**华 中 科 技 大 学**

**网络空间安全学院**

本科：《网络空间安全综合实践V》

实验报告

姓 名 MTX

班 级 信安XXXX班

学 号 U2021XXXXX

联系方式

分 数

评 分 人

2024 年12月26 日

**报告要求**

**1. 报告不可以抄袭，发现雷同者记为0分。**

**2. 报告中不可以只粘贴大段代码，应是文字与图、表结合的，需要说明流程的时候，也应该用流程图或者伪代码来说明；如果发现有大段代码粘贴者，报告需重写。**

**3. 报告格式严格按照要求规范，并作为评分标准。**

**课程目标评价标准**

（1）课程目标 1的实验报告评价标准

|  |  |  |  |
| --- | --- | --- | --- |
| 实验报告评价标准 | | | |
| 优秀 | 良好 | 及格 | 不及格 |
| 实验报告内容详实，分析深入，条理清晰，能准确反映学生对网络攻击与防护原理的理解及重新实现过程。报告能详细阐述工业互联网领域访问控制、安全通信等问题的解决方案，以及对典型工控设备及安全设备的深入了解。 | 实验报告内容完整，分析合理，条理清晰，能反映学生对网络攻击与防护原理的基本理解及部分重新实现过程。报告对工业互联网领域访问控制、安全通信等问题有较为清晰的认识，并了解典型工控设备及安全设备。 | 实验报告内容完整，但分析不够深入，条理稍显混乱。对网络攻击与防护原理方面有基本了解，但未能充分展示其重新实现过程。报告对工业互联网领域访问控制、安全通信等概念有初步认识，但对工控设备及安全设备的了解不够深入。 | 实验报告内容不完整，分析错误或缺乏深度，条理混乱。在网络攻击与防护原理方面，缺乏深入了解，未能展示其理解及重新实现过程。报告对工业互联网领域访问控制、安全通信等概念不熟悉，对工控设备及安全设备缺乏了解。 |

（2）课程目标 2的实验报告评价标准

|  |  |  |  |
| --- | --- | --- | --- |
| 实验报告评价标准 | | | |
| 优秀 | 良好 | 及格 | 不及格 |
| 实验报告内容详实，分析深入，条理清晰，能准确反映在真实工业互联网中攻防技术的掌握程度以及解决复杂工程问题的能力。报告中详细记录了实验过程、遇到的问题及解决方法，展现了较高的专业素养和攻坚能力。 | 实验报告内容完整，分析合理，条理清晰。能反映自己在真实工业互联网中攻防技术的掌握情况，并尝试解决复杂工程问题。报告中记录了实验过程及部分解决方法，展现出一定的专业素养和攻坚能力。 | 实验报告内容基本完整，但分析不够深入，条理稍显混乱。报告能简单描述自己在真实工业互联网中攻防技术的掌握情况，但对解决复杂工程问题的能力有限。报告中仅记录了实验过程，缺乏深入的分析和解决方法。 | 实验报告内容不完整，分析错误或缺乏深度，条理混乱。报告中对真实工业互联网中攻防技术的掌握情况不明确，未能有效展示其解决复杂工程问题的能力。报告中缺乏实验过程的详细描述和分析。 |

（3）课程目标 3 的实验报告评价标准

|  |  |  |  |
| --- | --- | --- | --- |
| 实验报告评价标准 | | | |
| 优秀 | 良好 | 及格 | 不及格 |
| 报告内容详尽，逻辑清晰，深入分析关键信息基础设施安全保护的法律制度体系。对于网络安全和防御技术的正确运用，报告能全面阐述并给出有效的法律建议，展现出强烈的社会责任感和使命感。 | 报告内容完整，逻辑较清晰，能准确描述关键信息基础设施安全保护的法律制度体系。对于网络安全和防御技术的正确运用，报告有一定的分析，但法律素养和责任感不够强。 | 报告内容基本完整，但描述较为简单，对关键信息基础设施安全保护的法律制度体系的理解不够深入。对于网络安全和防御技术的正确运用，报告有初步涉及，但缺乏详细分析。 | 报告内容不完整，逻辑混乱，对关键信息基础设施安全保护的法律制度体系缺乏理解。对于网络安全和防御技术的正确运用，报告几乎没有涉及。 |

**报告评分表**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **评 分 项 目** | | | **满分** | **得分** | **评分标准** |
| 实验  内容 | 任务分析（目标1） | | 10 |  | 10-8：能够给出明确实验需求分析，系统功能完整、正确和适当。  7-4：能够给出需求分析，但不够完整，能够阐述系统的设计，但不够完整、恰当和准确。  3-0：需求分析不够明确，系统设计不够完整、正确和恰当。 |
| 方案设计（目标2） | | 10 |  | 10-8：方案设计完整、正确和适当，流程设计正确规范。  7-4：方案设计基本完整，流程设计基本正确。  3-0：方案设计不完整，流程设计有错误。 |
| 操作过程（目标1） | | 25 |  | 25-17：清晰描述整个实验过程，确保完整性，对工具运用的操作准确性，提出创新思路或改进方法。  16-9：能够基本按指导步骤进行实验，但可能存在部分遗漏或简化，表现出一定思考，但未能将思考转化为具体的创新实践或改进方案。  8-0：描述的实验步骤和观察到的现象不完整，缺乏关键信息或存在明显的错误，未能对实验方案进行任何形式的改进或优化。 |
| 问题描述及解决方案（目标2、3） | | 15 |  | 15-10：遇到的问题及解决方案真实具体。  9-5：遇到描述不够详细，解决方案不够具体。  4-0：没有写什么内容。 |
| 感想（含思政）（目标3） | | | 15 |  | 15-10：感想真实具体。  9-5：感想比较空洞。  4-0：没有写什么感想。 |
| 意见和建议（目标2） | | | 15 |  | 15-10：意见和建议有的放矢。  9-5：意见和建议不够明确。  4-0：没有写什么内容。 |
| 文档格式（段落、行间距、缩进、图表、编号等）（目标1） | | | 10 |  | 基本要求：目录、标题、行间距、缩进、正文字体字号按照模板要求执行，图、表清晰且有标号。  10-8：格式规范美观，满足要求。  7-4：基本满足要求。  3-0：格式较为混乱。 |
| **总分** | | | **100** |  |  |
| 教师签名 | |  | | 日 期 |  |

目 录

[1 实验需求分析 4](#_Toc1043322991)

[1.1 实验名称 4](#_Toc925531982)

[1.2 实验任务 4](#_Toc1191966253)

[2 实验过程 5](#_Toc1649354955)

[2.1 实时流量分析实验过程 5](#_Toc989813209)

[2.2 城市水务系统网络安全攻防虚拟仿真实验攻击思路 14](#_Toc1382274001)

[2.3 城市水务系统网络安全攻防虚拟仿真实验主要攻击过程 15](#_Toc401041561)

[2.4 智慧校园网络安全实验攻击思路及防守思路 19](#_Toc1501831441)

[2.5 实验中的漏洞利用及其防御方法 21](#_Toc1905725696)

[3 分工与合作 25](#_Toc1960661314)

[3.1 本人在本次小组实验中完成的工作 25](#_Toc1845624830)

[3.2 小组实验成员工作完成情况 25](#_Toc1162720542)

[4 总结与思考 27](#_Toc1890445341)

[4.1 针对Metasploit渗透测试实验的改进意见与建议 27](#_Toc694288822)

[4.2 针对实时流量分析实验的改进意见与建议 27](#_Toc1633577203)

