



CentOS-Stream9 上安装 Postgresql 17 from Source Code

原创

ByteHouse

2024-10-08

395

1.下载指定版本的源码

下载地址：http://www.postgresql.org/ftp/source/ ,下载17 版本为例

wget https://ftp.postgresql.org/pub/source/v17.0/postgresql-17.0.tar.gz

2.安装依赖

dnf install -y gcc gzip bzip2 tar perl perl-ExtUtils-Embed readline-devel zlib-devel pam-de

在 17 版本中，新增两个语法解析器？在编译安装 16 版本时，未安装以下两个解析器。

Bison - 一个强大的语法分析器生成器

Bison是Linux下的一个语法分析器生成器，用于将上下文无关文法转换为C代码，简化编译器或解释器开发。它提供性能优化和灵活的语义动作定制，常用于创建解析器，如SQL解析器或自定义脚本语言解释器。通过编写.y文件定义语法规则，使用Bison生成解析器代码，然后集成到项目中，搭配词法分析器如Flex使用。Bison帮助开发者专注于应用逻辑，而非解析器实现。

在软件开发中，特别是当涉及到编译器或解释器的编写时，语法分析是一个至关重要的步骤。Bison，作为一个强大的语法分析器生成器，极大地简化了这一复杂过程。本博客将带您深入了解 Bison，包括其基本概念、使用方法以及它在项目中的应用。

什么是 Bison？

Bison（之前称为 Yacc - Yet Another Compiler-Compiler）是一个将上下文无关文法（Context-Free Grammars）转化为 C 语言代码的工具。这些生成的代码可以进一步被用作解析器（Parser），将输入的文本字符串转化为抽象语法树（Abstract Syntax Tree, AST）或其他形式的中间表示。

为什么要使用 Bison？

- 简化开发：通过 Bison，开发人员可以专注于语法规则的定义，而不需要手动编写复杂的解析器代码。

• 性能：生成的 C 代码可以经过优化，以提供高效的解析性能。

• 灵活性：Bison 支持多种类型的语法和语义动作，允许开发人员根据需求定制解析器的行为。

在大多数 Linux 发行版中，您可以通过包管理器来安装 Bison。

dnf install bison

Flex

在Linux上，Flex是一个非常强大的词法分析器生成器。它通常用于编写编译器和解释器中的词法分析器。Flex可以根据用户定义的正则表达式规则，将输入文件分割为一系列的标记(token)，然后将这些标记传递给后续的语法分析器进行进一步处理。

要在Linux上使用Flex命令，首先需要安装Flex工具。可以通过以下命令安装Flex：

dnf install flex



ByteHouse

LV3

关注

90

文章

41

粉丝

33K+

浏览量

- 获得了 109 次点赞
- 内容获得 35 次评论
- 获得了 178 次收藏

TA的专栏

ORACLE02.Data Guard

收录 14 篇内容

ORACLE01.Database

收录 18 篇内容

ORACLE03.Golden Gate

收录 5 篇内容

<>

热门文章

ORA-12154: TNS:could not resolve the connect identifier specified

2024-04-27

2590浏览

总结：Automatic SQL Tuning + Optimizer Statistics Advisor 带来的 SYS AUX ...

2024-04-30

1866浏览

040001.pg_rewind 修复主备环境的时间线

2022-07-30

1220浏览

Oracle ASM 存储的剿灭战

2024-05-12

1161浏览

standby database 告警日志 ORA-00313 ORA-00312 ORA-27037

2024-05-27

1000浏览

最新文章

dbfs目录被core.XXXX 塞满

6天前

154浏览

RMAN SWITCH 命令

2025-07-19

149浏览

Postgresql 17新特性 – 块级别增量备份的实验

2025-07-07

164浏览

DBA 每一天日常，从清理归档日志开始

2025-06-11

369浏览

备库数据文件自动创建错误引发的 ORA-16766: Redo Apply is stopped

2025-05-27

196浏览

安装完成后，可以使用flex命令来生成词法分析器。

3.源码编译安装

3.1.编译postgresql源码

```
tar -zxvf postgresql-17.0.tar.gz
cd postgresql-17.0

./configure --prefix=/opt/postgresql --with-systemd
```

