Oracle数据库加密技术分析

徐江峰[1] 庄海燕[1] 杨 有[2]

(郑州大学信息工程学院 郑州 450052)[1] (重庆师范大学数学与计算机科学学院 重庆 400047)[2]

摘 要 网络技术的快速发展,数据库的广泛应用,使得数据库安全问题变得越来越重要。本文对Oracle数据库系统的加密功能进行了分析和研究,并结合实例说明了如何利用Oracle系统软件包实现数据库数据的加密与解密。分析结果表明,利用Oracle 10g的软件包DBMS_CRYPTO可以实现随机密钥的产生,但系统缺少密钥存储与管理的安全方法,因而仅依靠数据库系统自身提供的加密手段对数据行进行保护是不够的。

关键词 Oracle数据库,加密技术,密钥,安全性

The Analysis of Encryption Technology of Oracle Database

XU Jiang-Feng[1] ZHUANG Hai-Yan[1] YANG You[2]

(School of Information and Engineering, Zhengzhou University, Zhengzhou 450052)[1]

(College of Mathematics and Computer Science, Chongqing Normal University, Chongqing 400047)[2]

Abstract With the rapidly development of network and database application, the security of DBMS has becoming more and more important. In this paper, the encryption technology of Oracle database is discussed, and the method of encryption and decryption database using software package DBMS_CRYPTO are presented. Theoretical analysis demonstrates that the random key can be produced by Oracle 10g database, but the security strategy of data storage is lacking. For improving the security of Oracle database encryption, the external means should be used.

Keywords Oracle database, Encryption technology, Key, Security

1 引言

随着网络技术的不断发展及信息处理的不断增多,巨量级数据扑面而来。无论对于银行、电力、统计局、保险公司,还是其他一些数字、数据密集型企业来说,数据的重要性日益凸现,从而使数据安全问题变得也非常显著。针对安全问题,目前多数的网络安全策略都是处于一种被动状态,主要用于保护网络的边缘安全,如:防火墙、入侵检测及VPN等,而忽视了存在于网络服务器内的机密数据的安全隐患。外部攻击者只要突破网络就可以得到服务器中的所有机密数据,内部具有一定权限的雇员也可以直接获得或破坏这些数据。

实际上,在目前网络安全技术下,任何网络都是有漏洞的,在数据不加密的情况下,任何系统数据库数据对内部和外部黑客都是公开的,黑客可以在几分钟内用移动硬盘等工具盗窃全部服务器的文件和数据库数据。因此,要想提高数据库数据及服务器文件的安全性,数据库加密及文件加密策略应是最佳选择,它们也是网络安全的最终解决方案。

近年来,数据库加密技术的研究不但引起了各国学者的注意,也受到了数据库生产厂商的重视。在最新的Oracal数据库系统中,通过其提供的软件包已经可以实现数据库数据的加密与解密。本文将对Oracle数据库系统的加密功能进行分析和研究,并结合实例说明如何利用Oracle系统软件包实现数据库数据的加密与解密,最后对其存在的问题及解决方法提出意见及建议。

2 数据库加密的一般策略

近年来,数据库加密技术研究取得一定的研究成果,提出了多种数据库加密策略,但无论何种策略,都属于下面两种之一:(1) DBMS内核层加密;(2) DBMS外围层加密。这两种策略的特性及优缺点如表1所示。

表1 DBMS内外层加密策略比较

项目	DBMS内核层加密	DBMS外围层加密
对应用程序影响	安全需求改动时,应用程序升级简单,不必改动.	安全需求改动,需要修改应用程序.
对数据库性能影响	额外的加密处理可能会造成数据库运行性能的降低.	加密过程与数据库服务器分离.
密钥存储和管理	若采用数据库表存储方式,则密钥和密文存储未分离,有一定安全隐患;若分离存储密钥和密文,则需要其它软硬件资源支持.	密钥和密文分离,可以分离管理员的角色,对数据访问有更多控制,可以通过加密服务器提供更强的认证功能,因而相对更加安全.
加密算法	加密算法选择受限.	加密算法选择灵活多样,但通讯负载大,需要更多服务器.

