安全技术经典译丛

The Oracle Hacker's Handbook
Hacking and Defending Oracle

Oracle数据库攻防之道

(英) David Litchfield 著
杨 华 译

WILEY
清华大学出版社

Oracle 数据库攻防之道

(英) David Litchfield 著

杨 华 译

清华大学出版社

北 京

David Litchfield
The Oracle Hacker's Handbook: Hacking and Defending Oracle
EISBN：978-0-470-08022-1

北京市版权局著作权合同登记号　图字：01-2007-1430

图书在版编目(CIP)数据

Oracle 数据库攻防之道/(英)里奇费尔德(Litchfield, D.)著；杨华　译. —北京：清华大学出版社，2007.11
书名原文：The Oracle Hacker's Handbook：Hacking and Defending Oracle
ISBN 978-7-302-16429-6
Ⅰ. O…　Ⅱ. ①里…　②杨…　Ⅲ. 关系数据库—数据库管理系统，Oracle—安全技术
Ⅳ. TP311.138　TP309

中国版本图书馆 CIP 数据核字(2007)第 170717 号

责任编辑：王　军　李　阳
装帧设计：孔祥丰
责任校对：成凤进
责任印制：李红英
出版发行：清华大学出版社　　**地　　址**：北京清华大学学研大厦 A 座
http://www.tup.com.cn　**邮　　编**：100084
c—service@tup.tsinghua.edu.cn
社 总 机：010-62770175　**邮购热线**：010-62786544
投稿咨询：010-62772015　**客户服务**：010-62776969
印 刷 者：清华大学印刷厂
装 订 者：三河市新茂装订有限公司
经　　销：全国新华书店
开　　本：160×230　**印　张**：13.75　**字　数**：253 千字
版　　次：2007 年 11 月第 1 版　　**印　次**：2007 年 11 月第1 次印刷
印　　数：1～4000
定　　价：29.80 元

前　言

Oracle 获得安全权限非常重要，不过迄今为止，Oracle 在这方面的表现却比较差。与其他的数据库服务器产品相比，Oracle RDBMS(关系型数据库管理系统，Relational Database Management System)存在的安全漏洞更加严重。这里的严重是指远程攻击者不需要用户 ID 和密码就可以利用那些缺陷，而且可以完全控制数据库服务器。IBM 的 DB2 有 1 个漏洞；Informix 有 2 个漏洞；Microsoft SQL Server 有 2 个漏洞；Oracle 有 9 个漏洞。Oracle 存在的漏洞比其他数据库所有的安全漏洞加起来还要多。另一方面 Oracle 的隐患也比其他数据库多，这些隐患需要用到用户 ID 和密码，攻击者利用它们可以完全控制数据库。这些事实与号称自己产品“无懈可击”的 Oracle 市场宣传大相径庭。当 Oracle 的执行官说“我们已经解决了安全问题，这正是我们所擅长的...”，您可能会想他们到底在说什么。迄今为止，问题仍然没有解决，而且为绝大多数的政府网络开发安装软件的企业永远都不能满意。这也正是 Oracle 为什么要把它做好的原因——国家安全正受到威胁。

Oracle 对于安全的认识在很大程度上是依据美国国防部的保证标准形成的。这就是为什么 Oracle 可以宣称“他们是安全的”。这或许在 15 年前是对的，不过从那时起安全的范畴已经完全改变了。下面进一步解释这个问题。Oracle RDBMS 是按照通用标准 EAL4(保证等级 4)进行评估的，这是很难做到的。虽然 Oracle 先前的几个版本通过了 EAL4 的评估，但是在身份认证机制中仍存在缓冲区溢出漏洞。若传给服务器一个长的用户名，基于堆栈的缓冲区就会溢出，程序控制信息会被覆盖，攻击者会获得完整的控制权。这些到底是如何实现的以及如何不被人发现的呢？问题的答案是因为安全“标准”的意义与现实安全的意义有很大的区别。当然，标准的作用很重要，但是它们并不是安全所追求的全部目标和最终目标，Oracle