[4.3 针对城市水务系统网络安全攻防虚拟仿真实验的改进意见与建议 27](#_Toc2101107573)

[4.4 针对智慧校园网络安全实验的改进意见与建议 27](#_Toc93888143)

[4.5 实验过程中遇到的难点及解决方案 28](#_Toc1725022503)

[4.6 对关键信息基础设施安全保护的理解 28](#_Toc1423973421)

[4.7 其他感想 29](#_Toc1163524579)

# 实验需求分析

## 实验名称

实验一：Metasploit渗透测试实验。

实验二：实时流量分析实验。

实验三：城市水务系统网络安全攻防虚拟仿真实验。

实验四：智慧校园网络安全实验。

## 实验任务

实验一：学习使用渗透测试工具Metasploit，掌握几种渗透测试技术，具备网络系统安全分析能力，具体完成系统级渗透与安全防护实验，搜寻实验靶机的系统漏洞，并通过实验攻击机对实验靶机进行攻击。

实验二：深入了解并掌握实时流量分析、IP地址查询、CDN分析及数字证书解析等网络安全和数据分析技术，具体完成IP地址国别信息查询系统设计、实时分析流量的CDN、数字证书来源分析等实验。

实验三：学习Web安全、典型工控通信协议及漏洞利用，具体从水务官网服务器IP入手，利用各类攻击工具逐步渗透，侵入水务官网服务器主机和水务系统工控上位机，完成排水阀启停、高压泵转速攻击。

实验四：通过“大学电子邮件系统发现含木马程序的钓鱼邮件”事件背景展开，利用各类漏洞，逐步对数据中心各服务器发起攻击，学习安全设备配置，学习攻防技术分析。

# 实验过程

## 实时流量分析实验过程

### 任务一 IP 地址国别信息查询系统设计

实验资料中，xdb\_search\_ipv4.cc 和 xdb\_search\_ipv6.cc 分别负责实现 IPv4 和 IPv6 地址的查询逻辑，通过使用二分查找算法和优化的索引结构，实现对 IP 地址地理信息的高效定位。而 IP2Region.cc 提供了 IP 地址到地理信息的映射功能，整合了查询算法，并对外提供了相应的接口支持。main.cc 作为程序的入口文件，主要负责系统的初始化、接收输入的 IP 数据、调用相关查询逻辑，并输出最终的查询结果。

在目前的程序中，Python 和 C++ 之间的通信是通过 Socket 实现的。这种方式虽然可以依托 C++ 的高效性，在一秒内解析多达 1,000,000 个 IP 地址，并能在高流量场景下保持可靠的输出，但 Socket 通信并不是最佳的解决方案。为了进一步优化性能，可以采用一种更加直接和高效的方法，可以使用 Python 的外部函数接口，让 Python 直接调用 C++ 的类和方法，而无需依赖 Socket 传递数据。具体操作是将 C++ 代码编译为共享库文件（.so 文件），然后通过 Python 的 ctypes 或 cffi 库加载该共享库，并直接调用 C++ 的功能接口。这种方法既能发挥 C++ 的性能优势，又能显著简化通信流程，提高程序整体效率。

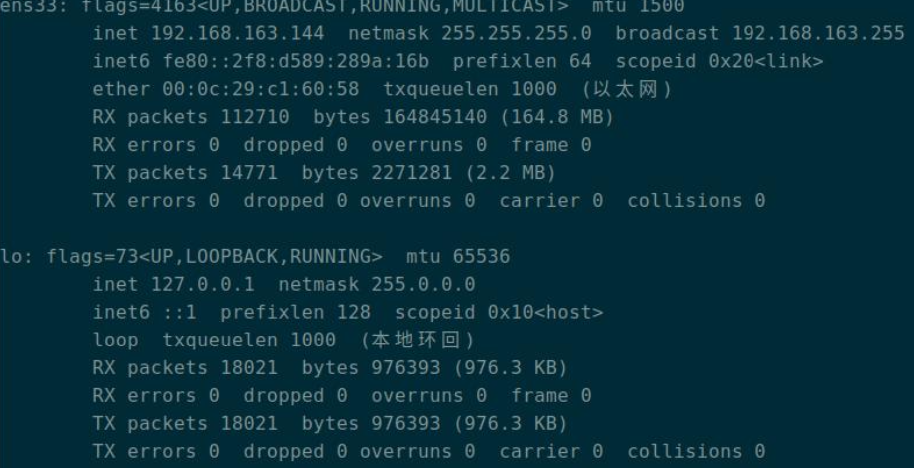


图2-1 查询网口

查询网口后，我们编写下图代码，这段代码通过封装 C++ 的 IP2Region 类，提供了三个用于查询 IP 地址归属地的跨语言调用接口，目的是方便其他语言（如 Python）通过共享库直接调用这些功能。代码采用了 extern "C" 关键字，将 C++ 的方法暴露为 C 风格的接口，确保编译后的共享库可以被不同语言加载使用。具体而言，IP2Region\_new 函数负责根据提供的 IPv4 和 IPv6 数据库文件路径创建并初始化一个 IP2Region 对象，用于查询 IP 地址的地理信息；IP2Region\_delete 函数用于释放创建的对象资源，避免内存泄漏；IP2Region\_search 函数则通过传入的 IP2Region 对象和目标 IP 地址字符串，查询该 IP 地址的地理归属信息，并返回结果。这种封装方式不仅提高了代码的复用性，还通过简化的接口设计，保证了跨语言调用的便利性和高效性，为构建基于 IP 地址查询的系统提供了核心支持。

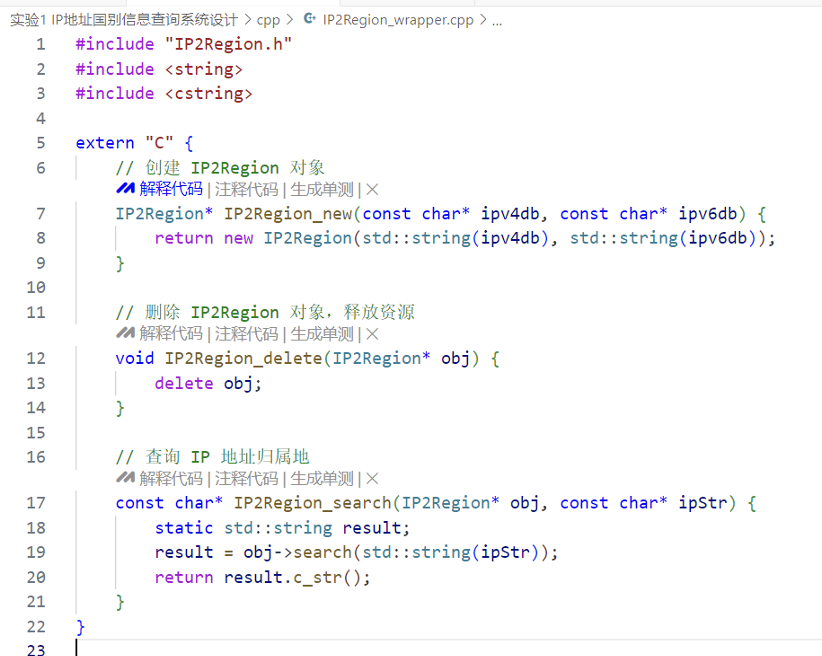


图2-2 IP2Region\_wrapper.cpp代码

其中IP2Region\_search 函数的作用是提供一个简单的接口，用于查询指定 IP 地址的地理归属信息。它接收两个参数，一个是已经初始化的 IP2Region 对象指针（obj），另一个是需要查询的 IP 地址字符串（ipStr）。函数内部首先使用传入的 IP 地址创建一个 std::string 类型的对象，然后调用 IP2Region 对象的 search 方法进行查询，将查询结果存储在一个静态的 std::string 变量中。最终，该变量的 C 风格字符串（const char\*）形式被返回，供调用者使用。通过这种设计，该函数将 C++ 的类方法封装成了跨语言可调用的 C 风格接口，使其他语言（如 Python）可以直接使用共享库来高效查询 IP 地址的归属地，同时避免了内存管理的复杂性。

