

利用Oracle许可实现更好控制Linux (2) PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/143/2021\\_2022\\_\\_E5\\_88\\_A9\\_E7\\_94\\_A8Orac\\_c102\\_143414.htm](https://www.100test.com/kao_ti2020/143/2021_2022__E5_88_A9_E7_94_A8Orac_c102_143414.htm) 数据和行为的耦合 将代码封装进Oracle的存储过程里使得Oracle的设计人员能够将数据库

实体同作用于这些实体的代码紧密地耦合在一起。例如，Oracle的成员方法（member method）能够为客户表格而创建，这样Oracle就知道数据库同什么代码相关联。这同面向对象的模型是相同的，在面向对象的模型里，方法是同C或者Java对象的类相关联的。数据库代码的隔离 由于最终用户的屏幕上没有过程代码或者SQL，因此所有对数据库的访问都是通过调用标准的存储过程而实现的。这就使得应用程序不再依赖于数据库，这意味着该应用程序能够被轻易地移植到另一个数据库，而不要对前端的代码作出任何修改。数据访问的严密控制 通过使用定义者权利，最终用户只有在使用存储过程和函数的情况下才能够访问数据库。这就使得过程代码能够严密地控制访问规则并消除任何从后门对数据库的访问。许可执行模式所能做到的还不止对Oracle表格的访问控制。由于存储过程会控制对数据库的访问，因此存储过程能够定义过程规则，而最终用户需要这些规则才能够看到Oracle的数据。行这一层、列这一层，以及依赖于数据的访问规则都是在Oracle存储过程的PL/SQL或者Java内被编码的。拒绝后门 最终用户只有在其执行存储过程的时候才拥有对数据库的（访问）权限，超出了其过程的范围，他们就无法访问Oracle了。和所有的Oracle安全模式一样，许可执行模式不是没有局限性的。现在让我们来看看其中的一些。许可执行安全模

式的局限性 许可执行安全模式最适合于正式的系统设计项目，在这样的项目里，项目分析人员会在开始编程之前仔细地规划访问代码和规则。许可执行模式的主要局限性有：需要仔细的前期设计 专用系统的开发并不是很容易就能够实现的，因为其设计需要对主要的程序包、存储过程和函数预先进行定义。可选的过程语言有限 Oracle的函数和存储过程的代码需要在PL/SQL或者Java里编写。但是，有些Oracle的设计人员要求只有SQL才能够被保存在存储过程里。这就使得开发人员能够使用他们所希望使用的任何过程语言。前端并没有被要求完全独立于代码，它包含有过程代码，但是所有的数据库SQL都会被存储过程和函数调用所替代。难以审计 由于最终用户只有在执行存储过程的时候才能够具有访问数据库的权限，因此创建列表来包括数据库实体以及那些能够访问这些数据库实体的最终用户是很困难的。对定义者权利和许可执行安全的审计需要你编写复杂的审计程序，以剖析和解释这些存储过程。许可的执行 正如你能够看到的，许可执行这一方法同传统的许可安全模式相比具有很多优势，但是它也存在某些不足之处。由于Oracle是最强大和最灵活的数据库，所以你还有其他的方法来控制用户对数据的访问。Oracle为数据访问的控制提供了很多选择，而你的任务就是去选择和实现能够满足你设计要求的最佳访问控制法。

100Test 下载频道开通，各类考试题目直接下载。详细请访问 [www.100test.com](http://www.100test.com)