从表中可以看出,两种加密方法各有优缺点,在DBMS内核层实现数据加密,对应用程序的影响较小,可以适应更多的应用需求,但加密算法选择受所采用的数据库系统限制,并且会降低数据库操作性能。而在DBMS外围层加密,既可采用通用的加密算法,也可采用自己开发的加密算法,算法选择灵活多样,加密安全度高,并且加/解密过程与数据库服务器

徐江峰 博士,副教授,研究领域:网络安全及混沌加密通信。**庄海燕** 研究领域:信息安全及数据库加密。**杨 有** 博士研究生,讲师,研究领域:嵌入式微小型信息处理系统。

分离，不影响数据库系统的效率，但通讯负载大。在实际应用中采用何种加密策略，需要权衡利弊，根据需求来确定。

3 Oracle 数据库加密策略分析

3.1 加密软件包及功能

与 Sysbase、SqlServer 等数据库管理系统一样，Oracle 数据库管理系统的主要功能仍然是实现对数据的管理及操作，数据加密只是为了提高存储数据的安全性而附加的功能。Oracle 数据库加密功能的实现由数据库平台提供的软件包来支持。DBMS_OBFUSCATION_TOOLKIT(DOTK)是 Oracle 9i 以及更低版本的数据库中唯一可用的加密方法，而 Oracle Database 10g 中，用户可以通过一种更加完善的内置软件包 DBMS_CRYPTO 来执行数据加密，它对早期版的 DBMS_OBFUSCATION_TOOLKIT 在功能和函数上进行了加强。两个软件包加密功能比较见表 2。

表 2 DBMS_CRYPTO 和 DBMS_OBFUSCATION_TOOLKIT 加密软件包提供的加密功能

项目	DBMS_CRYPTO	DBMS_OBFUSCATION_TOOLKIT
加密算法	DES, 3DES, AES, RC4, 3DES_2KEY	DES, 3DES
填充模式	PKCS5, zeroes	none supported
块加密链接模式	CBC, CFB, ECB, OFB	CBC
加密 hash 算法	MD5, SHA-1, MD4	MD5
Keyed hash algorithms	HMAC_MD5, HMAC_SH1	none supported
伪随机数生成器	RAW, NUMBER, BINARY_INTEGER	RAW, VARCHAR2
数据类型	RAW, CLOB, BLOB	RAW, VARCHAR2

3.2 加密算法

从表中可以看出，在软件包 DBMS_OBFUSCATION_TOOLKIT 中，加密算法仅支持 DES 以及三倍 DES(DES3)，而在软件包 DBMS_CRYPTO 中不仅支持上述算法，而且还支持 256 位的高级加密标准 AES 等，使用户有了更多的选择，也使加密数据可以达到更高的安全强度。两个软件包中支持的算法，无论哪种算法都属于分组密码算法，其分组大小是固定的。如果加密信息长度不在分组边界，就需要附加数据填充，使它充满整个分组。填充的方法有固定值填充、随机填充、PKCS 填充。固定值填充，最常用的填充值是 0x00/0x80 加 0、1 字符串，不常用的填充值是 0xff；随机填充采用随机位来填充；PKCS(Public Key Cryptography Standard，公开密钥密码标准)填充使用信息长度编码，因此不需要预先知道信息长度就可以去掉填充。Oracle 10g 中，可以根据需要用不同的参数来确定所选择的填充类型，而 Oracle 9i 并不支持数据填充。在运算模式上，Oracle 10g 中支持 ECB、CBC、CFB 及 OFB，而 Oracle 9i 中仅支持 CBC。

3.3 密钥存储和管理

在数据加密中，密钥的存储和管理是非常重要的，它直接影响到数据加密的安全性。但是，在数据库管理系统内核层加密策略中，并没有提供密钥存储的方法，这也是在以 Oralce 提供的安全包为基础制定加密策略时最难解决的部分。在制定密钥的存储和管理方案时，要确保以下两点：(1)密钥存储足够可靠，以确保能够保护数据；(2)要保证合法用户且只有合法用户可以获取密钥。在数据库加密中，密钥通常存在数据库中、文件系统中或用户自身的存储设备中。Oracle 数据库加密中，密钥仍然是以数据表形式存放在数据库中。如果密钥以明文形式存放在数据库中，那么攻击者只要进入数据库系统中，他就很容易找到破解密文的密钥。如果密钥以密文形式存放在数据库中，那么加密密钥的密钥如何存放就成了需要解决的新问题。常用的解决方法是采用多级密钥存储管理，把用户密钥与数据密钥结合使用，提高数据库加密的安全性。

