日志管理和统计

write by Kevinzou
kissingwolf@gmail.com

Kevinzou (kissingwolf@gmail.com)

本文目录

第1节　日志在系统中的应用

1. 日志系统就是记账系统
2. 日志系统分类
   - 操作系统日志
   - 应用程序日志
   - 安全系统日志
3. 存放日志的方式
   - ACSII 码文本日志
   - 二进制日志
4. 日志的自动化分析

在一个完整的系统管理体系里，日志系统是一个非常重要的功能组成部分。它负责记录下系统或程序所产生的行为，并按照某种规范表达出来。系统管理员可以使用日志系统所记录的信息为系统或程序进行排错、优化系统的性能和根据这些信息调整系统或程序运行的行为。在安全领域，日志系统的重要地位尤甚，可以说是安全审计方面最主要的工具之一。

按照系统类型进行区分的话，日志系统可以分为操作系统日志、应用程序日志、安全系统日志等等。每种操作系统的日志都有其自身特有的设计和规范，例如 Windows 系统的日志通常按照其惯有的应用程序、安全和系统这样的分类方式进行存储，而 Linux 和其他类 UNIX 系统通常都使用兼容 Syslog 规范的日志系统。

而很多硬件设备的操作系统也具有独立的日志功能，以 Cisco 路由器为代表的网络设备通常都具有输出 Syslog 兼容日志的能力。应用系统日志主要包括各种应用程序服务器（例如 Web 服务器、FTP 服务器）的日志系统和应用程序自身的日志系统，不同的应用系统都具有根据其自身要求设计的日志系统。安全系统日志从狭义上讲指信息安全方面设备或软件如防火墙系统的日志，从更广泛的意义上来说，所有为了安全目的所产生的日志都可归入此类。

值得注意的是，对于文本格式的日志文件，我们通过基本的 grep 命令就可以执行这样

的搜索。而对于使用二进制格式存储的日志文件，执行这种检查会复杂一些，通常需要利用日志系统本身的阅读程序进行读取或解码。这需要管理员非常熟悉所维护系统相关的安全漏洞，以及相应系统下的脚本编写技术，但是在这些方面付出汗水是绝对值得的，至少像你在维护相当多的系统时还可以挤出时间阅读技术类杂志是很让人羡慕的。

编写脚本定期分析日志，脚本报警后采取相应的补救行动，能阻挡不少恶意访问行为和系统漏洞。但这并不是一个系统管理员生活的全部，还是有很多问题会安然通过这些检测，对系统形成破坏。从本质上来讲，自动化的日志处理是相对粗旷的，主要起到过滤没有意义的干扰信息，有效降低系统管理员工作负荷的作用。在日常监测之外，也应该每隔一段时间对这些日志进行更细致的审查。在新漏洞出现时，系统管理员针对漏洞的特征进行日志检查，并及时的将攻击指纹更新到自动化检查脚本中。

在日志系统的设置方面，系统管理员应该根据实际情况仔细的斟酌报警信息的详细程度，例如日志条目中我们除了记录网络访问的 IP 地址是否还需要记录主机名，但如果你在公网上的服务器这么设置了，此操作会相当损耗你的系统资源，其原因主要是 IP 到域名的反向解析耗时且不一定有结果。

在很多情况下，日志系统默认的设置并没有将对系统管理有用的字段记录下来，这需要系统管理员对系统的情况始终保持警觉，并根据经验不断的做出完善。另外，日志记录的保存期限也应与日志的管理计划相适应，各种系统的日志保存期限通常应保持同步，否则会发生管理员在上个月的操作系统日志中发现疑点，而 logrotate 却是每周清空这样的窘况。 这种情况的发生乍看之下不太可能，实际工作中确是经常出现的，因为 Apache 日志通常膨胀速度很快，每小时有几十兆、几百兆、甚至几千兆的增长是极为常见的。这种日志容量的增长也是和日志详细程度和保存期限密切相关的，很多文件系统对文件最大容量是有限制的，同时过于庞大的日志也不利于我们对日志的分析。所以我们在制订日志管理计划之前，应在综合所有因素的情况下，从全局的角度做出判断。

