

Введение в Amazon Web Services

Amazon Simple Storage Service (Amazon S3)

Amazon S3 предоставляет простой интерфейс в виде веб-сервиса, который может быть использован для хранения и обращения к любому количеству данных, в любое время, отовсюду. Это дает разработчикам доступ к гибко масштабируемой, надежной, быстрой и недорогой инфраструктуре хранения данных, используемой Amazon для работы своей глобальной сети веб-сайтов. Сервис нацелен на максимизацию выгод масштабирования, и предоставления этих выгод разработчикам.

Функционал Amazon S3

- ⌚ Запись, чтение и удаление объектов, содержащих от 1 байта до 5 гигабайт данных каждый. Число объектов хранения не ограничено. Каждый объект сохраняется и возвращается по уникальному присвоенному разработчиком ключу.
- Механизмы аутентификации гарантируют безопасность и целостность данных, и их недоступность для неавторизованного доступа. Объекты могут быть частными или общедоступными, а также ограничены доступом для отдельных пользователей.
- Использование стандартизованных протоколов REST и SOAP, совместимых с подавляющим большинством инструментария разработки для Internet.
- Построено гибко, что позволяет легко добавлять слои протоколов или функционала. Протокол по умолчанию – HTTP. Поддержка протокола BitTorrent(TM) предоставлена для снижения стоимости высокомасштабного распределения данных. Дополнительные интерфейсы в будущем могут добавиться.

Ценообразование

Платите за то, что используете. Минимальный тариф отсутствует.

С 1 июня 2007

Хранение

\$0.15 за гигабайто-месяц хранения

Трафик

\$0.10 за гигабайт входящего трафика.

\$0.18 за гигабайт исходящего трафика - до 10 терабайт в месяц

\$0.16 за гигабайт исходящего трафика - от 10 до 50 терабайт в месяц

\$0.13 за гигабайт исходящего трафика - свыше 50 терабайт в месяц

Данные, передаваемые между Amazon S3 и Amazon EC2 не оплачиваются.

Запросы

\$0.01 за 1,000 PUT или LIST запросов

\$0.01 за 10,000 GET и других запросов*

* запросы на удаление не оплачиваются

Концепции Amazon S3

Amazon S3 основано на идее Интернет-хранилища. Это освобождает разработчиков от размышлений, где их данные будут храниться, насколько хранение будет безопасным и надежным, и насколько его хватит. Функциональность проста и надежна: сохраняйте любое количество данных недорого и безопасно, будучи уверенными, что данные всегда доступны, когда они нужны вам. Amazon S3 позволяет разработчикам сфокусироваться на самих данных, а не на способе их хранения.

Amazon S3 построено с использованием следующих принципов:

- **Масштабируемость:** Amazon S3 может масштабироваться в терминах объема, частоты запросов, и количества пользователей, для поддержки неограниченного количества веб-приложений. Оно использует масштабируемость как преимущество – добавление узлов в систему добавляет ей доступность, емкость, и надежность.
- **Надежность:** Сохраняйте данные надежно, с гарантией доступности 99.99%. Не может быть единичных сбоев. Все сбои должны обрабатываться и восстанавливаться системой без простоя в обслуживании.
- **Скорость:** Amazon S3 должен быть быстр настолько, насколько нужно для высокопроизводительных приложений. Задержка сервера должна быть незначительна по отношению к задержкам Интернета в целом. Любые узкие места должны ликвидироваться простым добавлением узлов в систему.
- **Дешевизна:** Amazon S3 строится на недорогих типовых компонентах. В результате, частые отказы узлов являются нормой, и не подвергают всю систему краху.
- **Простота:** Построение масштабируемых, надежных, быстрых и недорогих хранилищ довольно сложная задача. Amazon S3 делает попытка сочетать эти характеристики.

Amazon Elastic Compute Cloud (Amazon EC2)

Так же как Amazon Simple Storage Service (Amazon S3) позволяет «хранение-в-облаке», Amazon EC2 позволяет «вычисления-в-облаке». Amazon EC2 – простой веб-сервис, позволяющий получить и настроить вычислительные емкости с минимальной задержкой. Он предоставляет Вам полный контроль над вычислительными ресурсами, и позволяет работать на вычислительном окружении Amazon. Amazon EC2 уменьшает время на получение и загрузку новых серверов до минут, позволяя вам быстро масштабировать емкость вверх и вниз. Amazon EC2 меняет экономику вычислений, позволяя вам платить только за то, что вы потребляете.

