СОЗДАНИЕ ЗАШИФРОВАННОЙ ФАЙЛОВОЙ СИСТЕМЫ В ТОМЕ БЛОЧНОГО ХРАНИЛИЩА

Апрель 2, 2018 12:40 пп 450 views | Комментариев нет
VPS | Amber | 0 Comments

Блочные хранилища SSD позволяют расширить возможности хранения данных в вашей инфраструктуре, не изменяя размера самого сервера.

Тома хранилища шифруются, а это значит, что данные на томе нельзя прочитать вне хранилища. Когда вы монтируете том на сервере, сервер может дешифровать блочное устройство хранения; все данные передаются по изолированным сетям.

Для большей безопасности в томе можно также создать файловую систему на зашифрованном диске LUKS. Это значит, что операционная система должна расшифровать диск, чтобы прочитать данные.

Данный мануал научит вас:

- Создавать в томе защищенный паролем зашифрованный диск для хранения файловой системы.
- Вручную монтировать и демонтировать зашифрованную файловую систему.
- Автоматически монтировать файловую систему при запуске сервера.

Требования

Для работы вам понадобится виртуальный сервер и монтированный том.

Читайте также: Управление накопителями в Linux: основные понятия и подходы

Важно! Этот процесс может уничтожить все данные тома. Используйте новый, пустой том, либо создайте резервную копию данных тома, прежде чем приступать к работе.

1: Создание шифрованного диска

cryptsetup – это утилита для управления томами LUKS. Для начала используйте cryptsetup для инициализации зашифрованного диска в томе.

```
sudo cryptsetup -y -v luksFormat /dev/disk/by-id/scsi-Volume_volume-lon1-01
```

Обязательно замените volume-lon1-01 названием вашего тома. Флаг -y потребует дважды ввести парольную фразу, когда вам будет предложено ее создать. Флаг -v выдает удобочитаемый вывод для проверки результатов команды.

На выходе будет предложено подтвердить перезапись данных в томе. Введите YES капсом, затем нажмите Enter, чтобы продолжить.

```
WARNING!

========

This will overwrite data on /dev/disk/by-id/scsi-Volume_volume-lon1-01
irrevocably.

Are you sure? (Type uppercase yes): YES
```

Затем на выходе будет предложено создать парольную фразу для зашифрованного диска. Введите уникальную парольную фразу и подтвердите ее. Эта парольная фраза не восстанавливается, поэтому храните ее в надежном месте.

```
. . .

Enter passphrase:

Verify passphrase:

Command successful.
```

Если нужно, вы можете изменить эту парольную фразу в будущем с помощью команды cryptsetup luksChangeKey. Вы также можете добавить до 8 дополнительных парольных фраз для каждого устройства с помощью cryptsetup luksAddKey.

Зашифрованный диск готов. Затем расшифруйте его и присвойте ему метку, чтобы на него было удобно ссылаться. В мануале используется метка secure-volume.

Читайте также: Управление накопителями в Linux: основные понятия и подходы

```
sudo cryptsetup luksOpen /dev/disk/by-id/scsi-Volume_volume-lon1-01
secure-volume
```

Вам будет предложено ввести парольную фразу. После того, как вы введете ее, том станет отображаться в /dev/mapper/secure-volume.

Чтобы убедиться, что все работает, проверьте состояние зашифрованного диска.

```
cryptsetup status secure-volume
```

В выводе вы увидите тип и метку тома:

```
/dev/mapper/secure-volume is active.

type: LUKS1

cipher:  aes-xts-plain64

keysize: 256 bits

device:  /dev/sda

offset:  4096 sectors

size: 209711104 sectors

mode: read/write
```

Теперь у вас есть защищенный паролем шифрованный диск. Следующим этапом будет создание файловой системы на этом диске, чтобы операционная система могла использовать ее для хранения файлов.

2: Создание и монтирование файловой системы

Сначала посмотрите на текущее доступное дисковое пространство сервера.

```
df -h
```

Вы увидите примерно такой вывод:

```
Filesystem   Size  Used Avail Use% Mounted on
udev             2.0G   0  2.0G   0% /dev
tmpfs            396M  5.6M  390M   2% /run
/dev/vda1        78G  877M   77G   2% /
tmpfs            2.0G   0  2.0G   0% /dev/shm
tmpfs            5.0M   0  5.0M   0% /run/lock
tmpfs            2.0G   0  2.0G   0% /sys/fs/cgroup
/dev/vda15       105M  3.4M  101M   4% /boot/efi
tmpfs            396M   0  396M   0% /run/user/1000
```

Сейчас /dev/mapper/secure-volume не отображается в этом списке, потому что том еще не доступен для сервера. Чтобы сделать его доступным, необходимо создать и смонтировать файловую систему.