然后我们只需要修改python文件中get\_ip\_location函数，将其更改为直接调用IP2Region\_search进行查询，接下来开始捕获并查看捕获结果。

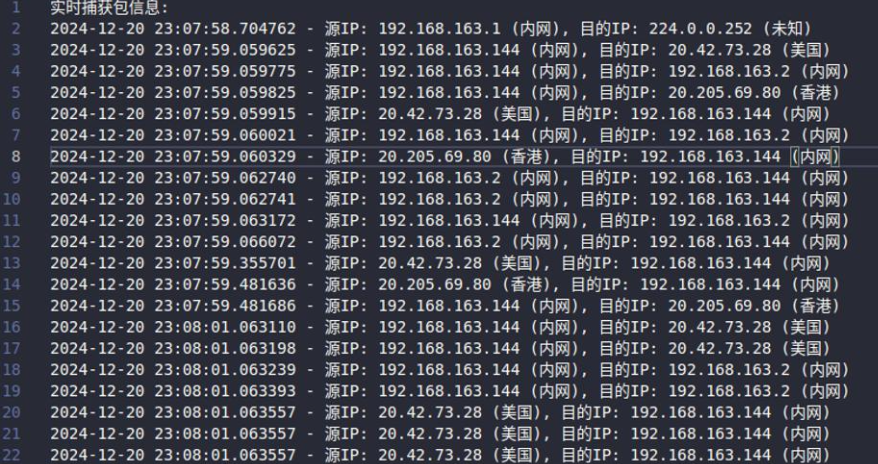


图2-3 查看数据包信息

### 任务二 实时分析流量的CDN

#### 查询CDN

首先，利用在线工具手动查询网站CDN服务商，结果如下：

表2-1 网站CDN查询表

|  |  |
| --- | --- |
| 网站 | CDN 服务提供商 |
| www.geekflare.com | Cloudflare |
| www.github.com | Microsoft, Fastly |
| www.youtube.com | Google |
| www.bilibili.com | ChinaNetCenter/Wangsu |
| www.facebook.com | Facebook |
| www.taobao.com | Alibaba |
| www.zhihu.com | Tencent Cloud |
| www.qq.com | Akamai, Tencent Cloud |
| www.yahoo.com | Yahoo, Amazon CloudFront |

#### 获取官方网站的IP地址范围

然后根据附件find.py文件编写两个脚本来爬去Cloudflare和Amazonaws两个网站的ip地址。

首先是从 Cloudflare 的官方地址（https://www.cloudflare.com/ips-v4）获取 Cloudflare 的 IPv4 地址列表，并将这些地址保存到本地文件 cloudflare\_ips.txt 中。具体来说，代码使用 requests 模块发送一个 HTTP GET 请求，从指定的 URL 获取包含 IP 地址的响应数据，并将数据按行分割成一个列表。然后通过遍历每一行的内容，逐一提取有效的 IP 地址并存储到一个新的列表 ip\_list 中。最后，代码将这些 IP 地址写入到一个名为 cloudflare\_ips.txt 的文件中，文件中包含 IP 地址的总数以及每个 IP 地址的具体内容。运行代码后，用户可以在本地查看并使用保存的 Cloudflare IP 地址列表，可以看到一共生成了1524736个IP。

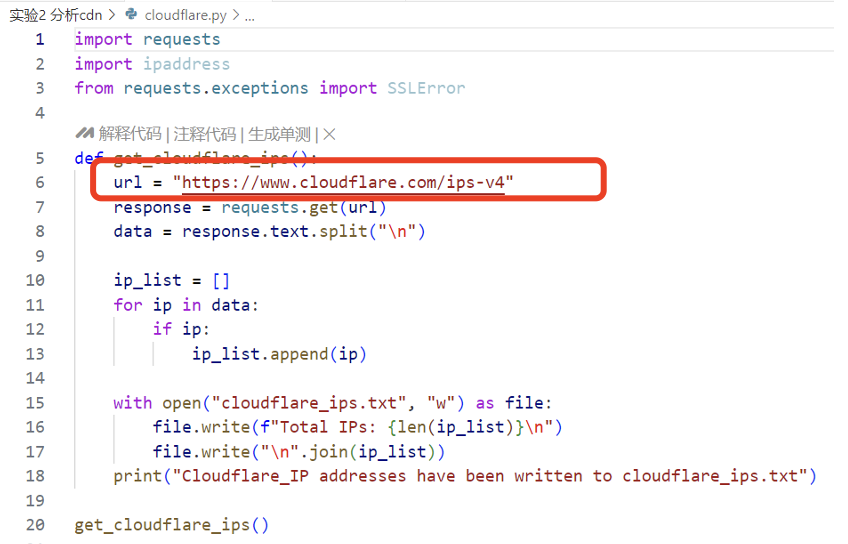


图2-4 获取Cloudflare的IP地址相关代码

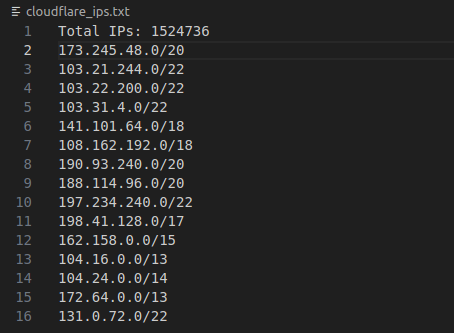


图2-5 Cloudflare网站爬取结果

然后是从 Amazon AWS 的官方 API（https://ip-ranges.amazonaws.com/ip-ranges.json）获取 AWS 的 IP 地址范围，并将这些地址写入到本地文件 amazonaws\_ips.txt 中。代码通过 requests 模块发送 HTTP GET 请求以获取 JSON 格式的 IP 范围数据，并使用 raise\_for\_status() 方法确保请求成功。接着，从返回的 JSON 数据中提取 prefixes 字段下的 ip\_prefix 数据，这些数据表示 AWS 的 IP 地址范围。提取的 IP 范围存储在列表 ip\_prefixes 中。随后，将这些 IP 地址范围写入到文件 amazonaws\_ips.txt 中，文件内容包括 IP 总数和每个 IP 前缀的具体内容。运行代码后，用户可以在本地查看和使用保存的 AWS IP 范围信息，可以看到一共生成了152700193个IP。



