

# **Отчет по лабораторной работе 12**

Генералов Даниил, НПИбд-01-21, 1032202280

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Задание</b>	<b>6</b>
<b>3</b>	<b>Выполнение лабораторной работы</b>	<b>7</b>
<b>4</b>	<b>Выводы</b>	<b>10</b>
<b>5</b>	<b>Контрольные вопросы</b>	<b>11</b>

# Список иллюстраций

3.1	date . . . . .	7
3.2	chronyd . . . . .	8
3.3	vagrant . . . . .	9

## **Список таблиц**

# 1 Цель работы

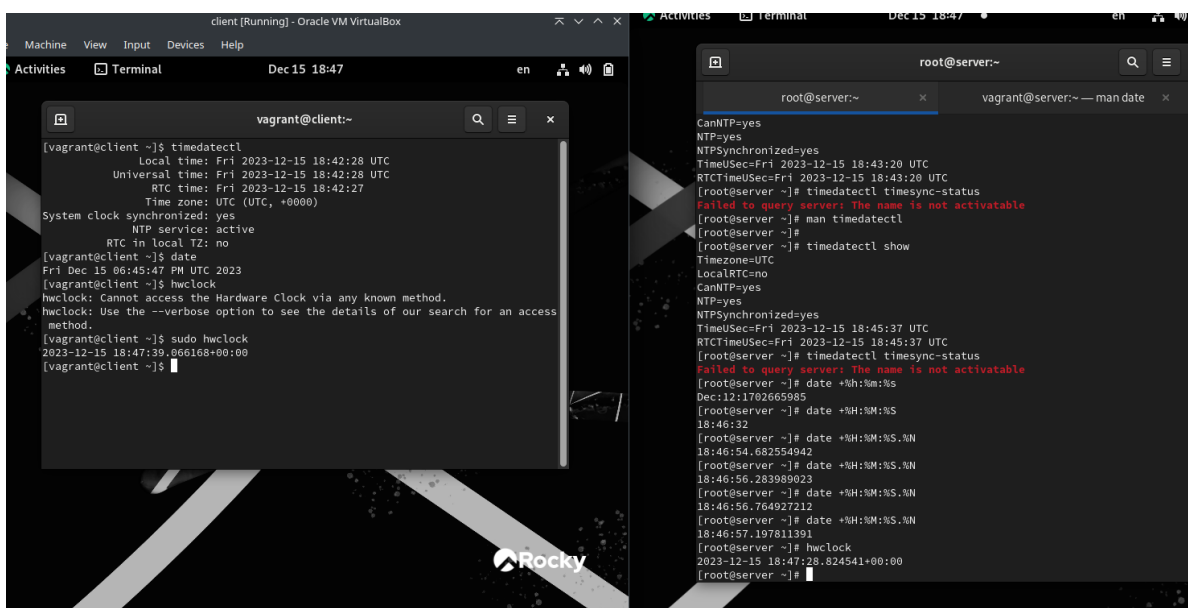
Получение навыков по управлению системным временем и настройке синхронизации времени.

## 2 Задание

1. Изучите команды по настройке параметров времени (см. раздел 12.4.1).
2. Настройте сервер в качестве сервера синхронизации времени для локальной сети (см. раздел 12.4.2).
3. Напишите скрипты для Vagrant, фиксирующие действия по установке и настройке NTP-сервера и клиента (см. раздел 12.4.3).

### 3 Выполнение лабораторной работы

Сначала мы посмотрели на информацию о службе синхронизации времени systemd через `timedatectl`, и узнали, что они находятся в UTC+0-зоне времени, и синхронизируют время с сетью. После этого мы узнали текущую дату и время с точностью до наносекунд, и узнали, что на клиенте RTC-время отстает от системного времени на одну секунду.



The image contains two terminal screenshots. The left screenshot shows a client terminal with the following commands and output:

```
vagrant@client:~$ timedatectl
Local time: Fri 2023-12-15 18:42:28 UTC
Universal time: Fri 2023-12-15 18:42:28 UTC
RTC time: Fri 2023-12-15 18:42:27
Time zone: UTC (UTC, +0000)
System clock synchronized: yes
NTP service: active
RTC in local TZ: no

[vagrant@client ~]$ date
Fri Dec 15 06:45:47 PM UTC 2023

[vagrant@client ~]$ hwclock
hwclock: Cannot access the Hardware Clock via any known method.
hwclock: Use the --verbose option to see the details of our search for an access method.

[vagrant@client ~]$ sudo hwclock
2023-12-15 18:47:39.066168+00:00
[vagrant@client ~]$
```