Функциональность Amazon EC2

Для того чтоб использовать Amazon EC2, вы просто:

- Создаете an Amazon Machine Image (AMI), содержащий ваши приложения, библиотеки, данные, и настройки. Или используете готовые шаблонные образы.
- Загружаете AMI в Amazon S3. Amazon EC2 предоставляет инструменты для простого сохранения AMI. Amazon S3 предоставляет надежный, безопасный и быстрый репозиторий хранения ваших образов.
- Используете Amazon EC2 веб-сервис для настройки безопасности и сетевого доступа.
- Используете Amazon EC2 веб-сервис для запуска, отсановки и мониторинга экземпляров ваших AMI по необходимости.
- Платите за время использования и пропускную способность, использованную по факту.

Выгоды

- **Эластичность**
Amazon EC2 позволяет вам увеличивать или уменьшать емкость в минуты. Вы можете использовать один, сотни, или тысячи серверов одновременно. Поскольку все управляется веб-сервисом, ваше приложение может масштабироваться автоматически в зависимости от потребности.
- **Полная управляемость**
Вы осуществляете полное управление над вашими экземплярами. Вы имеете административный доступ к каждому экземпляру. Каждый экземпляр эквивалентен системе с 1.7 ГГц x86 процессором, 1.75 Гб ОЗУ, 160Гб локального диска, и 250 Мб/сек полосы пропускания.
- **Совместная работа с Amazon S3**
Amazon EC2 работает совместно с Amazon Simple Storage Service (Amazon S3) для обеспечения комбинированного решения хранения и вычислений.
- **Надежность**
Amazon EC2 предлагает высоконадежное окружение. Сервис запущен на отказоустойчивых дата-центрах Amazon.
- **Безопасность**
Amazon EC2 предоставляет интерфейс к веб-сервису для управления сетевой безопасностью. Вы назначаете группы экземпляров и их доступность.
- **Дешевизна**
Вы платите только за используемые мощности. Абонентская плата за обслуживание не взимается.

Ценообразование

Платите за то, что используете. Минимального тарифа не существует.

С 1 июня 2007

Экземпляры

\$0.10 за 1 час работы (полный или неполный)

Трафик

\$0.10 за 1 Гб – входящего трафика

\$0.18 за 1 Гб исходящего трафика – первые 10 Тб в месяц

\$0.16 за 1 Гб исходящего трафика – от 10 до 50 Тб в месяц

\$0.13 за 1 Гб исходящего трафика – свыше 50 Тб в месяц

Данные, передаваемые внутри подсистемы Amazon EC2, или между Amazon EC2 и Amazon S3, не оплачиваются. В остальных случаях действуют текущие тарифы Amazon S3.

Счет на использование Amazon S3 выставляется отдельно от счета за использование Amazon EC2. Начисления производятся в конце месяца.

Подробное описание

Создание Amazon Machine Image (AMI)

Amazon EC2 позволяет вам устанавливать и настраивать все – от операционной системы до приложений. Amazon Machine Image (AMI) – простое упакованное окружение, которое включает все необходимое для загрузки вашей машины. AMI являются единицей доставки. У вас может быть один AMI или несколько (вебсервера, сервера приложений, баз данных). Amazon EC2 предоставляет набор инструментов командной строки для легкого создания AMI. Создав произвольный AMI, вы должны загрузить его на Amazon S3. Amazon EC2 использует Amazon S3 для надежного и масштабируемого хранения ваших AMI.

Если у вас нету желания создавать AMI с нуля, вы можете выбрать любой из имеющихся общедоступных. Например, если Вам требуется всего лишь Linux-сервер, вы можете выбрать один из стандартных дистрибутивов.

Выполнение AMI

После того как у вас уже есть учетная запись и загруженные AMI, вы готовы к запуску вашего экземпляра. Вы можете запустить AMI на множестве экземпляров вызовом *RunInstances*. Если Вам требуется запуск более чем 20 экземпляров, пожалуйста свяжитесь с Amazon по адресу aws@amazon.com.

Если Amazon EC2 выполнила Ваш запрос, *RunInstances* вернет положительный результат, и мы можем переходить к настройке экземпляров. Вы можете проверить состояние экземпляров вызовом *DescribeInstances*. Кроме того, вы можете программно останавливать экземпляры вызовом *TerminateInstances*.

Оплата по факту

В конце каждого месяца производятся начисления за потребленные экземпляро-часы и каждый гигабайт данных, переданных вовне дата-центров Amazon. Оплата за данные между экземплярами Amazon EC2 и другими сервисами Amazon (например Amazon S3) не взимается.

