containing your private key.

Elastic IP 52.24.142.213

User name ec2-user

Key name skondla_aws_keys_oregon.pem.pem

Private key path E:\AWS\keys\skondla_aws_keys_

Save key location Store in browser cache



NPAPI deprecation on Chrome

As of April 2015, Chrome disables plugins using NPAPI, which may prevent the in-browser Java SSH client from loading. To continue to SSH to your EC2 instances using Chrome, go to <chrome://flags/#enable-npapi> in a new tab, enable the NPAPI flag and restart your browser. Alternatively, use Firefox, Internet Explorer 9 or higher, or Safari.

[Learn more about NPAPI deprecation on Chrome](#).

Launch SSH Client

Close

```
##### From NAT Server ##### Connect to MongoDB Primary and Secondary Instances
cd /drives/c/Users/Sudhe_000/Downloads
ssh -i skondla_aws_keys_oregonpem.pem ec2-user@52.24.142.213 #NAT Server
ssh -i "skondla_aws_keys_oregonpem.pem" ec2-user@10.0.2.161 #Primary Replica Server 0 OR
ssh -i "skondla_aws_keys_oregonpem.pem" ec2-user@10.0.3.138 # Secondary Node 00
ssh -i "skondla_aws_keys_oregonpem.pem" ec2-user@10.0.2.236 # Config Server 0.
ssh -i "skondla_aws_keys_oregonpem.pem" ec2-user@10.0.2.84 # Primary Replica Server 10
```

```
[2015-09-05 22:27.10] /drives/c/Users/Sudhe_000/Downloads  
[Sudhe_000.rose-win8] > scp -i skondla_aws_keys_oregonpem.pem skondla_aws_keys_oregonpem.pem ec2-user@52.24.142.213:/home/ec2-user/skondla_aws_keys_oregonpem.pem  
skondla_aws_keys_oregonpem.pem 100% 1696      1.7KB/s   00:00
```

Connect To Your Instance



I would like to connect with

- A standalone SSH client
- A Java SSH Client directly from my browser (Java required)

To access your instance:

1. Open an SSH client. (find out how to [connect using PuTTY](#))
2. Locate your private key file (`skondla_aws_keys_oregon.pem.pem`). The wizard automatically detects the key you used to launch the instance.
3. Your key must not be publicly viewable for SSH to work. Use this command if needed:

```
chmod 400 skondla_aws_keys_oregon.pem.pem
```

4. Connect to your instance using its Private IP:

10.0.2.161

Example:

```
ssh -i "skondla_aws_keys_oregon.pem.pem" ec2-user@10.0.2.161
```

Please note that in most cases the username above will be correct, however please ensure that you read your AMI usage instructions to ensure that the AMI owner has not changed the default AMI username.

If you need any assistance connecting to your instance, please see our [connection documentation](#).

Close

```
[ec2-user@ip-10-0-1-144 ~]$ ssh -i "skondla_aws_keys_oregonpem.pem" ec2-user@10.0.2.161
The authenticity of host '10.0.2.161 (10.0.2.161)' can't be established.
ECDSA key fingerprint is d1:26:a4:a8:80:b9:66:ae:ea:3c:e7:56:af:b9:c6:b3.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.0.2.161' (ECDSA) to the list of known hosts.
@@@@@@@@@@@@@@@@@@@WARNING: UNPROTECTED PRIVATE KEY FILE! @
@@@@@@@@@@@@@@@Permissions 0670 for 'skondla_aws_keys_oregonpem.pem' are too open.
It is required that your private key files are NOT accessible by others.
This private key will be ignored.
bad permissions: ignore key: skondla_aws_keys_oregonpem.pem
Permission denied (publickey).
[ec2-user@ip-10-0-1-144 ~]$ ls -ltr
total 4
-rw-rw-r-- 1 ec2-user ec2-user 0 Sep 6 02:25 a.txt
-rw-rwx--- 1 ec2-user ec2-user 1696 Sep 6 02:27 skondla_aws_keys_oregonpem.pem
[ec2-user@ip-10-0-1-144 ~]$ chmod 400 skondla_aws_keys_oregonpem.pem
[ec2-user@ip-10-0-1-144 ~]$ ssh -i "skondla_aws_keys_oregonpem.pem" ec2-user@10.0.2.161
```

