偶然下载了今年 ISC 大会 360 应急响应中心的一个 ppt，在最后有个攻防领域专家注册考试目录，其中有很大一块就是中间件的安全，包括 Apache、IIS、Tomcat、Weblogic 等等，后面我会针对这些中间件，并且借着上面的考试要求进行一个安全配置的讲解。(自己之前也很少接触这块，因此只能算一个学习记录了。。)

Apache 方面的考核内容：

1.Apache 服务器权限配置

2.Apache 服务器文件解析漏洞

3.Apache 服务器日志文件审计方法

4.Apache 服务器 WEB 目录权限的配置

下面我将针对上面几个要求，进行一个实例讲演！

问题 1：Apache 服务器权限配置

这里的权限配置指的就是在访问 Apache 服务器时限制哪些 ip，允许哪些 ip 来访问。

配置文件在 Apache\conf 目录下的 httpd.conf 文件中

```
DocumentRoot  "C:\phpmystudy\WWW"
<Directory />
    Options +Indexes +FollowSymLinks +ExecCGI
    AllowOverride All
    Order allow,deny
    Allow from all
    Require all granted
```

</Directory>

这里是初始状态，最重要的就是 Order allow,deny 下面的配置，可以看到我们这里的配置为 Allow from all，因此允许任何人来访问我们的 web 服务器，但是我们修改成只允许 10.10 开头的 ip 访问，修改如下

Order allow,deny

Allow from 10.10

我们可以看到我们访问服务器时就会出现访问禁止，实现了权限的配置。

Order allow,deny

Allow from 192.168

如果想让内网实现访问，这里只允许内网网段的 ip 访问

但是实际上，这里没有添加 127.0.0.1 网段的，会导致我们的 web 主机登录不上 web 服务

下面作一个演示，我们只想让 web 主机登录，其他主机都登陆不上

Order allow,deny

Allow from 127.0

Deny from all

这里只有 127.0.0.1 的主机能访问，其余主机均登录不上，Apache 服务器权限配置就告一段落。。

问题 2：Apache 服务器文件解析漏洞

这个应该很老套了，就是一个解析漏洞。

Apache 对于文件名的解析是从后往前解析的，直到遇见一个它认识的文件类型为止。因此，如果 web 目录下存在以类似 webshell.php.test 这样格式命名的文件,Apache 在解析时因为不认识.test 这个文件类型，所以会一直往前解析，当解析到.php 时，它认识了，因此会将它解析为 PHP 文件。

实际上我在复现的时候没有成功利用，原因很简单，目前这个解析漏洞只适用于以 module 方式解析 php 的 apache，使用 fastcgi 方式解析 php 的 apache 不受影响。而我测试使用的 phpmystudy 正是使用的 fastcgi 方式。

防御方法：

apache 配置文件，禁止.php.这样的文件执行，配置文件里面加入

<Files ~ "\.(php.|php3.)">

Order Allow,Deny

Deny from all

</Files>

问题 3：Apache 服务器日志文件审计方法

这里主要有两个 log 文件，一个是 access.log，一个是 error.log，这是在 windows 环境下，其实一开始由于配置问题，没有开启 access.log，配置同样是在 Apache\conf 目录下的 httpd.conf 文件中

##CustomLog "logs/access.log" common

...

#CustomLog "logs/access.log" combined

将前面的注释符删去即可~

然后就会生成 access.log

::1 - - [07/Nov/2017:15:23:55 +0800] "GET / HTTP/1.1" 200 369

::1 - - [07/Nov/2017:15:23:55 +0800] "GET / HTTP/1.1" 200 369 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:56.0) Gecko/20100101 Firefox/56.0"

这里给出一张截图，是我网站服务器的 Apache 日志

一般情况下，我们都会写一个自定义的脚本来审计日志文件或者说下载市面上拥有成熟技术的日志审计软件。假如这是一台 linux 服务器，日志文件中出现了/etc/passwd 的字样，就可以在一定程度上说明有人已经可以访问你的密钥文件，也就反映了你的网站可能已经被人攻破了，或者说出现了大量的 sql 注入语句，这些黑客行为都可以帮助你来审计一个网站的安全性。

而 error.log 则是记录了服务器运行时的出错情况

[Tue Nov 07 15:23:33.541825 2017] [mpm_winnt:notice] [pid 3232:tid 636] AH00455: Apache/2.4.23 (Win32) OpenSSL/1.0.2j PHP/5.4.45 configured -- resuming normal operations

[Tue Nov 07 15:23:33.542826 2017] [mpm_winnt:notice] [pid 3232:tid 636] AH00456: Server built: Jul 1 2016 16:42:20

[Tue Nov 07 15:23:33.542826 2017] [core:notice] [pid 3232:tid 636] AH00094: Command line: 'D:\\Server\\phpstudy\\Apache\\bin\\httpd.exe -d D:/Server/phpstudy/Apache'

[Tue Nov 07 15:23:33.548828 2017] [mpm_winnt:notice] [pid 3232:tid 636] AH00418: Parent: Created child process 8984

[Tue Nov 07 15:23:35.457435 2017] [mpm_winnt:notice] [pid 8984:tid 676] AH00354: Child: Starting 150 worker threads.

问题 4：Apache 服务器 WEB 目录权限的配置

在挖洞过程中，时常会出现目录泄露这样的漏洞，其实此类漏洞就是因为中间件的配置问题而造成的。

能够进行目录的遍历，这的确会造成很多的安全问题，那么如何进行 web 目录权限的配置呢？

其实这里跟第一个问题是类似的，第一个问题是对服务器的权限配置，这里是对某个目录配置访问限制，同样是在 Apache\conf 目录下的 httpd.comf 文件中添加对目录的权限配置

```
<Directory "C:\phpmystudy\WWW\test">
    Order allow,deny
    deny from all
</Directory>
```

这里 test 目录设置成任何人都不允许访问

这样也就避免了目录的泄露问题。

灵光一闪

最后突然想到一个问题，如何禁止某个目录运行 php 脚本文件，这种情形现在来说应该仍然十分受用，假如存在文件上传漏洞，那么通过这种方法禁止了脚本文件的运行，从而达到保护服务器的作用。

方法非常多，例如禁止当前目录访问，这里给出一种直接的方法，那就是禁止 php 脚本的执行

这里模拟了下文件上传，然后我们通过文件任意上传漏洞上传了我们的 php 脚本，发现是可以执行的，然后挂菜刀，传大马继续。。

这里的防御方法是给 Apache\conf 目录下的 vhosts.conf 添加如下信息

```
<Directory "C:\phpmystudy\WWW\upload">
php_flag engine off
</Directory>
```

这里即表示当前目录下关闭 php 脚本执行功能