Enterprise EC2
Quick Start Guide

v1.7.0

rev. 1.0.0

Table of Contents

Introduction

Amazon Web Services (AWS) provides a cloud based platform to deploy web services. It allows services to be deployed as and when required. Charges are made for what is used making it an extremely flexible and cost effective solution.

Loadbalancer.org's EC2 based load balancer allows customers to rapidly deploy and configure a load balancing solution. The load balancer utilizes HAProxy & Pound, both well proven in our existing product range.

Why use Loadbalancer.org's EC2 Load Balancer?

Amazon already enable users to setup an EC2 based load balancer for load balancing other EC2 instances running in the cloud. Loadbalancer.org's EC2 load balancer offers the following additional functionality:

1. WAN or SNAT load balancing (i.e. non-EC2 based servers)
2. URL matching rules or multiple back-end clusters
3. Customizable timeouts for custom applications
4. Statistics reports providing extensive session related information
5. Session distribution based on actual server load (utilizing Loadbalancer.org's CPU idle agents which are available for both Linux & Windows)
6. Source IP based persistence
7. HTTP Cookie based persistence
8. RDP Cookie based persistence
9. Terminal Services / Remote Desktop Services Connection Broker support
10. Support for multiple IP addresses when used within a VPC
11. UDP protocol support

Amazon Terminology

Acronym	**Definition**
Amazon AWS	Amazon Web Services
Amazon S3	Amazon Simple Storage Service
Amazon EC2	Amazon Elastic Compute Cloud
Amazon VPC	Amazon Virtual Private Cloud
Amazon AMI	Amazon Machine Image
Amazon EBS	Elastic Block Store
EIP	Eiastic IP Address

Getting Started

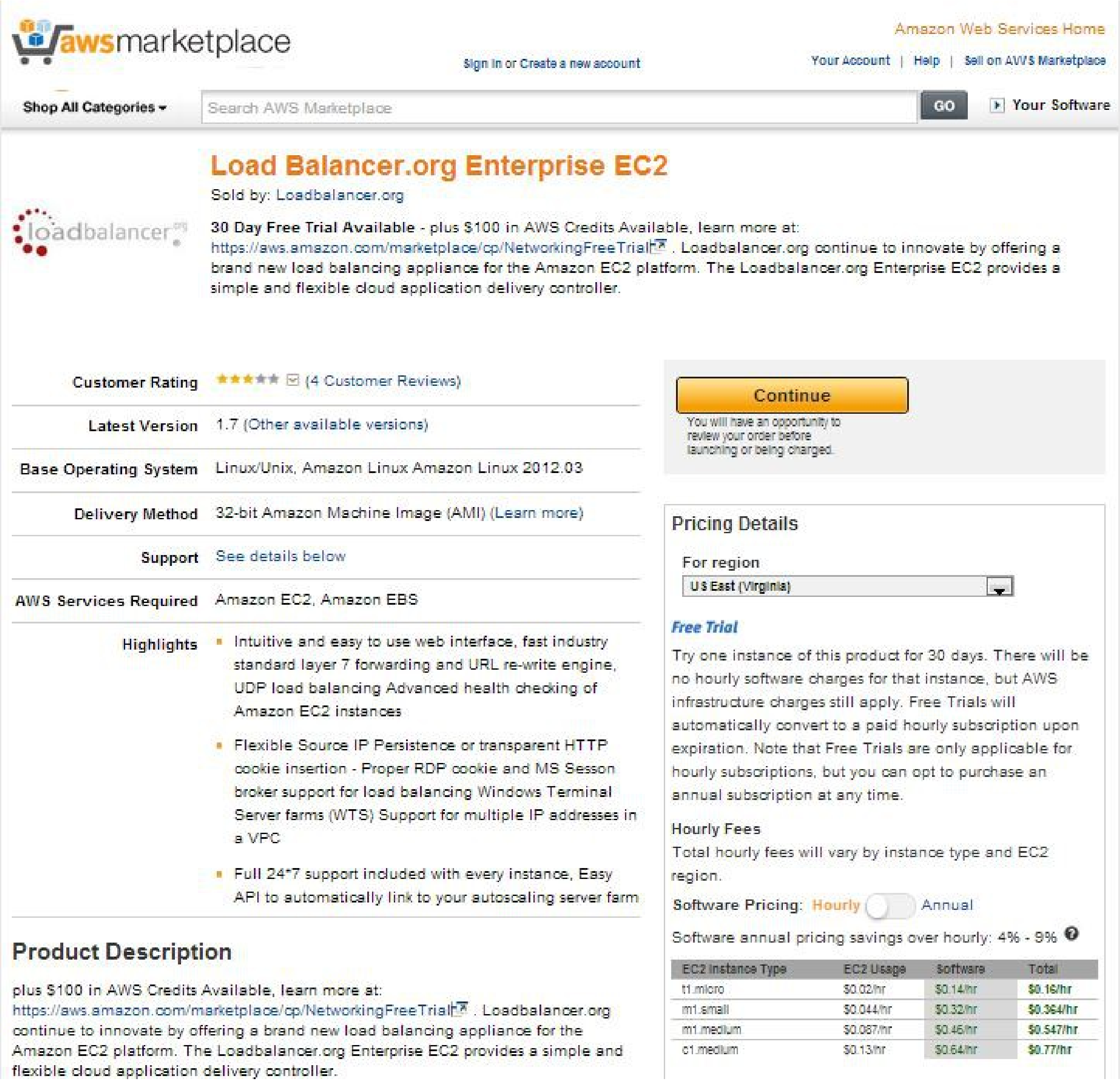
To start using Amazon web Services (AWS), you'll first need to create an account. This can be done using the following link : https://aws.amazon.com/

Accessing the Load Balancer AMI

Using AWS Market Place

The Loadbalancer.org EC2 AMI can be accessed using the following Amazon AWS Marketplace link:

https://aws.amazon.com/marketplace/pp/B008VJWTHO/ref=srh_res_product_title



Load Balancer.org Enterprise EC2

Sold by: Loadbalancer.org

30 Day Free Trial Available - plus $100 in AWS Credits Available, learn more at: https://aws.amazon.com/marketplace/cp/NetworkingFreeTrial . Loadbalancer.org continue to innovate by offering a brand new load balancing appliance for the Amazon EC2 platform. The Loadbalancer.org Enterprise EC2 provides a simple and flexible cloud application delivery controller.

Customer Rating	★★★★★ (4 Customer Reviews)
Latest Version	1.7 (Other available versions)
Base Operating System	Linux/Unix, Amazon Linux Amazon Linux 2012.03
Delivery Method	32-bit Amazon Machine Image (AMI) (Learn more)
Support	See details below
AWS Services Required	Amazon EC2, Amazon EBS
Highlights	▪ Intuitive and easy to use web interface, fast industry standard layer 7 forwarding and URL re-write engine, UDP load balancing Advanced health checking of Amazon EC2 instances ▪ Flexible Source IP Persistence or transparent HTTP cookie insertion - Proper RDP cookie and MS Sesson broker support for load balancing Windows Terminal Server farms (WTS) Support for multiple IP addresses in a VPC ▪ Full 24*7 support included with every instance, Easy API to automatically link to your autoscaling server farm

You will have an opportunity to review your order before launching or being charged.

Product Description

plus $100 in AWS Credits Available, learn more at: https://aws.amazon.com/marketplace/cp/NetworkingFreeTrial . Loadbalancer.org continue to innovate by offering a brand new load balancing appliance for the Amazon EC2 platform. The Loadbalancer.org Enterprise EC2 provides a simple and flexible cloud application delivery controller.

Pricing Details

For region

US East (Virginia)

Free Trial

Try one instance of this product for 30 days. There will be no hourly software charges for that instance, but AWS infrastructure charges still apply. Free Trials will automatically convert to a paid hourly subscription upon expiration. Note that Free Trials are only applicable for hourly subscriptions, but you can opt to purchase an annual subscription at any time.

Hourly Fees

Total hourly fees will vary by instance type and EC2 region.

Software Pricing: Hourly Annual

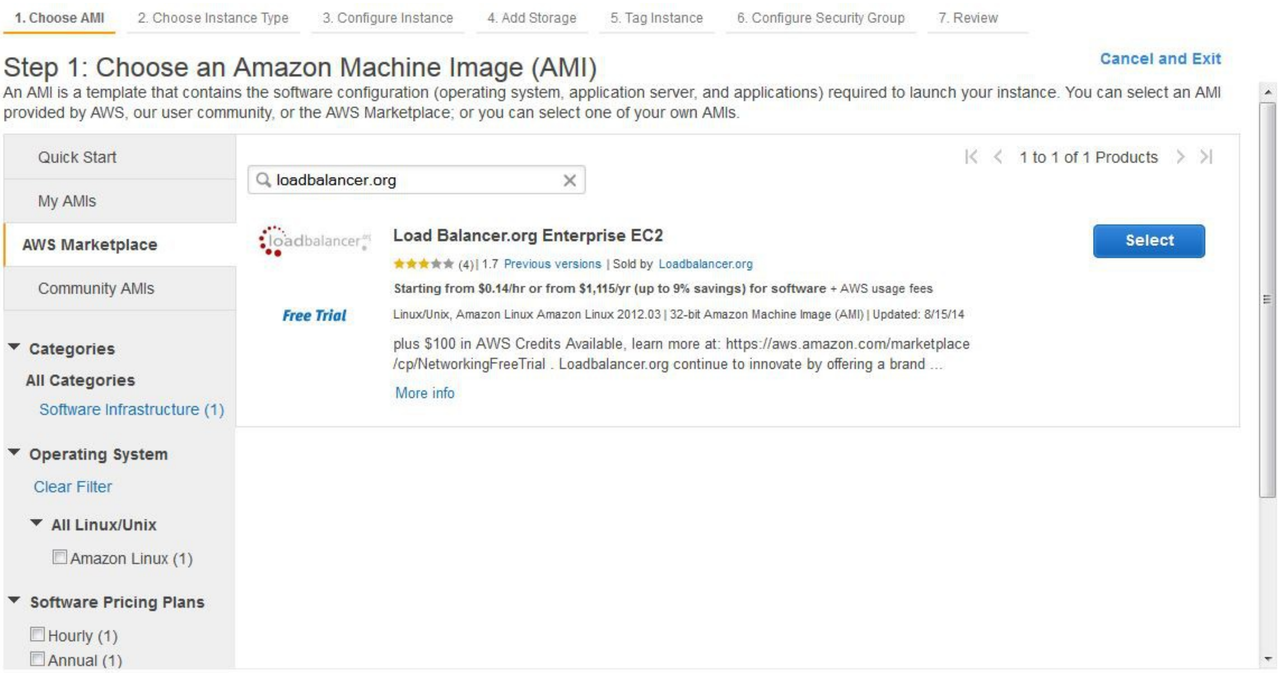
Software annual pricing savings over hourly: 4% - 9%

EC2 Instance Type	EC2 Usage	Software	Total
t1.micro	$0.02/hr	$0.14/hr	$0.16/hr
m1.small	$0.044/hr	$0.32/hr	$0.364/hr
m1.medium	$0.087/hr	$0.46/hr	$0.547/hr
c1.medium	$0.13/hr	$0.64/hr	$0.77/hr

To deploy a new instance, click **Continue**. You'll then be taken to the AWS login page where you can either create a new AWS account if you don't have one already, or login using your existing credentials. Once logged in, you'll be presented with 2 deployment options: **1-Click Launch** & **Manual Launch**. To use these options please continue on page 12 of this guide.

Using the EC2 Console

The Loadbalancer.org EC2 AMI can also be accessed via the EC2 Management Console. Once logged in to the console, select *EC2* and click the **Launch Instance** button. Then select *AWS Marketplace* and enter 'Loadbalancer.org' in the search box. This will display the Loadbalancer.org EC2 AMI as shown below:



Step 1: Choose an Amazon Machine Image (AMI)

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.

loadbalancer.org

Load Balancer.org Enterprise EC2

★★★★★ (4) | 1.7 Previous versions | Sold by Loadbalancer.org

Starting from $0.14/hr or from $1,115/yr (up to 9% savings) for software + AWS usage fees

Linux/Unix, Amazon Linux Amazon Linux 2012.03 | 32-bit Amazon Machine Image (AMI) | Updated: 8/15/14

plus $100 in AWS Credits Available, learn more at: https://aws.amazon.com/marketplace /cp/NetworkingFreeTrial . Loadbalancer.org continue to innovate by offering a brand ...

To deploy a new instance, click the **Select** button. You'll then be taken to the *Step 2: Choose an instance Type* page where instance configuration commences. The configuration steps from here on are the same as those when accessing the product directly from the AWS Marketplace. Please see page 12 of this guide for more details.

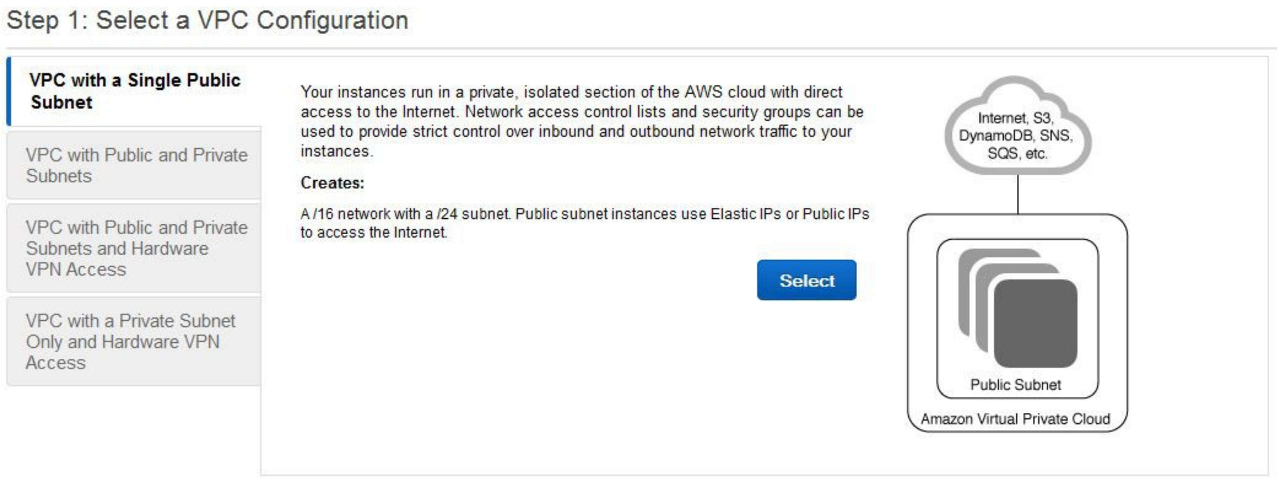
Deployment Concepts

Deployment Options

Instances can be deployed in 2 fundamental ways, these are described below:

Platform	Introduced In	Description
EC2-Classic	The original release of Amazon EC2	Instances run in a single, flat network that you share with other customers.
EC2-VPC	The original release of Amazon VPC	Instances run in a virtual private cloud (VPC) that's logically isolated to your AWS account.

When using the wizard to configure VPC's there are 4 types of VPC as detailed below.



Step 1: Select a VPC Configuration

VPC with a Single Public Subnet
VPC with Public and Private Subnets
VPC with Public and Private Subnets and Hardware VPN Access
VPC with a Private Subnet Only and Hardware VPN Access
Select
Internet, S3, DynamoDB, SNS, SQS, etc.
Public Subnet
Amazon Virtual Private Cloud

Your instances run in a private, isolated section of the AWS cloud with direct access to the Internet. Network access control lists and security groups can be used to provide strict control over inbound and outbound network traffic to your instances.

Creates:

A /16 network with a /24 subnet. Public subnet instances use Elastic IPs or Public IPs to access the Internet.

Type	Description	Creates
VPC with a Single Public Subnet	Instances run in a private, isolated section of the AWS cloud with direct access to the Internet. Network access control lists and security groups can be used to provide strict control over inbound and outbound network traffic to your instances.	A /16 network with a /24 subnet. Public subnet instances use Elastic IPs or Public IPs to access the Internet.
VPC with Public and Private Subnets	In addition to containing a public subnet, this configuration adds a private subnet whose instances are not addressable from the Internet. Instances in the private subnet can establish outbound connections to the Internet via the public subnet using Network Address Translation (NAT).	A /16 network with two /24 subnets. Public subnet instances use Elastic IPs to access the Internet. Private subnet instances access the Internet via a Network Address Translation (NAT) instance in the public subnet. (Hourly charges for NAT instances apply.)
VPC with Public and Private Subnets and Hardware VPN Access	This configuration adds an IPsec Virtual Private Network (VPN) connection between your Amazon VPC and your data center - effectively extending your data center to the cloud while	A /16 network with two /24 subnets. One subnet is directly connected to the Internet while the other subnet is connected to your

	also providing direct access to the Internet for public subnet instances in your Amazon VPC.	corporate network via IPsec VPN tunnel. (VPN charges apply.)
VPC with a Private Subnet Only and Hardware VPN Access	Your instances run in a private, isolated section of the AWS cloud with a private subnet whose instances are not addressable from the Internet. You can connect this private subnet to your corporate data center via an IPsec Virtual Private Network (VPN) tunnel.	A /16 network with a /24 subnet and provisions an IPsec VPN tunnel between your Amazon VPC and your corporate network. (VPN charges apply.)

EC2 IP address Types

Private

The internal RFC 1918 address of an instance that is only routable within the EC2 Cloud. Network traffic originating outside the EC2 network cannot route to this IP, and must use the Public IP or Elastic IP Address mapped to the instance.

Public

Internet routable IP address assigned by the system for all instances. Traffic routed to the Public IP is translated via 1:1 Network Address Translation (NAT) and forwarded to the Private IP address of an instance. The mapping of a Public IP to Private IP of an instance is the default launch configuration for all instance types. Public IP Addresses are no longer usable upon instance termination.

Elastic

Internet routable IP address allocated to an AWS EC2 account. Similar to EC2 Public Address, 1:1 NAT is used to map Elastic IP Addresses with their associated Private IP addresses. Unlike a standard EC2 Public IP Address, Elastic IP Addresses are allocated to accounts and can be remapped to other instances when desired.

Loadbalancer.org EC2 Appliance IP Address Allocations

EC2 Classic

When an instance is launched in EC2-Classic, 1 private IP address and 1 public IP address are automatically assigned to the instance. The instance can only have a single public IP address at anytime. If an EIP is allocated, this will replace the default IP address. If the EIP is removed, a new public IP will be automatically assigned to the instance within a few minutes. An EIP is persistent across a reboot, a standard public IP address is not.

EC2 VPC

When an instance is launched in EC2-VPC, 1 single private IP address is assigned by default, additional private IP addresses can be assigned if needed. If the instance is deployed in the default subnet, a public IP address is assigned by default. If deployed to a non default subnet no public address is assigned by default. Multiple EIPs can be allocated by defining multiple private IP's and then associating the EIPs.