良好的日志管理对于安全预警、系统维护乃至起诉攻击者都是必不可少的，所有的系统管理员都应该学习如何善用日志系统，它不但能使大家的工作更加轻松，也能有效的保护企业中计算机设施不受侵害。

第2节　Linux 系统日志

1. syslogd 日志系统
2. 系统日志子系统
   - 连接时间日志
   - 进程统计日志
   - 错误日志
   - 程序日志
3. syslogd 日志系统配置规则

syslogd 可以简单地被称为记录系统活动的一个 daemons。比如可以记录谁，在什么时间，在哪里，做了什么事情(像是在写记叙文啊)；也可以记录您的系统曾经发生过什么事情，比如什么时候重新引导过、软硬件的错误信息等；当然也记录着您系统上运行着的服务的信息。类 UNIX 系统中，syslog 一般会记录大量的数据，其中包括由不同硬件和系统报告的登录、性能信息和故障。除 syslog 外，类 UNIX 系统还有用来记录关于计算机及其操作信息的各种服务、环境和应用程序日志。

尽管分析和提取 syslogd 日志文件内容的信息可能非常耗时和复杂，但是不能忽略这些日志中信息的价值。syslogd 日志文件可以提供关于潜在问题、错误和安全漏洞等方面的提示，如果使用正确，甚至可以提供关于服务器负载和容量方面的警告。

Linux 系统日志系统一般有 4 个主要的日志子系统：连接时间日志、进程统计日志、错误日志和程序日志。

连接时间日志由多个程序执行，把纪录写入到/var/log/wtmp 和 /var/run/utmp ，login 等 程序更新 wtmp 和 utmp 文 件，使系统管理员能够跟踪谁在何时登录到系统。

utmp、wtmp 和 lastlog 日志文件是多数 UNIX 日志子系统的关键日志文件–保持用户登录进入和退出的纪录。有关当前登录用户的信息记录在文件 utmp 中；登录进入和退出纪录在文件 wtmp 中；最后一次登录文件可以用 lastlog 命令察看。数据交换、关机和重起也记录在 wtmp 文件中。所有的纪录都包含时间戳。这些文件（lastlog 通常不大）在具有大

量用户的系统中增长十分迅速。例如 wtmp 文件可以无限增长，除非定期截取。为了保证系统登录日志的安全防止被篡改，这些日志是二进制的。每次有一个用户登录时， login 程序在文件 lastlog 中察看用户的 UID。如果找到了，则把用户上次登录、退出时间和主机名写到标准输出中，然后 login 程序在 lastlog 中纪录新的登录时间。在新的 lastlog 纪录写入后，utmp 文件打开并插入用户的 utmp 纪录。该纪录一直用 到用户登录退出时删除。utmp 文件被各种命令文件使用，包括 who、w、users 和 finger。下一步，login 程序打开文件 wtmp 附加用户的 utmp 纪录。当用户登录退出时，具有更新时间戳的同一 utmp 纪录附加到文件中。Wtmp 文件被程序 last 和 ac 使用。

进程统计日志由系统内核执行。当一个进程终止时，为每个进程往进程统计文件（ pacct 或 acct ）中写一个纪录。 进程统计的目的是为系统中的基本服务提供命令使用统计。