Быстрый старт

Лучший способ понять Amazon EC2 – это работа с руководством быстрого старта, являющегося частью [технической документации](#).

Установка инструментария

Перед использованием Amazon EC2, необходимо скачать утилиты командной строки и настроить вашу учетную запись.

Закачка утилит командной строки

Утилиты доступны в виде ZIP архива в [Amazon EC2 Resource Center](#). Они написаны на Java и имеют скрипты оболочки для Windows 2000/XP и Linux/Unix/Mac OSX. Установка не требуется – просто распакуйте архив. Однако потребуется некоторая первоначальная настройка.

Настройка домашнего каталога

Утилитам требуется переменная окружения (EC2_HOME) для определения библиотек поддержки. Вам потребуется определить эту переменную прежде чем вы начнете использовать утилиты. Она должна быть установлена в имя каталога, который поименован ec2-api-tools-A.B-rrrr (A, B и r – номера версии и релиза), и содержать подкаталоги bin и lib.

Linux:

```
$ export EC2_HOME=<path-to-tools>
```

Windows:

```
C:\> set EC2_HOME=<path-to-tools>
```

Советуем добавить этот путь в PATH

Linux:

```
$ export PATH=$PATH:$EC2_HOME/bin
```

Windows:

```
C:\> set PATH=%PATH%;%EC2_HOME%\bin
```

Настройка аутентификации

Утилитами потребуется доступ к закрытому ключу и сертификату X.509, сгенерированному после [регистрации](#) в Amazon EC2.

Так как Вам ничего не мешает иметь более чем одну учетную запись, возникает необходимость в аутентификации при использовании утилит. Это можно делать при каждом запуске, однако сильно затруднительно. Проще установить переменные окружения.

Необходимо две переменных среды. EC2_PRIVATE_KEY, указывающая на ваш закрытый ключ, и EC2_CERT, указывающая на ваш X509 сертификат.

Linux:

```
$ export EC2_PRIVATE_KEY=~/.ec2/pk-HKZYKTAIG2ECMXYIBH3HXV4ZBZQ55CLO.pem  
$ export EC2_CERT=~/.ec2/cert-HKZYKTAIG2ECMXYIBH3HXV4ZBZQ55CLO.pem
```

Windows:

```
C:\> set EC2_PRIVATE_KEY=c:\ec2\pk-HKZYKTAIG2ECMXYIBH3HXV4ZBZQ55CLO.pem  
C:\> set EC2_CERT=c:\ec2\cert-HKZYKTAIG2ECMXYIBH3HXV4ZBZQ55CLO.pem
```

Можете работать с Amazon EC2.

Создание образа

Создание нового образа

Вы можете создавать новый AMI из уже имеющихся у вас машин, или изменяя существующий образ. Это раздел описывает изменение существующего образа.

Изменение существующего образа

Подготовка к упаковке

После того как вы создали образ, или внесли необходимые изменения, необходимо упаковать образ системы в AMI при помощи **ec2-bundle-vol**.

ec2-bundle-vol зашифровывает и подписывает образ, чтобы кроме вас и Amazon EC2 никто более не смог вскрыть образ

Убедитесь, что у вас есть закрытый ключ и сертификат, полученные как это описано в разделе [«Регистрация в Amazon EC2»](#), и содержатся они в файлах `pk-HKZYKTAIG2ECMXYIBH3HXV4ZBZQ55CLO.pem` и `cert-HKZYKTAIG2ECMXYIBH3HXV4ZBZQ55CLO.pem`.

```
PROMPT> scp -i id_rsa-gsg-keypair pk-HKZYKTAIG2ECMXYIBH3HXV4ZBZQ55CLO.pem cert-
HKZYKTAIG2ECMXYIBH3HXV4ZBZQ55CLO.pem root@domU-12-34-31-00-00-
05.usma1.compute.amazonaws.com:/mnt
pk-HKZYKTAIG2ECMXYIBH3HXV4ZBZQ55CLO.pem          100% 717   0.7KB/s  00:00
cert-HKZYKTAIG2ECMXYIBH3HXV4ZBZQ55CLO.pem        100% 684   0.7KB/s  00:00
```

Note: Ваша система может иметь иной синтаксис для команды защищенного копирования **scp**, ознакомьтесь с разделом [“SSH Clients”](#)

Note: Важно, чтобы сертификат и закрытый ключ находились в каталоге `/mnt` для предотвращения их слияния с создаваемым AMI.