图2-6 获取Amazonaws的IP地址相关代码

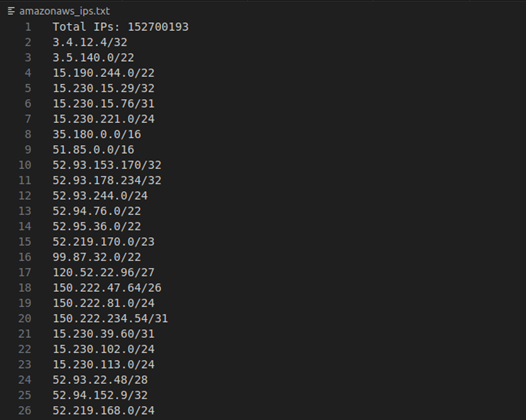


图2-7 AmazonAWS网站爬取结果

#### 实时流量分析

完善Capture\_demo.py,将interface字段改为本机的网口名称，运行脚本进行分析。

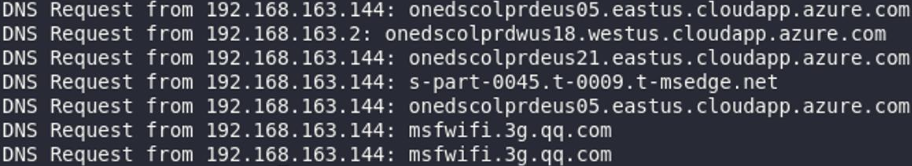


图2-8 实时捕获网络流量

#### 挑战任务



图2-9 自动化CDN查询片段

这段代码的主要功能是捕获 DNS 请求，并查询这些请求对应的 CDN 服务提供商，然后将结果保存到本地文件中。具体来说，代码首先定义了 query\_cdn\_provider 函数，通过调用一个 API（https://api.cdnprovider.com/check）查询指定域名的 CDN 服务提供商。接着，利用 pyshark 库通过网络接口 ens33 抓取 UDP 53 端口的 DNS 流量，并从数据包中提取域名信息。提取的域名被存储到 dns\_domains 列表中。随后，代码对捕获的域名去重，并为每个唯一域名调用 query\_cdn\_provider 函数查询对应的 CDN 提供商，将结果存储在字典 cdns\_list 中。最后，代码将域名和对应的 CDN 提供商信息写入到本地文件 cdn\_providers.txt 中，并打印保存成功的提示信息。通过这段代码，可以快速分析网络中的 DNS 请求并获取对应的 CDN 服务提供商信息，其结果如下。

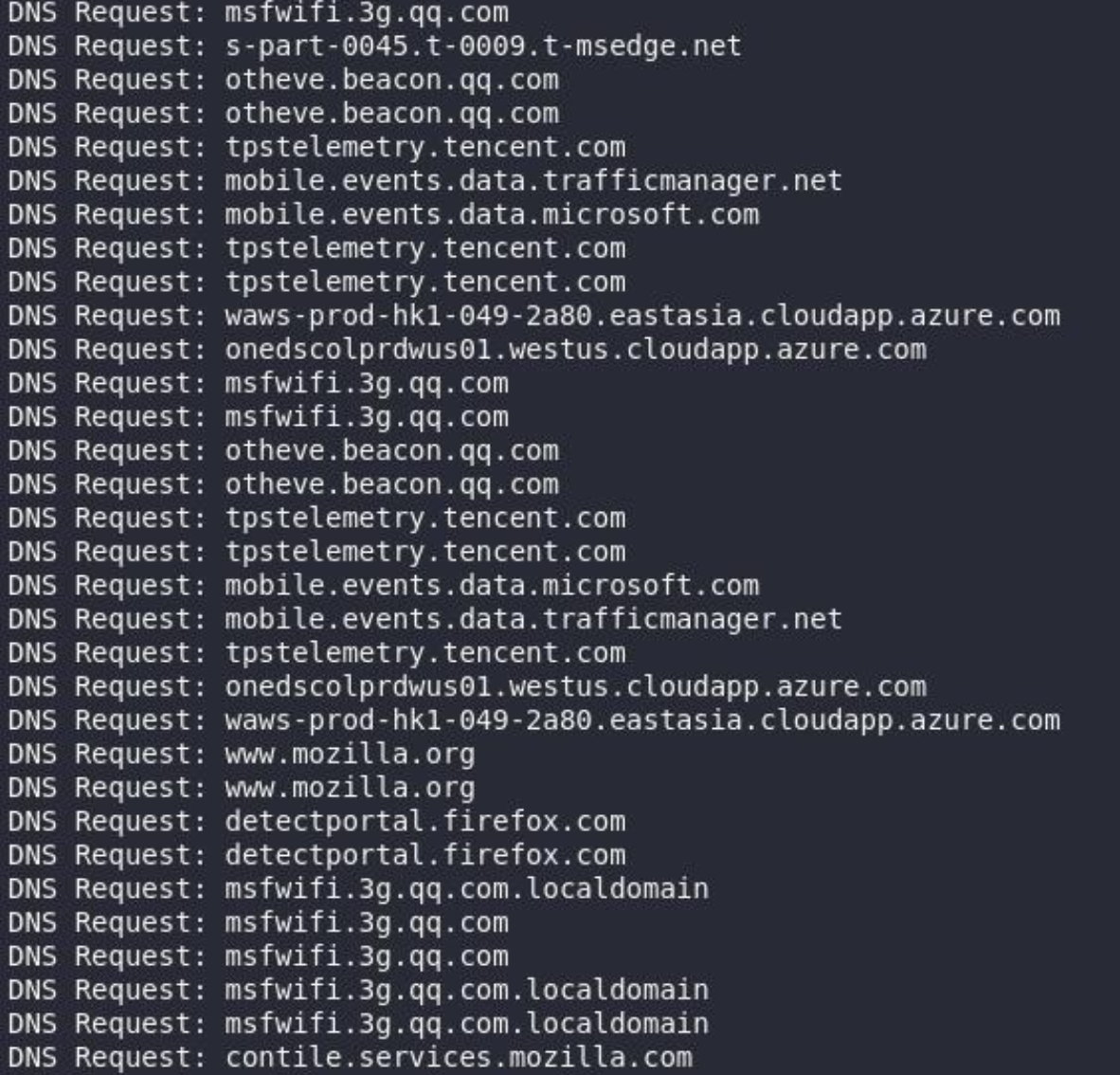


图2-10 自动化CDN查询结果

### 任务三 数字证书来源分析

#### 提取数字证书并查看信息

实验使用浏览器访问目标网站，同时在 Wireshark 中对特定网络接口（如 ens33）进行流量监听。为了高效筛选出与数字证书相关的流量，设置了过滤条件 tls.handshake.type==11，使得显示的仅为握手阶段的 TLS 数据包。在抓包数据中，选择“跟随 TLS 流”，可以查看目标网站与客户端交互的完整通信过程。于是就能正确定位到包含数字证书的 TLS 数据包。

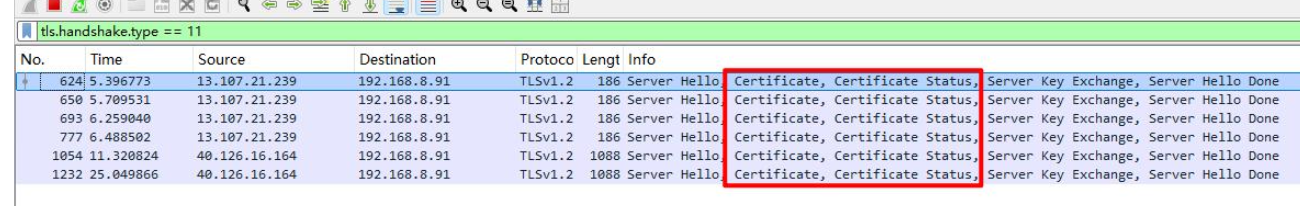


图2-11 Wireshark捕获证书



图2-12 Certificate字段

展开其 TLS 层的信息，找到 Certificate 字段。该字段包含目标网站的数字证书信息，可以以十六进制格式呈现。实验进一步将这些十六进制数据复制下来，并保存为一个名为 ca.txt 的文本文件，以备后续分析。