类 UNIX 可以跟踪每个用户运行的每条命令，如果想知道昨晚弄乱了哪些重要的文件，进程统计日志子系统可以告诉你。它还对跟踪侵入者行为有一定的帮助。与连接时间日志不同，进程统计子系统缺省状态未被激活，我们必须手工启动它。在 Linux 系统中启动进程统计使用 accton 命令，必须用 root 身份运行。Accton 命令的形式 accton file，file 必须先存在。先使用 touch 命令来创建 pacct 文件：touch /var/log/pacct，然后运行 accton： accton /var/log/pacct。一旦 accton 被激活，就可以 使用 lastcomm 命令监测系统中任何时候执行的命令。若要关闭统计，可以使用不带任何参数的 accton 命令。lastcomm 命令报告以前执行的文件。不带参数时，lastcomm 命令显示当前统计文件生命周期内纪录的所有命令的有关信息。包括命令名、用户、tty、命令花费的 CPU 时间和一个时间戳。如果系统有许多用户，输入则可能很长。进程统计的一个问题是 pacct 文件可能增长的十分迅速。这时需要交互式的或经过计划任务机制（ crond ）运行 sa 命令来保持日志数据在系统控制的范围内。sa 命令报告、清理并维护进程统计文件。它能把/var/log/pacct 中的信息压缩到摘要文件/var/ log/savacct 和/var/log/usracct 中，或者生成 sar 自己需要的文件结构。这些摘要包含按命令名和用户名分类的系统统计数据。 sa 缺省情况下先读它们，然后读 pacct 文件，使报告能包含所有的可用信息。sa 的输出有下面一些标记项： avio–每次执行的平均 I/O 操作次数，cp–用户和系统时间总和（以分钟计），cpu–和 cp 一样， k–内核使用的平均 CPU 时间（以 1k 为单位）， k*sec–CPU 存储完整性（以 1k-core 秒）， re–实时时间（以分钟计）， s--系统时间（以分钟计）， tio–I/O

操作的总数， u—用户时间（以分钟计）。

错误日志由 syslogd（8）执行。各种系统守护进程、用户程序和内核通过 syslog（3）向 文件/var/log/messages 报告值得注意的事件。另外有许多程序也创建错误日志，像 HTTP 和 FTP 这样提供网络服务程序也需要保持详细的错误日志。

Syslog 已被许多日志函数采纳，它用在许多保护措施中–任何程序都可以通过 syslog 纪录其事件。Syslog 可以纪录系统事件，可以写到一个文件或设备中，或给用户发送一个信息。它能纪录本地事件或通过网络纪录另一个主机上的事件。Syslog 依据两个重要的文件：syslogd（守护进程）和/etc/syslog.conf 配置文件，多数 syslog 信息被写到 /var/adm 或 /var/log 目录下的信息文件中（messages）。一个典型的 syslog 纪录包括生成程序的名字和一个文本信息。它还包括 一个设备和一个优先级范围。

每个 syslog 消息被赋予下面的主要设备之一：

LOG_AUTH–认证系统：login、su、getty 等

LOG_AUTHPRIV：同 LOG_AUTH，但只登录到所选择的单个用户可读的文件中

LOG_CRON：cron 守护进程

LOG_DAEMON：其他系统守护进程，如 routed

LOG_FTP-：文件传输协议，ftpd、tftpd

LOG_KERN：内核产生的消息

LOG_LPR：系统打印机缓冲池：lpr、lpd

LOG_MAIL：电子邮件系统

LOG_NEWS：网络新闻系统

LOG_SYSLOG：由 syslogd（8）产生的内部消息

LOG_USER：随机用户进程产生的消息

LOG_UUCP：UUCP 子系统

LOG_LOCAL0~LOG_LOCAL7：为本地使用保留

Syslog 为每个事件赋予几个不同的优先级：

LOG_EMERG：紧急情况

LOG_ALERT：应该被立即改正的问题，如系统数据库破坏

LOG_CRIT：重要情况，如硬盘错误

LOG_ERR：错误

LOG_WARNING：警告信息

LOG_NOTICE：不是错误情况，但是可能需要处理

LOG_INFO：情报信息

LOG_DEBUG：包含情报的信息，通常旨在调试一个程序时使用

syslog.conf 文件指明 syslogd 程序纪录日志的行为，该程序在启动时查询配置文件。该文件由不同程序或消息分类的单个条目组成，每个配置占一行。对每类消息提供一个选择域和一个动作域。这些域由 tab 或空格隔开：选择域指明消息的类型和优先级；动作域指明 syslogd 接收到一个与选择标准相匹配的消息时所执行的动作。每个选项是由设备和优先级组成。当指明一个优先级时，syslogd 将纪录一个拥有相同或更高优先级的消息。所以如果指明"crit"，那所有标为 crit、alert 和 emerg 的消息将被纪录。每行的行动域指明当选择域 选择了一个给定消息后应该把他发送到哪儿。