在生产环境，一般都会用 systemd 来管理服务，而非直接使用 pg_ctl 来进行数据库启停操作。源码编译时，需要传递参数 --with-systemd 源码会对此进行一些优化处理，比如服务启动超时时间。但使用此参数需要先安装依赖 systemd-devel，否则会报错：

```
configure: error: header file <systemd/sd-daemon.h> is required for systemd support
```

编译安装，依据cpu核心数，可以指定多个并行度加快编译速度：

```
make -j 4 && make install
```

如是想将contrib下的扩展一同安装，使用

```
make world
make install world
```

清理编译时产生的临时文件

```
make clean
```

3.2.创建用户、用户组

```
groupadd postgres
useradd -g postgres postgres

echo "Admin123" | passwd --stdin postgres
```

3.3.创建数据主目录

```
mkdir -p /opt/postgresql/{pgdata,archive,scripts,backup,soft}

chown -R postgres:postgres /opt/postgresql
chmod -R 775 /opt/postgresql
```

3.4.配置环境变量

```
cat >> /etc/profile <<EOF

# Postgresql Environment Setting
export PGHOME=/opt/postgresql
export PGDATA=/opt/postgresql/pgdata
export LD_LIBRARY_PATH=/opt/postgresql/lib
export PATH=$PATH:$HOME/bin:/opt/postgresql/bin
EOF
```

立即生效

目录

- 1.下载指定版本的源码
- 2.安装依赖
 - Bison – 一个强大的语法分析器生成器

```
source /etc/profile
```

3.5初始化数据库

```
[root@postgresql-server ~]# su - postgres
[postgres@postgresql-server ~]$ /opt/postgresql/bin/initdb -D /opt/postgresql/pgdata/

.....
performing post-bootstrap initialization ... ok
syncing data to disk ... ok

initdb: warning: enabling "trust" authentication for local connections
initdb: hint: You can change this by editing pg_hba.conf or using the option -A, or --auth-

Success. You can now start the database server using:

    /opt/postgresql/bin/pg_ctl -D /opt/postgresql/pgdata/ -l logfile start

[postgres@pgcp-01 ~]$
```

initdb:警告：为本地连接启用“信任”身份验证
initdb:hint：您可以通过编辑 pg_hba.conf 或在下次运行 initdb 时使用选项 -A或--auth-local和--auth-host来更改这一点。
成功现在可以使用以下命令启动数据库服务器：
/opt/postgresql/bin/pg_ctl -D /opt/postgresql/pgdata/ -l logfile start

启动实例

```
[postgres@pgcp-01 ~]$ /opt/postgresql/bin/pg_ctl -D /opt/postgresql/pgdata/ -l logfile star
waiting for server to start.... done
server started
[postgres@pgcp-01 ~]$
```

创建数据库

```
[postgres@pgcp-01 ~]$ /opt/postgresql/bin/createdb byte
[postgres@pgcp-01 ~]$ /opt/postgresql/bin/psql byte
psql (17.0)
Type "help" for help.

byte=# \l

                                List of databases
  Name      | Owner   | Encoding | Locale Provider | Collate  |  Ctype   | Locale | I
-----+-----+-----+-----+-----+-----+-----+--
 byte       | postgres | UTF8     | libc            | en_US.UTF-8 | en_US.UTF-8 |      | |
 postgres   | postgres | UTF8     | libc            | en_US.UTF-8 | en_US.UTF-8 |      | |
 template0  | postgres | UTF8     | libc            | en_US.UTF-8 | en_US.UTF-8 |      | |
            |          |          |                  |            |            |      | |
 template1  | postgres | UTF8     | libc            | en_US.UTF-8 | en_US.UTF-8 |      | |
            |          |          |                  |            |            |      | |
(4 rows)

byte=#
```

