中南大學

CENTRAL SOUTH UNIVERSITY

HEARTBLEED 实验报告

学生姓名 王琪

学　　号 0906140120

专业班级 信安 1401

指导教师 王伟平

学　　院 信息科学与工程学院

完成时间 2016 年 12 月

Heartbleed 实验

Heartbleed 实验

1.概述

Heartbleed 错误（CVE-2014-0160）是 OpenSSL 库中的一个严重的实现缺陷，它启用攻击者从受害者服务器的内存窃取数据。被盗数据的内容取决于什么是在服务器的内存中。它可以潜在地包含私钥，TLS 会话密钥，用户名称，密码，信用卡等。漏洞是在 Heartbeat 协议的实现，其由 SSL / TLS 用于保持连接活动。

本实验的目的是让学生了解这个漏洞是多么严重，如何攻击工作，以及如何解决这个问题。受影响的 OpenSSL 版本范围为 1.0.1 到 1.0.1f。的版本在我们的 Ubuntu VM 是 1.0.1。

2.实验室环境

在本实验中，我们需要设置两个虚拟机：一个称为攻击者计算机，另一个称为受害服务器。我们使用预构建的 SEEDUbuntu12.04 VM。 VM 需要使用 NAT-网络适配器网络设置。这可以通过转到 VM 设置，选择网络，然后单击适配器来完成标签将适配器切换到 NAT-Network。确保两个虚拟机在同一 NAT 网络上。

此攻击中使用的网站可以是使用 SSL / TLS 的任何 HTTPS 网站。然而，因为它是非法攻击一个真实的网站，我们在我们的 VM 中设置了一个网站，并自己进行攻击 VM。我们使用一个名为 ELGG 的开源社交网络应用程序，并将其托管在以下 URL 中：https://www.heartbleedlabelgg.com。我们需要修改攻击者计算机上的/ etc / hosts 文件，将务器名称映射到 IP 地址的服务器 VM。搜索/ etc / hosts 中的以下行，并替换 IP 地址 127.0.0.1 与托管 ELGG 应用程序的服务器 VM 的实际 IP 地址。127.0.0.1 www.heartbleedlabelgg.com

3.实验室任务

在开展实验室任务之前，您需要了解心跳协议的工作原理。心跳协议由两种消息类型组成：HeartbeatRequest 包和 HeartbeatResponse 包。客户向服务器发送 HeartbeatRequest 包。当服务器接收到它时，它发送回的副本在 HeartbeatResponse 包中接收到的消息。目标是保持连接活着。

3.1 任务 1：启动 Heartbleed 攻击。

在这个任务中，学生将在我们的社交网站上启动 Heartbleed 攻击，看看是什么样的可以实现损坏。 Heartbleed 攻击的实际伤害取决于什么样的信息存储在服务器存储器中。如果服务器上没有太多活动，您将无法访问窃取有用数据。因此，我们需要作为合法用户与 Web 服务器进行交互。让我们做它的管理员，并执行以下操作：

•从浏览器访问 https://www.heartbleedlabelgg.com。

•以站点管理员身份登录。 （用户名：admin;密码：seedelgg）

•将 Boby 添加为朋友。 （转到更多 - >成员，然后单击 Boby - >添加好友）

•向 Boby 发送私人消息。

在您作为合法用户进行了足够的交互后，您可以启动攻击并查看什么信息，您可以从受害服务器。编写程序启动 Heartbleed 攻击划伤并不容易，因为它需要心跳协议的低级知识。幸运的是，其他人们已经写了攻击代码。因此，我们将使用现有代码来获得第一手在 Heartbleed 攻击的经验。我们使用的代码称为 attack.py，它最初是由 Jared Stafford 写。我们对教育目的的代码做了一些小的改动。您可以从实验室的网站下载代码，更改其权限，以便文件是可执行的。然后可以运行攻击代码如下：

