维护准确的时间

培训目标

学完本节后，您应能够利用 NTP 维护准确的时间同步，并且配置时区以确保系统日志和日志记录的事件标有正确的时间戳。

设置本地时钟和时区

对于在多个系统间分析日志文件而言，正确同步系统时间至关重要。网络时间协议 (NTP) 是计算机用于通过互联网提供并获取正确时间信息的一种标准方法。计算机可以通过互联网上的公共 NTP 服务获取正确的时间信息，如 NTP Pool Project。另一种选择是通过高质量硬件时钟为本地客户端提供准确时间。

timedatectl 命令简要显示当前的时间相关系统设置，如系统的当前时间、时区和 NTP 同步设置。

```
[user@host ~]$ timedatectl
               Local time: Fri 2019-04-05 16:10:29 CDT
           Universal time: Fri 2019-04-05 21:10:29 UTC
                 RTC time: Fri 2019-04-05 21:10:29
                Time zone: America/Chicago (CDT, -0500)
System clock synchronized: yes
              NTP service: active
          RTC in local TZ: no
```

系统提供了包含时区的数据库，可通过 **timedatectl list-timezones** 命令列出。

```
[user@host ~]$ timedatectl list-timezones
Africa/Abidjan
Africa/Accra
Africa/Addis_Ababa
Africa/Algiers
Africa/Asmara
Africa/Bamako
...
```

时区名称基于由 NANA 维护的公共时区数据库。时区的命名基于大陆或海洋名称，通常但并不总是使用该时区地区中的最大城市名称。例如，大多数美国山地时区为“美洲/丹佛”。

当时区中的各个地区拥有不同的夏时制规则时，选择正确的名称可能会不那么直观。例如，美国亚利桑那州（美国山地时间）的大部分地区不进行夏时制调整，处于“美洲/凤凰城”时区内。

可以使用 **tzselect** 命令识别正确的 zoneinfo 时区名称。它以交互方式向用户提示关于系统位置的问题，然后输出正确时区的名称。它不会对系统的时区设置进行任何更改。

超级用户可以利用 **timedatectl set-timezone** 命令更改系统设置来更新当前的时区。以下 **timedatectl** 命令将当前时区更新为 **America/Phoenix**。

海量视频题库 myitpub.com QQ:556462
www.52myit.com

```
[root@host ~]# timedatectl set-timezone America/Phoenix
[root@host ~]# timedatectl
               Local time: Fri 2019-04-05 14:12:39 MST
           Universal time: Fri 2019-04-05 21:12:39 UTC
                 RTC time: Fri 2019-04-05 21:12:39
                Time zone: America/Phoenix (MST, -0700)
System clock synchronized: yes
              NTP service: active
          RTC in local TZ: no
```

注意

如果您需要在特定服务器上使用协调世界时 (UTC)，可将其时区设置为 UTC。**tzselect** 命令不包括 UTC 时区的名称。使用 **timedatectl set-timezone UTC** 命令可将系统的当前时区设置为 **UTC**。

使用 **timedatectl set-time** 命令可更改系统的当前时间。时间以 "YYYY-MM-DD hh:mm:ss" 格式指定，其中可以省略日期或时间。以下 **timedatectl** 命令将时间更改为 **09:00:00**。

```
[root@host ~]# timedatectl set-time 9:00:00
[root@host ~]# timedatectl
               Local time: Fri 2019-04-05 09:00:27 MST
           Universal time: Fri 2019-04-05 16:00:27 UTC
                 RTC time: Fri 2019-04-05 16:00:27
                Time zone: America/Phoenix (MST, -0700)
System clock synchronized: yes
              NTP service: active
          RTC in local TZ: no
```

timedatectl set-ntp 命令可启用或禁用 NTP 同步（自动调整时间）。该选项需要 **true** 或 **false** 参数将它打开或关闭。以下 **timedatectl** 命令打开 NTP 同步。

```
[root@host ~]# timedatectl set-ntp true
```

注意

在红帽企业 Linux 8 中，**timedatectl set-ntp** 命令将对是否运行 chronyd NTP 服务做出调整。其他 Linux 发行版可使用此设置来调整不同的 NTP 或 SNTP 服务。

通过使用红帽企业 Linux 中的其他实用程序（如图形 GNOME Settings 应用程序）来启用或禁用 NTP，也可以更新此设置。

配置和监控 CHRONYD

chronyd 服务通过与配置的 NTP 服务器进行同步，使通常不准确的本地硬件时钟 (RTC) 保持正确运行。如果没有可用的网络连接，chronyd 将计算 RTC 时钟漂移，记录在 **/etc/chrony.conf** 配置文件指定的 **driftfile** 变量中。

默认情况下，chronyd 服务使用 NTP Pool Project 的服务器同步时间，不需要额外的配置。当涉及的计算机位于孤立网络中时，可能需要更改 NTP 服务器。

海量视频题库 myitpub.com QQ:556462 www.52myit.com

NTP 时间源的 stratum 决定其质量。stratum 确定计算机与高性能参考时钟偏离的跃点数。参考时钟是 **stratum 0** 时间源。与之直接关联的 NTP 服务器是 **stratum 1**，而与该 NTP 服务器同步时间的计算机则是 **stratum 2** 时间源。

server 和 peer 是您可以在 **/etc/chrony.conf** 配置文件中配置的两种时间源类别。server 比本地 NTP 服务器高一个级别，而 peer 则属于同一级别。可以指定多个 server 和多个 peer，每行指定一个。

server 行的第一个参数是 NTP 服务器的 IP 地址或 DNS 名称。在服务器 IP 地址或名称后，可以列出该服务器的一系列选项。建议使用 **iburst** 选项，因为在服务启动后，会在很短时间内执行四种测量，获得更加精确的初始时钟同步。

/etc/chrony.conf 文件中的以下 **server classroom.example.com iburst** 行会使 chronyd 服务使用 classroom.example.com NTP 时间源。

```
# Use public servers from the pool.ntp.org project.
...output omitted...
server classroom.example.com iburst
...output omitted...
```

