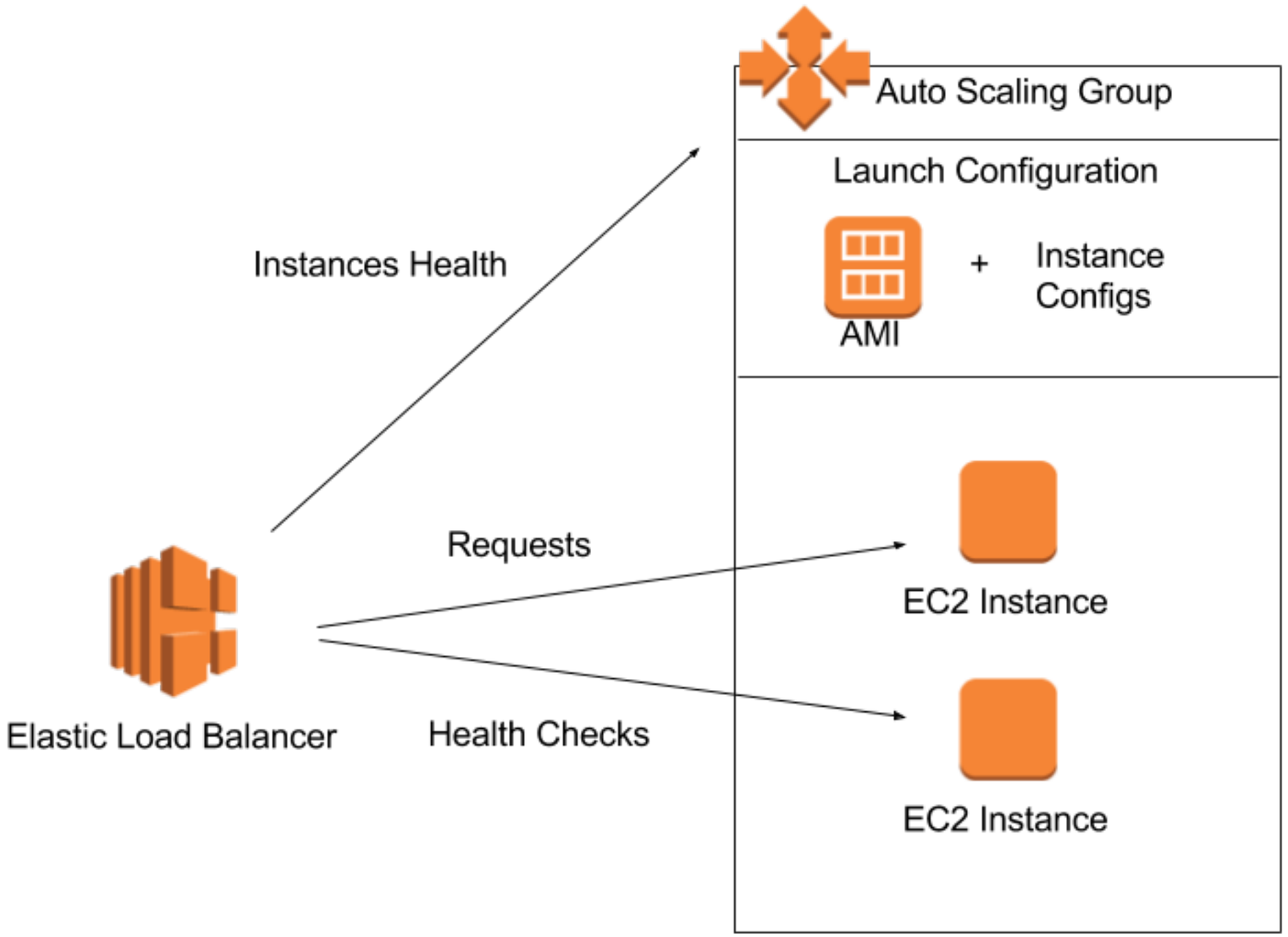
Deploy escalável simples

Utilizando Amazon AWS

Benefícios

- Escalabilidade
- Alta disponibilidade
- Zero Downtime Deploy

Overview


Auto Scaling Group
Launch Configuration
AMI
+
Instance
Configs
Instances Health
Requests
EC2 Instance
EC2 Instance
Elastic Load Balancer
Health Checks

Componentes

- **Amazon Machine Image (AMI)**
- EC2 Load Balancer
- EC2 Auto Scaling

Passo a passo
Amazon Machine Image

- **Criar máquina no ec2**

- Instalar dependencias

- Configurar nginx e php-fpm

- Criar script de inicialização

- Criar AMI (Amazon Machine Image)