程序日志主要是由第三方程序自己控制的日志。

许多程序通过维护日志来反映系统的安全状态。su 命令允许用户获得另一个用户的权限， 所以它的安全很重要，它的文件为 sulog。同样的还有 sudolog。像 Apache 有两个日志：access_log 和 error_log。

知道系统中正在发生什么事，是我们观察系统状态和处理系统问题的关键。Linux 中提

供了日志系统，并且日志的细节是可配置的。Linux 日志系统中多数日志以明文形式存储，所以用户不需要特殊的工具就可以搜索和阅读它们。同时我们也可以编写脚本，来扫描这些日志，并基于它们的内容去自动执行某些功能。 这些日志默认配置存储在 /var/log 目录中。为了维护系统安全，大多数日志或日志目录设置了仅 root 用户读取和进入。

常用的日志文件如下（在不同的发行版中会有所不同，具体请查看相关配置文件中的定义）：

access-log 纪录 HTTP/web 的传输

acct/pacct 纪录用户命令

aculog 纪录 MODEM 的活动

btmp 纪录失败的纪录

lastlog 纪录最近几次成功登录的事件和最后一次不成功的登录

messages 从 syslog 中记录信息（有的链接到 syslog 文件）

sudolog 纪录使用 sudo 发出的命令

sulog 纪录使用 su 命令的使用

syslog 从 syslog 中记录信息（通常链接到 messages 文件）

utmp 纪录当前登录的每个用户

wtmp 一个用户每次登录进入和退出时间的永久纪录

xferlog 纪录 FTP 会话

第3节　日志更新和备份

1. logrotate 日志更新和备份工具
2. 缺省的 logrotate 配置
3. 使用 include 关键字读取外部配置文件
4. 使用 include 关键字覆盖缺省配置
5. 配置其他 logrotate 参数

logrotate

logrotate 程序是一个日志文件管理工具。用来把旧的日志文件删除或改名，并创建新的日志文件，我们把它叫做“转储”。我们可以根据日志文件的大小，也可以根据其天数来转储，这个过程一般通过计划任务 cron 程序来执行。

logrotate 程序还可以用于压缩日志文件，以及发送日志到指定的 E-mail 。

logrotate 的配置文件是 /etc/logrotate.conf。主要参数如下表：

参数	功能
compress	通过 gzip 压缩转储以后的日志
nocompress	不需要压缩时，用这个参数
copytruncate	用于还在打开中的日志文件，把当前日志备份并截断
nocopytruncate	备份日志文件但是不截断
create mode owner group	转储文件，使用指定的文件模式创建新的日志文件
nocreate	不建立新的日志文件
delaycompress	和 compress 一起使用时，转储的日志文件到下一次转储时才压缩

nodelaycompress	覆盖 delaycompress 选项，转储同时压缩。
errors address	专储时的错误信息发送到指定的 Email 地址
ifempty	即使是空文件也转储，这个是 logrotate 的缺省选项。
notifempty	如果是空文件的话，不转储
mail address	把转储的日志文件发送到指定的 E-mail 地址
nomail 转储时	不发送日志文件
olddir directory	转储后的日志文件放入指定的目录，必须和当前日志文件在同一个文件系统
noolddir	转储后的日志文件和当前日志文件放在同一个目录下
prerotate/endscript 在转储以前需要执行的命令可以放入这个对，这两个关键字必须单独成行	
postrotate/endscript 在转储以后需要执行的命令可以放入这个对，这两个关键字必须单独成行	
daily	指定转储周期为每天
weekly	指定转储周期为每周
monthly	指定转储周期为每月
rotate count	指定日志文件删除之前转储的次数，0 指没有备份，5 指保留 5 个备份
tabootext [+] list 让 logrotate 不转储指定扩展名的文件，缺省的扩展名是：.rpm-orig, .rpmsave, v, 和 ~	
size size	当日志文件到达指定的大小时才转储，Size 可以指定 bytes (缺省)以及 KB (sizek)或者 MB (sizem).