应该吸取这个教训。标准意味着规则，但是攻击者却并不遵守这个规则。

或许 Oracle 已经意识到了这一点。他们已经动摇并改善了自己的编码标准，而且投资了许多工具来帮助自己开发更加安全的代码。有证据表明，它在安全方面的表现越来越好。Oracle 10g Release 2 对 10g Release 1 做了巨大的改进。虽然在 10g Release 2 中仍然存在安全漏洞，但是数量比 10g Release 1 少了很多。Oracle 还改进了自己的安全补丁发布机制。在每个季度，Oracle 会发布一个关键补丁更新(Critical Patch Update，CPU)，而且至 2006 年 7 月，每个 CPU 都会因为修复失败、遗漏故障或其他问题而被多次重新发布。2006 年 7 月的 CPU 与此不同，它只发布了一次——希望这会是一种新趋势的开始。

考虑到情况正在改善，即 Oracle 正在一步步地接近“安全”的理想境界——这是否意味着安全的产品完全匹配市场的要求？为了回答这个问题，对每一个出售者而言，最关键的是要观察他们如何回应安全研究人员。在 2006 年夏天的 Blackhat Security Briefing 中，笔者所在的小组就是讨论揭露安全隐患的。来自 Gartner 公司的小组主持人 Paul Proctor 很有远见地评价“Microsoft 正在接纳阶段。Cisco 正在缓慢地从愤怒阶段向接纳阶段过渡。另一方面，Oracle 才刚刚走出否定的阶段，并开始走进愤怒阶段。”

在笔者看来，这个评价非常准确。就像几年前的 Microsoft，也就是在 Scott Culp 发布自己的论文“*Information Anarchy*”的时候，Oracle 也曾经评论过安全研究人员。Oracle 的首席安全官 Mary-Ann Davidson 撰写了自己的文章：“*When Security Researchers Become the Problem*”。Mary-Ann 的文章与 Scott 的论文的不同之处在于 Scott 的论文不需要说明，因为这篇论文是在信息混乱出现很久之后而且不再继续揭露很多漏洞的时候发表的。这是说服安全研究人员和供应商一起合作的一次尝试。Mary-Ann 的文章几年之后仍然不能深入人心，这是因为她所藐视的安全研究人员已经在与 Oracle 一起尽力改善他们的产品了。Oracle 未能看到自己与安全研究人员们一道朝着共同的目标——更加安全的服务器而努力。Mary-Ann 文章的某些部分讨论了公开和私下制造威胁的安全研究人员，例如“在接下来的三周内修复它，因为要在 Black Hat 上提交论文。”但是，Oracle 应该理解安全研究人员没有义务通知他们自己要提交论文，如果他们确实告知了 Oracle，Oracle 应该对此举表示感谢。这样的举措是一种礼貌。把它称之为“含蓄的威胁”是毫无诚意的，而且还会疏远有助于自己保护产品安全的人们。Oracle 度过自己的愤怒阶段并尽快进入接纳阶段，这应该对所有的人都有好处。

对 Oracle 的评论已经足够了，至少就目前阶段而言已经足够了。下面看一看为什么要写本书来详细说明他们的 RDBMS 中存在的漏洞以及如何利用这些隐患。简而言之，这是因为它是一个非常流行的数据库服务器，它是黑客、犯罪组织、商业间谍和外国间谍的主要攻击目标。因此，数据库和安全管理员应该能够知道自己的系统是如何被攻击的，以及哪里存在漏洞。这样在设计防御策略和减轻攻击的时候他们才能处于有利的地位。