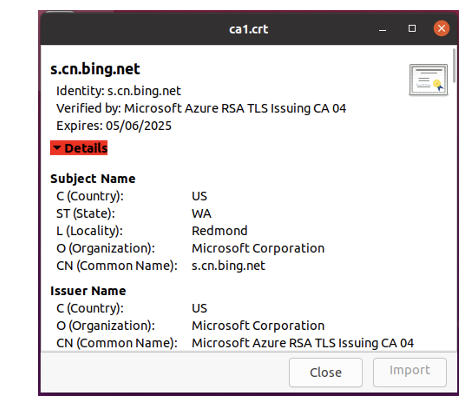


图2-13 手动提取证书截图

#### 自动化实现数字证书的提取

利用自动化脚本analysisi.py，访问网址提取证书。

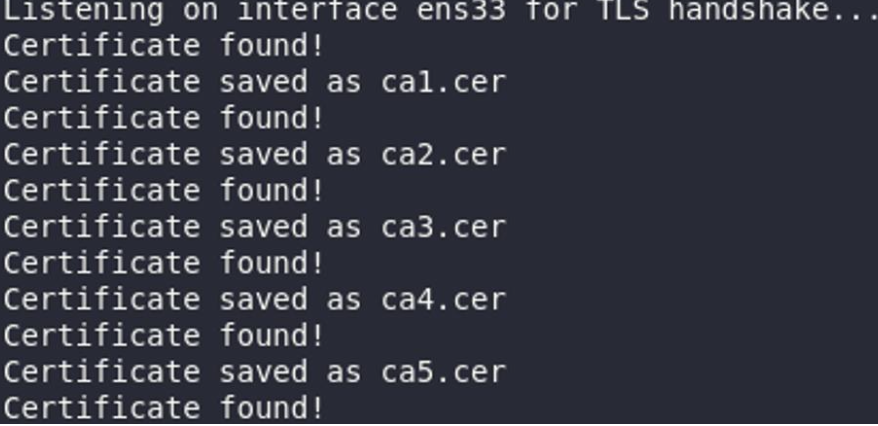


图2-14 自动提取证书

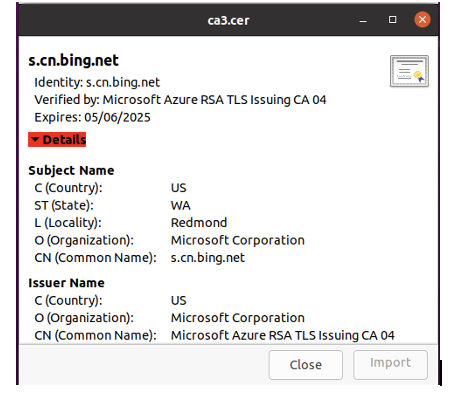


图2-15 自动提取证书截图

可以发现自动提取的证书和手动一样。

## 城市水务系统网络安全攻防虚拟仿真实验攻击思路

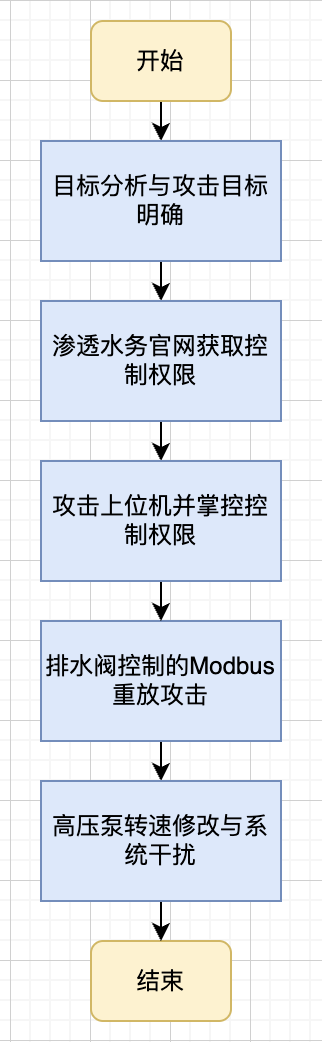


图2-16 城市水务系统攻击流程图

在城市水务系统的网络安全攻防实验中，攻击者通常从水务官网入手，进行系统渗透。首先，使用 Nmap 或 Zenmap 工具对水务官网服务器的端口进行扫描，识别出开放的端口及其对应的服务。通过访问HTTP服务端口（通常是80端口），攻击者可以进一步分析网站架构并定位管理员后台登录页面。接着，利用工具如御剑批量扫描，定位后台路径，结合字典文件进行路径枚举。找到管理员登录页面后，攻击者通过查看页面源码探测潜在漏洞，如未授权访问漏洞或phpMyAdmin的路径，从而尝试利用默认用户名和密码或暴力破解手段登录后台，获取管理员权限。一旦成功获取管理员权限，攻击者可通过文件上传漏洞上传恶意PHP脚本（如Webshell），实现远程代码执行，并使用提权工具如PrintSpoofer.exe 创建新用户，获得更高的系统权限，最终提取桌面上的flag文件。

接下来的攻击阶段是上位机攻击，攻击者首先通过远程桌面登录上位机，使用默认管理员账户激活FTP服务。完成激活后，攻击者对FTP服务进行弱口令暴力破解，利用常见的弱口令字典获取FTP账户信息。在成功登录FTP服务后，攻击者可以浏览服务器文件结构，找到包含“工程密码”的文件，提取出用于控制上位机的工程账户和密码。通过这些凭证，攻击者可以登录上位机控制界面，完全控制上位机的运行状态，包括启动高压泵、调整频率等操作。

在控制上位机后，攻击者进一步利用Modbus协议进行排水阀控制攻击。通过Wireshark工具抓取上下位机之间的通信报文，分析Modbus协议中的控制指令。攻击者通过重放攻击，即截获合法的Modbus指令并修改其中的控制字段，发送伪造的控制命令来控制排水阀的开关状态。这种攻击方式不需要系统认证，可以模拟合法操作，影响工业控制系统的运行。

最后，攻击者还可能对高压泵进行攻击，通过修改Modbus协议中的寄存器数据来改变高压泵的运行频率。攻击者使用工具如ModScan32观察高压泵相关寄存器的地址和数据内容，修改其中存储的频率值，进而控制高压泵的转速。这种攻击可以通过抓取和修改Modbus报文，使用Python脚本模拟发送伪造的写寄存器指令，成功改变设备的工作状态。

整个实验的攻击过程模拟了多种可能的攻击方式，测试了水务系统在遭遇不同层级攻击时的脆弱性，从官网渗透、上位机控制到排水阀与高压泵的控制，每一步都展示了攻击者如何利用网络协议的漏洞及系统配置的缺陷，进行逐步渗透与控制，为反制措施的设计提供了重要依据。

## 城市水务系统网络安全攻防虚拟仿真实验主要攻击过程

在小组中，我负责了实验1和实验3的完成。

### 水务官网渗透

首先，使用Zenmap扫描目标靶机，即对水务官网服务器IP进行端口扫描，发现其开放端口有80（http），135（mscpc），443（https），3306（mysql）等

