数据库用户名和密码安全

数据库用户名和密码安全

在今天我们更加需要注意确保数据库安全。数据库安全当然包括数据库的用户名和密码安全问题。本文介绍设置用户名和密码的最佳方法、在授予或撤消用户访问权限时我们应该注意的事项以及我们能采取的真正确保用户名和密码安全的方法。

创建用户名 vs.设置密码

若你想创建不能自行修改密码的用户，唯一的方法就是执行密码验证函数，但很少会成功。所以，如果你的公司和应用程序允许的话，应该为文件的所有参数设值。记住今天更加需要保证数据库安全，但密码修改应该遵循基本的原则。为了达到这一目的，我们可以对文档设置复杂密码。

- 如何创建不能自行修改密码的用户
- 设置密码的最佳方法
- 设置复杂密码

授予和撤销用户访问权限

仅仅创建密码文档还不够，还必须让 Oracle 使用它。如果想对其他用户密码作稍微的修改，不改变系统文件及其密码的，我们也可以通过利用相同的密码登录。通过 GUI 增加用户权限改变 System 初始默认值授予用户管理和使用数据库的部分权限，如 select、update 和 insert 等。在本节中还提到了撤销 ALTER USER X IDENTIFIED BY Y 权限的操作方法。

- 授予用户修改密码的权限
- 授权失败 vs. 密码文档
- 通过 GUI 增加用户权限 改变 System 初始默认值

- 撤销 ALTER USER X IDENTIFIED BY Y 的权限

确保用户名和密码安全

创建新的应用软件程序时，我们是否该创建一个包含多个的用户名及其密码的表还是应该创建不同级别的用户表？最好的办法就是为每个用户设立数据库帐号。在 Windows 平台上确保密码安全的最佳方法就是不用密码登录数据库。对于无法通过 IDS 但是能通过 SQL*PLUS 登录的用户，我们有两种方法阻止。在本节里详还细介绍能够用到的相关性能。

- 确保创建用户名安全的最佳方法
- 如何确保用户名和密码安全
- 锁定 SQL*PLUS 命令安全
- 用 as sysdba 登录时发现安全漏洞该怎么办？

隐藏密码

对于存储了用户名和密码的表，我们采取的办法就是变换密码而不是加密。如果忘记了口令，用户只需要重新设置一个密码。在访问数据库时一般都要避免使用符合验证标准但是相当明显的密码。如果一名非 DBA 用户想访问 internal，为每一名有使用权限的用户创建帐号的最佳方法就是更改用户。而当你拥有两个或两个以上的数据库时，可能要求你指明你想登录的数据库的名称。在这种情况下，你需要隐藏 Oracle 数据库的密码。当你通过 Corn 运行程序时（在 Unix 包上），就要求你隐藏 Oracle 用户的密码。如何隐藏密码是本节重点解决的问题。

- 数据库表应该加密吗？
- 保护 Internal 的密码
- 使用隐藏的密码登录
- 如何隐藏 Oracle 密码

- 如何隐藏用户密码

不用密码访问数据库

使用管理员或者 ORA_DBA 组成成员的身份登录 Windows 系统，在忘记 SYS 密码的情况下你可以不使用密码登录数据库。如果没有密码可以用/ as sysdba 语句登录，而 system as sysdba 语句却实现不了。若想不用密码也能运行 SQL*PLUS 脚本，本文详细介绍了两种处理方法。

- 忘记 SYS 密码该怎么办？
- 为什么不用密码也能够登录 SYSTEM
- 不用密码也能运行 SQL*PLUS 脚本的最佳方法

设置密码的最佳方法

问：在dba_profile表里，设置密码和其他参数的最佳方法是什么？

答：我认为如果您公司和应用程序允许的话，应该为文件的所有参数设值。我会先进行如下操作：

```
failed_login_attempts 3
password_life_time 60
password_reuse_time 250
password_reuse_max unlimited
password_lock_time unlimited
password_grace_time 7
password_verify_function your_custom_func_here
```