本书就是要提供这些信息。的确——这样的一本书在本质上是荒谬的：试图有助于防御，但是它不但把信息告诉了防御者而且也告诉了攻击者。但是根据笔者的经验，绝大多数攻击者都已经知道了这些信息，而防御者却并不知道。即便是现在，假设所有的证据恰恰相反，前面已经介绍过 Oracle 的“专家们”声称 Oracle 是安全的，引用的证据是 Oracle“常常安装在防火墙之下”而且“运行在 Unix 平台上”。坦白地说，这些“原因”与 Oracle 是否安全无关。进入运行在 Linux 或 Solaris 平台上的 Oracle 服务器与进入运行在 Windows 平台上的 Oracle 服务器一样容易。一旦在防火墙上打开了一个洞，就可以通过它令自己的业务逻辑和 Web 应用程序连接到数据库服务器，这样防火墙就会变得无关紧要——SQL 注入是一个主要问题。

此外，说 Oracle 常常安装在防火墙之下也是无稽之谈。根据笔者在 2005 年 12 月完成并于 2006 年 6 月发布的“数据库曝光调查”，相对于 210 000 个可以访问的 Microsoft SQL Server，网络上大约有 140 000 个 Oracle 数据库服务器可供访问。假设许多 SQL Server 都是 MSDE，那么考虑一下 Oracle Express 会对这个数字产生什么样的影响。Oracle Express 是在这个调查完成之后发布的。但是回到问题的核心部分，Oracle 领域里的那些人却没有充分意识到自己的服务器存在危机。Oracle 已经同意每 3 个月发布一次关键补丁更新，至少到 2007 年 7 月(写这本书期间)，这意味着在这期间 Oracle 数据库服务器处于极度危险的状态中。这确实发人深省。这也正是原因所在。如果要对自己的系统安全负责，在知道了它们确实不安全之后，需要知道它们不安全的原因——只有这样才能采取措施，防止自己的系统受到威胁。

笔者希望本书能够对读者有所帮助并能增长见识，同时也希望能够激起读者阅读的兴趣。笔者一直希望能够回答这些有关数据库安全的问题，所以请尽管提问。

祝好，

David Litchfield
david@databasesecurity.com

书中的代码示例

若想下载书中相关的代码示例，请访问本书的 Web 站点 www.wiley.com/ go/ohh 或合作站点 www.tupwk.com.cn/downpage。

Oracle 与安全性

1997 年 6 月，Larry Ellison 和 Robert Miner 成立了公司 Software Development Labs。他们两个都曾经在 Ampex 工作过。Robert 曾经是 Larry 的上司。他们两个都受到了 Edgar Codd 工作的启发。Codd 曾经做过 IBM 的研究员并提出了关系数据库的思想。在 1970 年他发表了论文 *"RelationalModel of Data for Large Shared Data Banks"*。IBM 很久以后才看到 Codd 思想的前景，而 Larry 和 Robert 却不是。在 1979 年，他们把自己的公司更名为 Relational Software, Inc.,而且没过多久就又改名为 Oracle。Oracle 曾经是 Larry 和 Robert 在 Ampex 工作时所参与的一个 CIA 项目的代码名。其实，Oracle 公司最初几年的客户就是 CIA 和 NSA。知道了这些，就可以猜想到安全性应该是 Oracle 的首要日程。

在 1999 年，Oracle 开始受到了安全研究团体的关注。根据 SecurityFocus.com，同年 4 月 29 日公开了 Oracle 的第 1 个安全故障：Dan Sugalski 公开 oratclsh 程序是 setuid root 而且可以被*nix 组的"其他人"执行。这意味着任何人都能够以 root 用户的身份运行 TCL 脚本。不久之后，作者和 Georgi Guninski 公开了与 Oracle Web Listener 有关的许多安全隐患，以及与默认许可有关的其他问题。在 2000 年，Oracle 开始发布自己的警报，不过这些都是基于 ad-hoc 发布的，而且常常是在某个缺陷被公布之后的几个月内发布的。

