题目1：芯片电路安全漏洞挖掘

一、题目背景：

芯片安全是设备安全可信的基石，且芯片安全漏洞极有可能存在不可补丁修复的风险，对设备及整个网络的信息安全尤为重要。随着针对芯片攻击技术的发展，当前芯片除了面临逻辑攻击外，还面临侧信道、故障注入、物理攻击等专门针对芯片的近端攻击，从而衍生出对应的芯片漏洞的可能性。芯片存在开发周期短且后续无法升级等特点，如何完备、高效、自动化、智能化完成芯片安全漏洞的挖掘，成为当前芯片安全重要难题。

二、题目描述：

给定一个芯片电路和电路安全需求说明书，选手需要设计一个工具，挖掘芯片电路verilog中的安全漏洞。芯片电路以verilog语言由多个源文件实现。电路安全说明书定义电路关键资产path，电路各输入/输出port的安全权限属性（安全/非安全）。工具可支持电路关键资产path、电路各port的安全属性设定，挖掘出芯片电路中存在的安全漏洞。

三、考察选手芯片电路漏洞识别和挖掘能力，挑战内容如下：

1、掌握业界芯片电路典型漏洞的脆弱性和攻击方式。

2、芯片电路的安全分析和漏洞挖掘能力。

三、交付件

1、工具源代码。

2、工具源代码技术说明书：工具使用关键技术原理的详细描述，代码功能实现的详细描述。

3、工具使用说明书及DEMO演示视频，说明工具运行的依赖环境和操作说明。

四、评价方式：

1、工具设计阶段参赛者需要自行收集、构造和测试芯片电路各种漏洞，验收环节出题方会使用参赛者提交的工具分析预设有典型芯片漏洞的考题代码，从如下两个层面评价工具得分：

1）挖掘出的漏洞数量不少于考题电路预设的漏洞数量总数的80%。

2）挖掘的漏洞数量相同时，时间越少越好。

五、咨询邮箱： penglong2@huawei.com

六：赛题互动交流答疑社区链接：

https://www.chaspark.com/#/races/competitions/1005987214781644800

题目 2：Web 应用越权漏洞检测

一、题目背景：

随着网络安全水平的不断提高，传统黑盒扫描对一些常见的漏洞，如SQL注入和RCE等具有较好的检测效果，但对越权类漏洞检能力较弱，而越权漏洞作为一种常见的安全漏洞，给系统和用户带来了严重的风险，由于其存在范围广、危害大，被OWASP列为Web应用十大安全隐患的第一名。

目前，传统黑盒扫描器对于越权漏洞的检测能力尚处于起步状态，为了提高对越权漏洞的检测和防范能力，本赛题期望选手可以设计和开发一款漏洞扫描工具，通过黑盒测试的方式，检测Web系统中的越权类漏洞。

二、题目描述：

参赛选手需要针对给出的Web漏洞靶场，利用黑盒测试的方法，尝试发现和利用这些越权漏洞，并能够生成相应的检测报告。

交付内容：完整系统程序源码；系统设计相关文档等。

三、考察选手编程能力，挑战内容如下：

1、具备登录检测能力，针对给定的账号，进行网站登录，同时对于验证码具有一定的自动识别能力。

2、通过网页爬虫，能够全面完整的收集到目标网站页面。

3、分析爬取到的页面，构造数据包，针对可能存在越权的参数，识别是否存在越权漏洞，包括：垂直越权和水平越权和未授权访问。

四、评价方式：

1、漏洞发现数量与漏报量：评估选手能否全面地发现应用程序中的越权漏洞。

2、报告质量：评估选手提交的报告是否清晰详细，是否包含可运行的源代

码。

3、创新性：是否有创新的思路、算法或方法在程序中得以应用。

4、编程技能：评估选手编写程序代码的质量和运行效率。

5、初赛给出的附件用于选手系统设计和测试，决赛给出的附件用于评分。

五、咨询邮箱： xuxiaoqiang6@huawei.com

六：赛题互动交流答疑社区链接：

https://www.chaspark.com/#/races/competitions/1005988897337737216

题目 3：HTTP 载荷变异与 WAF 绕过

一、题目背景：

在当今网络安全防御体系中，Web应用防火墙（WAF）和入侵防御系统（IPS）等是保护Web应用免受恶意攻击的关键系统。这些系统通过分析传入的HTTP请求来识别并阻挡潜在的攻击。然而，随着攻击技术的不断进步和变化，传统的防御机制面临着日益严峻的挑战，尤其是在处理经过精心构造的变异HTTP载荷时，WAF的有效性可能会大打折扣。因此，研究和开发能够有效绕过WAF的技术，不仅能够揭示现有防御系统的潜在弱点，也对提升整体网络安全防御能力具有重要意义。

