2011 年 11 月

目录

来自编辑的序言

Deepak Subramanian

OWASP通讯简报正在改版。改版的目的是使通讯简报更成熟和更负责任。目前，我、Tom Halleway和Kate Hartman正负责进行改版的工作。我们希望OWASP的项目负责人们，通过告诉我们各项目的最新进展，来帮助我们进行改版工作。我们也想借此机会，欢迎所有安全爱好者们提供各种有关安全的文章和有趣的信息。

OWASP通讯简报希望刊登更多的研究类型文章。我们以极大的热情欢迎研究类型文章的投稿。

另外，我们感谢任何旨在将通讯简报做得更好的建议！

Email: deepak.subramanian@owasp.org

中文翻译：王颉　审核：高雯，张平

内部项目的信息

OWASP 项目委员会

OWASP 项目是 OWASP 基金会的一个内部部分。一些开发的工具和项目因频繁的版本发布和评估而被经常更新。以下提供的信息由 OWASP 项目委员会在 2011 年 7 月底的时候更新。OWASP 新闻简报将为你提供有关这些项目的更多详细信息。

新项目

OWASP Web 应用安全可接入性项目，PetrZávodský。
https://www.owasp.org/index.php/OWASP_Web_Application_Security_Accessibility_Project

OWASP Cloud – 10 项目，VinayBansal, , Shankar BabuChebrolu，PankajTelang，Ken Huang 和 Ove Hansen。
https://www.owasp.org/index.php/Category:OWASP_Cloud_-_10_Project

OWASP Web 测试环境项目，Matt Tesauro。
https://www.owasp.org/index.php/Projects/OWASP_Web_Testing_Environment_Project

OWASP iGoat项目，Kenneth R. van Wyk。
https://www.owasp.org/index.php/OWASP_iGoat_Project

Opa项目，Henri Binsztok和Adam Koprowski。
https://www.owasp.org/index.php/Opa

OWASP移动安全项目-移动威胁模型，项目领导人未定。
https://www.owasp.org/index.php/Projects/OWASP_Mobile_Security_Project_-_Mobile_Threat_Model

OWASP行为代码项目，Colin Watson。
https://www.owasp.org/index.php/OWASP_Codes_of_Conduct

OWASP GoatDroid项目，Jack Mannino。
https://www.owasp.org/index.php/Projects/OWASP_GoatDroid_Project

OWASP WhatTheFuzz项目，Joe Basirico。
https://www.owasp.org/index.php/OWASP_WhatTheFuzz_Project

OWASP ESAPI C++ 项目，项目领导人未定。
https://www.owasp.org/index.php/Projects/OWASP_ESAPI_C%2B%2B_Project

OWASP ESAPI C 项目，David Anderson 。
https://www.owasp.org/index.php/Projects/OWASP_ESAPI_C_Project

OWASP开发人员安全工具项目，Mark Curphey。
https://www.owasp.org/index.php/OWASP_Security_Tools_for_Developers_Project

OWASP数据交换格式项目，Psiinon和Dinis Cruz。
https://www.owasp.org/index.php/OWASP_Data_Exchange_Format_Project

OWASP小抄项目，SherifKoussa。
https://www.owasp.org/index.php/Cheat_Sheets

通过审核的发布

OWASP Zed攻击代理项目-ZAP 1.3.0版本发布，Psiinon。
https://www.owasp.org/index.php/Projects/OWASP_Zed_Attack_Proxy_Project/Releases/ZAP_1.3.0

最新发布

OWASP Zed攻击代理项目-ZAP 1.3.1版本发布，Psiinon。
https://www.owasp.org/index.php/Projects/OWASP_Zed_Attack_Proxy_Project/Releases/ZAP_1.3.1

OWASP HatkitDatafiddler 项目 –Hatkit Fiddler v0.5.0版本发布，Martin Holst Swende。
https://www.owasp.org/index.php/Projects/OWASP_Hatkit_Datafiddler_Project/Releases/Hatkit_Fiddler_v_0.5.0

OWASP Hatkit Proxy项目- Hatkit Proxy 0.5.1版本发布，Martin Holst Swende。
https://www.owasp.org/index.php/Projects/OWASP_Hatkit_Proxy_Project/Releases/Hatkit_Proxy_0.5.1

OWASP Mantra – 安全架构 –Mantra Security Toolkit 0.61版本发布，Abhi M BalaKrishnan。
https://www.owasp.org/index.php/Projects/OWASP_Mantra_Security_Framework/Releases/Mantra_Security_Toolkit_-_0.61

OWASP ESAPI Objective – C语言项目–Alpha版本发布，Deepak Subramanian。
http://code.google.com/p/owasp-esapi-objective-c/downloads/detail?name=ESAPI_ObjC_Framework_v0.0.1_Alpha.tar.gz

OWASP X5s 项目 –x5s v1.0.1 版本发布，ChrisWeber。
https://www.owasp.org/index.php/Projects/OWASP_X5s_Project/ Releases/x5s_v1.0.1

OWASP ModSecurity核心规则集项目-ModSecurity 2.2.0版本发布， Ryan Barnett。
https://www.owasp.org/index.php/Projects/OWASP_ModSecurity_Core_Rule_Set_Project/Releases/Current