N.B. the maximum number of EIP's depends on the instance type as defined here:

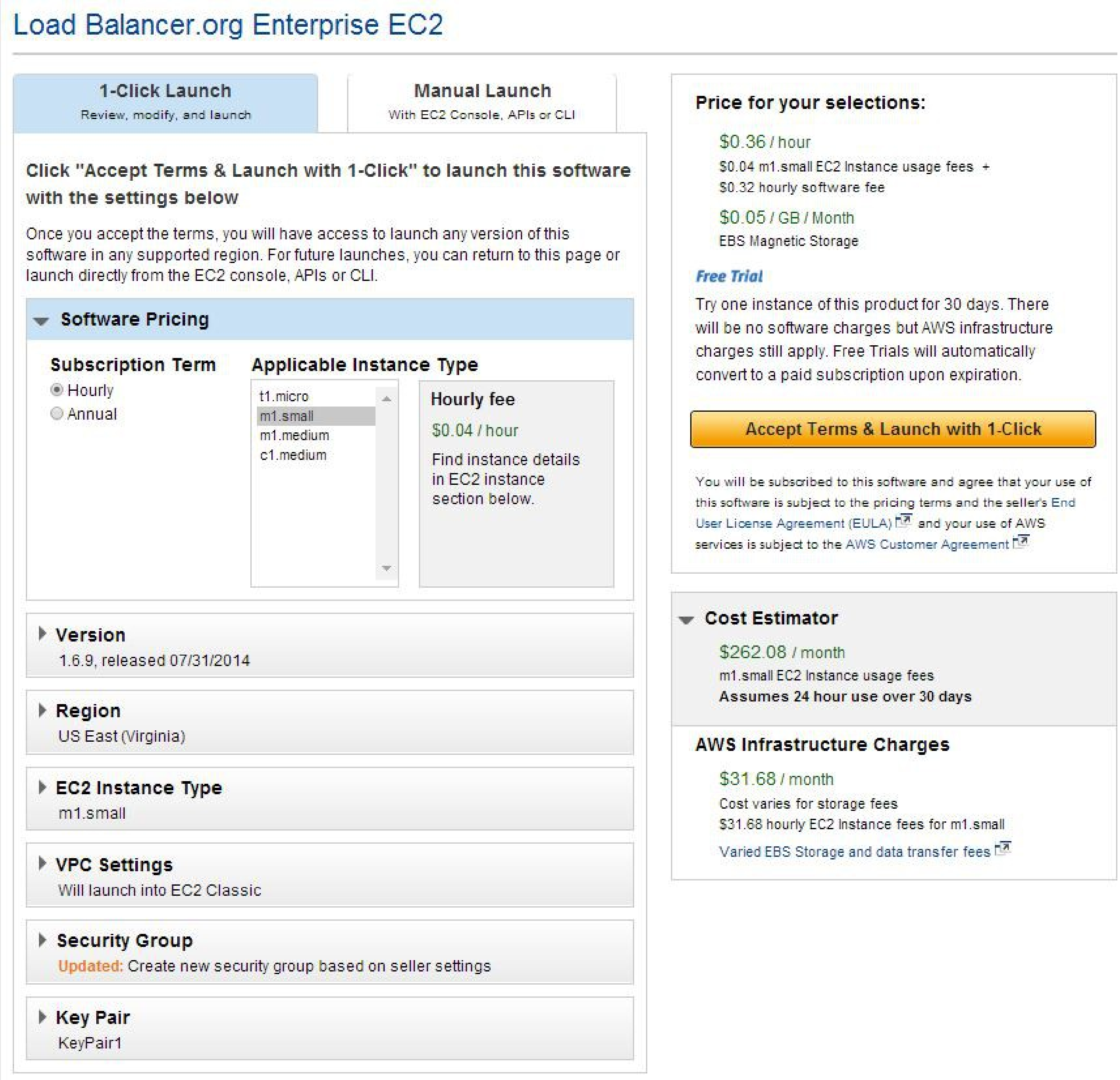
http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-eni.html#AvailableIpPerENI

Deploying the Load Balancer AMI

N.B. The first time this product is launched, the instance will be deployed as 30 day trial. There will be no software charges but AWS infrastructure charges do still apply. Free Trials will automatically convert to a paid subscription upon expiration.

AWS MarketPlace 1-Click launch

This method is the quickest way to get up and running, but doesn't permit access to all configuration options.



Load Balancer.org Enterprise EC2

Click "Accept Terms & Launch with 1-Click" to launch this software with the settings below

Once you accept the terms, you will have access to launch any version of this software in any supported region. For future launches, you can return to this page or launch directly from the EC2 console, APIs or CLI.

Software Pricing

Subscription Term
● Hourly
○ Annual

Applicable Instance Type
t1.micro
m1.small
m1.medium
c1.medium

Hourly fee
$0.04 / hour
Find instance details in EC2 instance section below.

Version
1.6.9, released 07/31/2014

Region
US East (Virginia)

EC2 Instance Type
m1.small

VPC Settings
Will launch into EC2 Classic

Security Group
Updated: Create new security group based on seller settings

Key Pair
KeyPair1

Price for your selections:

$0.36 / hour
$0.04 m1.small EC2 Instance usage fees +
$0.32 hourly software fee

$0.05 / GB / Month
EBS Magnetic Storage

Free Trial
Try one instance of this product for 30 days. There will be no software charges but AWS infrastructure charges still apply. Free Trials will automatically convert to a paid subscription upon expiration.

You will be subscribed to this software and agree that your use of this software is subject to the pricing terms and the seller's End User License Agreement (EULA) and your use of AWS services is subject to the AWS Customer Agreement

Cost Estimator
$262.08 / month
m1.small EC2 Instance usage fees
Assumes 24 hour use over 30 days

AWS Infrastructure Charges
$31.68 / month
Cost varies for storage fees
$31.68 hourly EC2 Instance fees for m1.small
Varied EBS Storage and data transfer fees

- Select the **Version** – the latest version is always recommended
- Select the **Region** – the default is *US East (Virginia)*
- Select the **EC2 Instance Type** – the default is *Small*

- Select the **VPC Settings** – the default is *EC2 Classic*, i.e. do not use a VPC

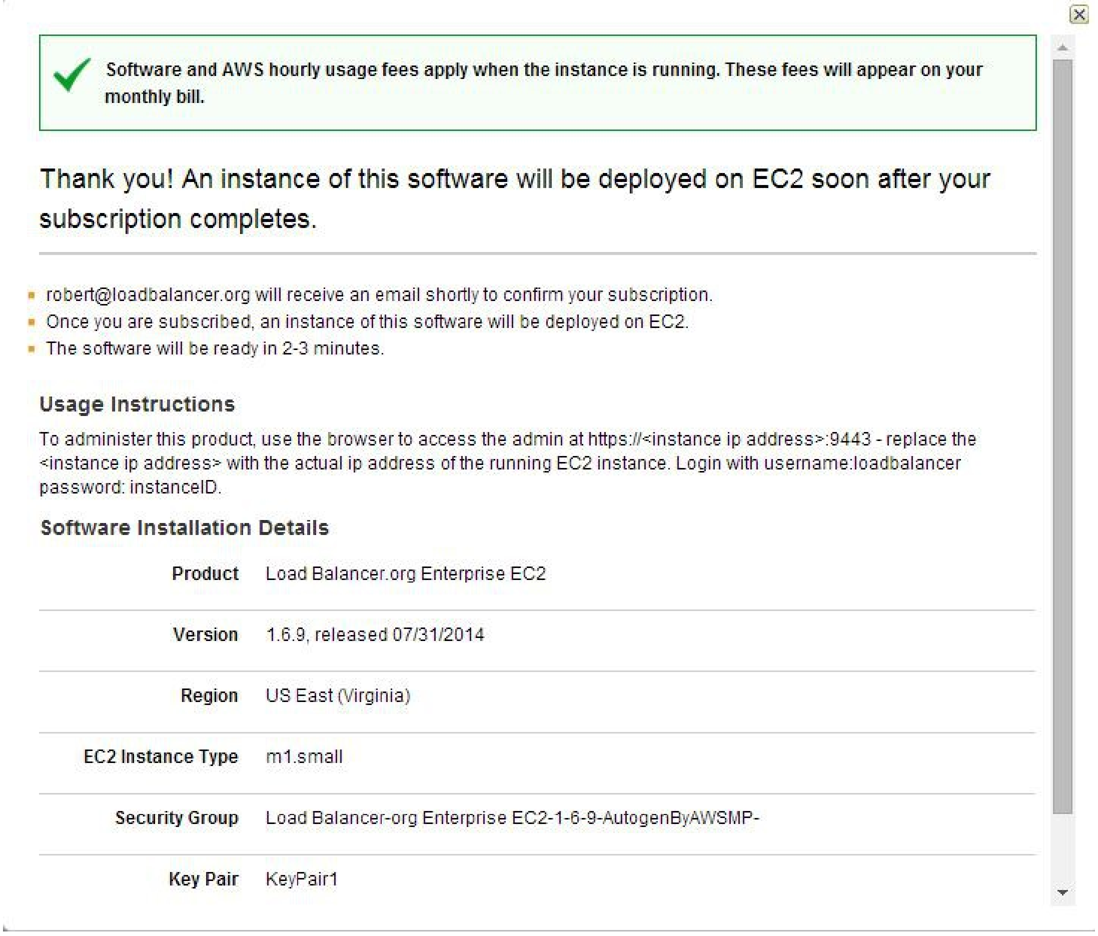
- Select the required **Security Group** – the default is an auto-created custom group that allows access on ports 80, 443, 7777 & 9443. Port 7777 is used for HAPproxy Layer 7 statistics, port 9443 is used to access the WUI using HTTPS. The group can be customized in the normal way using the AWS console

- Select the **Key Pair** – a drop-down list of valid key pairs is displayed

- Once the settings are correct, click the **Accept Terms & Launch with 1-Click** button

  *N.B. On subsequent deployments the button will be named **Launch with 1-Click***

- An instance will now be deployed in the region selected, the following confirmation message is displayed:



✓ Software and AWS hourly usage fees apply when the instance is running. These fees will appear on your monthly bill.

Thank you! An instance of this software will be deployed on EC2 soon after your subscription completes.

- robert@loadbalancer.org will receive an email shortly to confirm your subscription.
- Once you are subscribed, an instance of this software will be deployed on EC2.
- The software will be ready in 2-3 minutes.

Usage Instructions

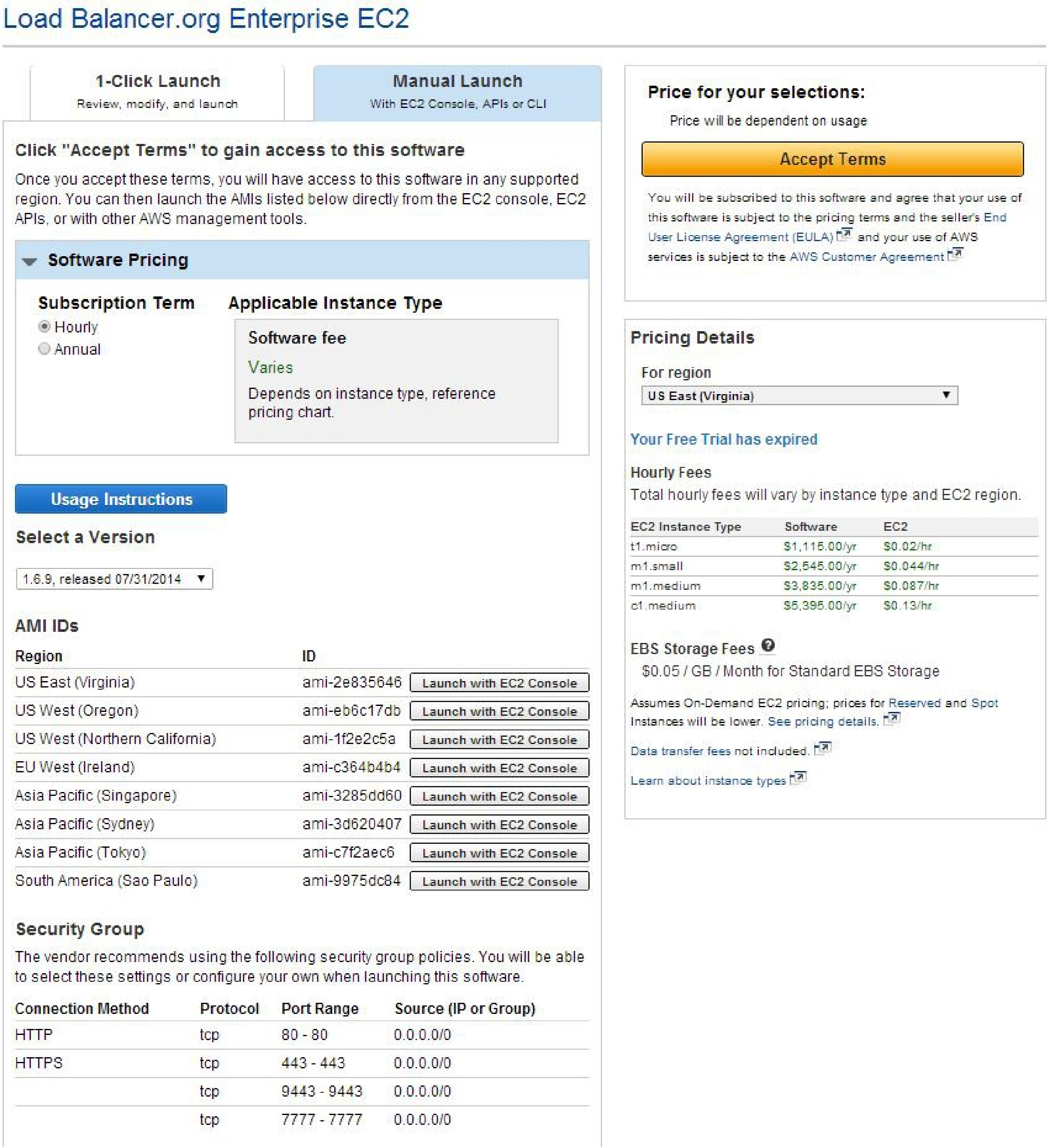
To administer this product, use the browser to access the admin at https://<instance ip address>:9443 - replace the <instance ip address> with the actual ip address of the running EC2 instance. Login with username:loadbalancer password: instanceID.

Software Installation Details

Product	Load Balancer.org Enterprise EC2
Version	1.6.9, released 07/31/2014
Region	US East (Virginia)
EC2 Instance Type	m1.small
Security Group	Load Balancer-org Enterprise EC2-1-6-9-AutogenByAWSMP-
Key Pair	KeyPair1

AWS MarketPlace Manual Launch

The option allows full access to all deployment options.



Load Balancer.org Enterprise EC2

Click "Accept Terms" to gain access to this software

Once you accept these terms, you will have access to this software in any supported region. You can then launch the AMIs listed below directly from the EC2 console, EC2 APIs, or with other AWS management tools.

Software Pricing

Subscription Term

● Hourly

○ Annual

Applicable Instance Type

Software fee

Varies

Depends on instance type, reference pricing chart.

Select a Version

1.6.9, released 07/31/2014

AMI IDs

Region	ID	
US East (Virginia)	ami-2e835646	Launch with EC2 Console
US West (Oregon)	ami-eb6c17db	Launch with EC2 Console
US West (Northern California)	ami-1f2e2c5a	Launch with EC2 Console
EU West (Ireland)	ami-c364b4b4	Launch with EC2 Console
Asia Pacific (Singapore)	ami-3285dd60	Launch with EC2 Console
Asia Pacific (Sydney)	ami-3d620407	Launch with EC2 Console
Asia Pacific (Tokyo)	ami-c7f2aec6	Launch with EC2 Console
South America (Sao Paulo)	ami-9975dc84	Launch with EC2 Console

Security Group

The vendor recommends using the following security group policies. You will be able to select these settings or configure your own when launching this software.

Connection Method	Protocol	Port Range	Source (IP or Group)
HTTP	tcp	80 - 80	0.0.0.0/0
HTTPS	tcp	443 - 443	0.0.0.0/0
	tcp	9443 - 9443	0.0.0.0/0
	tcp	7777 - 7777	0.0.0.0/0

Price for your selections:

Price will be dependent on usage

You will be subscribed to this software and agree that your use of this software is subject to the pricing terms and the seller's End User License Agreement (EULA) and your use of AWS services is subject to the AWS Customer Agreement

Pricing Details

For region

US East (Virginia)

Your Free Trial has expired

Hourly Fees

Total hourly fees will vary by instance type and EC2 region.

EC2 Instance Type	Software	EC2
t1.micro	$1,115.00/yr	$0.02/hr
m1.small	$2,545.00/yr	$0.044/hr
m1.medium	$3,835.00/yr	$0.087/hr
c1.medium	$5,395.00/yr	$0.13/hr

EBS Storage Fees

$0.05 / GB / Month for Standard EBS Storage

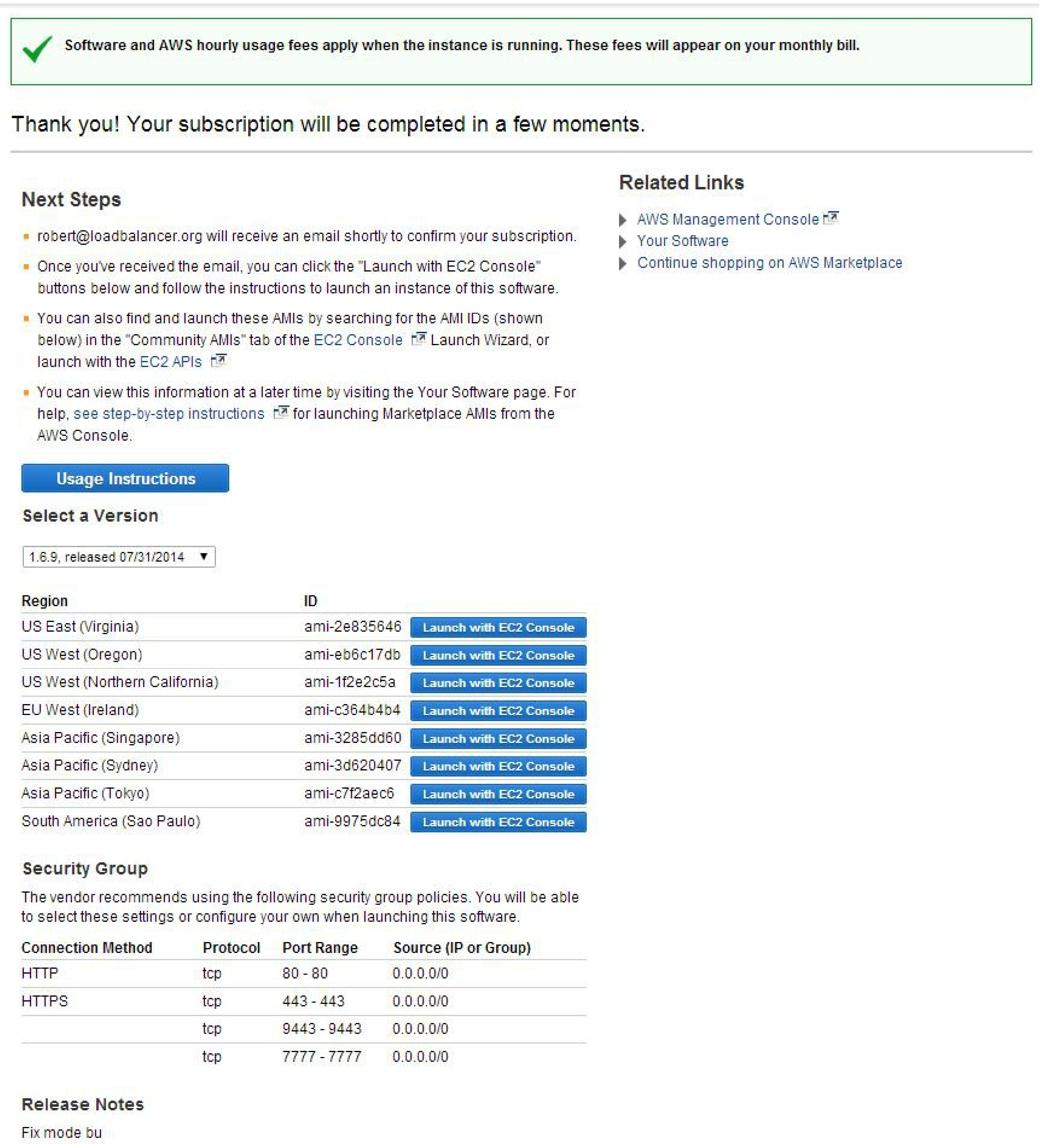
Assumes On-Demand EC2 pricing; prices for Reserved and Spot Instances will be lower. See pricing details.

Data transfer fees not included.

Learn about instance types

- Select the required **Subscription Term** (Hourly or Annual)
- Click the **Accept Terms** button

N.B. On subsequent deployments this step will not be required



✓ Software and AWS hourly usage fees apply when the instance is running. These fees will appear on your monthly bill.