__| (_ | _)
__| __|__|
Amazon Linux AMI

<https://aws.amazon.com/amazon-linux-ami/2015.03-release-notes/>
[ec2-user@ip-10-0-2-161 ~]\$ █

```
[ec2-user@ip-10-0-2-161 ~]$ mongo
MongoDB shell version: 3.0.6
connecting to: test
Welcome to the MongoDB shell.
For interactive help, type "help".
For more comprehensive documentation, see
      http://docs.mongodb.org/
Questions? Try the support group
      http://groups.google.com/group/mongodb-user
mongos> █
```

```
mongos> sh.help()
sh.addShard( host )
sh.enableSharding(dbname)
sh.shardCollection(fullName,key,unique)
sh.splitFind(fullName,find)
sh.splitAt(fullName,middle)
sh.moveChunk(fullName,find,to)
sh.setBalancerState( <bool on or not> )
sh.getBalancerState()
sh.isBalancerRunning()
sh.disableBalancing(coll)
sh.enableBalancing(coll)
sh.addShardTag(shard,tag)
sh.removeShardTag(shard,tag)
sh.addTagRange(fullName,min,max,tag)
sh.removeTagRange(fullName,min,max,tag)
sh.status()
mongos> sh.status()
--- Sharding Status ---
sharding version: {
    "_id" : 1,
    "minCompatibleVersion" : 5,
    "currentVersion" : 6,
    "clusterId" : ObjectId("55eb7db7466e357a0952e616")
}
shards:
{   "_id" : "s0-rs0",   "host" : "s0-rs0/10.0.2.161:27018,10.0.3.138:27018,10.0.4.56:27018" }
{   "_id" : "s0-rs1",   "host" : "s0-rs1/10.0.2.161:27019,10.0.3.138:27019,10.0.4.56:27019" }
{   "_id" : "s1-rs0",   "host" : "s1-rs0/10.0.2.84:27018,10.0.3.22:27018,10.0.4.19:27018" }
{   "_id" : "s1-rs1",   "host" : "s1-rs1/10.0.2.84:27019,10.0.3.22:27019,10.0.4.19:27019" }
balancer:
    Currently enabled: yes
    Currently running: no
    Failed balancer rounds in last 5 attempts: 0
    Migration Results for the last 24 hours:
        No recent migrations
databases:
{   "_id" : "admin",   "partitioned" : false,   "primary" : "config" }

mongos> █
```

```
mongos> sh.getBalancerState()
true
mongos> sh.isBalanceRunning()
2015-09-06T03:07:47.103+0000 E QUERY    TypeError: Object function () { return "try sh.help();" } has no method 'isBalanceRunning'
    at (shell):1:4
mongos> sh.isBalancerRunning()
false
mongos> sh.getBalancerState()
true
mongos> exit
bye
[ec2-user@ip-10-0-2-161 ~]$ netstat -an | grep 27017
tcp      0      0 0.0.0.0:27017          0.0.0.0:*                  LISTEN
tcp      0      0 127.0.0.1:52386        127.0.0.1:27017          TIME_WAIT
unix  2      [ ACC ]     STREAM      LISTENING      15235  /tmp/mongodb-27017.sock
[ec2-user@ip-10-0-2-161 ~]$
```

```
[ec2-user@ip-10-0-2-161 ~]$ netstat -an | grep 27017
tcp        0      0 0.0.0.0:27017          0.0.0.0:*
                                         LISTEN
tcp        0      0 127.0.0.1:52386       127.0.0.1:27017      TIME_WAIT
unix  2      [ ACC ]     STREAM      LISTENING      15235  /tmp/mongodb-27017.sock
[ec2-user@ip-10-0-2-161 ~]$ netstat -an | grep 27018
tcp        0      0 0.0.0.0:27018          0.0.0.0:*
                                         LISTEN
tcp        0      0 10.0.2.161:27018       10.0.2.84:54824    ESTABLISHED
tcp        0      0 10.0.2.161:35286       10.0.2.161:27018      ESTABLISHED
tcp        0      0 10.0.2.161:39028       10.0.4.56:27018    TIME_WAIT
tcp        0      0 10.0.2.161:53090       10.0.3.22:27018    ESTABLISHED
tcp        0      0 10.0.2.161:27018       10.0.2.161:35286    ESTABLISHED
tcp        0      0 10.0.2.161:36144       10.0.4.56:27018    ESTABLISHED
tcp        0      0 10.0.2.161:35284       10.0.2.161:27018      ESTABLISHED
tcp        0      0 10.0.2.161:49885       10.0.4.19:27018    ESTABLISHED
tcp        0      0 10.0.2.161:59992       10.0.2.84:27018    ESTABLISHED
tcp        0      0 10.0.2.161:27018       10.0.2.84:54825    ESTABLISHED
tcp        0      0 10.0.2.161:46503       10.0.3.138:27018   TIME_WAIT
tcp        0      0 10.0.2.161:27018       10.0.2.161:35284    ESTABLISHED
tcp        0      0 10.0.2.161:46499       10.0.3.138:27018   TIME_WAIT
tcp        0      0 10.0.2.161:43616       10.0.3.138:27018      ESTABLISHED
tcp        0      0 10.0.2.161:27018       10.0.4.56:57975    ESTABLISHED
tcp        0      0 10.0.2.161:39032       10.0.4.56:27018    TIME_WAIT
tcp        0      0 10.0.2.161:46508       10.0.3.138:27018      ESTABLISHED
tcp        0      0 10.0.2.161:39036       10.0.4.56:27018    ESTABLISHED
tcp        0      0 10.0.2.161:27018       10.0.3.138:40345    ESTABLISHED
tcp        0      0 10.0.2.161:59993       10.0.2.84:27018    ESTABLISHED
unix  2      [ ACC ]     STREAM      LISTENING      14719  /tmp/mongodb-27018.sock
[ec2-user@ip-10-0-2-161 ~]$ netstat -an | grep 27019
tcp        0      0 0.0.0.0:27019          0.0.0.0:*
                                         LISTEN
tcp        0      0 10.0.2.161:27019       10.0.2.161:50770    ESTABLISHED
tcp        0      0 10.0.2.161:53828       10.0.4.56:27019    ESTABLISHED
tcp        0      0 10.0.2.161:27019       10.0.2.84:53203    ESTABLISHED
tcp        0      0 10.0.2.161:56719       10.0.4.56:27019    ESTABLISHED
tcp        0      0 10.0.2.161:56715       10.0.4.56:27019   TIME_WAIT
tcp        0      0 10.0.2.161:50770       10.0.2.161:27019    ESTABLISHED
tcp        0      0 10.0.2.161:27019       10.0.3.138:33912    ESTABLISHED
tcp        0      0 10.0.2.161:56711       10.0.4.56:27019   TIME_WAIT
tcp        0      0 10.0.2.161:52130       10.0.3.138:27019   TIME_WAIT
tcp        0      0 10.0.2.161:47224       10.0.2.84:27019    ESTABLISHED
tcp        0      0 10.0.2.161:52133       10.0.3.138:27019    ESTABLISHED
tcp        0      0 10.0.2.161:39830       10.0.4.19:27019    ESTABLISHED
tcp        0      0 10.0.2.161:48498       10.0.3.22:27019    ESTABLISHED
tcp        0      0 10.0.2.161:52126       10.0.3.138:27019   TIME_WAIT
tcp        0      0 10.0.2.161:27019       10.0.4.56:44606    ESTABLISHED
tcp        0      0 10.0.2.161:27019       10.0.2.84:53204    ESTABLISHED
tcp        0      0 10.0.2.161:49244       10.0.3.138:27019    ESTABLISHED
tcp        0      0 10.0.2.161:27019       10.0.2.161:50769    ESTABLISHED
tcp        0      0 10.0.2.161:50769       10.0.2.161:27019    ESTABLISHED
tcp        0      0 10.0.2.161:47225       10.0.2.84:27019    ESTABLISHED
unix  2      [ ACC ]     STREAM      LISTENING      14890  /tmp/mongodb-27019.sock
```

```
[ec2-user@ip-10-0-3-138 ~]$ netstat -an | grep 27017
[ec2-user@ip-10-0-3-138 ~]$ netstat -an | grep 27019
tcp      0      0 0.0.0.0:27019          0.0.0.0:*                  LISTEN
tcp      0      0 10.0.3.138:27019        10.0.4.56:42046        ESTABLISHED
tcp      0      0 10.0.3.138:33896        10.0.2.161:27019        TIME_WAIT
tcp      0      0 10.0.3.138:27019        10.0.2.84:55988        ESTABLISHED
tcp      0      0 10.0.3.138:57676        10.0.4.56:27019        ESTABLISHED
tcp      0      0 10.0.3.138:57668        10.0.4.56:27019        TIME_WAIT
tcp      0      0 10.0.3.138:57672        10.0.4.56:27019        TIME_WAIT
tcp      0      0 10.0.3.138:33900        10.0.2.161:27019        TIME_WAIT
tcp      0      0 10.0.3.138:27019        10.0.2.161:52126        ESTABLISHED
tcp      0      0 10.0.3.138:33904        10.0.2.161:27019        ESTABLISHED
tcp      0      0 10.0.3.138:27019        10.0.2.161:49244        ESTABLISHED
unix  2      [ ACC ]     STREAM      LISTENING      14745  /tmp/mongodb-27019.sock
[ec2-user@ip-10-0-3-138 ~]$ netstat -an | grep 27018
tcp      0      0 0.0.0.0:27018          0.0.0.0:*                  LISTEN
tcp      0      0 10.0.3.138:27018        10.0.4.56:38171        ESTABLISHED
tcp      0      0 10.0.3.138:27018        10.0.2.161:46503        ESTABLISHED
tcp      0      0 10.0.3.138:40337        10.0.2.161:27018        TIME_WAIT
tcp      0      0 10.0.3.138:40341        10.0.2.161:27018        ESTABLISHED
tcp      0      0 10.0.3.138:51867        10.0.4.56:27018        TIME_WAIT
tcp      0      0 10.0.3.138:51871        10.0.4.56:27018        ESTABLISHED
tcp      0      0 10.0.3.138:51863        10.0.4.56:27018        TIME_WAIT
tcp      0      0 10.0.3.138:40333        10.0.2.161:27018        TIME_WAIT
tcp      0      0 10.0.3.138:27018        10.0.2.161:43616        ESTABLISHED
tcp      0      0 10.0.3.138:27018        10.0.2.84:34218        ESTABLISHED
unix  2      [ ACC ]     STREAM      LISTENING      14572  /tmp/mongodb-27018.sock
```

```
[ec2-user@ip-10-0-2-236 ~]$ ps -ef|grep -i mongo
mongod    7636      1  0 Sep05 ?          00:00:32 /usr/bin/mongod -f /etc/mongod.conf
ec2-user  8630  8600  0 03:18 pts/0    00:00:00 grep -i mongo
[ec2-user@ip-10-0-2-236 ~]$ cat /etc/mongod.conf | grep -v "#" | sed '/^$/d'
logpath=/data/mongod.log
logappend=true
fork=true
port=27030
dbpath=/data
configsvr=true
pidfilepath=/var/run/mongodb/mongod.pid
[ec2-user@ip-10-0-2-236 ~]$ tail -f /data/mongod.log
2015-09-06T03:18:23.664+0000 I STORAGE [conn12] CMD fsync: sync:1 lock:0
2015-09-06T03:18:30.741+0000 I STORAGE [conn7]  CMD fsync: sync:1 lock:0
2015-09-06T03:18:34.234+0000 I STORAGE [conn12] CMD fsync: sync:1 lock:0
2015-09-06T03:18:41.311+0000 I STORAGE [conn7]  CMD fsync: sync:1 lock:0
2015-09-06T03:18:44.868+0000 I STORAGE [conn12] CMD fsync: sync:1 lock:0
2015-09-06T03:18:51.945+0000 I STORAGE [conn7]  CMD fsync: sync:1 lock:0
2015-09-06T03:18:55.437+0000 I STORAGE [conn12] CMD fsync: sync:1 lock:0
2015-09-06T03:19:02.547+0000 I STORAGE [conn7]  CMD fsync: sync:1 lock:0
2015-09-06T03:19:05.970+0000 I STORAGE [conn12] CMD fsync: sync:1 lock:0
2015-09-06T03:19:13.180+0000 I STORAGE [conn7]  CMD fsync: sync:1 lock:0
2015-09-06T03:19:16.669+0000 I STORAGE [conn12] CMD fsync: sync:1 lock:0
```