二、题目描述：

给定一组Web攻击的HTTP报文（攻击类型限定为文件上传、SQL注入、RCE），并给出WAF及直接访问Web服务的接口。选手对攻击报文进行分析，并与WAF及后端接口交互，变异出能绕过WAF的有效载荷。

交付内容：完整系统程序源码；系统设计相关文档等。

三、考察选手对HTTP协议、RFC规范、攻防对抗技术的理解，挑战内容如下：

1、设计并实现开箱即用的载荷层WAF绕过方法，能够自动产生变异的HTTP载荷。

2、变异后载荷能够绕过WAF的检测并成功触发后端漏洞。

3、选手需要考虑到不同的WAF实现、不同的后端应用环境（编程语言、版

本、框架）之间对HTTP请求处理的差异进行绕过。

四、评价方式：

1、初赛和决赛分别给出新的HTTP报文及WAF和后端服务，考察选手针对各种环境下变异出的有效载荷的个数。

2、创新性：选手是否有创新的思路、算法或方法在程序中得以应用。

3、加分项：与后端及WAF的交互次数较少。

4、初赛给出的附件用于选手系统设计和测试，决赛给出的附件用于评分。

五、咨询邮箱： xuxiaoqiang6@huawei.com

六：赛题互动交流答疑社区链接：

https://www.chaspark.com/#/races/competitions/1005989551778738176

题目4：WEB漏洞数据流分析与挖掘

一、题目背景：

在WEB源码审计过程中，通常会基于调用流正向分析的方法，从外部请求入口中逐层分析请求的处理过程，并判断请求处理过程中是否存在漏洞。现在需要设计一个程序，能够基于调用流分析的方法，识别并判断是否存在漏洞，以及漏洞的类型；

二、题目描述：

给定一个漏洞靶场（见附件），选手需要编写程序，自动化提取URL和入口函数，并基于URL入口识别出所有处理路径，并分析出处理路径中存在的漏洞点。

注：给定的源码为基于Spring框架二次开发的Java网站源码

三、考察选手编程能力，挑战内容如下：

1、识别URL和关联的入口函数，并针对项目代码识别出分析出所有的调用流；

2、识别调用流中的存在的漏洞，并标识漏洞所处的位置，并进行可达性分析；

四、评价方式：

1、识别出的URL入口和关联函数的准确性与数量；

2、针对每一个URL入口函数，分析出的调用流数量（忽略JDK中函数调用，但需

要步入第三方Lib中）；

3、分析出的存在漏洞的数据流的数量以及准确性；

加分项：函数调用静态分析中，针对变量值设计动态融合获取的方法并能增强调用流分析的准确性和多样性；

五、咨询邮箱： yuhao2@huawei.com

题目附件链接如下：

https://github.com/WebGoat/WebGoat/archive/refs/tags/v2023.8.zip

六：赛题互动交流答疑社区链接：

https://www.chaspark.com/#/races/competitions/1005990312268967936

题目 5：系统文件访问行为白名单规则学习算法比赛

一、题目背景：

当系统被入侵后，攻击者常常会运行异常的命令或访问文件获取或篡改数据。如何判定异常命令和异常文件访问是系统检测防护的难题。根据进程和文件访问白名单，可以有效过滤不在白名单中的系统进程文件访问行为，通过调查发现攻击。

例如某个系统中进程和文件有如下访问关系：

/usr/bin/sh /etc/ld.so.cache
/usr/local/bin/php /etc/passwd
/usr/local/bin/apache /etc/ld.so.cache
/usr/local/bin/apache /mnt/openfire/cache
/usr/local/bin/apache /mnt/openfire/cache/index.html
…

因为操作系统中进程和文件访问关系非常多，有千或万数量级，需要用一定的方法，如掩码，降低规则数量。如

/usr/local/bin/* /etc/ld.so.cache
/usr/local/bin/php /etc/passwd
/usr/ bin/* /etc/ld.so.cache

/usr/local/bin/apache /mnt/openfire/*

掩码可以使用在进程路径、被访问文件路径匹配上。可以使用在路径末尾、开头、中间任何位置。如：

/usr/*/bin/* /etc/*

/bin/ */ ld.so.cache

二、题目要求：

需要设计实现一个无监督学习算法，自动学习系统中的进程和文件访问关系白名单规则。通过聚类等方法降低白名单规则数量，一方面可以有效对白名单规则进行管理，另一方面在用白名单规则监控时能够快速发现异常发现速度。