Thank you! Your subscription will be completed in a few moments.

Next Steps

- robert@loadbalancer.org will receive an email shortly to confirm your subscription.
- Once you've received the email, you can click the "Launch with EC2 Console" buttons below and follow the instructions to launch an instance of this software.
- You can also find and launch these AMIs by searching for the AMI IDs (shown below) in the "Community AMIs" tab of the EC2 Console Launch Wizard, or launch with the EC2 APIs
- You can view this information at a later time by visiting the Your Software page. For help, see step-by-step instructions for launching Marketplace AMIs from the AWS Console.

Related Links

- AWS Management Console
- Your Software
- Continue shopping on AWS Marketplace

Select a Version

1.6.9, released 07/31/2014 ▼

Region	ID	
US East (Virginia)	ami-2e835646	Launch with EC2 Console
US West (Oregon)	ami-eb6c17db	Launch with EC2 Console
US West (Northern California)	ami-1f2e2c5a	Launch with EC2 Console
EU West (Ireland)	ami-c364b4b4	Launch with EC2 Console
Asia Pacific (Singapore)	ami-3285dd60	Launch with EC2 Console
Asia Pacific (Sydney)	ami-3d620407	Launch with EC2 Console
Asia Pacific (Tokyo)	ami-c7f2aec6	Launch with EC2 Console
South America (Sao Paulo)	ami-9975dc84	Launch with EC2 Console

Security Group

The vendor recommends using the following security group policies. You will be able to select these settings or configure your own when launching this software.

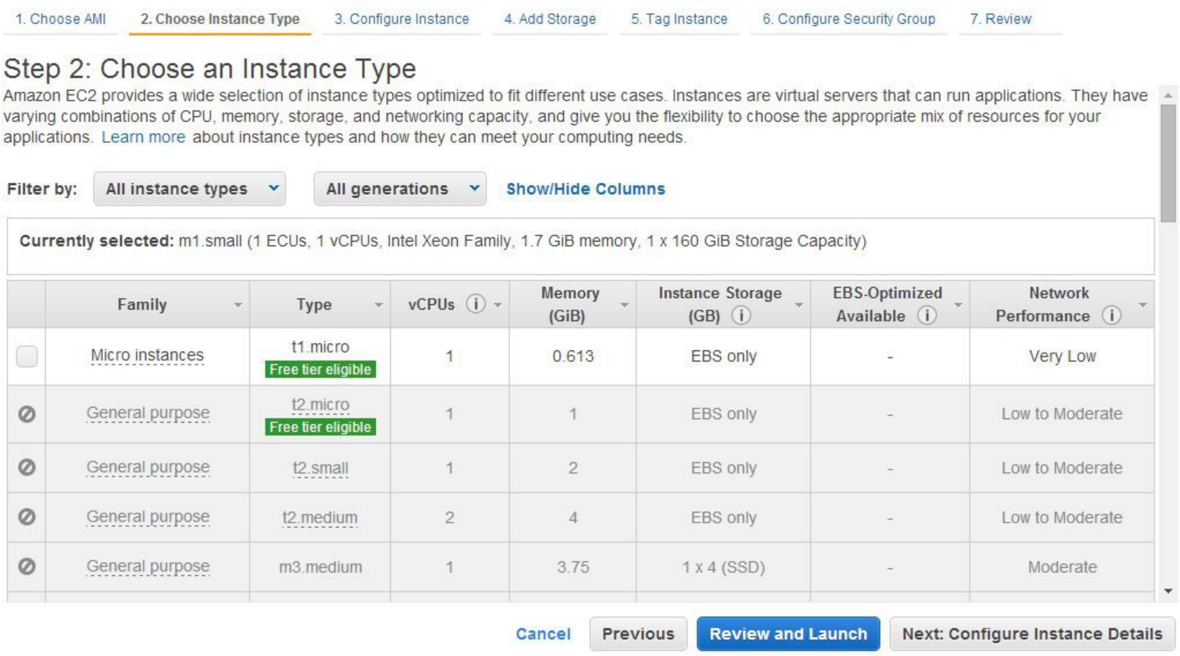
Connection Method	Protocol	Port Range	Source (IP or Group)
HTTP	tcp	80 - 80	0.0.0.0/0
HTTPS	tcp	443 - 443	0.0.0.0/0
	tcp	9443 - 9443	0.0.0.0/0
	tcp	7777 - 7777	0.0.0.0/0

Release Notes

Fix mode bu

- Select the **Version** – the latest version is always recommended
- Click the **Launch with EC2 Console** button next to the required region

N.B. If the launch buttons are displayed grey, hit the refresh button after the AWS confirmation email has been received. They should then change to blue and be click-able



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.

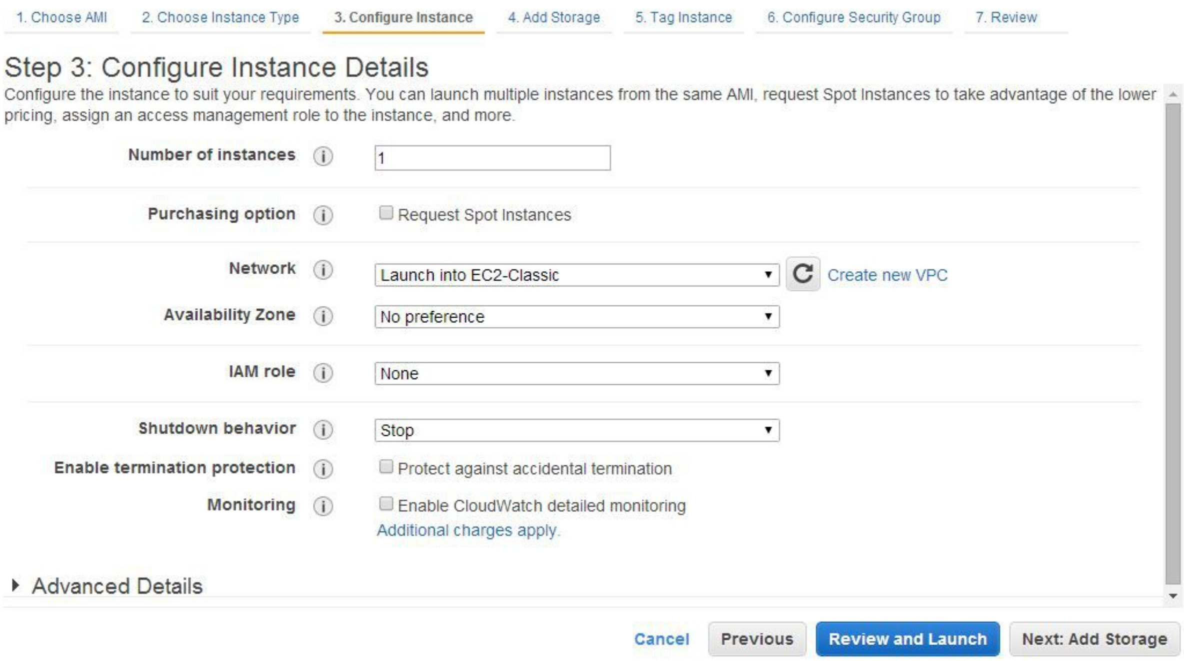
Filter by: All instance types | All generations | Show/Hide Columns

Currently selected: m1.small (1 ECUs, 1 vCPUs, Intel Xeon Family, 1.7 GiB memory, 1 x 160 GiB Storage Capacity)

	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.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	m3.medium	1	3.75	1 x 4 (SSD)	-	Moderate

- Filter by **All instance types** & **All generations**, then select the required instance type (*general purpose, m1.small* is our general recommendation, although this does depend on how the load balancer will be used)

- Click **Next: Configure Instance Details**



Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot Instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

Number of instances: 1

Purchasing option: □ Request Spot Instances

Network: Launch into EC2-Classic | Create new VPC

Availability Zone: No preference

IAM role: None

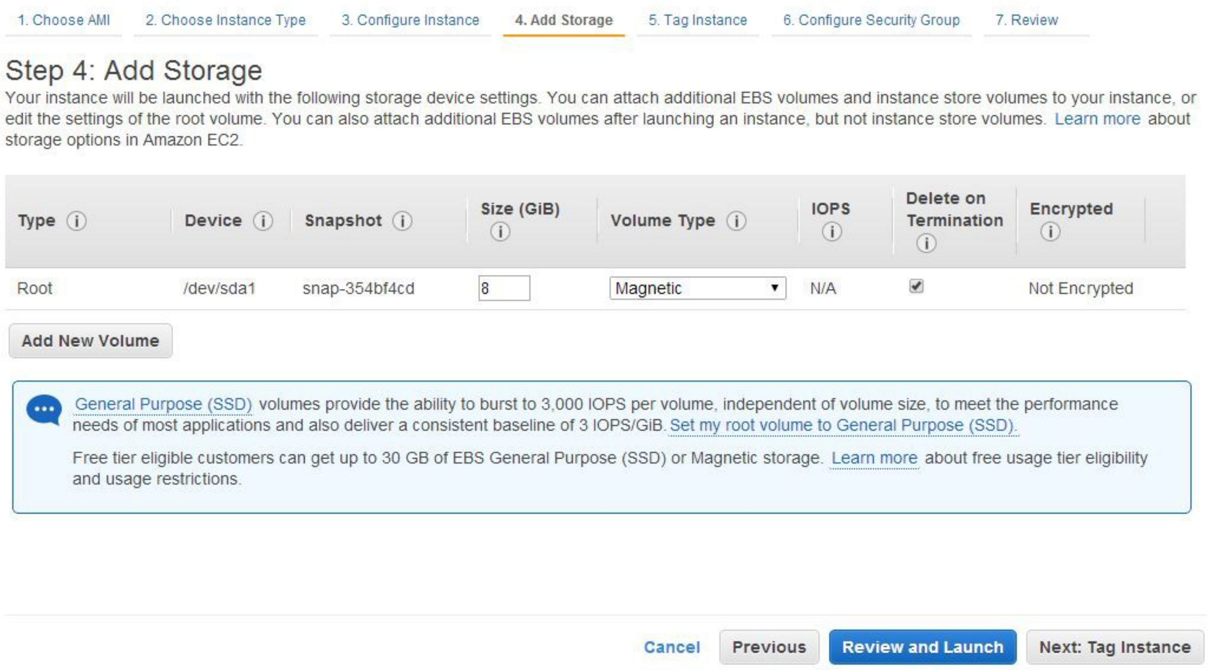
Shutdown behavior: Stop

Enable termination protection: □ Protect against accidental termination

Monitoring: □ Enable CloudWatch detailed monitoring
Additional charges apply.

Advanced Details

- Configure the required options and click **Next: Add Storage**



Step 4: Add Storage

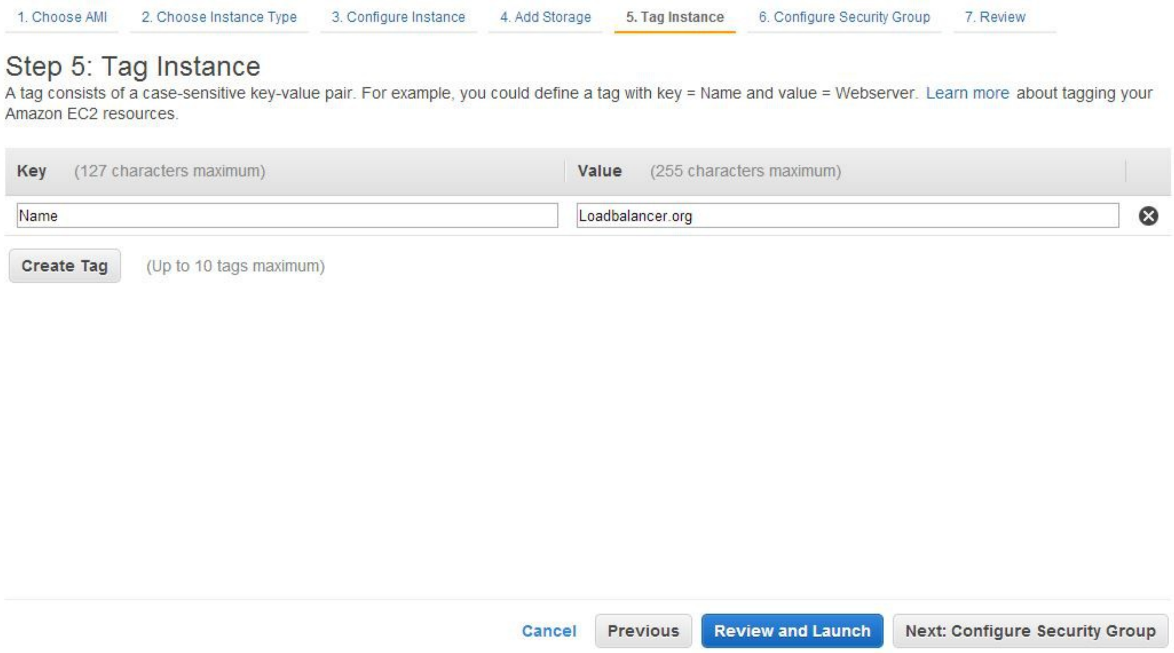
Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. Learn more about storage options in Amazon EC2.

Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Delete on Termination	Encrypted
Root	/dev/sda1	snap-354bf4cd	8	Magnetic	N/A	☑	Not Encrypted

General Purpose (SSD) volumes provide the ability to burst to 3,000 IOPS per volume, independent of volume size, to meet the performance needs of most applications and also deliver a consistent baseline of 3 IOPS/GiB. Set my root volume to General Purpose (SSD).

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. Learn more about free usage tier eligibility and usage restrictions.

- Configure the required options and click **Next: Tag Instance**



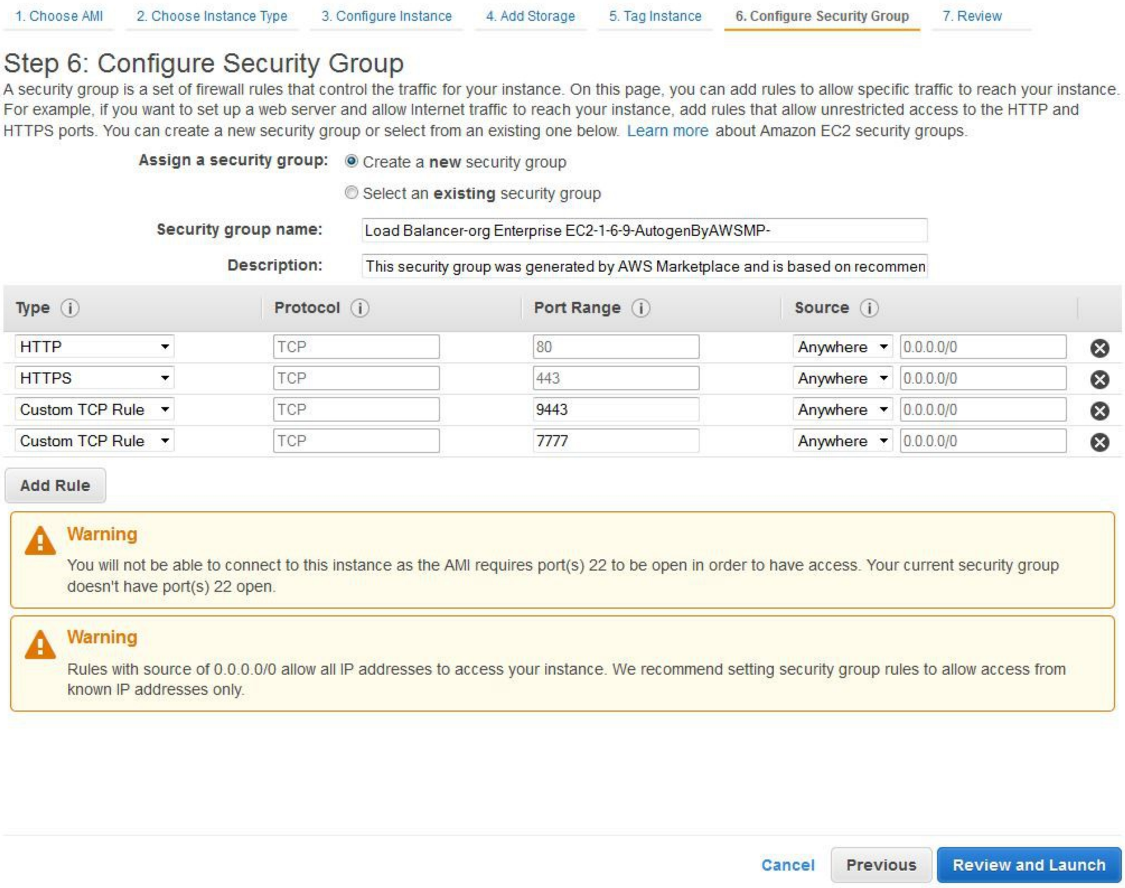
Step 5: Tag Instance

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver. Learn more about tagging your Amazon EC2 resources.

Key (127 characters maximum)	Value (255 characters maximum)
Name	Loadbalancer.org

(Up to 10 tags maximum)

- Configure the required options and click **Next: Configure Security Group**



Step 6: Configure Security Group

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 name: Load Balancer-org Enterprise EC2-1-6-9-AutogenByAWSMP-

Description: This security group was generated by AWS Marketplace and is based on recommen

Type	Protocol	Port Range	Source	
HTTP	TCP	80	Anywhere	0.0.0.0/0
HTTPS	TCP	443	Anywhere	0.0.0.0/0
Custom TCP Rule	TCP	9443	Anywhere	0.0.0.0/0
Custom TCP Rule	TCP	7777	Anywhere	0.0.0.0/0

Warning

You will not be able to connect to this instance as the AMI requires port(s) 22 to be open in order to have access. Your current security group doesn't have port(s) 22 open.

Warning

Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

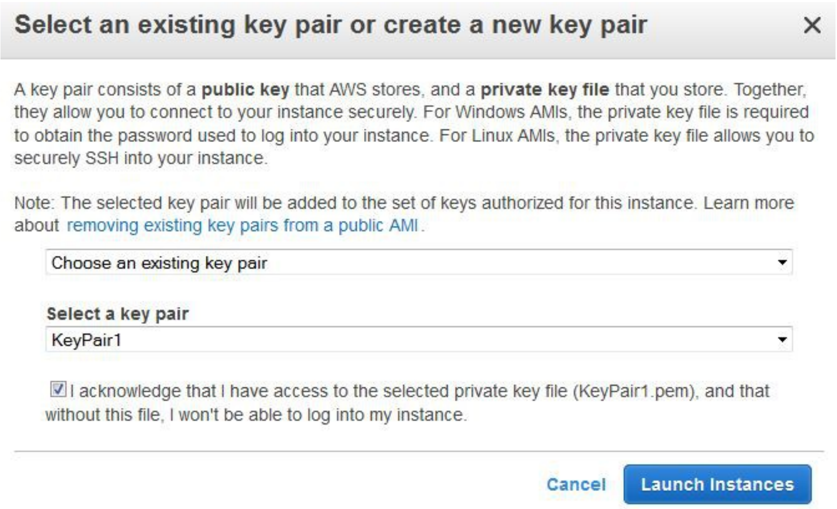
- By default, 4 rules are automatically created. Review and edit these if required then click **Review and Launch**

  N.B. If you are not load balancing HTTP traffic, change the ports accordingly. e.g. for RDP you'll need to specify port 3389 rather than 80 & 443. 7777 & 9443 should always be included to permit access to the load balancers management and monitoring interface. If SSH access is also needed, add TCP port 22 too.

- If prompted, select the required SSD option

- Now review/edit the Instance Launch Details and click **Launch** to start the instance

- Now choose an existing or create a new key pair



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
KeyPair1

☑ I acknowledge that I have access to the selected private key file (KeyPair1.pem), and that without this file, I won't be able to log into my instance.