Connect To Your Instance



I would like to connect with

- A standalone SSH client
- A Java SSH Client directly from my browser (Java required)

To access your instance:

1. Open an SSH client. (find out how to [connect using PuTTY](#))
2. Locate your private key file (`skondla_aws_keys_oregon.pem.pem`). The wizard automatically detects the key you used to launch the instance.
3. Your key must not be publicly viewable for SSH to work. Use this command if needed:

```
chmod 400 skondla_aws_keys_oregon.pem.pem
```

4. Connect to your instance using its Private IP:

`10.0.2.236`

Example:

```
ssh -i "skondla_aws_keys_oregon.pem.pem" ec2-user@10.0.2.236
```

Please note that in most cases the username above will be correct, however please ensure that you read your AMI usage instructions to ensure that the AMI owner has not changed the default AMI username.

If you need any assistance connecting to your instance, please see our [connection documentation](#).

Close

Connect To Your Instance



I would like to connect with

- A standalone SSH client
- A Java SSH Client directly from my browser (Java required)

To access your instance:

1. Open an SSH client. (find out how to connect using [PuTTY](#))
2. Locate your private key file (`skondla_aws_keys_oregon.pem.pem`). The wizard automatically detects the key you used to launch the instance.
3. Your key must not be publicly viewable for SSH to work. Use this command if needed:

```
chmod 400 skondla_aws_keys_oregon.pem.pem
```

4. Connect to your instance using its Private IP:

`10.0.2.84`

Example:

```
ssh -i "skondla_aws_keys_oregon.pem.pem" ec2-user@10.0.2.84
```

Please note that in most cases the username above will be correct, however please ensure that you read your AMI usage instructions to ensure that the AMI owner has not changed the default AMI username.

If you need any assistance connecting to your instance, please see our [connection documentation](#).

Close

```
[ec2-user@ip-10-0-2-161 ~]$ ps -ef|grep -i mongo
ec2-user  5184 13545  0 03:24 pts/0    00:00:00 grep -i mongo
mongod    8074      1  0 Sep05 ?        00:00:30 /usr/bin/mongod -f /etc/mongod0.conf
mongod    8209      1  0 Sep05 ?        00:00:29 /usr/bin/mongod -f /etc/mongod1.conf
mongod    8547      1  0 Sep05 ?        00:00:14 /usr/bin/mongos -f /etc/mongos.conf
[ec2-user@ip-10-0-2-161 ~]$ cat /etc/mongod0.conf | grep -v "#" | sed '/^$/d'
logpath=/log/mongod0.log
logappend=true
fork=true
port=27018
dbpath=/data/s0-rs0
pidfilepath=/var/run/mongodb/mongod0.pid
replSet=s0-rs0
[ec2-user@ip-10-0-2-161 ~]$ cat /etc/mongod1.conf | grep -v "#" | sed '/^$/d'
logpath=/log/mongod1.log
logappend=true
fork=true
port=27019
dbpath=/data/s0-rs1
pidfilepath=/var/run/mongodb/mongod1.pid
replSet=s0-rs1
[ec2-user@ip-10-0-2-161 ~]$ cat /etc/mongos.conf | grep -v "#" | sed '/^$/d'
logpath=/log/mongos.log
logappend=true
fork=true
port=27017
pidfilepath=/var/run/mongodb/mongos.pid
configdb=10.0.4.189:27030,10.0.3.241:27030,10.0.2.236:27030
[ec2-user@ip-10-0-2-161 ~]$
```

```
[ec2-user@ip-10-0-2-236 ~]$ ps -ef|grep -i mongo
mongod    7636      1  0 Sep05 ?          00:00:35 /usr/bin/mongod -f /etc/mongod.conf
ec2-user  8668  8600  0 03:35 pts/0    00:00:00 grep -i mongo
[ec2-user@ip-10-0-2-236 ~]$ cat /etc/mongod.conf | grep -v "#" | sed '/^$/d'
logpath=/data/mongod.log
logappend=true
fork=true
port=27030
dbpath=/data
configsvr=true
pidfilepath=/var/run/mongodb/mongod.pid
[ec2-user@ip-10-0-2-236 ~]$ netstat -an | grep 27030
tcp        0      0 0.0.0.0:27030          0.0.0.0:*                LISTEN
tcp        0      0 10.0.2.236:27030        10.0.2.84:50683          ESTABLISHED
tcp        0      0 10.0.2.236:27030        10.0.2.161:57316          ESTABLISHED
tcp        0      0 10.0.2.236:27030        10.0.2.84:50680          ESTABLISHED
tcp        0      0 10.0.2.236:27030        10.0.2.84:50719          ESTABLISHED
tcp        0      0 10.0.2.236:27030        10.0.2.84:48058          ESTABLISHED
tcp        0      0 10.0.2.236:27030        10.0.2.161:57097          ESTABLISHED
tcp        0      0 10.0.2.236:27030        10.0.2.161:54405          ESTABLISHED
tcp        0      0 10.0.2.236:27030        10.0.2.161:54427          ESTABLISHED
tcp        0      0 10.0.2.236:27030        10.0.2.84:50676          ESTABLISHED
tcp        0      0 10.0.2.236:27030        10.0.2.84:48073          ESTABLISHED
tcp        0      0 10.0.2.236:27030        10.0.2.84:48064          ESTABLISHED
tcp        0      0 10.0.2.236:27030        10.0.2.161:57104          ESTABLISHED
tcp        0      0 10.0.2.236:27030        10.0.2.161:54411          ESTABLISHED
tcp        0      0 10.0.2.236:27030        10.0.2.161:57101          ESTABLISHED
tcp        0      0 10.0.2.236:27030        10.0.2.161:57203          ESTABLISHED
unix  2      [ ACC ]     STREAM     LISTENING     13824  /tmp/mongodb-27030.sock
[ec2-user@ip-10-0-2-236 ~]$ ls -l /etc/mon*
-rw-r--r-- 1 root root 1507 Sep  5 23:40 /etc/mongod.conf
[ec2-user@ip-10-0-2-236 ~]$
```

```
[root@ip-10-0-2-161 ec2-user]# find /etc -name "mongo*" -print
/etc/yum.repos.d/mongodb-org-3.0.repo
/etc/mongod.conf
/etc/sysconfig/mongod1
/etc/sysconfig/mongod0
/etc/sysconfig/mongod
/etc/mongod1.conf
/etc/mongos.conf
/etc/rc.d/init.d/mongod1
/etc/rc.d/init.d/mongod0
/etc/rc.d/init.d/mongos
/etc/rc.d/init.d/mongod
/etc/mongod0.conf
[root@ip-10-0-2-161 ec2-user]#
```

```
[root@ip-10-0-3-138 ec2-user]# find /etc -name "mongo*" -print
/etc/yum.repos.d/mongodb-org-3.0.repo
/etc/mongod.conf
/etc/sysconfig/mongod1
/etc/sysconfig/mongod0
/etc/sysconfig/mongod
/etc/mongod1.conf
/etc/rc.d/init.d/mongod1
/etc/rc.d/init.d/mongod0
/etc/rc.d/init.d/mongod
/etc/mongod0.conf
[root@ip-10-0-3-138 ec2-user]#
```