AWS | Services | Edit | Alexandre Paez Sequeira

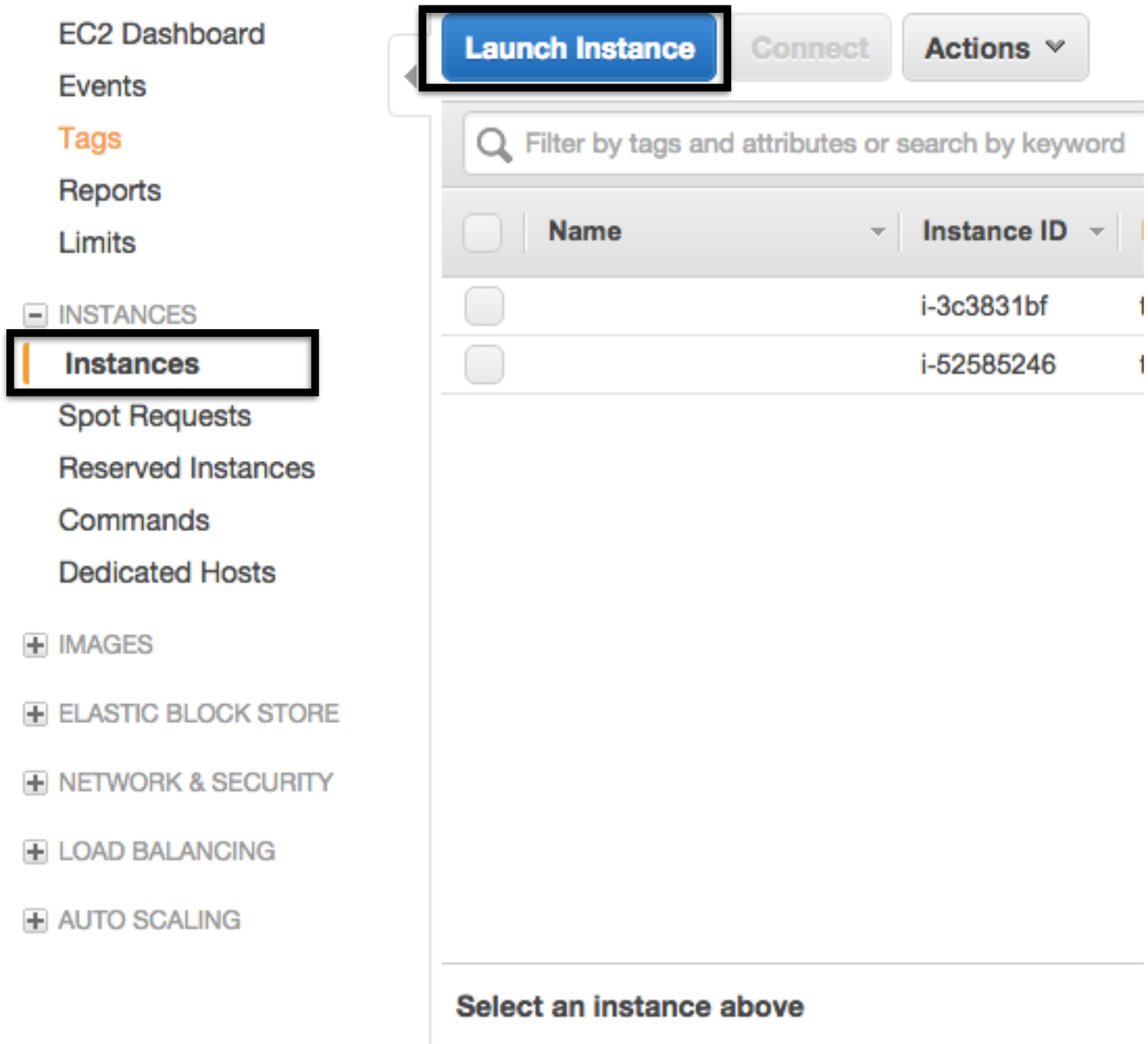
History

- EC2
- Console Home
- RDS
- CloudWatch
- Billing
- Elastic Beanstalk

All AWS Services

- Compute
- Storage & Content Delivery
- Database
- Networking
- Developer Tools
- Management Tools
- Security & Identity
- Analytics
- Internet of Things
- Mobile Services
- Application Services
- Enterprise Applications
- Game Development

- API Gateway
- AppStream
- AWS IoT
- Certificate Manager
- CloudFormation
- CloudFront
- CloudSearch
- CloudTrail
- CloudWatch
- CodeCommit
- CodeDeploy
- CodePipeline
- Cognito
- Config
- Data Pipeline
- Device Farm
- Direct Connect
- Directory Service
- DMS PREVIEW
- DynamoDB
- EC2
- EC2 Container Service
- Elastic Beanstalk
- Elastic File System PREVIEW
- Elastic Transcoder
- ElastiCache
- Elasticsearch Service
- EMR
- GameLift
- Glacier
- IAM
- Import/Export Snowball
- Inspector PREVIEW
- Kinesis
- Lambda
- Machine Learning
- Mobile Analytics
- Mobile Hub
- OpsWorks
- RDS
- Redshift
- Route 53
- S3
- Service Catalog
- SES
- SNS
- SQS
- Storage Gateway
- SWF
- Trusted Advisor
- VPC
- WAF
- WorkDocs
- WorkMail
- WorkSpaces


EC2 Dashboard
Events
Tags
Reports
Limits
INSTANCES
Instances
Spot Requests
Reserved Instances
Commands
Dedicated Hosts
IMAGES
ELASTIC BLOCK STORE
NETWORK & SECURITY
LOAD BALANCING
AUTO SCALING
Launch Instance
Connect
Actions
Filter by tags and attributes or search by keyword
Name
Instance ID
i-3c3831bf
i-52585246
Select an instance above

Ubuntu Server 14.04 LTS (PV), SSD Volume Type - ami-feb73692

Ubuntu

Free tier eligible

Ubuntu Server 14.04 LTS (PV),EBS General Purpose (SSD) Volume Type. Support available from Canonical (http://www.ubuntu.com/cloud/services).

Root device type: ebs Virtualization type: paravirtual

Select

64-bit

Step 2: Choose an Instance Type

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. Learn more about instance types and how they can meet your computing needs.

Filter by: All instance types All generations Show/Hide Columns

Currently selected: t1.micro (Variable ECUs, 1 vCPUs, 0.613 GiB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
☑	Micro instances	t1.micro Free tier eligible	1	0.613	EBS only	-	Very Low
⊘	General purpose	t2.nano	1	0.5	EBS only	-	Low to Moderate
⊘	General purpose	t2.micro Free tier eligible	1	1	EBS only	-	Low to Moderate
⊘	General purpose	t2.small	1	2	EBS only	-	Low to Moderate
⊘	General purpose	t2.medium	2	4	EBS only	-	Low to Moderate
⊘	General purpose	t2.large	2	8	EBS only	-	Low to Moderate
☐	General purpose	m3.medium	1	3.75	1 x 4 (SSD)	-	Moderate
☐	General purpose	m3.large	2	7.5	1 x 32 (SSD)	-	Moderate
☐	General purpose	m3.xlarge	4	15	2 x 40 (SSD)	Yes	High

Cancel Previous Review and Launch Next: Configure Instance Details

Select an existing key pair or create a new key pair

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about removing existing key pairs from a public AMI.

Create a new key pair

Key pair name

Download Key Pair

You have to download the **private key file** (*.pem file) before you can continue. **Store it in a secure and accessible location.** You will not be able to download the file again after it's created.

Cancel **Launch Instances**

Passo a passo
Amazon Machine Image

- Criar máquina no ec2
- **Instalar dependencias**
- Configurar nginx e php-fpm
- Criar script de inicialização
- Criar AMI (Amazon Machine Image)

Instalar dependencias

- ssh -i ~/.ssh/chave ubuntu@x.x.x.x
- apt-get update
- apt-get install nginx php5-fpm git

Passo a passo
Amazon Machine Image

- Criar máquina no ec2
- Instalar dependencias
- **Configurar nginx e php5-fpm**
- Criar script de inicialização
- Criar AMI (Amazon Machine Image)

Configurar NGINX

```
server {
        #listen 80 default_server;
        #listen [::]:80 default_server ipv6only=on;

        root /home/ubuntu/app;
        index index.php index.html index.htm;

        # Make site accessible from http://localhost/
        server_name _;

        location / {
                # First attempt to serve request as file, then
                # as directory, then fall back to displaying a 404.
                try_files $uri $uri/ =404;
                # Uncomment to enable naxsi on this location
                # include /etc/nginx/naxsi.rules
        }

        location /doc/ {
                alias /usr/share/doc/;
                autoindex on;
                allow 127.0.0.1;
                allow ::1;
                deny all;
        }

        # Only for nginx-naxsi used with nginx-naxsi-ui : process denied requests
        #location /RequestDenied {
        #       proxy_pass http://127.0.0.1:8080;
        #}

        #error_page 404 /404.html;

        # redirect server error pages to the static page /50x.html
        #
        error_page 500 502 503 504 /50x.html;
        location = /50x.html {
                root /usr/share/nginx/html;
        }
```

- /etc/nginx/sites-available/default
- service nginx restart

```
        # pass the PHP scripts to FastCGI server listening on 127.0.0.1:9000
        #
        location ~ \.php$ {
                try_files $uri =404;
                fastcgi_split_path_info ^(.+\.php)(/.+)$;
                # NOTE: You should have "cgi.fix_pathinfo = 0;" in php.ini

                # With php5-cgi alone:
                #fastcgi_pass 127.0.0.1:9000;
                # With php5-fpm:
                fastcgi_pass unix:/var/run/php5-fpm.sock;
                fastcgi_index index.php;
                include fastcgi_params;
        }

        # deny access to .htaccess files, if Apache's document root
        # concurs with nginx's one
        #
        location ~ /\.ht {
                deny all;
        }
}
```

