1 Curso POO PHP Configuración segura de PHP

1.1 Sumario

1.2 Configuración segura de PHP

Na configuración de PHP temos algunhas directivas que nos permiten restrinxir certas características da linguaxe para aumentar a súa seguridade:

1.2.1 Deshabilitar funcións e clases

As directivas **disable_functions** e **disable_clases** permiten deshabilitar funcións que consideremos que poderían supor un risco de seguridade.

disable_functions recibe como parámetro unha lista separada por comas cos nomes das funcións a deshabilitar. Por exemplo:

```
disable_functions = exec,passthru,shell_exec,system,proc_open,popen,curl_exec,curl_multi_exec,parse_ini_file,show_source
```

disable_classes é similar, pero recibe como parámetro unha lista de nomes de clases.

1.2.2 Restrinxir a execución de código remoto

Un potencial risco de seguridade é a execución no noso servidor de código remoto. Isto podería facerse empregando include, include_once, require ou require_once indicando como destino unha URL. Pódese desahabilitar a utilización de URLs nas sentenzas anteriores empregando a directiva **allow_url_include** (xa ven deshabilitado na configuración por defecto de PHP).

```
allow_url_include = Off
```

Outra posibilidade é empregar **fopen** pasando como parámetro unha URL para abrir un recurso externo. A directiva **allow_url_fopen** desactiva a utilización de envolturas de tipo URL, como http:// ou ftp://, o que afecta tamén ao comportamento das sentenzas include, include_once, require e require_once (ven habilitado na configuración por defecto de PHP).

```
allow_url_fopen = Off
```

1.2.3 Minimizar a información dispoñible aos usuarios

No apartado de xestión de erros xa se viron duas directivas de configuración que afectan á información que reciben os usuarios cando se produce algún tipo de erro na execución dos scripts: **error_reporting** e **display_errors**.

Además existe outra directiva, **expose_php**, que convén axustar para evitar dar publicidade ao feito de que PHP se atopa instalado no servidor.

```
error_reporting = E_ALL & ~E_DEPRECATED & ~E_STRICT
display_errors = Off
expose_php = Off
```

1.2.4 Control de recursos

É unha boa práctica limitar os recursos que consumen os nosos scripts, pensando en evitar posibles ataques de denegación de servizo (DOS) ou as consecuencias non desexadas de fallos na programación. En PHP podemos empregar, entre outras, as seguintes directivas:

- **max_execution_time**. Tempo máximo en segundos que pode estar executándose un script.
- **max_input_time**. Tempo máximo en segundos que o script pode estar analizando os datos recibidos, como os que proveñen dos

formularios.
- **memory_limit**. Límite de memoria en bytes que pode consumir un script.

```
max_execution_time = 30
max_input_time = 60
memory_limit = 128M
```

1.2.5 Control do acceso ao sistema de ficheiros

Tamén é posible limitar as rutas ás que pode acceder PHP para ler ou escribir ficheiros. Faise definindo a directiva **open_basedir**. O seu valor é unha cadea de texto coas rutas ás que se permite o acceso (separadas unha da outra polo caracter ":", ou por ";" en sistemas Windows).

```
open_basedir = "/xampp/php/tmp;/xampp/htdocs/execicios"
```

1.2.6 Subida de ficheiros

Por defecto PHP permite que se empregue HTTP para subir ficheiros ao servidor. Se non imos a empregar esta característica nos nosos scripts, convén desactivala mediante a directiva **file_uploads**. Outras directivas como **upload_max_filesize** ou **max_file_uploads** permiten por límites ás subidas de ficheiros no caso de que estean permitidas.

Para desactivar as subidas de ficheiros:

```
file_uploads = Off
```

Para activalas con certos límites:

```
file_uploads = On
upload_max_filesize = 2M
max_file_uploads = 5
```

Tamén é posible definir coa directiva **post_max_size** a cantidade máxima de datos a enviar nunha mensaxe POST de HTTP.

```
post_max_size = 2M
```

1.2.7 Características de seguridade obsoletas

- **Modo seguro**. Nas versións de PHP anteriores á 5.4, existía unha directiva de configuración chamada **safe_mode**, que limitaba a utilización de algunhas funcións da linguaxe consideradas potencialmente inseguras. Esta directiva pasou a considerarse obsoleta en PHP5.3, desaparecendo na seguinte versión, 5.4. A súa utilización hoxe en día xera un erro fatal na execución do código.
- **Comiñas máxicas**. Tamén ata a versión PHP5.4 estivo activa a directiva **magic_quotes**, que cando estaba habilitado limpiaba de forma automática os datos de entrada que recollían os formularios dos scripts.
- **Variables globais automáticas** . E nesa mesma versión de PHP, a 5.4, eliminouse tamén definitivamente da configuración, por razóns de seguridade, a directiva **register_globals**, que introducía nos scripts como variables globais todo tipo de información, como as variables que se obteñen dos formularios.

--Víctor Lourido 14:24 18 jul 2013 (CEST)