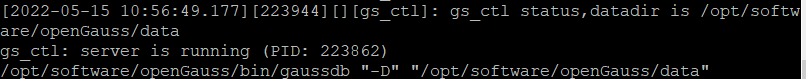
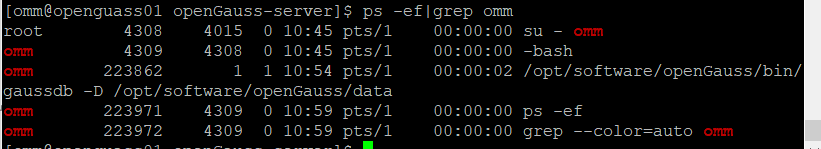
**关卡1：**

1. **验证数据库的状态：**

****

1. **验证数据库进程：**

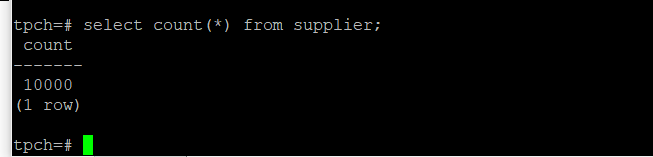
****

**思考题：**为什么需要通过源码编译，完成数据库的安装？

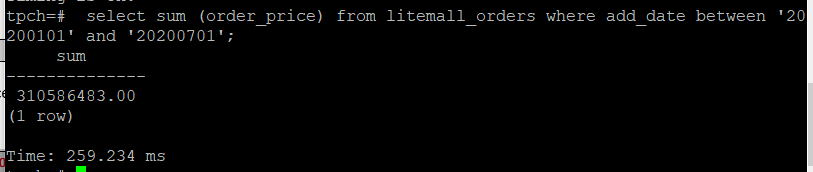
Opengauss数据库是在linux服务器上安装的，而为了能在Linux上运行，必须是二进制文件，所以拿到数据库源码后，需要将它编译成二进制文件才能成功运行。

**关卡2**

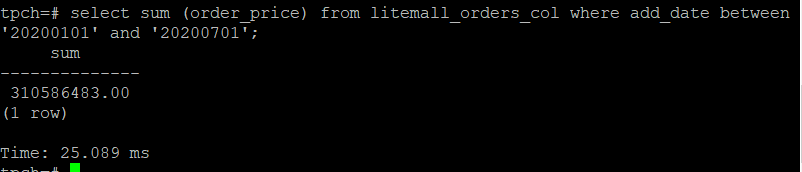
1. **登录数据库验证，返回查询结果：**

****

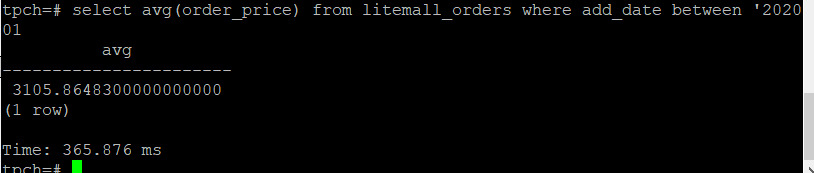
1. **使用SQL语句查询，2020年上半年litemall\_orders行存表中的order\_price的总和：**

****

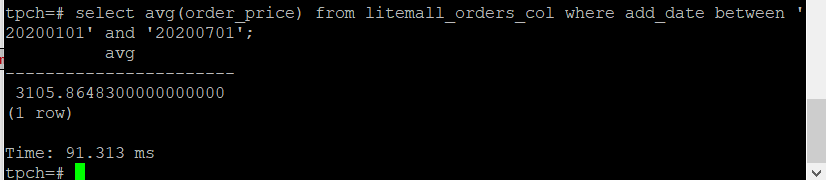
1. **使用SQL语句查询，2020年上半年litemall\_orders\_col列存表中的order\_price的总和：**

****

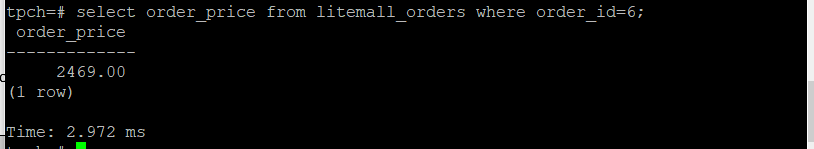
1. **使用SQL语句查询，2020年上半年litemall\_orders行存表中的order\_price的平均值：**