Configurar php5-fpm

```
; cgi.fix_pathinfo provides *real* PATH_INFO/PATH_TRANSLATED support for CGI.  PHP's
; previous behaviour was to set PATH_TRANSLATED to SCRIPT_FILENAME, and to not grok
; what PATH_INFO is.  For more information on PATH_INFO, see the cgi specs.  Setting
; this to 1 will cause PHP CGI to fix its paths to conform to the spec.  A setting
; of zero causes PHP to behave as before.  Default is 1.  You should fix your scripts
; to use SCRIPT_FILENAME rather than PATH_TRANSLATED.
; http://php.net/cgi.fix-pathinfo
cgi.fix_pathinfo=0
```

- /etc/php5/fpm/php.ini
- service php5-fpm reload

Passo a passo
Amazon Machine Image

- Criar máquina no ec2
- Instalar dependencias
- Configurar nginx e php-fpm
- **Criar script de inicialização**
- Criar AMI (Amazon Machine Image)

Scripts Inicialização

- /etc/rc.local

- ~/getsource

/etc/rc.local

```
#!/bin/sh -e
#
# rc.local
#
# This script is executed at the end of each multiuser runlevel.
# Make sure that the script will "exit 0" on success or any other
# value on error.
#
# In order to enable or disable this script just change the execution
# bits.
#
# By default this script does nothing.

exec 2> /tmp/rc.local.log  # send stderr from rc.local to a log file
exec 1>&2                  # send stdout to the same log file
set -x                     # tell sh to display commands before execution

sudo -E su - ubuntu -c "./getsource"

exit 0
```

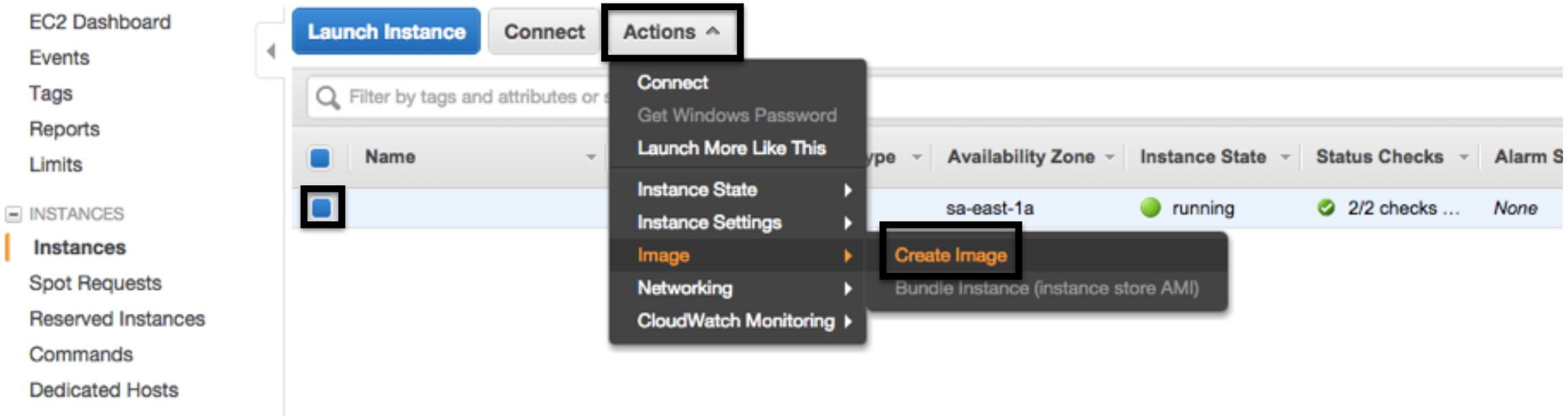
getsource

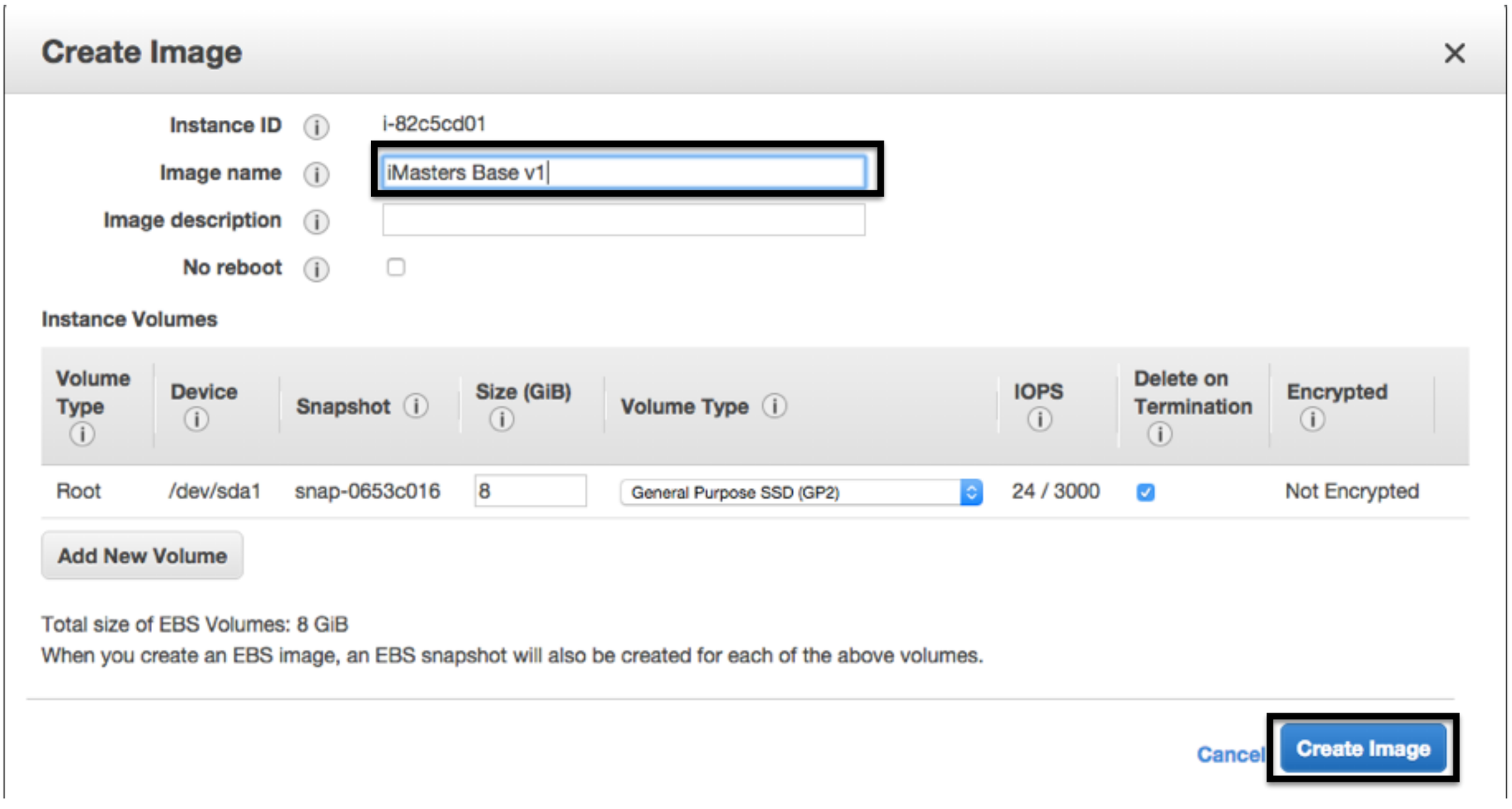
```
#!/bin/bash
source ~/.bashrc
# GET USER DATA
USER_DATA=`curl http://169.254.169.254/latest/user-data/ --retry 3 --silent --fail`

cd /home/ubuntu/app
git clone https://github.com/alepaez/helloAWS.git .
git reset --hard ${USER_DATA}
echo '<?php phpinfo(); ?>' >> info.php
```

