Daily Assessement format

Date: 13/06/2020

Name: Jyothi S Donne

USN: 4AL17EC037

Course: MYSQL

Topic: MYSQL JOINS

- MYSQL join type-ex
- MYSQL join diagrams
- creating nested lists
- MYSQL nested processing
- styling our lists

Github repository: Jyothi-courses

forenoon session details

Report

Creating a custom error handler

It is a quite simple. we simply create a special function. that can be called when an error occurs in PHP. this function must be able to handle a min of two parameters. but can accept upto five parameters

syntax

error_function(error_level, error_msg, error_file, error_line, error_context)

Types of erros

Basically there are four types errors in PHP, which are as follows

→ **parse error**

It occurs if there is a syntax mistkake in the script; the output in parse errors. stops the execution of the script there are many reasons for the occurance of parse error in PHP. the common reasons for errors are as follows

- unclosed quotes
- missing or extra pasentheses
- uncloced braces
- missing semicolon

ex:
```
<?php
Echo "cat";
Echo "Lion";
?>
```

→ fatal errors

this errors are caused when php understands what you've written, however what you're asking it to do can't be done. fatal errors stop the execution of the script. if you are trying to acess the undefined functions, then the output is fatal error

ex:
```
<?php
function fun1()
{ Echo "vineeth";
}
fun2();
Echo "fatal error!";
?>
```

→ warning errors

it will not stop execution of the script the main reason for warning errors are to include a missing file or using the incorrect number of parameters in a fuc

ex:
```
<?php
echo
"warning error!";
include
("welcome.php");
?>
```

→ notice errors

notice that an error is the same as a warning error i.e in the notice error execution of the script doesn't stop notice that the error occurs when you try to access the undefined variable, then produce a notice error

ex

```
<?php $a="vineet";
Echo "notice error!!";
Echo $b;
?>
```

→ Create a user defined func in php

syntax

```
function functionname(){
 code to be executed;
}
```

→ php security

with php security, there are two sides to error reporting. one is beneficial to increasing security, the other is detrimental. Regardless of the method of error handling the ability to probe a system for errors leads to providing an attacker with more information.