缺省配置 logrotate

logrotate 缺省的配置文件是/etc/logrotate.conf。

Red Hat Enterprise Linux 缺省安装的文件内容是：

```
# see "man logrotate" for details
# rotate log files weekly
weekly
# keep 4 weeks worth of backlogs
rotate 4
# create new (empty) log files after rotating old ones
create
# uncomment this if you want your log files compressed
#compress
# RPM packages drop log rotation information into this directory
include /etc/logrotate.d
# no packages own wtmp – we'll rotate them here
/var/log/wtmp {
    monthly
    create 0664 root utmp
    rotate 1
}
```

```
# system-specific logs may be also be configured here.
```

缺省的配置一般放在 logrotate.conf 文件的最开始处，影响整个系统。

weekly 指定所有的日志文件每周转储一次。

rotate 4 指定转储文件的保留 4 份。

create 指定 logrotate 自动建立新的日志文件，新的日志文件具有和 原来的文件一样的权限。

#compress 指定不压缩转储文件，如果需要压缩，去掉注释就可以了。

include 选项可以读取其他配置文件 ，include 选项允许系统管理员把分散到几个文件的转储信息，集中到一个主要的配置文件。当 logrotate 从 logrotate.conf 读到 include 选项时，会从指定文件读入配置信息，就好像他们已经在/etc/logrotate.conf 中一样。

include /etc/logrotate.d 告诉 logrotate 读入存放在/etc/logrotate.d 目录中的日志转储参数，当系统中安装了 RPM 软件包时，使用 include 选项十分有用。RPM 软件包的日志转储参数一般存放在/etc/logrotate.d 目录。 include 选项十分重要，一些应用把日志转储参数存放在 /etc/logrotate.d 。典型的应用有：apache, linuxconf, samba, cron 以及 syslog。这样，系统管理员只要管理一个 /etc/logrotate.conf 文件就可以了。

使用 include 选项覆盖缺省配置

当 /etc/logrotate.conf 读入文件时，include 指定的文件中的转储参数将覆盖缺省的参数，例如/etc/logrotate.d/psacct 文件：

```
/var/account/pacct {

#prerotate loses accounting records, let's no

#    prerotate

#        /usr/sbin/accton

#    endscript
```

```
    compress
    delaycompress
    notifempty
    daily
    rotate 31
    create 0600 root root
    postrotate
        /usr/sbin/accton /var/account/pacct
    endscript
}
```

在这个例子中，当 /etc/logrotate.d/psacct 文件被读入时，compress 和 rotate 等参数将覆盖/etc/logrotate.conf 中缺省的参数。

为指定的文件配置转储参数

经常需要为指定文件配置参数，一个常见的例子就是每月转储/var/log/wtmp。

设置单独文件参数格式如下：

```
#注释部分
/full/path/of/file
{
option(s)
```

```
}
```

需要注意的是尽管花括号的开头可以和其他文本放在同一行上，但是结尾的花括号必须单独成行。

请跟随老师完成 logrotate 的试验。

第4节　远程集中日志

1. 分散的服务器日志难于统计和分析
2. syslogd 的远程日志汇总方法

当系统管理员需要处理多台服务器或着上万台网络设备的时候，为了避免对多个日志进行排序、合并，系统管理员一般采取的方法是，在设备本地写入日志的同时将日志内容流向一个统一的中心位置，并按照事件到达中心日志服务器的顺序将其记录下来。