Упаковка

Упакуем AMI с указанием вашего *AWS account ID* (идентификатор учетной записи *Amazon Web Services*) в качестве вашего имени пользователя (не путайте с идентификатором ключа доступа!):

```
# ec2-bundle-vol -d /mnt -k /mnt/pk-HKZYKTAIG2ECMXYIBH3HXV4ZBZQ55CLO.pem -c /mnt/cert-HKZYKTAIG2ECMXYIBH3HXV4ZBZQ55CLO.pem -u 495219933132 -s 1536
```

```
Copying / into the image file /mnt/image.img...
```

```
Excluding:
```

```
  /sys  
  /dev/shm  
  /proc  
  /dev/pts  
  /proc/sys/fs/binfmt_misc  
  /dev  
  /media  
  /mnt  
  /proc  
  /sys  
  /tmp/image.img  
  /mnt/img-mnt
```

```
1+0 records in
```

```
1+0 records out
```

```
mke2fs 1.38 (30-Jun-2005)
```

```
warning: 256 blocks unused.
```

```
Splitting /mnt/image.gz.crypt...
```

```
Created image.part.00
```

```
Created image.part.01
```

```
Created image.part.02
```

```
Created image.part.03
```

```
...
```

```
Created image.part.22
```

```
Created image.part.23
```

```
Generating digests for each part...
```

```
Digests generated.
```

```
Creating bundle manifest...
```

```
Bundle Volume complete.
```

Эта команда может занять продолжительное время. После выполнения AMI и его манифест будут в каталоге `/mnt`. Убедитесь в этом:

ls -l /mnt/image.*

```
-rw-r--r-- 1 root root 1611661312 Jun 20 10:12 /mnt/image.image
-rw-r--r-- 1 root root 10485760 Jun 20 10:12 /mnt/image.part.00
-rw-r--r-- 1 root root 10485760 Jun 20 10:12 /mnt/image.part.01
-rw-r--r-- 1 root root 10485760 Jun 20 10:12 /mnt/image.part.02
-rw-r--r-- 1 root root 10485760 Jun 20 10:12 /mnt/image.part.03
...
-rw-r--r-- 1 root root 10485760 Jun 20 10:12 /mnt/image.part.22
-rw-r--r-- 1 root root 10485760 Jun 20 10:12 /mnt/image.part.23

-rw-r--r-- 1 root root 2970 Jun 20 10:12 /mnt/image.manifest
```

Выгрузка AMI на Amazon S3

S3 сохраняет объекты данных в блоках, по концепции напоминающие каталоги. Вы должны указать имя блока вместо `<your-s3-bucket>`. Блоки имеют глобально-уникальные имена и принадлежат уникальным пользователям. Если вы использовали S3 ранее, укажите любой из ваших блоков, или укажите для **ec2-upload-bundle** любое осмысленное имя блока. **ec2-upload-bundle** загрузит AMI в указанный блок. Если блок не существует, он будет создан. Если указанный блок принадлежит другому пользователю, **ec2-upload-bundle** сгенерирует ошибку.

Вам потребуется ваш идентификатор ключа доступа (`<aws-access-key-id>`) и закрытый ключ доступа (`<aws-secret-access-key>`).

Процесс загрузки может быть длительным, но вы будете получать регулярные уведомления о ходе загрузки.

```
# ec2-upload-bundle -b <your-s3-bucket> -m /mnt/image.manifest.xml -a <aws-access-key-id> -s  
<aws-secret-access-key>  
Encrypting bundle manifest...  
Completed encryption.  
Uploading encrypted manifest...  
Uploaded encrypted manifest to https://s3.amazonaws.com/<your-s3-bucket>/image.manifest.xml.  
Uploading bundled AMI parts to https://s3.amazonaws.com/<your-s3-bucket>/image...  
Uploaded image.part.00 to https://s3.amazonaws.com/<your-s3-bucket>/image.part.00.  
Uploaded image.part.01 to https://s3.amazonaws.com/<your-s3-bucket>/image.part.01.  
Uploaded image.part.02 to https://s3.amazonaws.com/<your-s3-bucket>/image.part.02.  
Uploaded image.part.03 to https://s3.amazonaws.com/<your-s3-bucket>/image.part.03.  
...  
Uploaded image.part.23 to https://s3.amazonaws.com/<your-s3-bucket>/image.part.23.  
Uploaded image.part.24 to https://s3.amazonaws.com/<your-s3-bucket>/image.part.24.  
Upload Bundle complete.
```

Note: Обратите внимание, что выгруженные на S3 данные подлежат оплате в соответствии с тарифами S3.

