vscalecommunity

Начальная настройка сервера под управлением ОС Ubuntu

Руслан
6 октябрь 2016 г. 18:33

Введение

После создания нового сервера необходимо предпринять несколько шагов по его базовой настройке. Это повысит безопасность и удобство использования Вашего сервера и заложит прочную основу для последующих действий.

Шаг 1 - Логин с учетной записью root

Для того, чтобы осуществить вход на Ваш сервер, Вам необходимо знать публичный IP-адрес сервера и пароль учетной записи пользователя ***root***. Это нужно для подключения к серверу по SSH.

Если Вы ещё не зашли на сервер, зайдите под учетной записью `root` при помощи следующей команды (замените *SERVER_IP_ADDRESS* на публичный IP-адрес Вашего сервера):

```
ssh root@SERVER_IP_ADDRESS
```

Завершите процесс входа, приняв предупреждение о подлинности хоста (host authenticity), если оно возникнет, а затем идентифицируя себя как ***root*** пользователя (с помощью пароля или секретного ключа). Если вы впервые заходите на сервер с использованием пароля, Вам будет предложено изменить пароль учетной записи ***root***.

Об учетной записи root

Пользователь ***root*** является администратором в среде Linux и имеет очень широкий набор привилегий (прав). Из-за повышенных привилегий root-аккаунта не

рекомендуется пользоваться этой учетной записью на регулярной основе. Причиной этого является возможность случайно внести в систему деструктивные изменения.

Следующий шаг заключается в создании альтернативной пользовательской учетной записи с ограниченными привилегиями для повседневной работы. Мы продемонстрируем, как при необходимости получить расширенные полномочия во время использования этой учетной записи.

Шаг 2 - Создание нового пользователя

Осуществив вход с помощью учетной записи root-пользователя вы можете создать новую учетную запись, которую можно будет использовать для входа на сервер в дальнейшем.

В этом примере мы создаем новую учетную запись пользователя с именем "demo". Вы можете придумать другое имя своей учетной записи, заменив "demo":

```
adduser demo
```

Система предложит вам установить пароль для нового пользователя и ввести дополнительную информацию.

Задайте надежный пароль. Вводить дополнительную информацию не обязательно. Вы можете просто нажать "ENTER" в любом поле, которое хотите пропустить.

Шаг 3 - Привилегии пользователя "root"

Теперь у нас есть новая учетная запись по стандартными привилегиями. Однако иногда нам может потребоваться выполнять задачи с привилегиями администратора.

Во избежание необходимости выхода из-под учетной записи обычного пользователя и входа с учетной записью ***root***-пользователя, мы можем настроить возможность использования режима так называемого "супер-пользователя", в котором наша обычная учетная запись временно получает привилегии ***root***-пользователя. Это позволит нашему обычному пользователю выполнять команды с привилегиями администратора с помощью добавления слова ***sudo*** перед каждой командой.

Чтобы добавить эти привилегии нашей новой учетной записи, необходимо добавить ее в группу "sudo". По умолчанию, в Ubuntu 14.04 пользователи, входящие в группу "***sudo***", могут использовать команду ***sudo***.

Из-под ***root***-пользователя выполните следующую команду для добавления Вашего нового пользователя в группу "sudo" (замените "demo" на имя вашей новой учетной записи):

```
gpasswd -a demo sudo
```

Теперь ваш пользователь сможет выполнять команды с привилегиями супер-пользователя!

Шаг 4 - Добавление авторизации по публичному ключу (Public Key Authentication) (Рекомендуется!)

Следующий шаг в усилении безопасности Вашего сервера - это настройка авторизации по публичному ключу для Вашего нового пользователя. Данная настройка повысит безопасность Вашего сервера, требуя секретный SSH ключ для входа.

Создание пары ключей

Если у Вас ещё нет пары SSH-ключей, которая состоит из публичного (открытого) и секретного (закрытого) ключей, Вам необходимо её создать. Если у Вас уже есть ключ, который Вы хотите использовать, перейдите к подразделу "Копирование публичного ключа".