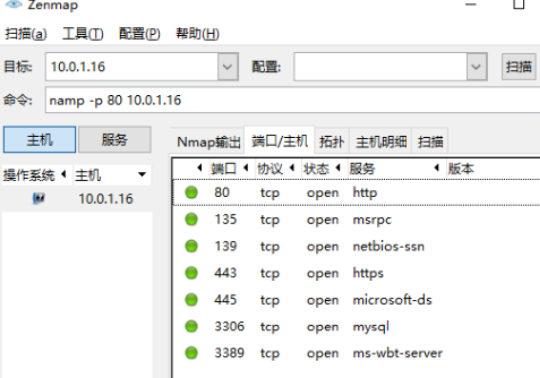


图2-17 开放端口

尝试直接访问web端口进行测试，没有在主页发现明显漏洞



图2-18 直接访问web进行测试

然后我们使用“御剑”扫描web目录，找到admin，登陆后台

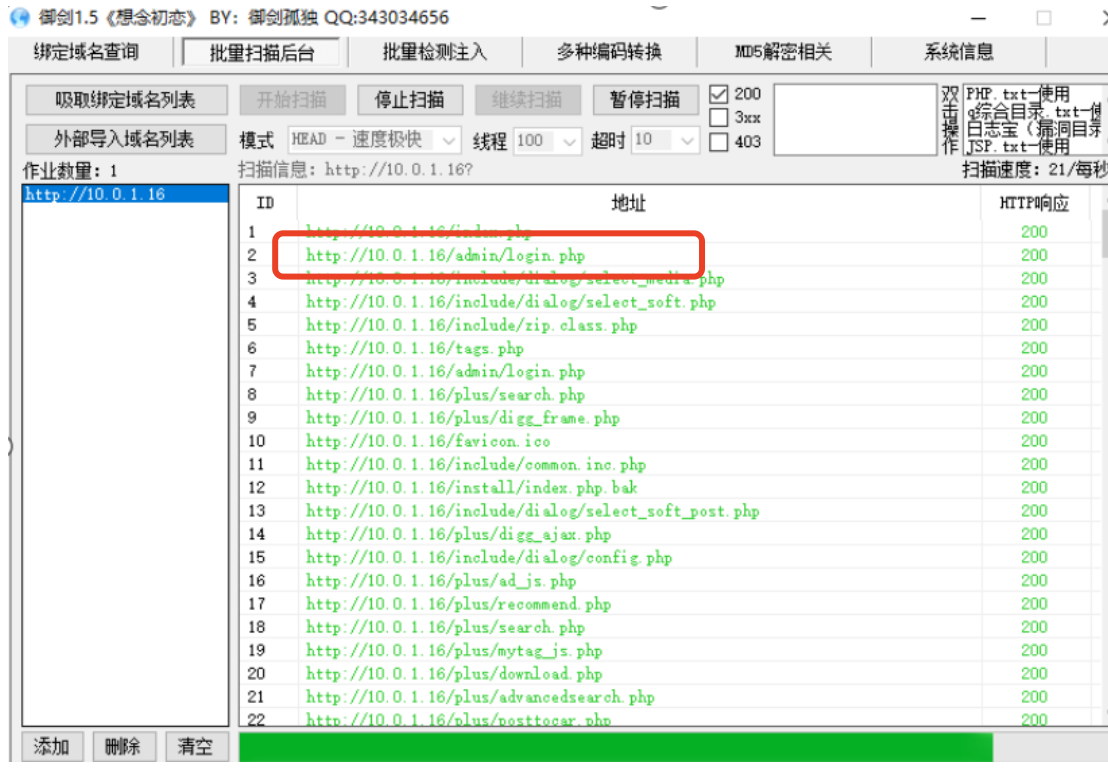


图2-19 御剑扫描结果

访问admin/login.php页面源代码，发现有一个提示DebugPage in /Pma\_Debug

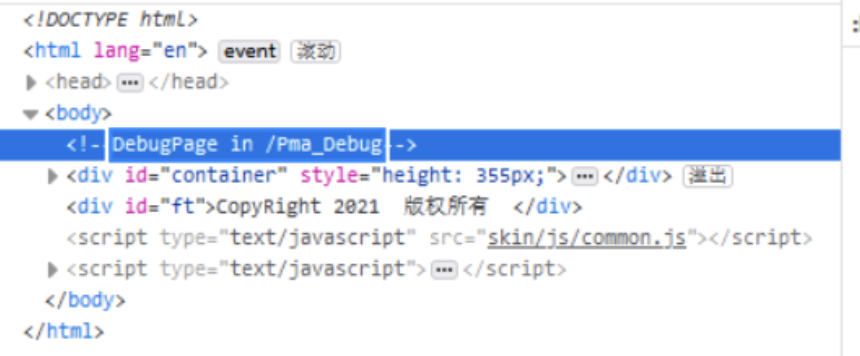


图2-20 提示

紧接着，访问对应的提示页面，进入phpMyAdmin的后台，即可修改管理员密码了

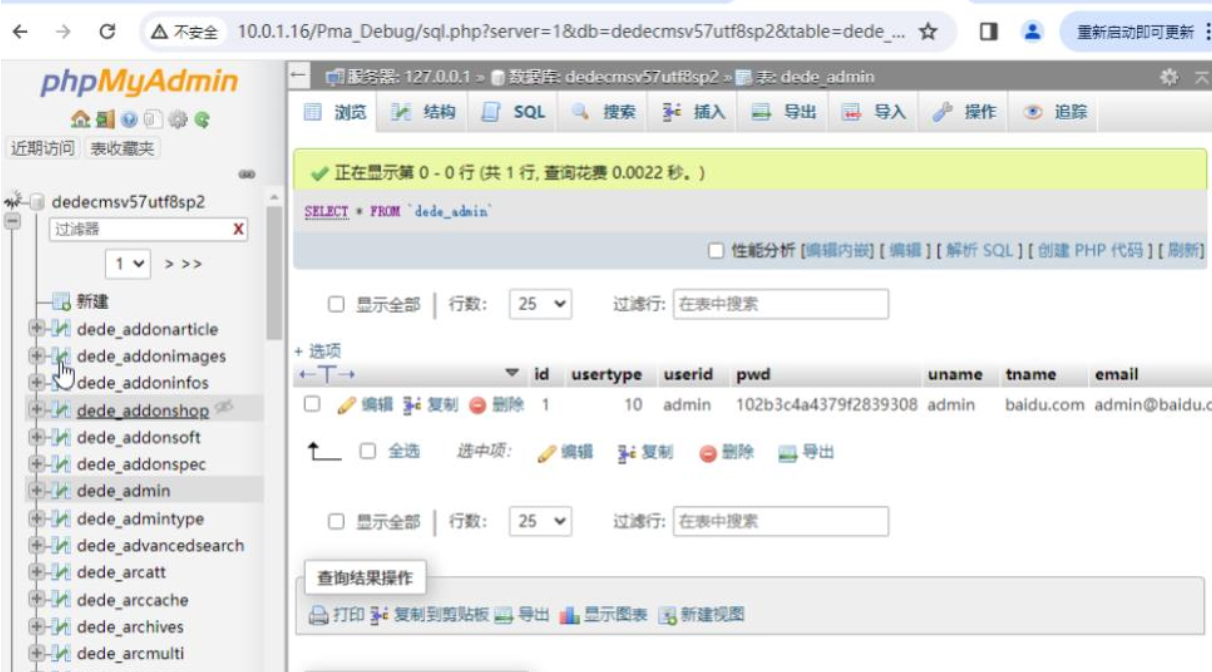


图2-21 phpMyAdmin后台

在后台登录页面找到数据库接口后，利用弱口令字典进行暴力破解操作，成功获取管理员的登录凭据。随后，通过 burpsuite 工具捕获请求数据并分析密码加密方式，确认是否能够通过 PHPMyAdmin 直接访问数据库以提取管理员密码。发现目标为 phpmyadmin 未授权访问状态，可直接修改 dedecms 数据库中的管理员密码字段，例如执行语句 UPDATE \dede\_admin` SET `pwd` = 'c3949ba59abbe56e057f' WHERE `dede\_admin`.`id` = 1;`，或者通过手动方式更改 pwd 字段值后保存更改。此时，管理员后台密码被重置为 123456，可以使用账户 admin 和密码 123456 成功登录后台。