3.4 加密实例

下面用一个例子介绍如何利用 Oracle 软件包 DBMS_CRYPTO 提供的功能，保护在线联机事务数据库。

1)数据表结构　本例中用到两个数据表，其中表 custom_info_enc 存放客户基本信息，Custom_id 是客户编号，Id_no_enc 是加密后的客户身份证号。而表 custom_info_keys 中，Custom_id 存放的仍然是客户编号，它与表 custom_info_enc 中的客户编号是一致的，key 中存放的是加密相应客户身份证号码的密钥。其表结构如下：

表 custom_info_enc 结构

Name	Null	Type	Desc
Custom_id	NOT NULL	Number	客户编号
Id_no_enc		RAW(200)	身份证号码密文

表 custom_info_keys 结构

Name	Null	Type	Desc
Custom_id	NOT NULL	Number	客户编号
Key	NOT NULL	RAW(200)	用于加密身份证号码的密钥

2)数据查询及更新　为了查询客户信息，首先需要生成视图 vw_custom_info，生成语句为：

```
create or replace view vm_custom_info as
  select e. custom_id as m. custom_id,
       cast(get_dec_val(e. id_no_enc, k. key) as varchar2(20))
       as Id_no
  from custom_info_enc e,
       custom_info_keys k
  where k. custom_id = e. custom_id;
```

在建立视图后，可以把其权限授予其他用户，并建立一个视图的同义词。这种情况下，用户只能从视图中查找数据。

当需要插入或修改客户数据时，则可以通过一个触发器来更新表 cusom_info_enc 的信息。对表 custom_info_enc 操作的触发器定义为：

```
create or replace trigger io_vm_custom_info
  instead of insert or update on vw_ustom_info for each row
  declare l_key raw(200);
  begin
    if (inserting) then
       l_key:= dbms_crypto. randombytes(128);
       insert into custom_info_enc(custom_id, id_no_enc)
       values(:new. custom_id, get_enc_val(:new. id_no, l_key
       ));
       insert into custom_info_keys(custom_id, key)
       values (:new. custom_id, l_key);
    else
       select key into l_key
       from custom_info_keys
       where custom_id = :new. custom_id;
       update custom_info_enc
       set id_no_enc =get_enc_val (:new. id_no,l_key)
       where custom_id = :new. custom_id;
```

```
    end if;
```

3)加密及解密函数　在上述操作中，函数 get_enc_val 及 get_dec_val 的功能分别是对数据进行加密及解密。在函数中，利用了 Oralce 软件包 DBMS_CRYPTO 中支持的加密函数、运算模式及填充模式。

(1)加密函数

```
create or replace function get_enc_val
( p_in in varchar2, /*定义待加密数据
  p_key in raw /*定义加密密钥变量
)
    return raw is
      l_enc_val raw (2000); /*定义密文变量
      l_mod number := dbms_crypto.ENCRYPT_AES128 /*选
                    择由 oracle 软件包 DBMS_CRYPTO 提供的
                    AES128 位的加密算法
                    + dbms_crypto.CHAIN_CBC /*分组模式
                    为密码链接
                    + dbms_crypto.PAD_PKCS5; /*填充方式
                    为 PKCS
    begin
      l_enc_val := dbms_crypto.encrypt
      (UTL_I18N.STRING_TO_RAW(p_in,'AL32UTF8'),
      l_mod,p_key);
      /*利用 DBMS_CRYPTO 提供的加密算法，根据上面生成的
      随机数(密钥)加密数据，l_enc_val 为加密后的数据
      return l_enc_val;
end;
```

(2)解密函数

```
create or replace function get_dec_val
( p_in in raw, /*定义密文变量
  p_key in raw /*定义加密和填充类型
)
return varchar2 is
l_ret varchar2 (2000);
l_dec_val raw(2000);
l_mod number := dbms_crypto.ENCRYPT_AES128
                + dbms_crypto.CHAIN_CBC
                + dbms_crypto.PAD_PKCS5;
begin
   l_dec_val := dbms_crypto.decrypt
      ( p_in,
        l_mod,
        p_key
      );
   l_ret := UTL_I18N.RAW_TO_CHAR
          (l_dec_val, 'AL32UTF8');
          return l_ret;
end;
```

在加密与解密函数中，分别用到了函数 UTL_I18N.STRING_TO_RAW(p_in, 'AL32UTF8')及 UTL_I18N.RAW_TO_CHAR(l_enc_val, 'AL32UTF8')，其作用是进行字符类型与 RAW 类型的转换。由于 Oracle 10g 中加密函数 Encrypt()仅支持 RAW 类型的变量，因此在数据加密时需要先把其它类型的数据转换成 RAW 类型，而解密后则需要相反的操作。