syslogd 对这种集中远程日志的支持很简单，配置也比较方便。

中心日志服务器需要配置可以接受外部 syslogd 服务器发送日志。

修改中心日志服务器的/etc/sysconfig/syslog 文件:

```
# vi /etc/sysconfig/syslog
```

将

```
SYSLOGD_OPTIONS="-m 0"
```

修改成

```
SYSLOGD_OPTIONS="-r -m 0"
```

-r 表示启用记录远程主机的日志。

发送日志的服务器配置发送到中心服务器的 IP 地址或主机名。

修改每个发送日志的服务器的/etc/syslog.conf 文件：

```
# vi /etc/syslog.conf
```

加入一行

```
*.* @192.168.0.x
```

每台发送日志服务器都将本机的登录日志记录到 192.168.0.x 这台中心日志服务器上。

完成中心日志服务器和每台日志发送服务器的配置修改后，需要重启它们的 syslog 服务。

```
#service syslog restart
```

然后你就可以在中心日志服务器上测试是否正确的集中日志了。

第5节　WEB 程序日志分析工具

用户访问日志统计系统在 WEB 站点的用户行为分析中扮演了重要的角色，尤其是对于来自搜索引擎的关键词访问统计：是很有效的用户行为分析数据来源。随着互联网多年的发展，WEB 日志统计工具已经越来越成熟，功能也越来越丰富。其中有很多是开放源代码的， AWStats 就是其中非常优秀的一款。 接下来我们将以 AWStats 为例讲解 Apache 日志统计工具的安装和使用。

AWStats（Advanced Web Statistics）是发展很快的一个基于 Perl 的 WEB 日志分析工具。相对于另外一个非常优秀的开放源代码的日志分析工具 Webalizer，AWStats 的优势在于：

界面友好：可以根据浏览器直接调用相应语言界面（有简体中文版） 参考输出样例：http://awstats.sourceforge.net/cgi-bin/awstats.pl 。

基于 Perl：并且很好的解决了跨平台问题，系统本身可以运行在大多数类 UNIX 系统上或 Windows 系统上（安装了 ActivePerl 后）；分析的日志直接支持 Apache 和 Squid 格式 (combined)和 IIS 格式(需要修改)。Webalizer 虽然也有 Windows 平台版，但目前已经缺乏维护； AWStats 完全可以实现用一套系统完成对自身站点不同服务器：GNU/Linux/ Apache 和 Windows/IIS 服务器的统一统计。

效率比较高：AWStats 输出统计项目比 Webalizer 丰富了很多，速度仍可以达到 Webalizer 的 1/3 左右，对于一个日访问量百万级的站点，这个速度都是足够的。

配置/定制方便：系统提供了足够灵活但缺省也很合理的配置规则，需要修改的缺省配置不超过 3，4 项就可以开始运行，而且修改和扩展的插件还是比较多的；

AWStats 的设计者是面向精确的"Human visits"设计的，因此很多搜索引擎的机器人访问都被过滤掉了，因此有可能比其他日志统计工具统计的数字要低，来自公司内部的访问也可以通过 IP 过滤设置过滤掉。

提供了很多扩展的参数统计功能：使用 ExtraXXXX 系列配置生成针对具体应用的参数分析会对产品分析非常有用。

AWStats 运行后将日志统计结果归档到一个 AWStats 的数据库（纯文本）里，然后以两种形式输出：一种是通过 cgi 程序读取统计结果数据库输出;另 一种是运行后台脚本将输

出导出成静态文件 。

数据源日志格式和按天的截断规则

对于 Apache：日志格式好设置：设置成 combined 格式即可，日志截断麻烦一点：需要安装 cronolog 工具，将日志设置成按天截断：

```
CustomLog "|/usr/local/sbin/cronolog /path/to/apache/logs/access_%y%m%d.log" combined
```