图 0-1 给出了"早"几年所报告的故障数量。可以看到，故障的数量呈指数级增长，而且实际上这也是图中没有给出 2003 年和之后故障数目的唯一原因——可以说故障的数目会超出最高限度。绝大多数的故障都和 RDBMS 中的缓冲区溢出或应用程序服务器有关。Oracle 的一个严重缺陷就是 PL/SQL 注入，这个问题直到 2003 年 11 月 5 日 Oracle 发布了第 61 号警报的时候才被公开——这个警报涉及笔者发现的许多注入隐患。这引发人们发现了大量的 PL/SQL 注入隐患，而且即使是在今天，Oracle 所修复的绝大多数问题都是注入问题。在本书的后面会看到，PL/SQL 中的隐

患是 RDBMS 的弊端，而且它们一旦被利用就会允许攻击者完全控制数据库服务器。

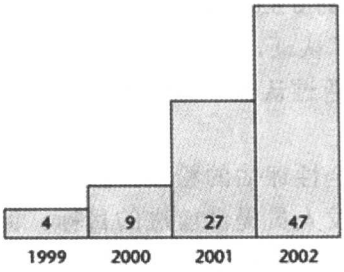


图 0-1　1999~2002 年之间故障数量稳步增长

“不可攻破”的市场宣传

伴随着 Oracle 9i 的发布，Oracle 在 2001 年 12 月展开了新的市场宣传。他们宣称自己的产品是“不可攻破的”，而且没有人可以“破坏它”或“强行进入它”。对许多安全行业的人来说，这就是所谓的短暂瞬间：每个人都记得自己第一次听到 Oracle 的发布内容时自己在做什么。笔者当然记得自己当时在做什么，而且毫无疑问，还记得此后做过的许多其他事情——下载这个“不可攻破的”软件并看看它到底有多健壮。在发布期间，笔者和弟弟 Mark 发给 Oracle 一些报告，上面详细说明了破解和强行进入服务器的许多方法。Oracle 修复了这些隐患之后，笔者在 2002 年 2 月的 BlackHat Securtiy Briefing 上提交了一篇关于这个问题的论文。Oracle 完全被破解了。

现在，Oracle 会说自己的宣传口号表明了自己对安全的承诺，但是并不能看作是对自己产品安全性的评价。

独立安全性评估

因此该如何看待 Oracle 的安全性承诺呢？他们那样说到底意味着什么？Oracle 已经投入了大量的资金来独立评估自己的产品。这些证明 Oracle 是安全的积极证据是用来欺骗消费者的。但是在“现实的”安全领域中，通过了通用标准下的 EAL4(保证等级 4)并不能说明什么。Oracle 和 IBM 的 Informix(EAL2)都通过了通用标准，但是它们都存在长用户名导致的缓冲区溢出漏洞。产品中的许多特征都能够通过这个标准，但是它们都

存在漏洞。如果城堡的门都是软的，那么这个城堡就不再是城堡了。

最早通过 EAL4 的 Oracle 版本是 1998 年 9 月发布的 7.2 版本，接下来就是 2000 年 10 月的 Oracle 8.0.5，然后是 2001 年 7 月的 8.1.7.0。2003 年 9 月，Oracle 9iR2 通过了认证，接下来在 2005 年 9 月，10g Release 1 也通过了认证。既然是为了通过认证，那么所有的这些版本都是有所欠缺的，是不合格的。

笔者并不是独立安全性评估的粉丝，但是却认为独立安全性评估确实有自己的作用，能为开发人员提供前进的目标。但是只有在软件开发不是为了遮掩或美观的时候，这种说法才成立。