OWASP Esapi- Ruby，0.30.0版本发布，Paolo Perego。
https://rubygems.org/gems/owasp-esapi-ruby/versions/0.30.0

OWASP社团 | 建设者，破坏者，守卫者

Michael Coats michael.coates@owasp.org
https://www.owasp.org/index.php/User:MichaelCoates

OWASP提供了丰富的应用安全材料。从流行的OWASP Top 10文档，到ESAPI Web应用程序安全控制库，OWASP研究并指引了许多安全软件开发生命周期的焦点领域和阶段。尽管OWASP提供了大量高质量的资料，一个严峻挑战是：如何在海量的信息中，寻找到一个与个人特定目标和与组织内的责任相关的项目、资源以及专业人士。

OWASP社团的建立就是为了解决这个问题，并提出三个主要目标：
1. 提供逻辑分类的资料，以使个人可以很容易的找到与他们的兴趣和在组织里所承担的责任相关的OWASP资料；
2. 将那些在各自企业组织里拥有相似的安全目标的人们联系在一起，以通过OWASP更好地、更方便的分享他们的知识。
3. 建立一个集中的区域，并针对每个社团的受众人群，以使开发的项目可以对于有意向的使用者进行自定义。

我们将OWASP的资料逻辑分组成为了三个社团：建设者、破坏者和守卫者。
建设者社团由涉及开发应用的个人组成，比如：开发人员和应用架构师。那些推动安全开发的OWASP项目被划分进建设者分组。
破坏者社团包括了那些旨在以发现应用的漏洞和威胁为目标的个人。这些个人包括：渗透测试人员和专注于安全的质量保证团队。那些协助确定应用和代码中安全威胁的OWASP项目被划分进破坏者分组。
守卫者社团是一组旨在保护已部署的应用免于网络攻击并研究潜在入侵的人。守卫者社团的成员包括安全监控团队，以及针对安全的事件响应和系统操作人员。那些为应用提供安全监控、安全部署和事件相应的OWASP项目被划分进守卫者分组。
最后，我们应该认识到，OWASP社团的目的是将有才华的专业人士聚集在各个项目，以将他们在广阔的应用安全领域里直接与他们特定的工作相关联。虽然OWASP社团将项目划分为了这些逻辑分组，但是目的并不是建立信息库。这些项目推荐选择一个主要的OWASP社团以最好的达到他们的目标受众，但是许多项目将跨过多个OWASP社团提供好处，因此，这些项目应提供其他信息，使受众人群能使用其他与安全生命周期中相关阶段特定的OWASP项目。
OWASP社团是一个新的思路，以及为所有成功的OWASP项目聚集有才华的个人的希望。通过提供一个可伸缩的机制，以将项目对于类似的目标受众而分组，从而增加安全专家在各自的应用程序的安全性之间交流，并为目标受众人群提供了OWASP的材料，而OWASP社区的概念也可以大大提高OWASP项目、工具和指南的质量和可用性。

有关该项目的任何问题，请联系：Michael Coates michael.coates@owasp.org
如果你发现了任何错误并希望修正或重新发布文中的内容，请联系：
Deepak Subramanian deepak.subramanian@owasp.org
Kate Hartmann kate.hartmann@owasp.org

对于XSS攻击的保护

Gareth Heyes, gazheyes@gmail.com

.....我发现的问题

从哪里开始呢？首先，让我告诉你，你读的书大多是错误的。你为了做一个特定任务而从互联网上复制的代码示例是错误的（错误的方式来处理一个GET请求），你同事从一个论坛复制而来并复制给你的函数是错误的（错误的方式来处理重定向）。从怀疑一切开始！也许这个博客帖子是错误的，这是一种心态，你需要为了保护你的网站免受XSS攻击。作为一名开发人员，你需要开始更多得思考你的代码。如果你正在阅读的文章中包含$ _GET或Response，并且在编写是未经过滤，那么是时候关闭该文章了。
答案是框架吗？我想，我的意见是：不是。是的，一个框架可能在短期内会阻止XSS攻击，但是长期的框架代码将被证明含有错误，所以，当它被利用时，它的危险性可能会比你自己写的代码更严重。为什么更严重呢？一个框架的漏洞可以很容易得被自动攻击，因为许多网站使用相同的代码库，如果你使用不同的过滤方法并编写自己的过滤代码，那么一个攻击者可能只攻击一个个人网站，但很难自动攻击一系列网站。这是今天互联网工作的主要原因之一，不是因为一切都是安全的，而只是因为所有的事物都是不一样的。

我听到的争论之一是，开发人员无法被信任去为一个站点创建一个完美的过滤系统，而使用一个框架能确保开发人员遵守最佳指引。我不同意这个观点，因为开发人员是聪明的，他们编写代码，并了解代码，如果你能建立一个系统并可以保护它，那是因为你处于最佳的位置。

如何处理输入

当你处理用户输入的信息时，你可以自己思考“一个数字是一个向量”。想象一个网站，提供一个图像服务器端，并允许你选择图形的宽度和高度。如果你认为一个数字不是一个向量，那么，你可能不会给将生成图形的宽度和高度设置任何限制。当攻击者请求一个100000×100000图形时，会发生什么呢？如果你的代码无法处理输入的最大值和最小值，那么攻击者就可以用多个任务请求DOS攻击你的服务器。因此，你不能对每一个将处理的输入偷懒，你必须确保每个值是都得到了正确验证。

