Расширенное администрирование FreeBSD.

Блок 3

v 1.04

Оглавление

Служба точного времени

Принцип работы

Функциональность NTP основана на понятии главных серверов времени (называемых серверами первого эшелона), получающих сведения о точном времени из высокоточных источников, например от локально подключенной Глобальной системы рекогносцировки (GPS) или снимающих их с цезиевых часов.
Сервер, синхронизирующийся с сервером первого эшелона, называется сервером второго эшелона — эшелона исходного сервера + 1. По мере увеличения номера слоя точность времени может слегка снижаться. Принципиальными проблемами синхронизации времени являются учет сетевого ожидания и времени обработки пакетов и серверы с неточной установкой времени. Например, если сервер времени отправляет пакет «Точное время — 12:00:00, устано вите часы на 12:00:00», а пакету требуется 2 секунды на достижение места назначения, то часы на клиентском компьютере будут отставать на 2 секунды, Если на обработку пакета клиенту требуется еще 1 секунда, тогда клиентский компьютер будет отставать на 3 секунды.

NTP преодолевает эти проблемы несколькими способами:

- Измерением времени ожидания с помощью временных меток клиента и сервера;
- Учетом времени, необходимого на обработку сетевых пакетов;
- Использованием кратных выборок с множественных серверов для обеспечения точности;
- Составлением «черных списков» серверов, выдающих непоследовательные или неточные результаты.

NTP использует порт 123 UDP

В пакет входит следующее:

- *ntpq* для запроса серверов NTP;
- *ntpd* поддерживает точность локальных часов и (опционально) обеспечивает клиентам службу NTP;
- *ntptrace* прослеживает цепь сервера NTP к исходному серверу;
- *ntpdate* — одноразовая программа обновления часов.

Настройка на сервер точного времени

Для обеспечения большей точности часов сервера и снижения зависимости от доступности тех или иных серверов точного времени следует опрашивать пул серверов точного времени вместо одиночного сервера.

Онлайн список общедоступных серверов NTP
http://support.ntp.org/bin/view/Servers/WebHome

Синхронизация времени через NTP

Для того, чтобы время полученное с NTP-сервера соответствовало вашему часовому поясу необходимо установить корректную временную зону. Для Москвы это будет выглядеть следующим образом:

```
cp /usr/share/zoneinfo/Europe/Moscow /etc/localtime
```

Синхронизация времени через NTP

Ручная синхронизация

```
ntpdate time.nist.gov ntp.bmstu.ru
```

18 Aug 17:32:35 ntpdate[3558]: step time server 80.127.4.179 offset -358.420872 sec

Установка времени:

```
ntpdate -bs ntp.bmstu.ru
```

Через Crontab

```
crontab –e
0 * * * * /usr/sbin/ntpdate [серверы NTP]
```

Троекратное упоминание сервера europe.pool.ntp.org говорит об использовании трех разных серверов, включенных в пул серверов времени.

Автоматическая синхронизация

Разрешение на запуск сервера:

В /etc/rc.conf необходимо разрешить запуск NTP-сервера.

```
ntpd_enable="YES"
```

Файл конфигурации:

```
/etc/ntp.conf
server ntp.bmstu.ru
server time.nist.gov
server europe.pool.ntp.org
```

Разрешение доступа из локальной сети:

```
По умолчанию ваш сервер NTP будет доступен всем хостам в Интернет. Параметр
restrict в файле /etc/ntp.conf позволяет вам контролировать, какие машины могут
обращаться к вашему серверу.
Если вы хотите запретить всем машинам обращаться к вашему серверу NTP, добавьте
следующую строку в файл /etc/ntp.conf:
restrict default ignore
```

Если вы хотите разрешить синхронизировать свои часы с вашим сервером только машинам в вашей сети, но запретить им настраивать сервер или быть равноправными участниками синхронизации времени, то вместо указанной добавьте строчку

```
restrict 10.0.0.0 mask 255.0.0.0 nomodify notrap
/etc/ntp.conf может содержать несколько директив restrict
restrict 10.0.0.0 mask 255.0.0.0 noquery
```

Запуск сервера:

```
/etc/rc.d/ntpd restart
```

Сетевой суперсервер

Inetd является классическим вариантом программы.

После запуска сетевой суперсервер открывает на прослушивание порты, которые описаны в его конфигурационном файле /etc/inetd.conf. Если на порт приходит запрос на соединение, суперсервер запускает необходимую программу (согласно конфигурационного файла) и передает ей соединение.

Формат конфигурационного файла /etc/inetd.conf

- Сервис
- Тип соединения
- Протокол
- Флаги
- Пользователь
- Программа
- Аргументы

На одну запись отводится одна строка. В строке может быть шесть или семь полей, в зависимости от ситуации. Поля разделяются пробелами или символами табуляции.