比如：logs/access_080808.log　　logs/access_080809.log

配置文件的命名规则是

```
awstats.SITENAME.conf
```

比如你的服务器站点名为 Zhangsan,配置文件就是 awstats.Zhangsan.conf。

AWStats 的主程序 awstats.pl 会自动根据站点名调用相应站点的配置文件：awstats.sitename.conf 比如：运行./awstats.pl -config=uplooking 调用的就是同目录下的 awstats.uplooking.conf 配置文件；如果没有指定-config，还会找当前目录下的 awstats.conf 或者/etc/awstats.conf 作为缺省配置文件。所以最好把缺省的 awstats.model.conf 重命名成 awstats.yoursite.conf；比如：awstats.uplooking.conf ，对于多个站点的统计，AWStats 的配置文件包含功能还是非常有用的，我们可以把通用的配置放在一个文档中，然后用 awstats 5.4 开始支持的 Include 配置将通用配置包含在各个具体配置文件的头部，然后用其他配置覆盖通用配置中的相应属性，比如：

```
Include="uplooking.common.conf"

LogFile="/path/to/bbs/access_log"

SiteName="stuxx.uplooking.com"
```

最少的配置文件修改：LogFile SiteDomain LogFormat ，对于在 GNU/Linux 上统

计 Apache 日志只需修改：LogFile SiteDomain 这 2 个选项：

```
.LogFile="/path/to/apache/logs/access_%YY-24%MM-24%DD-24.log"
```

这个配置的意思是用 24 小时前的年份，月份，日期拼出的日志文件名；

```
SiteDomain="stuxx.uplooking.com"
```

站点的名称，缺省是空的，如果为空，AWStats 将拒绝运行

其他需要注意的事项：AWStats 缺省不过滤 swf 文件，会把.swf 算成 PageView，所以如果站点上 swf 文件主要是广告的话最好还是要过滤掉。

日志分析命令

```
#./awstats.pl -update -config=sitename
```

比如，进入 awstats 安装目录下的 cgi-bin 目录后执行

```
#./awstats.pl -update -config=uplooking
```

查看统计输出，可以使用浏览器访问如下 URL

```
http://localhost/cgi-bin/awstats.pl?config=uplooking
```

配置 AWStats 日志统计自动运行,一般使用计划任务 cron:

#crontab -e

设置每天的 4 点 45 分执行日志统计：

```
45 4 * * * (cd /full/path/to/awstats/cgi-bin/; ./awstats.pl  - update -config=uplooking)
```

多站点日志统计

AWStats 自带了一个批处理工具：tools/awstats_updateall.pl，可以批量地遍历一个目录下所有地配置文件并运行统计。因此剩下的工作就主要是日志的同步问题了。针对多个站点，很多配置选项是重复的，如果每个配置文件都修改维护起来会很麻烦，AWStats 从 5.4 开始提供了配置文件包含的功能，所以我们可以配置一个通用配置，比如：uplooking.common.conf ,然后其他站点的配置设置为：

```
#cat awstats.bbs.uplooking.conf

Include "uplooking.common.conf"

LogFile "/path/to/bbs_log"

SiteName "bbs.uplooking.com"



#cat awstats.www.uplooking.conf

Include "uplooking.common.conf"

LogFile "/path/to/www_log"

SiteName "www.uplooking.com"

HostAliases="uplooking.com"
```

统计指标说明

- 参观者：按来访者不重复的 IP 统计，一个 IP 代表一个参观者；
- 参观次数：一个参观者可能 1 天之内参观多次（比如：上午一次，下午一次），所以按一定时间内（比如：1 个小时），不重复的 IP 数统计，参观者的访问次数；
- 网页数：不包括图片，CSS, JavaScript 文件等的纯页面访问总数，但如果一个页面使用了多个帧，每个帧都算一个页面请求；
- 文件数：来自浏览器客户端的文件请求总数，包括图片，CSS，JavaScript 等，用户请求一个页面是，如果页面中包含图片等，所以对服务器会发出多次文件请求，