处理的过程应该如下:
1. 验证数据类型——确保你获得的值是你所期望的；
2. 白名单——删除不被允许的字符，只提供允许出现的字符；
3. 验证长度——始终验证输入信息的长度，即使数据不被存储进入数据库。数据的长度越短，攻击者实施攻击就越困难；
4. 限制——筛选什么是你允许的字符范围内允许的（比如，最小值）；
5. 编码——根据环境（页面上的变量）正确地编码。

你可以通过将以上方法放入一个函数或者一个类里，使其非常容易。但是，你应该使每一个方法尽可能简单，并尽量避免让你的函数名字混乱。

HTML环境

让我们看一个使用以上方法的PHP代码示例。

```
<?php
$x = (string) $_GET['x']; //ensure we get a string not array
$x = preg_replace("/[^\w]/","", $x); //remove any characters that are not a-z, A-Z, 0-9 or _
$x = substr($x, 0, 10);//restrict to a maximum of 10 characters
if(!preg_match("/^a/i", $x)) {//this value must only begin with a or A
$x = '';
}
echo '<b>' . htmlentities($x, ENT_QUOTES) . '</b>'; //escape everything according to context
of $x
?>
```

结果是：“Warning:substr()expects parameter 1 to be string, array given”。

由于PHP允许通过一个GET请求来传递数组的特点，当使用白名单方法检验输入数据时，你可以针对不需要的数据类型，在PHP中创建一个警告。使用类型提示以确保你得到预期的类型。

很好！我们现在理解了如何对一个值进行限制和编码。让我们看看在另一个环境的情况。

脚本环境

当不处于XHTML/ XML的模式时，脚本标签无法解码HTML实体。如果你有一个已赋值的变量处于一个脚本标签中，那么你该怎么编码呢？比如：

```
<script>x='value here';</script>
```

在一个JavaScript的变量中，就象这样，你必须注意'和</script>。通过使用这些向量，这些值可以被利用以作为XSS攻击。下面列出了两个例子。

Vector 1: ',alert(1),//
Vector 2: </script><imgsrc=1 onerror=alert(1)>

第二个例子要求不使用引号，而很多开发人员认为它不会被执行，因为它仍处于一个JavaScript变量内。这显然是错误的，它被执行，因为浏览器并不知道脚本开头和结束的正确位置。

为了对一个脚本环境内的值进行编码，最好的办法是使用unicode编码，在JavaScript中的unicode编码看起来如下：

```
<script>alert('\u0061');//"a" in a unicode escape </script>
```

你可以尝试使用我开发的Hackvertor工具进行unicode编码。请了解它们是如何工作的，因为当你在了解如何对多种语言环境进行保护时，是很重要的。

```
<?php
functionjsEscape($input) {
	if(strlen($input) == 0) {
		return '';
	}
	$output = '';
	$input = preg_replace("/[^\\x01-\\x7F]/", "", $input);
	$chars = str_split($input);
	for($i=0;$i<count($chars);$i++) {
		$char = $chars[$i];
		if(preg_match("/^\t$/", $char)) {
			$output .= '\\t';//don't unicode escape but using a shorter \t instead.
Double escape remember!
			continue;//skip a line and move on the next char
		}
		$output .=sprintf("\\u%04x", ord($char));
	}
	return $output;
}
?>
```

对于你正在处理的特定变量，遵循前面所提到的步骤（验证数据类型、白名单、验证长度、限制和编码），这一点非常重要。但这个时候，我们将通过unicode方式其转换成我们的值。下面通过一个简单的函数来完成该功能。

在这个函数中，我特意没有设计一些优化处理，例如，可以使用十六进制转换来取代unicode的使用。因为我们限制了允许使用字符的范围，字母数字字符在转换时可以由它们的文字字符所取代，而当你在使用较短的等效符时，新行/制表符也可以被解码。让我们添加一行文字，使用制表符以代替\u0009。为什么要这么做呢？是为了减少发送的字符。

```
<?php
if(preg_match("/^\t$/", $char)) {
  $output .= '\\t';
continue;
}
?>
```

……处理标签的代码

我们专门将一个标签转换为“\t”。请注意我们如何区分输入和输出，并且，我们可以跳过输入的字符并重写成为更具体的东西。完整的代码如下。

```
<?php
functionjsEscape($input) {
        if(strlen($input) == 0) {
                return ‘’;
        }
        $output = ‘’;
        $input = preg_replace(“/[^\\x01-\\x7F]/”, “”, $input);//remove any characters outside
the range 0x01-0x7f
        $chars = str_split($input);
        for($i=0;$i<count($chars);$i++) {
                $char = $chars[$i];
                $output .=sprintf(“\\u%04x”, ord($char));//get the character code and convert to
hex and prefix with \u00
        }
        return $output;
}
?>
```