Чтобы создать новую пару ключей, выполните следующую команду в терминале на Вашей **локальной машине** (т.е. на Вашем компьютере):

```
ssh-keygen
```

Если Ваш локальный пользователь называется "localuser", Вы увидите вывод следующего вида:

```
Generating public/private rsa key pair.
Enter file in which to save the key (/Users/localuser/.ssh/id_rsa):
```

Нажмите "ENTER", чтобы согласиться с адресом и именем файла (или введите другой адрес/имя файла).

Далее Вам будет предложено ввести кодовую фразу для защиты ключа. Вы можете ввести кодовую фразу или оставить ее пустой.

Обратите внимание: Если вы оставите кодовую фразу пустой, то сможете использовать приватный ключ для авторизации без ввода кодовой фразы. Если вы зададите кодовую фразу, вам потребуется и приватный ключ, и кодовая фраза для входа. Добавление кодовой фразы к ключам повышает уровень безопасности, но оба

метода имеют свои области применения и являются более безопасными, чем базовая авторизация паролем.

В результате этого в поддиректории ***.ssh*** домашней директории пользователя *localuser* будет создан секретный ключ ***id_rsa*** и публичный ключ ***id_rsa.pub***. Не передавайте секретный ключ никому, кто не должен иметь иметь доступ к вашим серверам!

Копирование публичного ключа

После создания пары SSH-ключей вам необходимо скопировать публичный ключ на Ваш новый сервер.

Если вы создали пару SSH-ключей, как описано в предыдущем пункте, выполните следующую команду в терминале на Вашей **локальной машине** для печати публичного ключа (`id_rsa.pub`):

```
cat ~/.ssh/id_rsa.pub
```

В результате выполнения данной команды на экран будет выведен ваш публичный SSH-ключ. Он выглядит примерно так:

```
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQDBGTO0tsVejssuaYR5R3Y/i73SppJAhme1dH7W
```

Выделите публичный ключ и скопируйте его в буфер обмена.

Добавление публичного ключа к новому удаленному пользователю

Чтобы сделать возможным использование SSH-ключа для авторизации под учетной записью нового удаленного пользователя (remote user), вам необходимо добавить публичный ключ в специальный файл в домашней директории этого пользователя.

На сервере, осуществив вход с учетной записью ***root***-пользователя, выполните следующие команды для переключения на нового пользователя (замените ***demo*** на ваше имя пользователя):

```
su - demo
```

Теперь вы находитесь в домашней директории нового пользователя.

Создайте новую директорию под названием ***.ssh*** и ограничьте права на доступ к ней при помощи следующих команд:

```
mkdir .ssh
chmod 700 .ssh
```

Теперь откройте файл в директории ***.ssh*** с названием ***authorized_keys*** в текстовом редакторе. Мы будем использовать ***nano*** для редактирования файла:

```
nano .ssh/authorized_keys
```

Далее добавьте Ваш публичный ключ (который должен быть в буфере обмена) путем вставки в текстовый редактор.

Нажмите ***CTRL-X*** для закрытия файла, затем ***Y*** для сохранения внесенных изменений, затем ***ENTER*** для подтверждения имени файла.

Теперь ограничьте права на доступ к файлу ***authorized_keys*** при помощи следующей команды:

```
chmod 600 .ssh/authorized_keys
```

Введите следующую команду *один* раз для возврата к пользователю ***root***.

```
exit
```

Теперь dы можете заходить на сервер по SSH с учетной записью Вашего нового пользователя, используя секретный ключ для авторизации.

Шаг 5 - Настройка SSH

Теперь, когда у вас есть новый аккаунт, вы можете ещё больше обезопасить наш сервер путем изменения конфигурации его SSH (программа для удаленного подключения).