После этого этапа ваши AMI готовы к регистрации и запуску.

Регистрация AMI

Вы должны зарегистрировать свои AMI , чтобы Amazon EC2 могла обнаружить их и запускать экземпляры, базирующиеся на них.

```
PROMPT> ec2-register <your-s3-bucket>/image.manifest.xml  
IMAGE ami-5bae4b32
```

После маркера IMAGE вы получаете идентификатор AMI (ami-5bae4b32 в примере) пригодный для запуска экземпляров.

Запуск экземпляров

Вы можете запускать экземпляры конкретного AMI, указывая идентификатор образа.

```
PROMPT> ec2-run-instances ami-5bae4b32  
INSTANCE i-10a64379 ami-5bae4b32 pending 0
```

После маркера INSTANCE вы получаете идентификатор экземпляра, пригодный для управления экземпляром. Эта команда запускает одиночный экземпляр.

Note: При запуске общедоступных AMI будет выдано предупреждение, т.к. Amazon не может гарантировать устойчивости и безопасности их функционирования.

Запуск экземпляра

Введение

Прежде чем вы приступите, проверьте, что вы выполнили все требования.

- Убедитесь в наличии Java-машины версии 1.5.0 или совместимой, и переменной JAVA_HOME установленной корректно. Для справки обращайтесь к разделу [“Среда исполнения Java Runtime Environment”](#).
- Убедитесь, что у вас есть учетная запись Amazon Web Services и вы подписаны как на Amazon S3 так и Amazon EC2. Для справки обращайтесь к разделу [«Настройка учетной записи»](#).
- Убедитесь, что в вашем домашнем каталоге есть папка .ec2, содержащая ваш сертификат и закрытый ключ. Убедитесь что имена файлов ключа и сертификата присвоены корректно. Для справки обращайтесь к разделу [“Регистрация на Amazon EC2”](#).
- Убедитесь, что переменная EC2_HOME установлена корректно. Для справки обращайтесь к разделу [“Настройка домашнего каталога”](#).
- Убедитесь, что переменные EC2_CERT и EC2_PRIVATE_KEY установлены корректно. Для справки обращайтесь к разделу [“Настройка домашнего каталога”](#).

После этого можете переходить к запуску.

Поиск AMI

Первый шаг – поиск подходящего AMI.

```
PROMPT> ec2-describe-images -o self -o amazon
IMAGE ami-5bae4b32 ec2-public-images/getting-started.manifest.xml amazon available public
IMAGE ami-68ae4b01 ec2-public-images/fedora-core4-base.manifest.xml amazon available public
IMAGE ami-69ae4b00 ec2-public-images/fedora-core4-apache-mysql.manifest.xml amazon available public
IMAGE ami-6dae4b04 ec2-public-images/fedora-core4-apache.manifest.xml amazon available public
IMAGE ami-6fae4b06 ec2-public-images/fedora-core4-mysql.manifest.xml amazon available public
IMAGE ami-61a54028 <your-s3-bucket>/image.manifest.xml 495219933132 available private
```

Эта команда показывает ваши и общедоступные AMI. Обратите внимание на значение во второй колонке – это AMI ID, который необходимо знать для запуска экземпляра.