但是如果白名单规则粒度太粗，会导致不能发现异常，需要进行平衡性考虑。

赛题提供了数据样例。附件：fileaccessdata

选手也可以自己构建数据集，在系统中运行各种服务，如 Apache，开发桌面等。

根据第五部分参考信息部分的方法自己构建数据集

三、交付件：

设计文档和测试文档说明

二进制、运行说明与测试方法

源代码

四、评价方式：

算法效率与准确度平衡：用较少的规则覆盖进程文件访问行为，同时规则要能较高准确度，不漏掉攻击。规则数和准确度同时达到较高水平的作品有较高分数。

算法适应性：使用了较好的聚类等方法，能够对未知应用系统进程文件访问行为，也能学习到较好的白名单规则。

算法性能：算法学习时间不能太长，越快产生结果越好。

五、参考信息：

在系统中使用一条 Linux audit 系统调用审计规则进行系统中进程与文件访

问数据记录，

auditctl -a always,exit -F arch=b64 -S open

使用 ausearch -i 输出解析后的信息

ausearch -i > /path/ausearchFile

使用下面 python 文件转换为进程与文件的访问关系

python3 getexefiles.py ausearchFile

附件：getexefiles.py

Linux ausearch 使用

https://blog.csdn.net/yunweimao/article/details/106687910

六、咨询邮箱：plato.liu@huawei.com

七：赛题互动交流答疑社区链接：

https://www.chaspark.com/#/races/competitions/1005991145673424896

题目 6：大模型安全专业能力评测系统

一、题目背景：

随着AI/AIGC技术的发展，越来越多垂直行业使用大模型赋能，网络安全行业也不例外：越来越多的网络安全垂直领域大模型（也可能是能力足够强大的通用大模型）发布并投入到实际网络安全工作辅助中，例如安全知识问答、辅助告警分析、辅助代码分析、辅助漏洞挖掘、辅助渗透测试等。现需要设计一套评估大模型的**安全专业能力**的评测体系。

注意：是评估大模型的安全专业能力！！！

二、题目要求：

选手需要设计评测大模型安全专业能力的系统，待评测的大模型以AIGC服务的方式提供，选手可自行搭建开源大模型测试环境或者使用在线AIGC服务测试。推荐使用试题的方式，选手自行设计题目和使用外部题库，所覆盖测试的能力维

度尽量全面。

三、评价方式：

1、覆盖尽可能多的网络安全专业能力维度；

2、方案思路较新颖，且能快速、准确地输出评测结果；

3、能够对在线AIGC服务进行自动化评测；

四、交付件：

1、评测题库及答案、程序源代码

2、设计和测试文档、部署测试方法

3、现网及开源大模型实测排行榜及专业能力维度分析

五、参考：

C-Eval

SecEval

SecBench

六、咨询邮箱：hupo1@huawei.com

七：赛题互动交流答疑社区链接：

https://www.chaspark.com/#/races/competitions/1005992064209223680

题目7：多模态大模型可靠性评估

一、题目背景：

随着stable diffusion的发展，多模态大模型越来越多地用在图像编辑、视频创作、游戏场景创作等众多领域，但是其生成内容的可靠性却少有关注。现需设计一套评估方案，能在白盒/黑盒场景下评估多模态大模型生成内容的可靠性。为了简化问题，本次比赛仅考虑图像编辑这一任务，即多模态大模型针对给定的输入文本针对输入的图像进行修改输出新的生成的图像。本次比赛提供10张清晰的目标图片以及相应的模糊化和加噪音后的图片，选手需设计算法生成输入图片和提示词，使得生成的图片x与目标图片x_t尽可能接近。对于每张图片，采用cosine距离和SSIM共同进行打分：

$$\text{score}(\text{x}, \text{x}_\text{t}) = \text{cosine}(\text{x}, \text{x}_\text{t}) + SSIM(x, x_t)$$

二、题目要求

选手可自行搭建文生图大模型环境，在正常输入、对抗输入等诸多场景下测试目标多模态大模型的输出可靠性，以生成尽可能接近指定内容的图片。

三、交付件：

1. 测试方案、结果及部署测试方法；
2. 生成目标图片的输入样例；
3. 输入生成算法。

四、评价方式：

1. 基本涵盖输入各项维度；
2. 方案具有创新性；
3. 能够对多模态大模型进行白盒/黑盒评估，并具有一定的迁移性。

五、咨询邮箱：wangxiaosen@huawei.com

六、赛题互动交流答疑社区链接：

https://www.chaspark.com/#/races/competitions/1005992796100034560