

BACULA



Автор: Денис Горбатюк



Данная статья применима к:

- ОС CH Смоленск 1.6
- ОС CH Ленинград 8.1
- ОС ОН Орёл 2.12

- Введение
- Описание стенда
- Основные понятия, используемые в системе Bacula
 - Типы резервного копирования
 - Журнал сообщений Bacula
 - Главный конфигурационный файл
 - Хранилища, Пулы, Тома и Метки томов
 - Список источников использованных при написании данной статьи
- Bacula: Установка
- Bacula: Базовая настройка
- Bacula: Настройка
 - Bacula Director: настройка
 - Bacula Storage: настройка
 - Bacula FileDaemon: настройка
- Bacula: Проверка работоспособности.
- Bacula: BackUp and Restore
 - Bacula: Консольный способ
 - Bacula: С помощью графики

Введение

Данная статья предназначена для ознакомления с функционалом и логикой работы сетевого инструмента резервного копирования **BACULA**.

Все действия осуществляются в операционной системе **Astra Linux Special Edition Смоленск 1.6** с установленным обновлением безопасности № **20190222SE16**.



Для выполнения административных действий в операционной системе особого назначения Astra Linux SE 1.6 администратор должен выполнить вход с минимальным (нулевым) уровнем конфиденциальности и с максимальным уровнем целостности, определённым в ОС.



В других операционных системах и версиях ОС CH Astra Linux кроме как 1.6 с обновлением безопасности № **20190222SE16** не проверялось. Вероятность того, что будет работать в других версиях ОС CH Astra Linux - весьма высока (**но не гарантирована**).



Хочу отметить что данный материал необходимо изучить с самого начала и до самого конца, внимательно вчитываясь в каждый (!) абзац.

Объясню почему:

Начав изучать Bacula, находя в Интернет тематические статьи, я совершал одну большую ошибку следующего характера: открывая материал и видя «много текста» мотал колесом мыши до ближайших кракозябр в виде листинга файлов настройки и/или скриптов, пытаясь понять их в процессе эксплуатации. И впоследствии раз за разом натываясь на ошибки, понял, что что-то делаю не так. А именно не понимаю логики работы данного продукта.

НЕ ПОВТОРЯЙТЕ МОИ ОШИБКИ.

Освоение данного продукта с точки зрения понимания настроек, функционала и логики работы изначально лучше начинать на виртуальных машинах (далее - ВМ).

Не важно на каких ОСях используемых в качестве хостовых и с использованием каких средств виртуализации, подразумевается пошаговость выполнения действий и возможность отката к предыдущим шагам (в случае ошибок или неудач) благодаря сделанным вовремя снимкам ВМ.

Последнее, что хотелось бы добавить после всех лирических отступлений и перед тем как мы перейдем к конкретике это то, что все препятствия и сложности освоения любых софтин находятся лишь у нас в голове.

Нам ИТшникам и не только ИТшникам зачастую мешает навязанная руководителями необходимость выполнения других задач, нехватка времени и/или желания, лень и прочие факторы, а также пресловутые «более приоритетные задачи», мне кажется, что не только я сталкивался с этой как минимум противной и режущей слух формулировкой.

Так вот:

Уделите день/два/три или любое другое количество часов/дней личного/рабочего времени, не отвлекаясь/переключаясь на другие задачи/мысли и вы (я очень надеюсь) сможете освоить данное ПО.

Описание стенда

При установке и настройке данного продукта подразумевается "**чистый**" эксперимент:

"Чистые" ОС CH [Astra Linux Special Edition релиз Смоленск 1.6](#) в виртуальных машинах с установленным обновлением безопасности № 20181229SE16 и [B ACULA](#) версии 7.4.4.

Для развёртывания продукта использовались три виртуальные машины:

1. **dir** - [192.168.244.70](#) (**DirectorDaemon**, она же главная или **Директор**);
2. **stor** - [192.168.244.80](#) (**StorageDaemon**, она же **Хранилище**);
3. **client** - [192.168.244.90](#) (**FileDaemon**, она же **Клиент**).

Основные понятия, используемые в системе Bacula

Задача (Job) – операция создания архива файлов или восстановления информации из архива.

Том (Volume) – единица хранения информации, представляет собой отдельный файл, магнитную ленту, CD, DVD. Один том может содержать несколько задач, или одна задача может занимать несколько томов.

Пул (Pool) – группа томов. Пул объединяет в себе несколько томов, чтобы отдельная резервная копия не была ограничена объемом тома.

Набор файлов (FileSet) – списки директорий и отдельных файлов, которые должны быть заархивированы. Может содержать регулярные выражения.

Уведомление (Messages) – информация о состоянии компонентов Bacula и задач. Могут передаваться по электронной почте, либо записываться в журнал.