Создание ключевой пары

Вы запускаете экземпляр общедоступного AMI. Т.к он не имеет паролей, вам потребуется пара из открытого и закрытого ключей для входа в экземпляр. Одна половина ключей будет внедрена в ваш экземпляр, для безопасного соединения без пароля с использованием другой половины пары ключей. Пока вы учитесь, вы можете выбрать другие механизмы для защищенного входа. Каждая ключевая пара требует имени. Выберите легко запоминаемое имя. Для примера будем использовать имя `gsg-keypair`.

```
PROMPT> ec2-add-keypair gsg-keypair
```

```
KEYPAIR gsg-keypair 1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f
```

```
-----BEGIN RSA PRIVATE KEY-----
```

```
MIIIEoQIBAAKCAQBuLFg5ujHrtm1jnutSuoO8Xe56LIT+HM8v/xkaa39EstM3/aFXTHgEIQiJLChp  
HungXQ29VTc8rc1bW0lkdi23OH5eqkMHGhvEwqa0HWASUMII4o3o/IX+0f2UcPoKCOVUR+jx71Sg  
5AU52EQfanIn3ZQ8IFW7Edp5a3q4DhjGIUKToHVbicL5E+g45zfB95wlyywWZfeW/UUF3LpGZyq/  
ebiUlq1qTbHkLbCC2r7RTn8vpQWp47BGVYGtGSBMPTRP5hnbzzuqj3itkiLHjU39S2sJCJ0TrJx5  
i8BygR4s3mHKBj8I+ePQxG1kGbF6R4yg6sECmXn17MRQVXODNHZbAgMBAAEcggEAY1tsiUslwDI5  
91CXirkYGuVfLyLflXenxfI50mDFms/mumTqloHO7tr0oriHDR5K7wMcY/Y5YkcXNo7mvUVD1pM  
ZNUJs7rw9gZRTf7LylaJ58kOcyajw8TsC4e4LPbFaHwS1d6K8rXh64o6WgW4SrsB6ICmr1kGQI7  
3wcfgt5eclu4TZf0OE9IHjn+2eRlsrjBdeORi7KiUNC/pAG23I6MdDOFEQRcCSigCj+4/mciFUSA  
SWS4dMbrpb9FNSlcf9dcLxVM7/6KxgJNfZc9XWzUw77Jg8x92Zd0fVhHOux5IZC+UvSKWB4dyfcl  
tE8C3p9bbU9VGyY5vLCAilb4qQKBgQDLiO24GXrlkswF32YtBBMuVgLGcWU9h9HIO9mKAc2m8Cm1  
jUE5lpzRjTeduc9I2qilMUTwtgnw42auSCzbUeYMURPtDqyQ7p6AjMujp9EPemcSVOK9vXYL0Ptco  
xW9MC0dtV6iPKCN7gOqiZXP RKaFbWADp16p8UAlvS/a5XXk5jwKBgQCKkpHiEISh1uRkxhlyjWC  
iDCiK6JBRsMvpLbc0v5dKwP5alo1fmdR5PJaV2qvZSj5CYnpMAy1/EDNTY5OSIJU+0KFmQbyhsbm  
rdLNLDL4+TcnT7c62/aH01ohYaf/VCbRhtLIBfqGoQc7+sAc8vmKkesnF7CqCEKDyF/dhrxYdQKB  
gC0iZzzNAapayz1+JcVTwwEid6j9JqNXbBc+Z2YwMi+T0Fv/P/hwkX/ypeOXnIUcw0lh/YtGBVAC  
DQbsz7LcY1HqXiHKYNWNvXgwwO+oiChjxvEkSdsTTIfnK4VScvU9BxDbQHjdiNDJbL6oar92UN7V  
rBYvChJZF7LvUH4YmVpHAoGAbZ2X7XvoeEO+uZ58/BGKOIGHByHBDiXtzMhdJr15HTYjxK7OgTZm  
gK+8zp4L9IbvLGDMJO8vft3XPEWuvl8twCzFH+CsWLQADZMZKSSBasOZ/h1FwhdMgCMcY+Qlzd4  
JZKjTSu3i7vhvx6RzdSedXEMNTZWN4qllx3kR5aHcukCgYA9T+Zrvm1F0seQPbLknn7EqhXlJBaT  
P8TTvW/6bdPi23ExzxZn7KODrfclYRph1LHMpAONv/x2xALIf91UB+v5ohy1oDoasL0gij1houRe  
2ERKKdwz0ZL9SWq6VTdhr/5G994CK72fy5WhyERbDjUldHaK3M849JJuf8cSrvSb4g==
```

```
-----END RSA PRIVATE KEY-----
```