作者：Dan Norries 译者：April 来源：TT中国

如何创建不能自行修改密码的用户

问：我想创建一个不能自行修改密码的用户，该怎么办呢？

答：遗憾的是，所有的用户都有自行修改密码的权限。创建不能自行修改密码的用户的一种方法就是执行密码验证函数（但很少会成功）。然后将密码验证函数分配给用户。

记住今天更加需要确保数据库安全。所以密码修改应该遵循基本原则。不让用户修改密码这个主意并不好。

作者：Brian peasland 译者：April 来源：TT 中国

设置复杂密码

问：我想在一个文档上设置复杂密码，怎样才能记录并安装这种功能？每次都要修改密码吗？
我的 DBA 办公室安装的是 Oracle 8.1.6 ， Solaris 2.7。

答：如果你没有阅读过密码验证功能示例，请到 Oracle 网站 $ORACLE_HOME/rdbms/admin 目录中查询 utlpwdmg.sql。你可以自行修改。

如果已有这种验证功能，请将文档更改为你所指定的密码验证功能。

例如：
脚本在密码管理系统中更改默认值，这就意味着所有用户都拥有密码管理系统。在创建另一密码文档之前，这些密码还有效。

```
ALTER PROFILE DEFAULT
LIMIT PASSWORD_LIFE_TIME 60
PASSWORD_GRACE_TIME 10
PASSWORD_REUSE_TIME 1800
PASSWORD_REUSE_MAX UNLIMITED
FAILED_LOGIN_ATTEMPTS 3
PASSWORD_LOCK_TIME 1/1440
PASSWORD_VERIFY_FUNCTION verify_function;
```

如想了解更多信息，还可以查看 Oracle 相关资料（尤其是下面两份资料：

Oracle8i Administrator's Guide - Chapter 22 Establishing Security Policies
Oracle8i SQL Reference - ALTER PROFILE, CREATE PROFILE Statements

作者：Karen Morten 译者：April 来源：TT 中国

授予用户修改密码的权限

问：我有一个用户，他需要修改另外一些用户的密码。但是我不想让他改变系统文件及其密码。我不知道该怎么办。

答：我以前回答过一个类似问题。你可以查看 http://searchoracle.techtarget.com/ateQuestionNResponse/0,289625,sid41_cid581632_tax296096,00.html. 我认为你能够利用同一密码来达到你只作稍微修改的目的。注意启用触发器的这些用户需要直接被授予ALTER USER系统权力，而不是通过别人来达到这一目的。

作者：Dan Norries 译者：April 来源：TT 中国

通过 GUI 增加用户权限 改变 SYSTEM 初始默认值

问：最近我在服务器和客户机上安装了 Orace8i，我怎样改变系统文件和经理的初始默认值？我还需要授予一些用户管理权限，授予一些用户使用权限，如 select、update 和 insert。这些通过 GUI 能达到吗？

答：要更改用户密码，使用 DB A 用户权限登陆数据库。SYSTEM 帐户就可以登陆。然后再发出以下指令：

ALTER USER username IDENTIFIED BY newpassword;

通过 GUI 你就可以替换以上的用户名和密码。Oracle 公司的 GUI 管理工具叫做 Oracle Enterprise Manager。你需要安装这一工具并且正确设定它以便于管理数据库，包括用户密码和权限都需要你设定。

作者：Brian peasland 译者：April 来源：TT 中国

授权失败 vs. 密码文档

问：我有一个问题。我使用oradim -new -sid db7 -intpwd db7 创建了一个数据库密码文档，并且拥有 4 个用户。第一个用户使用sysdba时，就会出现“ORA-1994：授权失败：不能将用户加入公共密码文档”。请问为什么会出现这样的错误呢？谢谢。

答：仅仅创建密码文档还不够。你同时还必须给让Oracle使用密码文档。若要达到这一目的，需在参数文档中设定REMOTE_LOGIN_PASSWORDFILE初始化参数。

作者：Brian peasland 译者：April 来源：TT 中国

撤销 ALTER USER X IDENTIFIED BY Y 权限

问:我想撤消 alter USER X IDENTIFIED BY Y 文件的权限。虽然这一权限没有明确授予，我知道它已连接。即使我授权用户如创建会话、创建表格、创建类型等等，我发现用户 x 被授予更改密码的权限。作为用户，我同意他更改密码，但是当许多用户都在修改密码时，我该怎样阻止其中一个用户修改密码呢？