文件数一般远远大于文件数；

- 字节：传给客户端的数据总流量；
- 来自 REFERER 中的数据：日志中的参考（REFERER）字段，记录了访问相应网页之前地址，因此如果用户是通过搜索引擎的搜索结果点击进入网站的，日志中就会有用户在相应搜索引擎的查询地址，这个地址中就可以通过解析将用户查询使用的关键词提取出来。

AWStats 在搜索引擎的关键短语和关键词统计方面的功能还是比较完整的：可以对全世界 3 百多种机器爬虫进行识别，并且可以识别大部分主流国际化搜索引擎和很多地区的本地语言搜索引擎。

第6节　安装 AWStats

首先下载 awstats 软件包。

你可以到 AWSTATS 的官方站点下载它的 tar.gz 包，也可以下载 rpm 包，此试验中使用 tar 包！ 选择 tar 包的主要原因是:awstats rpm 包需要依赖很多系统不自带的 perl rpm。

官方站点下载位置 http://prdownloads.sourceforge.net/awstats

安装 awstats tar 包。

```
# tar -zxvf awstats-*.tar.gz -C /var/www/html/

#cd /var/www/html

#mv awstats* awstats
```

现在已经把 AWStats 安装到 /var/www/html/awstats/中了，awstats 有提供给 Apache 的辅助设置文件/var/www/html/awstats/tools/httpd_conf，将这个辅助设置文件 copy 到 /etc/httpd/conf.d/ 目录下，并且顺便改名改成 awstats.conf。

```
#cp /var/www/html/awstats/tools/httpd_conf
/etc/httpd/conf.d/awstats.conf

#cat /etc/httpd/conf.d/awstats.conf
```

AWStats 默认是需要安装在/usr/local 目录下的，所以 awstats.conf 的配置我们需要作少许改动。

```
#sed -i 's#/usr/local#/var/www/html#g' /etc/httpd/conf.d/awstats.conf
```

现在重启 httpd 服务设置就生效了。

```
# service httpd reload
```

接下来配置 awstats

在/var/www/html/awstats/wwwroot/cgi-bin 底下有一个例子配置文件 awstats.model.conf ，我们需要将其拷贝到/etc/awstats 目录下，并改名为我们需要的配置文件名。

```
#mkdir /etc/awstats

# cd /etc/awstats/

# cp  /var/www/html/awstats/wwwroot/cgi-bin/awstats.modul.conf awstats.stuxx.conf
```

请将 stuxx 改为你的主机名，并编辑这个文件。

```
# vi awstats.smallken.conf
```

需要修改的配置项：

```
// Apache log 档位置

LogFile="/var/log/httpd/access_log"

// 主机名称，没有 Domain Name 时设 IP 也没问题

SiteDomain="stuxx.uplooking.com"

// 执行 perl 的目录

DirCgi="/awstats"

// 小图示的目录

DirIcons="/awstatsicons"

// 语修改系
```

```
Lang="cn"
```

存盘退出，现在我们手动来执行分析脚本

```
# cd /var/www/html/awstats/wwwroot/cgi-bin/

# perl awstats.pl -config=stuxx -update
```

打开浏览器查看生成页面

```
firefox 访问 http://localhost/awstats/awstats.pl?config=stuxx
```

接著我们想让它每 5 分钟自己更新一次

编辑一个脚本 awstats.sh

vi /root/awstats.sh

输入

```
#!/bin/bash

cd /var/www/html/awstats/wwwroot/cgi-bin/

perl awstats.pl -config=stuxx -update
```

赋予它可执行权限

```
# chmod +x /root/awstats.sh
```

加入 cron，让它每 5 分钟执行 aswstats.sh 一次

```
# cronteb -e
```

输入

```
*/5 * * * * /root/awstats.sh
```

然后你就狂刷你的页面，等 5 分钟后再查看页面

```
http://localhost/awstats/awstats.pl?config=stuxx
```