[Launch Instance](#)[Connect](#)[Actions ▾](#) Filter by tags and attributes or search by keyword

<input type="checkbox"/>	Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	⋮
<input type="checkbox"/>	ConfigServer0	i-cbcd540d	t2.micro	us-west-2a	● stopped		
<input type="checkbox"/>	ConfigServer1	i-c8357a0d	t2.micro	us-west-2b	● stopped		
<input type="checkbox"/>	ConfigServer2	i-db74ff00	t2.micro	us-west-2c	● stopped		
<input type="checkbox"/>	NAT Instance (Public Subnet)	i-1aca53dc	t2.micro	us-west-2a	● stopped		
<input type="checkbox"/>	PrimaryReplicaNode00	i-2ecd54e8	t2.micro	us-west-2a	● stopped		
<input type="checkbox"/>	PrimaryReplicaNode10	i-2fcd54e9	t2.micro	us-west-2a	● stopped		
<input type="checkbox"/>	SecondaryReplicaNode00	i-95367950	t2.micro	us-west-2b	● stopped		
<input type="checkbox"/>	SecondaryReplicaNode01	i-dc74ff07	t2.micro	us-west-2c	● stopped		
<input type="checkbox"/>	SecondaryReplicaNode10	i-91367954	t2.micro	us-west-2b	● stopped		
<input type="checkbox"/>	SecondaryReplicaNode11	i-5175fe8a	t2.micro	us-west-2c	● stopped		

```
[ec2-user@ip-10-0-2-236 ~]$ sudo su
[root@ip-10-0-2-236 ec2-user]# find /etc -name "mongo*" -print
/etc/yum.repos.d/mongodb-org-3.0.repo
/etc/mongod.conf
/etc/sysconfig/mongod
/etc/rc.d/init.d/mongod
[root@ip-10-0-2-236 ec2-user]# █
```

[Create Volume](#)[Actions ▾](#) Filter by tags and attributes or search by keyword

	Name	Volume ID	Size	Volume Type	IOPS	Snapshot	Created	Availability Zone	State	Alarm Status	Attachment Information
	vol-9e616250	40 GiB	gp2	120 / 3000			September 5, 2015 ...	us-west-2a	● in-use	None	 i-cbcd540d (ConfigSe...
	vol-2fe631db	8 GiB	gp2	24 / 3000			September 5, 2015 ...	us-west-2b	● in-use	None	 i-91367954 (Secondar...
	vol-46616288	8 GiB	gp2	24 / 3000		snap-bfb086e1	September 5, 2015 ...	us-west-2a	● in-use	None	 i-cbcd540d (ConfigSe...
	vol-ba1b225d	8 GiB	gp2	24 / 3000			September 5, 2015 ...	us-west-2c	● in-use	None	 i-dc74ff07 (Secondar...
	vol-941b2273	8 GiB	gp2	24 / 3000		snap-bfb086e1	September 5, 2015 ...	us-west-2c	● in-use	None	 i-dc74ff07 (Secondar...
	vol-50e631a4	25 GiB	io1	250			September 5, 2015 ...	us-west-2b	● in-use	None	 i-91367954 (Secondar...
	vol-41e631b5	8 GiB	gp2	24 / 3000		snap-bfb086e1	September 5, 2015 ...	us-west-2b	● in-use	None	 i-91367954 (Secondar...
	vol-321821d5	8 GiB	gp2	24 / 3000		snap-bfb086e1	September 5, 2015 ...	us-west-2c	● in-use	None	 i-db74ff00 (ConfigSer...
	vol-0b1821ec	25 GiB	io1	250			September 5, 2015 ...	us-west-2c	● in-use	None	 i-dc74ff07 (Secondar...
	vol-951b2272	25 GiB	io1	200			September 5, 2015 ...	us-west-2c	● in-use	None	 i-dc74ff07 (Secondar...
	vol-501821b7	40 GiB	gp2	120 / 3000			September 5, 2015 ...	us-west-2c	● in-use	None	 i-db74ff00 (ConfigSer...
	vol-8ee6317a	25 GiB	io1	200			September 5, 2015 ...	us-west-2b	● in-use	None	 i-91367954 (Secondar...
	vol-3fe631cb	8 GiB	gp2	24 / 3000		snap-bfb086e1	September 5, 2015 ...	us-west-2b	● in-use	None	 i-c8357a0d (ConfigSe...
	vol-3ee631ca	40 GiB	gp2	120 / 3000			September 5, 2015 ...	us-west-2b	● in-use	None	 i-c8357a0d (ConfigSe...
	vol-b7616279	8 GiB	gp2	24 / 3000		snap-bfb086e1	September 5, 2015 ...	us-west-2a	● in-use	None	 i-2ecd54e8 (Primary...)
	vol-fc616232	25 GiB	io1	250			September 5, 2015 ...	us-west-2a	● in-use	None	 i-2ecd54e8 (Primary...)
	vol-8de63179	8 GiB	gp2	24 / 3000		snap-bfb086e1	September 5, 2015 ...	us-west-2b	● in-use	None	 i-95367950 (Secondar...
	vol-2ee631da	25 GiB	io1	200			September 5, 2015 ...	us-west-2b	● in-use	None	 i-95367950 (Secondar...
	vol-56e631a2	8 GiB	gp2	24 / 3000			September 5, 2015 ...	us-west-2b	● in-use	None	 i-95367950 (Secondar...
	vol-6e6162a0	8 GiB	gp2	24 / 3000			September 5, 2015 ...	us-west-2a	● in-use	None	 i-2ecd54e8 (Primary...)
	vol-ff616231	25 GiB	io1	250			September 5, 2015 ...	us-west-2a	● in-use	None	 i-2fcf54e9 (PrimaryR...
	vol-b6616278	8 GiB	gp2	24 / 3000		snap-bfb086e1	September 5, 2015 ...	us-west-2a	● in-use	None	 i-2fcf54e9 (PrimaryR...
	vol-83e63177	25 GiB	io1	250			September 5, 2015 ...	us-west-2b	● in-use	None	 i-95367950 (Secondar...
	vol-b9616277	25 GiB	io1	200			September 5, 2015 ...	us-west-2a	● in-use	None	 i-2ecd54e8 (Primary...)
	vol-606162ae	8 GiB	gp2	24 / 3000			September 5, 2015 ...	us-west-2a	● in-use	None	 i-2fcf54e9 (PrimaryR...

