从零开始学习CTF——CTF是什么

原创

洛柒尘 于 2020-06-03 12:01:20 发布

分类专栏： CTF学习 文章标签： 安全

本文链接：https://blog.csdn.net/ewyherayh/article/details/106497636

引言：

从2019年10月开始接触CTF，学习了sql注入、文件包含等web知识点，但都是只知道知识点却实用不上，后来在刷CTF题才发现知识点的使用方法，知道在哪里使用，哪里容易出漏洞，可是在挖src漏洞中还是很迷漫，学了快一年还是没挖过一条src漏洞。这一系列是把自己学习的CTF的过程详细写出来，方便大家学习时可以参考。

编辑不易，转载请联系说明用途，并标记作者姓名和文章来源！

一、CTF简介

简介

中文一般译作**夺旗赛**（对大部分新手也可以叫**签到赛**），在网络安全领域中指的是网络安全技术人员之间进行技术竞技的一种比赛形式

CTF起源于1996年**DEFCON全球黑客大会**，以代替之前黑客们通过互相发起真实攻击进行技术比拼的方式

竞赛模式

- 解题模式：
  在解题模式CTF赛制中，参赛队伍可以通过**互联网**或者**现场网络**参与，这种模式的**CTF竞赛**与**ACM编程竞赛**、**信息学奥赛**比较类似，以解决网络安全技术挑战题目的**分值**和**时间**来排名，通常用于**在线选拔赛**。题目主要包含**逆向**、**漏洞挖掘与利用**、**Web渗透**、**密码**、取证、隐写、**安全编程**等类别。
- 攻防模式：
  在攻防模式CTF赛制中，参赛队伍在网络空间互相进行**攻击**和**防守**，挖掘网络服务漏洞并**攻击对手服务**来**得分**，修补自身服务漏洞进行防御来避免丢分。攻防模式CTF赛制可以实时通过得分反映出比赛情况，最终也以得分直接分出胜负，是一种竞争激烈，具有很强观赏性和高度透明性的网络安全赛制。在这种赛制中，不仅仅是比参赛队员的**智力**和**技术**，也比**体力**（因为比赛一般都会持续48小时及以上），同时也比团队之间的分工配合与合作。
- 混合模式：
  结合了解题模式与**攻防模式**的**CTF赛制**，比如参赛队伍通过解题可以获取一些初始分数，然后通过攻防对抗进行得分增减的零和游戏，最终以得分高低分出胜负。采用混合模式CTF赛制的典型代表如iCTF国际CTF竞赛。

总结

一般大型比赛大都是i春秋承办的，以小组的形式比赛，分预选赛和总决赛。

- 预选赛
  都是线上比赛，比赛形式几乎都是**解题模式**。进入官方提供的网站，登录账号密码后会到一个页面，题目按类别分类（后续会讲有什么类别），点击后提供**题目链接**、**题目信息**、**题目提示**和**flag**（答案）提交，提交后会得到分数。其中，一血、二血、三血拿的分数比较高，后面答该题的队伍拿的分数就一样了，答题的队伍越少，分数就越多，然后按分数排名，确定进入决赛的队伍。

> 如：一道签到题，开始定的分为500，当有队伍提交正确的flag，那么分数就会下降，下降到470/450这样的，人数越多，分数下降幅度越大。而一血二血三血获得分数会高于后续答题正确的

- 总决赛
  几乎都是线下赛，进行**混合模式**，因为攻防模式的题目容易出现**易守难攻**、**比速度节题**这种学习不到知识的问题，也因为这样，大部分的总决赛都是以**逆向**为主，很容易变成**逆行大赛**。那么像web选手去干吗呢？喝茶打杂去吧！刚学没多久，又比较懒，关于总决赛的都是在网上获取信息的，想了解详细的参考
  第五届XCTF总决赛的赛制以及新型攻防赛题之探索
  CTF wiki

二、题目分类

题目一般为6大类：

1.Web(网络安全)

