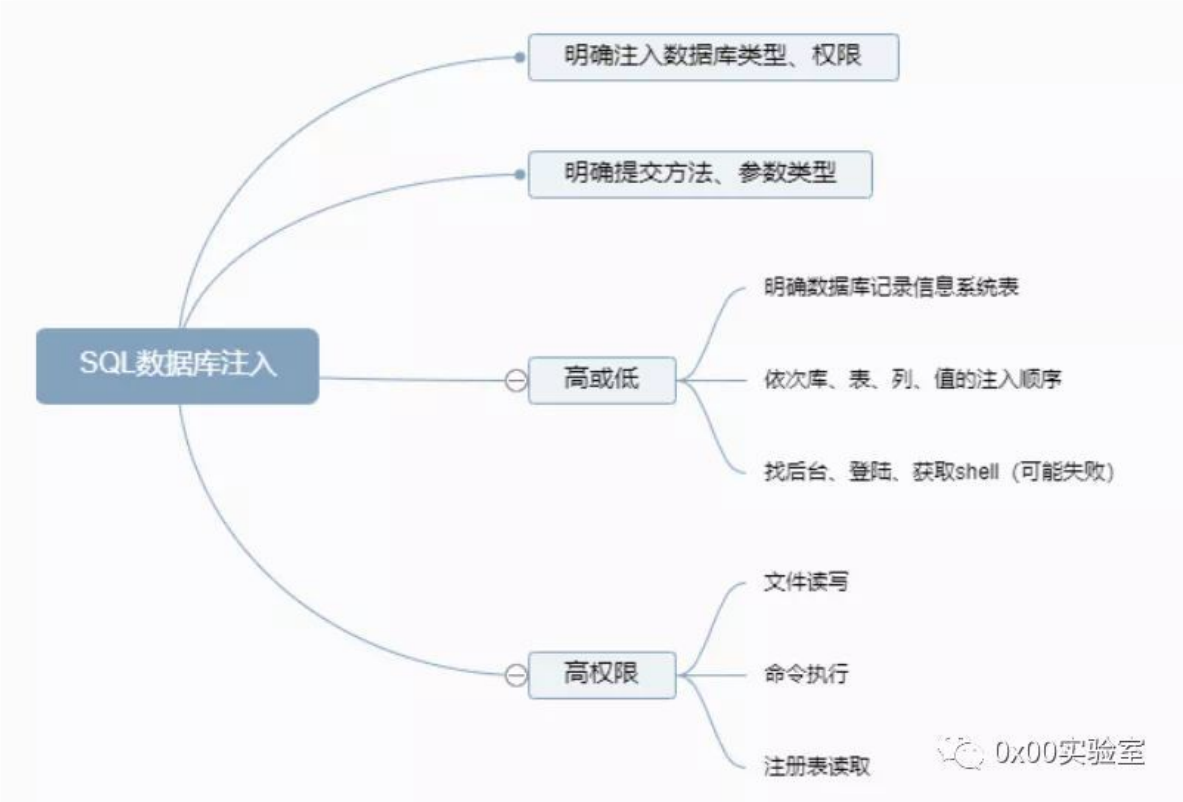
Day15　WEB漏洞–SQL注入之Oracle，MongDB等注入



15.1 简要学习各种数据库的注入特点

- Access, mysql ,mssql(sql server) mongoDB, postgresql, sqlite,oracle, sybase等
- Access数据库放在网站下
- 数据库架构组成，数据库高权限操作

15.2 简要学习各种注入工具的使用指南

15.2.1Sqlmap工具

简介：

> Sqlmap是一个开源的渗透测试工具，可以用来自动化的检测，利用SQL注入漏洞，获取数据库服务器的权限。它具有功能强大的检测引擎，针对各种不同类型数据库的渗透测试的功能选项，包括获取数据库中存储的数据，访问操作系统文件甚至可以通过外带数据连接的方式执行操作系统命令。

Sqlmap支持的注入方式：

- 基于布尔类型的盲注：即可以根据返回页面判断条件真假的注入。
- 基于时间的盲注：即不能根据页面返回的内容判断任何信息，要用条件语句查看时间延迟语句是否已执行(即页面返回时间是否增加)来判断。
- 基于报错注入：即页面会返回错误信息，或者把注入的语句的结果直接返回到页面中。
- 联合查询注入：在可以使用Union的情况下的注入。
- 堆查询注入：可以同时执行多条语句时的注入。
- 带外注入：构造SQL语句，这些语句在呈现给数据库时会触发数据库系统创建与攻击者控制的外部服务器的连接。以这种方式，攻击者可以收集数据或可能控制数据库的行为。