The right screenshot shows a server terminal with the following commands and output:

```
root@server:~$ timedatectl timesync-status
Failed to query server: The name is not activatable

[root@server ~]# man timedatectl
[root@server ~]# timedatectl show
Timezone=UTC
LocalRTC=no
CanNTP=yes
NTP=yes
NTPSynchronized=yes
TimeUseSec=Fri 2023-12-15 18:43:20 UTC
RTCTimeUseSec=Fri 2023-12-15 18:43:20 UTC

[root@server ~]# date +%H:%M:%S
Dec:12:170265985
[root@server ~]# date +%H:%M:%S
18:46:32
[root@server ~]# date +%H:%M:%S.%N
18:46:54.682554942
[root@server ~]# date +%H:%M:%S.%N
18:46:56.283989023
[root@server ~]# date +%H:%M:%S.%N
18:46:56.764927212
[root@server ~]# date +%H:%M:%S.%N
18:46:57.197011391
[root@server ~]# hwclock
2023-12-15 18:47:28.824541+00:00
[root@server ~]#
```

Рис. 3.1: date

После этого мы настроили на сервере `chronyd`, чтобы он принимал соединения из локальной сети. Он обрабатывает время из четырех источников – двух из них имеют стратум 3, а два из них – 2. Мы добавляем NTP как службу в `firewalld`, а затем настраиваем, чтобы клиент получал синхронизацию только с сервера. Поскольку

сервер вносит неточность в время, он обозначает себя стратумом ниже, чем свои источники – он становится стратумом 5. Сначала единственный источник не считается хорошим для синхронизации (что показывает вопросительный знак в его строчке), но через некоторое время его стабильность измеряется и он выбирается самым хорошим (среди единственного выбора) и выбранным для синхронизации. Как результат, клиент синхронизировался с сервером до четырех нулей, и определил, что часы сервера спешат на 13.438 миллионных долей.

```
client [Running] - Oracle VM VirtualBox
vagrant@client:~$ nano /etc/chrony.conf
# Please consider joining the pool (https://www.pool.ntp.org/join.html).
# Use NTP servers from the pool.
server pool.ntp.org iburst
# Record the rate at which the system clock gains/loses time.

vagrant@client:~$ sudo systemctl restart chronyd
(vagrant@client)~$ sudo chronyc sources
MS Name/IP address         Stratum Poll Reach LastRx Last sample
=====
*? www.dggeneralov.net      5 6 1 1 +39ms[ +39ms] +/- 64ms
(vagrant@client)~$ sudo chronyc sources
MS Name/IP address         Stratum Poll Reach LastRx Last sample
=====
*? www.dggeneralov.net      5 6 377 32 -162us[-523us] +/- 65ms
(vagrant@client)~$

Reference ID      : C0A80101 (dhcp.dggeneralov.net)
Stratum          : 6
Ref time (UTC)   : Fri Dec 15 19:10:18 2022
System time      : 0.000042171 seconds slow of NTP time
Last offset      : -0.0000061377 seconds
RMS offset       : 0.018401438 seconds
Frequency        : 13.438 ppm fast
Residual freq    : -0.033 ppm
Skew             : 2.187 ppm
Root delay       : 0.324557387 seconds
Root dispersion   : 0.001839423 seconds
Update interval   : 64.4 seconds
Leap status      : Normal
[vagrant@client]~$

server [Running] - Oracle VM VirtualBox
root@server:~$ dnf install -y chrony
Last metadata expiration check: 0:48:45 ago on Fri 15 Dec 2023 06:01:41 PM UTC.
Package chrony-4.2-1.el9.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
root@server:~$ chronyc sources
MS Name/IP address         Stratum Poll Reach LastRx Last sample
=====
*? time.cloudflare.com      2 8 377 183 -4880us[-3842us] +/- 113ms
*? time.cloudflare.com      3 7 377 55 +351us[ +613us] +/- 24ms
*? ntp.ubuntu.com           3 8 377 243 -5942us[-3714us] +/- 143ms
*? ntp.ubuntu.com           2 8 377 56 -2589us[-2326us] +/- 113ms
root@server:~$ sudo systemctl restart chronyd
root@server:~$ firewall-cmd --add-service=ntp
success
root@server:~$ firewall-cmd --runtime-to-permanent
success
root@server:~$

This section describes each of the commands available in the chrony program.

System clock tracking
The tracking command displays parameters about the system clock's performance. An example of the output is shown below.

Reference ID      : C0A8010F (foo.example.com)
Stratum          : 3
Ref time (UTC)   : Fri Jan 27 09:49:17 2023
System time      : 0.000000522 seconds slow of NTP time
Last offset      : -0.000000747 seconds
RMS offset       : 0.000035522 seconds
[vagrant@server]~$
```

Рис. 3.2: chronyd

Наконец, мы экспортируем настройки в Vagrantfile.