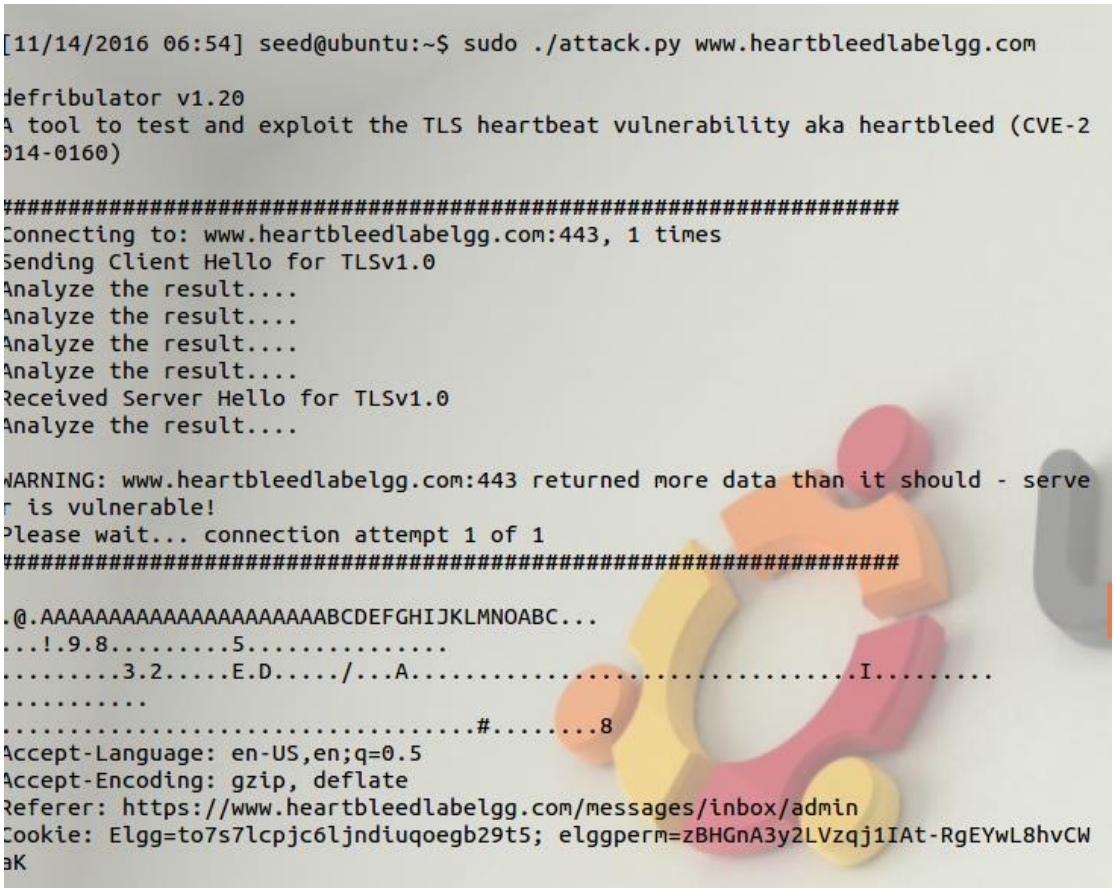
```
$ ./attack.py www.heartbleedlabelgg.com
```

