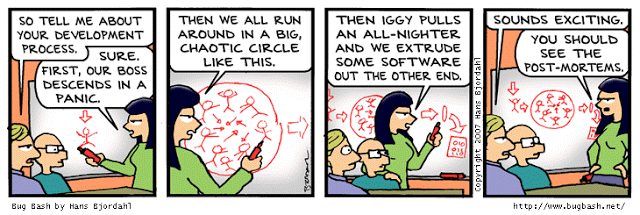
**Postmortem Report**

**504 Error while accessing a given URL**

[](https://raw.githubusercontent.com/MitaliSengupta/holberton-system_engineering-devops/master/0x19-postmortem/image.gif)

**Incident report for**[**504 error / Site Outage**](https://github.com/MitaliSengupta/holberton-system_engineering-devops/tree/master/0x17-web_stack_debugging_3)

**Summary**

On September 11th, 2018 at midnight the server access went down resulting in 504 error for anyone trying to access a website. Background on the server being based on a LAMP stack.

**Timeline**

* **00:00 PST** - 500 error for anyone trying to access the website
* **00:05 PST** - Ensuring Apache and MySQL are up and running.
* **00:10 PST** - The website was not loading properly which on background check revealed that the server was working properly as well as the database.
* **00:12 PST** - After quick restart to Apache server returned a status of 200 and OK while trying to curl the website.
* **00:18 PST** - Reviewing error logs to check where the error might be coming from.
* **00:25 PST** - Check /var/log to see that the Apache server was being prematurely shut down. The error log for PHP were nowhere to be found.
* **00:30 PST** - Checking php.ini settings revealed all error logging had been turned off. Turning the error logging on.
* **00:32 PST** - Restarting apache server and going to the error logs to check what is being logged into the php error logs.
* **00:36 PST** - Reviewing error logs for php revealed a mistyped file name which was resulting in incorrect loading and premature closing of apache.
* **00:38 PST** - Fixing file name and restarting Apache server.
* **00:40 PST** - Server is now running normally and the website is loading properly.

**Root Cause and Resolution**

The issue was connected with a wrong file name being reffered to in the wp-settings.php file. The error was raised when trying to curl the server, wherein the server responded with 500 error. By checking the error logs it was found that no error log file was being created for the php errors and reading the default error log for apache did not result in much information regarding the premature closing of the server. Once understood that the errors for php logs were not being directed anywhere the engineer chose to review the error log setting for the php in the php.ini file and found that all error logging was turned off. Once turned on, the error logging the apache server was restarted to check if any errors were being registerd in the log. As suspected, the php log showed that a file with a .phpp extension was not found in the wp-settings.php file. This was clearly a misspelled error that resulted in the error to site access. As this was one server that the error was found in, this error might have been replicated in other servers as well. An easy fix by changing the file extension by puppet would result in the fix being made to other servers as well. A quick deployment of the puppet code replaced all misspelled file extensions with the right one and restarting of the server resulted in properly loading of the site and server.

**Corrective and Preventive Measures**

* All servers and sites should have error logging turned on to easily identify errors if anything goes wrong.
* All servers and sites should be tested locally before deploying on a multi-server setup this will result in correcting errors before going live resulting in less fixing time if site goes down.