Возвращаемый закрытый ключ нужно сохранить в локальном файле. Воспользуйтесь текстовым редактором (vi, emacs, edit, блокнот), создайте файл id_rsa-gsg-keypair и вставьте все между (и вместе со) строками "-----BEGIN RSA PRIVATE KEY-----" и "-----END RSA PRIVATE KEY-----" в этот файл. Убедитесь, что текст выглядит примерно так:

```
-----BEGIN RSA PRIVATE KEY-----
MIIIEoQIBAAKCAQBuLFg5ujHrtm1jnutSuoO8Xe56LIT+HM8v/xkaa39EstM3/aFXTHgEIQiJLChp
HungXQ29VTc8rc1bW0lkdi23OH5eqkMHGhvEwqa0HWASUMII4o3o/IX+0f2UcPoKCOVUR+jx71Sg
5AU52EQfanIn3ZQ8IFW7Edp5a3q4DhjGIUKT0HVbicL5E+g45zfB95wlyywWZfeW/UUF3LpGZyq/
ebiUq1qTbHkLbCC2r7RTn8vpQWp47BGVYgtGSBmpTRP5hnbzquj3itkiLHjU39S2sJCJ0TrJx5
i8BygR4s3mHKBJ8I+ePQxG1kGbF6R4yg6sECmXn17MRQVXODNHZbAgMBAACggEAY1tsiUslwDI5
91CXirkYGuVfLyLflXenxf150mDFms/mumTqloHO7tr0oriHDR5K7wMcY/YY5YkcXNo7mvUVD1pM
ZNUJs7rw9gZRTf7LylaJ58kOcyajw8TsC4e4LPbFaHwS1d6K8rXh64o6WgW4SrsB6ICmr1kGQI7
3wcfgt5eclu4TZf0OE9IHjn+2eRlsrjBdeORi7KiUNC/pAG23I6MdDOFEQRcCSigCj+4/mciFUSA
SWS4dMbrpb9FNSlcf9dcLxVM7/6KxgJNfZc9XWzUw77Jg8x92Zd0fVhHOux5IZC+UvSKWB4dyfcl
tE8C3p9bbU9VgyY5vLCAilb4qQKBgQDLiO24GXrlksWf32YtBBMuVgLGcWU9h9HIO9mKAc2m8Cm1
jUE5lpzRjTecd9I2qilMUTwtgnw42auSCzbUeYMURPtDqyQ7p6AjMujp9EPemcSVOK9vXYL0Ptco
xW9MC0dtV6iPkCN7gOqiZXPRKaFbWADp16p8UAlvS/a5XXk5jwKbgQCKkpHi2EISh1uRkxhlyjWC
iDCiK6JBRsMvpLbc0v5dKwP5alo1fmdR5PJaV2qvZSj5CYNpMAy1/EDNTY5OSIJU+0KFmQbyhsbm
rdLNLDL4+TcnT7c62/aH01ohYaf/VCbRhtLIBfqGoQc7+sAc8vmKkesnF7CqCEKdyF/dhrxYdQKB
gC0iZzzNAapayz1+JcVTwwEid6j9JqNXbBc+Z2YwMi+T0Fv/P/hwkX/ypeOXnIUcw0lh/YtGBVAC
DQbsz7LcY1HqXiHKYNWNvXgwwO+oiChjxvEkSdsTTIfnK4VSCvU9BxDbQHjdiNDJbL6oar92UN7V
rBYvChJZF7LvUH4YmVpHAoGAbZ2X7XvoeEO+uZ58/BGKOIGHByHBDiXtzMhdJr15HTYjxK7OgTZm
gK+8zp4L9lbuLGDmJO8vft32XPEWuvl8twCzFH+CsWLQADZMzKsBasOZ/h1FwhdMgCMcY+Qlzd4
JZKjTSu3i7vhvx6RzdSedXEMNTZWN4qllx3kR5aHcukCgYA9T+Zrvm1F0seQPbLknn7EqhXlJbaT
P8TTvW/6bdPi23ExzxZn7KODrfclYRph1LHMpAONv/x2xALIf91UB+v5ohy1oDoasL0gij1houRe
2ERKKdwz0ZL9SWq6VTdhr/5G994CK72fy5WhyERbDjUldHaK3M849JJuf8cSrvSb4g==
-----END RSA PRIVATE KEY-----
```

Сохраните его в любой каталог, однако помните, что для того чтоб им воспользоваться, вам придется указать полный путь. Если вы используете OpenSSH вы должны явно разрешить доступ к этому файлу.

Linux:

```
$ chmod 600 id_rsa-gsg-keypair ; ls -l id_rsa-gsg-keypair
-rw----- 1 fred flintstones 1701 Jun 19 17:57 id_rsa-gsg-keypair
```

Note: Если вы используете PuTTY, вам потребуется сконвертировать ваш закрытый ключ в формат понятный PuTTY. См. [Chapter 8, Appendix: PuTTY](#)

Запуск экземпляра

Зная AMI ID, вы можете запускать экземпляр

```
PROMPT> ec2-run-instances ami-5bae4b32 -k gsg-keypair  
RESERVATION r-fea54097 495219933132 default  
INSTANCE i-10a64379 ami-5bae4b32 pending gsg-keypair 0
```

Идентификатор экземпляра во втором поле уникален и используется для управления.

