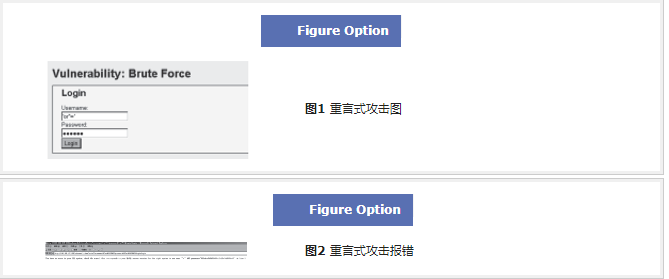
**SQL注入攻击与防御技术**

**1 SQL注入攻击技术**

在SQL注入攻击中, 攻击者通过构造SQL查询语句绕过服务器验证或者获取数据库中的内容。下面将介绍重言式、注释符、逻辑错误查询、联合查询、附带查询、存储过程等SQL注入攻击技术。

1）重言式攻击

攻击者构造SQL查询语句, 使得SQL查询语句中的条件语句恒为真, 其意图就是绕过服务器对用户的身份验证, 从而获取可注入的参数和数据库中的信息。这种攻击能否成功主要取决于SQL查询语句执行的结果是否能影响应用程序。如[图1](http://netinfo-security.org/article/2015/1671-1122-0-9-129.html" \l "outline_anchor_7)所示, 在登录界面中提交用户名为“ ’ or 1=1--” , 如果开发人员对输入进行验证, 那么可以出现[图2](http://netinfo-security.org/article/2015/1671-1122-0-9-129.html" \l "outline_anchor_8)的提示; 如果开发人员编写的代码没有对输入进行合法验证, 那么验证用户身份的SQL查询语句就可能变成“ select cout(\* ) from user where username=‘ ’ or l=1-- and passwd=‘ ’ ” , 这样where语句就成为重言式, 数据库将返回数据, 最终绕过用户的验证。



2）使用注释符

SQL语言中的注释符（--, #等）跟其他编程语言中的注释符一样, 可以使注释的代码不执行, 因此攻击者就可以在查询语句中插入注释符, 来绕过服务器验证。例如, 后台登录查询语句select \* from user where username=name and passwd=pwd。当攻击者输入用户名admin--时, 如果服务器没有对用户的输入进行验证并且后台管理员的用户名正好为admin, 那么这条查询语句就会变成select \* from user where username=admin, 后面的语句被注释不会执行, 这样攻击者在不提交密码的情况下就能渗透登录后台。

3）逻辑错误查询

逻辑错误查询的目的是获取数据库的类型、版本和参数, 就是前期的信息收集。该种攻击方式是攻击者为下一步攻击作铺垫, 攻击者故意构造不能执行的SQL查询语句导致数据库出现语法错误, 以至于可以从返回的错误信息当中得到一些有价值的参数、数据库的类型和关系结构, 甚至可以返回数据库中的表名和列名。这些信息为下一步的攻击奠定了基础。

4）联合查询攻击

联合（union）查询攻击就是非法用户使Union连接两个不同的查询语句, 从而得到额外的数据信息。非法用户首先构造联合查询将两个语句的查询结果合并起来, 然后经过order by等操作, 最终可以得想要的结果。如[图3](http://netinfo-security.org/article/2015/1671-1122-0-9-129.html" \l "outline_anchor_9)所示, 构造的SQL查询语句为“ http://192.168.127.128/vulnerabilities/sqli/?id=1'%20union%20select%201, 2%20+--+%20& Submit=Submit” 。

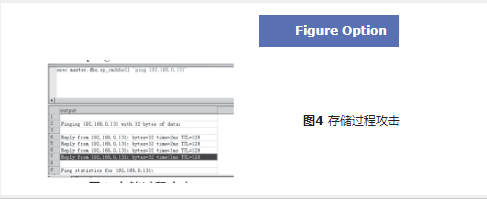


5）附带查询攻击

附带查询攻击是攻击者在不改变原来查询语句的情况下, 在语句后面加入新的查询语句, 从而对数据库进行增、删、改、查等操作, 甚至可以实现拒绝服务和执行远程命令。附带查询攻击能否成功取决于数据库的配置是否支持多语句, 如果支持, 那么这种攻击方式危害极大, 否则很难发生。

6）存储过程攻击

存储过程可以是系统的存储过程, 也可以是用户自定义的存储过程。非法用户通过执行数据库默认开启的存储过程可以提权、执行远程命令以及拒绝服务。例如, SQL Server的存储过程xp\_cmdshell, 该存储过程可以使用cmd命令, 包括增加用户、列目录和ping等功能, [图4](http://netinfo-security.org/article/2015/1671-1122-0-9-129.html" \l "outline_anchor_10)显示使用xp\_cmdshell进行ping命令。



7）盲注

盲注是在Web服务器没有返回任何有用的数据或信息情况下进行的SQL注入攻击, 主要是对某一个参数构造不同的查询语句, 根据结果来判断参数是否可以注入。采用盲注是因为开发人员对服务器返回的错信息进行了过滤, 攻击者无法获取数据库相关的信息。利用盲注, 非法用户可以根据Web应用的反应和变化以及抓取数据包来推测出数据库中有哪些参数、哪些表等。对于盲注来说, and 1=1和and 1=2是检测注入点的经典方法。如[图5](http://netinfo-security.org/article/2015/1671-1122-0-9-129.html" \l "outline_anchor_11)和[图6](http://netinfo-security.org/article/2015/1671-1122-0-9-129.html" \l "outline_anchor_12)所示, 当使用and 1=1时网页正常, 但是用and 1=2时, 网页出错。另外还有时间测试注入, 利用BENCHMARK()函数来观察Web应用反应的时间来判断注入是否成功。



8）替换编码

替换编码是非法用户对一些特殊的字符进行URL或者uncode编码来绕过程序的验证, 而经过这些字符编码之后程序仍然会在服务器或者浏览器中执行。目前有很多的应用安装了防止SQL注入的程序, 一般来说是对一些异常字符过滤, 如引号、注释符等, 还有对html中的标记进行过滤, 这样可以使非法用户的恶意代码失效。此时通过替换编码就可以绕过这些程序, 例如, shutdown命令可以用16进制进行字符串编码成ox73687574646f776来绕过过滤规则。