针对这段代码的小练习：

1．你能处理ASCII范围之外的字符吗？

2．将那些不危险的字符转换为它们的编码或文字形式。

……在XHTML中的脚本环境

在前面的章节中，当我说到“当不处于XHTML/XML模式时，脚本标签无法解码HTML实体”，你可能想知道有关XHTML的情况。在XHTML中，实体可以被解码，即使是在脚本块里面！幸运的是，本节中我提供的代码，因为使用了unicode编码，而可以进行相应的处理。如果你尝试了那节的练习，那么你有安全处理“&”吗？因为当你在操作XHTML网页时，这个是你必须想到的。为了使浏览器能使用XHTML，你必须在网页中使用正确的XHTML头信息。但是，我建议你不要使用XHTML的头信息。

即使前面的例子仍然保护你免受攻击，但我还是将向你展示对于XHTML网站的一对向量。

```
<script>x=’&#39;,alert(/This works in XHTML/)//’;</script>
<script>x=’&apos;,alert(/This also works in XHTML/)//’;</script>
```

这对于任何基于XML的格式都适用，实体可以被用来跳出字符串，另外一个简单的&lt/也可以实现该功能。不要使用XHTML，或者，如果你使用了unicode转换，切记不要允许“＆”。

……JavaScript事件

现在你知道了在XHTML中会发生的情况，你可能还有兴趣想知道在HTML属性中会发生什么。任何HTML属性，包括如onclick事件，会自动解码实体，并且当它们是字面字符时使用它们。我们用一段代码示例证明。

```
<div title=”&gt;” id=”x”>test</div><script>alert(document.getElementById(‘x’).title); </script>
```

正如你可以看到的，div元素的title属性的值，由“>”代替了“>”。因为它被自动解码。整个过程是由XSS其中的一个根本造成的，但是开发人员并不了解。让我们看看当一个onclick事件发生时，“X”变量会发生什么。

当点击链接时产生警报，因为像XHTML一样，实体得到解码，当你在属性的情况下，你需要做和在XHTML环境下完全一样的操作。重用你的jsecape函数将在属性和变量中充分保护你来自XSS的攻击。就像这样：

```
<a href=”#” onclick=”x=’&#39;,alert(1),&#39;’;”>test</a>
```

……innerHTML环境

我希望你已经掌握了以前的概念，因为从现在起，它要变得稍微有点混乱了。如果你在脚本环境中以任意方式设置了一个写入到dom中的值，那么，以前的编码规则将不再适用。虽然你在环境中正确得对值进行了编码，一旦它被应用到的innerHTML，则会发生改变。和前面一样，我们举一个例子：

```
<div id=”x”></div><script> //this is bad don’t do this with innerHTML document.
getElementById(‘x’).innerHTML=’<?php echo jsEscape($_GET[‘x’]);?>’;</script>
```

即使字符串是“\u003c\u0069\u006d\u0067\u0020\u0073\u0072...”等等，它仍会导致XSS攻击，因为innerHTML的写处理会查看到JavaScript字符串解码后的真实字符。你需要为HTML环境以及脚本环境进行编码，如果你将XHTML添加进入，那么它就会真的很复杂。我的建议是：在使用innerHTML环境时，不要允许HTML时使用innerHTML的情况下，取而代之的是对你的值使用白名单和限制方法，并使用innerText和textContent。如果你真的需要把HTML设置在innerHTML之中，那么请阅读后面的教程，以说明对于innerHTML如何编写一个基本的HTML过滤器。

……CSS环境

我先前说明的规则同样作用于 CSS，除了在 XHTML/XML 模式下，一个样式块将无法被解码，而样式属性将自动解码 HTML 实体。这使得在 CSS 环境下，如果你不知道自己在做什么，那么就很难面对注入式攻击提供保护。另外，对于常规的实体，CSS 还支持自己格式的十六进制转换。该格式是一个反斜线，外加一个需求字符的十六进制格式，该十六进制字符用零填充 2-6 位（供应商也支持超过 6 位长度的大量 0 填充。要看到它是什么样的，那让我们再次使用 Hackvertor 建立字符串。

正如你可以看到有相当多的组合，您可以使用的，还有更多。CSS 的详细说明中说明了注释可以被使用，并包括 C 语言的风格/* */，以及其他任何十六进制转换，包括在转换后添加一个空格以避免继续下一个十六进制字符。例如 CSS\61\62\63 仍然是“ABC”不管空格。希望你看了一会儿我的博客，了解使用实体以及十六进制转义或也许你刚刚意识到吗？是的，它是正确的，你可以使用十六进制转义，意见和 HTML 实体，构建一个有效的执行 CSS 值。

这就给你留下了一个关于保护 CSS 属性值的糟糕实例，IE7 和 IE7 兼容的浏览器支持 CSS 表达式。这基本上可以让你在 CSS 值内执行 JavaScrip 代码。一个简单的例子：

```
<div style=”xss:expression(open(alert(1)))”></div>
```