答:Oracle 默许用户更改密码。你也没有被授予取消用户修改密码的权力。因此，你不得不从另一个角度来解决这个问题。在 Oracle8i 或以上版本中，你能在 DDL 语句（如 alter）中创建触发器，还能利用这一特征创建触发器阻止用户修改帐号。以下是在 Oracle10g 以及其他版本的中运行的示例：

```
create OR REPLACE TRIGGER no_alter_user
BEFORE alter
ON DATABASE
BEGIN
```

如果只有这一个触发器，我们没有必要检查 ora_sysevent 文件。

激发 alter 语句：

```
IF (    ora_dict_obj_type =  'USER'
AND ora_dict_obj_name =  ora_login_user
AND ora_login_user    != 'SYS'
AND ora_login_user    != 'SYSTEM'
) THEN
DBMS_OUTPUT.PUT_LINE('altering the user: '||ora_dict_obj_name);
DBMS_OUTPUT.PUT_LINE('I am logged in as: '||ora_login_user);
RAISE_APPLICATION_ERROR(-
20000,'You are not allowed to alter yourself!');
END IF;
END NO_alter_USER;
/
```

注意避开数据库管理员用户，因为他们可能想修改密码。密码纯粹是他们的一个起点，你将承担相关部署的一切责任。至于如何利用一个相近的触发器创建自定义审计跟踪，你可进一步查出 MetaLink NOTE 199455.1。

作者：Dan Norries 译者：April 来源：TT 中国

确保创建用户安全的最佳方法

问：我有一个关于在数据库里用户创建和安全问题。创建新的应用软件程序时，我们是否该创建一个包含多个用户名及相应密码的表，如执行应用软件程序安全功能？还是，我们该创建不同级别的用户表（如 scott），即为每一个应用软件程序用户创建一个相应的数据库用户并保留数据库的安全？

答：数据安全不能用别的东西来代替。安全程序的应用可以不用直接连接到数据库。此外，拥有特别权限的应用软件用户可以打开一条通道轻松连接到数据库。

我会为每个用户建立数据库帐号。但事实上，数据库与个别用户连接将会消除有用的数据库连接。另外，每次一名用户登录应用软件时，你都要建立一个新数据库联系。

我会研究 Proxy Authentication 原理，因为它准确地说明了这种情况。配置数据库末端并不困难，但是这一应用软件有不同的连接方式。Oracle9i 文件 Proxy Authentication 可以从这里下载。
http://download-west.oracle.com/docs/cd/B10501_01/server.920/a96521/users.htm#17433

作者：Dan Norries 译者： April 来源：TT 中国

如何确保用户名和密码安全？

问：您能告诉我在 Windows 平台上安全管理用户名和密码的简单方法么？我们用它们去连接 Oracle 服务时，不想将它们写在代码里。我还想记录是哪些用户在什么时候对数据表进行了修改，你能告诉我该怎么做吗？

答：确保密码安全的最佳方法就是不用密码。如果你的应用程序是三层结构，你可以利用用操作系统的认证方式去实现数据库登录。如果你的应用程序只有 2 层，那么你可以使用数据库账号进行登录。但这不是普遍的做法，我不知道为什么仍然有很多人忽视了他们购买的数据库产品的安全性能。

至于数据库操作语言（DML）和数据库表格之间的切换，那是由审查系统来记录的。如果你想记录更换模式（如 ALTER），你可以在数据库 ALTER 语句利用系统事件触发器创建你自己的审查系统，当然如果你愿意也可以阻止。

作者：Dan Norries 译者：April 来源：TT 中国

锁定 SQL*Plus 命令安全

你的数据安全吗？用户能用无效用户名和密码从 SQL*Plus 命令访问数据库，并且从应用程序中访问或编辑危险数据，而这些数据在应用程序中是访问不到的。问题是，你如何不让用户通过 SQL*Plus 连接到数据？学会怎样在用户的 PRODUCT_USER_PROFILE (PUP)表中插入限定条件。

用户通过登录 IDS 能够访问应用程序。如果用户使用其用户名和密码无法登录数据库，那么他还能用用户名和密码通过 SQL*Plus 命令登录，这样用户还能够访问和编辑危险数据。问题是你怎么阻止这些用户通过 SQL*Plus 命令访问数据？