Select a volume above

[Create Security Group](#)[Actions ▾](#) Filter by tags and attributes or search by keyword

<input type="checkbox"/>	Name	Group ID	Group Name	VPC ID	Description
<input type="checkbox"/>	sg-12043677		default	vpc-9f3f8dfa	default VPC security group
<input type="checkbox"/>	sg-2da05e49		launch-wizard-4	vpc-9f3f8dfa	launch-wizard-4 created 2015-06-29T21:59:28.268-04:00
<input type="checkbox"/>	sg-42535b27		default	vpc-13cb4c76	default VPC security group
<input type="checkbox"/>	sg-5d0b1b38		launch-wizard-3	vpc-9f3f8dfa	launch-wizard-3 created 2015-05-07T16:23:34.509-04:00
<input type="checkbox"/>	sg-80c77ce4		AWS-MongoDB-Infrastructur...	vpc-d69fe7b3	Enable external access and allow communication (Trim as needed)
<input type="checkbox"/>	sg-81c77ce5		AWS-MongoDB-Infrastructur...	vpc-d69fe7b3	Enable external access and allow communication (Trim as needed)
<input type="checkbox"/>	sg-82c77ce6		AWS-MongoDB-Infrastructur...	vpc-d69fe7b3	Enable external access and allow communication (Trim as needed)
<input type="checkbox"/>	sg-83c77ce7		AWS-MongoDB-Infrastructur...	vpc-d69fe7b3	Enable external access and allow communication (Trim as needed)
<input type="checkbox"/>	sg-855ba4e1		rds-launch-wizard-2	vpc-9f3f8dfa	Created from the RDS Management Console
<input type="checkbox"/>	sg-857d82e1		rds-launch-wizard	vpc-9f3f8dfa	Created from the RDS Management Console
<input type="checkbox"/>	sg-87c77ce3		AWS-MongoDB-Infrastructur...	vpc-d69fe7b3	Enable external access and allow communication (Trim as needed)
<input type="checkbox"/>	sg-88c77cec		AWS-MongoDB-Infrastructur...	vpc-d69fe7b3	Enable external access and allow communication (Trim as needed)
<input type="checkbox"/>	sg-89c77ced		AWS-MongoDB-Infrastructur...	vpc-d69fe7b3	Enable internal access to the NAT device
<input type="checkbox"/>	sg-8cc77ce8		AWS-MongoDB-Infrastructur...	vpc-d69fe7b3	Enable external access and allow communication (Trim as needed)
<input type="checkbox"/>	sg-8dc77ce9		AWS-MongoDB-Infrastructur...	vpc-d69fe7b3	Enable external access and allow communication (Trim as needed)
<input type="checkbox"/>	sg-8dcc0e8		postgres-sg	vpc-9f3f8dfa	postgres security group
<input type="checkbox"/>	sg-8e727aeb		launch-wizard-2	vpc-13cb4c76	launch-wizard-2 created 2015-04-24T22:47:30.966-04:00
<input type="checkbox"/>	sg-8fc77ceb		AWS-MongoDB-Infrastructur...	vpc-d69fe7b3	Enable external access and allow communication (Trim as needed)
<input type="checkbox"/>	sg-937b84f7		rds-launch-wizard-1	vpc-9f3f8dfa	Created from the RDS Management Console
<input type="checkbox"/>	sg-b0ccc0d5		mysql-sg	vpc-9f3f8dfa	mysql security group
<input type="checkbox"/>	sg-bec77cda		default	vpc-d69fe7b3	default VPC security group

Select a security group above

Filter by attributes or search by keyword					
	Elastic IP	Instance	Private IP Address	Scope	Public DNS
<input checked="" type="checkbox"/>	52.24.142.213	i-1aca53dc (NAT Instance (...	10.0.1.144	vpc-d69fe7b3	ec2-52-24-142-213.us-west-

| **Key pair name**



skondla_aws_keys.pem

skondla_aws_keys_oregon.pem

Create Network Interface **Actions**

Filter by tags and attributes or search by keyword

	Name	Network interface	Subnet ID	VPC ID	Zone	Security groups	Description	Instance ID	Status
<input type="checkbox"/>	eni-094a7240	subnet-da3758ad	vpc-d69fe7b3	us-west-2a	AWS-MongoDB-Infrastructure-PrimaryReplicaNode00NodeSecurityGroup-EEO0JJGTL7QG	Network Interface for Mongo Node	i-2ecd54e8		in-use
<input type="checkbox"/>	eni-0a4a7243	subnet-da3758ad	vpc-d69fe7b3	us-west-2a	AWS-MongoDB-Infrastructure-PrimaryReplicaNode10NodeSecurityGroup-1WF0VFE9DXM50	Network Interface for Mongo Node	i-2fcd54e9		in-use
<input type="checkbox"/>	eni-0d4a7244	subnet-da3758ad	vpc-d69fe7b3	us-west-2a	AWS-MongoDB-Infrastructure-ConfigServer0NodeSecurityGroup-14HR43D0EGV13	Network Interface for Mongo Node	i-cbcd540d		in-use
<input type="checkbox"/>	eni-4cbd4f2b	subnet-21d35944	vpc-9f3f8dfa	us-west-2b	launch-wizard-1	secondary network			available
<input type="checkbox"/>	eni-774a723e	subnet-db3758ac	vpc-d69fe7b3	us-west-2a	AWS-MongoDB-Infrastructure-NATSecurityGroup-41LQD8H4UUPC	External interface for the NAT instance	i-1aca53dc		in-use
<input type="checkbox"/>	eni-e34cf5b8	subnet-367bfa6f	vpc-d69fe7b3	us-west-2c	AWS-MongoDB-Infrastructure-SecondaryReplicaNode01NodeSecurityGroup-1ISDZWJ3HJHM3	Network Interface for Mongo Node	i-dc74ff07		in-use
<input type="checkbox"/>	eni-ec4cf5b7	subnet-367bfa6f	vpc-d69fe7b3	us-west-2c	AWS-MongoDB-Infrastructure-ConfigServer2NodeSecurityGroup-4P5PKECJMP3Z	Network Interface for Mongo Node	i-db74ff00		in-use
<input type="checkbox"/>	eni-ee4cf5b5	subnet-367bfa6f	vpc-d69fe7b3	us-west-2c	AWS-MongoDB-Infrastructure-SecondaryReplicaNode11NodeSecurityGroup-87LCHJWEM5ET	Network Interface for Mongo Node	i-5175fe8a		in-use
<input type="checkbox"/>	eni-f8189f9e	subnet-e52b6280	vpc-d69fe7b3	us-west-2b	AWS-MongoDB-Infrastructure-SecondaryReplicaNode00NodeSecurityGroup-1T24F2H18IB97	Network Interface for Mongo Node	i-95367950		in-use
<input type="checkbox"/>	eni-f9189f9f	subnet-e52b6280	vpc-d69fe7b3	us-west-2b	AWS-MongoDB-Infrastructure-ConfigServer1NodeSecurityGroup-1GQX0L8VMONSH	Network Interface for Mongo Node	i-c8357a0d		in-use
<input type="checkbox"/>	eni-fd189f9b	subnet-e52b6280	vpc-d69fe7b3	us-west-2b	AWS-MongoDB-Infrastructure-SecondaryReplicaNode10NodeSecurityGroup-14159MTC8LGG7	Network Interface for Mongo Node	i-91367954		in-use