3.6.配置服务

PostgreSQL的开机自启动脚本位于postgresql源码目录的contrib/start-scripts路径下。

```
[root@postgresql-server start-scripts]# pwd
/root/postgresql-17/contrib/start-scripts
[root@postgresql-server start-scripts]# ls -la
total 12
drwxrwxrwx.  3 1107 1107   47 Nov  7 06:17 .
drwxrwxrwx. 61 1107 1107 4096 Nov  7 06:17 ..
-rw-r--r--.  1 1107 1107 1441 Nov  7 06:04 freebsd
-rw-r--r--.  1 1107 1107 3526 Nov  7 06:04 linux
drwxrwxrwx.  2 1107 1107   84 Nov  7 06:17 macos
[root@postgresql-server start-scripts]#
```

Linux文件即为Linux系统上的启动脚本,该脚本是Init 风格的。

创建 Systemd 风格的服务脚本：

```
cat > /etc/systemd/system/postgresql.service <<"EOF"
[Unit]
Description=PostgreSQL database server
Documentation=man:postgres(1)
After=network.target

[Service]
Type=forking
User=postgres
Group=postgres
Environment=PGPORT=5432
Environment=PGDATA=/opt/postgresql/pgdata
OOMScoreAdjust=-1000
ExecStart=/opt/postgresql/bin/pg_ctl start -D ${PGDATA} -s -o "-p ${PGPORT}" -w -t 300
ExecStop=/opt/postgresql/bin/pg_ctl stop -D ${PGDATA} -s -m fast
ExecReload=/opt/postgresql/bin/pg_ctl reload -D ${PGDATA} -s
KillMode=mixed
KillSignal=SIGINT
TimeoutSec=0

[Install]
WantedBy=multi-user.target
EOF
```

4.连接数据库服务

本地连接方式连接pgsql

```
[root@postgresql-server data]# su - postgres
Last login: Wed Nov 15 16:31:31 CST 2023 on pts/1
[postgres@postgresql-server ~]$ psql
psql (16.1)
Type "help" for help.

postgres=#
```

远程访问

修改pg_hba.conf

```
[root@postgresql-server data]# cat pg_hba.conf

# -----
#
# If you want to allow non-local connections, you need to add more
# "host" records.  In that case you will also need to make PostgreSQL
# listen on a non-local interface via the listen_addresses
# configuration parameter, or via the -i or -h command line switches.


# TYPE      DATABASE        USER            ADDRESS              METHOD

# "local" is for Unix domain socket connections only
local      all             all                                peer
# IPv4 local connections:
host       all             all             127.0.0.1/32         scram-sha-256
# IPv6 local connections:
host       all             all             ::1/128              scram-sha-256
# Allow replication connections from localhost, by a user with the
# replication privilege.
local      replication     all                                peer
host       replication     all             127.0.0.1/32         scram-sha-256
host       replication     all             ::1/128              scram-sha-256
```

参数作用：

- TYPE：连接类型一般有 local 和 host 两种，local 指的是本地连接，host 则类似外部的ssh远程服务器的访问方式。
- DATABASE：表示要连接的数据库，all 表示所有。这里可以改为具体的数据库名称
- USER：这里填写Linux用户。比如默认分配的postgres可以免密访问。
- ADDRESS：127.0.0.1/32表示只允许来自自己本机的连接，0.0.0.0/0 表示所有的ip放行。192.168.1.0/24表示允许192.168.1.1-192.168.1.255这个地址段的ip地址连接。
- METHOD：表示连接的认证方式，PostgreSQL的连接命令psql有两种连接方式：
 - 不带-h参数或host参数时，是local连接，用的是peer认证方式。
 - 使用-h localhost、-h 127.0.0.1、postgres@localhost 或 postgres@127.0.0.1 这样的格式，则会使用host类型，使用TCP/IP的方式连接，使用的是ident的认证方式。

为了实现外部远程访问，配置做如下修改：

```
host      all      all      0.0.0.0/0      trust
```

pg_ident.conf

数据库映射文件，ident 认证方式的扩展，标注操作系统用户与数据库用户的映射关系，配合pg_hba.conf 使用。允许数据库服务器上指定的操作系统用户，使用指定的数据库用户，免密连入数据库。

pg_ident.conf 文件

```
# MAPNAME      SYSTEM-USERNAME    PG-USERNAME
      ss              aaa              test
      ss              syd              syd
```