- Ler user-data
- Clonar projeto do github
- Usa o user-data para selecionar commit específico

Passo a passo
Amazon Machine Image

- Criar máquina no ec2
- Instalar dependencias
- Configurar nginx e php-fpm
- Criar script de inicialização
- **Criar AMI (Amazon Machine Image)**

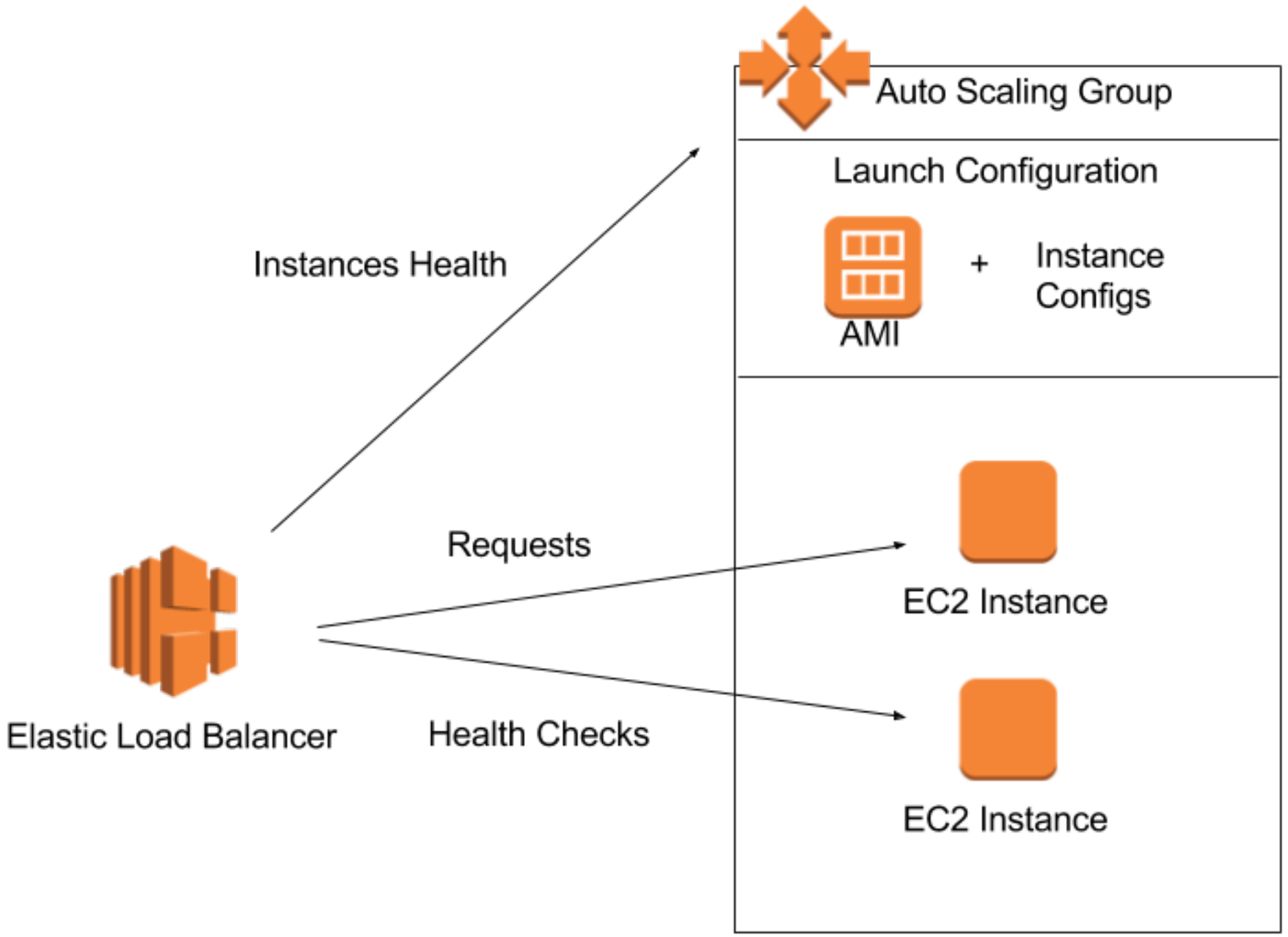

EC2 Dashboard
Events
Tags
Reports
Limits
INSTANCES
Instances
Spot Requests
Reserved Instances
Commands
Dedicated Hosts
Launch Instance
Connect
Actions
Connect
Get Windows Password
Launch More Like This
Instance State
Instance Settings
Image
Networking
CloudWatch Monitoring
Create Image
Bundle Instance (instance store AMI)
Filter by tags and attributes or
Name
Availability Zone
Instance State
Status Checks
Alarm S
sa-east-1a
running
2/2 checks ...
None


Create Image
Instance ID
i-82c5cd01
Image name
iMasters Base v1
Image description
No reboot
Instance Volumes
Volume Type
Device
Snapshot
Size (GiB)
Volume Type
IOPS
Delete on Termination
Encrypted
Root
/dev/sda1
snap-0653c016
8
General Purpose SSD (GP2)
24 / 3000
Not Encrypted
Add New Volume
Total size of EBS Volumes: 8 GiB
When you create an EBS image, an EBS snapshot will also be created for each of the above volumes.
Cancel
Create Image

Componentes

- Amazon Machine Image (AMI)
- **EC2 Load Balancer**
- EC2 Auto Scaling Group

Overview


Auto Scaling Group
Launch Configuration
AMI
+
Instance
Configs
Instances Health
Requests
EC2 Instance
EC2 Instance
Elastic Load Balancer
Health Checks

EC2 Dashboard

Events

Tags

Reports

Limits

INSTANCES

IMAGES

ELASTIC BLOCK STORE

NETWORK & SECURITY

LOAD BALANCING

Load Balancers

AUTO SCALING

Launch Configurations

Auto Scaling Groups

Create Load Balancer **Actions**

Filter: Search Load Balancers

Select a Load Balancer

1. Define Load Balancer | 2. Assign Security Groups | 3. Configure Security Settings | 4. Configure Health Check | 5. Add EC2 Instances | 6. Add Tags | 7. Review

Step 1: Define Load Balancer

Basic Configuration

This wizard will walk you through setting up a new load balancer. Begin by giving your new load balancer a unique name so that you can identify it from other load balancers you might create. You will also need to configure ports and protocols for your load balancer. Traffic from your clients can be routed from any load balancer port to any port on your EC2 instances. By default, we've configured your load balancer with a standard web server on port 80.