Sqlmap输出级别：

```
只显示python错误以及严重的信息。
同时显示基本信息和警告信息。（默认）
同时显示debug信息。
同时显示注入的payload。
同时显示HTTP请求。
同时显示HTTP响应头。
同时显示HTTP响应页面。
```

指令大全：

```
用法：python sqlmap.py [选项]

选项：
  -h, --help            显示基本帮助信息并退出
  -hh                   显示高级帮助信息并退出
  --version             显示程序版本信息并退出
  -v VERBOSE            输出信息详细程度级别：0-6（默认为 1）

  目标：
      至少提供一个以下选项以指定目标

    -d DIRECT           直接连接数据库
    -u URL, --url=URL   目标 URL（例如："http://www.site.com/vuln.php?id=1"）
    -l LOGFILE          从 Burp 或 WebScarab 代理的日志文件中解析目标地址
    -x SITEMAPURL       从远程网站地图（.xml）文件中解析目标
    -m BULKFILE         从文本文件中获取批量目标
    -r REQUESTFILE      从文件中读取 HTTP 请求
    -g GOOGLEDORK       使用 Google dork 结果作为目标
    -c CONFIGFILE       从 INI 配置文件中加载选项

  请求：
      以下选项可以指定连接目标地址的方式

    --method=METHOD     强制使用提供的 HTTP 方法（例如：PUT）
    --data=DATA         使用 POST 发送数据串
```

```
26      --param-del=PARA..  设置参数值分隔符
27      --cookie=COOKIE     指定 HTTP Cookie
28      --cookie-del=COO..  设置 cookie 分隔符
29      --load-cookies=L..  指定以 Netscape/wget 格式
存放 cookies 的文件
30      --drop-set-cookie   忽略 HTTP 响应中的 Set-
Cookie 参数
31      --user-agent=AGENT  指定 HTTP User-Agent
32      --random-agent      使用随机的 HTTP User-
Agent
33      --host=HOST         指定 HTTP Host
34      --referer=REFERER   指定 HTTP Referer
35      -H HEADER, --hea..  设置额外的 HTTP 头参数（例
如："X-Forwarded-For: 127.0.0.1"）
36      --headers=HEADERS   设置额外的 HTTP 头参数（例
如："Accept-Language: fr\nETag: 123"）
37      --auth-type=AUTH..  HTTP 认证方式（Basic，
Digest，NTLM 或 PKI）
38      --auth-cred=AUTH..  HTTP 认证凭证
（username:password）
39      --auth-file=AUTH..  HTTP 认证 PEM 证书/私钥文
件
40      --ignore-code=IG..  忽略 HTTP 错误码（例如：
401）
41      --ignore-proxy      忽略系统默认代理设置
42      --ignore-redirects  忽略重定向尝试
43      --ignore-timeouts   忽略连接超时
44      --proxy=PROXY       使用代理连接目标 URL
45      --proxy-cred=PRO..  使用代理进行认证
（username:password）
46      --proxy-file=PRO..  从文件中加载代理列表
47      --tor               使用 Tor 匿名网络
```

```
48      --tor-port=TORPORT   设置 Tor 代理端口代替默认端
    口
49      --tor-type=TORTYPE   设置 Tor 代理方式（HTTP,
    SOCKS4 或 SOCKS5（默认））
50      --check-tor          检查是否正确使用了 Tor
51      --delay=DELAY        设置每个 HTTP 请求的延迟秒
    数
52      --timeout=TIMEOUT    设置连接响应的有效秒数（默认
    为 30）
53      --retries=RETRIES    连接超时时重试次数（默认为
    3）
54      --randomize=RPARAM   随机更改给定的参数值
55      --safe-url=SAFEURL   测试过程中可频繁访问且合法的
    URL 地址（译者注:
56                           有些网站在你连续多次访问错误
    地址时会关闭会话连接,
57                           后面的“请求”小节有详细说明）
58      --safe-post=SAFE..   使用 POST 方法发送合法的数
    据
59      --safe-req=SAFER..   从文件中加载合法的 HTTP 请
    求
60      --safe-freq=SAFE..   每访问两次给定的合法 URL 才
    发送一次测试请求
61      --skip-urlencode     不对 payload 数据进行 URL
    编码
62      --csrf-token=CSR..   设置网站用来反 CSRF 攻击的
    token
63      --csrf-url=CSRFURL   指定可提取反 CSRF 攻击
    token 的 URL
64      --force-ssl          强制使用 SSL/HTTPS
65      --hpp                使用 HTTP 参数污染攻击
```

```
66      --eval=EVALCODE      在发起请求前执行给定的
    Python 代码（例如：