- MAPNAME：映射名，自定义配置在 pg_hba.conf 文件中。
- SYSTEM-USERNAME：系统用户名。
- PG-USERNAME：数据库用户名。

pg_hba.conf 文件

#	TYPE	DATABASE	USER	ADDRESS	METHOD
	local	all	all		md5
	host	all	all	127.0.0.1/32	md5
	host	all	all	0.0.0.0/0	ident map=ss
	host	all	all	:::1/128	md5
	local	replication	all		md5
	host	replication	all	127.0.0.1/32	md5
	host	replication	all	:::1/128	md5

重新加载即可生效

```
pg_ctl reload
```

重启postgresql之后我们继续执行psql，发现报错还是存在，root还是无法访问：

```
[root@postgresql-server data]# psql
psql: error: connection to server on socket "/var/run/postgresql/.s.PGSQL.5432" failed: FATAL:
[root@postgresql-server data]#
```

修改 postgresql.conf 文件

通过vim /var/lib/psql/pg版本号/postgresql.conf ，我们找到下面的内容，把listen_addresses的注释放开，并且把内容修改为*，然后wq保存退出即可。

```
#-----
# CONNECTIONS AND AUTHENTICATION
#-----

# - Connection Settings -

listen_addresses = '*'          # what IP address(es) to listen on;
```

```
[root@postgresql-server data]# psql -h 192.168.10.44 -U postgres
psql (16.1)
Type "help" for help.

postgres=#
postgres=# \conninfo
You are connected to database "postgres" as user "postgres" on host "192.168.10.44" at port
postgres=#
```

🔗 墨力计划

「喜欢这篇文章，您的关注和赞赏是给作者最好的鼓励」

关注作者

赞赏

【版权声明】本文为墨天轮用户原创内容，转载时必须标注文章的来源（墨天轮），文章链接，文章作者等基本信息，否则作者和墨天轮有权追究责任。如果您发现墨天轮中有涉嫌抄袭或者侵权的内容，欢迎发送邮件至：contact@modb.pro进行举报，并提供相关证据，一经查实，墨天轮将立刻删除相关内容。

评论

分享你的看法，一起交流吧～

相关阅读

腾讯iOA企业级安全办公解决方案

若城 110476次阅读 2025-08-08 13:45:43

【DBA坦白局】第三期：作为DBA，你加过最晚的班是到几点？在干什么？

墨天轮编辑部 1415次阅读 2025-07-15 10:28:44

2025年8月中国数据库排行榜：双星竞入三甲榜，TDSQL 连跃位次升

墨天轮编辑部 1100次阅读 2025-08-07 16:22:24

2025年7月国产数据库大事记：GoldenDB创千万级大单，可信数据库大会召开，openGauss HyBench打榜第一，电科金仓举办2025产品发布会……

墨天轮编辑部 846次阅读 2025-08-05 17:20:33

IDC报告：2024中国金融行业集中式事务型数据库市场破11.6亿元，Oracle领跑、达梦强势追赶

通讯员 674次阅读 2025-07-16 16:59:44

优炫数据库在山东省寿光市人民检察院成功应用！

优炫软件 648次阅读 2025-07-16 09:50:01

重磅发布：Oracle ADG 一键自动化搭建脚本

Lucifer三思而后行 611次阅读 2025-07-17 17:04:48

2025年7月国产数据库中标情况一览：长沙银行千万采购GoldenDB，秦皇岛银行近七百万采购TDSQL！

通讯员 608次阅读 2025-08-07 10:18:23

重磅 | 万里数据库GreatDB亮相上合组织数字经济论坛 以硬核科技共绘“数字丝路”新图景

万里数据库 592次阅读 2025-07-15 09:45:41

中国信通院2025上半年“可信数据库”新增标准解读

大数据技术标准推进委员会 567次阅读 2025-07-21 10:45:15