一种办法就是设置密码，授予用户在应用程序中的权限，这样一来用户就不能通过 SQL*Plus 命令登录。另一种方法就是撤销从网上访问 SQL*Plus 的权限。保证操作程序和网络的安全才能使用户访问不到数据，而且还避免了对所有用户安装访问数据库的程序。这么麻烦的任务和选择对象能可使用下面的功能完成：

产品层安全

用户在基于 SQL*Plus 产品使用上受到限制。通过在用户的 PRODUCT_USER_PROFILE (PUP)表里插入限定条件就能使他们在基于 SQL*Plus 产品使用上受限。当用户登录并用到那些记录限定条件时，SQL*Plus 命令就会在 PUP 表里读取限定条件。

如果用户用 SYSDBA 或 SYSOPER 登录时，SQL*Plus 就不会读取 PUP 表，因此也就不会涉及到限制条件。

创建

在 SYSTEM 里自动创建 SQLPLUS_PRODUCT_PROFILE 表。

PRODUCT_USER_PROFILE 和 PRODUCT_PROFILE 都能够访问这个表，以前就有这些表，但是现在变成了相近的 SQLPLUS_PRODUCT_PROFILE。PRODUCT_PRIVS 同样也存在。

为了创建这个表，需要运行 pupbld.sql 脚本。通常，这个脚本在 $ORACLE_HOME/sqlplus/admin 路径中运行，具体的位置由系统决定。

PUP表

以下是 PRODUCT_USER_PROFILE 表的主要内容：

PRODUCT——产品名称，SQL*Plus。

USERID——大写的用户名。

ATTRIBUTE——大写的不能用的指令。

CHAR_VALUE——也就是"DISABLED"。为限制这些对象，应该给这些列取一个对象名。

SYSTEM 保留了这一表的权限，所有用户有在该表上 SELECT 权限。避免其他用户连接到这个表里的数据库操纵语言。

可阻止的命令

这个功能能阻止 SQL, PL/SQL 以及 SQL*PLUS 命令运行。使用这一功能还能阻止运行下列命令：

SQL: ALTER, AUDIT, ANALYZE, CREATE, DELETE, DROP, INSERT, LOCK, NOAUDIT, RENAME,

SELECT, UPDATE, VALIDATE, TRUNCATE, GRANT, REVOKE, SET ROLE, SET TRANSACTION

PL/SQL:DECLARE, BEGIN

SQL*PLUS: COPY, HOST, SET, EDIT, PASSWORD, SPOOL, EXECUTE, QUIT, START, EXIT, RUN, GET, SAVE

示例：

```
insert into product_user_profile(product, userid, attribute, char_value)
                          values('SQL*Plus', 'APPS', 'DELETE', 'DISABLED');

insert into product_user_profile(product, userid, attribute, char_value)
                          values('SQL*Plus', 'APPS', 'INSERT', 'DISABLED');

insert into product_user_profile(product, userid, attribute, char_value)
                          values('SQL*Plus', 'APPS', 'SELECT', 'DISABLED');

insert into product_user_profile(product, userid, attribute, char_value)
```

如果需要对 SQL*Plus 阻止一个对象，就应该在下面的表中增加这个对象。

应该给 ATTRIBUTE COLUMN 付'ROLES'值，CHAR_VALUE 就是要阻止的对象。下列语句用于从内部设定对象，不包括没有设定的对象。

```
insert into product_user_profile(product, userid, attribute, char_value)
                values('SQL*Plus', 'APPS', 'ROLES', 'DBA');
```

避免使用PL/SQL

例如，用户 AMAR 没有 DELETE 权限，但他很容易通过 PL/SQL 块实现！删除 PL/SQL 块创建权限就能够避免这一点。通过这个功能还能锁定 DECLARE 和 BEGIN 语句阻止 PL/SQL 的运行。

```
insert into system.product_profile (product, userid, attribute, char_value)
                        values ('SQL*Plus', 'AMAR', 'DECLARE',
'DISABLED');

insert into system.product_profile (product, userid, attribute, char_value)
                        values ('SQL*Plus', 'AMAR', 'BEGIN', 'DISABLED');
```