67                           "import
    hashlib;id2=hashlib.md5(id).hexdigest()"）
68
69    优化：
70      以下选项用于优化 sqlmap 性能
71
72      -o                   开启所有优化开关
73      --predict-output     预测常用请求的输出
74      --keep-alive         使用持久的 HTTP(S) 连接
75      --null-connection    仅获取页面大小而非实际的
    HTTP 响应
76      --threads=THREADS    设置 HTTP(S) 请求并发数最
    大值（默认为 1）
77
78    注入：
79      以下选项用于指定要测试的参数，
80      提供自定义注入 payloads 和篡改参数的脚本
81
82      -p TESTPARAMETER     指定需要测试的参数
83      --skip=SKIP          指定要跳过的参数
84      --skip-static        指定跳过非动态参数
85      --param-exclude=..   用正则表达式排除参数（例
    如："ses"）
86      --dbms=DBMS          指定 DBMS 类型（例如：
    MySQL）
87      --dbms-cred=DBMS..   DBMS 认证凭据
    （username:password）
88      --os=OS              指定 DBMS 服务器的操作系统
    类型
89      --invalid-bignum     将无效值设置为大数
```

```
90      --invalid-logical    对无效值使用逻辑运算
91      --invalid-string     对无效值使用随机字符串
92      --no-cast            关闭 payload 构造机制
93      --no-escape          关闭字符串转义机制
94      --prefix=PREFIX      注入 payload 的前缀字符串
95      --suffix=SUFFIX      注入 payload 的后缀字符串
96      --tamper=TAMPER      用给定脚本修改注入数据
97
98    检测:
99      以下选项用于自定义检测方式
100
101     --level=LEVEL        设置测试等级（1-5，默认为
    1）
102     --risk=RISK          设置测试风险等级（1-3，默认
    为 1）
103     --string=STRING      用于确定查询结果为真时的字符
    串
104     --not-string=NOT..   用于确定查询结果为假时的字符
    串
105     --regexp=REGEXP      用于确定查询结果为真时的正则
    表达式
106     --code=CODE          用于确定查询结果为真时的
    HTTP 状态码
107     --text-only          只根据页面文本内容对比页面
108     --titles             只根据页面标题对比页面
109
110   技术:
111     以下选项用于调整特定 SQL 注入技术的测试方法
112
113     --technique=TECH     使用的 SQL 注入技术（默认为
    “BEUSTQ”，译者注:
```

```
                              B: Boolean-based blind SQL injection（布尔型盲注）
                              E: Error-based SQL injection（报错型注入）
                              U: UNION query SQL injection（联合查询注入）
                              S: Stacked queries SQL injection（堆查询注入）
                              T: Time-based blind SQL injection（时间型盲注）
                              Q: inline Query injection（内联查询注入）
    --time-sec=TIMESEC        延迟 DBMS 的响应秒数（默认为 5）
    --union-cols=UCOLS        设置联合查询注入测试的列数目范围
    --union-char=UCHAR        用于暴力猜解列数的字符
    --union-from=UFROM        设置联合查询注入 FROM 处用到的表
    --dns-domain=DNS..        设置用于 DNS 渗出攻击的域名（译者注:
                              推荐阅读《在SQL注入中使用DNS获取数据》
http://cb.drops.wiki/drops/tips-5283.html,
                              在后面的“技术”小节中也有相应解释）
    --second-order=S..        设置二阶响应的结果显示页面的URL（译者注:
                              该选项用于二阶 SQL 注入）

  指纹识别:
```

```
132     -f, --fingerprint   执行广泛的 DBMS 版本指纹识别
133
134   枚举：
135       以下选项用于获取后端数据库管理系统的信息，结构和数据表中的数据。
136       此外，还可以运行你输入的 SQL 语句
137
138     -a, --all           获取所有信息、数据
139     -b, --banner        获取 DBMS banner
140     --current-user      获取 DBMS 当前用户
141     --current-db        获取 DBMS 当前数据库
142     --hostname          获取 DBMS 服务器的主机名