将 **chronyd** 指向本地时间源 classroom.example.com 后，您应当重新启动该服务。

```
[root@host ~]# systemctl restart chronyd
```

chronyc 命令充当 chronyd 服务的客户端。设置 NTP 同步后，您应当使用 **chrony sources** 命令验证本地系统是否使用 NTP 服务器无缝同步系统时钟。如需获取包含额外说明的更详细输出，可使用 **chronyc sources -v** 命令。

```
[root@host ~]# chronyc sources -v
210 Number of sources = 1

  .-- Source mode  '^' = server, '=' = peer, '#' = local clock.
 / .- Source state '*' = current synced, '+' = combined , '-' = not combined,
| /   '?' = unreachable, 'x' = time may be in error, '~' = time too variable.
||                                                 .- xxxx [ yyyy ] +/- zzzz
||                                                /   xxxx = adjusted offset,
||         Log2(Polling interval) -.             |    yyyy = measured offset,
||                                  \            |    zzzz = estimated error.
||                                   |           |
MS Name/IP address         Stratum Poll Reach LastRx Last sample
===============================================================================
^* classroom.example.com         8   6    17    23   -497ns[-7000ns] +/-  956us
```

S（源状态）字段中的 ***** 字符表示 classroom.example.com 服务器已被用作时间源，是计算机当前与之同步的 NTP 服务器。

海量视频题库 myitpub.com QQ:556462
www.52myit.com

参考文献

`timedatectl`(1)、`tzselect`(8)、`chronyd`(8)、**`chrony.conf`**(5) 和 `chronyc`(1) man page

NTP Pool Project
http://www.pool.ntp.org/

时区数据库
http://www.iana.org/time-zones

海量视频题库 myitpub.com QQ:5565462
www.52myit.com

指导练习

维护准确的时间

在本练习中，您将使用调整服务器上的时区，并确保其系统时钟与 NTP 时间源同步。

成果

您应能够：

· 更改服务器上的时区。

· 配置服务器，以使其时间与 NTP 时间源同步。

在你开始之前

以 `student` 用户身份并使用 `student` 作为密码登录 `workstation`。

在 `workstation` 上，运行 **`lab log-maintain start`** 脚本来开始本练习。此脚本可确保 `servera` 系统上禁用时间同步，让您有机会手动更新该系统上的系统设置并启用时间同步。

```
[student@workstation ~]$ lab log-maintain start
```

1. 从 `workstation`，以 `student` 用户身份打开连接 `servera` 的 SSH 会话。

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

2. 在本活动中，假装 `servera` 系统被重新安置到海地，因此您需要相应地更新时区。在运行 **`timedatectl`** 命令来更新时区过程中使用 **`sudo`** 提升 `student` 用户的特权。如果系统要求，使用 `student` 作为密码。

2.1. 使用 **`tzselect`** 命令确定海地的适当时区。

```
[student@servera ~]$ tzselect
Please identify a location so that time zone rules can be set correctly.
Please select a continent, ocean, "coord", or "TZ".
 1) Africa
 2) Americas
 3) Antarctica
 4) Asia
 5) Atlantic Ocean
 6) Australia
 7) Europe
 8) Indian Ocean
 9) Pacific Ocean
10) coord - I want to use geographical coordinates.
11) TZ - I want to specify the time zone using the Posix TZ format.
```

海量视频题库 myitpub.com QQ:5565462
www.52myit.com

```
#? 2
Please select a country whose clocks agree with yours.
 1) Anguilla              19) Dominican Republic    37) Peru
 2) Antigua & Barbuda     20) Ecuador               38) Puerto Rico
 3) Argentina             21) El Salvador           39) St Barthelemy
 4) Aruba                 22) French Guiana         40) St Kitts & Nevis
 5) Bahamas               23) Greenland             41) St Lucia
 6) Barbados              24) Grenada               42) St Maarten (Dutch)
 7) Belize                25) Guadeloupe            43) St Martin (French)
 8) Bolivia               26) Guatemala             44) St Pierre & Miquelon
 9) Brazil                27) Guyana                45) St Vincent
10) Canada                28) Haiti                 46) Suriname
11) Caribbean NL          29) Honduras              47) Trinidad & Tobago
12) Cayman Islands        30) Jamaica               48) Turks & Caicos Is
13) Chile                 31) Martinique            49) United States
14) Colombia              32) Mexico                50) Uruguay
15) Costa Rica            33) Montserrat            51) Venezuela
16) Cuba                  34) Nicaragua             52) Virgin Islands (UK)
17) Curaçao               35) Panama                53) Virgin Islands (US)
18) Dominica              36) Paraguay
#? 28
The following information has been given:

 Haiti

Therefore TZ='America/Port-au-Prince' will be used.
Selected time is now: Tue Feb 19 00:51:05 EST 2019.
Universal Time is now: Tue Feb 19 05:51:05 UTC 2019.
Is the above information OK?
1) Yes
2) No
#? 1

You can make this change permanent for yourself by appending the line
 TZ='America/Port-au-Prince'; export TZ
to the file '.profile' in your home directory; then log out and log in again.

Here is that TZ value again, this time on standard output so that you
can use the /usr/bin/tzselect command in shell scripts:
America/Port-au-Prince
```

注意前面的 **tzselect** 命令显示了海地的适当时区。

2.2. 使用 **timedatectl** 命令，将 servera 上的时区更新为 **America/Port-au-Prince**。

```
[student@servera ~]$ sudo timedatectl set-timezone America/Port-au-Prince
[sudo] password for student: student
```

2.3. 使用 **timedatectl** 命令，验证时区是否已更新为 **America/Port-au-Prince**。

海量视频题库 myitpub.com QQ:5565462
www.52myit.com

```
[student@servera ~]$ timedatectl
           Local time: Tue 2019-02-19 01:16:29 EST
           Universal time: Tue 2019-02-19 06:16:29 UTC
                 RTC time: Tue 2019-02-19 06:16:29
                 Time zone: America/Port-au-Prince (EST, -0500)
System clock synchronized: no
              NTP service: inactive
          RTC in local TZ: no
```

3. 配置 servera 上的 chronyd 服务，以将系统时间与 NTP 时间源 classroom.example.com 同步。

3.1. 编辑 **/etc/chrony.conf** 文件，将 classroom.example.com 服务器指定为 NTP 时间源。您可以使用 **sudo vim /etc/chrony.conf** 命令来编辑配置文件。以下输出显示了您必须添加到配置文件中的配置行：

```
...output omitted...
server classroom.example.com iburst
...output omitted...
```