Manage Tags

Filter:		Search Keys	X	Search Values	X						
	Tag Key	Tag Value		Total	Instances	AMIs	Volumes	Snapshots	Network Interfaces		
Manage Tag	aws:cloudformation:logical-id	SecondaryReplicaNode01NodeInterface		1	0	0	0	0	1		
Manage Tag	aws:cloudformation:logical-id	NATInstance		1	1	0	0	0	0		
Manage Tag	aws:cloudformation:logical-id	PrimaryReplicaNode00NodeInstanceGP2		1	1	0	0	0	0		
Manage Tag	aws:cloudformation:logical-id	ConfigServer2NodeInterface		1	0	0	0	0	1		
Manage Tag	aws:cloudformation:logical-id	SecondaryReplicaNode00NodeSecurityGroup		1	0	0	0	0	0		
Manage Tag	aws:cloudformation:logical-id	PrimaryReplicaNode10NodeSecurityGroup		1	0	0	0	0	0		
Manage Tag	aws:cloudformation:logical-id	NATInterface		1	0	0	0	0	1		
Manage Tag	aws:cloudformation:logical-id	PrimaryReplicaNode10NodeInterface		1	0	0	0	0	1		
Manage Tag	aws:cloudformation:logical-id	SecondaryReplicaNode11NodeInstanceGP2		1	1	0	0	0	0		
Manage Tag	aws:cloudformation:logical-id	ConfigServer0NodeInterface		1	0	0	0	0	1		
Manage Tag	aws:cloudformation:logical-id	NATSecurityGroup		1	0	0	0	0	0		
Manage Tag	aws:cloudformation:logical-id	ConfigServer2NodeInstance		1	1	0	0	0	0		
Manage Tag	aws:cloudformation:logical-id	PrimaryReplicaNode00NodeSecurityGroup		1	0	0	0	0	0		
Manage Tag	aws:cloudformation:logical-id	SecondaryReplicaNode10NodeSecurityGroup		1	0	0	0	0	0		
Manage Tag	aws:cloudformation:logical-id	SecondaryReplicaNode01NodeInstanceGP2		1	1	0	0	0	0		
Manage Tag	aws:cloudformation:logical-id	SecondaryReplicaNode00NodeInstanceGP2		1	1	0	0	0	0		
Manage Tag	aws:cloudformation:logical-id	PrimaryReplicaNode00NodeInterface		1	0	0	0	0	1		
Manage Tag	aws:cloudformation:logical-id	SecondaryReplicaNode00NodeInterface		1	0	0	0	0	1		
Manage Tag	aws:cloudformation:logical-id	ConfigServer0NodeSecurityGroup		1	0	0	0	0	0		
Manage Tag	aws:cloudformation:logical-id	SecondaryReplicaNode10NodeInstanceGP2		1	1	0	0	0	0		
Manage Tag	aws:cloudformation:logical-id	ConfigServer2NodeSecurityGroup		1	0	0	0	0	0		
Manage Tag	aws:cloudformation:logical-id	PrimaryReplicaNode10NodeInstanceGP2		1	1	0	0	0	0		
Manage Tag	aws:cloudformation:logical-id	ConfigServer1NodeInstance		1	1	0	0	0	0		
Manage Tag	aws:cloudformation:logical-id	ConfigServer0NodeInstance		1	1	0	0	0	0		
Manage Tag	aws:cloudformation:logical-id	ConfigServer1NodeSecurityGroup		1	0	0	0	0	0		



EC2 Service Limits

Amazon EC2 provides different resources that you can use, such as instances and volumes. When you create your AWS account, AWS sets limits for these resources on a per-region basis. This page lists your EC2 service limits in US West (Oregon).

Instance Limits

Name	Current Limit	Action
Running On-Demand EC2 instances i	20	Request limit increase
Running On-Demand c1.medium instances	20	Request limit increase
Running On-Demand c1.xlarge instances	20	Request limit increase
Running On-Demand c3.2xlarge instances	5	Request limit increase
Running On-Demand c3.4xlarge instances	1	Request limit increase
Running On-Demand c3.8xlarge instances	1	Request limit increase
Running On-Demand c3.large instances	5	Request limit increase
Running On-Demand c3.xlarge instances	5	Request limit increase
Running On-Demand c4.2xlarge instances	5	Request limit increase
Running On-Demand c4.4xlarge instances	1	Request limit increase
Running On-Demand c4.8xlarge instances	1	Request limit increase
Running On-Demand c4.large instances	5	Request limit increase
Running On-Demand c4.xlarge instances	5	Request limit increase
Running On-Demand cc2.8xlarge instances	1	Request limit increase
Running On-Demand cr1.8xlarge instances	2	Request limit increase
Running On-Demand d2.2xlarge instances	8	Request limit increase
Running On-Demand d2.4xlarge instances	4	Request limit increase
Running On-Demand d2.8xlarge instances	2	Request limit increase
Running On-Demand d2.xlarge instances	8	Request limit increase

EBS Limits

Name	Current Limit	Action
Provisioned IOPS (i)	40,000	Request limit increase
Provisioned IOPS (SSD) volume storage (TiB) (i)	20	Request limit increase
General Purpose (SSD) volume storage (TiB) (i)	20	Request limit increase
Magnetic volume storage (TiB) (i)	20	Request limit increase

Networking Limits

Name	Current Limit	Action
EC2-Classic Elastic IPs (i)	5	Request limit increase
EC2-VPC Elastic IPs (i)	5	Request limit increase
Rules per VPC security group (i)	50	Request limit increase
VPC security groups per elastic network interface (i)	5	Request limit increase
VPCs (i)	5	Request limit increase
Subnets per VPC (i)	200	Request limit increase
Security groups per VPC (i)	100	Request limit increase
Network ACLs per VPC (i)	200	Request limit increase
Rules per network ACL (i)	20	Request limit increase
Route tables per VPC (i)	200	Request limit increase
Entries per route table (i)	50	Request limit increase

Auto Scaling Limits

Name	Current Limit	Action
Auto Scaling Groups (i)	20	Request limit increase

[Launch Instance](#)[Connect](#)[Actions ▾](#)

- Connect
- Get Windows Password
- Launch More Like This
- Instance State ▶
- Instance Settings ▶
- Image ▶
- Networking ▶
- CloudWatch Monitoring ▶

Instance Type ▾

Availability Zone ▾

Instance State ▾

Status Checks

- Start
- Stop
- Reboot
- Terminate

Name		Instance Type	Availability Zone	Instance State	Status Checks
ConfigServer0	i-2ecd54e8	t2.micro	us-west-2a	stopped	
ConfigServer1	i-2fcd54e9	t2.micro	us-west-2a	stopped	
ConfigServer2			us-west-2b	stopped	
NAT Instance (Public Subnet)		t2.micro	us-west-2c	stopped	
PrimaryReplicaNode00	i-2ecd54e8	t2.micro	us-west-2a	stopped	
PrimaryReplicaNode10	i-2fcd54e9	t2.micro	us-west-2a	stopped	
SecondaryReplicaNode00	i-95367950	t2.micro	us-west-2b	stopped	
SecondaryReplicaNode01	i-dc74ff07	t2.micro	us-west-2c	stopped	
SecondaryReplicaNode10	i-91367954	t2.micro	us-west-2b	stopped	
SecondaryReplicaNode11	i-5175fe8a	t2.micro	us-west-2c	stopped	

Start Instances



Are you sure you want to start these instances?

- i-cbcd540d (ConfigServer0)
- i-c8357a0d (ConfigServer1)
- i-db74ff00 (ConfigServer2)
- i-1aca53dc (NAT Instance (Public Subnet))
- i-2ecd54e8 (PrimaryReplicaNode00)
- i-2fcd54e9 (PrimaryReplicaNode10)
- i-95367950 (SecondaryReplicaNode00)
- i-dc74ff07 (SecondaryReplicaNode01)
- ... 10 more instances