Load Balancer name: iMasters

Create LB Inside: My Default VPC (172.31.0.0/16)

Create an internal load balancer: (what's this?)

Enable advanced VPC configuration:

Listener Configuration:

Load Balancer Protocol	Load Balancer Port	Instance Protocol	Instance Port
HTTP	80	HTTP	80

Add

Cancel | Next: Assign Security Groups

1. Define Load Balancer | **2. Assign Security Groups** | 3. Configure Security Settings | 4. Configure Health Check | 5. Add EC2 Instances | 6. Add Tags | 7. Review

Step 2: Assign Security Groups

You have selected the option of having your Elastic Load Balancer inside of a VPC, which allows you to assign security groups to your load balancer. Please select the security groups to assign to this load balancer. This can be changed at any time.

Assign a security group: (•) Create a **new** security group

( ) Select an **existing** security group

Security group name: web

Description: web

Type	Protocol	Port Range	Source	
HTTP	TCP	80	Anywhere 0.0.0.0/0	⊗
HTTPS	TCP	443	Anywhere 0.0.0.0/0	⊗

Add Rule

Cancel | Previous | Next: Configure Security Settings

1. Define Load Balancer | 2. Assign Security Groups | 3. Configure Security Settings | **4. Configure Health Check** | 5. Add EC2 Instances | 6. Add Tags | 7. Review

Step 4: Configure Health Check

Your load balancer will automatically perform health checks on your EC2 instances and only route traffic to instances that pass the health check. If an instance fails the health check, it is automatically removed from the load balancer. Customize the health check to meet your specific needs.

Ping Protocol HTTP

Ping Port 80

Ping Path /info.php

Advanced Details

Response Timeout 5 seconds

Health Check Interval 30 seconds

Unhealthy Threshold 2

Healthy Threshold 10

Cancel | Previous | Next: Add EC2 Instances

1. Define Load Balancer | 2. Assign Security Groups | 3. Configure Security Settings | 4. Configure Health Check | 5. Add EC2 Instances | 6. Add Tags | **7. Review**

Step 7: Review

Please review the load balancer details before continuing

Define Load Balancer

Edit load balancer definition

Load Balancer name: iMasters
Scheme: internet-facing
Port Configuration: 80 (HTTP) forwarding to 80 (HTTP)

Configure Health Check

Edit health check

Ping Target: HTTP:80/info.php
Timeout: 5 seconds
Interval: 30 seconds
Unhealthy Threshold: 2
Healthy Threshold: 10

Add EC2 Instances

Edit instances

Cross-Zone Load Balancing: Enabled
Connection Draining: Enabled, 300 seconds
Instances:

VPC Information

Edit subnets

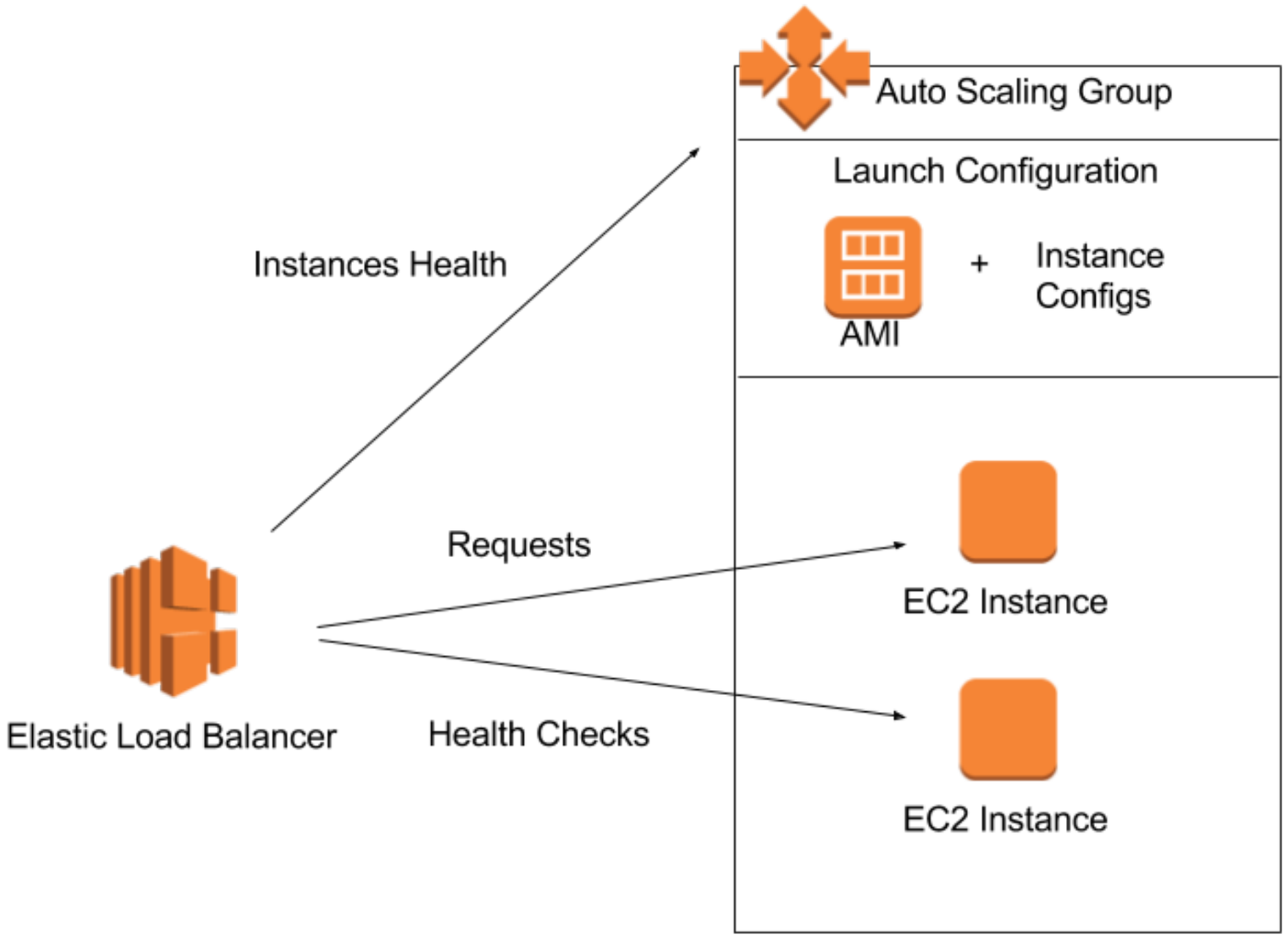
VPC: vpc-cfd9feaa
Subnets: subnet-7f7c9f1b, subnet-70fa4329, subnet-e0e49297

Cancel | Previous | Create

Componentes

- Amazon Machine Image (AMI)
- EC2 Load Balancer
- **EC2 Auto Scaling**

Overview


Auto Scaling Group
Launch Configuration
AMI
+
Instance
Configs
Instances Health
Requests
EC2 Instance
EC2 Instance
Elastic Load Balancer
Health Checks

Passo a passo
Auto Scaling Group

- **Launch Configuration**

- Auto Scaling Group

EC2 Dashboard
Events
Tags
Reports
Limits
INSTANCES
IMAGES
ELASTIC BLOCK STORE
NETWORK & SECURITY
LOAD BALANCING
AUTO SCALING
Launch Configurations
Auto Scaling Groups

Welcome to Auto Scaling

You can use Auto Scaling to manage Amazon EC2 capacity automatically, maintain the right number of instances for your application, operate a healthy group of instances, and scale it according to your needs.
Learn more

Create Auto Scaling group

Note: To create your Auto Scaling groups in a different region, select your region from the navigation bar.