- If creating a new pair use the Download Key Pair button to save the private key

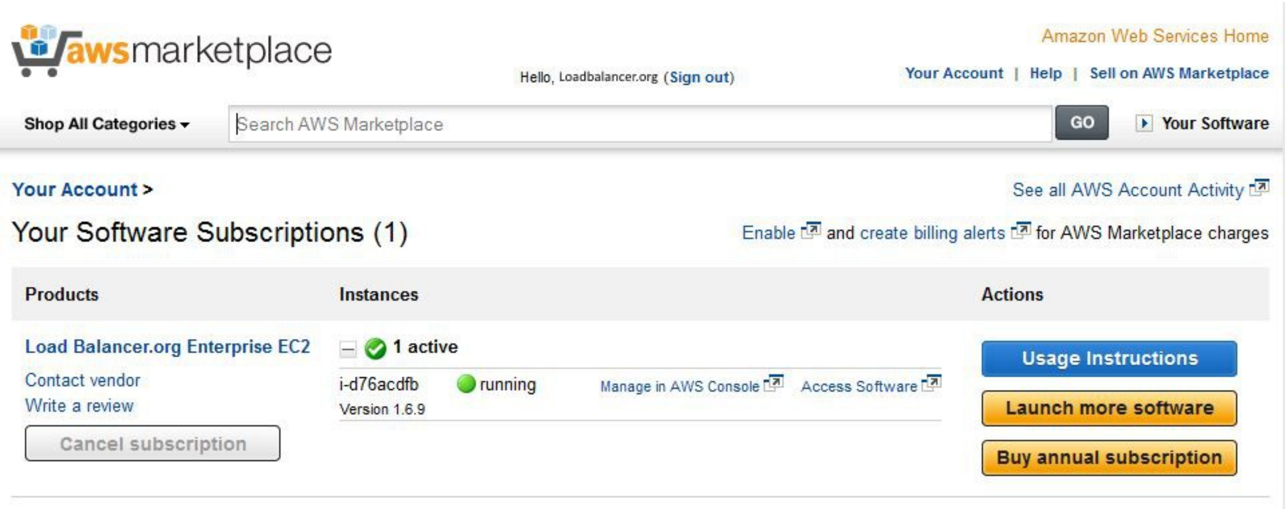
  N.B. This private key is used for secure access to the load balancer instance via SSH once its up and running. It's not used for SSL termination. For this please refer to the SSL Termination section later in this guide.

- If using an existing key pair, tick the acknowledgment check-box

- Finally, click the **Launch Instances** button

Checking your Subscriptions

Current subscriptions can be viewed and canceled using the *Your Account* > *Your Software* > *Manage your Software Subscriptions* option in the awsmarketplace console as shown below:



awsmarketplace
Amazon Web Services Home
Hello, Loadbalancer.org (Sign out)
Your Account | Help | Sell on AWS Marketplace
Shop All Categories
Search AWS Marketplace
GO
Your Software
Your Account >
See all AWS Account Activity
Enable and create billing alerts for AWS Marketplace charges

Your Software Subscriptions (1)

Products	Instances				Actions
Load Balancer.org Enterprise EC2 Contact vendor Write a review Cancel subscription	1 active				Usage Instructions Launch more software Buy annual subscription
	i-d76acdfb Version 1.6.9	running	Manage in AWS Console	Access Software	

Accessing the WUI

In a browser, navigate to the Public DNS name, Public IP address or the Elastic IP address on port 9443

i.e.

https://<Public DNS name>:9443

or

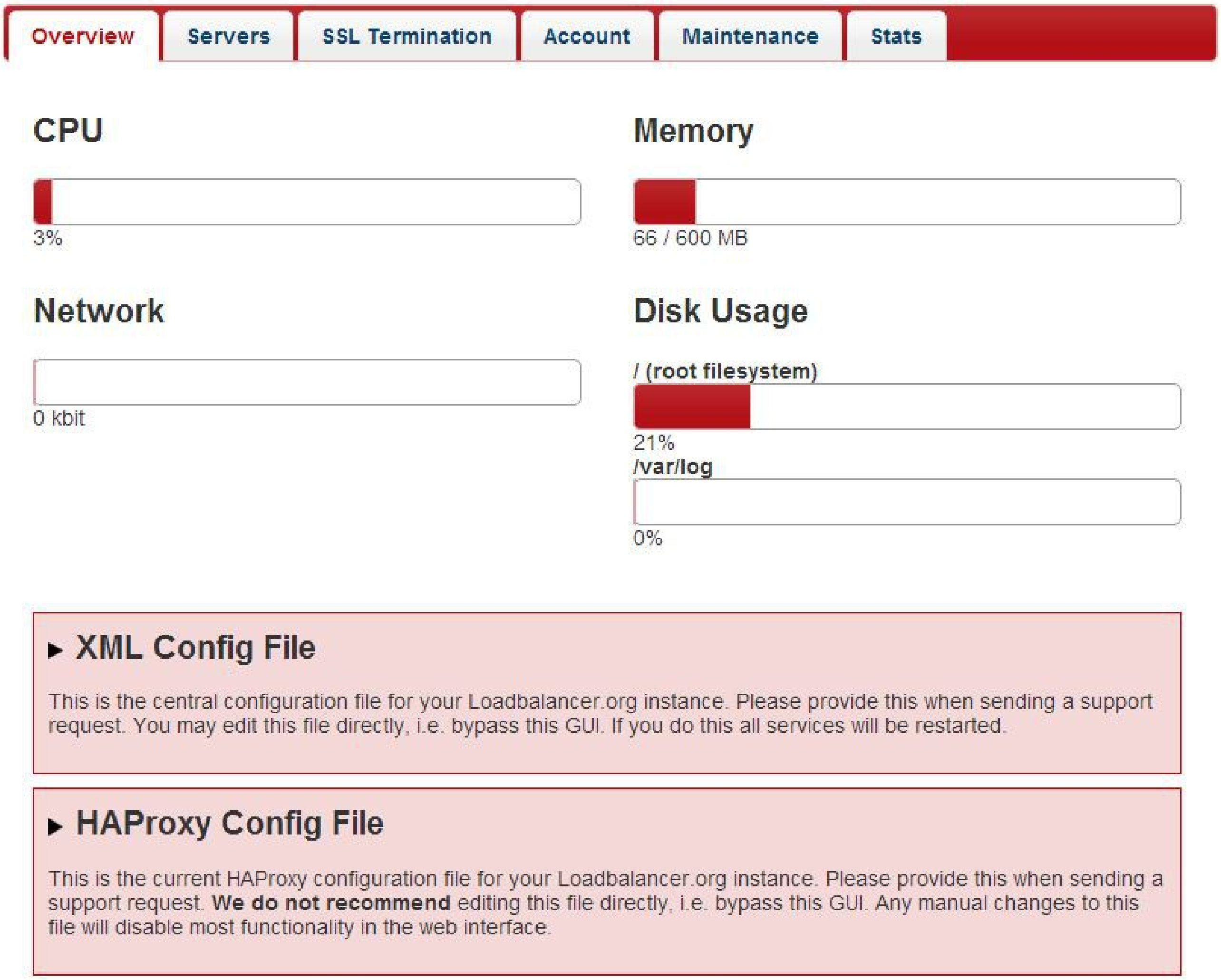
https://<EIP or **Public IP address>:9443**

You'll receive a warning about the certificate as it's a self signed cert not related to an Internet based CA. Confirm you want to continue and a login prompt will be displayed. Use the following default credentials:

Username: *loadbalancer*

Password: *<EC2 Instance-ID>*

Once logged in, the following screen is displayed:



▸ XML Config File

This is the central configuration file for your Loadbalancer.org instance. Please provide this when sending a support request. You may edit this file directly, i.e. bypass this GUI. If you do this all services will be restarted.

▸ HAProxy Config File

This is the current HAProxy configuration file for your Loadbalancer.org instance. Please provide this when sending a support request. **We do not recommend** editing this file directly, i.e. bypass this GUI. Any manual changes to this file will disable most functionality in the web interface.

Using the WUI

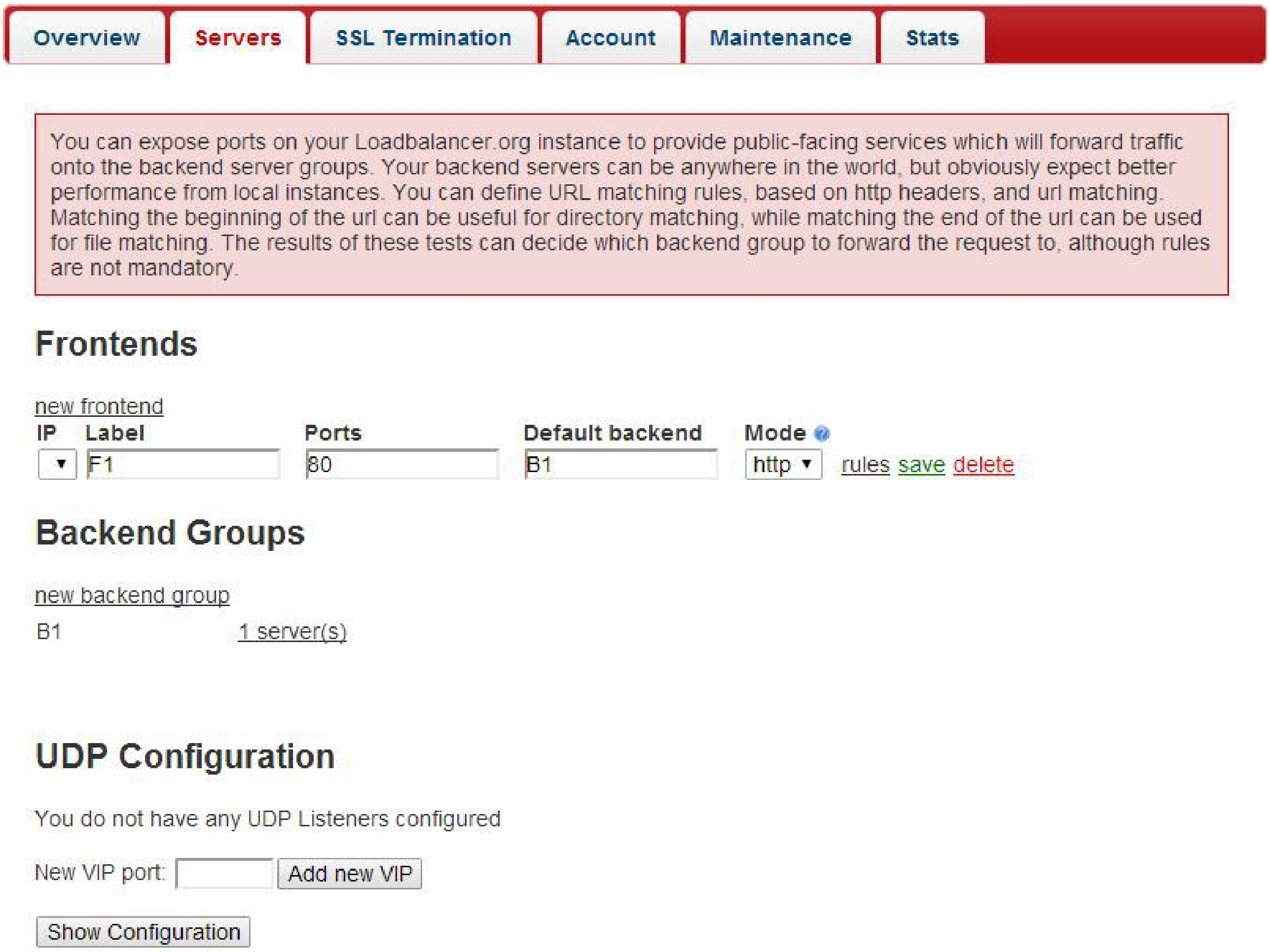
Overview

This tab shows the basic performance stats for the instance as well as the XML and HAProxy config files. It's possible to modify these files directly, but it's recommended to allow the interface to handle the configuration file changes to ensure syntax rules are followed.

Servers

This is where you specify the Front-ends and Back-end groups (used for TCP) and also the UDP configuration. A default front-end (F1) and back-end group (B1) is included as shown below:

- Front-ends are where clients connect to, Back-end groups are where the actual load balanced servers are defined.
- Each Front-end has a default back-end group
- Each Back-end group can be used by multiple Front-ends



You can expose ports on your Loadbalancer.org instance to provide public-facing services which will forward traffic onto the backend server groups. Your backend servers can be anywhere in the world, but obviously expect better performance from local instances. You can define URL matching rules, based on http headers, and url matching. Matching the beginning of the url can be useful for directory matching, while matching the end of the url can be used for file matching. The results of these tests can decide which backend group to forward the request to, although rules are not mandatory.

Frontends

new frontend

IP	Label	Ports	Default backend	Mode	
▾	F1	80	B1	http ▾	rules save delete

Backend Groups

new backend group

B1 1 server(s)

UDP Configuration

You do not have any UDP Listeners configured

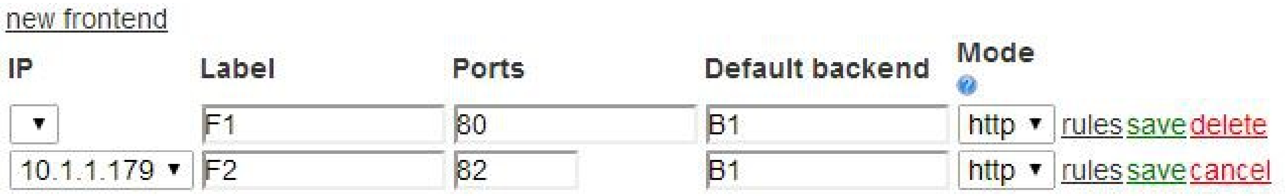
New VIP port: [ ] Add new VIP

Show Configuration

N.B. servers can be specified using their DNS name or by IP address

Creating a new TCP Configuration

A new Front-end can be created by clicking the **new frontend** link. Once clicked, the front-end can be defined as shown in the example below:



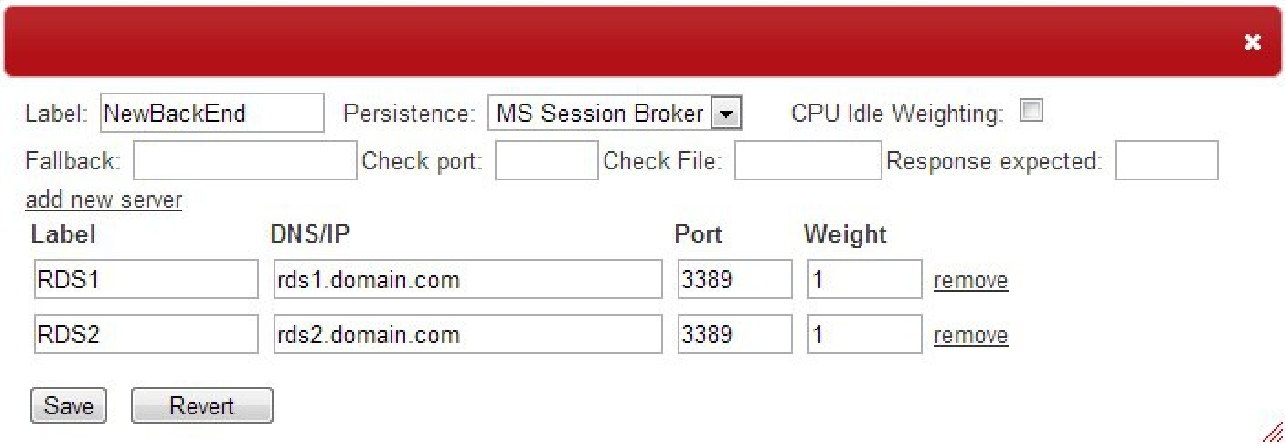
new frontend

IP	Label	Ports	Default backend	Mode	
▾	F1	80	B1	http ▾	rules save delete
10.1.1.179 ▾	F2	82	B1	http ▾	rules save cancel

- the **IP** drop-down enables all allocated IP's *(all) or individual addresses to be selected

NOTE: To assign additional IP addresses to the appliance use the EC2 Management Console. Then use the **Restart Networking** option in the Services area of the *Maintenance* Tab to restart the network. Once restarted, the additional IP addresses will be available when defining new Front-ends. For more details on IP addresses please refer to page 7 earlier in this guide.

If required a new Back-end group can be created by clicking the **new backend group** link. Once clicked, the group can be defined as shown in the example below.



Label: NewBackEnd Persistence: MS Session Broker ▾ CPU Idle Weighting: □

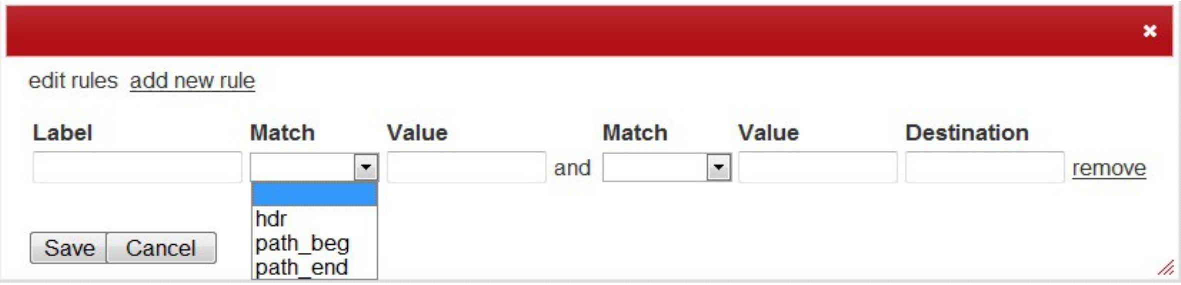
Fallback: Check port: Check File: Response expected:

add new server

Label	DNS/IP	Port	Weight	
RDS1	rds1.domain.com	3389	1	remove
RDS2	rds2.domain.com	3389	1	remove

Configuring Server Rules

It's possible to customize the way requests are handled. Rules can be added that examine the headers, the start of a URL path or the end of a URL path. For example, a 'path_end' rule could be added that sends requests that end in /blog to a different backend server. To configure rules click the 'rules' link next to the relevant Frontend.

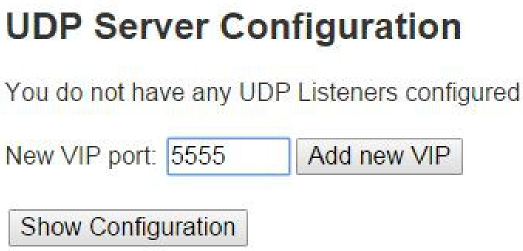


edit rules add new rule

Label	Match	Value		Match	Value	Destination	
	▾		and	▾			remove

Creating a new UDP Configuration

A new UDP listener can be created by specifying the required UDP port and clicking the **Add new VIP** button. First, enter the UDP port required for the listener then click **Add new VIP**, in the example below UDP port 5555 is specified:

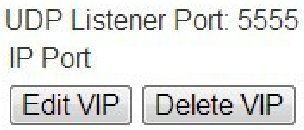


UDP Server Configuration

You do not have any UDP Listeners configured

New VIP port: 5555

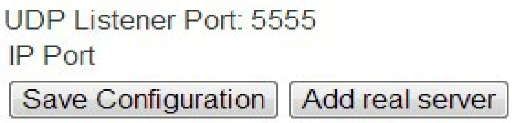
Once the **Add new VIP** button is clicked, the new listener is created and can then be edited or deleted using the buttons shown below:



UDP Listener Port: 5555

IP Port

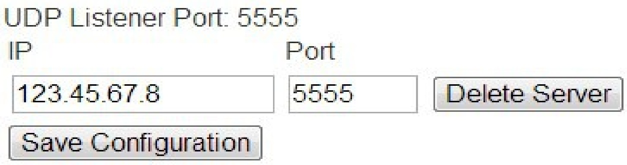
To add Real Servers (i.e. back-end servers), click **Edit VIP**, the following options will then be displayed:



UDP Listener Port: 5555

IP Port

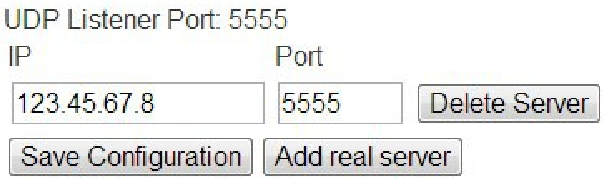
Click **Add real server** to be able to specify the first real server



UDP Listener Port: 5555

IP	Port
123.45.67.8	5555

Once defined, click **Save Configuration.** Once the first real server is added, additional **Add real server** buttons will be displayed which enable additional real servers to be added as shown below:



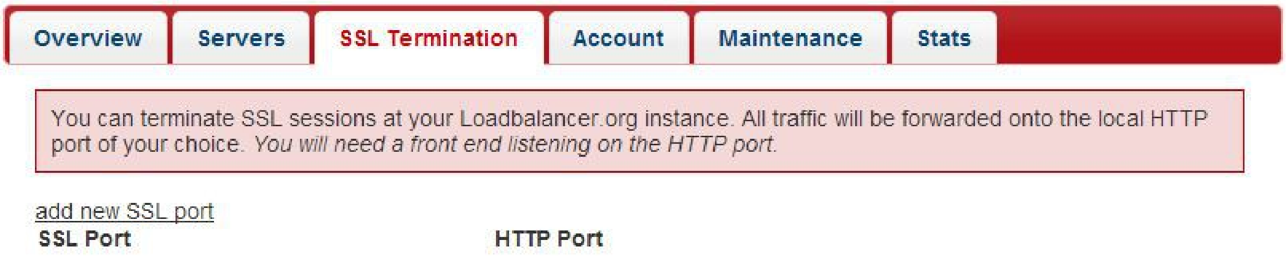
UDP Listener Port: 5555

IP	Port
123.45.67.8	5555

Once all real servers are added, click **Save Configuration.**

SSL Termination

SSL can be terminated on the load balancer. A default self-signed certificate is provided, although normally this will be replaced with your own certificate using the Upload Certificate option in the interface.