Cancel

Yes, Start

```
[ec2-user@ip-10-0-2-161 ~]$ cat /etc/mongod0.conf | grep -v "#" | sed '/^$/d'
logpath=/log/mongod0.log
logappend=true
fork=true
port=27018
dbpath=/data/s0-rs0
pidfilepath=/var/run/mongodb/mongod0.pid
replSet=s0-rs0
[ec2-user@ip-10-0-2-161 ~]$ cat /etc/mongod1.conf | grep -v "#" | sed '/^$/d'
logpath=/log/mongod1.log
logappend=true
fork=true
port=27019
dbpath=/data/s0-rs1
pidfilepath=/var/run/mongodb/mongod1.pid
replSet=s0-rs1
[ec2-user@ip-10-0-2-161 ~]$ cat /etc/mongod1.conf | grep -v "#" | sed '/^$/d'
logpath=/log/mongod1.log
logappend=true
fork=true
port=27019
dbpath=/data/s0-rs1
pidfilepath=/var/run/mongodb/mongod1.pid
replSet=s0-rs1
[ec2-user@ip-10-0-2-161 ~]$ mongo
MongoDB shell version: 3.0.6
connecting to: test
Server has startup warnings:
2015-09-06T12:59:15.579+0000 I CONTROL [initandlisten]
2015-09-06T12:59:15.579+0000 I CONTROL [initandlisten] ** WARNING: /sys/kernel/mm/transparent_hugepage/defrag is 'always'.
2015-09-06T12:59:15.579+0000 I CONTROL [initandlisten] **           We suggest setting it to 'never'
2015-09-06T12:59:15.579+0000 I CONTROL [initandlisten]
> exit
bye
[ec2-user@ip-10-0-2-161 ~]$ mongo --port 27018
MongoDB shell version: 3.0.6
connecting to: 127.0.0.1:27018/test
Server has startup warnings:
2015-09-06T12:59:16.065+0000 I CONTROL [initandlisten]
2015-09-06T12:59:16.065+0000 I CONTROL [initandlisten] ** WARNING: /sys/kernel/mm/transparent_hugepage/defrag is 'always'.
2015-09-06T12:59:16.065+0000 I CONTROL [initandlisten] **           We suggest setting it to 'never'
2015-09-06T12:59:16.065+0000 I CONTROL [initandlisten]
s0-rs0:PRIMARY> exit
bye
[ec2-user@ip-10-0-2-161 ~]$ mongo --port 27019
MongoDB shell version: 3.0.6
connecting to: 127.0.0.1:27019/test
Server has startup warnings:
2015-09-06T12:59:18.999+0000 I CONTROL [initandlisten]
2015-09-06T12:59:18.999+0000 I CONTROL [initandlisten] ** WARNING: /sys/kernel/mm/transparent_hugepage/defrag is 'always'.
2015-09-06T12:59:18.999+0000 I CONTROL [initandlisten] **           We suggest setting it to 'never'
2015-09-06T12:59:18.999+0000 I CONTROL [initandlisten]
s0-rs1:PRIMARY>
```

```
[ec2-user@ip-10-0-3-138 ~]$ mongo --port 27018
MongoDB shell version: 3.0.6
connecting to: 127.0.0.1:27018/test
Welcome to the MongoDB shell.
For interactive help, type "help".
For more comprehensive documentation, see
    http://docs.mongodb.org/
Questions? Try the support group
    http://groups.google.com/group/mongodb-user
Server has startup warnings:
2015-09-06T12:59:12.953+0000 I CONTROL  [initandlisten]
2015-09-06T12:59:12.953+0000 I CONTROL  [initandlisten] ** WARNING: /sys/kernel/mm/transparent_hugepage/defrag is 'always'.
2015-09-06T12:59:12.953+0000 I CONTROL  [initandlisten] ** We suggest setting it to 'never'
2015-09-06T12:59:12.953+0000 I CONTROL  [initandlisten]
s0-rs0:SECONDARY> exit
bye
[ec2-user@ip-10-0-3-138 ~]$ sudo su
[root@ip-10-0-3-138 ec2-user]# vi /etc/mongod.conf
[root@ip-10-0-3-138 ec2-user]# vi /etc/mongod0.conf
[root@ip-10-0-3-138 ec2-user]#
[root@ip-10-0-3-138 ec2-user]#
[root@ip-10-0-3-138 ec2-user]# ls -l /etc/mongo*
-rw-r--r-- 1 root root 1520 Sep  5 23:40 /etc/mongod0.conf
-rw-r--r-- 1 root root 1520 Sep  5 23:40 /etc/mongod1.conf
-rw-r--r-- 1 root root 1533 Aug 24 12:27 /etc/mongod.conf
[root@ip-10-0-3-138 ec2-user]# vi /etc/mongod.conf
[root@ip-10-0-3-138 ec2-user]#
[root@ip-10-0-3-138 ec2-user]#
[root@ip-10-0-3-138 ec2-user]# vi /etc/init.d/mongod
[root@ip-10-0-3-138 ec2-user]# vi /etc/init.d/mongod0
[root@ip-10-0-3-138 ec2-user]# vi /etc/init.d/mongod1
[root@ip-10-0-3-138 ec2-user]# service mongod restart
Stopping mongod:                                         [  OK  ]
Starting mongod:                                         [  OK  ]
[root@ip-10-0-3-138 ec2-user]# service mongod0 restart
Stopping mongod:                                         [  OK  ]
Starting mongod:                                         [  OK  ]
[root@ip-10-0-3-138 ec2-user]# service mongod1 restart
Stopping mongod:                                         [  OK  ]
Starting mongod:                                         [  OK  ]
[root@ip-10-0-3-138 ec2-user]# mongo --port 27018
MongoDB shell version: 3.0.6
connecting to: 127.0.0.1:27018/test
Welcome to the MongoDB shell.
For interactive help, type "help".
For more comprehensive documentation, see
    http://docs.mongodb.org/
Questions? Try the support group
    http://groups.google.com/group/mongodb-user
s0-rs0:SECONDARY> exit
bye
[root@ip-10-0-3-138 ec2-user]# mongo --port 27019
MongoDB shell version: 3.0.6
connecting to: 127.0.0.1:27019/test
s0-rs1:SECONDARY>
```

```

[ec2-user@ip-10-0-1-144 ~]$ scp -i "skondla_aws_keys_oregonpem.pem" ec2-user@10.0.2.161:~/*.gz .
[mongo_primary.etc.tar.gz
mongo_primary.var.tar.gz
[ec2-user@ip-10-0-1-144 ~]$ ls -l
total 2072
-rw-rw-r-- 1 ec2-user ec2-user 0 Sep 6 02:25 a.txt
drwxrwxr-x 5 ec2-user ec2-user 4096 Sep 6 03:40 conf
-rw-rw-r-- 1 ec2-user ec2-user 20480 Sep 6 03:44 mongodb.conf.tar
-rw-r--r-- 1 ec2-user ec2-user 1926349 Sep 6 14:56 mongo_primary.etc.tar.gz
-rw-r--r-- 1 ec2-user ec2-user 162185 Sep 6 14:56 mongo_primary.var.tar.gz
-r----- 1 ec2-user ec2-user 1696 Sep 6 02:27 skondla_aws_keys_oregonpem.pem
[ec2-user@ip-10-0-1-144 ~]$ scp -i "skondla_aws_keys_oregonpem.pem" ec2-user@10.0.3.138:~/*.gz .
[mongo_secondary.etc.tar.gz
mongo_secondary.var.tar.gz
[ec2-user@ip-10-0-1-144 ~]$ ls -l
total 4116
-rw-rw-r-- 1 ec2-user ec2-user 0 Sep 6 02:25 a.txt
drwxrwxr-x 5 ec2-user ec2-user 4096 Sep 6 03:40 conf
-rw-rw-r-- 1 ec2-user ec2-user 20480 Sep 6 03:44 mongodb.conf.tar
-rw-r--r-- 1 ec2-user ec2-user 1926349 Sep 6 14:56 mongo_primary.etc.tar.gz
-rw-r--r-- 1 ec2-user ec2-user 162185 Sep 6 14:56 mongo_primary.var.tar.gz
-rw-r--r-- 1 ec2-user ec2-user 1928437 Sep 6 14:57 mongo_secondary.etc.tar.gz
-rw-r--r-- 1 ec2-user ec2-user 161553 Sep 6 14:57 mongo_secondary.var.tar.gz
-r----- 1 ec2-user ec2-user 1696 Sep 6 02:27 skondla_aws_keys_oregonpem.pem
[ec2-user@ip-10-0-1-144 ~]$ scp -i "skondla_aws_keys_oregonpem.pem" ec2-user@10.0.2.236:~/*.gz .
[mongo_config.etc.tar.gz
mongo_config.var.tar.gz
[ec2-user@ip-10-0-1-144 ~]$ ls -ltr
total 6072
-rw-rw-r-- 1 ec2-user ec2-user 0 Sep 6 02:25 a.txt
-r----- 1 ec2-user ec2-user 1696 Sep 6 02:27 skondla_aws_keys_oregonpem.pem
drwxrwxr-x 5 ec2-user ec2-user 4096 Sep 6 03:40 conf
-rw-rw-r-- 1 ec2-user ec2-user 20480 Sep 6 03:44 mongodb.conf.tar
-rw-r--r-- 1 ec2-user ec2-user 162185 Sep 6 14:56 mongo_primary.var.tar.gz
-rw-r--r-- 1 ec2-user ec2-user 1926349 Sep 6 14:56 mongo_primary.etc.tar.gz
-rw-r--r-- 1 ec2-user ec2-user 1928437 Sep 6 14:57 mongo_secondary.etc.tar.gz
-rw-r--r-- 1 ec2-user ec2-user 161553 Sep 6 14:57 mongo_secondary.var.tar.gz
-rw-r--r-- 1 ec2-user ec2-user 75957 Sep 6 14:57 mongo_config.var.tar.gz
-rw-r--r-- 1 ec2-user ec2-user 1921201 Sep 6 14:57 mongo_config.etc.tar.gz
[ec2-user@ip-10-0-1-144 ~]$
```

```
[2015-09-06 11:00:35] /drives/c/Users/Sudhe_000/Downloads
[Sudhe_000.rose-win8] > scp -i skondla_aws_keys_oregon.pem ec2-user@52.24.142.213:/home/ec2-user/*.gz .
mongo_config.etc.tar.gz                                100% 1876KB 469.0KB/s  00:04
mongo_config.var.tar.gz                               100%   74KB  74.2KB/s  00:00
mongo_primary.etc.tar.gz                            100% 1881KB 627.1KB/s  00:03
mongo_primary.var.tar.gz                           100% 158KB 158.4KB/s  00:00
mongo_secondary.etc.tar.gz                         100% 1883KB 627.8KB/s  00:03
mongo_secondary.var.tar.gz                        100% 158KB 157.8KB/s  00:00

[2015-09-06 11:01:01] /drives/c/Users/Sudhe_000/Downloads
[Sudhe_000.rose-win8] > █
```