Benefits of Auto Scaling

Reusable Instance Templates

Provision instances based on a reusable template you define, called a launch configuration.

Learn more

Automated Provisioning

Keep your Auto Scaling group healthy and balanced, whether you need one instance or 1,000.

Learn more

Adjustable Capacity

Maintain a fixed group size or adjust dynamically based on Amazon CloudWatch metrics.

Learn more

Create Auto Scaling Group

Cancel and Exit

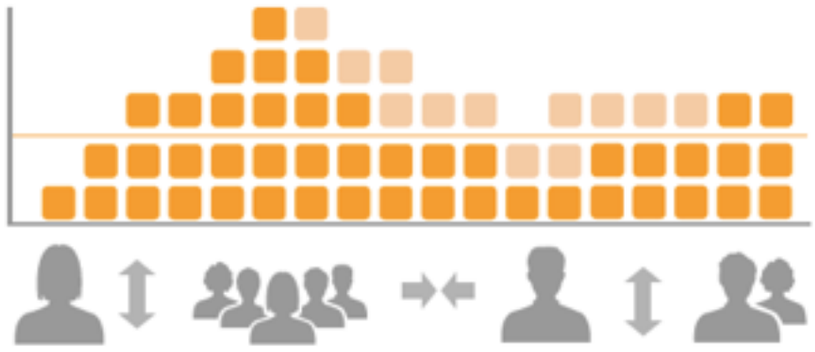
To create an Auto Scaling group, you will first need to choose a template that your Auto Scaling group will use when it launches instances for you, called a launch configuration. Choose a launch configuration or create a new one, and then apply it to your group.

Later, if you want to use a different template, you can create another launch configuration and apply it to this group, even if you already have instances running in it. Using this method, you can update the software that your group uses when it launches new instances.

Step 1: Create launch configuration

First, define a template that your Auto Scaling group will use to launch instances.

You can change your group's launch configuration at any time.



Step 2: Create Auto Scaling group

Next, give your group a name and specify how many instances you want to run in it.

Your group will maintain this number of instances, and replace any that become unhealthy or impaired.

You can optionally configure your group to adjust in capacity according to demand, in response to Amazon CloudWatch metrics.

Cancel | Create launch configuration

1. Choose AMI | 2. Choose Instance Type | 3. Configure details | 4. Add Storage | 5. Configure Security Group | 6. Review

Create Launch Configuration

Cancel and Exit

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, our user community, or the AWS Marketplace; or you can select one of your own AMIs.

- Quick Start
- **My AMIs**
- AWS Marketplace
- Community AMIs

Search my AMIs

1 to 1 of 1 AMIs

iMasters Base v0.3 - ami-8357d7ef

Root device type: ebs Virtualization type: paravirtual Owner: 370398177892

Select

64-bit

Ownership

- [x] Owned by me
- [ ] Shared with me

Architecture

- [ ] 32-bit
- [ ] 64-bit

Root device type

- [ ] EBS
- [ ] Instance store

1. Choose AMI | **2. Choose Instance Type** | 3. Configure details | 4. Add Storage | 5. Configure Security Group | 6. Review

Create Launch Configuration

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. Learn more about instance types and how they can meet your computing needs.

Filter by: All instance types | All generations | Show/Hide Columns

Currently selected: t1.micro (Variable ECUs, 1 vCPUs, 0.613 GiB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
■	Micro instances	t1.micro Free tier eligible	1	0.613	EBS only	-	Very Low
⊘	General purpose	t2.nano	1	0.5	EBS only	-	Low to Moderate
⊘	General purpose	t2.micro Free tier eligible	1	1	EBS only	-	Low to Moderate
⊘	General purpose	t2.small	1	2	EBS only	-	Low to Moderate
⊘	General purpose	t2.medium	2	4	EBS only	-	Low to Moderate
⊘	General purpose	t2.large	2	8	EBS only	-	Low to Moderate
☐	General purpose	m3.medium	1	3.75	1 x 4 (SSD)	-	Moderate
☐	General purpose	m3.large	2	7.5	1 x 32 (SSD)	-	Moderate
☐	General purpose	m3.xlarge	4	15	2 x 40 (SSD)	Yes	High

Cancel | Previous | **Next: Configure details**

1. Choose AMI | 2. Choose Instance Type | **3. Configure details** | 4. Add Storage | 5. Configure Security Group | 6. Review

Create Launch Configuration

Name (i) masters - 9e8a2cdf0960fd5ac63ba93ffbacf0dd23

Purchasing option (i) ☐ Request Spot Instances

IAM role (i) None

Monitoring (i) ☐ Enable CloudWatch detailed monitoring
Learn more

▼ Advanced Details

Kernel ID (i) Use default

RAM Disk ID (i) Use default

User data (i) ◉ As text ○ As file ☐ Input is already base64 encoded

```
9e8a2cdf0960fd5ac63ba93ffbacf0dd236c042b
```

IP Address Type (i)
◉ Only assign a public IP address to instances launched in the default VPC and subnet. (default)
○ Assign a public IP address to every instance.
○ Do not assign a public IP address to any instances.
Note: this option only affects instances launched into an Amazon VPC

Cancel | Previous | Skip to review | **Next: Add Storage**

1. Choose AMI | 2. Choose Instance Type | 3. Configure details | **4. Add Storage** | 5. Configure Security Group | 6. Review

Create Launch Configuration

Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes.
https://docs.aws.amazon.com/console/ec2/launchinstance/storage about storage options in Amazon EC2.

Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Delete on Termination	Encrypted
Root	/dev/sda1	snap-0653c016	8	General Purpose (SSD)	24 / 3000	☑	No

Add New Volume

Free tier eligible customers can get up to 30 GB of EBS storage. Learn more about free usage tier eligibility and usage restrictions.

Cancel | Previous | Skip to review | Next: Configure Security Group

1. Choose AMI | 2. Choose Instance Type | 3. Configure details | 4. Add Storage | **5. Configure Security Group** | 6. Review

Create Launch Configuration

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. Learn more about Amazon EC2 security groups.

Assign a security group: ○ Create a **new** security group

● Select an **existing** security group

	Security Group ID	Name	VPC ID	Description	Actions
☐	sg-ad4615c8	default	vpc-cfd9feaa	default VPC security group	Copy to new
☐	sg-393d085c	web	vpc-cfd9feaa	web	Copy to new
☑	sg-2d4f1c48	web-ssh	vpc-cfd9feaa	web and ssh	Copy to new

Inbound rules for sg-2d4f1c48 Selected security groups: sg-2d4f1c48.