接下来，获取远程控制权限，通过 burpsuite 修改 HTTP 请求以上传恶意 WebShell 文件，从而实现远程代码执行。随后，利用远程工具连接目标服务器，并通过执行命令的方式添加一个远程访问账户，如以下命令所示：

PrintSpoofer.exe -c "net user hack Admin@123 /add"

PrintSpoofer.exe -c "net localgroup administrators hack /add"

完成操作后，使用新建的账户登录目标服务器，查找 administrator 用户桌面上的文件，并成功找到和提交 flag 文件。

### 排水阀攻击 - 启动/停止排水阀

首先，根据实验文档的要求，对上位机与PLC控制器进行配置，确保两者之间的通信正常。登录上位机后，启动 Wireshark 抓包工具，对上位机与下位机之间的通信数据进行捕获，目的是提取和分析 Modbus 协议的数据包，为接下来的重放攻击提供技术支持。在上位机的组态界面中，找到排水阀的控制选项，并点击“启动”按钮，以触发排水阀的开启操作。同时，停止 Wireshark 的抓包，以确保抓取到的数据仅限于排水阀启动时的 Modbus 通信。随后，在 Wireshark 中使用特定的过滤条件（如 modbus.func\_code == 5）对抓包数据进行筛选，进一步过滤掉其他无关数据包，只提取与排水阀控制相关的 Modbus 数据包，从中获取功能码和相应的写入数据值，确保数据的精准性。

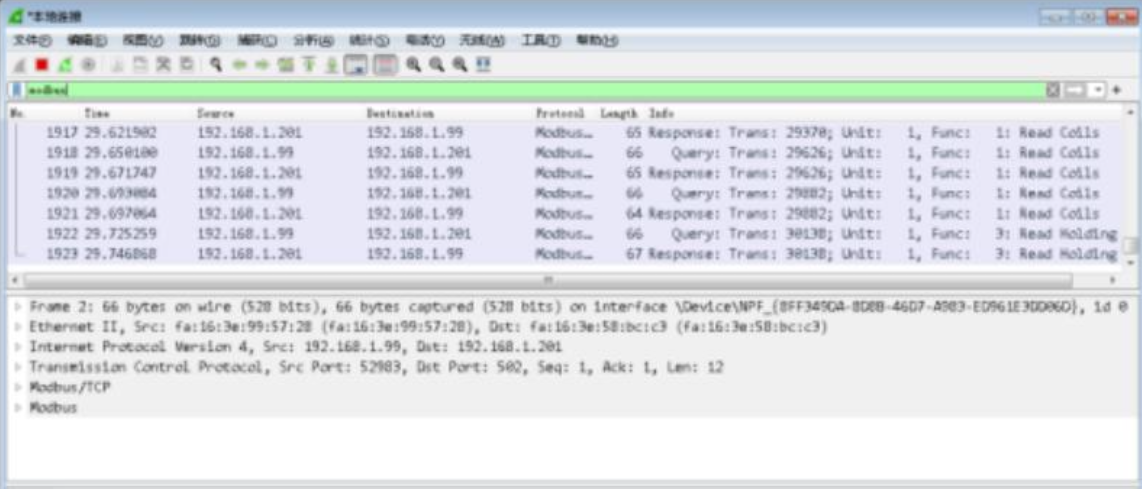


图2-22 提取数据包

通过分析捕获到的通信数据，可以发现大量流量中存在 func\_code 为 1 和 3 的数据包，表示读取功能的通信内容。这些与排水阀的控制操作无关，因此需要将它们过滤掉，专注于与写操作相关的数据包。在明确了启动排水阀所需的通信格式和参数之后，基于这些关键信息开发一个 Python 脚本。该脚本通过 Modbus 协议的重放攻击机制，模拟向下位机发送完全相同的控制数据包，能够精确地执行排水阀的启动或停止操作。脚本设计中应包含对抓包数据的解析与重构功能，以确保发送的伪造数据包与原始数据包保持一致。



图2-23 模拟数据包脚本

最后，通过运行该脚本，并观察排水阀的实际动作状态，可以验证攻击的成功与否。如果排水阀按照预期完成启动或停止操作，说明重放攻击已成功实施。此过程展示了通过抓包、协议分析、脚本开发和控制命令注入的完整攻击流程，揭示了工业控制系统在通信安全方面的潜在漏洞，同时为防御措施的设计提供了重要的实践依据。

## 智慧校园网络安全实验攻击思路及防守思路

在智慧校园网络安全实验中，攻击方（红方）的思路模拟了一次全面且层层递进的网络渗透与控制过程。从最初的外部入口突破，到内网核心资源的深度控制，再到隐匿行为掩盖攻击痕迹，红方通过技术和策略的结合，完成了对智慧校园网络的深入渗透和破坏性操作。同时，防守方（蓝方）则以应对红方攻击为目标，建立起从监控、溯源到系统加固的完整防御链条，全面强化网络安全。

红方的攻击首先从外部入侵点展开，例如智慧校园的教学区Web服务器。红方利用各种技术手段对目标进行侦察，包括端口扫描、目录猜解、服务识别以及漏洞检测。通过这些信息收集，红方可能发现服务器存在文件上传漏洞、弱口令登录或其他未修补的漏洞。在确认漏洞后，红方进一步生成一个包含反向连接功能的木马文件，将其上传到目标Web服务器中，并利用文件上传漏洞获取WebShell。这一步使红方得以在目标服务器上执行命令，初步实现对外部服务器的控制。

随后，红方利用内网扫描工具（如Nmap）对内部网络的拓扑结构进行分析，试图发现更多的攻击目标。通过扫描，红方可能识别出存在未授权访问的服务（如Redis、SMB）或其他高危漏洞（如永恒之蓝、Log4j）。针对这些目标，红方可以采用漏洞利用工具进行横向渗透。例如，通过永恒之蓝漏洞，红方可以直接溢出目标主机并获取SYSTEM权限，进一步扩大对内网的控制。在这一过程中，红方会利用已控制主机作为跳板，不断向内网深处渗透，从而逐步接近域控制器或其他核心资源。

为了提升攻击的效果，红方还会结合社会工程学手段，如发送钓鱼邮件或伪造系统通知，诱骗用户下载恶意文件或泄露登录凭据。这些手段能够帮助红方获取更多的账户信息，并通过合法身份进一步深入目标系统。在控制了域控制器后，红方几乎掌握了整个智慧校园网络的核心资源，可以访问所有用户和设备。此时，红方可能通过创建隐藏账户、安装后门文件或利用日志清理工具隐匿攻击痕迹，从而确保后续的持续访问和控制能力。

红方在攻击的最后阶段，通常会实施破坏性操作。例如，红方可能通过内网的受控主机向更多设备传播恶意软件，或者对敏感数据进行加密并勒索赎金。同时，红方可能利用控制权限锁定系统资源，使网络服务不可用，甚至直接瘫痪校园网络。通过这些手段，红方不仅实现了对智慧校园网络的全面破坏，还为防守方留下了复杂的修复与加固工作。