局限性：

很容易就能看出这一功能有一些局限性：

1、它只适用于 SQL*Plus 命令！对其他可以代替 SQL*Plus 命令的工具不管用。Oracle 为数据层安全提供了 FGAC 机制，这一机制能够限制使用其他工具。

2、PUP 表适用于本地数据库。使用数据库链接会使 PUP 表里的限制条件也适用于其他远程数据库。

总结：

我用安全性能封闭应用程序用户权限，不许他们通过 SQL*Plus 命令访问数据库。然而这一功能有它自身的优点，应该注意防止使用错误的数据。正如我的 DDL Event Security 一文中提到的，你可使用的一些类似的功能。它们不是通过设置密码而建立的基本安全，也不是通过设置对象和权限而建立的安全。请按照 setup 要求启用上面的功能。

作者：Amar Kumar Padhi 译者：April 来源：TT 中国

用 as sysdba 登录发现安全漏洞该怎么办？

问：当你用"as sysdba"登录时，为什么 SQL*Plus 允许你访问数据库？我怎样才能够保证 Oracle 服务器安全？您能解决这个问题吗？

答：为了证实以上情况，我执行了如下操作：

SQL> connect peasland as sysdba

输入密码：连接。

PEASLAND 用户没有被授予 SYSDBA 权限。所以从表面上看来，这种情况是绝对不会发生的。但是我首先就运行了 SQL*Plus 指令，而且我的数据库在 Unix 服务器上运行。我是以'dba'的身份注册的，但如果我是 Windows 服务器用户，我就需要以'ora_dba'身份注册。在数据库服务器上注册的任何人并且是'ora_dba'群中的一员都能用登录代号连接数据库 SYSDBA。如果这个群中的用户没有登录权限，那就说明存在安全漏洞。只有 OS 帐户管理员才可能出现上述情况，一般的用户是不会出现的。如果普通用户想运行上面的指令，那么一定会有错误信息提醒他们权限不够。这一权限只属于直接登录数据库服务器的用户。如果有人从其他工作站点连接数据库，那么这一操作就会无效。

作者：Brian peasland 译者：April 来源：TT 中国

数据库表应该加密吗？

问：我的一个表中储存了用户名和密码，我想知道是否应该给表的口令加密？

答：不要加密，变换这些密码。你也不必知道密码。如果忘记了口令，用户只需要重置一个新密码。这和*nix hosts 长期存储密码实际上是一样的原理。

Oracle 有一个 builtin 函数：DBMS_UTILITY.GET_HASH_VALUE。然而使用这一函数时，你必须将明码文件密码输入数据库服务。一旦你使用 ASO，SQL .NET 就会通过网络将数据传送到明码文件中。所以，以下就是最佳操作方法。

在数据库传输到数据库并插入、升级之前，使用应用程序中的 hash 函数变换字符。这只有在 2 层结构以上的应用服务上才能执行，或者在 2 层结构的应用程序代码里才能执行。

使用数据库函数，但通过 SQL. Net 加密（ASO 的一部份）确保. Net traffiic 安全。这一操作的缺点就是你必须购买 ASO，并且它还是一种外接式附加的 Option。

使用数据库函数，但利用 SSH 确保 SQL. Net traffic 安全 。如果你的应用程序结构是两层以上的（如有一个或多个小的应用程序服务器），你可以建立应用程序服务器和数据库服务之间的 SSH 渠道，通过这些渠道运行 SQL. Net traffic 。这一过程需要多次设置和维护才能完成，但不用购买任何其他软件（所有的平台都只需要用 SSH）。

不论你使用哪一种 Option，想要从数据库获取数据或是将数据传输到数据库，你都可以执行以下操作设置密码：

1、从用户那里获取密码
2、将密码进行散列处理
3、在数据库表格中存储散列密码

登录：

1、从用户那里得到口令
2、将密码进行散列处理
3、从数据库表中获取新密码

比较#2 和#3 中的字符串。如果他们匹配，那么密码就是正确的。

作者：Dan Norries 译者：April 来源：TT 中国

保护 INTERNAL 的密码

问：当一名非 DBA 用户想连接 internal 时，我想让服务器提示密码。我怎样做才能让密码保护 internal？更改用户吗？

答：更改用户而不是 INTERNAL，是为使用权限操作（如：关闭数据库）的每一名用户创建帐号的最佳方法。

以下是两个最主要的原因：第一，你可以更好地控制保护程序。如果愿意你还可以对每个帐户单独审计。不存在共享的“群”密码，因为如果有人离开的话这一密码就必须更改。第二，INTERNAL 在 Oracle 9i 中不能使用，所以如果你现在找到了解决方法，在将 Oracle9i 升级时你必须重新考虑。