您可能需要多次运行攻击代码以获取有用的数据。试着看看你能不能得到从目标服务器获取以下信息。

•用户名和密码。

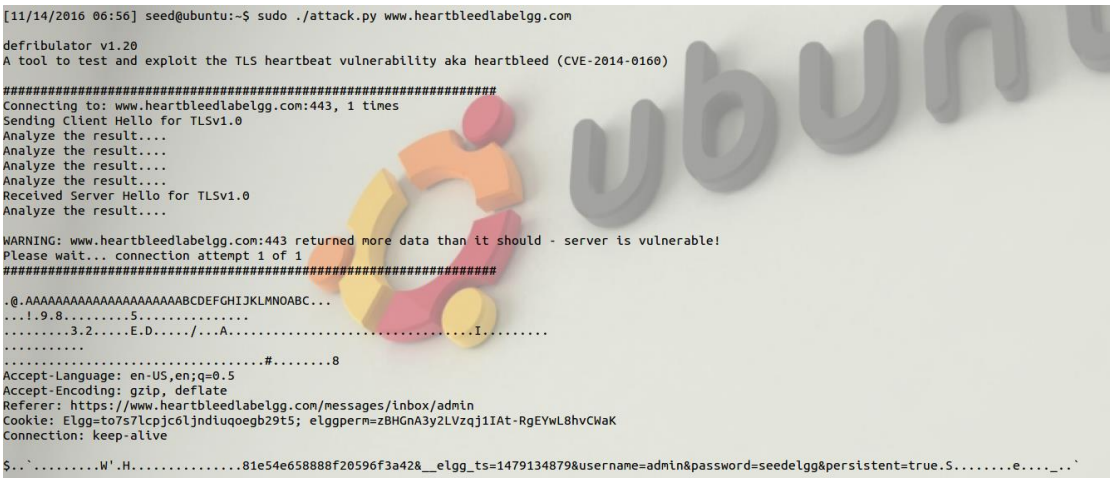
•用户的活动（用户做了什么）。

•私人消息的确切内容。



以上为不断运行攻击语句而获得的结果。

捕捉到的登录名和密码



3.2 任务 2：查找 Heartbleed 漏洞的原因

在这个任务中，学生将比较良性数据包的结果和发送的恶意数据包攻击者代码来找出 Heartbleed 漏洞的根本原因。Heartbleed 攻击基于 Heartbeat 请求。这个请求只是发送一些数据到服务器，并且服务器将数据复制到其响应包，因此所有数据被回送。在正常情况下，假设请求包括 3 字节的数据“ABC”，因此长度字段具有值 3.服务器将将数据放置在存储器中，并且从数据的开始将 3 个字节复制到其响应分组。在里面攻击情形，请求可能包含 3 个字节的数据，但长度字可能表示为 1003.当服务器构造其响应分组，它从数据的开始（即“ABC”）复制，但是它复制 1003 字节，而不是 3 个字节。这些额外的 1000 个类型显然不是来自请求包;他们来自服务器的私人内存，它们可能包含其他用户的信息，密钥，密码等。

在这个任务中，我们将使用请求的长度字段。首先，让我们来了解心跳响应包从图 2 构建。当 Heartbeat 请求包到来时，服务器将解析分组以获得有效载荷和有效载荷长度值（在图 2 中突出显示）。这里，有效负载只是一个 3 字节的字符串“ABC”，Payload 长度值正好是 3.服务器程序将盲目地从请求分组中出这个长度值。然后它通过指向构建响应数据包
存储器存储“ABC”并将有效负载长度字节复制到响应有效载荷。这样，
响应包将包含一个 3 字节的字符串“ABC”。我们可以启动 HeartBleed 攻击，如图 3 所示。我们保持相同的有效负载（3 字节），但将 Payload 长度字段设置为 1003.服务器将再次盲占该有效负载长度值构建响应包。这一次，服务器程序将指向字符串“ABC”和将 1003 字节从存储器复制到作为有效载荷的响应分组。除了字符串“ABC”，额外 1000 字节被复制到响应包中，其可以是来自存储器

的任何东西，诸如秘密活动，日志信息，密码等。

我们的攻击代码允许你使用不同的有效载荷长度值。默认情况下，值为设置为相当大的一个（0x4000），但是您可以使用命令选项“-l”（字母 ell）减大小，或“ - length”，如以下示例所示：

$。/ attack.py www.heartbleedlabelgg.com -l 0x015B

$。/ attack.py www.heartbleedlabelgg.com --length 83

Length=15 时

Length=250 时：

Length=600 时：

你的任务是玩攻击程序具有不同的有效载荷长度值，并回答以下问题：

•问题 2.1：随着长度变量减少，您可以观察到什么样的差异？

答：当 length 值不断减小时，所能捕捉到的信息也越来越多，当 length=600 时，已经很难捕捉到有用的信息了。

•问题 2.2：随着长度变量减小，输入长度变量有一个边界值。在该边界处或以下，心跳查询将接收响应分组而不附加任何额外的数据（这意味着请求是良性的）。请找到边界长度。你可能需要尝试许多不同的长度值，直到 Web 服务器发回回复，没有额外的数据。至帮助你这样，当返回的字节数小于预期的长度，程序将打印“服务器处理的畸形心跳，但没有返回任何额外数据。”

答：当 length 不断减小时，经过不断测试得出，当 length=22 时没有任何返回数据。

Length=24

```
[11/14/2016 20:34] seed@ubuntu:~$ sudo ./attack.py www.heartbleedlabelgg.com --length 24

defribulator v1.20
A tool to test and exploit the TLS heartbeat vulnerability aka heartbleed (CVE-2014-0160)

################################################################
Connecting to: www.heartbleedlabelgg.com:443, 1 times
Sending Client Hello for TLSv1.0
Analyze the result....
Analyze the result....
Analyze the result....
Analyze the result....
Received Server Hello for TLSv1.0
Analyze the result....

WARNING: www.heartbleedlabelgg.com:443 returned more data than it should - server is vulnerable!
Please wait... connection attempt 1 of 1
```