/etc/chrony.conf 配置文件的上面这一行包括可加快初始时间同步的 **iburst** 选项。

3.2. 使用 **timedatectl** 命令，打开 servera 上的时间同步。

```
[student@servera ~]$ sudo timedatectl set-ntp yes
```

前面的 **timedatectl** 命令使用 **/etc/chrony.conf** 配置文件中已更改的设置激活 NTP 服务器。前面的 **timedatectl** 命令可能会激活 chronyd 或 ntpd 服务，具体取决于系统上当前安装的服务。

4. 验证 servera 上的时间设置目前是否已配置为与课堂环境中的 classroom.example.com 时间源同步。

4.1. 使用 **timedatectl** 命令，验证 servera 当前是否启用了时间同步。

```
[student@servera ~]$ timedatectl
               Local time: Tue 2019-02-19 01:52:17 EST
           Universal time: Tue 2019-02-19 06:52:17 UTC
                 RTC time: Tue 2019-02-19 06:52:17
                Time zone: America/Port-au-Prince (EST, -0500)
System clock synchronized: yes
              NTP service: active
          RTC in local TZ: no
```

注意

如果前面的输出显示时钟未同步，请等待两秒钟，然后再次运行 **timedatectl** 命令。成功将时间设置与时间源同步需要几秒钟。

海量视频题库 myitpub.com QQ:556462
www.52myit.com

4.2. 使用 **chronyc** 命令，验证 servera 系统当前是否将自身的时间设置与 classroom.example.com 时间源同步。

```
[student@servera ~]$ chronyc sources -v
210 Number of sources = 1

  .-- Source mode  '^' = server, '=' = peer, '#' = local clock.
 / .- Source state '*' = current synced, '+' = combined , '-' = not combined,
| /   '?' = unreachable, 'x' = time may be in error, '~' = time too variable.
||                                                 .- xxxx [ yyyy ] +/- zzzz
||      Reachability register (octal) -.           |  xxxx = adjusted offset,
||      Log2(Polling interval) --.      |          |  yyyy = measured offset,
||                                \     |          |  zzzz = estimated error.
||                                 |    |           \
MS Name/IP address         Stratum Poll Reach LastRx Last sample
===============================================================================
^* classroom.example.com         2   6   377    62   +105us[ +143us] +/-   14ms
```

注意在前面的输出中，classroom.example.com NTP 时间源的源状态 (**S**) 字段中显示了一个星号 (*****)。星号表示本地系统时间当前与 NTP 时间源成功同步。

4.3. 从 servera 注销。

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$
```

完成

在 workstation 上，运行 **lab log-maintain finish** 来完成本练习。此脚本可确保 servera 上恢复原始的时区，以及所有原始的时间设置。

```
[student@workstation ~]$ lab log-maintain finish
```

本引导式练习到此结束。

海量视频题库 myitpub.com QQ:556546 2
www.52myit.com

开放研究实验

分析和存储日志

任务执行清单

在本实验中，您将更改现有服务器上的时区，并且配置一个新的日志文件用于与身份验证失败相关的所有事件。

成果

您应能够：

- 更新现有服务器上的时区。
- 配置一个新日志文件，以存储与身份验证失败相关的所有消息。

在你开始之前

以 `student` 用户身份并使用 `student` 作为密码登录 `workstation`。

在 `workstation` 上，运行 **`lab log-review start`** 脚本来开始本练习。此脚本将记录 `serverb` 系统的当前时区，并确保环境的设置正确无误。

```
[student@workstation ~]$ lab log-review start
```

1. 从 `workstation`，以 `student` 用户身份打开连接 `serverb` 的 SSH 会话。
2. 假定 `servera` 系统被重新安置到牙买加，因此您必须相应地更新时区。使用 **`sudo`** 可提升 `student` 用户的特权，以便运行 **`timedatectl`** 命令来更新时区。如果系统要求，使用 `student` 作为密码。
3. 显示 `serverb` 上在过去 30 分钟内记录的日志事件。
4. 创建 **`/etc/rsyslog.d/auth-errors.conf`** 文件并进行配置，从而让 `rsyslog` 服务将身份验证和安全问题相关的消息写入新的 **`/var/log/auth-errors`** 文件。在配置文件中使用 **`authpriv`** 设备和 **`alert`** 优先级。

评估

在 `workstation` 上，运行 **`lab log-review grade`** 命令以确认本练习是否成功。

```
[student@workstation ~]$ lab log-review grade
```

完成

在 `workstation` 上，运行 **`lab log-review finish`** 脚本来完成本实验。此脚本可确保 `serverb` 上恢复原始的时区，以及所有原始的时间设置。

```
[student@workstation ~]$ lab log-review finish
```

海量视频题库 myitpub.com QQ:556462 www.52myit.com

本引导式练习到此结束。

海量视频题库 myitpub.com QQ:5565462
www.52myit.com

解决方案

分析和存储日志

任务执行清单

在本实验中，您将更改现有服务器上的时区，并且配置一个新的日志文件用于与身份验证失败相关的所有事件。

成果

您应能够：

· 更新现有服务器上的时区。

· 配置一个新日志文件，以存储与身份验证失败相关的所有消息。

在你开始之前

以 `student` 用户身份并使用 `student` 作为密码登录 `workstation`。

在 `workstation` 上，运行 **`lab log-review start`** 脚本来开始本练习。此脚本将记录 `serverb` 系统的当前时区，并确保环境的设置正确无误。

```
[student@workstation ~]$ lab log-review start
```

1. 从 `workstation`，以 `student` 用户身份打开连接 `serverb` 的 SSH 会话。

```
[student@workstation ~]$ ssh student@serverb
...output omitted...
[student@serverb ~]$
```

2. 假定 `servera` 系统被重新安置到牙买加，因此您必须相应地更新时区。使用 **`sudo`** 可提升 `student` 用户的特权，以便运行 **`timedatectl`** 命令来更新时区。如果系统要求，使用 `student` 作为密码。

   2.1. 使用 **`timedatectl`** 命令查看可用时区并确定牙买加的适当时区。