143     --is-dba            探测 DBMS 当前用户是否为DBA（数据库管理员）
144     --users             枚举出 DBMS 所有用户
145     --passwords         枚举出 DBMS 所有用户的密码哈希
146     --privileges        枚举出 DBMS 所有用户特权级
147     --roles             枚举出 DBMS 所有用户角色
148     --dbs               枚举出 DBMS 所有数据库
149     --tables            枚举出 DBMS 数据库中的所有表
150     --columns           枚举出 DBMS 表中的所有列
151     --schema            枚举出 DBMS 所有模式
152     --count             获取数据表数目
153     --dump              导出 DBMS 数据库表项
154     --dump-all          导出所有 DBMS 数据库表项
155     --search            搜索列，表和/或数据库名
156     --comments          获取 DBMS 注释
157     -D DB               指定要枚举的 DBMS 数据库
158     -T TBL              指定要枚举的 DBMS 数据表
```

```
159         -C COL                指定要枚举的 DBMS 数据列
160         -X EXCLUDECOL         指定要排除的 DBMS 数据列
161         -U USER               指定枚举的 DBMS 用户
162         --exclude-sysdbs      枚举所有数据表时，指定排除特
    定系统数据库
163         --pivot-column=P..    指定主列
164         --where=DUMPWHERE     在转储表时使用 WHERE 条件语
    句
165         --start=LIMITSTART    指定要导出的数据表条目开始行
    数
166         --stop=LIMITSTOP      指定要导出的数据表条目结束行
    数
167         --first=FIRSTCHAR     指定获取返回查询结果的开始字
    符位
168         --last=LASTCHAR       指定获取返回查询结果的结束字
    符位
169         --sql-query=QUERY     指定要执行的 SQL 语句
170         --sql-shell           调出交互式 SQL shell
171         --sql-file=SQLFILE    执行文件中的 SQL 语句
172
173       暴力破解:
174         以下选项用于暴力破解测试
175
176         --common-tables       检测常见的表名是否存在
177         --common-columns      检测常用的列名是否存在
178
179       用户自定义函数注入:
180         以下选项用于创建用户自定义函数
181
182         --udf-inject          注入用户自定义函数
183         --shared-lib=SHLIB    共享库的本地路径
184
```

```
185     访问文件系统:
186       以下选项用于访问后端数据库管理系统的底层文件系统
187
188       --file-read=RFILE    读取后端 DBMS 文件系统中的
      文件
189       --file-write=WFILE   写入后端 DBMS 文件系统中的
      文件
190       --file-dest=DFILE    使用文件绝对路径写入到后端
      DBMS
191
192     访问操作系统:
193       以下选项用于访问后端数据库管理系统的底层操作系统
194
195       --os-cmd=OSCMD       执行操作系统命令
196       --os-shell           调出交互式操作系统 shell
197       --os-pwn             调出 OOB shell,
      Meterpreter 或 VNC
198       --os-smbrelay        一键调出 OOB shell,
      Meterpreter 或 VNC
199       --os-bof             利用存储过程的缓冲区溢出
200       --priv-esc           数据库进程用户提权
201       --msf-path=MSFPATH   Metasploit 框架的本地安装
      路径
202       --tmp-path=TMPPATH   远程临时文件目录的绝对路径
203
204     访问 Windows 注册表:
205       以下选项用于访问后端数据库管理系统的 Windows 注册
      表
206
207       --reg-read           读取一个 Windows 注册表键
      值
```

```
208       --reg-add             写入一个 Windows 注册表键
      值数据
209       --reg-del             删除一个 Windows 注册表键
      值
210       --reg-key=REGKEY      指定 Windows 注册表键
211       --reg-value=REGVAL    指定 Windows 注册表键值
212       --reg-data=REGDATA    指定 Windows 注册表键值数
      据
213       --reg-type=REGTYPE    指定 Windows 注册表键值类
      型
214
215     通用选项:
216       以下选项用于设置通用的参数
217
218       -s SESSIONFILE        从文件（.sqlite）中读入会话
      信息
219       -t TRAFFICFILE        保存所有 HTTP 流量记录到指