当 length=22 时没有任何返回数据

```
[11/14/2016 20:36] seed@ubuntu:~$ sudo ./attack.py www.heartbleedlabelgg.com --length 22

defribulator v1.20
A tool to test and exploit the TLS heartbeat vulnerability aka heartbleed (CVE-2014-0160)

################################################################
Connecting to: www.heartbleedlabelgg.com:443, 1 times
Sending Client Hello for TLSv1.0
Analyze the result....
Analyze the result....
Analyze the result....
Analyze the result....
Received Server Hello for TLSv1.0
Analyze the result....
Server processed malformed heartbeat, but did not return any extra data.
Analyze the result....
Received alert:
Please wait... connection attempt 1 of 1
################################################################
```

3.3 任务 3：对策和错误修复

要修复 Heartbleed 漏洞，最好的方法是将 OpenSSL 库更新到最新版本这可以使用以下命令实现。应该注意的是，一旦更新，很难回到脆弱的版本。因此，请确保您在完成之前的任务更新。您还可以在更新之前创建 VM 的快照。

#sudo apt-get update

#sudo apt-get upgrade

任务 3.1 在更新 OpenSSL 库之后再次尝试您的攻击。请描述你的观察。

任务 3.2 此任务的目标是找出如何修复 Heartbleed 错误在源代码中。

以下 C 样式结构（与源代码不完全相同）是心跳的格式请求/响应分组。

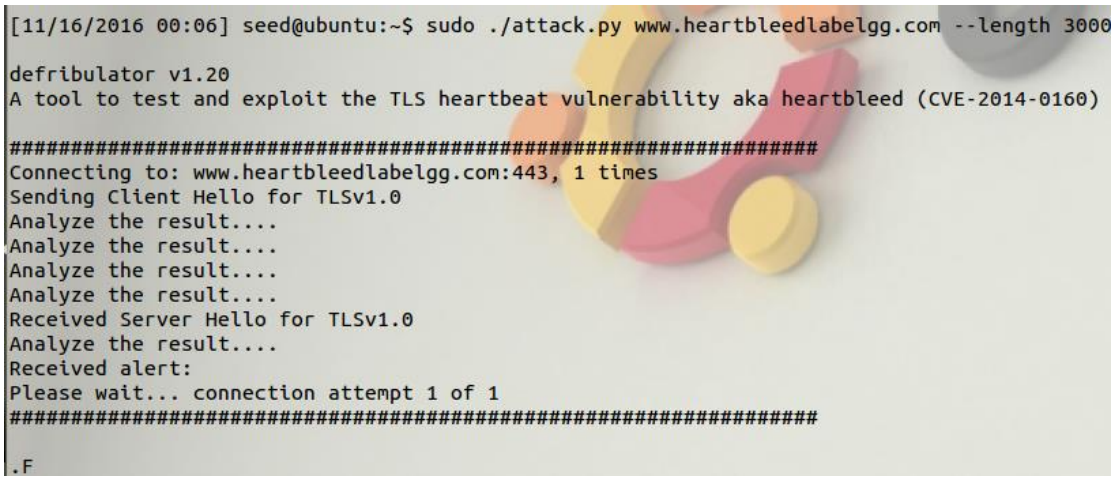
```
struct {
   HeartbeatMessageType type; // 1 byte: request or the response
   uint16 payload_length;     // 2 byte: the length of the payload
   opaque payload[HeartbeatMessage.payload_length];
   opaque padding[padding_length];
} HeartbeatMessage;
```

分组的第一字段（1 字节）是类型信息，并且第二字段（2 字节）是有效载荷长度，其次是实际有效载荷和填充。有效载荷的大小应该与值相同在有效载荷长度字段中，但在攻击情形中，有效载荷长度可以设置为不同值。

更新后发送新信息后捕包：

更新后捕包：

更新后捕包无 warning 提醒 length=3000



4.实验总结

通过此次实验，我明白了心脏滴血漏洞的原理，它可从特定服务器上随机获取64k 的工作日志，由于数据是随机获取的，所以攻击者也不一定可以获得想要的信息，因此整个过程如同钓鱼， 攻击可能一次次持续进行，大量敏感数据可能泄露。由于一台服务器的密钥也记录在其工作日志中，并且在大量数据中可被轻易辨别，因此将是首当其冲的获取目 标，获取密钥后，攻击者可以掌握某网站或服务的实时流量情况，甚至可以破解被加密的以往流量日志。危害极大。

此次实验也使我更加明白了网络安全的重要性，身为一名信息安全专业的学生，更应该了解更多的安全问题及相关防御知识，平时应该多积累。日后的学习还有很长的路要走，也感谢在这期间老师和同学给予我的帮助。