```
[student@serverb ~]$ timedatectl list-timezones | grep Jamaica
America/Jamaica
```

   2.2. 使用 **`timedatectl`** 命令，将 `serverb` 系统的时区设为 **`America/Jamaica`**。

```
[student@serverb ~]$ sudo timedatectl set-timezone America/Jamaica
[sudo] password for student: student
```

   2.3. 使用 **`timedatectl`** 命令，验证时区是否已成功设置为 **`America/Jamaica`**。

海量视频题库 myitpub.com QQ:556462 www.52myit.com

```
[student@serverb ~]$ timedatectl
               Local time: Tue 2019-02-19 11:12:46 EST
           Universal time: Tue 2019-02-19 16:12:46 UTC
                 RTC time: Tue 2019-02-19 16:12:45
                 Time zone: America/Jamaica (EST, -0500)
System clock synchronized: yes
              NTP service: active
          RTC in local TZ: no
```

3. 显示 `serverb` 上在过去 30 分钟内记录的日志事件。

3.1. 使用 **`date`** 命令，确定要查看日志条目的时间范围。

```
[student@serverb ~]$ date
Fri Feb 22 07:31:05 EST 2019
[student@serverb ~]$ date -d "-30 minutes"
Fri Feb 22 07:01:31 EST 2019
```

3.2. 使用 **`journalctl`** 命令 **`--since`** 和 **`--until`** 选项显示 `serverb` 上在过去 30 分钟内记录的日志事件。要退出 **`journalctl`**，请按 **`q`** 键。

```
[student@serverb ~]$ journalctl --since 07:01:00 --until 07:31:00
...output omitted...
Feb 22 07:24:28 serverb.lab.example.com systemd[1138]: Reached target Timers.
Feb 22 07:24:28 serverb.lab.example.com systemd[1138]: Reached target Paths.
Feb 22 07:24:28 serverb.lab.example.com systemd[1138]: Starting D-Bus User Message
 Bus Socket.
Feb 22 07:24:28 serverb.lab.example.com systemd[1138]: Listening on D-Bus User
 Message Bus Socket.
Feb 22 07:24:28 serverb.lab.example.com systemd[1138]: Reached target Sockets.
Feb 22 07:24:28 serverb.lab.example.com systemd[1138]: Reached target Basic
 System.
Feb 22 07:24:28 serverb.lab.example.com systemd[1138]: Reached target Default.
Feb 22 07:24:28 serverb.lab.example.com systemd[1138]: Startup finished in 123ms.
Feb 22 07:24:28 serverb.lab.example.com systemd[1]: Started User Manager for UID
 1000.
Feb 22 07:24:28 serverb.lab.example.com sshd[1134]: pam_unix(sshd:session):
 session opened for user student by (uid=0)
Feb 22 07:26:56 serverb.lab.example.com systemd[1138]: Starting Mark boot as
 successful...
Feb 22 07:26:56 serverb.lab.example.com systemd[1138]: Started Mark boot as
 successful.
lines 1-36/36 (END) q
[student@serverb ~]$
```

4. 创建 **`/etc/rsyslog.d/auth-errors.conf`** 文件并进行配置，从而让 `rsyslog` 服务将身份验证和安全问题相关的消息写入新的 **`/var/log/auth-errors`** 文件。在配置文件中使用 **`authpriv`** 设备和 **`alert`** 优先级。

4.1. 创建 **`/etc/rsyslog.d/auth-errors.conf`** 文件，以将新的 **`/var/log/auth-errors`** 文件指定为存储身份验证和安全问题相关的消息的目标位置。您可以使用 **`sudo vim /etc/rsyslog.d/auth-errors.conf`** 命令来创建配置文件。

海量视频题库 mvitpub.com QQ:556462
www.52mvit.com

```
authpriv.alert  /var/log/auth-errors
```

4.2. 重新启动 rsyslog 服务，使配置文件中的更改生效。

```
[student@serverb ~]$ sudo systemctl restart rsyslog
```

4.3. 使用 **logger** 命令将新的日志消息写入 **/var/log/auth-errors** 文件。应用 **-p authpriv.alert** 选项，以生成与身份验证和安全问题相关的日志消息。

```
[student@serverb ~]$ logger -p authpriv.alert "Logging test authpriv.alert"
```

4.4. 使用 **tail** 命令，确认 **/var/log/auth-errors** 文件中包含了带有 **Logging test authpriv.alert** 消息的日志条目。

```
[student@serverb ~]$ sudo tail /var/log/auth-errors
Feb 19 11:56:07 serverb student[6038]: Logging test authpriv.alert
```

4.5. 从 serverb 注销。

```
[student@serverb ~]$ exit
logout
Connection to serverb closed.
[student@workstation ~]$
```

评估

在 workstation 上，运行 **lab log-review grade** 命令以确认本练习是否成功。

```
[student@workstation ~]$ lab log-review grade
```

完成

在 workstation 上，运行 **lab log-review finish** 脚本来完成本实验。此脚本可确保 serverb 上恢复原始的时区，以及所有原始的时间设置。