只有创建个人帐户才能让他们才能拥有自己的密码。在 INIT.ORA.中设置 REMOTE_LOGIN_PASSWORDFILE=EXCLUSIVE，然后使用 ORADIM 创建密码文档。最后重启数据库。使用 SYSDBA 权限登陆 INTERNAL 或帐户，对这些个人帐户执行 INTERNAL-type 函数并对每一位用户发出"GRANT sysdba TO the_user;"指令。他们需使用 'as sysdba'权限登陆数据库。例如：

```
edcsns18 gast% svrmgrl

Oracle Server Manager Release 3.1.7.0.0 - Production

Copyright (c) 1997, 1999, Oracle Corporation.  All
Rights Reserved.

Oracle8i Enterprise Edition Release 8.1.7.2.0 -
Production
With the Partitioning option
JServer Release 8.1.7.2.0 - Production

SVRMGR> connect my_user as sysdba
```

以上就是全部过程。

作者：Brian peasland 译者：April 来源：TT 中国

如何隐藏 Oracle 密码

当你有两个或两个以上 Oracle 数据库时，可能要求你指出你想登陆的数据库名称。如果你使用的 SQL*Plus，你需要在同一数据库里（db1 是数据库名）说明三个登录参量（用户名、密码和 db_name），如下：

>> sqlplus scott/tiger@db1

这是一例安全风险。为了隐藏密码（只显示星号通配符），你可以输入以下指令：

>>sqlplus scott@db1

输入以上指令后按 ENTER 键，要求你输入密码。输入密码后，星号通配符(*)就代替了以前你输入的字符，这就是 Oracleto 通常处理密码的方法。事实上，你是将密码隐藏了……

我已经在 Oracle8 企业修订版 8.0.5.0.0 和 Oracle Server 7.3.4.0.0 版本中测试了这一技巧。

读者反馈：

Michael P. writes：它在 8.0.5.以上的版本上运行得很好。但是 SQL*PLUS 调用在其他脚本里运行得怎样呢？在这种情况下，我该怎样隐藏密码呢？

Hing M.：要从 OS 脚本中调用 SQL*Plus，目前我发现隐藏密码的唯一方法就是不使用密码，而是连接 internal.。很显然这一方法有局限性，因为你必须从当地服务器开始连接。这种方法出现的问题是输出和输出不接受 userid=internal。那么这种情况下有没有隐藏密码的其他窍门呢？

Phillip D：我多年都在使用 hideargs 覆盖隐藏指令密码。

要了解更多的信息，你可以查看：
http://www.usg.edu/OIIT/support/oracle/General/Unix/HideArgs.html

我经常使用 OS 配置验证递规背景，而不是将密码保留在脚本里面。

作者：是的，SQL*Plus 在脚本里和密码是和密码一起发出的指令。我一点也不知道如何在脚本里面隐藏密码。然而，一个工作区（如我们在 Unix 的环境下进行操作）就是在脚本上对相关的用户发出指令。

Denis D.：假定是 Unix 环境，使用 SQL*Plus 还可以用下面的方法：

```
sqlplus /nolog <<EOF
connect usr/pass@somewhere

rem put any commands inline:
select sysdate from dual;
@my_batch_program

quit
EOF
```

以上操作针对 Oracle8 用户。对于旧版本，可以进行如下操作：

```
sqlplus  <<EOF
usr/pass@somewhere

rem put any commands inline:
select sysdate from dual;
@my_batch_program

quit
EOF
```