展望未来

Oracle 10g Release 2 是一个很好的产品。在安全性方面，它对 10g Release 1 做了很大的改进。为此应该对 Oracle 加以称赞。但是，“任务还没有完成”。10g Release 2 中仍然存在故障(本书中讨论了很多)，而且还有很多故障没有被发现。尽管如此，在 10g Release 1 中只需要进行 5~10 分钟的搜索就可以发现一个新故障，而在 10g Release 2 中要花费一天或更多的精力才能找到一个新故障。这些进步在很大程度上得益于对源代码审计工具的巨大投资。这些工具负责寻找绝大多数隐患，但是它们也存有局限——例如，在 PL/SQL 过程调用 C 函数或 Java 函数的时候，这个工具似乎不能找到在这些交叉点所出现的隐患。为了找出这些残留的问题，Oracle 需要对这些工具进行改进。应该把源代码审计工具用作一种“最终的防御”机制。若要在安全性改善方面取得巨大的进步，开发者代码是关键，必须有好的安全编码标准和过程。就像 Microsoft 发布关于安全开发生命周期(Security Development Lifecycle，SDL)的标准和过程一样，笔者也希望 Oracle 能够这样做。这对整个行业也是有益的。

“安全技术经典译丛”系列图书简介

《入侵的艺术》

978-7-302-14272-0

Kevin D. Mitnick　William L. Simon 著

袁月杨　谢衡　译

继《欺骗的艺术》之后，安全领域又一传世佳作！在网络攻击和利用方面的传奇生涯使得本书作者 Kevin D. Mitnick 成为真正的黑客英雄，也使得本书对黑客入侵行为的分析更加专业、深入，并提供各种应对和防范措施。本书值得每一位对安全感兴趣的 IT 从业人员研读！

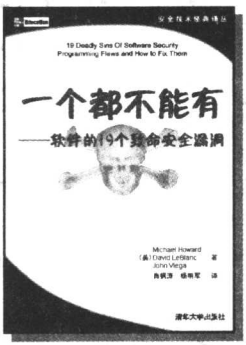
《ROOTKITS——Windows 内核的安全防护》

978-7-302-14652-0

Greg Hoglund　James Butler　著　韩智文　译

本书是目前第一本关于 rootkit 的详细指南，世界顶级安全专家、rootkit.com 创始人 Greg Hoglund 和 James Butler 帮助你全面掌握 rootkit，提升自己的安全防范能力！

《一个都不能有》



7-302-13804-4

Michael Howard　David LeBlanc　John Viega 著

肖枫涛　杨明军　译

本书全面展示软件中的 9 个常见安全漏洞以及应对策略！

《终极守护——针对黑客的高级防御技术》

978-7-302-15481-5

Victor Opplemen Oliver Friedrichs 等著　袁野　关翔　译

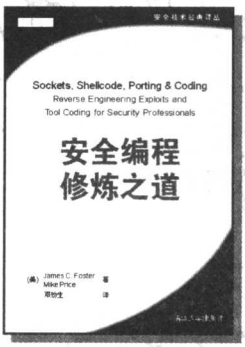
三位业界实战专家(包括 Symantec 安全响应中心的高级经理)帮助你保护自己的网络和网站免受恶意攻击，并奉献很多从未公布过的高级安全技术！