```
[student@workstation ~]$ lab log-review finish
```

本引导式练习到此结束。

海量视频题库 myitpub.com QQ:556462
www.52myit.com

总结

在本章中，您学到了：

- `systemd-journald` 和 `rsyslog` 服务可捕获日志消息并写入到适当的文件中。
- **`/var/log`** 目录包含日志文件。
- 定期轮换日志文件可防止它们填满文件系统空间。
- `systemd` 日志是临时的，不会在系统重启后持久保留。
- `chronyd` 服务有助于将时间设置与时间源同步。
- 可以根据服务器的位置更新其时区。

海量视频题库 myitpub.com QQ:5565462
www.52myit.com

海量视频题库 myitpub.com QQ:5565462
www.52myit.com

章 12

管理网络

目标

配置红帽企业 Linux 服务器上的网络接口和设置。

培训目标

- 描述服务器的网络寻址和路由的基本概念。
- 使用命令行实用程序，测试并检查当前的网络配置。
- 使用 **nmcli** 管理网络设置和设备。
- 通过编辑配置文件修改网络设置。
- 配置服务器的静态主机名及其名称解析，并测试结果。

章节

- 描述网络概念（及测验）
- 验证网络配置（及引导式练习）
- 从命令行配置网络（及引导式练习）
- 编辑网络配置文件（及引导式练习）
- 配置主机名和名称解析（及引导式练习）

实验

管理网络

海量视频题库 myitpub.com QQ:556462
www.52myit.com

描述网络概念

培训目标

学完本节后，您应能够描述服务器的网络寻址和路由的基本概念。

TCP/IP 网络模型

TCP/IP 网络模型是一种简化的四层抽象集合，用于描述不同的协议如何进行互操作，以便计算机通过互联网将流量从一台计算机发送到另一台计算机。它由 RFC 1122 互联网主机要求 -- 通信层规定。这四层是：

- **应用**

  每一应用程序具有用于通信的规范，以便客户端和服务器可以跨平台通信。常用的协议有 SSH（远程登录）、HTTPS（安全 Web）、NFS 或 CIFS（文件共享），以及 SMTP（电子邮件递送）等。

- **传输**

  传输协议有 TCP 和 UDP。TCP 是可靠连接导向型通信，而 UDP 属于无连接数据报协议。应用协议使用 TCP 或 UDP 端口。`/etc/services` 文件中可以找到常用和已注册的端口列表。

  当数据包在网络上发送时，服务端口和 IP 地址组合形成套接字。每个数据包具有一个源套接字和目标套接字。此信息可以在监控和过滤时使用。

- **互联网**

  互联网或网络层将数据从源主机传送到目标主机。IPv4 和 IPv6 协议是互联网层协议。每一主机具有 IP 地址和前缀，用于确定网络地址。路由器用于连接网络。

- **链路**

  链路或介质存取层提供与物理介质的连接。最常见的网络类型是有线以太网 (802.3) 和无线局域网 (802.11)。每一物理设备具有一个硬件地址 (MAC)，用于标识局域网络段中数据包的目的地。

描述网络接口名称

系统上的每个网络端口都有一个名称，您可以使用该名称来配置和识别它。

旧版红帽企业 Linux 将 `eth0`、`eth1` 和 `eth2` 等名称用于各个网络接口。名称 `eth0` 是操作系统检测到的第一个网络端口，`eth1` 则是第二个，以此类推。但是，随着设备的添加和移除，检测设备并给它们命名的机制可能会改变哪个接口获得哪个名称。此外，PCIe 标准无法保证在启动时检测 PCIe 设备的顺序；鉴于设备或系统启动期间的变化，这可能会意外改变设备命名。

较新版本的红帽企业 Linux 采用另一种命名体系。系统将基于固件信息、PCI 总线拓扑及网络设备的类型来分配网络接口名称，而非基于检测顺序。

网络接口名称以接口类型开头：

- 以太网接口以 en 开头
- WLAN 接口以 wl 开头
- WWAN 接口以 ww 开头

海量视频题库 myitpub.com QQ:556462
www.52myit.com

在类型之后，接口名称的其余部分将基于服务器固件所提供的信息，或由 PCI 拓扑中设备的位置确定。

- `oN` 表示这是一个板载设备，且服务器的固件提供设备的索引编号 N。因此，`eno1` 代表板载以太网设备 1。许多服务器不提供此信息。

- `sN` 表示该设备位于 PCI 热插拔插槽 N 中。因此，`ens3` 代表 PCI 热插拔插槽 3 中的以太网卡。

- `pMsN` 表示这是一个位于插槽 N 中总线 M 上的 PCI 设备。因此，`wlp4s0` 代表位于插槽 0 中 PCI 总线 4 上的 WLAN 卡。如果该卡是一个多功能设备（可能是有多个端口的以太网卡，或是具有以太网外加其他一些功能的设备），设备名称中就可能会添加 `fN`。因此，`enp0s1f0` 代表插槽 1 中总线 0 上的以太网卡的功能 0。可能还有一个名为 `enp0s1f1` 的接口，它代表了同一设备的功能 1。

持久性命名意味着一旦您知道系统上网络接口的名称，也就知道它以后不会再发生改变。您要权衡的是不能认为有一个接口的系统就会将该接口命名为 `eth0`。

IPV4 网络

IPv4 是当今互联网上使用的主要网络协议。您应该至少对 IPv4 网络有基本的了解，以便管理服务器的网络通信。

IPv4 地址

IPv4 地址是一个 32 位数字，通常使用点号分隔的四个十进制八位字节（取值范围从 0 到 255）表示。此类地址分为两个部分：网络部分和主机部分。位于同一子网中的所有主机可以在彼此之间直接通信，无需路由器，这些主机具有相同的网络部分。网络部分用于标识子网。同一子网中的任何两台主机都不能具有相同的主机部分。主机部分用于标识子网中的特定主机。

在现代互联网中，IPv4 子网的大小是可变的。要分清 IPv4 地址中的网络部分和主机部分，管理员必须知道分配给子网的子网掩码。子网掩码指明有多少位的 IPv4 地址属于子网。可供主机部分使用的位数越多，子网中就能有越多的主机。

有时，将子网中可能达到的最低地址（主机部分的二进制值全为零）称为网络地址。在 IPv4 中，子网中可能达到的最高地址（主机部分的二进制值全为一）用于广播消息，该地址称为广播地址。

子网掩码可用两种格式表示。较早的子网掩码语法中将 24 位用于网络部分，即 `255.255.255.0`。较新的语法称为 CIDR 表示法，它指定了一个网络前缀 `/24`。两种格式都传达同样的信息，即 IP 地址中有多少前导位组成其网络地址。

以下示例说明 IP 地址、前缀（子网掩码）、网络部分和主机部分之间的关系。

海量视频题库 myitpub.com QQ:5565462
www.52myit.com

```
IP Address:
   172.17.5.3 = 10101100.00010001.00000101.00000011
                        Prefix: /16
Netmask:
  255.255.0.0 = 11111111.11111111.00000000.00000000

                10101100.00010001.00000101.00000011
                Network           Host

IP Address:
   192.168.5.3 = 11000000.10101000.00000101.00000011
                        Prefix: /24
Netmask:
255.255.255.0 = 11111111.11111111.11111111.00000000