Начните с открытия конфигурационного файла в текстовом редакторе под пользователем ***root***:

```
nano /etc/ssh/sshd_config
```

Смена SSH-порта (опционально)

Первая настройка, которую Вы вероятно захотите сменить, это порт, на котором работает SSH. Найдите строчку, которая выглядит похожим образом:

```
Port 22
```

Если мы изменим этот номер на что-нибудь между 1025 и 65536, то SSH на нашем сервере будет ждать соединений на другом порту. Иногда это полезно, поскольку злоумышленники иногда пытаются попасть на сервер путем атаки через SSH. Если вы измените порт SSH, им придется сделать лишний шаг для его определения.

Если вы изменяете номер порта, вам придется помнить, что SSH на Вашем сервере работает на новом порту. В этом руководстве мы изменим порт на ***4444*** для демонстрации. Это означает, что для подключения нам необходимо будет указать SSH клиенту использовать новый порт вместо порта, используемого по умолчанию. Как это сделать, мы рассмотрим чуть позже. Измените порт на значение по вашему выбору:

```
Port 4444
```

Запрет входа под root

Затем нам необходимо найти следующую строчку:

```
PermitRootLogin yes
```

Это настройка позволяет отключить возможность входа на сервер с помощью учетной записи ***root***-пользователя через SSH. Это повышает безопасность сервера, поскольку теперь мы можем осуществлять вход на наш сервер с помощью учетной записи обычного пользователя и повышать полномочия, когда это требуется.

Для отключения возможности входа на сервер с помощью учетной записи ***root*** измените строчку следующим образом (замените "yes" на "no"):

```
PermitRootLogin no
```

Отключение возможности удаленного доступа с помощью ***root***-пользователя настоятельно рекомендуется для всех серверов!

После окончания внесения изменений сохраните и закройте файл так же, как мы делали ранее (***CTRL-X***, затем ***Y***, затем ***ENTER***).

Шаг 6 - Перезапуск SSH

После внесения изменений необходимо перезапустить сервис SSH, чтобы он начал использовать новую конфигурацию.

Выполните следующую команду для перезапуска SSH:

```
service ssh restart
```

Теперь, перед тем как выйти с сервера, **протестируйте** новую конфигурацию.

Откройте **новое** окно терминала. В новом окне необходимо открыть новое соединение с сервером. В этот раз вместо ***root***-аккаунта вы будете использовать новый аккаунт, созданный ранее.

Если вы изменили номер порта, на котором работает SSH, необходимо сообщить об этом SSH-клиенту. Сделать это можно при помощи синтаксиса ***-p 4444***, где "4444" - заданный вами номер порта.

Для сервера, который мы настраивали выше, будем использовать следующую команду (замените параметры вашими, где это необходимо):

```
ssh -p 4444 demo@SERVER_IP_ADDRESS
```

Обратите внимание: Если вы используете PuTTY для подключения к вашим серверам, не забудьте изменить номер порта, чтобы он соответствовал текущей конфигурации сервера.

Система запросит пароль нового пользователя. После этого вы войдёте на сервер под его учётной записью.

Помните, что если вам необходимо выполнить команду с привилегиями ***root***-пользователя, перед ней понадобитс поставить слово ***sudo*** :

```
sudo command_to_run
```

Если все нормально, можно завершить ваши сессии следующей командой:

```
exit
```

Что дальше?

Теперь у вас есть хорошо настроенный сервер и можно устанавливать любое необходимое вам программное обеспечение. Также можно использовать дополнительные настройки сервера - например ,включить ***fail2ban*** для снижения эффективности bruteforce-атак на сервер, задать базовые настройки файервола, NTP и swap-файлы. Но подробно мы рассмотрим это в следующем руководстве.

Server

1

Комментарии

1 комментарий Сортировка

Никита Коновалов
7 октябрь 2016 г. 16:34

2

Отключаем авторизацию по паролю:

```
sudo nano /etc/ssh/sshd_config
```

Находим строчку: Используем поиск для быстроты (ctrl+W)

```
PasswordAuthentication yes
```

И заменяем на:

```
PasswordAuthentication no
```

Войдите в службу, чтобы оставить комментарий.

Не нашли то, что искали?

Задать вопрос