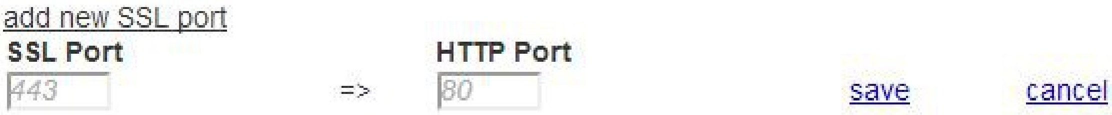


You can terminate SSL sessions at your Loadbalancer.org instance. All traffic will be forwarded onto the local HTTP port of your choice. *You will need a front end listening on the HTTP port.*

SSL Port	HTTP Port

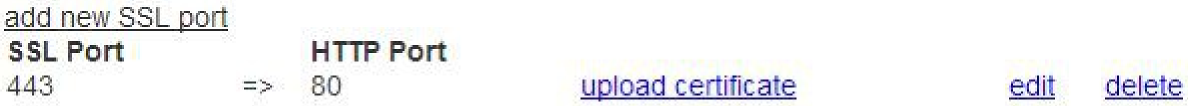
Setting up SSL Termination

To setup SSL click **add new SSL port**, this will enable the HTTPS port (typically 443) and the backend HTTP port (typically 80) to be defined as shown below:



SSL Port		HTTP Port		
443	=>	80	save	cancel

Once the required port have been defined click **save**



SSL Port		HTTP Port			
443	=>	80	upload certificate	edit	delete

Example Certificate Configuration using a VeriSign (Symantec) Test Certificate

Symantec offer a 30 day trial which can be used with the load balancer. The following steps cover the complete process from generating the CSR to installing the certificate.

1) **Connect to the Console of the load balancer** – refer to the section '*Accessing the Load Balancer using SSH*' on page 32 for details on how to do this with Linux and Windows.

2) **Generate the Private Key & set permissions**

```
mkdir certs
openssl genrsa -out ./certs/lb.key 1024
chmod 400 ./certs/lb.key
```

3) **Generate the CSR**

```
openssl req -new -nodes -key ./certs/lb.key -out ./certs/lb.csr
```

the following section shows the various prompts and sample answers:

```
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [GB]:GB
State or Province Name (full name) [Berkshire]:Hampshire
Locality Name (eg, city) [Newbury]:Portsmouth
Organization Name (eg, company) [My Company Ltd]:loadbalancer.org
Organizational Unit Name (eg, section) []:support
Common Name (eg, your name or your server's hostname) []:www.loadbalancer.org
Email Address []:support@loadbalancer.org
Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
```

===> *The file **lb.csr** has now been created on the load balancer in the directory **./certs/***

N.B. When prompted for the x509 common name, make sure you enter the fully qualified hostname the certificate will be used with. e.g. www.loadbalancer.org

4) **Copy the Private Key & CSR to your workstation**

Under Linux:

(replace lbkeypair1.pem with the name of your private key file)

```
scp -i lbkeypair1.pem ec2-user@<IP address>:/remote-path/lb.key /local-path/lb.key
scp -i lbkeypair1.pem ec2-user@<IP address>:/remote-path/lb.csr /local-path/lb.csr
```

Under Windows:

- Connect to the load balancer as ec2-user using WinSCP as described on page 35
- Copy the files ./certs/lb.key & ./certs/lb.csr to your workstation

5) **Create your Symantec Test Certificate**

- Open the following URL and click **Try it Free** : https://www.symantec.com/en/uk/ssl-certificates
- Copy the contents of lb.csr to the first CSR screen, select **Server not listed** and click **Continue**
- When prompted for a challenge phrase and reminder question use only simple letters and numbers
- Symantec will then email your new test certificate - normally within a few minutes

6) **Installing the Certificate on the Load Balancer**

- Under the SSL port created earlier click **upload certificate**
- Delete the contents of the top & bottom pane (these are for the default self signed cert)
- Copy/paste the private key to the top pane & the certificate obtained from Symantec to the bottom pane. You should also add Symantec's trial intermediate certificate (a link is provided in the email from Symantec) in the bottom pane after the certificate as follows:

```
-----BEGIN CERTIFICATE-----
 certificate contents goes here
------END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
 intermediate certificate contents go here
------END CERTIFICATE-----
```

e.g.

```
-----BEGIN CERTIFICATE-----
MIICJTCCAY4CAQAwgaoxCzAJBgNVBAYTAkdCMRIwEAYDVQQIEwlIYW1wc2hpcmUx
EzARBgNVBAcTClBvcnRzbW91dGgxGTAXBgNVBAoTEExvYWRiYWxhbmNlci5vcmcx
MSYwJAYJKoZIhvcNAQkBFhdyb2JlcnRAbG9hZGJhbGFuY2VyLm9yZzCBnzANBgkq
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
MIICJTCCAY4CAQAwgaoxCzAJBgNVBAYTAkdCMRIwEAYDVQQIEwlIYW1wc2hpcmUx
EzARBgNVBAcTClBvcnRzbW91dGgxGTAXBgNVBAoTEExvYWRiYWxhbmNlci5vcmcx
EDAOBgNVBAsTB1N1cHBvcnQxHTAbBgNVBAMTFHd3dy5sb2FkYmFsYW5jZXIub3Jn
-----END CERTIFICATE-----
```

- Click **Save**
- Now restart Pound using the pop-out message or *Maintenance* > *Services* and click **Restart Pound**

Using your Windows IIS Certificate

For Windows, its often easiest to get the certificate working on the server first. The certificate can then be exported from Windows in .pfx format, then converted to .pem format and copied to the load balancer. The steps for this process are:

1) **Export the certificate from IIS** - once the certificate is working correctly on your Windows server, export the certificate from Windows – including the private key. Make sure you select the option for all certs in the chain, the format must be .pfx

(for the common name, make sure you enter to fully qualified domain name for your web server, e.g. www.loadbalancer.org)

2) **Download & install Openssl** - download openssl using the following link & install on your PC: http://www.slproweb.com/products/Win32OpenSSL.html

(Select the latest full download rather than the 'light' version)

3) **Extract the Private Key** – in a command window, type the following:

```
\openssl\bin\openssl pkcs12 -in drive:\path\cert.pfx -nocerts -out drive:\path\cert.pk
```

(You'll be prompted for the password used to create the pfx file, and a passphrase to write the output file)

4) **Unencrypt the Private Key** – in a command window, type the following:

```
\openssl\bin\openssl rsa -in drive:\path\cert.pk -out drive:\path\cert.pkun
```

(You'll be prompted for the passphrase that you entered in the previous step)

5) **Extract the Certificate** – in a command window type the following:

```
\openssl\bin\openssl pkcs12 -in drive:\path\cert.pfx -clcerts -nokeys -out drive:\path\cert.cer
```

(You'll be prompted for the password used to create the pfx file)

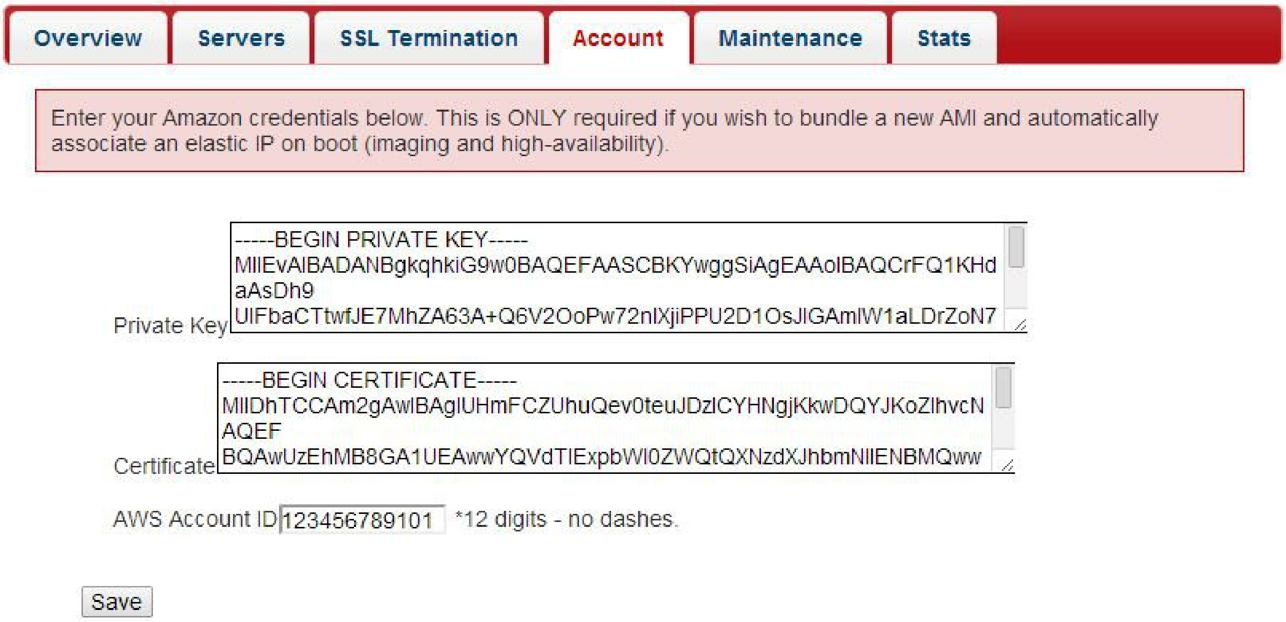
6) **Copy the Private Key (cert.pkun) and Certificate (cert.cer) to the Load Balancer**

- go to SSL Termination
- click **upload certificate**
- paste the contents of the private key into the top pane and the certificate into the lower pane
- click **Save**

7) **Now Restart Pound -** (*Maintenance > Services)*

Account

Enter the relevant Amazon credentials here to enable auto-scaling. The Certificate can be created and downloaded in the AWS Management Console under *My Account > Security Credentials > Access Credentials > X.509 Certificates.* Select **Create a new Certificate**, the will enable you to download both the certificate and the private key. The AWS account number is displayed under the Sign-Out option on the AWS console. See page 29 for details on setting up Auto-scaling.



Enter your Amazon credentials below. This is ONLY required if you wish to bundle a new AMI and automatically associate an elastic IP on boot (imaging and high-availability).

Private Key

```
-----BEGIN PRIVATE KEY-----
MIIEvAIBADANBgkqhkiG9w0BAQEFAASCBKYwggSiAgEAAoIBAQCrFQ1KHd
aAsDh9
UIFbaCTtwfJE7MhZA63A+Q6V2OoPw72nlXjiPPU2D1OsJIGAmIW1aLDrZoN7
```

Certificate

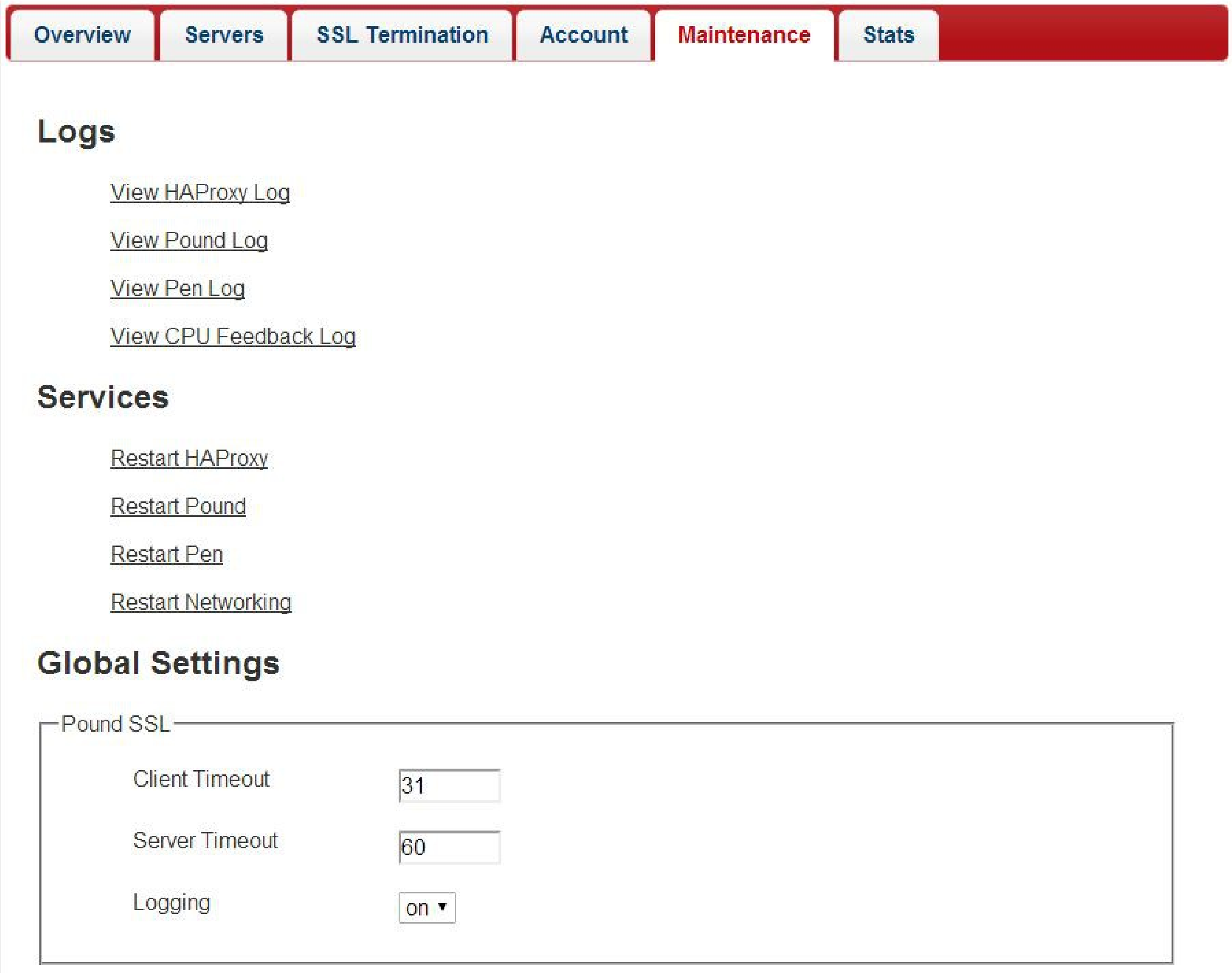
```
-----BEGIN CERTIFICATE-----
MIIDhTCCAm2gAwIBAgIUHmFCZUhuQev0teuJDzlCYHNgjKkwDQYJKoZIhvcN
AQEF
BQAwUzEhMB8GA1UEAwwYQVdTIExpbWl0ZWQtQXNzdXJhbmNlIENBMQww
```

AWS Account ID 123456789101 *12 digits - no dashes.

Once entered, click the **Save** button to validate & save these details.

Maintenance

This section allows logs to be viewed, services to be restated, global settings to be changed (N.B. for most applications the global settings can be left at their default values), the WUI password to be changed and a number of other administration related tasks as shown below:



Logs

- View HAProxy Log
- View Pound Log
- View Pen Log
- View CPU Feedback Log

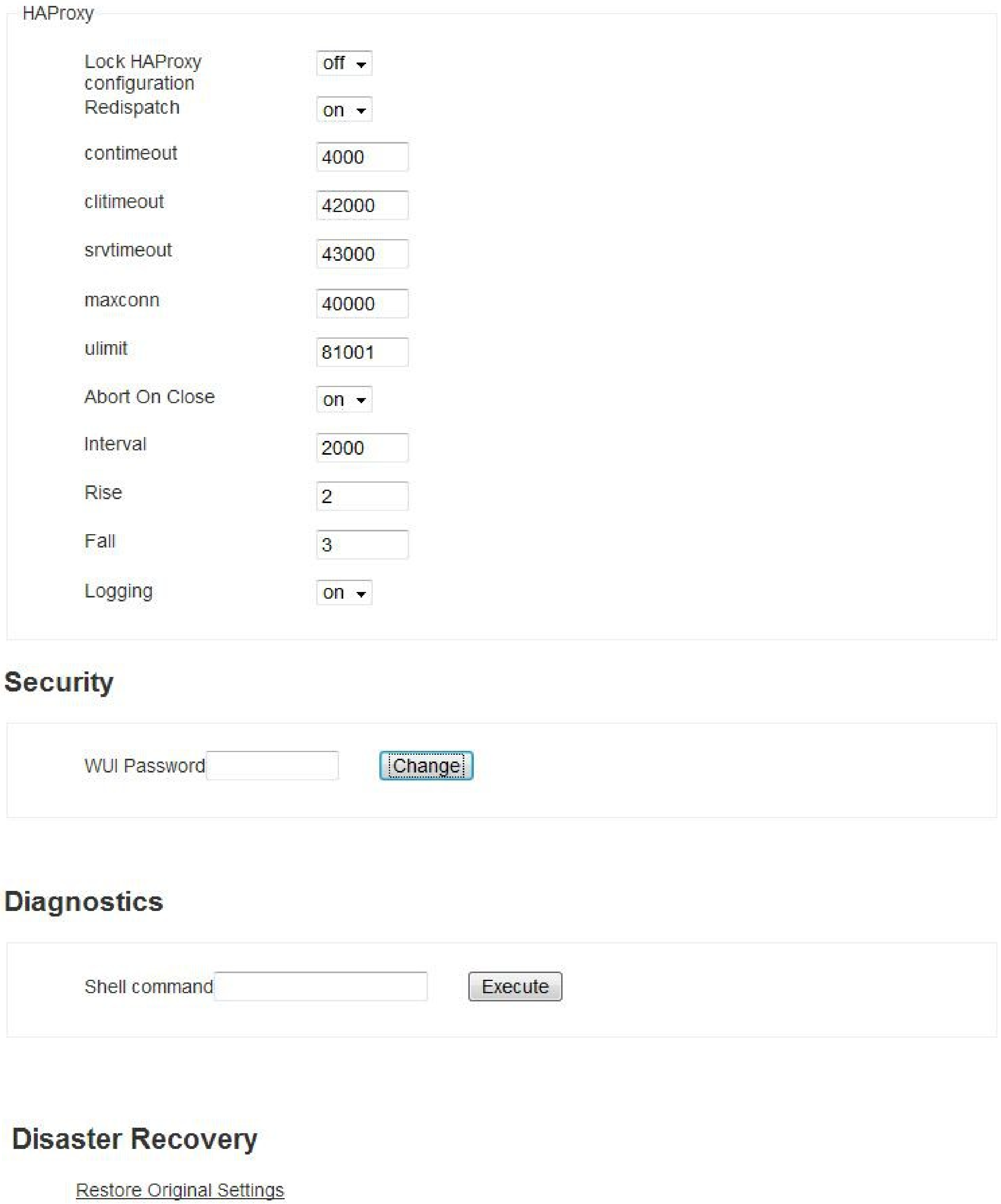
Services

- Restart HAProxy
- Restart Pound
- Restart Pen
- Restart Networking

Global Settings

Pound SSL

Client Timeout	31
Server Timeout	60
Logging	on ▾



HAProxy

Lock HAProxy configuration	off
Redispatch	on
contimeout	4000
clitimeout	42000
srvtimeout	43000
maxconn	40000
ulimit	81001
Abort On Close	on
Interval	2000
Rise	2
Fall	3
Logging	on