                11000000.10101000.00000101.00000011
                Network                    Host
```

图 12.1: IPv4 地址和子网掩码

计算 192.168.1.107/24 的网络地址

主机地址	192.168.1.107	**11000000.10101000.00000001.01101011**
网络前缀	/24 (255.255.255.0)	**11111111.11111111.11111111.00000000**
网络地址	192.168.1.0	**11000000.10101000.00000001.00000000**
广播地址	192.168.1.255	11000000.10101000.00000001.11111111

计算 10.1.1.18/8 的网络地址

主机地址	10.1.1.18	**00001010.00000001.00000001.00010010**
网络前缀	/8 (255.0.0.0)	**11111111.00000000.00000000.00000000**
网络地址	10.0.0.0	**00001010.00000000.00000000.00000000**
广播地址	10.255.255.255	00001010.11111111.11111111.11111111

计算 172.16.181.23/19 的网络地址

主机地址	172.168.181.23	**10101100.10101000.10110101.00010111**
网络前缀	/19 (255.255.224.0)	**11111111.11111111.11100000.00000000**

海量视频题库 myitpub.com QQ:556462
www.52myit.com

网络地址	172.168.160.0	10101100.10101000.10100000.00000000
广播地址	172.168.191.255	10101100.10101000.10111111.11111111

特殊地址 127.0.0.1 始终指向本地系统（“localhost”），而网络 127.0.0.0/8 属于本地系统，所以它能够使用网络协议与自己通信。

IPv4 路由

不管使用 IPv4 还是 IPv6，网络流量都需要以主机到主机和网络到网络的形式进行传输。每一主机具有一个路由表，该表告诉主机如何路由特定网络的通信。路由表条目将列出目标网络、用于向其发送流量的接口，以及任何中间路由器的 IP 地址（用于将消息中继到最终目的地）。与网络流量目的地相符的路由表条目用于路由该流量。如果两个条目匹配，则使用前缀最长的那一个。

如果网络流量不匹配更为具体的路由，路由表通常具有一个代表整个 IPv4 互联网的默认路由条目：0.0.0.0/0。此默认路由指向可通达的子网上的路由器（也就是说，在主机路由中具有更具体路由的子网上）。

如果路由器收到的流量并非将其作为寻址目标，则路由器不会像普通主机那样忽略该流量，而是根据自己的路由表转发该流量。这种处理方式可能会将流量直接发送到目标主机（如果路由器恰巧与目标位于同一子网中），也可能转发到其他路由器。这种转发过程会一直进行，直到流量到达最终目标。

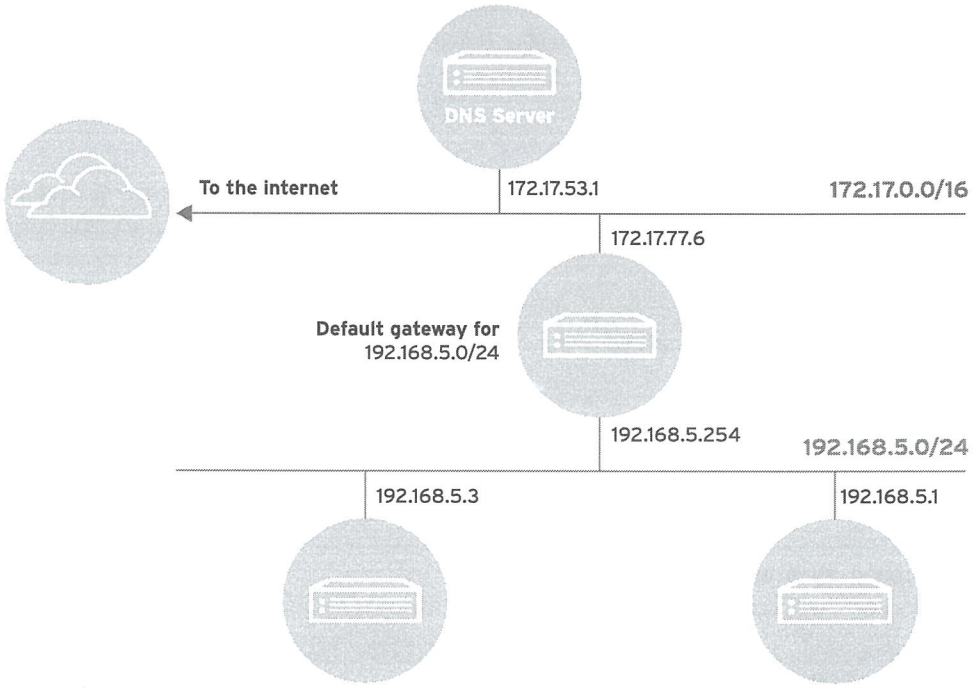

图 12.2: 网络拓扑示例

海量视频题库 myitpub.com QQ:5565462
www.52myit.com

路由表示例

目的地	接口	路由器（若需要）
192.0.2.0/24	wlo1	
192.168.5.0/24	enp3s0	
0.0.0.0/0（默认）	enp3s0	192.168.5.254

在本例中，从此主机发往 IP 地址 192.0.2.102 的流量将通过 wlo1 无线接口直接传输到该目的地，因为它与 192.0.2.0/24 路由的匹配度最高。发往 IP 地址 192.168.5.3 的流量将通过 enp3s0 以太网接口直接传输到该目的地，因为它与 192.168.5.0/24 路由的匹配度最高。

发往 IP 地址 10.2.24.1 的流量将从 enp3s0 以太网接口发送到 192.168.5.254 的路由器，该路由器将该通信转发到其最终目的地。该流量与 0.0.0.0/0 路由的匹配度最高，因为此主机的路由表中没有更加具体的路由。该路由器将使用自身的路由表来判断流量需要转发到的下一个位置。

IPv4 地址和路由配置

服务器可以在引导时从 DHCP 服务器自动配置其 IPv4 网络设置。本地客户端守护进程查询链路以获取服务器和网络设置，并获得租约以便在特定时间内使用这些设置。如果客户端未定期请求续订租约，则可能会丢失其网络配置设置。

作为替代方案，您可以将服务器配置为使用静态网络配置。在这种情况下，网络设置读取自本地配置文件。您必须从网络管理员处获取正确的设置，并根据需要手动更新它们，以避免与其他服务器冲突。

IPV6 网络

IPv6 旨在最终取代 IPv4 网络协议。您将需要了解它的工作原理，因为越来越多的生产系统使用 IPv6 寻址。例如，许多 ISP 已经将 IPv6 用于内部通信和设备管理网络，以便将稀缺的 IPv4 地址保留给客户使用。

IPv6 也可以在双栈模型中与 IPv4 并行使用。在这种配置中，网络接口可以同时具有 IPv6 地址和 IPv4 地址。红帽企业 Linux 默认在双栈模式下运行。

IPv6 地址

IPv6 地址是一个 128 位数字，通常表示为八组以分号分隔的四个十六进制半字节。每个半字节均表示 4 位的 IPv6 地址，因此每个组表示 16 位的 IPv6 地址。