Файл начальной загрузки (bootstrap) – специальный текстовый файл, содержащий информацию об архивированных файлах и томах. Этот файл используется во время операции восстановления.

Типы резервного копирования

- Полное копирование (**Full backup**) – копирование данных в полном объеме. Самый надежный способ копирования. В случае выхода из строя свежей копии данные можно восстановить из предыдущих копий. Эффективный и быстрый метод восстановления. Недостаток – требует носителей большого объема и длительного времени выполнения.
- Дифференциальное копирование (**Differential backup**) – копируются файлы, изменившиеся после последнего Full backup. Данные копируются «нарастающим итогом», так что последняя копия всегда будет содержать все изменения с момента последнего Full backup. Выполняется быстрее чем Full backup, и повреждение одной из копий не приводит к потере всех данных за последующий и предыдущий период (при наличии живого Full backup). Так или иначе требуется регулярный Full backup и бывает, что последняя копия (при длительной работе) по размеру изменений приближается к Full backup.
- Инкрементное копирование (**Incremental backup**) – выполняется копирование только информации, измененной после выполнения предыдущего Incremental backup. Это самый быстрый метод резервирования и занимает меньше всего объема, но и самый ненадежный метод. В случае повреждения одной из копий все последующие становятся бесполезны. И соответственно при повреждении Full backup все становится негодным. Восстановление данных занимает продолжительное время.

Предпочтителен Full backup, но это «дорого» обходится. Если данные не очень ценны или ресурсов мало, то используйте Incremental backup. Differential backup это компромисс, но бывает, что последняя копия приближается к объему полного копирования так что, следите за ним.

Журнал сообщений Bacula

Bacula поддерживает следующие типы сообщений:

- информационное (info)
- предупреждение (warning)
- ошибка (error)
- критическая ошибка (fatal)
- остановка (terminate)
- перечень пропущенных файлов (skipped);
- список файлов, сохраненных без ошибок (saved)

- список файлов, которые не удалось сохранить (notsaved)
- перечень восстановленных файлов (restored)
- потребность подключить новый том (mount)
- ошибка авторизации (security)
- все типы (all).

Для отправки сообщений Bacula использует собственное средство – bsmtp, которое может отправлять сообщения не только локальным пользователям, но и на удаленный smtp-сервер.

Главный конфигурационный файл

Главный конфигурационный файл Director Daemon **bacula-dir.conf**, как и остальные конфигурационные файлы Bacula, состоит из логических разделов, описывающих отдельные ресурсы.

Каждый логический раздел ограничен фигурными скобками {}.

В ключах регистр и пробелы полностью игнорируются. Поэтому ключи: name, Name, и "N a m e" полностью идентичны.

Пробелы до и после знака «равно» игнорируются. Если «значение» содержит пробелы, оно должно быть заключено в двойные кавычки, а пробелы должны быть экранированы обратным слешем.

Для каждого компьютера, файлы которого архивируются, рекомендуется создавать отдельные файлы настроек содержащие в себе ресурс Client, FileSet, Pool, Job Backup или Job Restore (Storage будет один для всех клиентов), и разделять их по папкам. Удобнее описывать эти ресурсы в отдельном файле/ах, который включается в bacula-dir.conf с помощью директивы @. Делается это в первую очередь для того, чтобы избежать разрастания одного конфигурационного файла до больших размеров, а так же для гибкости настройки. Например, ресурсы Client, FileSet, Pool, Job Backup и Job Restore для сервера, на котором установлена система Bacula, можно поместить в файл localhost.conf, и включить его в bacula-dir.conf:

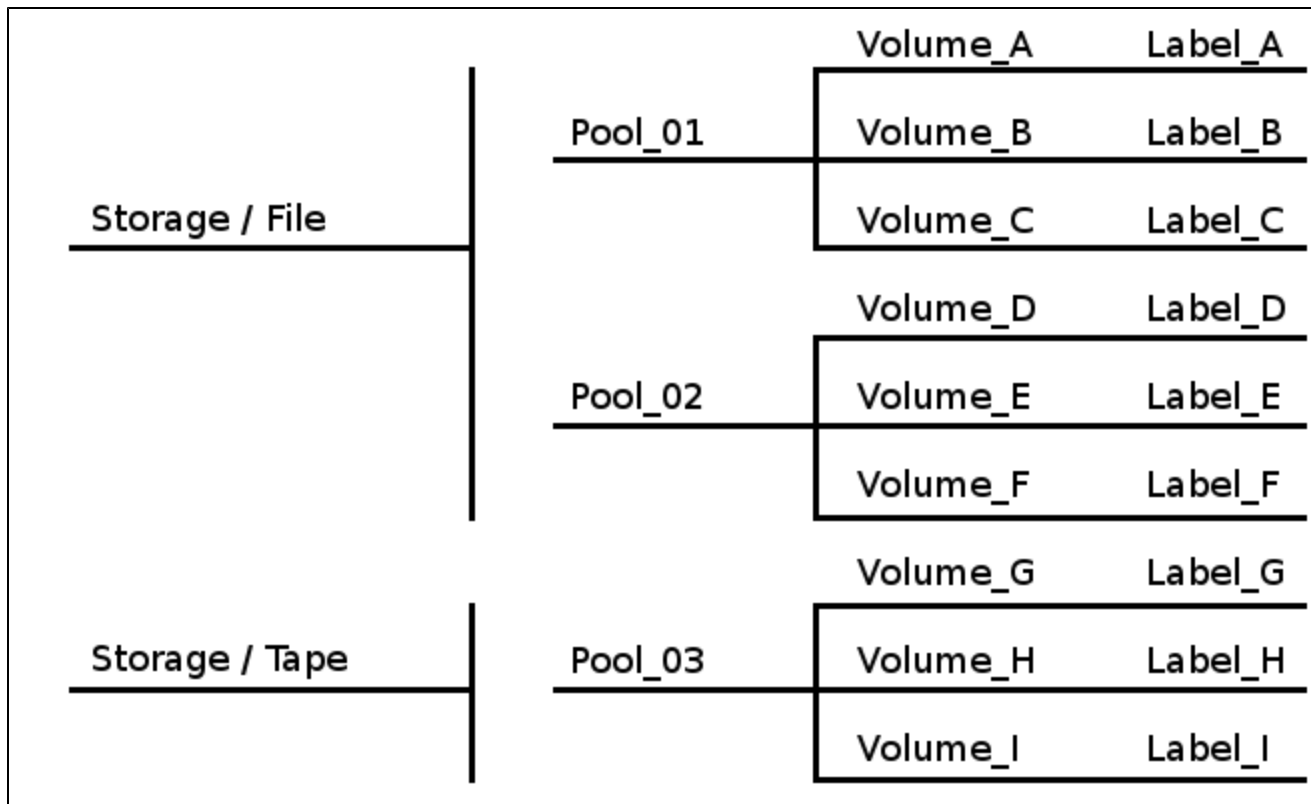
```
@/etc/bacula/conf.d/localhost.conf
```

Каждому файлу и каждой переменной присваивается понятное имя и далее в Job указываются необходимые имена переменных чтобы это было законченной сформированной задачей и могло корректно работать согласно ваших указаний. В конфигурационных файлах необходимо описать к примеру работу для одного клиента несколькими переменными такими как Job, Pool, Client, FileSet, Storage, Schedule.

Основной набор файлов рабочего клиента состоит из 6-ти элементов:

1. **Job** (Работа) — это объект которому присваивают понятное имя и это совокупность вещей и указаний как, кого, что и куда будем резервировать. По этому имени эту задачу можно определять, исполнять и вызывать из консоли.
2. **Client** (Клиент) — здесь указывается имя клиента для этой работы, сам клиент и его имя определяется отдельно.
3. **FileSet** (Набор файлов) — указывается набор файлов\директорий и их исключений для этой работы, сам набор файлов и его имя определяется отдельно.
4. **Schedule** (Планировщик) — планировщик для этой работы, в нем описываем периодичность работы и перераспределение ресурсов, это когда делать полное копирование, когда дифференциальное и тд. Сам планировщик и его имя определяется отдельно, и вы можете для удобства завести отдельный конфиг.
5. **Storage** (Хранилище) — Определение устройства хранения файлов для этой работы. В этом хранилище есть секция Device (см по имени) в котором есть указание, где будем хранить резервные копии.
6. **Pool** (Пул) — определяем для работы нужный нам Пул, это отдельное описание для каждого набора томов (лент, DVD, файлов). У каждого клиента будет свой пул – метки на тома в этом пуле будут выставляться согласно меткам для каждого пула клиента автоматически (опция **LabelFormat** в пуле клиента). Это возможно благодаря опции **LabelMedia = yes**; в файле bacula-sd.conf в настройках хранилища т.е. нет необходимости вручную размечать тома.

Хранилища, Пулы, Тома и Метки томов



- Том (**Volume**) – единица хранения информации, представляет собой отдельный файл, магнитную ленту, CD, DVD. Один том может содержать несколько задач, или одна задача может занимать несколько томов.
- Пул (**Pool**) – группа томов. Пул объединяет в себе несколько томов, чтобы отдельная резервная копия не была ограничена объемом тома.

Прежде чем bacula сможет использовать том (volume), том должен получить метку (label).

В случае, когда том является файлом, файл создаётся при нанесении метки и метка используется в качестве названия. Метки могут наноситься автоматически. Это определяется параметрами: "LabelMedia = yes", в секции device конфигурации sd и "Label Format = LableName-" в конфигурации пула

При этом к заданному формату будет прибавляться цифровой индекс: LableName-0001.