《缓冲区溢出攻击》

7-302-13942-3

James C. Foster　等著　蔡勉　译

如果你想切实掌握缓冲区溢出攻击方法和防范措施，那么该书就是你的不二选择！



《安全编程修炼之道》

7-302-13216-X

James C. Foster　Mike Price　著

邓劲生　译

一本可以帮助程序员全面提升编写安全代码的大作！

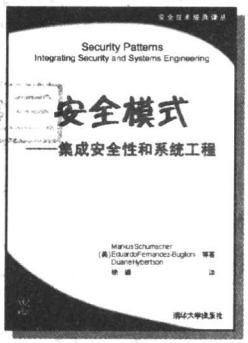
《数据库黑客大曝光》

7-302-14078-2

David Litchfield　Chirls Anley　John Heasman

Bill Grindlay　著　闫雷鸣　邢苏霄　译

四位世界顶级安全专家将手把手教你如何入侵和防御当前最流行的数据库服务器！



《安全模式》

978-7-302-14587-5

Markus Schumacher　Eduardo Fernandez-Buglioni

Duane Hybertson　著　徐璐　译

本书从不同的系统级别(企业级、体系结构级和操作层)对安全性进行解析，并展示如何在较为广泛的工程过程中集成安全性！

目　录

第 1 章

Oracle RDBMS 概述

本章概述了 Oracle 数据库服务器。除了一种新的攻击(见 1.2 节的“进程”部分)以外，这里介绍的内容适用于那些以前从未接触过 Oracle 的读者，因此对 Oracle 具有一定了解的读者可以直接学习本书的核心内容。本章涵盖了 Oracle 体系结构、数据库对象、用户和角色，以及权限，最后讨论了 Oracle 补丁。

1.1 体系结构

这里所谈论的 Oracle 数据库服务器究竟是指什么呢？数据库是指一个系统的所有数据——例如在磁盘上的所有数据。数据库实例是指支持数据查询的所有正在运行的进程和内存。通常，实例只服务于一个数据库，不过在集群(clustering)的情况下，可以为单个数据库服务运行多个实例。系统标识符(System Identifier)或 SID 是为数据库实例分配的名称。我们所使用的术语“主机”或“服务器”一般是指运行数据库服务器软件的机器，而不考虑实例或数据库。

该软件有两个主要组件：TNS Listener 和关系型数据库管理系统(Relational Database Management System，RDBMS)本身。TNS Listener 是所有 Oracle 通信的中心。当启动一个数据库实例的时候，它要向 TNS Listener 注册。当客户机希望访问数据库时，它首先要与 Listener 建立连接，然后再由 Listener 将客户机转接到数据库服务器。当 RDBMS 启动一个外部过程时，它首先会连接到 Listener，随后由 Listener 启动一个名为 extproc

的程序。由清华大学出版社引进并出版的《数据库黑客大曝光——数据库服务器防护术》一书已经完全涵盖了这部分内容，所以这里将不再赘述。本书中关于这部分内容的唯一例外是由数据库的作业调度器启动的外部作业， RDBMS 直接连接到了外部的作业服务器。稍后您将从本书中学到更多有关这方面的内容。

1.2 进程

根据是否在 Windows 或*nix 平台上考察 Oracle，组成 Oracle 服务器或服务的进程会有所变化。数据库实例描述了提供数据库访问的所有进程和内存结构。进程有两种——后台(background)进程和影像进程(shadow)或服务器(server)进程。影像进程或服务器进程服务于客户机请求。换句话说，当一个客户机连接到 TNS Listener 并请求访问数据库服务的时候，Listener 会将它们传给一个服务器进程，由该服务器进程提取出 SQL 查询语句并代表客户机执行这些语句。后台进程的存在是为了对它们提供支持。有很多不同的后台进程，每一个都有不同的角色，包括数据库写进程(Database Writer)、日志写进程(Log Writer)、存档(Archiever)、系统监控(System Monitor)、进程监控(Process Monitor)等。

可以说，在*nix 平台上，每个后台进程都是独立运行的进程，就像在一个操作系统进程中一样。在 Windows 平台上，它们被包装成一个更大的进程，即 Oracle.exe。有一个特殊的内存区域被映射为*nix 平台上的所有进程，该区域称为系统全局区(System Global Area，SGA)。SGA 是作为一个内存映射文件实现的，而且它包含实例和数据库的信息。此外，它还包含一个名为共享池(Shared Pool)的区域，共享池包含所有用户共享的结构，例如表定义等。

运行在 Windows 平台上的 Oracle 进程有一个非常有趣的方面，那就是每个用户组都可以程序化地打开进程。这是一个重要的安全隐患。请考虑下面的代码：

```
/*
Steps:

1) Get the Process ID for Oracle.exe - e.g. 1892
2) Get the Database SID - e.g. ORCL
3) Open 2 command shells - let's call them A and B
```