ВАЖНО: После запуска экземпляра вам начисляется стоимость использованного процессорного времени. После использования убедитесь что вы остановили экземпляр Для справки обращайтесь к разделу [“Остановка экземпляров”](#).

Запуск займет некоторое время, вы можете получить информацию о состоянии командой:

```
PROMPT> ec2-describe-instances i-10a64379  
RESERVATION r-fea54097 495219933132  
INSTANCE i-10a64379 ami-5bae4b32 ec2-72-44-33-55.z-2.compute-1.amazonaws.com domU-12-34-31-00-00-05.z-2.compute-1.internal running gsg-keypair 0
```

Когда значение поля перед полем ключевой пары принимает значение "running" – экземпляр начинает загрузку, и некоторое время еще может быть недоступен для работы через Internet. Первое доменное имя – это внешнее имя для обращения через Интернет, второе – локальное, используемое для связи между экземплярами внутри Amazon EC2.

Авторизация сетевого трафика

Перед тем как связаться с экземпляром, необходимо явно разрешить сетевой трафик к нему.

```
PROMPT> ec2-authorize default -p 22  
PERMISSION default ALLOWS tcp 22 22 FROM CIDR 0.0.0.0/0  
PROMPT> ec2-authorize default -p 80  
PERMISSION default ALLOWS tcp 80 80 FROM CIDR 0.0.0.0/0
```

Если не указано иное, ваш экземпляр запускается в контексте безопасности группы default. Первая команда разрешает трафик на порт SSH (22). Вторая – на порт HTTP (80). За деталями обращайтесь к руководству разработчика «Amazon EC2 Developer Guide».

Связь с экземпляром

После авторизации вы можете соединиться с экземпляром. Запустите веб-браузер и наберите `http://<hostname>/`, где `<hostname>` - внешнее имя узла вашего экземпляра, возвращенное `ec2-describe-instances`, `ec2-72-44-33-55.z-2.compute-1.amazonaws.com` в примере выше. Откроется страница входа в ваш экземпляр.

Note: Если вы получили ошибку что время ожидания истекло, возможно ваш экземпляр запустился неудачно, или еще запускается.

Теперь вы можете входить в экземпляр и осуществлять полный контроль.

```
PROMPT> ssh -i id_rsa-gsg-keypair root@ec2-72-44-33-55.z-2.compute-1.amazonaws.com
The authenticity of host 'ec2-72-44-33-55.z-2.compute-1.amazonaws.com (216.182.225.42)' can't be
established.
RSA key fingerprint is 9d:0a:a9:51:32:38:11:83:a8:a7:1c:9c:1c:e6:58:b0.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'ec2-72-44-33-55.z-2.compute-1.amazonaws.com' (RSA) to the list of known
hosts.
Last login: Wed Jun 21 08:02:08 2006
root@my-instance #
```

Очистка

Откат регистрации AMI

Если вам больше не нужен конкретный AMI, вы можете снять регистрацию.

Вам потребуются идентификатор ключа доступа и закрытый ключ доступа .

Откат регистрации осуществляется командой:

```
PROMPT> ec2-deregister ami-5bae4b32  
IMAGE ami-5bae4b32
```

Удаление AMI из Amazon S3

Вы можете удалить AMI из Amazon S3, чтобы не платить за хранение.

Удаление осуществляется командой:

```
PROMPT> ec2-delete-bundle -b <your-s3-bucket> -p image -a <aws-access-key-id> -s <aws-secret-access-key>  
Deleting AMI bundle parts from https://s3.amazonaws.com/<your-s3-bucket>...  
Deleted image.part.00.  
Deleted image.part.01.  
Deleted image.part.02.  
Deleted image.part.03.  
...  
Deleted image.part.23.  
Deleted image.part.24.  
Delete Bundle complete.
```

Note: Вы можете использовать любой инструмент, рассчитанный на работу с S3 чтобы удалить ненужные AMI.

Остановка экземпляров

После запуска экземпляра вам начисляется оплата за потребляемые ресурсы. После того как у вас отпала необходимость в конкретном экземпляре, вы можете его остановить, чтобы более не оплачивать процессорное время. Для остановки вам потребуется знать идентификатор экземпляра, полученный при запуске.

```
PROMPT> ec2-terminate-instances i-10a64379  
INSTANCE i-10a64379 running shutting-down
```

Это может занять некоторое время. Вы можете узнать состояние экземпляра в любой момент как описано в разделе [“Запуск экземпляров”](#).

Кроме того, вы можете остановить ваш экземпляр командой оболочки.

```
# /sbin/shutdown -h now
```