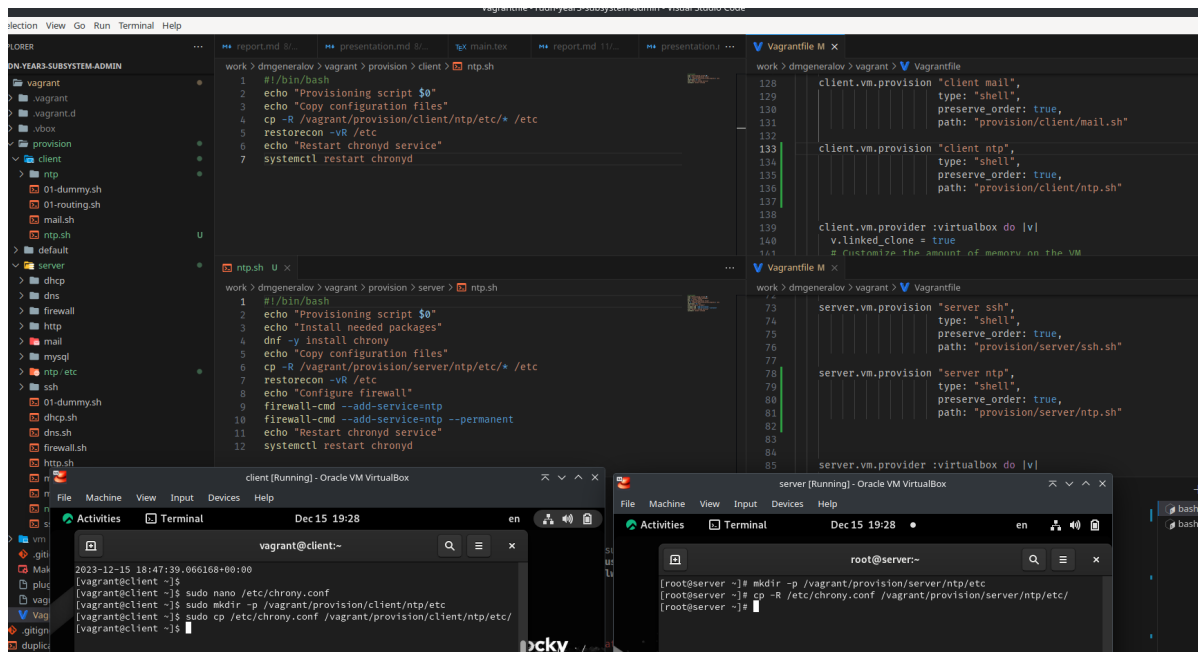


Рис. 3.3: vagrant

## 4 Выводы

Я получил опыт настройки локального NTP-сервера.

## 5 Контрольные вопросы

### 1. Почему важна точная синхронизация времени для служб баз данных?

Потому, что часто нужно иметь возможность сериализовать события, то есть указать, в каком порядке они произошли. Если события происходят на разных машинах, и между ними нет точной синхронизации времени, то они могут записать события, которые произошли в такие времена, чтобы логически противоречить друг другу.

### 2. Почему служба проверки подлинности Kerberos сильно зависит от правильной синхронизации времени?

Потому что Kerberos выдает подписанные тикеты, которые работают только очень маленькое время, и если между машинами есть слишком большое расхождение, то тикет, который получил клиент только что, сервер может считать уже истекшим.

### 3. Какая служба используется по умолчанию для синхронизации времени на RHEL 7?

`systemd-timesync` и `chronyd`

### 4. Какова страта по умолчанию для локальных часов?

По умолчанию это не определено, но параметр `local stratum` говорит серверу, каким стратумом считать его локальные часы.

5. Какой порт брандмауэра должен быть открыт, если вы настраиваете свой сервер как одноранговый узел NTP?

123/udp

6. Какую строку вам нужно включить в конфигурационный файл `chrony`, если вы хотите быть сервером времени, даже если внешние серверы NTP недоступны?

Нужно назначить ему стратум, который относится к локальным часам, и он должен быть сравнительно низким: `local stratum 10`

7. Какую страту имеет хост, если нет текущей синхронизации времени NTP?

Он имеет стратум 16, потому что по протоколу NTP может существовать только стратум 1 до 15.

8. Какую команду вы бы использовали на сервере с `chrony`, чтобы узнать, с какими серверами он синхронизируется?

`chronyc sources`

9. Как вы можете получить подробную статистику текущих настроек времени для процесса `chrony` вашего сервера?

`chronyc tracking`