****

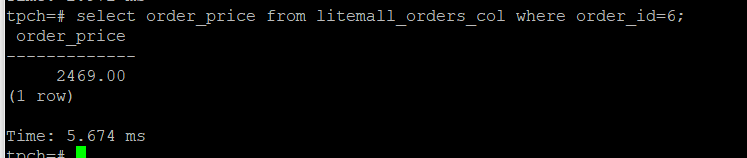
1. **步骤7**

****

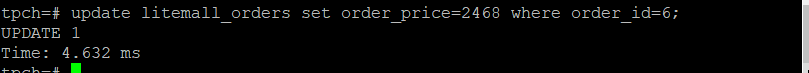
1. **行**

****

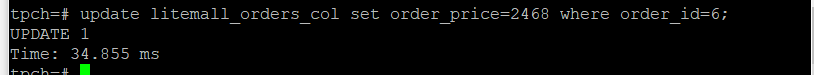
1. **列**

****

1. **行**

****

1. **列**

****

1. **对比行存表列存表的操作，得出结论：**

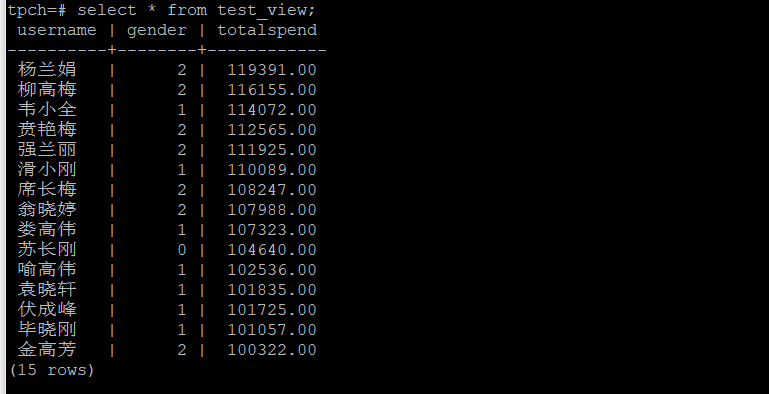
**行式数据库写入是一次性完成，保证了数据的完整性，写入方便，但是读取麻烦并且可能会有冗余列的操作；列式数据库读取方便，无冗余性问题，但把一行记录拆成单列保持，写入次数比行式多，写入较慢。所以行式存储更适用于传统的关系型数据库，通常用于业务数据存储；而列式存储则适用于进行数据分析的分析型数据库，通常用于分析数据存储。**

**思考题：行存表与列存表在执行相同的SQL语句时，为何执行的时间不同？在执行哪些类型SQL时，行存表效率更高？在执行哪些类型SQL时，列存表效率更高？**

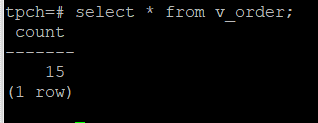
**行存表与列存表的写入与读取操作不同，即读取和写入的次数不相同，造成执行时间上的差异。执行查询类SQL时列存表效率高，执行写入类SQL时行存表效率高。**

**关卡3：**

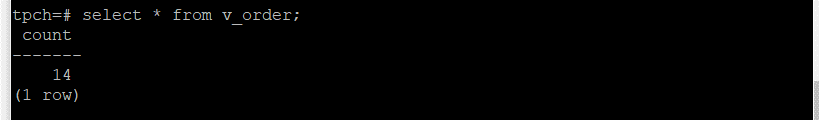
1. **查表**

****

1. **使用物化视图统计人数，查询物化视图结果，将执行结果截图。**

****

1. **使用物化视图统计人数，查询物化视图结果，将执行结果截图。**

****

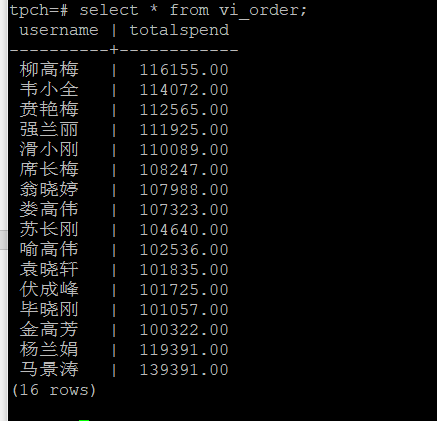
1. **查询物化视图：**

****

1. **查：**

****

1. **查：**

****

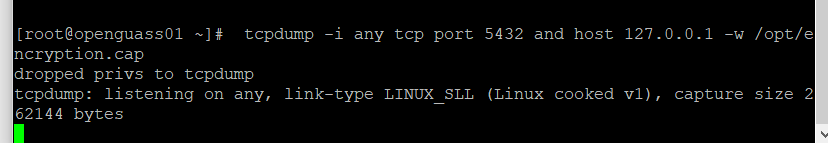
**思考题：对比全量物化视图与增量物化视图，得出区别；物化视图适用那些使用场景？**