Create Stack Update Stack Delete Stack				
Filter: Active ▾ By Name: <input type="text"/>				
	Stack Name	Created Time	Status	Description
<input checked="" type="checkbox"/>	AWS-MongoDB-Infrastructure	2015-09-05 19:19:03 UTC-0400	DELETE_IN_PROGRESS	(000F) Deploy MongoDB on a New VPC in AWS

[Overview](#)
[Outputs](#)
[Resources](#)
[Events](#)
[Template](#)
[Parameters](#)
[Tags](#)
[Stack Policy](#)

Stack name: AWS-MongoDB-Infrastructure

Stack ID: arn:aws:cloudformation:us-west-2:518599548208:stack/AWS-MongoDB-Infrastructure/7ee411e0-5424-11e5-a278-50d50031c644

Status: DELETE_IN_PROGRESS

Status reason: User Initiated

Description: (000F) Deploy MongoDB on a New VPC in AWS



All Free Tier services by usage

Service	Month-to-date actual usage		Month-end forecasted usage		Free Tier usage limit	Status
EBS - Volumes	3.10 GB-Mo	10.33%	15.50 GB-Mo	51.67%	30 GB-Mo	
EC2 - Linux	49.00 Hrs	6.53%	245.00 Hrs	32.67%	750 Hrs	
CloudWatch - Alarms	0.18 Alarms	1.81%	0.90 Alarms	9.03%	10 Alarms	
S3 - Puts	8.00 Requests	0.40%	40.00 Requests	2.00%	2,000 Requests	
S3 - Gets	19.00 Requests	0.10%	95.00 Requests	0.48%	20,000 Requests	
KMS - Requests	1.00 Requests	0.01%	5.00 Requests	0.03%	20,000 Requests	
S3 - Storage	0.00 GB-Mo	0.00%	0.00 GB-Mo	0.00%	5 GB-Mo	

zon rvices SIMPLE MONTHLY CALCULATOR

AWS pricing helps you reduce costs in multiple ways. [Learn more about AWS pricing](#)

FREE USAGE TIER: New Customers get free usage tier for first 12 months

Services

Estimate of your Monthly Bill (\$ 111.31)

Choose region: US-West-2 (Oregon) ▾

Inbound Data Transfer

 Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides resizable compute capacity in the cloud. It is designed to make web-scale computing easier.

Compute: Amazon EC2 Instances:

	Description	Instances	Usage	Type	Billing Option	Monthly Cost
⊖	ConfigServer2NodeInstance	1	4 Hours/Day ▾	Linux on t2.micro	On-Demand (No Co	\$ 1.59
⊖	SecondaryReplicaNode00NodeInsta	1	4 Hours/Day ▾	Linux on t2.micro	On-Demand (No Co	\$ 1.59
⊖	PrimaryReplicaNode10NodeInsta	1	4 Hours/Day ▾	Linux on t2.micro	On-Demand (No Co	\$ 1.59
⊖	NATInstance	1	4 Hours/Day ▾	Linux on t2.micro	On-Demand (No Co	\$ 1.59
⊖	PrimaryReplicaNode00NodeInsta	1	4 Hours/Day ▾	Linux on t2.micro	On-Demand (No Co	\$ 1.59
⊖	SecondaryReplicaNode10NodeInsta	1	4 Hours/Day ▾	Linux on t2.micro	On-Demand (No Co	\$ 1.59
⊖	SecondaryReplicaNode11NodeInsta	1	4 Hours/Day ▾	Linux on t2.micro	On-Demand (No Co	\$ 1.59
⊖	ConfigServer0NodeInstance	1	4 Hours/Day ▾	Linux on t2.micro	On-Demand (No Co	\$ 1.59
⊖	ConfigServer1NodeInstance	1	4 Hours/Day ▾	Linux on t2.micro	On-Demand (No Co	\$ 1.59
⊖	SecondaryReplicaNode01NodeInsta	1	4 Hours/Day ▾	Linux on t2.micro	On-Demand (No Co	\$ 1.59
⊕	Add New Row					

Storage: Amazon EBS Volumes:

	Description	Volumes	Volume Type	Storage	IOPS	Snapshot Storage	
	EBS volumes	10	Provisioned IOPS (SSD) ▾	8 GB	100	0 GB-month of Storage	▼
	Add New Row						

Elastic IP:

Number of Additional Elastic IPs:

Elastic IP Non-attached Time: Hours/Month ▾

Number of Elastic IP Remaps: Per Month ▾

Data Transfer:

Inter-Region Data Transfer Out: GB/Month ▾

Data Transfer Out: GB/Month ▾

Data Transfer In: GB/Month ▾

VPC Peering Data Transfer: GB/Month ▾

Intra-Region Data Transfer: GB/Month ▾

Public IP/Elastic IP Data Transfer: GB/Month ▾

Elastic Load Balancing:

Number of Elastic LBs:

Total Data Processed by all ELBs: GB/Month ▾

Estimate of Your Monthly Bill Show First Month's Bill (include all one-time fees, if any)

	Amazon EC2 Service (US-West-2)	\$ 110.00
	AWS Data Transfer In	\$ 0.00
	AWS Data Transfer Out	\$ 0.00
	AWS Support (Basic)	\$ 0.00
Total Monthly Payment:		\$ 111.00