Security

WUI Password

Diagnostics

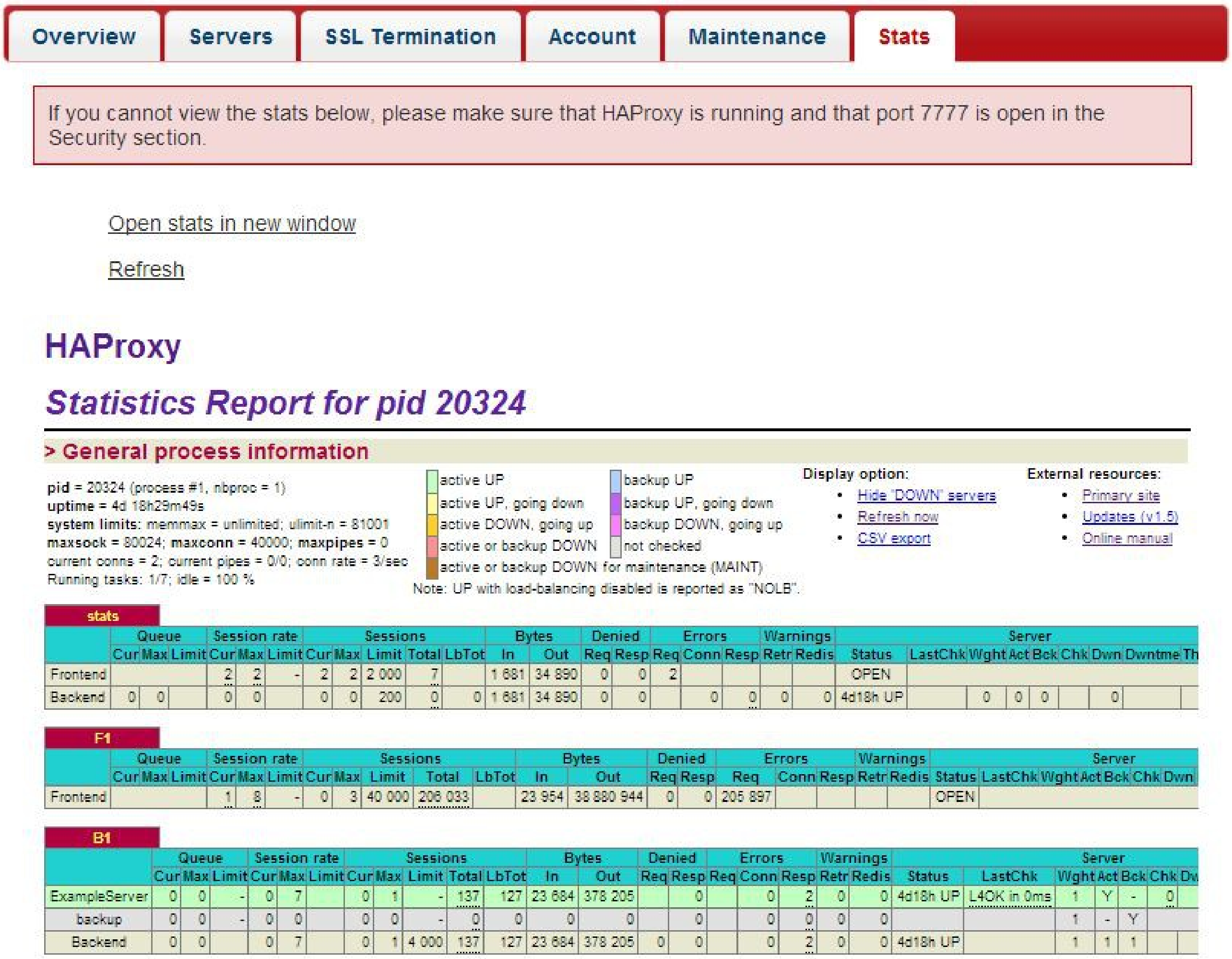
Shell command

Disaster Recovery

Restore Original Settings

Stats

This displays HAProxy statistics. A separate section is created for each Front-end & each Back-end server.



If you cannot view the stats below, please make sure that HAProxy is running and that port 7777 is open in the Security section.

Open stats in new window

Refresh

HAProxy

Statistics Report for pid 20324

> General process information

pid = 20324 (process #1, nbproc = 1)
uptime = 4d 18h29m49s
system limits: memmax = unlimited; ulimit-n = 81001
maxsock = 80024; **maxconn** = 40000; **maxpipes** = 0
current conns = 2; current pipes = 0/0; conn rate = 3/sec
Running tasks: 1/7; idle = 100 %

🟩 active UP	🟦 backup UP
🟨 active UP, going down	🟪 backup UP, going down
🟧 active DOWN, going up	🟪 backup DOWN, going up
🟥 active or backup DOWN	⬜ not checked
🟫 active or backup DOWN for maintenance (MAINT)	

Note: UP with load-balancing disabled is reported as "NOLB".

Display option:

- Hide 'DOWN' servers
- Refresh now
- CSV export

External resources:

- Primary site
- Updates (v1.5)
- Online manual

stats

	Queue			Session rate			Sessions					Bytes		Denied		Errors			Warnings		Server								
	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	In	Out	Req	Resp	Req	Conn	Resp	Retr	Redis	Status	LastChk	Wght	Act	Bck	Chk	Dwn	Dwntme	Th
Frontend				2	2	-	2	2	2 000	7		1 681	34 890	0	0	2					OPEN								
Backend	0	0		0	0		0	0	200	0	0	1 681	34 890	0	0		0	0	0	0	4d18h UP		0	0	0		0		

F1

	Queue			Session rate			Sessions					Bytes		Denied		Errors			Warnings		Server						
	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	In	Out	Req	Resp	Req	Conn	Resp	Retr	Redis	Status	LastChk	Wght	Act	Bck	Chk	Dwn
Frontend				1	8	-	0	3	40 000	206 033		23 954	38 880 944	0	0	205 897					OPEN						

B1

	Queue			Session rate			Sessions					Bytes		Denied		Errors			Warnings		Server						
	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	In	Out	Req	Resp	Req	Conn	Resp	Retr	Redis	Status	LastChk	Wght	Act	Bck	Chk	Dw
ExampleServer	0	0	-	0	7		0	1	-	137	127	23 684	378 205		0		0	2	0	0	4d18h UP	L4OK in 0ms	1	Y	-	0	
backup	0	0	-	0	0		0	0	-	0	0	0	0		0		0	0	0	0			1	-	Y		
Backend	0	0		0	7		0	1	4 000	137	127	23 684	378 205	0	0		0	2	0	0	4d18h UP		1	1	1		

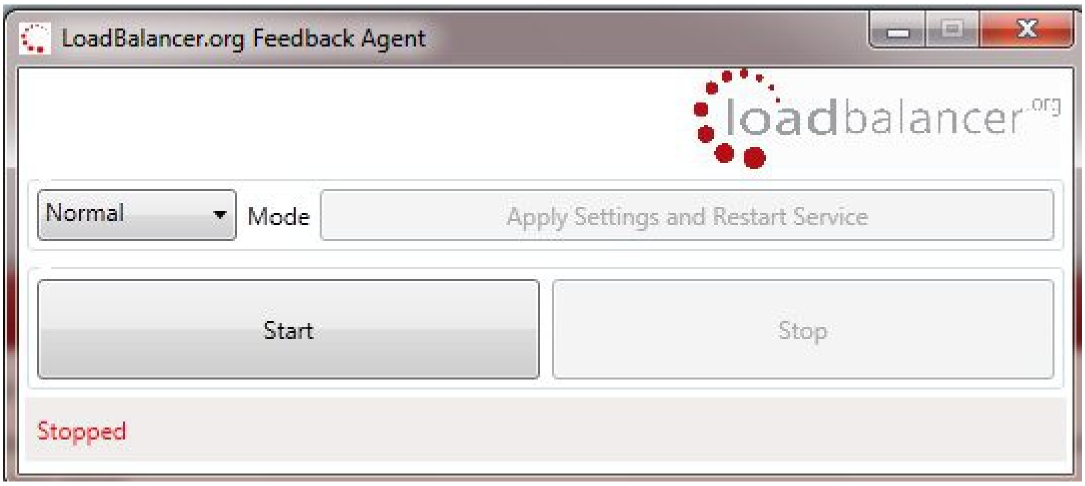
Using the Server Feedback Agent

To enable the load balancer to be aware of the backend servers current status, a feedback agent can be installed. The agent provides an integer value to the load balancer that represents its current utilization level. The agent monitors various system parameters including RAM, CPU etc.

The latest Windows agent is available for download at the following link:

http://downloads.loadbalancer.org/agent/windows/LBCPUMonInstallation.msi

Simply run the installer on each back-end Windows server. The monitor runs as a Windows service which is managed using a simple console. Once installed, navigate to *All Programs* > *Loadbalancer.org* > *Monitor* on the Windows server to start the console:

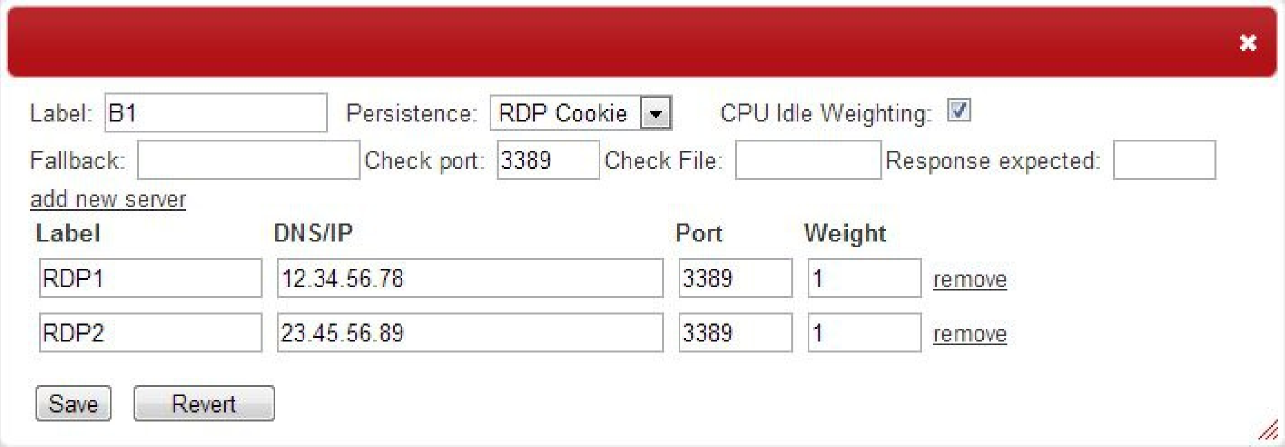

LoadBalancer.org Feedback Agent
loadbalancer.org
Apply Settings and Restart Service
Start
Stop

Mode: Normal

Stopped

Simply click **Start** to start the service.

To enable the feedback agent on the load balancer, select the **CPU Idle Weighting** checkbox in the backend group definition on the load balancer as shown below:



Label: B1 Persistence: RDP Cookie CPU Idle Weighting: ☑

Fallback: Check port: 3389 Check File: Response expected:

Label	DNS/IP	Port	Weight
RDP1	12.34.56.78	3389	1
RDP2	23.45.56.89	3389	1

Configuring High Availability using Auto-Scaling

This procedure should be followed to setup HA for your instance. If the instance is terminated or stops for any reason, auto-scaling will automatically start a new instance with the same settings and configuration. The steps required to set this up are shown below:

1) **Enter your Amazon Credentials** – copy and paste your Private key, Certificate and AWS Account ID from your AWS Security Credentials page to the corresponding sections of the Account tab, then click **Save -** once saved, the message '*Account credentials successfully saved and validated*' will be displayed

2) **Create an image of the instance** – right click the running instance and select '*Create Image (EBS AMI)*'

3) **Enter an appropriate name for the image** – e.g. 'AutoScaleImage-1'

4) **Start image creation** – click **Yes, Create**

5) **Check the creation status** – click on the displayed link : '*View Pending image ami-<code>*' , you can also click on the AMIs option under IMAGES in the navigation pane. Note that a new code is used and therefore it will be different than the original source instance

6) **Connect to the existing active Load balancer instance using SSH** – once the image has been created, i.e. when the status changes from '*pending*' to '*available*' start an SSH session to the load balancer

N.B. For SSH access, make sure that TCP port 22 is included in the security group for the load balancer

Using Linux:

first ensure that the private key is available on your Linux host, then use the following command to connect to the load balancer instance:

```
ssh -i <private-key-name>.pem ec2-user@<IP address of load balancer>
```

e.g.

```
ssh -i lbkeypair1.pem ec2-user@12.34.56.78
```

Using Windows:

Please refer to page 32 for details on using PuTTY.

7) **Configure variables on the load balancer** – run the following commands :

```
# switch to root user

sudo su

# setup the required variables

export KEY_HOME=/etc/loadbalancer.org/aws
export EC2_PRIVATE_KEY=$KEY_HOME/pk.pem
export EC2_CERT=$KEY_HOME/cert.pem
export EC2_HOME=/opt/aws/amitools/ec2
export JAVA_HOME=/usr
export AWS_AUTO_SCALING_HOME=/opt/aws/apitools/as/
export PATH=$EC2_HOME/bin:$AWS_AUTO_SCALING_HOME/bin:$PATH
```

N.B. If preferred, these can be placed in the root users .bashrc file. This will make these permanent so new instances will also have the same settings – this will be useful if you need to completely delete the instance.

8) **Create the Launch Configuration** – run the following command:

```
# ensure that --image-id is set to the new image created in step 2
# ensure that --region is set to the correct region
# ensure that --group is set to the relevant security group
# ensure that --key is set to your key pair
# ensure that --user-data is set to your elastic IP
```

```
as-create-launch-config autoscaleconf --image-id ami-a14142d5 --region us-east-1 --instance-type t1.micro
--group default --key lbkeypair-name --monitoring-disabled --user-data "12.34.56.78"
```

Once completed successfully, you should get the message : OK-Created launch config

9) **Create the Auto-Scale Group** – run the following command:

```
# ensure that --availability-zones is set to the correct zone
# ensure that --region is set to the correct region
```

```
as-create-auto-scaling-group autoscalegrp --availability-zones us-east-1d --launch-configuration
autoscaleconf --min-size 1 --max-size 1 --region us-east-1
```

Once completed successfully, you should get the message : OK-Created AutoScalingGroup

New Instance

The new load balancer instance should start up immediately (you can remove your old copy when you are fully happy with the new indestructible one). After it boots it should correctly assign itself the elastic IP address that you specified in the user-data field.

Testing

Now you can test the new indestructible instance using the Amazon Web Management Console to terminate the server, after a few seconds the auto-scaling policy should start a brand new copy of the instance.

Terminating the Instance

Since terminating the instance using the console causes another replacement instance to start, you'll need to use a different procedure if you want to completely terminate the image:

1) **Configure variables on the load balancer** – run the following commands :

N.B. this step is not required if the .bashrc file was modified as mentioned on the previous page

```
# switch to root user

sudo su

# setup the required variables

export KEY_HOME=/etc/loadbalancer.org/aws
export EC2_PRIVATE_KEY=$KEY_HOME/pk.pem
export EC2_CERT=$KEY_HOME/cert.pem
export EC2_HOME=/opt/aws/amitools/ec2
export JAVA_HOME=/usr
export AWS_AUTO_SCALING_HOME=/opt/aws/apitools/as/
export PATH=$EC2_HOME/bin:$AWS_AUTO_SCALING_HOME/bin:$PATH
```

2) **Remove the EC2 instance from the Auto Scaling group:**

The command below will terminate the instance.

```
as-update-auto-scaling-group autoscalegrp ––region us-east-1 ––min-size 0 ––max-size 0
```

Once completed successfully, you should get the message : OK-Updated AutoScalingGroup

N.B. It can take a few minutes for the instance to terminate, so you might have to refresh the status more than once on the web console.

3) **Delete the Auto Scaling Group:**

If required, the auto-scaling group can also be deleted. Since the load balancer instance is now terminated, the command below will need to be run on a different instance running in the same zone.

```
as-delete-auto-scaling-group autoscalegrp
```

4) **Delete the Launch Configuration:**

If required, the auto-scaling launch configuration can also be deleted. Since the load balancer instance is now terminated, the command below will need to be run on a different instance running in the same zone.

```
as-delete-launch-config autoscaleconf
```

NOTE: If the load balancer's settings are later changed, then the auto scaling image will need to be re-created.

Accessing the Load Balancer using SSH

This uses the private key that you downloaded when setting up your instance (please refer to page 14 of this guide). To connect to the load balancer using SSH, this private key must be used. Under Linux, the key can be used immediately, for PuTTY under Windows, the key must first be converted to a format required by PuTTY as detailed below.

N.B. For SSH access make sure that TCP port 22 is included in the security group for the load balancer

Linux

# First change the permission of the private key file to allow only the owner read access

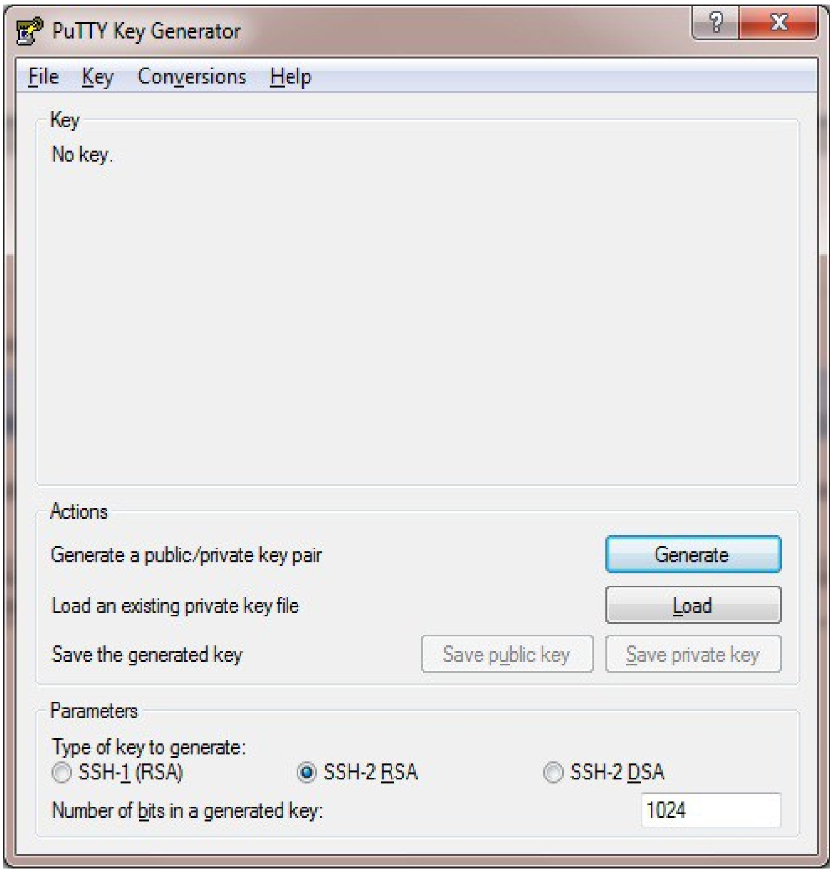
```
chmod 400 /path-where-saved/ec2-key-name.pem
```

# Now start SSH specifying the private key file

```
ssh -i /path-where-saved/ec2-key-name.pem ec2-user@dns-name or IP
```

Windows

For PuTTY, the private key must be converted into an appropriate format. To do this the PuTTYgen utility (included with PuTTY) must be used. Start PuTTYgen:



No key.