Type ⓘ	Protocol ⓘ	Port Range ⓘ	Source ⓘ
HTTP	TCP	80	0.0.0.0/0
SSH	TCP	22	0.0.0.0/0
HTTPS	TCP	443	0.0.0.0/0

Cancel | Previous | **Review**

1. Choose AMI | 2. Choose Instance Type | 3. Configure details | 4. Add Storage | 5. Configure Security Group | **6. Review**

Create Launch Configuration

Review the details of your launch configuration. You can go back to edit the details of each section before you finish.

Improve security of instances launched using your launch configuration, imasters - 9e8a2cdf0960fd5ac63ba93ffbacf0dd236c042b. Your security group, web-ssh, is open to the world.

Your instances may be accessible from any IP address. We recommend that you update your security group rules to allow access from known IP addresses only.
You can also open additional ports in your security group to facilitate access to the application or service you're running, e.g., HTTP (80) for web servers. Edit security groups

AMI Details

Edit AMI

iMasters Base v0.3 - ami-8357d7ef

Root device type: ebs Virtualization Type: paravirtual

Instance Type

Edit instance type

Instance Type	ECUs	vCPUs	Memory GiB	Instance Storage (GiB) GiB	EBS-Optimized Available	Network Performance
t1.micro	Variable	1	0.613	EBS only	-	Very Low

Launch configuration details

Edit details

Name imasters - 9e8a2cdf0960fd5ac63ba93ffbacf0dd236c042b
Purchasing option On demand
EBS Optimized No
Monitoring No
IAM role None

Cancel | Previous | **Create launch configuration**

Select an existing key pair or create a new key pair

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about removing existing key pairs from a public AMI.

Choose an existing key pair

Select a key pair

testlab

I acknowledge that I have access to the selected private key file (testlab.pem), and that without this file, I won't be able to log into my instance.

Cancel **Create launch configuration**

Passo a passo
Auto Scaling Group

- Launch Configuration

- **Auto Scaling Group**

1. Configure Auto Scaling group details | 2. Configure scaling policies | 3. Configure Notifications | 4. Configure Tags | 5. Review

Create Auto Scaling Group

Cancel and Exit

Launch Configuration imasters - 9e8a2cdf0960fd5ac63ba93ffbacf0dd236c042b

Group name version 9e8a2cd

Group size Start with 3 instances

Network vpc-cfd9feaa (172.31.0.0/16) (default) Create new VPC

Subnet
- subnet-7f7c9f1b(172.31.0.0/20) | Default in sa-east-1a
- subnet-e0e49297(172.31.16.0/20) | Default in sa-east-1b
- subnet-70fa4329(172.31.32.0/20) | Default in sa-east-1c

Create new subnet

Each instance in this Auto Scaling group will be assigned a public IP address.

Advanced Details

Load Balancing Receive traffic from Elastic Load Balancer(s)

iMasters

Health Check Type ELB EC2

Health Check Grace Period 300 seconds

Monitoring Amazon EC2 Detailed Monitoring metrics, which are provided at 1 minute frequency, are not enabled for the launch configuration imasters -

Cancel | Next: Configure scaling policies

1. Configure Auto Scaling group details | 2. Configure scaling policies | 3. Configure Notifications | 4. Configure Tags | 5. Review

Create Auto Scaling Group

You can optionally add scaling policies if you want to adjust the size (number of instances) of your group automatically. A scaling policy is a set of instructions for making such adjustments in response to an Amazon CloudWatch alarm that you assign to it. In each policy, you can choose to add or remove a specific number of instances or a percentage of the existing group size, or you can set the group to an exact size. When the alarm triggers, it will execute the policy and adjust the size of your group accordingly. Learn more about scaling policies.

- ○ **Keep this group at its initial size**
- ◉ **Use scaling policies to adjust the capacity of this group**

Scale between 1 and 3 instances. These will be the minimum and maximum size of your group.

Create Auto Scaling Group

Increase Group Size

Name: Increase Group Size

Execute policy when: awsec2-imasters-stress-test-CPU-Utilization Add new alarm

breaches the alarm threshold: CPUUtilization >= 50 for 300 seconds
for the metric dimensions AutoScalingGroupName = imasters - stress test

Take the action: Add 1 instances when 50 <= CPUUtilization < +infinity

Add step

Instances need: 300 seconds to warm up after each step

Create a simple scaling policy

Decrease Group Size

Name: Decrease Group Size

Execute policy when: awsec2-imasters-stress-test-High-CPU-Utilization Add new alarm

breaches the alarm threshold: CPUUtilization <= 30 for 900 seconds
for the metric dimensions AutoScalingGroupName = imasters - stress test

Take the action: Remove 1 instances when 30 >= CPUUtilization > -infinity

Add step

Create a simple scaling policy

Cancel | Previous | Review | Next: Configure Notifications

1. Configure Auto Scaling group details | 2. Configure scaling policies | 3. Configure Notifications | 4. Configure Tags | **5. Review**

Create Auto Scaling Group

Please review your Auto Scaling group details. You can go back to edit changes for each section. Click **Create Auto Scaling group** to complete the creation of an Auto Scaling group.

Auto Scaling Group Details

Edit details

Group name	version 9e8a2cd
Group size	3
Minimum Group Size	3
Maximum Group Size	3
Subnet(s)	subnet-7f7c9f1b,subnet-70fa4329,subnet-e0e49297
Load Balancers	iMasters
Health Check Type	ELB
Health Check Grace Period	300
Detailed Monitoring	No
Instance Protection	None