- Сервис — имя сервиса. Можно писать только сервисы, определенные в файле /etc/services.
- Тип соединения — возможные варианты: stream, dgram и raw.
- Протокол — имя транспортного протокола, используемого при соединении.
- Флаги — если для каждого соединения необходимо запускать новую программу, значение флага nowait. Если одна запущенная программа может обработать несколько соединений — wait. В дополнении к параметру, можно написать число, определяющее максимальное количество подключений в минуту, например: nowait.30.
- Пользователь, с правами которого будет запущен данный процесс. Возможно указание группы. Например: user.group или user:group.
- Программа, которую запустит сетевой суперсервер. Если в этом поле стоит ключевое поле INTERNAL — inetd сам обработает такую службу и седьмое поле определять не надо.
- Параметры, которые будут переданы программе. Первый параметр — это обязательно имя самой программы.

Программа tcpd

Конфигурационный файл:

/etc/hosts.allow

В конфигурационном файле inetd.conf вместо имени программы у многих сервисов указывалась одна и таже программа — tcpd. Это, так называемый, tcp wrapper — программа, предназначенная для ограничения доступа к сервису.
У программы есть конфигурационный файл /etc/hosts.allow
В этом файле описано кому можно и кому нельзя, подключаться к сервисам.

Формат файлов прост:

```
программа : откуда : запрет/разрешение [: приложение]
```

В первом поле указывается имя программы, а не имя сервиса. Во втором — откуда можно или откуда нельзя подключаться. В этом поле можно писать:
•IP адрес машины
•IP адрес сети с указанием маски подсети
•Имя машины. В именах можно использовать символ *.

Третье поле содержит одно из ключевых слов allow или deny.

Четвертое поле не обязательно присутствует, но если оно есть, то содержит приложение которое следует запустить.

Лабораторная работа

Цель работы

Научиться настраивать сетевой суперсервер inetd.

Задачи

1. Убедитесь, что вы работаете с правами пользователя root.

2. Разрешите запуск inetd в /etc/rc.conf:
```
inetd_enable="YES"
```

3. Раскомментируйте следующую строку в /etc/inetd.conf:
```
telnet   stream  tcp   nowait  root  /usr/libexec/telnetd  telnetd
```

4. Запустите inetd
```
/etc/rc.d/inetd start
```

5. Убедитесь, что порты 23 открыт на прослушивание:
```
sockstat -4 | grep :23
```

6. Проверка работоспособности telnet-сервера
```
telnet cX
…..
User (root): admin
Password:
…..
$
```

7. Для ограничения доступа к сервису при помощи tcpd откройте на редактирование файл /etc/hosts.allow и вставьте в него первой строкой следующее правило:
```
telnetd : ALL : deny
```

8. Еще раз попробуйте подключиться к telnet-серверу
```
telnet cX
…..
Connection closed by foreign host.
```

Вопросы

1. При помощи какого параметра можно указать сетевому суперсерверу, что он должен на каждое новое соединение запускать новый экземпляр программы?

2. Каким образом можно ограничить доступ с определенной машины к программе?

3. Вы разрешили использование службы. Каким образом заставить сетевой суперсервер перечитать свой конфигурационный файл?

FTP - протокол передачи файлов

1.Введение

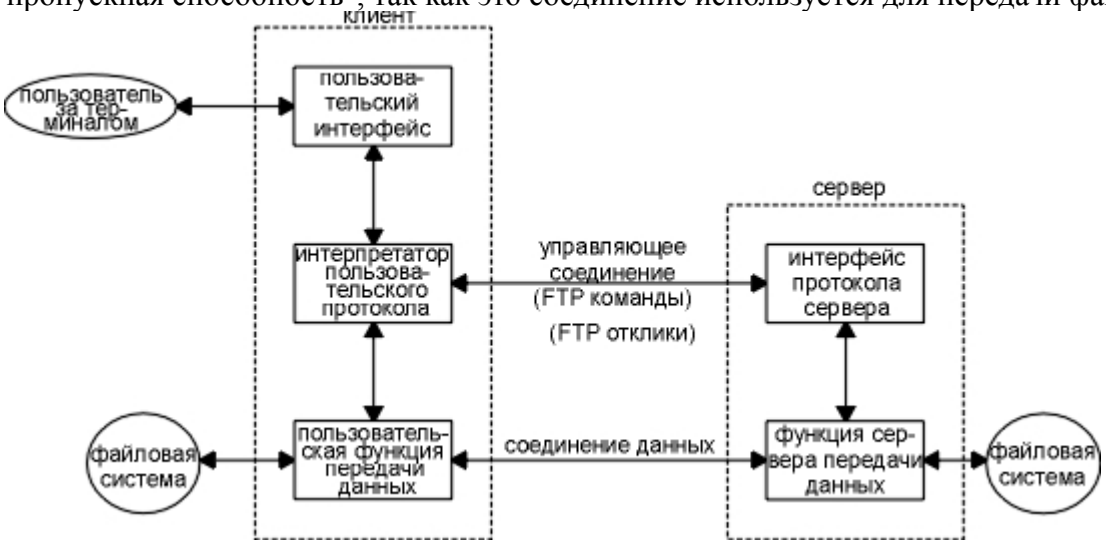
FTP является стандартом Internet для передачи файлов. Необходимо различать передачу файлов, именно то, что предоставляет FTP, и доступ к файлам, что предоставляется такими приложениями как NFS (Network File System, глава 29). Передача файлов заключается в копировании целого файла из одной системы в другую. Чтобы использовать FTP, необходимо иметь учетную запись (бюджет) на сервере, или можно воспользоваться так называемым анонимным FTP (anonymous FTP).