- **Web**是CTF竞赛中主要的题型之一，题目涉及到许多常见的**WEB漏洞**，诸如**XSS**、**文件包含**、**代码执行**、**上传漏洞**、**SQL注入**。也有一些简单的关于**网络基础知识**的考察，例如**返回包**、**TCP-IP**、**数据包内容**和**构造**。可以说题目环境比较接近**真实环境**。
- 所需知识点：PHP、Python、SQL(以mysql为主)、TCP-IP、linux命令、html、javascript等。

2.MISC(安全杂项)

- **MISC**是大型CTF竞赛的题目难度很大，是一个可以拉开分数的类型，而在小型竞赛和题库中却难度不大。题目涉及**隐写术、流量分析、电子取证、人肉搜索、数据分析、大数据统计**等等，覆盖面比较广，主要考查参赛选手的各种基础综合知识。
- 所需知识点：熟悉使用众多隐写工具、流量审查工具、了解编码等。

3.Crypto(密码学)

- 主要包括**古典密码学**和**现代密码学**两部分内容，古典密码学趣味性强，种类繁多，现代密码学安全性高，对算法理解的要求较高。
- 所需知识点：矩阵、数论、古典密码学、算法等。

4.Reverse(逆向)

- 题目涉及到**软件逆向、破解技术**等，要求有较强的**反汇编、反编译**扎实功底。主要考查参赛选手的**逆向分析**能力。
- 所需知识点：**汇编语言、加密与解密、常见反编译工具**。

5.PWN(二进制安全)

- PWN在黑客俚语中代表着**攻破**，取得权限，在CTF比赛中它代表着溢出类的题目，其中常见类型溢出漏洞有**栈溢出、堆溢出**。主要考察参数选手**对漏洞的利用能力**。
- 所需知识点：**C，OD+IDA，数据结构，操作系统**。

6.Mobile(移动安全)

- 主要介绍了**安卓逆向**中的常用工具和主要题型，安卓逆向常常需要一定的安卓开发知识，iOS 逆向题目在 CTF 竞赛中较少出现，因此不作过多介绍。

7.(区块链)

-近来多个CTF比赛均出现**区块链**题目，区块链应用越来越成为热门应用，在未来区块链会成为一个**重点**。因为没有接触过，这里不详细讲。

三、怎么入门

入门这一块是我们这些新手的一道大大的门槛，很多人都是不知道怎么入门。

个人入门步骤

1.确定方向

一般分为两个方向
A 方向：**PWN+Reverse+Crypto** 随机搭配
B 方向：**Web+Misc** 组合
Misc 所有人都可以做

入门知识：
都要学的内容：Windows 基础、Linux 基础、计算机组成原理、操作系统原理、网络协议分析
A 方向：IDA 工具使用（fs 插件）、逆向工程、密码学、缓冲区溢出等
B 方向：Web 安全、网络安全、内网渗透、数据库安全等前10的安全漏洞

2.怎么刷题

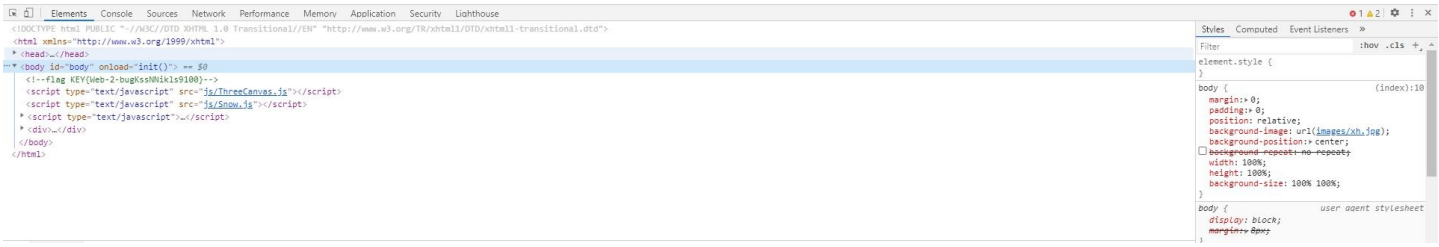
刷题非常重要，很多人在其他ctf入门教程中都可以看到**刷题**是第一步，也是进阶的重要一步，所以**刷题**非常重要，可是刷题也是需要有技巧的。