Scaling Policies

Edit scaling policies

Notifications

Edit notifications

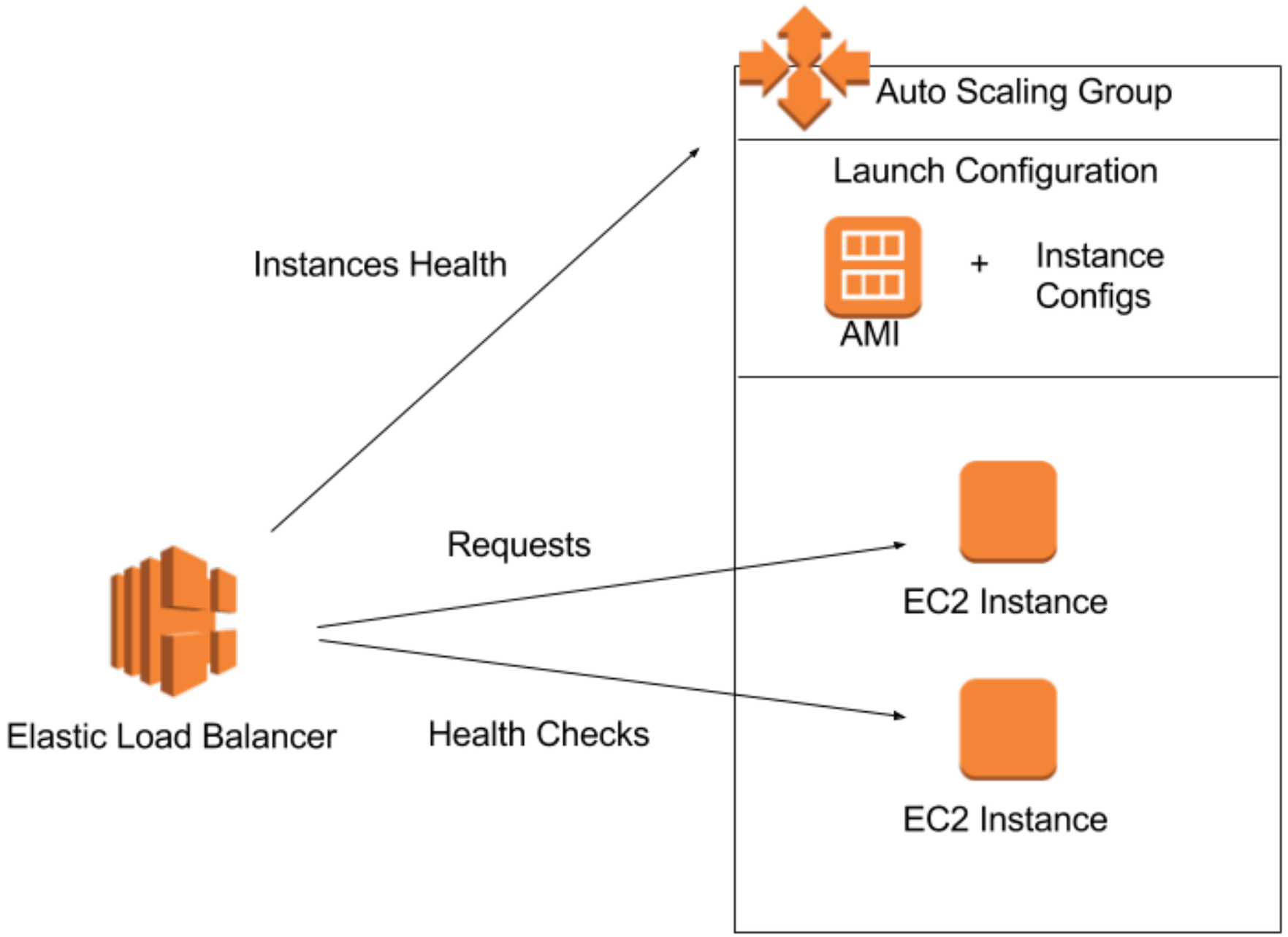
Tags

Edit tags

name	iMasters Instance - 9e8a2cd	*tag new instances*

Cancel | Previous | **Create Auto Scaling group**

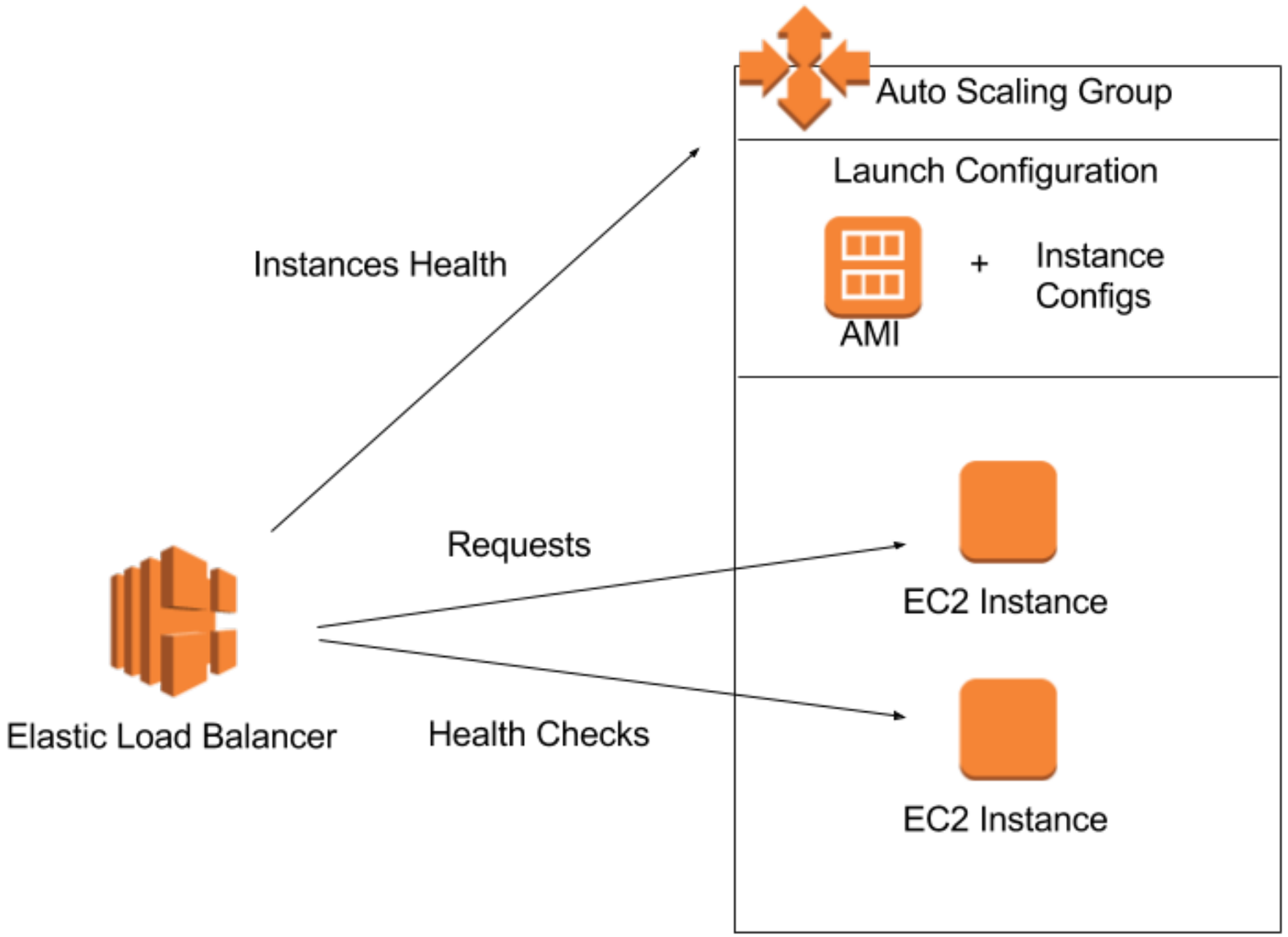
Overview


Auto Scaling Group
Launch Configuration
AMI
+
Instance
Configs
Instances Health
Requests
EC2 Instance
EC2 Instance
Elastic Load Balancer
Health Checks

Zero Downtime Deploy

- Criar Launch configuration para versão nova
- Criar Auto Scaling Group usando nova launch configuration
- Esperar máquinas da versão nova começarem a servir a aplicação
- Diminuir a quantidade de máquinas do Auto Scaling Group antigo para zero

Overview


Auto Scaling Group
Launch Configuration
AMI
+
Instance
Configs
Instances Health
Requests
EC2 Instance
EC2 Instance
Elastic Load Balancer
Health Checks

Demonstração

Perguntas

Muito obrigado!!