**赛题1：基于源码比对或二进制补丁比对的自动化漏洞发现技术**

**一、题目背景：**

传统的漏洞挖掘依赖人工分析，而人工分析往往耗时长、成本高且容易遗漏漏洞。通过引入人工智能模型，可以对源码和二进制补丁进行自动化对比，快速发现已修复的Nday漏洞，并达到一定的准确率要求。

**二、题目描述：**

对于主流的Linux发行版本内核，流行的、应用广泛的开源应用框架，中间件的二进制补丁或源代码commit进行监控，自动化的发现漏洞点，漏洞信息，实现PoC或者EXP的自动化实现。

结合AI的通用研究方法，多维数据融合分析：如结合源码commit日志、二进制文件、历史CVE数据库、项目Issue等多种信息来源。

融合策略：通过深度学习模型（如Transformer）自动化提取安全相关特征，统一处理不同维度来源的数据，实现漏洞模式精准识别：

课题可分为2个方向，源代码的补丁分析，和二进制补丁分析，可以任选其一。

研究方向1：对于开源软件，基于源码commit比对已知漏洞（Nday）发现

源码commit比对是通过分析软件项目的版本控制系统中的历史提交记录和更新内容，以自动化方式识别潜在漏洞的一种方法。可以结合人工智能技术，构建高效的漏洞自动发现工具。

可以从操作系统内核，基础库或者应用框架中选择比较流行的3~4个开源项目进行跟踪和发现，以23年24年的补丁为输入开始研究。

研究方向2：基于二进制补丁比对的Nday漏洞发现

通过对软件补丁发布前后的二进制文件版本进行比对，挖掘已修复的Nday漏洞。

研究内容：跟踪某个操作系统的补丁文件，通过静态和动态二进制分析技术，定位更新中修复的安全漏洞。

静态对比：对不同行版本的内核二进制文件使用函数级Diff工具进行分析。

动态对比：结合模糊测试，在内核系统调用接口中寻找潜在的漏洞路径。

**三、考察选手漏洞识别和挖掘能力，挑战内容如下：**

1、代码和补丁的分析能力，发现漏洞的能力。

2、挑战使用AI智能体，实现自动化的漏洞利用编写能力。

**四、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、 通过工具准确发现漏洞的漏洞列表，及其漏洞点。

**五、评价方式**：

1、源代码的可扩展性，可以通过简单配置，扩展监控领域：

2、能够通过系统，准确发现代码补丁或者二进制补丁中的漏洞率达到70%以上。

3、挑战：能够基于发现的漏洞，自动化实现PoC，exp的编写

**六、咨询邮箱：** zhangjiade@huawei.com

**七、赛题交流、答疑链接：**

https://www.chaspark.com/#/races/competitions/1124130549314236416