我使用open()函数调用，以避免客户端不断的弹出恼人的警报DOS。在表达式中处于“（”和“）”的东西是一行JavaScript的调用。在这个例子中，我使用的是一个叫做“XSS”的无效属性，但它更可能是“color”或“font-family”。让我们找到一个缺口，并开始编码CSS的值，并看看执行什么。我将只对表达式的“e”进行编码，使其更容易遵循。

```
Hex escape: <div style=”xss:\65xpression(open(alert(1)))”></div> Hex escape with trailing
space: <div style=”xss:\65 xpression(open(alert(1)))”></div> Hex escape with trailing
space and zero padded: <div style=”xss:\000065 xpression(open(alert(1)))”></div> Hex escape
with trailing space and zero padded and comment: <div style=”xss:\000065 /*comment*/
xpression(open(alert(1)))”></div> Hex escape with trailing space and zero padded and HTML encoded
comment: <div style=”xss:\000065 &#x2f;&#x2a;comment*/xpression(open(alert(1)))”></div> and
finally hex escape with encoded backslash with trailing space and zero padded and HTML encoded
comment: <div style=”xss:&#x5c;000065 &#x2f;&#x2a;comment*/xpression(open(alert(1)))”></div>
```

我相信你会同意这很难遵循，因为有数以百万计的组合。不幸的是，你不能对值进行简单的十六进制转换，并期望它对于注入式攻击是安全的，因为即使你所看到的CSS编码可作为向量使用。从防守的角度而言，你可以选择的做法是对CSS的每一个属性值进行白名单检验。幸运的是，我已经对CSS Reg做到这一点，而且，Norman Hippert帮助我将其转换到了PHP之中。

……作用于你的网页

你网站的每一个网页上的meta标签中应包括一个文档类型和一个UTF-8字符集，现在我们有一个缩短的HTML5头信息，我们可以使用下面的：

```
<!doctype html><html><head><meta http-equiv=”Content-Type” content=”text/html;
charset=UTF-8” /> ... your content ....
```

这是为了防止字符集攻击，使用E4X向量，并强制你的文档符合IE浏览器上的标准模式，是十分重要的。另外，我也建议你执行Dave Ross在博客里提供的标准模式。

这篇长博文章的最后一节，将是如何编写自己的过滤器。我不认为我是世界上最伟大的程序员，但我觉得我已经做出了一个很酷的技术，利用很少的代码去过滤内容，并只匹配你想要的内容，而你不会得到什么坏的结果。我希望你在这个代码的基础上，学习它，改善它。这段代码是故意不完整的。我写了一个称为 HTMLReg 的更完整的 HTML 过滤器，你可以用它来检验你是否想提高这个基本的过滤器。但是，我建议你尝试改善自己的过滤器并攻破它。

```
<script>
functionyourFilter(input) {
	var output = '' , pos = 0;
	input = input + ''; //ensure we have a string
	functionisNewline(chr) {
		eturn /^[\f\n\r\u000b\u2028\u2029]$/.test(chr);
	}
	functionoutputSpace(chr) {
		if(!/^\s$/.test(output.slice(-1)) && !isNewline(chr)) { //skip new lines and
multiple spaces
			output += chr;
		}
	}
	functionoutputChars(chrs) {
		output += chrs;
	}
	function error(m) {
		throw {
			description: m
		};
	}
	functionparseHTML() {
		arallowedTags = /^<\/?(?:b|i|strong|s)>/,
			match;
			if(allowedTags.test(input.substr(pos))) {
				match = allowedTags.exec(input.substr(pos));
				if(match === null) {
					error("Invalid tag");
				} else {
					pos += match[0].length;
					outputChars(match[0]);
				}
			} else {
				outputChars('&lt;');
				pos++;
			}
	}
	functionparseEntities() {
		varallowedEntities = /^&(?:amp|gt|lt);/,
			match;
			if(allowedEntities.test(input.substr(pos))) {
```

```
                                match = allowedEntities.exec(input.substr(pos));

                                if(match === null) {
                                        error("Invalid entity");
                                } else {
                                        pos += match[0].length;
                                        outputChars(match[0]);
                                }
                        } else {
                                outputChars('&amp;');
                                pos++;
                        }
                }
        while(pos<input.length) {
                chr = input.charAt(pos);
                if(chr === '<') {
                        parseHTML();
                } else if(chr === '&') {
                        parseEntities();
                } else if(/^\s$/.test(chr)) {
                        outputSpace(chr);
                        pos++;
                } else if(chr === '>') {
                        outputChars('&gt;');
                        pos++;
                } else if(chr === '"') {
                        outputChars('&quot;');
                        pos++;
                } else if(chr === "'") {
                        outputChars('&#39;');
                        pos++;
                } else if(/^[\w]$/.test(chr)) {
                        outputChars(chr);
                        pos++;
                } else {
                        pos++;//move to the next character but don't output it
                }
        }
        return output;
}
</script>
```

上面的代码区分了输入和输出，并显示了如何改变输入，而产生出一个不同的输出。这证明了如何使用输出去阻止重复出现的字符。你可以并且应该改变行为以匹配你的需求代码由JavaScript编写，但可以很容易地定制成你想使用的语言。

小练习
1.你可以安全地处理属性吗？
2.当需要时，你可以将换行符转换成
吗？
对本文有任何疑问，请联系：
Gareth Heyes，gazheyes@gmail.com
如果文中出现任何错误，而你想看到这篇文章经过修改和/或重新发布后的内容，请联系：
Deepak Subramanian deepak.subramanian@owasp.org
Kate Hartmann kate.hartmann@owasp.org