以上操作适用于所有 Unix 外壳程序。

Geoff H.：我的确喜欢 Philip D 的指令，但是我却不了解 hidearg 覆盖。我认为你隐藏密码的方法可以再细分为两种：第一种，执行'ps -ef '观察这一过程，建议你的用户登录 Unix 服务器，因此为什么在 Oracle 帐户里不能简单地在外面识别用户身份呢？如果运行得不够好，而且他们希望能连接到更远的地方，那么就在 Unix 包上使用 Batch 用户帐号登录。如果你希望限制用户登录脚本的权限并想隐藏密码，为什么不允许用户登录 SUDO，登录超级用户执行脚本呢？这些方法都能运行，整个都取决于你怎样安装。希望这些建议能对你们有所帮助。

Witold I.：我有时用另一诀窍隐藏密码，即利用我设定的环境变量。在脚本中可设定环境变量选择用户。如：

```
set the_password=mypass
sqlplus user1/%the_password%@mydb
```

事实上，我常在 TKPROF 脚本里面使用这种方法，但是我用 sqlplus 也测试了，这种方法也很管用。

作者：Rochus Mission 译者：April 来源：TT 中国

如何隐藏用户密码

通过 cron 运行程序（在 Unix 包上），要求你在操作系统层运行 ps 指令时隐藏 Oracle 用户的密码。这样做有各种各样的方式，但是简单的方法就是用 init.ora 参数（os_authent_prefix），在外面对用户进行身份识别。

在 init.ora 文件中给任一字符串（如 OPS$）设定 os_authent_prefix。现在 V$参数输出结果应该是：

```
NAME                 TYPE    VALUE
-----------------    -------  ---------
os_authent_prefix    string   ops$
```

现在无论何时创建一名用户，你只需在外面输入 create user ops$<username> identified。例如，假如用户名称为 DBGUY：

```
create user ops$DBGUY identified externally;
```

你可以按如下操作：

```
$ id
uid=12997(DBGUY) gid=1(other)

$ sqlplus /

SQL*Plus: Release 8.1.5.0.0 - Production on Mon Jun 17 09:28:46 2000
(c) Copyright 1999 Oracle Corporation.  All rights reserved.
Connected to:
Oracle8i Enterprise Edition Release 8.1.5.0.0 - Production
With the Partitioning and Java options
PL/SQL Release 8.1.5.0.0 - Production

ops$dbguy ((Content component not found.)) i> show user
USER is "OPS$DBGUY"
ops$dbguy ((Content component not found.)) i>
```

读者反馈：

Geoff H.：作者可能爱用SUDO指令隐藏用户密码，防止密码在ps –ef里显示。你能掌握用户在Unix上能做什么，如同在sudoer文挡里面那么精确。真正有效的uid 和gid就是在passwd文件里将目标用户匹配（当目标用户不是最开始的用户时，参数还是被初始化了的）。Sudo要求用户通过默认用一个字/词对自己的身份进行验证（注：这是用户的密码，而不是源密码）。一旦用户身份通过验证，timestamp列就更新了，并且用户在段时间内可以使用sudo而不是密码登录（能覆盖sudoers的话为 5 分钟）。Sudo通过访问/etc/sudoers文件决定谁是通过身份验证的用户。通过给sudo发出-v flag指令，用户不用运行命令就能更新timestamp列。如果用户在 5 分钟内没有输入密码的话，密码自动弹出的提

示框也会过时（除非通过sudoer覆盖）。相关链接：http://www.courtesan.com/sudo/。希望对能你们有帮助。

作者：Database Person 译者：April 来源：TT 中国

如何使用隐藏密码登录？

问：目前，我们正在使用 Oracle 的密码功能（utlpwdmg.sql 修订版）即数据库 9i 中安全验证功能。我想进一步了解并且不允许使用一些明显但符合验证标准的密码（如#1）。很显然的方法就是将这些密码和-20002 组合并检查得到的结果是否太简单。由于这一验证功能还保留了一系列的数据，最终我创建了一个表格。您能告诉我怎样从验证功能中而不是从一般的密码中 IN ('welcome', 'blah1', 'blah2')查找这一表格吗？