结束语　随着网络技术的不断发展，信息处理的不断加大，数据安全问题变得越来越严重。作为数据存储及管理核心的数据库系统，其安全性也是整个网络系统的核心。加密作为保护数据安全的重要手段，不仅对传统的数据信息有用，对数据库数据同样适用。目前实现数据库数据加密的主要方法是 DBMS 内核层加密和 DBMS 外围层加密，两者各有优缺点。实现 DBMS 内核层加密需要数据库厂商的支持，Orcale 10g 数据库系统就可以实现这一功能。本文首先对 Oracle 数据库系统可以实现的加密功能进行了分析，而后给出了一个实现实例，说明了如何利用 Oracle 10g 提供的软件包实现数据库数据的加密与解密。任何加密策略的安全性都依赖于密钥的安全，但是 Oracle 数据库系统提供的加密方案中，并没有给出密钥存储与管理的安全方法。要想提高数据库加密系统的安全性，仅依靠数据库系统自身提供的加密手段显然是不够的，还需要辅助其他的外部手段来实现密钥的安全存储与管理。

参 考 文 献

1　Arup N. Encrypt your data assets[R]. http://www.oracle.com/

2　PL/SQL packages and types reference[R]. Oracle database 10g release 1(10.1). http://www.comp.hkbu.edu.hk/docs/

3　Hwang M S, Yang W P. Multilevel secure database encryption with subkeys[J]. Data & Knowledge Engineering, 1997, 22: 117～131

4　姚炎炎，陈怀义，等. 密码体制与分布式 WEB 数据库的安全设计[J]. 计算机科学，2001，28：6～9

5　余祥宣，倪晓俊. 加密数据库系统中的密钥管理[J]. 华中理工大学学报，1995，23(7)：52～55

6　潘瑞芳，陈专红. 谈数据库安全控制策略[J]. 计算机与现代化，2001，6：140～145

7　王宏杰. 应用数据加密技术对数据库加密的探讨[J]. 天津职业技术师范学院学报，2003，13(3)：34～36

8　朱鲁华，陈荣良. 数据库加密系统的设计与实现[J]. 计算机工程，2002，8：61～63

9　王洪，吕述望，刘振华. 基于关系 DBMS 的一种数据库加密实现方法[J]. 计算机系统应用，2005，3：43～46

10　王元珍，冯超. 数据库加密系统的研究与实现[J]. 计算机工程与应用，2005，8：170～172

(上接第 129 页)

GGSP(e-Gov Grid System Protocol)和政务网格计算协议 GCP(e-Gov Grid Computing Protocol)等，这些协议的性能将直接影响到政务网格的成败。

结论　尽管政务网格目前存在诸多问题，但应该看到网格技术对电子政务建设起很大提升作用。政务部门在多年的信息化工作中都积累了大量的数据信息资源和网络设备，这些异构和分布的资源和设备不能共享，不能形成“合力”，利用率很低，人们使用极不方便。网格技术可以方便地解决这些问题，把网络中各种资源组织起来，形成“合力”，把高性能的网格技术送到用户的桌面上，消除信息孤岛，方便透明地实现信息共享，最终形成电子政务的协同工作能力。

参 考 文 献

1　周宏仁，唐铁汉. 电子政务全球透视与我国电子政务的发展. 电子政务的理论与实践—电子政务知识读本. 北京：国家行政学院出版社，2002

2　Foster I, Kesselman C, Tuecke S. The anatomy of the grid: Enabling Scalable Virtual Organizations. Intl. Journal Supercomputer Applications, 2001, 15(3): 200～222

3　Foster I, Kesselman C. The grid: Blueprint for a New Computing Infrastructure. Morgan Kaufmann Publishers, 1998

4　都志辉，陈渝，刘鹏. 网格技术. 北京：清华大学出版社，2002

5　何淑贞，王晓海. 主宰互联网未来发展的信息网格技术. 中国数据通讯，2002(10)：47～50

6　陈述彭，陈秋晓，周成虎. 网格地图与网格技术. 测绘科学，2002，27(4)：1～6

7　IDG 消息. 新加坡启动永久性网格技术计划. 计算机世界，2003-5-5，(16)：A16