OWASP Podcast

由Jim Manico主持

OWASP的播客系列由Jim Manico先生主持，并特邀了多位安全专家。本周，我们特别邀请了来自OWASP全球项目委员会的Jason Li先生，请他来为大家介绍项目委员会是如何工作的，以及这些项目是如何推动OWASP前进的。

Podcast 链接: https://www.owasp.org/download/jmanico/owasp_podcast_89.mp3

OWASP Zed 攻击代理 (ZAP)

项目领导：Simon Bennetts

2011 年 6 月，OWASP Zed 攻击代理（ZAP）1.3.0 版本正式发布。这是一个具有里程碑意义的版本，因为它通过了正式评估，并被授予“稳定级”的发布地位。它也包含了显着增强，其中包括：

- 模糊，使用 JBroFuzz 代码，并包括了能够自动生成抵抗 CSRF 攻击令牌的能力；
- 动态的 SSL 证书，因此，你就可以生成一个可信任的 CA，然后截获浏览器透明的 SSL 连接；
- 一个 API 和“无头”或 daemon 模式，它允许开发人员将 ZAP 添加到他们的持续集成环境中；
- BeanShell 的整合，因此，ZAP 可以动态扩展到脚本中；
- 全面的国际化；
- 10 种语言的支持（除英语以外）。

随后的 bug 修复版本已经证明了团队的承诺，以支持他们的用户，并在用户报道重大问题尽快为他们解决。

项目负责人：Simon Bennetts（又名 Psiinon），在 AppSec EU 大会和 AppSec USA 大会上发表关于 ZAP 的演讲。

现在有 5 个开发人员为 ZAP 和一个真正的社团关注的焦点工作，ZAP 继续发展壮大，并已被指定为新的“旗舰”项目之一。

全球董事会董事现已宣布

OWASP 基金会董事会主管由六名选举出的志愿者组成。这些无偿服务的志愿者们投身到该组织的各项使命，支持 OWASP 全球委员会并在软件安全社区中发挥了举足轻重的作用。OWASP 董事会成员和委员会主席由民主选举产生，致使 OWASP 的使命得到自下而上的发展。

每年，一半的董事会席位和所有委员会主席席位，均由当时的注册会员选举产生。全球董事会董事的完整信息，请见：https://www.owasp.org/images/d/d6/2011-06-OWASPBYLAWS.pdf。

当前董事会主管及其职务

Michael Coats
董事会主席

Matt Tesauro
会计

Dave Wichers
董事会成员

EoinKeary
副主席

Tom Brennan
秘书

Sebastian Deleersnyder
董事会成员

OWASP 基金会人员

Kate Hartmann
全球运营官
Kate.Hartmann@owasp.org

Sarah Baso
全球会议和分部委员
Sarah.Baso@owasp.org

Alison Shrader
会计
Alison.Shrader@owasp.org

Kelly Santalucia
全球会员委员
Kelly.Santalucia@owasp.org

全球委员会

全球分会委员会

任务：
为全球各地的分会提供所需的帮助，以使 OWASP 的总体任务和目标发展并对其作出贡献。

委员会主席：Tin Zaw

全球会议委员会

任务：
OWASP 的全球会议委员会（GCC）的存在是为了协调全球的 OWASP 会议和活动，并为它们提供方便。

委员会主席：Mark Bristow

全球连接委员会

任务：
帮助 OWASP 基金会一统一一致的方式对外进行沟通。我们还协助不同 OWASP 项目和委员会之间的内部沟通。

委员会主席：Jim Manico

全球教育委员会

任务：
为公司企业、政府部门和教育机构提供对于应用安全的宣传、培训和教育服务。

委员会主席：Martin Knobloch

全球行业委员会

任务：
OWASP 全球行业委员会（GIC）旨在在行业、政府机构、学术界和监管机构中宣传和促进软件安全的最佳实例，并代表行业发出声音。这将通过各类宣传活动，包括：演讲、编写重要的文档，并与其他实体开展合作。

委员会主席：Rex Booth

全球会员委员会

任务：
会员委员会推荐各种政策、流程和策略以加强 OWASP 成员的数值和质量。该委员会提供一个书面计划，并推荐政策、流程和措施，以确保 OWASP 是一个日益扩大的重要成员组织。
委员会主席：Dan Cornell

全球项目委员会

任务：
营造一个积极的 OWASP 开发人员社区，为来自 OWASP 社区成员做出的贡献提供便利，为新项目提供支持和方向，并鼓励在大型国际组织采用 OWASP 的项目。
委员会主席：Jason Li