例题1：bugku中的web2

打开题目发现一堆笑脸疯狂向你怼来，而且速度越来越快，那么这一题怎么做呢？

这一题考察的是**信息收集**。在Chrome(谷歌浏览器)点击**f12**，就可以打开**控制台**，可以看见其中有一条

```
<!--flag KEY{Web-2-bugKssNNikls9100}-->
```



这一句是什么意思，就是把答案告诉你了，你可以把`KEY{Web-2-bugKssNNiklS9100}`提交到输入框就可以获得分数了。

总结

提交完flag后要做什么呢？这时你就要看这一题考察的是什么？**控制台**，那么控制台又是什么呢？要是知道就做下一题，如果不知道那么我就**谷歌**（什么是**网页控制台**），也可以百度，然后找到一篇知乎的回答Chrome 按下 F12 之后出现的功能是做什么用的？

在记住后看这一知识点难不难，自己能不能掌握，如果怕忘记，那你就记笔记，可以用**印象笔记**、**有道云笔记**，推荐使用印象笔记，因为它有多级分层，也没必要买会员，它送的那点容量只要不放视频或则图片不要放太多，一个月是用不到30m的。

再来一题了解一下

例题2：bugku中的计算器

打开题目可以看见是个加法，18+91=109，可是我却只能输入一个1，这是为什么呢？

打开控制台看了一下，发现一个很特殊属性**maxlength**，那么我就谷歌一下**maxlength**是什么东西。

查询结果发现，**maxlength**是一个**input**元素中的用来**限制字符数**的，那我们就尅改一下，把他改成3、4之类的数值，只要能容纳我的答案就行了。输入后就可以获得**flag**了。

总结
这一题考察的是**HTML**中**input**元素的**maxlength**属性，这时你就可以去把**HTML**学习一下了，可以试着怎么写网站，哪怕是最普通的也可以，学完以后做到关于**HTML**相关知识点的时候你就会很快找到哪里有问题。

3.以练促赛，以赛养练

选择一场已经存在Writeup的比赛或者参加一场最新的CTF比赛。
总结解题过程，最好能写一写博客之类的。

4.推荐平台

极力推荐**bugku**，其他的后面慢慢来，先把**bugku**做好，然后在做**攻防世界**，你就会跨入进阶的门槛了。

5.工具收集

不要先去下载别人推荐的工具，反正你是不会去学的，一定要在解题的过程中寻找，那样你回加深印象，也能直接总结一套经验出来。

四、编程

很多人会纠结到底要不要深入编程。不要太深入，差不多就可以了，对照文档就可以写出程序就差不多了且能看得懂就好了，因为每个语言都是学不完的，一直在更新，如果过分专注于一个语言，安全知识就挺容易漏下的。或则是针对一些热门的编程进行深入，比如**PHP**、**python**、**java**之类的，其中**PHP**可以说是必学项目，如果可以的话，把这门语言学透，那么**代码审计**一点压力都没有。

上面都是个人想法，怎么选都要靠自己决定，要不要深入、深入哪一门都是问题，一定要考虑清楚再下手，别等下学了一半就改变方向，这是学习的**大忌**，因为你学其它的也会这样，除非是你发现这个不是很适合你的发展。

总结

刷题刷题刷题最重要的是刷题，，不会就看别人的**Writeup**，不要怕做不出，刷题一开始是很无聊的，但这是学习的过程，如果不做就永远不会。

后面的章节将慢慢讲解ctf中的各项知识点