第一图书网, tushu007.com

# <<SQL注入攻击与防御>>

## 图书基本信息

书名：<<SQL注入攻击与防御>>

13位ISBN编号：9787302224136

10位ISBN编号：7302224137

出版时间：2010-6

出版时间：清华大学出版社

作者：克拉克

页数：339

字数：589000

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：http://www.tushu007.com

## 前言

十几年前，基于数据库的Web应用刚流行时，几乎所有开发商都忽略了SQL注入漏洞，导致当时大多数网站的登录入口形同虚设。
时至今日，Web应用已愈加成熟，安全性也不断得到加强。
遗憾的是，针对SQL注入漏洞的各种攻击工具也在推陈出新，不断地向安全管理人员发出新的挑战。
如何最大程度地降低SQL注入风险，从根本上实施SQL注入防御，成为网络管理人员和开发人员亟需解决的“ 烫手山芋 ”。
现在网络上关于SQL注入方面的教程比较零散，大多针对某一类具体应用，难以作为预防SQL注入的完整解决方案。
本书弥补了这一缺憾！
本书作者均是专门研究SQL注入的安全专家，他们集众家之长，对应用程序的基本编码和升级维护进行全程跟踪，详细讲解可能引发SQL，注入的行为以及攻击者的利用要素，并结合长期实践经验提出了相应的解决方案。
SOL注入利用的是正常的HTTP服务端口，表面上和正常的Web访问没有差别，隐蔽性极强。
针对这种情况，书中重点讲解了SQL注入的排查方法和可以借助的工具，总结了常见的利用SQL注入漏洞的方法。
开发人员和系统管理人员在SQL注入防御中扮演着重要角色，因此，书中专门从代码层和系统层角度介绍了避免SQL注入的各种策略和需要考虑的问题。
全书共l0章，分别介绍了SOL注入的基本概念，如何发现、确认并利用SQL注入和SQL盲注，利用操作系统防御SQL注入，SQL注入的一些高级话题，代码层和平台层防御等知识，书中主要针对的是Microsoft SQL Server、My SQL和Oracle这三大主流数据库。
本书注重于实践，涉及的内容也比较前沿，另外，还包含了大量翔实的案例，它们都具有很好的现实指导作用，读者可从中学到最新的攻击和防御技术。
本书主要由黄晓磊和李化翻译完成，全书由李化统稿。
由于本书内容较新、知识面广且译者水平有限，译文中难免存在错误之处，敬请读者批评指正。

## 内容概要

SQL注入是Internet上最危险、最有名的安全漏洞之一，本书是目前唯一一本专门致力于讲解SQL威胁的图书。
本书作者均是专门研究SQL注入的安全专家，他们集众家之长，对应用程序的基本编码和升级维护进行全面跟踪，详细讲解可能引发SQL注入的行为以及攻击者的利用要素，并结合长期实践经验提出了相应的解决方案。
针对SQL注入隐蔽性极强的特点，本书重点讲解了SQL注入的排查方法和可以借助的工具，总结了常见的利用SQL漏洞的方法。
另外，本书还专门从代码层和系统层的角度介绍了避免SQL注入的各种策略和需要考虑的问题。
本书主要内容 SQL注入一直长期存在，但最近有所增强。
本书包含所有与SQL注入攻击相关的、当前已知的信息，凝聚了由本书作者组成的、无私奉献的SQL注入专家团队的所有深刻见解。
什么是SQL注入?理解它是什么以及它的基本原理 查找、确认和自动发现SQL注入 查找代码中SQL注入时的提示和技巧 使用SQL注入创建利用 通过设计来避免由SQL攻击所带来的危险

## 作者简介

凭借这本《SQL注入攻击与防御》，测试人员现在有了一把弥补Internet上各种分散式教程不足的利器。
阅读本书您可以学会识别并利用各种平台上不同种类的SQL注入缺陷。