即将举行的活动

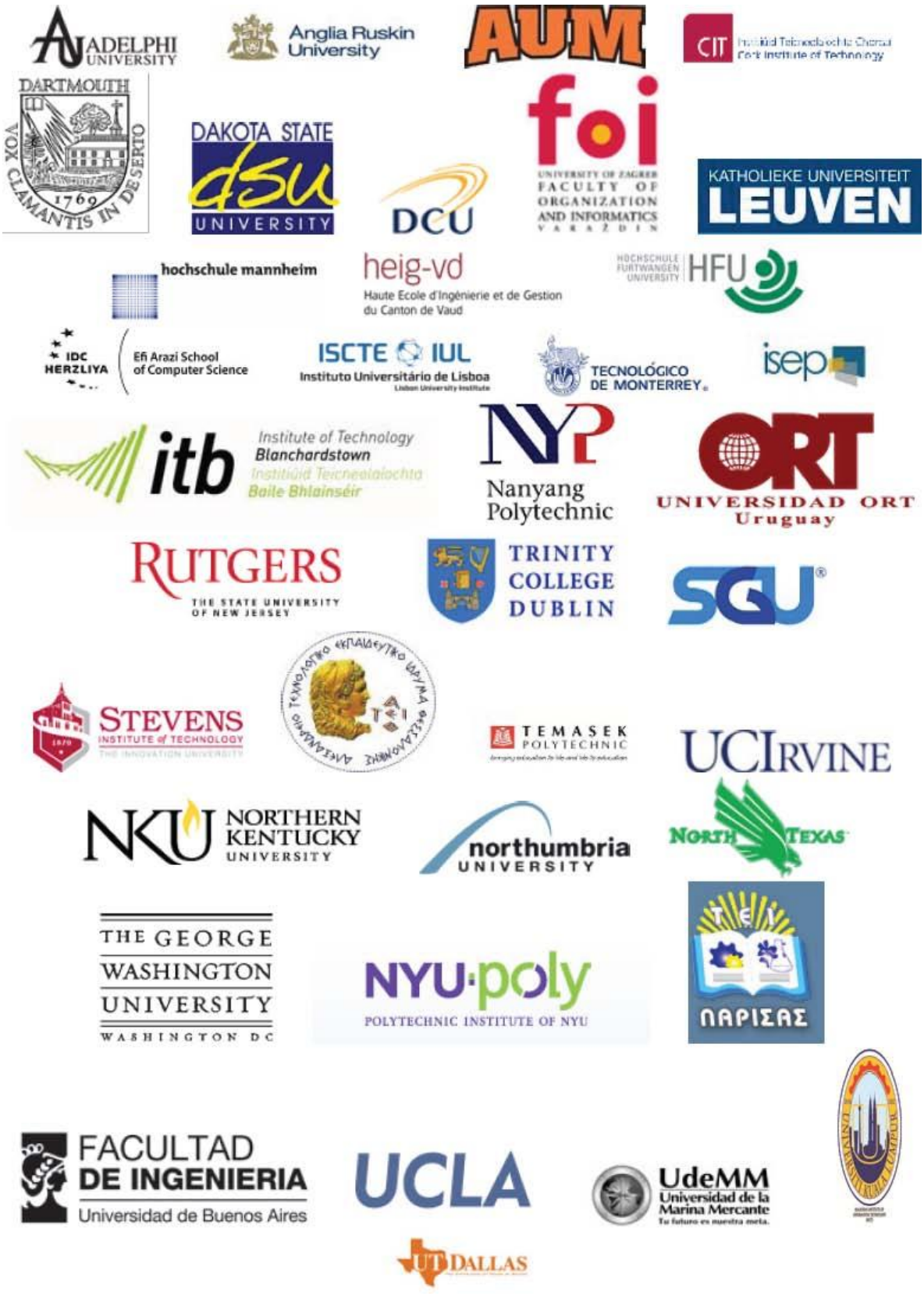
（所有活动的名单，请见：https://www.owasp.org/index.php/Category:OWASP_AppSec_Conference）

- OWASP BeNeLux 2011： 2011年11月30日 – 2011年12月1日，卢森堡；
- Global AppSecAsiaPac 2012：2012年4月11日 – 2012年4月14日，澳大利亚，悉尼；
- AppSec DC 2012：2012年4月2日 – 2012年4月5日，美国，华盛顿；
- Global AppSec Research 2012 (Wiki) 2012年7月10日 – 2012年7月13日，希腊，雅典；
- Global AppSec North America 2012： 2012年10月22日 – 2012年10月26日，美国德州奥斯丁；
- Global AppSec Latin America 2012：2012年11月14日 - 2012年11月16日。

业界合作伙伴


acunetix
ACCUVANT LABS
Adobe
ADP
CERT
Akamai
amazon
AppliedTrust
ascure
ASPECT SECURITY
Application Security Experts
ASTECH security
art defence
BEST BUY
Black Hat
Cargill
cigital
CORE SECURITY TECHNOLOGIES
DENIM GROUP
DIGITAL DEFENSE INCORPORATED
DREAMLAB TECHNOLOGIES
ENGINEERING
fishnet SECURITY
FUJITSU
gemalto
security to be free
GOTHAM DIGITAL SCIENCE
HARRIS CONNECT
HIGH-TECH BRIDGE
INFORMATION SECURITY SOLUTIONS
HUAWEI
IBM
(ISC)2
IMPERVA
Information Builders
INTREPIDUS GROUP
IOActive
COMPREHENSIVE COMPUTER SECURITY SERVICES
mozilla Firefox
mnemonic
securing your business
nixu
NetIQ
NOKIA
ORACLE
Penta SECURITY
PRAETORIAN
protiviti
Independent Risk Consulting
QUALYS
ON DEMAND SECURITY
Quotium
rackspace
Redspin
salesforce.com
Success On Demand
SD ELEMENTS
Security Innovation
SECURITY PS
Seeker
Symantec
TENABLE
Network Security
Trustwave
ups
WhiteHat
SECURITY