RFC 959 [Postel and Reynolds 1985] является официальной спецификацией FTP. Этот RFC описывает историю и развитиие передачи файлов в течение времени.

Протокол FTP

FTP отличается от других приложений тем, что он использует два TCP соединения для передачи файла.

1. Управляющее соединение устанавливается как обычное соединение клиент-сервер. Сервер осуществляет пассивное открытие на заранее известный порт FTP (21) и ожидает запроса на соединение от клиента. Клиент осуществляет активное открытие на TCP порт 21, чтобы установить управляющее соединение. Управляющее соединение существует все время, пока клиент общается с сервером. Это соединение используется для передачи команд от клиента к серверу и для передачи откликов от сервера.
2. Соединение данных открывается каждый раз, когда осуществляется передача файла между клиентом и сервером. (Оно также открывается и в другие моменты, как мы увидим позже.) Тип сервиса IP для соединения данных должен быть "максимальная пропускная способность", так как это соединение используется для передачи файлов.



Из рисунка видно, что интерактивный пользователь обычно не видит команды и отклики, которые передаются по управляющему соединению. Эти детали оставлены двум интерпретаторам протокола. Квадратик, помеченный как "пользовательский интерфейс", это именно то, что видит интерактивный пользователь (полноэкранный интерфейс, основанный на меню, командные строки и так далее). Интерфейс конвертирует ввод пользователя в FTP команды, которые отправляются по управляющему соединению. Отклики, возвращаемые сервером по управляющему соединению, конвертируются в формат, удобный для

пользователя.
FTP-сервер поддерживает 2 режима передачи данных: ascii и binary, что определяется переданными ему командами.

Команды FTP

Команды и отклики передаются по управляющему соединению между клиентом и сервером в формате NVT ASCII. В конце каждой строки команды или отклика присутствует пара CR, LF. Команды состоят из 3 или 4 байт, а именно из заглавных ASCII символов, некоторые с необязательными аргументами.

Команда*	**Описание**
help	получить список команд поддерживаемых ftp-сервером
ls или dir	список файлов или директорий
pwd	показать текущую директорию
cd	перейти к указанной директории
mkdir	создать директорию
rmdir	удалить директорию, если она не пустая
[m]get	получить файл[ы] с сервера
[m]put	отправить файл[ы] на сервер
TYPE {binary | ascii}	указать режим передачи данных
quit или exit	завершить работу с сервером

2. Настройка и запуск FTP-сервера

Все команды выполняются от имени суперпользователя root. Создаем нового пользователя ftpuser с паролем test:

```
adduser
Username: ftpuser
Full name: ftpuser
Uid (Leave empty for default):
Login group [ftpuser]:
Login group is ftpuser. Invite ftpuser into other groups? []:
Login class [default]:
Shell (bash sh csh tcsh bash rbash nologin) [sh]:
Home directory [/home/ftpuser]:
Home directory permissions (Leave empty for default):
Use password-based authentication? [yes]:
Use an empty password? (yes/no) [no]:
Use a random password? (yes/no) [no]:
Enter password:
Enter password again:
Lock out the account after creation? [no]:
Username   : ftpuser
Password   : *****
Full Name  : ftpuser
Uid        : 1002
Class      :
Groups     : ftpuser
Home       : /home/ftpuser
Home Mode  :
Shell      : /bin/sh
Locked     : no
OK? (yes/no): y
adduser: INFO: Successfully added (ftpuser) to the user database.
Add another user? (yes/no): n
```

Необходимо разрешить запуск ftp-сервера в /etc/rc.conf

```
ftpd_enable="YES"
```

* В данной таблице указаны только те команды которые поддерживаются практически всеми ftp-серверами

Из соображений безопасности вам необходимо сменить оболочку для ftpuser

```
which passwd >> /etc/shells
pw usermod ftpuser -s /usr/bin/passwd
```

Ограничение доступа ftp пользователей домашним каталогом

```
echo "@ftpuser" > /etc/ftpchroot
```

Разрешение анонимного доступа для пользователя ftp

```
pw useradd ftp -d /usr/ports/ -s /sbin/nologin
```

И запустить сервер

```
/etc/rc.d/ftpd start
```

Проверка

```
ftp ftpuser@localhost
```