——Devon Kearna，安全分析师

## 书籍目录

第1章 什么是SQL注入 1.1 概述 1.2 理解Web应用的工作原理 1.2.1 一种简单的应用架构 1.2.2 一种较复杂的架构 1.3 理解SQL注入 1.4 理解SQL注入的产生过程 1.4.1 构造动态字符串 1.4.2 不安全的数据库配置 1.5 本章小结 1.6 快速解决方案 1.7 常见问题解答 第2章 SQL注入测试 2.1 概述 2.2 寻找SQL注入 2.2.1 借助推理进行测试 2.2.2 数据库错误 2.2.3 应用响应 2.2.4 SQL盲注 2.3 确认SQL注入 2.3.1 区分数字和字符串 2.3.2 内联SQL注入 2.3.3 终止式SQL注入 2.3.4 时间延迟 2.4 自动寻找SQL注入 2.5 本章小结 2.6 快速解决方案 2.7 常见问题解答 第3章 复查代码中的SQL注入 3.1 概述 3.2 复查源代码中的SQL注入 3.2.1 危险的编码行为 3.2.2 危险的函数 3.2.3 跟踪数据 3.2.4 复查PL/SQL和T-SQL代码 3.3 自动复查源代码第1章 什么是SQL注入 3.3.1 YASCA 3.3.2 Pixy 3.3.3 AppCodeScan 3.3.4 LAPSE 3.3.5 SWAAT 3.3.6 Microsoft SQL注入源代码分析器 3.3.7 CAT.NET 3.3.8 商业源代码复查工具 3.3.9 Ounce 3.3.10 Fortify源代码分析器 3.3.11 CodeSecure 3.4 本章小结 3.5 快速解决方案 3.6 常见问题解答 第4章 利用SQL注入 4.1 概述 4.2 理解常见的利用技术 4.3 识别数据库 4.3.1 非盲跟踪 4.3.2 盲跟踪 4.4 使用UINON语句提取数据 4.4.1 匹配列 4.4.2 匹配数据类型 4.5 使用条件语句 4.5.1 方法1：基于时间 4.5.2 方法2：基于错误 4.5.3 方法3：基于内容 4.5.4 处理字符串 4.5.5 扩展攻击 4.5.6 利用SQL注入错误 4.5.7 Oracle中的错误消息 4.6 枚举数据库模式 4.6.1 SQL Server 4.6.2 MySQL 4.6.3 Oracle 4.7 提升权限 4.7.1 SQL Server 4.7.2 Oracle 4.8 窃取哈希口令 4.8.1 SQL Server 4.8.2 MySQL 4.8.3 Oracle 4.9 带外通信 4.9.1 E-mail 4.9.2 HTTP/DNS 4.9.3 文件系统 4.10 自动利用SQL注入 4.10.1 Sqlmap 4.10.2 Bobcat 4.10.3 BSQL 4.10.4 其他工具 4.11 本章小结 4.12 快速解决方案 4.13 常见问题解答 第5章 SQL盲注利用 5.1 概述 5.2 寻找并确认SQL盲注 5.2.1 强制产生通用错误 5.2.2 注入带副作用的查询 5.2.3 拆分与平衡 5.2.4 常见的SQL盲注场景 5.2.5 SQL盲注技术 5.3 使用基于时间的技术 5.3.1 延迟数据库查询 5.3.2 基于时间推断的考虑 5.4 使用基于响应的技术 5.4.1 MySQL响应技术 5.4.2 SQL Server响应技术 5.4.3 Oracle响应技术 5.4.4 返回多位信息 5.5 使用非主流通道 5.5.1 数据库连接 5.5.2 DNS渗漏 5.5.3 E-mail渗漏 5.5.4 HTTP渗漏 5.6 自动SQL盲注利用 5.6.1 Absinthe 5.6.2 BSQL Hacker 5.6.3 SQLBrute 5.6.4 Sqlninja 5.6.5 Squeeza 5.7 本章小结 5.8 快速解决方案 5.9 常见问题解答 第6章 利用操作系统 6.1 概述 6.2 访问文件系统 6.2.1 读文件 6.2.2 写文件 6.3 执行操作系统命令 6.4 巩固访问 6.5 本章小结 6.6 快速解决方案 6.7 常见问题解答 6.8 尾注 第7章 高级话题 7.1 概述 7.2 避开输入过滤器 7.2.1 使用大小写变种 7.2.2 使用SQL注释 7.2.3 使用URL编码 7.2.4 使用动态的查询执行 7.2.5 使用空字节 7.2.6 嵌套剥离后的表达式 7.2.7 利用截断 7.2.8 避开自定义过滤器 7.2.9 使用非标准入口点 7.3 利用二阶SQL注入 7.4 使用混合攻击 7.4.1 修改捕获的数据 7.4.2 创建跨站脚本 7.4.3 在Oracle上运行操作系统命令 7.4.4 利用验证过的漏洞 7.5 本章小结 7.6 快速解决方案 7.7 常见问题解答 第8章 代码层防御 8.1 概述 8.2 使用参数化语句 8.2.1 Java中的参数化语句 8.2.2 .NET(C#)中的参数化语句 8.2.3 PHP中的参数化语句 8.2.4 PL/SQL中的参数化语句 8.3 输入验证 8.3.1 白名单 8.3.2 黑名单 8.3.3 Java中的输入验证 8.3.4 .NET中的输入验证 8.3.5 PHP中的输入验证 8.4 编码输出 8.5 规范化 8.6 通过设计来避免SQL注入的危险 8.6.1 使用存储过程 8.6.2 使用抽象层 8.6.3 处理敏感数据 8.6.4 避免明显的对象名 8.6.5 创建数据库Honeypot 8.6.6 附加的安全开发资源 8.7 本章小结 8.8 快速解决方案 8.9 常见问题解答 第9章 平台层防御 9.1 概述 9.2 使用运行时保护 9.2.1 Web应用防火墙 9.2.2 截断过滤器 9.2.3 不可编辑的输入保护与可编辑的输入保护 9.2.4 URL策略/页面层策略 9.2.5 面向方面编程 9.2.6 应用入侵检测系统 9.2.7 数据库防火墙 9.3 确保数据库安全 9.3.1 锁定应用数据 9.3.2 锁定数据库服务器 9.4 额外的部署考虑 9.4.1 最小化不必要信息的泄露 9.4.2 提高Web服务器日志的冗余 9.4.3 在独立主机上部署Web服务器和数据库服务器 9.4.4 配置网络访问控制 9.5 本章小结 9.6

