论PHP环境加固的知识

Sec盾 渗透云笔记 2月2日

文章来源于；Sec盾

随着php用的越来越多，安全问题也变得更为重要，php环境提供的安全模式是个非常重要的内嵌的安全机制，能够控制一些php中的函数，比如system()，同时把很多文件操作函数进行了权限控制，也不允许对某些关键字文件的文件，比如/etc/passwd，但是默认的php.ini是没有打开安全模式的。下面讲解如何使用php的安全功能来保护网站的安全性：

一.启用php的安全模式

php的安全模式是个非常重要的内嵌的安全机制，能够控制一些php中的函数，比如system()，同时把很多文件操作函数进行了权限控制，也不允许对某些关键字文件的文件，比如/etc/passwd，但是默认的php.ini是没有打开安全模式的，我们可以把它打开：

```
1. safe_mode = on
```

二.用户组安全

当safe_mode打开时，safe_mode_gid被关闭，那么php脚本能够对文件进行访问，而且相同组的用户也能够对文件进行访问。而且相同组的用户也能够对文件进行访问。建议设置为：

```
1. safe_mode_gid = off
```

如果不进行设置，可能我们无法对我们服务器网站目录下的文件进行操作了，比如我们需要对文件进行操作的时候。

三.安全模式下执行程序主目录

如果安全模式打开了，但是却是要执行某些程序的时候，可以指定要执行程序的主目录：

```
1. safe_mode_exec_dir = /usr/bin
```

一般情况下是不需要执行什么程序的，所以推荐不要执行系统程序目录，可以指向一个目录：然后把需要执行的程序拷贝过去，比如：

```
1. safe_mode_exec_dir = /temp/cmd
```

但是，我更推荐不要执行任何程序，那么就可以指向我们网页目录：

```
1. safe_mode_exec_dir = /usr/www
```

四.安全模式下包含文件

如果要在安全模式下包含某些公共文件，那么就修改一下选项：

```
1. safe_mode_include_dir = /usr/www/include/
```

其实一般php脚本中包含文件都是在程序自己已经写好了，这个可以根据具体需要设置。

五.控制php脚本能访问的目录

使用open_basedir选项能够控制PHP脚本只能访问指定的目录，这样能够避免PHP脚本访问不应该访问的文件，一定程序上显示了phpshell的危害，我们一般可以设置为只能访问网站目录：

```
open_basedir = /usr/www
```

六.关闭危险函数

如果打开了安全模式，那么函数禁止是可以不需要的，但是我们为了安全还是考虑进去。比如，我们觉得不希望执行包括system()等在内的执行明了的php函数，或者能够查看php信息的phpinfo()等函数，那么我们就可以禁止它们：

```
1. disable_functions= system, passthru, exec, shell_exec, popen, phpinf
   o, escapeshellarg, escapeshellcmd, proc_close, proc_open, dl
```

如果你要禁止任何文件和目录的操作，那么可以关闭很多文件操作

```
1. disable_functions= chdir, chroot, dir, getcwd, opendir, readdir, sca
   ndir, fopen, unlink, delete, copy, mkdir, rmdir, rename, file, file_get
   _contents, fputs, fwrite, chgrp,chmod, chown
```

以上只是列了部分比较常用的文件处理函数，你也可以把上面执行命令函数和这个函数结合，就能给抵制大部分的phpshell了。

七.关闭php版本信息在http头中的泄露

我们为了防止黑客获取服务器中php版本的信息，可以关闭该信息泄露在http头中：

```
1. expose_php = off
```

比如黑客在 telnet domain 80 的时候，那么将无法看到PHP的信息

八.关闭注册全局变量

在PHP中提交的变量，包括使用POST或者GET提交的变量，都将自动注册为全局变量，能够直接访问，这是对服务器非常不安全的，所以我们不能让它注册为全局变量，就把注册全局变量选项关闭：

```
1. register_globals = off
```

当然，如果这样设置了，那么获取对应变量的时候就要采取合理方式，比如获取GET提交的变量var，那么就要用$_GET['var']来进行获取，这个php程序员要注意。

九.SQL注入防护

SQL注入是非常危险的问题，小则网站后台被入侵，重则整个服务器沦陷，所以一定要小心。php.ini中有一个设置：

```
1. magic_quotes_gpc = off
```

这个默认是关闭的，如果它打开后将自动把用户提交对sql的查询进行转换，比如把'转为\'等，这对防止sql注入有很大作用，所以我们推荐设置为：

```
magic_quotes_gpc = off
```

十.错误信息控制

一般php在没有连接到数据库或者其他情况下会有错误提示，一般错误信息中会包含php脚本当前的路径信息或者查询的SQL语句等信息，这类信息提供给黑客后，是不安全的，所以一般服务器建议禁止错误提示：

```
1. display_errors = Off
```

如果你确实要显示错误信息，一定要设置显示错误的级别，比如只显示警告以上的信息：

```
1. error_reporting = E_WARNING & E_ERROR
```

当然，我还是建议关闭错误提示。

十一.错误日志

建议在关闭display_errors后能够把是错误信息记录下来，便于排查服务器运行的原因：

```
1. log_errors = On
```

同时也要设置错误日志存放的目录，建议根apache的日志存在一起：

```
1. error_log = /usr/local/apache2/logs/php_error.log
```

注意：给文件必须允许apache用户或组具有写的权限。