

Cambios especificados en el php.ini disable_functions por dominio

Cambiar individualmente en un dominio disable_functions

Un escenario muy común en sistemas, es la salida hacia adelante obviando los problemas que puede causar nuestra decisión.

Un caso común es el quitar las protecciones de forma global a través de el proceso general del servidor, en lugar de buscar una solución individual, para el caso concreto.

En el caso de Directadmin, si tenemos la opción [secure_php](#) esta añadirá a la configuración del php del servidor

```
disable_functions =  
exec,system,passthru,shell_exec,proc_close,proc_open,dl,popen,show_source,posix_kill,posix_mkfifo,posix_getp  
wuid,posix_setpgid,posix_setsid,posix_setuid,posix_setgid,posix_seteuid,posix_setegid,posix_uname  
mysqli.allow_local_infile = Off  
expose_php = Off  
register_globals = Off
```

Existe el camino fácil, que es ir quitando las funciones que molestan a nuestros clientes, [How to customize the disable_functions list](#) y existe otro camino, que es deshabilitar al que nuestro cliente necesita, si esto es totalmente necesario, ya sea por un tiempo limitado, o de forma definitiva.

Opciones de php.ini por dominio

[Opciones por dominio](#) nos ayuda a esto último ya que tratar de modificar `disable_functions` via `.user_ini` aunque este autorizado, no funcionará para modificar esa clave.

- Lo primero es buscar el path de nuestra instalación para nuestro Directadmin y nuestra distro.

En mi caso esta en `/usr/local/phpXX/lib/php.conf.d`

Así que, siguiendo el manual, introducido el path y el dominio o subdominio del que quiero tal corrección, el nos lo dará.

En nuestro caso el cliente tenia algunos despliegues con Laravel que necesitaban `shell_exec` por lo que lo eliminamos del fichero **ini** personalizado,

```
/usr/local/php83/lib/php.conf.d/30-subdomain.full.tld.ini
```

```
disable_functions =  
exec,system,passthru,proc_close,proc_open,dl,popen,show_source,posix_kill,posix_mkfifo,posix_getpuid,posix_setpgid,posix_setsid,posix_setuid,posix_setgid,posix_seteuid,posix_setegid,posix_uname
```

Después simplemente tenemos que hacer un rebuild de la función de seguridad.

da build secure_php

Aviso

Esta documentación y su contenido, no implica que funcione en tu caso o determinados casos.

También implica que tienes conocimientos sobre lo que trata, y que en cualquier caso tienes copias de seguridad.

El contenido el contenido se entrega, tal y como está, sin que ello implique ningún obligación ni responsabilidad por parte de [Castris](<https://castris.com>)

Si necesitas soporte profesional puedes contratar con Castris [soporte profesional](<https://intranet.castris.com/store/soporte-profesional>).

Revision #2

Created 16 December 2024 16:41:28 by Abkrim

Updated 7 January 2025 06:20:26 by Abkrim