```
2001:0db8:0000:0010:0000:0000:0000:0001
```

为了便于编写 IPv6 地址，不需要编写分号分隔的组中的前导零。但是，每个冒号分隔的组中必须至少写入一个十六进制数字。

```
2001:db8:0:10:0:0:0:1
```

由于带有很长的零字符串的地址很常见，一组或多组连续零可以通过正好一个 :: 块来合并。

```
2001:db8:0:10::1
```

海量视频题库 myitpub.com QQ:556546 2
www.52myit.com

注意，根据这些规则，2001:db8::0010:0:0:0:1 将成为编写示例地址的另一种不太方便的方法。但它也是相同地址的有效表示法，而这可能让不熟悉 IPv6 的管理员感到困惑。有关编写始终可读的地址的一些提示如下：

- 抑制组中的前导零。
- 使用 :: 来尽可能地缩短。
- 如果地址包含两个连续的零组，且长度相同，则最好将每个组最左边的零组缩短为 ::，最右边的组缩短为 :0:。
- 尽管允许这样做，但不要使用 :: 来缩短一组零。应改为使用 :0:，而将 :: 留给连续的零组。
- 始终对十六进制数字使用小写字母 a 到 f。

重要

如果在 IPv6 地址后面包括 TCP 或 UDP 网络端口，请始终将 IPv6 地址括在方括号中，以便端口不会被误认为是地址的一部分。

```
[2001:db8:0:10::1]:80
```

IPv6 子网划分

普通的 IPv6 单播地址分为两部分：网络前缀和接口 ID。网络前缀标识子网。同一子网上的任何两个子网接口都不能具有相同接口 ID，接口 ID 可标识子网上的特定接口。

与 IPv4 不同的是，IPv6 具有一个标准的子网掩码 /64，用于几乎所有的普通地址。在此情况下，地址的一半是网络前缀，另一半是接口 ID。这意味着单个子网可以根据需要容纳任意数量的主机。

通常，网络提供商将为组织分配一个较短的前缀，如 /48。这会保留其余网络部分以用于通过这一分配的前缀来指定子网（长度始终为 /64）。对于 /48 分配，将保留 16 位以用于子网（最多 65536 个子网）。

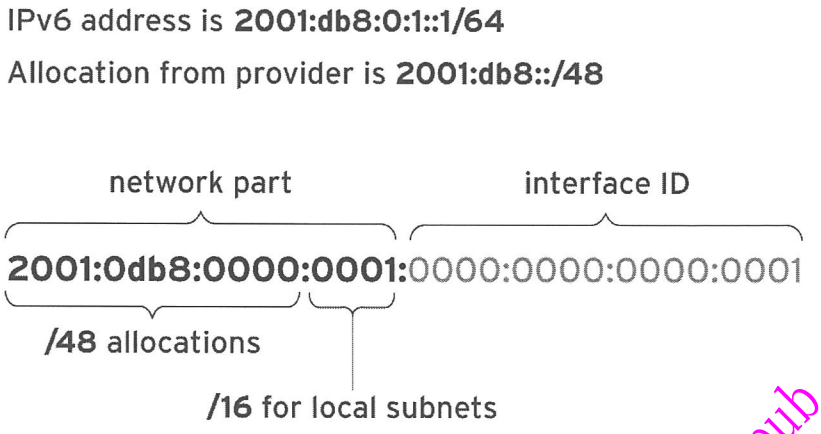

图 12.3: IPv6 地址部分和子网划分

海量视频题库 myitpub.com QQ:556546 2
www.52myit.com

常用 IPv6 地址和网络

IPV6 地址或网络	用途	描述
::1/128	本地主机	等效于 127.0.0.1/8 的 IPv6，在回环接口上设置。
::	未指定的地址	等效于 0.0.0.0 的 IPv6。对于网络服务，这可能表示其正在侦听所有已配置的 IP 地址。
::/0	默认路由（IPv6 互联网）	等效于 0.0.0.0/0 的 IPv6。路由表中的默认路由与此网络匹配；此网络的路由器是在没有更好路由的情况下发送所有流量的位置。
2000::/3	全局单播地址	“普通”的 IPv6 地址目前由 IANA 从该空间进行分配。这等同于范围从 2000::/16 到 3fff::/16 的所有网络。
fd00::/8	唯一本地地址 (RFC 4193)	IPv6 没有 RFC 1918 专用地址空间的直接等效对象，尽管这很接近。站点可以使用这些以在组织中自助分配可路由的专用 IP 地址空间，但是这些网络不能在全局 Internet 上使用。站点必须随机从该空间中选择一个 /48，但是它可以正常将分配空间划分为 /64 网络。
fe80::/10	本地链路地址	每个 IPv6 接口自动配置一个本地链路地址，该地址仅在 fe80::/64 网络中的本地链路有效。 但是，整个 fe80::/10 范围保留供本地链路以后使用。这在后面将更加详细地讨论。
ff00::/8	多播	等效于 224.0.0.0/4 的 IPv6。多播用于同时传输到多个主机，并且在 IPv6 中特别重要，因为其没有广播地址。

重要
上表列出了保留用于特定用途的网络地址分配。这些分配可能包含许多不同的网络。请记住，从全局单播和本地链路单播空间分配的 IPv6 网络具有一个标准的 /64 子网掩码。

IPv6 中的本地链路地址是一个无法路由的地址，仅用于与特定网络链路上的主机进行通信。系统上的每个网络接口都通过 fe80::/64 网络上的本地链路地址来自动配置。为确保其唯一性，本地链路地址的接口 ID 是通过网络接口的以太网硬件地址来构建的。将 48 位 MAC 地址转换为 64 位接口 ID 的一般方法是反转 7 位的 MAC 地址，然后在其两个中间字节之间插入 ff:fe。

- 网络前缀：fe80::/64
- MAC 地址：00:11:22:aa:bb:cc
- 本地链路地址：fe80::211:22ff:feaa:bbcc/64

其他计算机的本地链路地址可以由相同链路上的其他主机像普通地址那样使用。由于每个链路具有 fe80::/64 网络，不能使用路由表来正确地选择出站接口。在地址的结尾必须使用作用域标识

海量视频题库 myitpub.com QQ:556462 www.52myit.com

符来指定与本地链路地址进行通信时使用的链路。作用域标识符由 % 以及后跟的网络接口名称组成。

例如，要使用 **ping6** 对本地链路地址 fe80::211:22ff:feaa:bbcc 进行 ping 操作（使用连接到 ens3 网络接口的链路），正确的命令语法如下所示：