学术界合作伙伴


ADELPHI UNIVERSITY
Anglia Ruskin University
AUM
CIT
DARTMOUTH
VOX CLAMANTIS IN DESERTO
1769
DAKOTA STATE
dsu
UNIVERSITY
DCU
foi
UNIVERSITY OF ZAGREB
FACULTY OF ORGANIZATION AND INFORMATICS
VARAŽDIN
KATHOLIEKE UNIVERSITEIT
LEUVEN
hochschule mannheim
heig-vd
Haute Ecole d'Ingénierie et de Gestion du Canton de Vaud
HOCHSCHULE FURTWANGEN UNIVERSITY
HFU
IDC HERZLIYA
Efi Arazi School of Computer Science
ISCTE IUL
Instituto Universitário de Lisboa
TECNOLÓGICO DE MONTERREY
isep
itb
Institute of Technology Blanchardstown
Institiúid Teicneolaíochta Baile Bhlainséir
NYP
Nanyang Polytechnic
ORT
UNIVERSIDAD ORT
Uruguay
RUTGERS
THE STATE UNIVERSITY OF NEW JERSEY
TRINITY COLLEGE DUBLIN
SGU
STEVENS
INSTITUTE of TECHNOLOGY
TEMASEK POLYTECHNIC
UCIRVINE
NKU
NORTHERN KENTUCKY UNIVERSITY
northumbria
UNIVERSITY
NORTH TEXAS
THE GEORGE WASHINGTON UNIVERSITY
WASHINGTON DC
NYU·poly
POLYTECHNIC INSTITUTE OF NYU
ΠΑΡΙΣΑΣ
FACULTAD DE INGENIERIA
Universidad de Buenos Aires
UCLA
UdeMM
Universidad de la Marina Mercante
UT DALLAS

OWASP 基金会

开放的 Web 应用安全项目（OWASP）是一个国际的专业安全社团，致力于帮助企业和组织设计、开发、获取、操作和维护安全的应用系统。为了改善应用软件的安全，OWASP 的所有工具、文件、论坛和分会都是免费和开源的。我们认为应用安全的问题是人、流程和技术的问题，因为同时处理这三个问题是到达应用安全的最佳途径。OWASP 是一个新型的组织。由于没有商业压力，我们可以提供应用安全方面的的公正、实用和有效的信息。虽然 OWASP 提倡使用商业技术，但是我们与任何技术公司都没有关联。跟许多开源项目类似，OWASP 以合作和公开的方式制作了多种应用安全材料供大家使用。

.....核心价值

- 开放性——OWASP中的一切，从我们的财政状况到我们的代码，从根本上都是透明的；
- 创新性——OWASP鼓励并支持对于软件安全问题解决方案的创新和实验；
- 全球性——世界各地的人都被鼓励参加OWASP社团；
- 完整性——OWASP是一个诚实的、可信任的、厂商中立的和国际的社团。

OWASP会员制

OWASP基金会专业联盟是一个不以营利为目的、不与任何商业业务或服务想联系的501c3慈善机构。要取得成功，我们需要您的支持。OWASP的个人，以及应用安全社团支持的教育和商业组织共同协作，创建了文章、方法、文档、工具和技术。
OWASP 所有成员的完整列表，可以在这里找到：https://www.owasp.org/index.php/Membership。

.....个人支持者 50美金/每年

- 强调您的 web 应用软件安全的意识；
- 享有参加 OWASP 会议的折扣；
- 扩大您的个人社交网；
- 获取一个 owasp.org 的电子邮件地址；
- 分配您会员费的 40%，以直接支持你选择的地方分会；
- 参与全球选举，并在社团的关键问题上拥有投票权。

.....企业支持者 5000美金/每年

- 减税捐赠；
- 享有在 OWASP 会议展示的产品或服务的折扣；
- 有机会在 owasp.org 的网站上免费粘贴 30 天的广告（价值$ 2,500）；
- 在 OWASP 的网站上粘贴您公司的徽标，以被确认为 OWASP 的支持者；
- 在每季度的新闻简报中被列为一个支持者，新闻简报将被发送给上万个个人；
- 通过全球行业委员会代表，有一个集体的声音；
- 参与全球选举，并在社团的关键问题上拥有投票权；
- 分配你每年捐赠的 40%，以直接支持你选择的地方分会和/或项目。

有关会员及赞助机会的更多信息，请联系：Kelly SantaluciaKelly.santalucia@owasp.org

新闻简报广告业务

- 1/4 单页广告：2000 美金；
- 1/2 单页广告：2500 美金；
- 1/2 单页广告 + 一个 OWASP 网站中的标题横幅或者 10 本 Top 10 的书：3000 美金；
- 一整页单页广告：5000 美金；
- 全年（每季度的新闻简报中占有 1/2 单页广告）：9000 美金。

预知详情，请联系：Kelly.Santalucia@owasp.org 或 Kate.Hartmann@owasp.org。