      定文本文件
220       --batch               从不询问用户输入，使用默认配
      置
221       --binary-fields=..    具有二进制值的结果字段（例
      如："digest"）
222       --check-internet      在访问目标之前检查是否正常连
      接互联网
223       --crawl=CRAWLDEPTH    从目标 URL 开始爬取网站
224       --crawl-exclude=..    用正则表达式筛选爬取的页面
      （例如："logout"）
225       --csv-del=CSVDEL      指定输出到 CVS 文件时使用的
      分隔符（默认为“,”）
226       --charset=CHARSET     指定 SQL 盲注字符集（例
      如："0123456789abcdef"）
```

```
227       --dump-format=DU..   导出数据的格式（CSV（默
      认），HTML 或 SQLITE）
228       --encoding=ENCOD..   指定获取数据时使用的字符编码
      （例如：GBK）
229       --eta                显示每个结果输出的预计到达时
      间
230       --flush-session      清空当前目标的会话文件
231       --forms              解析并测试目标 URL 的表单
232       --fresh-queries      忽略存储在会话文件中的查询结
      果
233       --har=HARFILE        将所有 HTTP 流量记录到一个
      HAR 文件中
234       --hex                获取数据时调用 DBMS 的 hex
      函数
235       --output-dir=OUT..   自定义输出目录路径
236       --parse-errors       从响应中解析并显示 DBMS 错
      误信息
237       --save=SAVECONFIG    将选项设置保存到一个 INI 配
      置文件
238       --scope=SCOPE        用正则表达式从提供的代理日志
      中过滤目标
239       --test-filter=TE..   根据 payloads 和/或标题
      （例如：ROW）选择测试
240       --test-skip=TEST..   根据 payloads 和/或标题
      （例如：BENCHMARK）跳过部分测试
241       --update             更新 sqlmap
242
243     其他选项：
244       -z MNEMONICS         使用短助记符（例如：
      “flu,bat,ban,tec=EU”）
245       --alert=ALERT        在找到 SQL 注入时运行 OS
      命令
```

```
246       --answers=ANSWERS    设置问题答案（例如:
      “quit=N,follow=N”）
247       --beep               出现问题提醒或在发现 SQL 注
      入时发出提示音
248       --cleanup            指定移除 DBMS 中的特定的
      UDF 或者数据表
249       --dependencies       检查 sqlmap 缺少什么（非核
      心）依赖
250       --disable-coloring   关闭彩色控制台输出
251       --gpage=GOOGLEPAGE   指定页码使用 Google dork
      结果
252       --identify-waf       针对 WAF/IPS/IDS 保护进行
      彻底的测试
253       --mobile             使用 HTTP User-Agent 模仿
      智能手机
254       --offline            在离线模式下工作（仅使用会话
      数据）
255       --purge-output       安全地删除输出目录的所有内容
256       --skip-waf           跳过启发式检测 WAF/IPS/IDS
      保护
257       --smart              只有在使用启发式检测时才进行
      彻底的测试
258       --sqlmap-shell       调出交互式 sqlmap shell
259       --tmp-dir=TMPDIR     指定用于存储临时文件的本地目
      录
260       --web-root=WEBROOT   指定 Web 服务器根目录（例
      如: "/var/www"）
261       --wizard             适合初级用户的向导界面
```

15.2.2NoSQLAttack工具

简介

> NoSQLAttack是一个python编写的专门用于对MongoDB数据库进行SQL注入的渗透测试工具，非常简单好用，并且极大的弥补了sqlmap的不足（sqlmap不能对MongoDB数据库进行渗透）。

使用:

```
1-Scan attacked IP
2-Configurate parameters
3-MongoDB Access Attacks
4-Injection Attacks
x-Exit
```

资源

```
https://www.cnblogs.com/xishaonian/p/6173644.html
https://www.cnblogs.com/bmjoker/p/9326258.html
https://github.com/youngyangyang04/NoSQL Attack
linux环境下支持MongoDB的工具，P14
https://github.com/sqlmapproject/sqlmap
https://blog.csdn.net/qq_39936434/category_9103379.html
https://blog.csdn.net/hack8/article/details/6427911
```