```
[user@host ~]$ ping6 fe80::211:22ff:feaa:bbcc%ens3
```

> **注意**
> 仅当联系具有“链路”作用域的地址时，才需要使用作用域标识符。普通全局地址的使用就如同在 IPv4 中的用法一样，并且从路由表中选择其出站接口。

多播允许一个系统将流量发送到多个系统接收的特殊 IP 地址。它与广播不同，因为只有网络上的特定系统才能接收流量。它也与 IPv4 中的广播不同，因为某些多播流量可能会路由到其他子网，具体取决于网络路由器和系统的配置。

多播在 IPv6 中比在 IPv4 中扮演着更重要的角色，因为 IPv6 中没有广播地址。IPv6 中的一个重要多播地址是 ff02::1，即全节点本地链路地址。对此地址进行 Ping 操作会将流量都发送到链路上的所有节点。与本地链路地址一样，需要使用作用域标识符来指定链路作用域多播地址（从 ff02::/8 开始）。

```
[user@host ~]$ ping6 ff02::1%ens3
PING ff02::1%ens3(ff02::1) 56 data bytes
64 bytes from fe80::211:22ff:feaa:bbcc: icmp_seq=1 ttl=64 time=0.072 ms
64 bytes from fe80::200:aaff:fe33:2211: icmp_seq=1 ttl=64 time=102 ms (DUP!)
64 bytes from fe80::bcd:efff:fea1:b2c3: icmp_seq=1 ttl=64 time=103 ms (DUP!)
64 bytes from fe80::211:22ff:feaa:bbcc: icmp_seq=2 ttl=64 time=0.079 ms
...output omitted...
```

IPv6 地址配置

在 IPv4 中，有两种方法可用于在网络接口上配置地址。可以由管理员手动在接口上配置网络地址，也可以使用 DHCP 通过网络进行动态配置。IPv6 也支持手动配置以及两种动态配置方法，其中一种便是 DHCPv6。

与 IPv4 一样，可以随意选择静态 IPv6 地址的接口 ID。在 IPv4 中，网络上有两个地址无法使用：子网中的最低地址和子网中的最高地址。在 IPv6 中，以下接口 ID 是保留的，无法用于主机上的普通网络地址。

- 由链路上的所有路由器使用的全零标识符 0000:0000:0000:0000（“子网路由器任意广播”）。（对于 2001:db8::/64网络，这可能是地址 2001:db8::）
- 标识符 fdff:ffff:ffff:ff80 到 fdff:ffff:ffff:ffff。

由于没有广播地址，DHCPv6 的工作原理与适用于 IPv4 的 DHCP 有所不同。基本上，主机将 DHCPv6 请求从其本地链路地址发送到 ff02::1:2 上的端口 547/UDP，即全 dhcp 服务器本地链路多播组。然后 DHCPv6 服务器通常向客户端的本地链路地址上的端口 546/UDP 发送一个包含相应信息的回复。

红帽企业 Linux 8 中的 dhcp 软件包提供对 DHCPv6 服务器的支持。

除了 DHCPv6 之外，IPv6 也支持另外一个动态配置方法，称为无状态地址自动配置 (SLAAC)。使用 SLAAC 时，主机通常使用本地链路 fe80::/64 地址来调出其接口。主机随后向 ff02::2（即，全

海量视频题库 myitpub.com QQ:556462 www.52myit.com

路由器本地链路多播组）发送一个“路由器请求”。本地链路上的 IPv6 路由器以网络前缀以及其他可能的信息来响应主机的本地链路地址。主机随后将该网络前缀与其通常构建的接口 ID（构建方式与本地链路地址相同）配合使用。路由器定期发送多播更新（“路由器播发”）以确认或更新其提供的信息。

红帽企业 Linux 8 中的 radvd 软件包允许基于红帽企业 Linux 的 IPv6 路由器通过路由器播发提供 SLAAC。

重要

配置为通过 DHCP 获取 IPv4 地址的典型红帽企业 Linux 8 计算机通常也配置为使用 SLAAC 来获取 IPv6 地址。当网络中添加了 IPv6 路由器时，这可能导致计算机意外获取 IPv6 地址。

部分 IPv6 部署将 SLAAC 与 DHCPv6 组合，SLAAC 仅用于提供网络地址信息，而 DHCPv6 仅用于提供其他信息（如，要配置的 DNS 服务器和搜索域）。

主机名和 IP 地址

如果您总是必须使用 IP 地址连接您的服务器，这会很不方便。人们通常更愿意使用名称而不是一长串难记的数字。因此，Linux 有多种机制可以将主机名映射到 IP 地址，统称为名称解析。

一种方法是在各个系统上的 **/etc/hosts** 文件中为每个名称设置一个静态条目。这需要您手动更新每台服务器的文件副本。

对于大多数主机，您可以借助称为域名系统 (DNS) 的网络服务，从主机名查找地址（或从地址查找主机名）。DNS 是提供主机名到 IP 地址映射的分布式服务器网络。为使名称服务起作用，主机需要指向某一个名称服务器。该名称服务器无需与主机位于同一子网上，只需可供主机访问即可。这通常通过 DHCP 或名为 **/etc/resolv.conf** 文件中的静态设置来配置。本章后续小节将讨论如何配置名称解析。

海量视频题库 myitpub.com QQ:556546 2
www.52myit.com