**全量物化视图的全量刷新是完全刷新整个物化视图，相当于重新生成物化视图，增量物化视图的增量刷新是依照相应规则对物化视图进行更新，对大表特别有效，而大表全量刷新非常慢。**

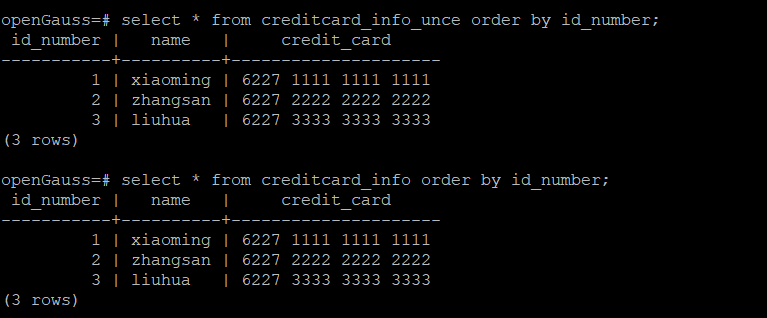
**适用场景：加速查询、数据仓库、数据集成、数据备份等。**

**关卡4**

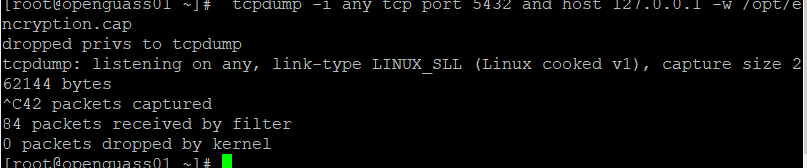
1. **抓取数据流：**

****

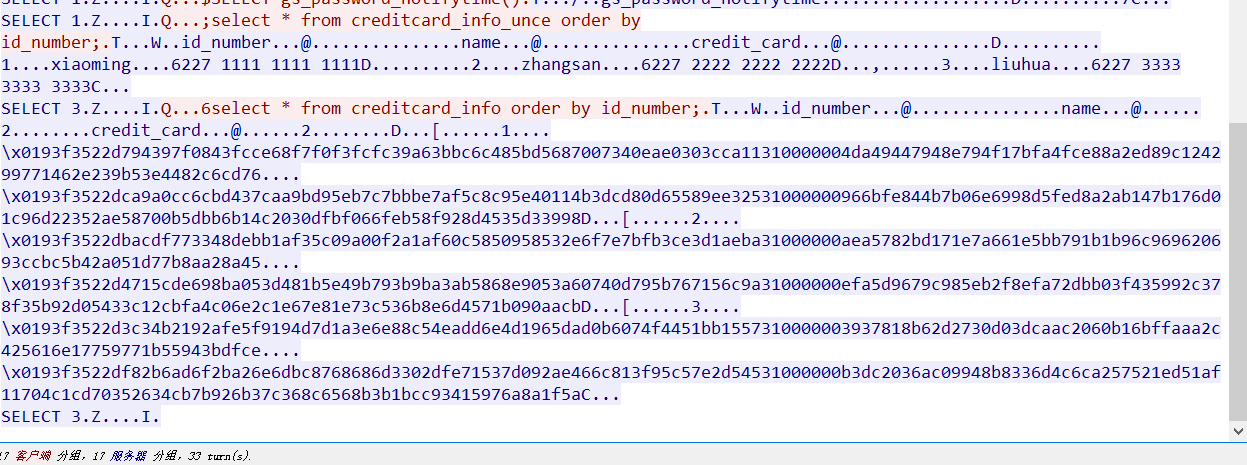
1. **查询加密表和非加密表：**

****

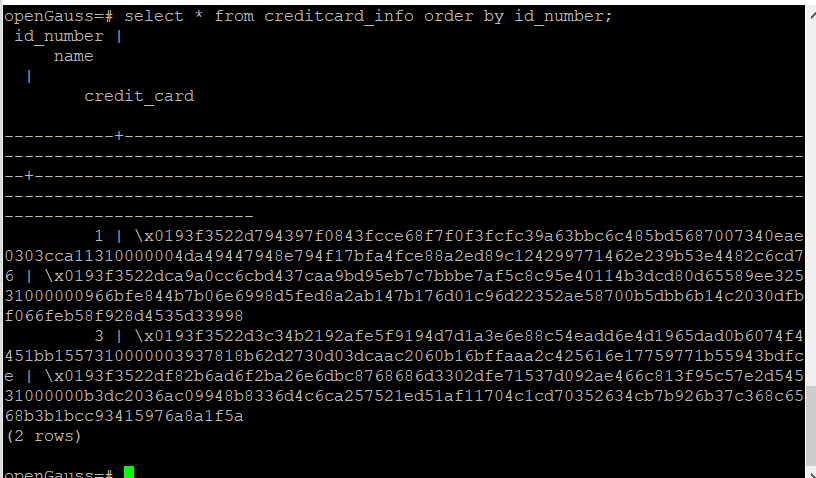
1. **暂停抓取：**

****

1. **追踪流：**

****

1. **使用不加密态功能的客户端登录查询：**

****

1. **对比密态数据库场景下和非密态数据库场景下对加密表做增删改查的区别。**

**在密态数据库机制下，参数加密完成后整个查询任务被变更成一个加密的查询任务并通过安全传输通道发到数据库服务端,由数据库服务端完成基于密文的查询检索。检索得到的结果仍然为密文,并最终返回客户端进行解密，而在非密态数据库则为明文。**

**思考题：数据实际存储在物理磁盘上的时候是明文还是密文？数据的加解密的动作是在客户端完成的还是服务端完成的？**

**数据是以明文的形式存储在物理磁盘上；对于加密通信，客户端和服务端都采用非对称加密方法进行加解密。**