Используйте утилиту mkfs.xfs (make file system) для создания файловой системы XFS в томе.

```
sudo mkfs.xfs /dev/mapper/secure-volume
```

Когда файловая система будет создана, вы сможете смонтировать ее, что означает, что она будет доступна для операционной системы вашего сервера.

Создайте точку монтирования, к которой будет прикреплена файловая система. Для точки монтирования хорошо подходит пустой каталог в каталоге /mnt, например, /mnt/secure.

```
sudo mkdir /mnt/secure
```

Смонтируйте файловую систему:

```
sudo mount /dev/mapper/secure-volume /mnt/secure
```

Чтобы убедиться, что все работает, проверьте доступное дисковое пространство сервера:

```
df -h
```

Теперь в списке появится /dev/mapper/secure-volume:

```
Filesystem              Size  Used Avail Use% Mounted on
udev                    2.0G  0  2.0G   0% /dev
tmpfs                   396M  5.6M  390M   2% /run
/dev/vda1               78G  877M   77G   2% /
tmpfs                   2.0G  0  2.0G   0% /dev/shm
tmpfs                   5.0M  0  5.0M   0% /run/lock
tmpfs                   2.0G  0  2.0G   0% /sys/fs/cgroup
/dev/vda15              105M  3.4M  101M   4% /boot/efi
tmpfs                   396M  0  396M   0% /run/user/1000
/dev/mapper/secure-volume  100G   33M  100G   1% /mnt/secure
```

Это означает, что зашифрованная файловая система подключена и доступна для использования.

Когда данные в томе станут ненужными, вы можете размонтировать файловую систему и заблокировать зашифрованный диск.

```
sudo umount /mnt/secure
```

```
sudo cryptsetup luksClose secure-volume
```

С помощью df –h вы можете убедиться, что файловая система больше недоступна. Чтобы снова сделать данные тома доступными, нужно снова открыть диск (cryptsetup luksOpen…), создать точку монтирования и смонтировать файловую систему.

Чтобы этот процесс не пришлось повторять вручную каждый раз, когда вы хотите использовать том, настройте автоматическое монтирование файловой системы при загрузке сервера.

3: Автоматическое монтирование файловой системы

Зашифрованный диск может поддерживать до 8 парольных фраз. На этом заключительном этапе нужно создать ключ и добавить его в качестве парольной фразы. Затем вы сможете пользоваться этим ключом, чтобы настроить расшифровку и монтирование тома по мере загрузки сервера.

Создайте файл ключа в /root/.secure_key. Эта команда создаст файл размером 4 КБ со случайным содержимым:

```
sudo dd if=/dev/urandom of=/root/.secure-key  bs=1024 count=4
```

Заблокируйте доступ к файлу; право на чтение файла должно быть только у пользователя root.

```
sudo chmod 0400 /root/.secure-key
```

Добавьте ключ как парольную фразу для зашифрованного диска:

```
cryptsetup luksAddKey /dev/disk/by-id/scsi-Volume_volume-lon1-01
/root/.secure-key
```

Вам будет предложено ввести парольную фразу. Вы можете ввести фразу, которую выбрали во время создания диска.

/etc/crypttab – это конфигурационный файл, который позволяет монтировать зашифрованные диски при запуске системы. Откройте этот файл с помощью nano или другого текстового редактора.

```
sudo nano /etc/crypttab
```

Добавьте следующую строку в конец файла, чтобы автоматически монтировать том:

```
. . .

secure-volume /dev/disk/by-id/scsi-Volume_volume-lon1-01 /root/.secure-key
luks
```

Строки в файле /etc/crypttab придерживаются формата:

```
имя_устройства путь_устройства путь_ключа опции
```

В данном случае имя устройства – secure-volume, путь – /dev/disk/by-id/…, файл ключа – это тот, который вы только что создали в /root/.secure_key, а в опциях включается шифрование luks.

Сохраните и закройте файл.

Файл /etc/fstab автоматизирует монтирование. Откройте его:

```
sudo nano /etc/fstab
```

Добавьте в конец файла следующую строку для настройки автоматического монтирования диска при загрузке.

```
. . .
```

```
/dev/mapper/secure-volume /mnt/secure xfs  defaults,nofail 0 0
```

Первые три аргумента строк /etc/fstab – это параметры:

```
путь_устройства точка_монтирования тип_файловой_системы
```

Здесь используется тот же путь к устройству и точка подключения, что и в разделе 2. Тип файловой системы – XFS. Вы можете прочитать о других параметрах fstab в справке (команда man fstab).

Сохраните и закройте файл. Теперь зашифрованная файловая система будет автоматически монтироваться при загрузке сервера. Вы можете проверить это, перезагрузив свой сервер, но будьте осторожны со всеми запущенными сервисами.