快速解决方案 9.7 常见问题解答 第10章 参考资料 10.1 概述 10.2 SQL入门 10.3 SQL注入快速参考 10.3.1 识别数据库平台 10.3.2 Microsoft SQL Server备忘单 10.3.3 MySQL备忘单 10.3.4 Oracle备忘单

## 章节摘录

8.9 常见问题解答　　问题：为什么不能使用参数化语句来提供表名或列名？
解答：不能在参数化语句中提供SQL标识符，是因为在数据库中它们会被编译并且之后会被提供的数据填充。
这要求SQL标识符在提供数据之前的编译期间出现。
问题：为什么不能拥有参数化的ORDERBY子句？
解答：这个问题的答案与上一问题相同，因为ORDERBY包含一个SQL标识符，也就是要进行排序的列。
问题：如何在x技术中对Y数据库使用参数化语句？
解答：大多数现代编程语言和数据库均支持参数化语句。
请查看当前使用的数据库访问API的文档。
请记住，有时也将这些语句称为预处理语句。
问题：怎样参数化一个存储过程调用？
解答：在大多数编程语言中，这与使用参数化语句非常类似或者完全相同。
请查询当前使用的数据库访问API的文档。
请记住，有时也将这些语句称为可调用语句。
问题：从哪里获取良好的用于验证x的黑名单？
解答：非常不幸，向黑名单中放入什么内容取决于应用的语境。
如果可能的话，请尽量不要使用黑名单，因为我们无法列举出所有的潜在攻击或恶意输入。
如果必须使用黑名单，则请确保您要么使用输出编码，要么将黑名单输入验证作为唯一的验证方法。
问题：使用白名单输入验证是安全的吗？
解答：不是。
这取决于您允许通过的内容。
例如，可能允许输入单引号，当在动态SQL中包含这样的输入时就会产生问题。
问题：哪些场合比较适合使用白名单输入验证？
哪些场合适合使用黑名单输入验证？
解答：应该在应用中接收输入的地方使用白名单输入验证，以便对敏感内容应用验证。
在Web应用防火墙或类似的位置适合将黑名单验证作为附加的控制，以此来检测明显的SQL注入攻击企图。
问题：需要对发送给数据库和从数据库获取的输入都进行编码吗？
为什么？
解答：不管在哪里使用动态SQL，都需要确保提交给数据库的内容不会引发SQL注入问题。
这并不意味着恶意内容已经变得安全。
当从数据库查询这些内容并在其他地方的动态SOL中使用时，还是会存在危险。
问题：应该在哪些位置进行编码？
解答：应该在使用信息的位置附近进行编码。
如果在数据未到达数据库之前向数据库提交数据，那么就应该对数据进行编码。
应该在有可能使用数据的位置附近（例如，将数据展示给用户之前针对跨站脚本编码）或者在动态SQL中使用数据之前（针对SQL注入编码）对来自数据库的数据进行编码。

## 编辑推荐

唯一一本关于SQL注入攻击与防御的专业书籍　　理解，发现、利用和防御SQL注入的最佳指导
见解精辟，丰富、精彩的SQL注入示例及防御策略　　作者多年长期实践经验的总结

## 版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:http://www.tushu007.com