答：你需要对密码验证功能重新编码，并将所有的记录保存在数据库的表格中。假设这些记录都在 BAD_WORD 这一列的 BAD_PASSWORDS 表格中，密码验证功能将显示一个 BAD_COUNT 的 NUMBER 变量。最简单的就是将与后面相同的密码添加在密码验证功能中，这样就统计出了次数，新密码也在 BAD_PASSWORDS 表格中了，具体操作为：

```
SELECT COUNT(*) INTO bad_count FROM bad_passwords
WHERE bad_word = new_password;
```

如果 COUNT 值为 0，那么新密码就没有在 BAD_PASSWORDS 中，所以 COUNT 的值至少应为 1。

```
IF bad_count > 0 THEN RAISE APPLICATION ERROR (-20002, '密码不能为一个字');
```

若想在 BAD_PASSWORDS 表格中增加一个新词，只需将其插入这一表格，并且不需要对密码验证功能重新编码。

作者：Brian peasland 译者：April 来源：TT 中国

忘记 SYS 密码时该怎么办？

问：我使用 sys 帐户登陆数据库的时候，访问遭到拒绝。我可能输入了一个错误的密码。我忘记了 sys 用户的密码，现在想改变 INITSID.ORA 文件的一些参数。但是忘记了密码，就改变不了这些参数值，因为所有的操作都需要密码，但我也不能重启 Oracle 服务。在 Oracle 运行时，我将密码文件（PWDSID.ORA）删除了，后来我利用 ORAPWD 指令重新创建了密码文件。重启 Oracle 服务，新密码好像也可以登录。请问如果重启 Windows，Oracle 实例能自动启动吗（Windows Oracle SID 服务自动启动吗）？ Oracle 能自动识别 PWDSID.ORA 文件中的新密码吗？

答：使用管理员或者 ORA_DBA 组成员的身份登陆 windows 系统。这时你可以不使用 SYS 密码登陆数据库，然后执行以下操作：

```
sqlplus /nolog
connect / as sysdba
```

一旦连接成功，你就能修改密码：

```
ALTER USER sys IDENTIFIED BY new_password;
```

这些操作完毕之后，不需要重启 Oracle。

作者：Brian peasland 译者：April 来源：TT 中国

如何做到不用密码也能登陆系统

问：我知道 DBA 已经设置了系统帐户，但是我用任意字符串当密码就能登录，有时我甚至不用密码就能登陆系统（如 SYSDBA）。是什么问题呢？我的操作系统是 WinXP，我也不是 ORA_DBA 中的一员。

答：不，没有密码你用 system as sysdba 一定不能登录。不用密码的话，你可用/ as sysdba 登录（如果 DBA 建了这个命令）。如果你用 OEM 登录，可能你在 OEM gui 中已有登录记录。

如果能随便输入文字作为密码登录，你的平台可能存在着安全隐患，我建议你在它上面建立 TAR 文档。我不知道 Oracle 目前支持什么版本的 Windows，所以你还应该验证你的数据库版本是否支持 XP。

作者：Database Person 译者：April 来源：TT 中国

在不输入密码的情况下运行 SQL*Plus 脚本的最佳方法

问：在不输入密码的情况下运行 SQL*Plus 脚本的最佳方法是什么？例如：SQL*Plus userid/pw @script

答：我推断你之所以问这个问题是因为你不想在 Unix 系统上操作 ps -ef|grep sqlplus 时别人看见你的密码。如果在操作过程中别人能看见你的密码，那就存在安全漏洞。有很多种处理这个问题的方法：

其中一种方法就是用连接脚本。创建一个 connect.sql 脚本，它只有一行：
connect userid/password

现在使你所有的脚本都以上面的脚本开头。为了不使 SQL*Plus 脚本提示你输入密码，开始时可再"/nolog 选项中进行如下操作：

sqlplus /nolog @my_script

另一种方法是重新设置可执行密码，具体操作如下：

sqlplus username @my_script << password_file

"password_file"文件中仅有一个字/词，这个单独的字/词就是用户密码。SQL*Plus 脚本正常提示输入密码时，"password_file"文件中的密码就可以用。

无论是哪一种方法，如果你不采取措施保护任何含有密码的文件，那么你的密码一定会被泄露。所以一定要保证在 OS 文件允许的情况下只有 Oracle 用户或 DBA 才能使用这样的文件。

作者：Dan Norries 译者：April 来源：TT 中国