Generate a public/private key pair

Load an existing private key file

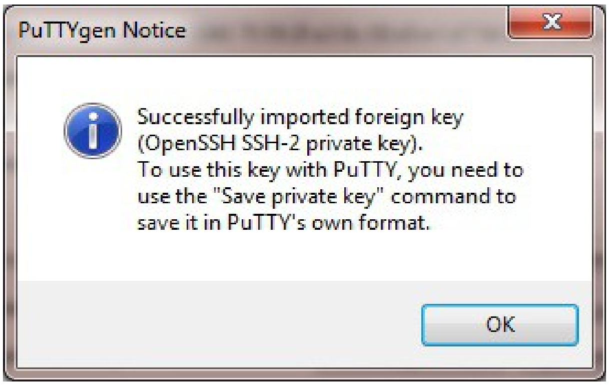
Save the generated key

Type of key to generate:
○ SSH-1 (RSA) ● SSH-2 RSA ○ SSH-2 DSA

Number of bits in a generated key: 1024

Click **Load**, change the file-type to all files and select the pem file saved earlier when creating your Key Pair.

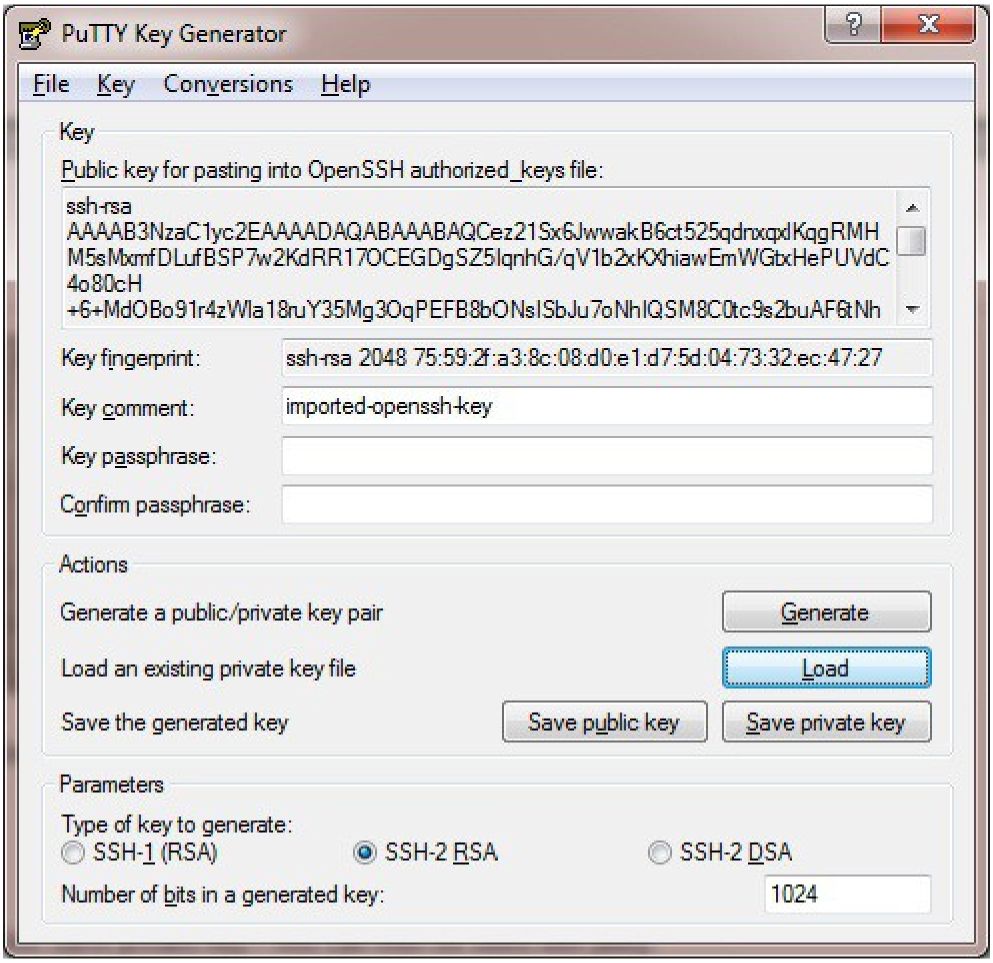
You should see the following message:



PuTTYgen Notice

Successfully imported foreign key (OpenSSH SSH-2 private key).
To use this key with PuTTY, you need to use the "Save private key" command to save it in PuTTY's own format.

Click **OK**



PuTTY Key Generator

Key

Public key for pasting into OpenSSH authorized_keys file:

```
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABAQCez21Sx6JwwakB6ct525qdnxqdKqgRMH
M5sMxmfDLufBSP7w2KdRR17OCEGDgSZ5IqnhG/qV1b2xKXhiawEmWGtxHePUVdC
4o80cH
+6+MdOBo91r4zWla18ruY35Mg3OqPEFB8bONsISbJu7oNhIQSM8C0tc9s2buAF6tNh
```

Key fingerprint: ssh-rsa 2048 75:59:2f:a3:8c:08:d0:e1:d7:5d:04:73:32:ec:47:27

Key comment: imported-openssh-key

Key passphrase:

Confirm passphrase:

Actions

Generate a public/private key pair

Load an existing private key file

Save the generated key

Parameters

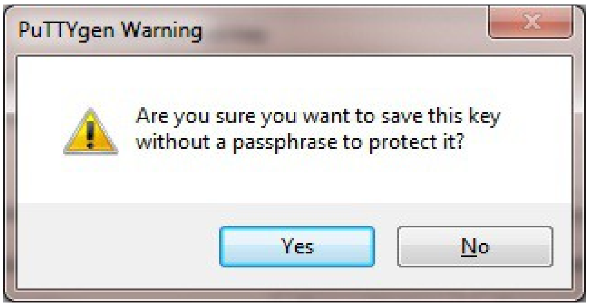
Type of key to generate:

○ SSH-1 (RSA) ● SSH-2 RSA ○ SSH-2 DSA

Number of bits in a generated key: 1024

Now Click **Save private key** – this can then be used with PuTTY.

NB. You can also choose to enter an additional pass-phrase for improved security, if you don't, the following message will be displayed:



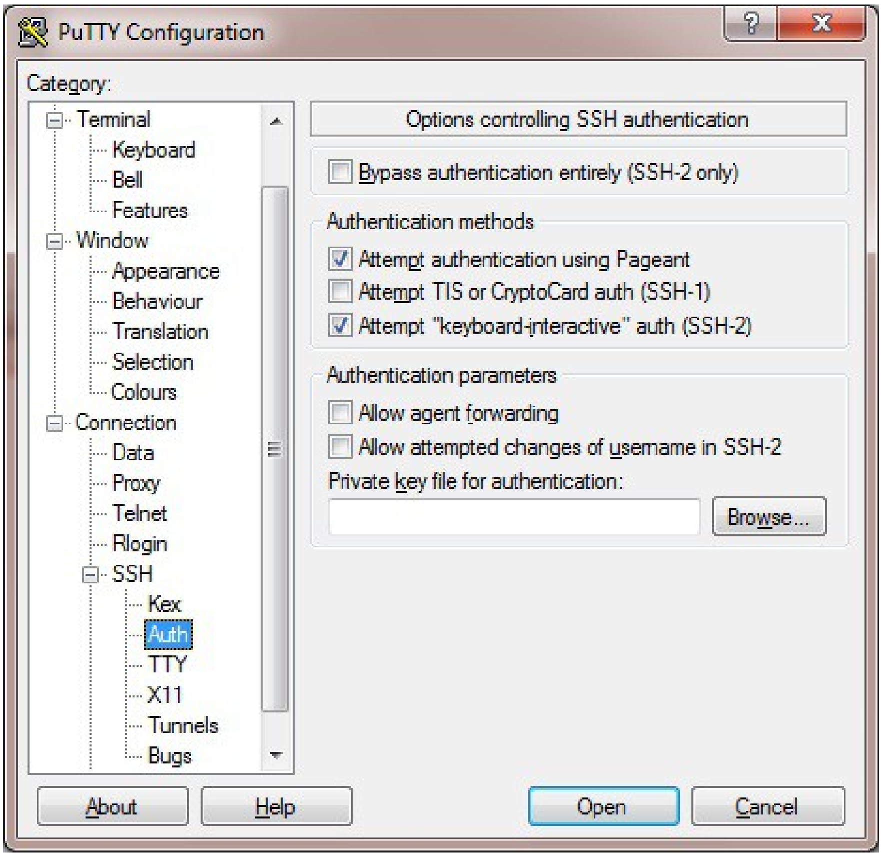
PuTTYgen Warning

Are you sure you want to save this key without a passphrase to protect it?

Click **Yes** and save the file with the default .ppk extension

Now close PuTTYgen and start PuTTY

Expand the SSH section as shown below:



PuTTY Configuration

Options controlling SSH authentication

☐ Bypass authentication entirely (SSH-2 only)

Authentication methods

☑ Attempt authentication using Pageant

☐ Attempt TIS or CryptoCard auth (SSH-1)

☑ Attempt "keyboard-interactive" auth (SSH-2)

Authentication parameters

☐ Allow agent forwarding

☐ Allow attempted changes of username in SSH-2

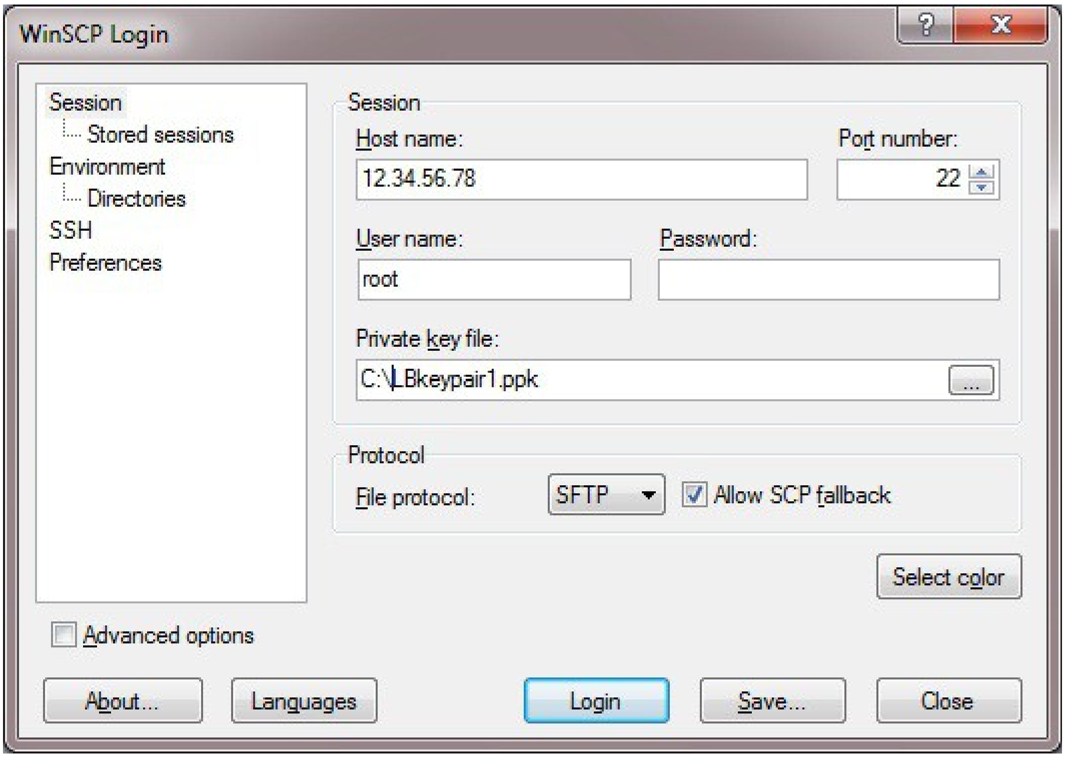
Private key file for authentication:

Click **Browse** and select the new .ppk file just created

When you open the SSH session, login as ec2-user – no password will be required.

Accessing the Load Balancer using WinSCP

With WinSCP, enter the relevant IP address and username root, then browse to the private key file created previously using PuTTYgen.



WinSCP Login

Session

Host name: 12.34.56.78

Port number: 22

User name: root

Password:

Private key file: C:\LBkeypair1.ppk

Protocol

File protocol: SFTP

☑ Allow SCP fallback

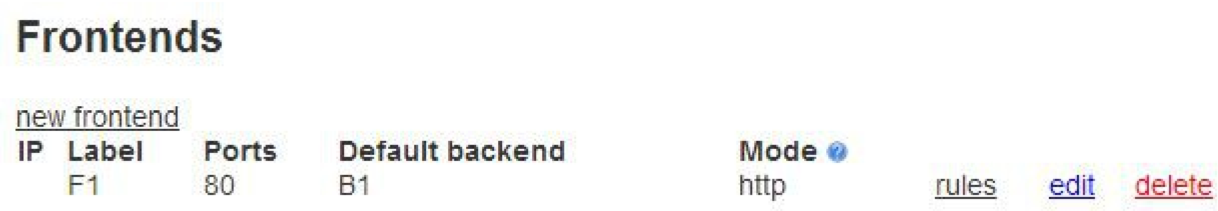
□ Advanced options

Click **Login**

Example Configurations

Example 1 - the Default Setup (HTTP Mode)

The Frontend:

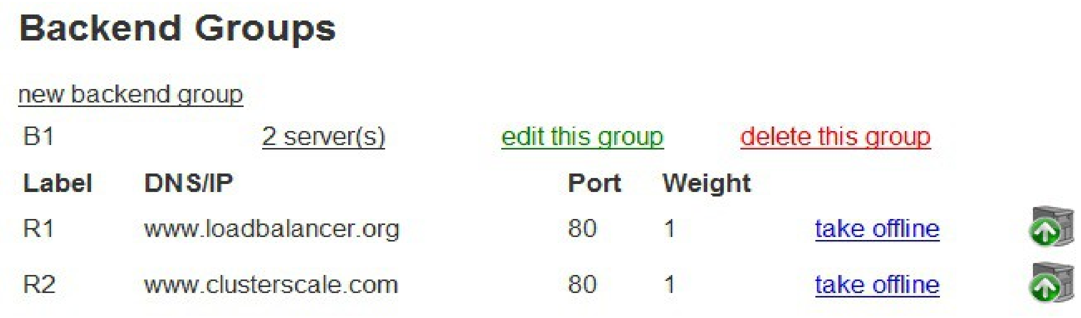


Frontends

new frontend

IP	Label	Ports	Default backend	Mode			
	F1	80	B1	http	rules	edit	delete

The Backend:



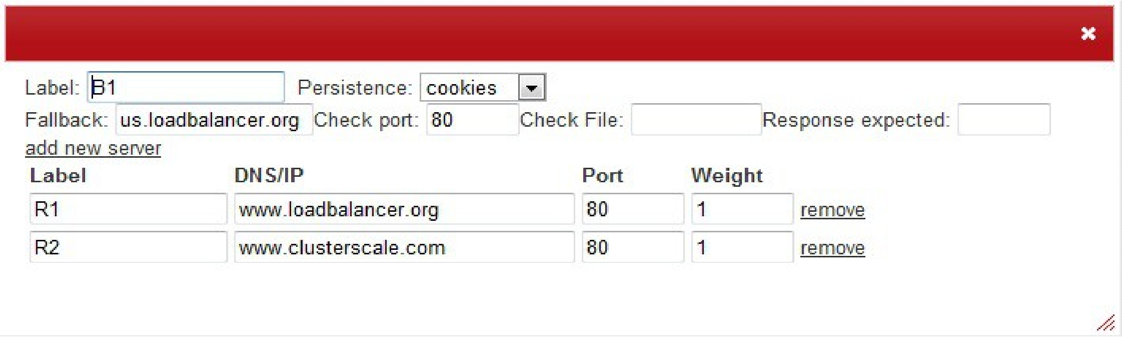
Backend Groups

new backend group

B1 2 server(s) edit this group delete this group

Label	DNS/IP	Port	Weight	
R1	www.loadbalancer.org	80	1	take offline
R2	www.clusterscale.com	80	1	take offline

Editing the Backends:



Label: B1 Persistence: cookies

Fallback: us.loadbalancer.org Check port: 80 Check File: Response expected:

add new server

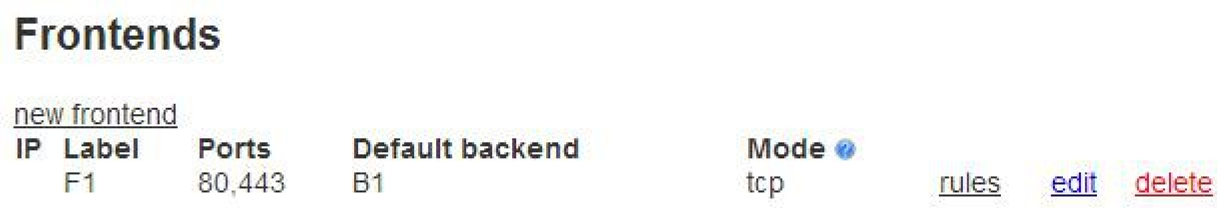
Label	DNS/IP	Port	Weight	
R1	www.loadbalancer.org	80	1	remove
R2	www.clusterscale.com	80	1	remove

Key Points:

- the frontend mode is set to HTTP
- the front-end listens on port 80
- persistence is set to *cookies*

Example 2 - SSL Termination on the Backend Servers (TCP Mode)

The Frontend:

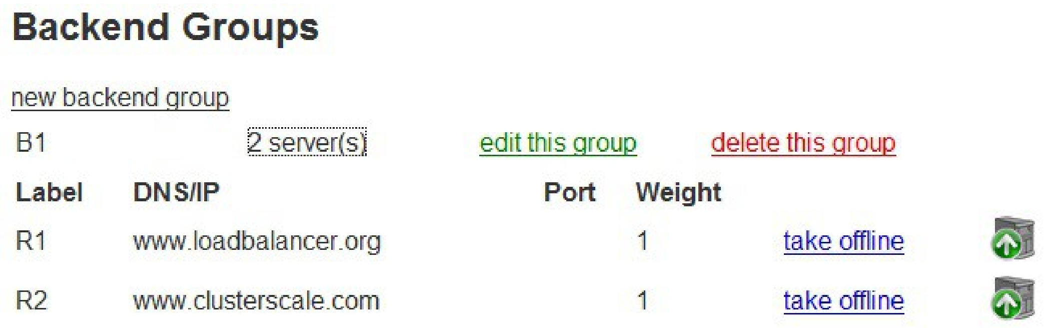


Frontends

new frontend

IP	Label	Ports	Default backend	Mode			
	F1	80,443	B1	tcp	rules	edit	delete

The Backend:



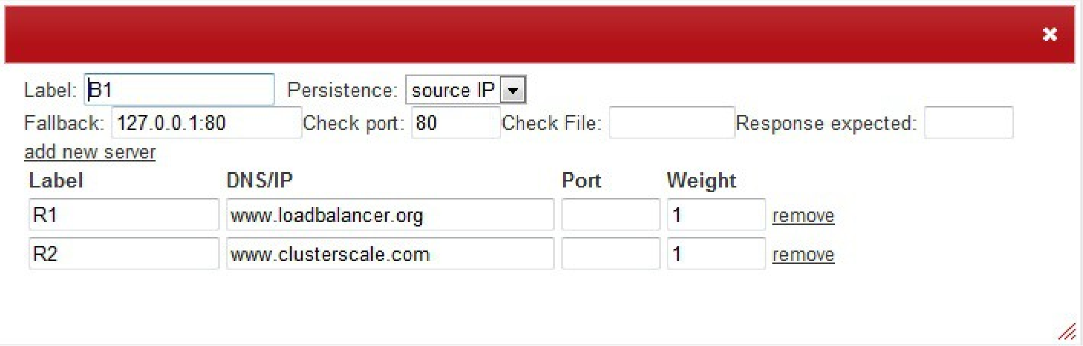
Backend Groups

new backend group

B1 2 server(s) edit this group delete this group

Label	DNS/IP	Port	Weight		
R1	www.loadbalancer.org		1	take offline	
R2	www.clusterscale.com		1	take offline	

Editing the Backend:



Label: B1 Persistence: source IP

Fallback: 127.0.0.1:80 Check port: 80 Check File: Response expected:

add new server

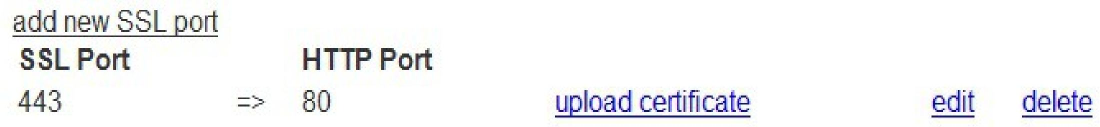
Label	DNS/IP	Port	Weight	
R1	www.loadbalancer.org		1	remove
R2	www.clusterscale.com		1	remove

Key Points:

- the frontend mode is set to TCP
- the frontend listens on both ports 80 & 443
- persistence is set to *source IP*
- no port is specified in the backend, traffic is then passed through to the same port on which it was received

Example 3 - SSL Termination on the Load Balancer (HTTP Cookie Backend)

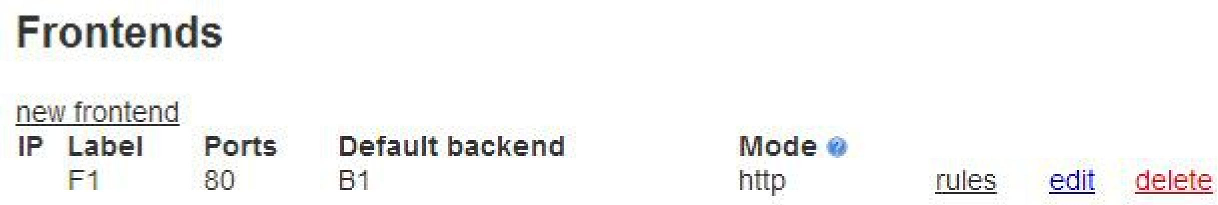
SSL Termination:



add new SSL port

SSL Port		HTTP Port			
443	=>	80	upload certificate	edit	delete

The Frontend:

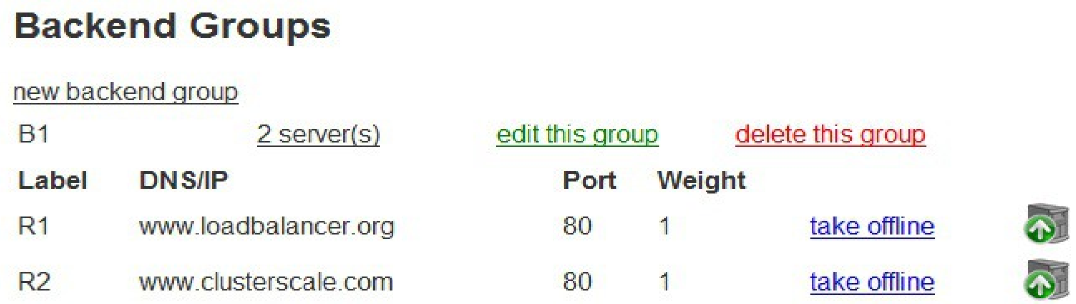


Frontends

new frontend

IP	Label	Ports	Default backend	Mode			
	F1	80	B1	http	rules	edit	delete

The Backend:



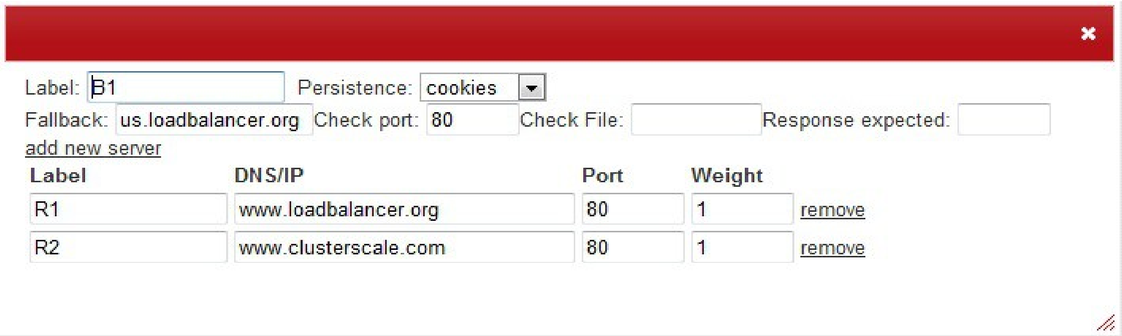
Backend Groups

new backend group

B1 2 server(s) edit this group delete this group

Label	DNS/IP	Port	Weight		
R1	www.loadbalancer.org	80	1	take offline	⬆
R2	www.clusterscale.com	80	1	take offline	⬆

Editing the Backend:



Label: B1 Persistence: cookies

Fallback: us.loadbalancer.org Check port: 80 Check File: Response expected:

add new server

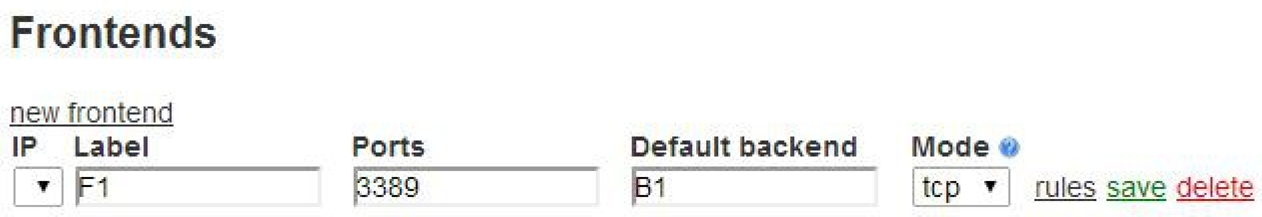
Label	DNS/IP	Port	Weight	
R1	www.loadbalancer.org	80	1	remove
R2	www.clusterscale.com	80	1	remove

Key Points:

- the backend only listens on port 80, SSL traffic is terminated on the load balancer and passed on to the backend servers unencrypted

Example 4 - Terminal Server / RDP using Source IP Persistence

The Frontend:

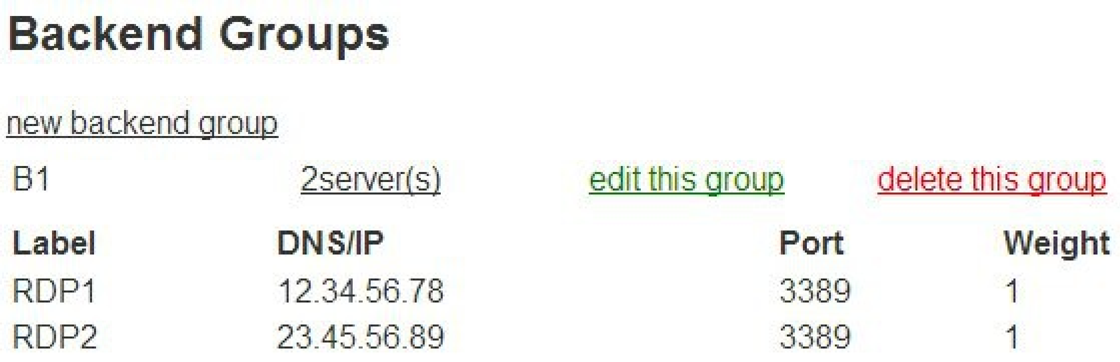


Frontends

new frontend

IP	Label	Ports	Default backend	Mode	
▾	F1	3389	B1	tcp ▾	rules save delete

The Backend:



Backend Groups

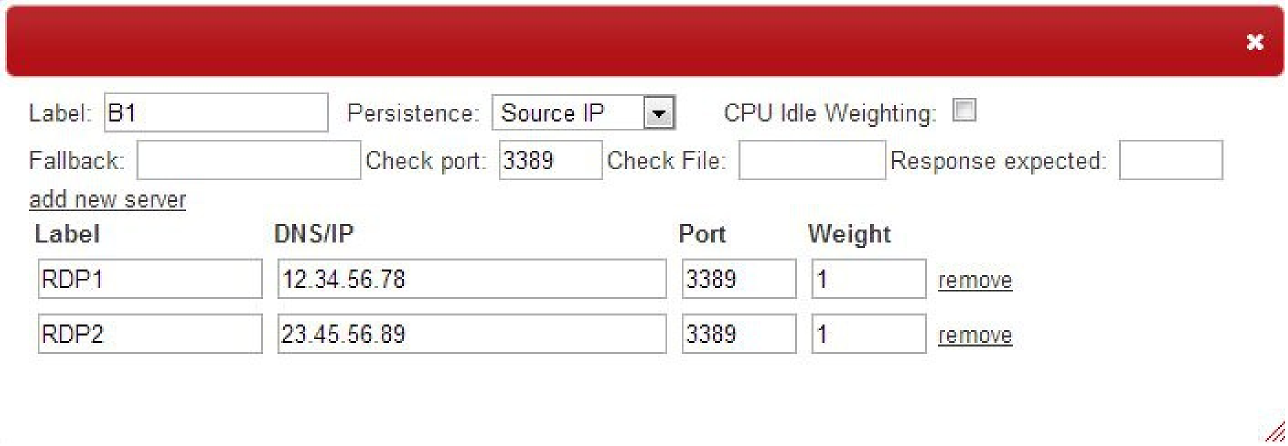
new backend group

B1 2server(s) edit this group delete this group

Label	DNS/IP	Port	Weight
RDP1	12.34.56.78	3389	1
RDP2	23.45.56.89	3389	1

Editing the Backend:

make sure that Persistence is set to Source IP



Label: B1 Persistence: Source IP ▾ CPU Idle Weighting: □

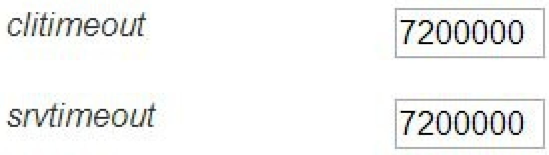
Fallback: Check port: 3389 Check File: Response expected:

add new server

Label	DNS/IP	Port	Weight	
RDP1	12.34.56.78	3389	1	remove
RDP2	23.45.56.89	3389	1	remove

Global Settings:

Change HAProxy's client & server timeouts to 7200000 (i.e. 2 hours)



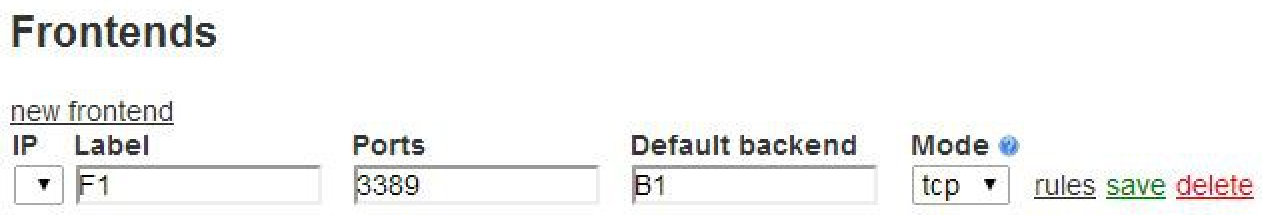
clitimeout	7200000
srvtimeout	7200000

Key Points:

- the frontend mode is set to TCP
- the frontend listens on port 3389
- persistence is set to *source IP*
- client & server timeouts need to be changed

Example 5 - Terminal Server / RDP using RDP Cookie Persistence

The Frontend:

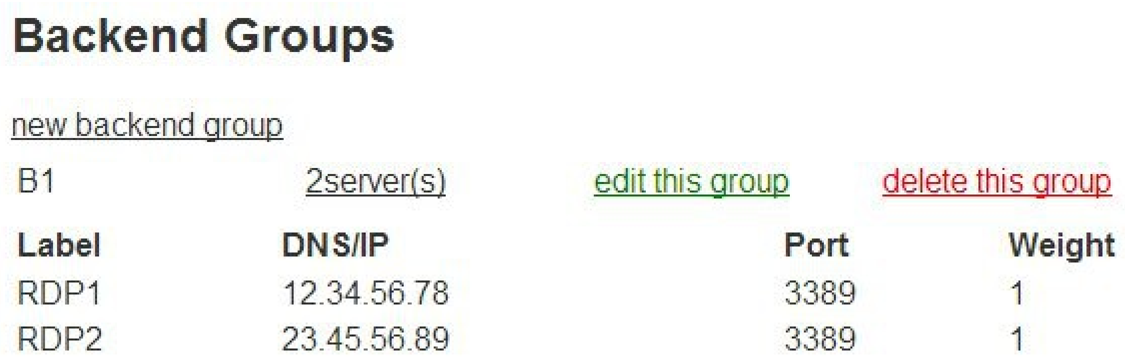


Frontends

new frontend

IP	Label	Ports	Default backend	Mode	
▾	F1	3389	B1	tcp ▾	rules save delete

The Backend:



Backend Groups

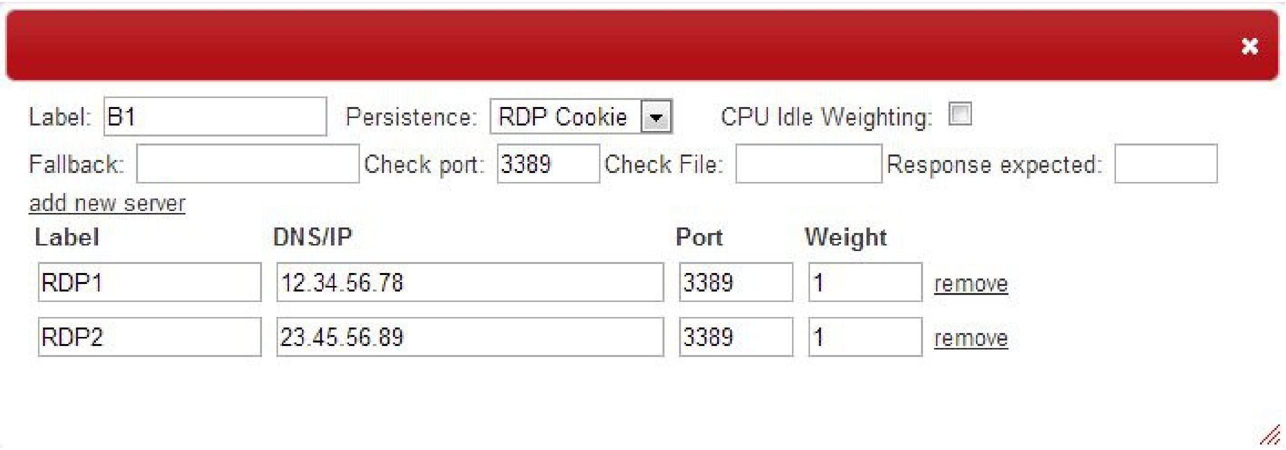
new backend group

B1 2server(s) edit this group delete this group

Label	DNS/IP	Port	Weight
RDP1	12.34.56.78	3389	1
RDP2	23.45.56.89	3389	1

Editing the Backend:

make sure that Persistence is set to RDP Cookie



Label: B1 Persistence: RDP Cookie ▾ CPU Idle Weighting: □

Fallback: Check port: 3389 Check File: Response expected:

add new server

Label	DNS/IP	Port	Weight	
RDP1	12.34.56.78	3389	1	remove
RDP2	23.45.56.89	3389	1	remove

Global Settings:

Change HAProxy's client & server timeouts to 7200000 (i.e. 2 hours)

clitimeout 7200000

srvtimeout 7200000

Key Points:

- the frontend mode is set to TCP
- the frontend listens on port 3389
- persistence is set to *RDP Cookie*
- client & server timeouts need to be changed

API

An API is available for modifying the running instance of the EC2 loadbalancer. It allows you to easily add and remove Real Servers to any defined Back-end Group.

If you have more than one back-end group you will need to specify the backend group to add the real server to, if however, you only have one backend group defined the real server will be added to this group. The only thing which needs to be specified is the dns name / IP address of the real server being added. If the real server label is not specified then the API will try a choose a default label.

Using function 'lb_modify' to add a server to a backed group

In this example only one backend group is defined on the EC2 load balancer. The following command can be entered on the real server:

```
ssh -i ec2_keypair.pem root@ec2loadbalancer “lb_modify -d realserverIP”
```

Doing the same but adding the server to backend group B1 with real server label L4:

```
ssh -i ec2_keypair.pem root@ec2loadbalancer “lb_modify -d realserverIP -b B1 -l L4”
```

Automatically Adding Real Servers to a BackEnd

If you have only one back end defined on the EC2 Loadbalancer Instance, you can use the following script to automatically add additional servers to the backend:

```
#!/bin/sh
PATH="/sbin:/bin:/usr/sbin:/usr/bin";
AMI_KEY_PAIR="<path-to-ssh-key>";
EC2_LOADBALANCER_IP="<ip-address-of-ec2-loadbalancer>";
CURL=`which curl`;
SSH=`which ssh`;
AMI_ID="`$CURL -s http://169.254.169.254/latest/meta-data/ami-id`";
AMI_IP="`$CURL -s http://169.254.169.254/latest/meta-data/local-ipv4`";

case "$1" in
start)
      $SSH -i $AMI_KEY_PAIR root@$EC2_LOADBALANCER_IP \"lb_modify -l $AMI_ID -d
      $AMI_IP \";
      exit 0;
      ;;
stop)
      $SSH -i $AMI_KEY_PAIR root@$EC2_LOADBALANCER_IP \"lb_modify -l $AMI_ID
      -d $AMI_IP -r \";
      ;;
*)
      exit 1;
      ;;
esac;
exit 0;
```

If you put this script in /etc/init.d on your real servers and link it to your startup/shutdown scripts (in /etc/rc2.d and above), when the real server boots up it will automatically be added to the cluster backend.

You will have to fill in the variables AMI_KEY_PAIR and EC2_LOADBALANCER_IP with the correct values for your instance.

If you have more than one backend defined in the SSH line (located under start/stop statements) will have to be amended with a switch to specify which backend to add the server to. The following example uses a backend called “BackEnd2”:

```
#!/bin/sh
PATH="/sbin:/bin:/usr/sbin:/usr/bin";
AMI_KEY_PAIR="<path-to-ssh-key>";
EC2_LOADBALANCER_IP="<ip-address-of-ec2-loadbalancer>";
CURL=`which curl`;
SSH=`which ssh`;
AMI_ID="`$CURL -s http://169.254.169.254/latest/meta-data/ami-id`";
AMI_IP="`$CURL -s http://169.254.169.254/latest/meta-data/local-ipv4`";
case "$1" in
start)
      $SSH -i $AMI_KEY_PAIR root@$EC2_LOADBALANCER_IP \"lb_modify -l $AMI_ID -d
      $AMI_IP -b BackEnd2 \";
      exit 0;
      ;;
stop)
      $SSH -i $AMI_KEY_PAIR root@$EC2_LOADBALANCER_IP \"lb_modify -l $AMI_ID
      -d $AMI_IP -r -b BackEnd2 \";
      ;;
*)
      exit 1;
      ;;
esac;
exit 0;
```

Loadbalancer.org Technical Support

If you have any questions don't hesitate to contact the support team: support@loadbalancer.org