NSS

Log Analysis

Logs

Logs are records of events and actions on a computer.

Why are they useful

Logs allow developers to see what is happening on the server while specific processes are being run.

Examination of Site traffic

Error logs

Detecting data exfiltration

Server/SQL Performance

Operating System Logging

Operating system logging (syslog)-

The system log file contains events that are logged by the operating system components. These events are often predetermined by the operating system itself. System log files may contain information about device changes, device drivers, system changes, events, operations and more.

Logging in OSX

Open the Console:

Application/Utilities/Console

Take a moment to look at this application

Linux Logging

var/log/ -typical logging repository

Apache Access Log File

Apache server records all incoming requests and all requests processed to a log file. Default apache access log file location:

- RHEL / Red Hat / CentOS / Fedora Linux Apache access file location - **/var/log/httpd/access_log**
- Debian / Ubuntu Linux Apache access log file location - **/var/log/apache2/access.log**
- FreeBSD Apache access log file location - **/var/log/httpd-access.log**

Apache Error Log File

All apache errors / diagnostic information other errors found during serving requests are logged to this file. Location of error log is set using ErrorLog directive. If there is any problem, you should first take a look at this file using cat, grep or any other UNIX / Linux text utilities. This apache log file often contain details of what **went wrong and how to fix it**. Default error log file location:

- RHEL / Red Hat / CentOS / Fedora Linux Apache error file location -**/var/log/httpd/error_log**
- Debian / Ubuntu Linux Apache error log file location - **/var/log/apache2/error.log**
- FreeBSD Apache error log file location - **/var/log/httpd-error.log**

Formatting Apache Logs

The format of the access log is highly configurable.

The location and content of the access log are controlled by the **CustomLog** directive.

Log Analysis documentation

http://httpd.apache.org/docs/current/logs.html

https://developer.apple.com/library/mac/documentation/macosx/conceptual/bpsystemstartup/chapters/LoggingErrorsAndWarnings.html

Logging format

The format for logs should comply with the W3C standards for logs which can be found here (http://www.w3.org/TR/WD-logfile.html).

There are 3 types of log formats: Common, Extended & Custom.

Common Log example

`127.0.0.1 user-identifier frank [10/Oct/2000:13:55:36 -0700] "GET /apache_pb.gif HTTP/1.0" 200 2326`

- *127.0.0.1* is the IP address of the client (remote host) which made the request to the server.
- *user-identifier* is the RFC 1413 identity of the client.
- *frank* is the userid of the person requesting the document.
- *[10/Oct/2000:13:55:36 -0700]* is the date, time, and time zone when the server finished processing the request, by default in strftime format*%d/%b/%Y:%H:%M:%S %z*
- *"GET /apache_pb.gif HTTP/1.0"* is the request line from the client. The method *GET*, */apache_pb.gif* the resource requested, and *HTTP/1.0*the HTTP protocol.
- *200* is the HTTP status code returned to the client. 2xx is a successful response, 3xx a redirection, 4xx a client error, and 5xx a server error.
- *2326* is the size of the object returned to the client, measured in bytes.

Extended Example

```
#Version: 1.0
#Date: 12-Jan-1996 00:00:00
#Fields: time cs-method cs-uri
00:34:23 GET /foo/bar.html
12:21:16 GET /foo/bar.html
12:45:52 GET /foo/bar.html
12:57:34 GET /foo/bar.html
```

Lines beginning with the # character contain directives. The following directives are defined:

Version: <*integer*>.<*integer*> The version of the extended log file format used. This draft defines version 1.0.

Fields: [<*specifier*>...] Specifies the fields recorded in the log.

Software: *string* Identifies the software which generated the log.

Start-Date: <*date*> <*time*> The date and time at which the log was started.

End-Date:<*date*> <*time*> The date and time at which the log was finished.

Date:<*date*> <*time*> The date and time at which the entry was added.

Remark: <*text*> Comment information. Data recorded in this field should be ignored by analysis tools.

Logging levels

There are various levels of information which can be addressed within your logging plan.

Log Analysis Tools

http://www.debianhelp.co.uk/webalizer.htm

Terminal commands

- vi/vim text editor
- tail -f for watching logs in real time
- less text viewer
- The grep command

Middleware logs

Python, Tomcat, PHP, and ColdFusion may have their own logging. You must be weary of this fact when looking for events.

For example /var/log/apache/php.errors

HTTP Error codes

http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html

Common Error codes

100-Informational

200-Success

300-Redirection

400-Client Error

500-Server Error

100

100 Continue

101 Switching Protocols

102 Processing

200 Class

200 OK

201 Created

202 Accepted

300 Class

300- Multiple Choices

301- Moved Permanently

302- Found

308- Permanent Redirect

400 Class

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found (know this one)

408 Request Timeout

500 Class

500 Internal Server Error

501 Not Implemented

502 Bad Gateway

507 Insufficient Storage