В формате метки можно использовать макросы, например:

```
"Label Format = LableName-${Year}_${Month}_${Day}"
```

Для задания (Job) определяется не том, а пул. При этом: При этом индекс добавляться не будет. И если, например, при проверке, будет выполнено более одного резервного копирования в день, то будет использован один и тот же том, если позволит параметр Maximum Volume Jobs.

- тома этого пула выбираются автоматически, и новые используются по мере необходимости;
- при отсутствии возможности использования тома, формируется запрос на подключение нужного.

Шаги по созданию Пула (Pool), добавлению Томов к нему, и запись программных меток на Тома, могут казаться утомительными сначала, но фактически они весьма просты, и позволяют использовать множество Томов (вместо того, чтобы быть ограниченным размером одной единственной ленты/диска /файла). Пулы также дают существенную гибкость в процессе резервирования. Например, можно иметь "Daily" (ежедневный) Пул Томов для инкрементальных (Incremental) резервных копий и "Weekly" (еженедельный) Пул Томов для полных (Full) резервных копий. Назначение соответствующих Пулов в ежедневных и еженедельных резервных Заданиях (Job) гарантирует, что никакое ежедневное Задание не переписывает данные в Еженедельном Пуле и наоборот. Кроме того, Bacula укажет, какой том необходимо подключить и когда.

Чтобы было проще ориентироваться в конфигурационных файлах, можно выносить часть конфигурации в отдельные файлы. Например, вся последующая конфигурация (Pool, File set, Schedule и Job) может быть вынесена в отдельный файл /etc/bacula/conf.d/testpool.conf, который можно подключить в конфигурацию директора (bacula-dir.conf) следующим образом:

```
@/etc/bacula/conf.d/testpool.conf
```

Теперь, описав пул (pool), набор файлов (fileset) и расписание (schedule), можно описать задание (Job) в котором они используются. Очевидно, что описывать всё нужно не всегда, можно использовать в разных заданиях одинаковые пулы, наборы файлов и расписания. Параметр Maximum Volume Jobs рекомендуется установить в значение 1. Это будет означать, что в рамках одного носителя данных могут быть размещены резервные данные, полученные в ходе выполнения только одного задания. При использовании файлов, рекомендуется придерживаться правила "один файл – одна копия", т.е. в одном

файле Bacula должны храниться резервные данные, которые были сформированы в рамках выполнения одного задания. Для каждого последующего выполнения будут создаваться новые файлы.

Для распределения резервного хранения по разным точкам необходимо:

1. В конфигурации Storage Daemon определить устройства (device), которые могут быть ленточными накопителями и/или каталогами на различных файловых системах (в случае если Media Type = File). В секции определения себя, sd определяется как storage. Не следует путать с последующими определениями storage в конфигурации директора с которыми мы далее будем работать.
2. В конфигурации Director Daemon определить storage, при описании задать:
 - a. Адрес sd к которому подключаться;
 - b. Устройство (device) на данном sd
 - c. Название, по которому этот storage будет использоваться в заданиях (job).
 - d. Описать пул (Pool)
 - e. Использовать описанный storage в задании (job), указав: Storage и Pool

Пулы не привязаны к хранилищам (storage) и любой пул может быть использован в любом хранилище.

То есть, достаточно на каждое устройство для записи (например, каталог с файлами) определить storage, после чего направлять произвольный пул на произвольное устройство.

Допустим, используется единственный Storage и он описан в шаблоне JobDefault, а все типы расписания описаны в файле defaultchedule.conf. Поэтому мы будем в клиентских файлах задавать только эти значения: Client, FileSet, Pool, Job Backup и Job Restore.

Список источников использованных при написании данной статьи

```
http://www.bacula.org/7.4.x-manuals/en/main/index.html
http://www.bacula.org/7.4.x-manuals/en/console/Console_Operators_Guide.html
https://ru.wikibooks.org/wiki/Bacula
https://ru.wikibooks.org/wiki/Bacula/Storage,_Volumes_and_Labels
https://kamaok.org.ua/?p=1517
https://habr.com/post/135291/
http://bog.pp.ru/work/bacula.html
https://www.altlinux.org/Bacula
```

Что читать дальше:

- [Bacula: Установка](#)
- [Bacula: Базовая настройка](#)
- [Bacula: Настройка](#)
 - [Bacula Director: настройка](#)
 - [Bacula Storage: настройка](#)
 - [Bacula FileDaemon: настройка](#)
- [Bacula: Проверка работоспособности.](#)
- [Bacula: BackUp and Restore](#)
 - [Bacula: Консольный способ](#)
 - [Bacula: С помощью графики](#)