另一方面，蓝方作为防守方，主要目标是对红方的攻击行为进行监控、溯源和应急响应，同时在攻击结束后完成全面的系统修复与防御体系的优化。蓝方首先通过初始化网络环境和部署基础防护设备，为整个网络的安全运行建立坚实的基础。例如，蓝方会在实验环境中配置防火墙、入侵检测系统（IPS）、Web应用防火墙（WAF）和态势感知系统，监控并限制网络流量的异常行为。通过精心设定访问控制列表（ACL）和NAT规则，蓝方能够有效防止外部流量对内网敏感区域的直接访问，从而缩小攻击面。

在面对红方攻击时，蓝方会通过多种手段进行应急响应。首先，通过分析WAF、IPS和Sysmon的安全日志，蓝方能够发现红方的攻击行为特征，如SQL注入、目录遍历和恶意脚本执行等。结合这些日志信息，蓝方可以判断攻击者利用了哪些漏洞以及攻击路径。例如，通过IPS的告警，蓝方可能发现红方利用永恒之蓝漏洞对内网主机进行了渗透攻击，或者通过暴力破解试图获取远程桌面服务的访问权限。

在攻击发生后，蓝方会迅速采取措施阻止攻击的进一步扩展。例如，蓝方可以根据WAF和IPS日志中记录的攻击源IP地址，通过防火墙规则阻断恶意IP的访问，切断攻击路径。同时，蓝方会检查系统中是否仍然存在恶意文件或隐藏账户，并通过杀毒软件进行全面扫描，将潜在威胁隔离或清除。此外，蓝方会对所有账户强制修改高强度密码，关闭不必要的服务（如远程桌面服务）和高风险端口，并对共享文件夹进行权限审查和清理。

在系统修复阶段，蓝方会通过安装官方补丁修复已知漏洞，如永恒之蓝漏洞补丁或Log4j安全更新，并利用系统快照功能对比系统在攻击前后的变化情况，确保所有异常配置和文件被彻底清除。例如，通过Sysmon日志分析，蓝方可以确认红方是否添加了隐藏账户，或者是否利用了特定端口进行横向移动。蓝方还需要优化防护策略，例如强化IPS的检测规则，提高WAF的拦截精度，并更新防火墙的ACL规则，进一步减少未来的攻击可能性。

通过以上一系列操作，蓝方不仅能够快速响应和处理攻击，还能够通过日志分析和系统加固，提升整个网络的防御能力，防范类似攻击的再次发生。这种从攻击到防御的闭环模拟，不仅反映了智慧校园网络的复杂性和潜在威胁，也为网络安全的实际应用提供了重要的参考价值。

## 实验中的漏洞利用及其防御方法

**2.5.1 城市水务系统网络安全攻防虚拟仿真实验**

在城市水务系统网络安全攻防实验中，攻击方利用了多种常见的漏洞来对系统进行渗透和破坏，以下列出了5个主要攻击点及其防御方法：

（1）Web服务器未授权访问漏洞

攻击点：由于Web服务器配置不当，攻击者可以直接访问敏感数据或执行管理操作。例如，攻击者通过未授权访问的phpMyAdmin界面，修改管理员账号密码并获得完整的后台控制权限。

防御方法：对Web服务器进行严格的访问控制配置，限制管理接口的访问范围（如仅允许内网IP访问）；使用Web应用防火墙（WAF）监控流量，识别并阻断未授权访问行为；定期审计和修复服务器配置及漏洞，确保系统安全性。

（2）文件上传漏洞

攻击点：攻击者利用Web应用的文件上传功能上传恶意文件（如WebShell），获取远程控制权限。例如，攻击者通过修改上传文件类型绕过校验，使用工具（如中国菜刀）控制服务器。

防御方法：在服务器端严格限制上传文件类型和大小，采用白名单机制只允许特定文件类型上传；对文件内容进行病毒扫描和隔离；设置文件上传目录的最低权限，禁止上传文件被直接执行。

（3）FTP服务弱口令漏洞

攻击点：攻击者利用暴力破解工具结合弱口令字典，对FTP服务进行爆破，从而获取上位机的FTP访问权限，进而窃取或篡改关键文件。

防御方法：启用高强度密码策略，要求密码长度、复杂度以及定期更换；限制登录尝试次数，检测并阻止暴力破解行为；将FTP升级为更安全的SFTP或FTPS，确保数据传输的安全性。

（4）Modbus协议未加密通信漏洞

攻击点：Modbus协议缺乏加密和身份验证机制，攻击者可以通过抓包分析和重放攻击控制下位机设备，如操作排水阀或高压泵，导致系统异常运行。

防御方法：采用TLS/SSL加密对Modbus通信进行保护；在协议上增加身份验证和数据完整性校验；通过网络隔离和VPN加密限制Modbus服务的访问范围，避免未经授权的访问。

（5）代码泄露漏洞

攻击点：攻击者通过访问未受保护的代码库，分析源代码并发现系统中的潜在漏洞，从而利用这些漏洞对系统发起进一步的攻击。

防御方法：对代码库设置严格的访问控制权限，限制访问范围；通过代码混淆技术增加代码分析难度；实施代码审计并定期检查代码库的安全性，确保未包含敏感信息。

**2.5.2 智慧校园网络安全实验**

在智慧校园网络安全实验中，攻击方通过对Web应用和内部网络的多重漏洞进行利用，以下是5个主要攻击点及其防御方法：

（1）文件上传漏洞

攻击点：攻击者利用上传功能上传恶意文件（如WebShell），进而控制服务器。例如，上传木马文件后通过浏览器直接执行恶意代码，获取系统访问权限。

防御方法：采用严格的文件上传检查机制，仅允许白名单文件类型；对上传文件进行病毒扫描；对文件目录权限进行严格限制，避免脚本文件被执行。

（2）未授权访问漏洞

攻击点：攻击者通过未授权访问直接进入管理页面，利用管理功能获取系统数据或执行敏感操作。

防御方法：配置Web服务器的访问控制策略，限制管理页面仅能被特定IP访问；通过WAF识别并拦截未授权的访问流量；定期审计系统的权限配置，修复配置缺陷。

（3）SQL注入漏洞

攻击点：攻击者在Web应用的输入字段（如登录框）中注入恶意SQL代码，绕过身份验证或获取数据库中的敏感信息。

防御方法：使用预编译语句或ORM框架处理SQL查询，避免直接拼接用户输入；对用户输入进行严格的验证和过滤，防止恶意数据注入；限制数据库账户权限，确保即使被利用，攻击影响范围也最小化。

（4）SMB协议漏洞

攻击点：攻击者利用Windows SMB协议漏洞（如永恒之蓝）发起远程溢出攻击，直接获取目标系统的最高权限。

防御方法：及时安装针对SMB漏洞的补丁（如MS17-010）；关闭不必要的SMB服务和445端口；使用网络防火墙限制对SMB端口的访问，防止外部攻击者连接。

（5）XSS（跨站脚本）漏洞

攻击点：攻击者在用户浏览器中执行恶意脚本，导致用户会话劫持或敏感信息泄露。

防御方法：对用户输入进行过滤并严格校验，避免恶意脚本被注入；对输出内容进行编码，防止脚本执行；部署内容安全策略（CSP）限制网页资源加载，减少攻击面。

# 分工与合作

## 本人在本次小组实验中完成的工作

在城市水务系统网络安全攻防虚拟仿真实验中，本人主要参与实验一水务官网渗透和实验三排水阀攻击两个部分，同时也学习了实验二和四的内容，占比25%。

在智慧校园网络安全实验中，主要负责红方的工作，我参与了文件上传木马、获取WebShell以及内网攻击的部分。首先，我通过目标系统的文件上传功能上传了一个恶意WebShell文件，成功绕过类型限制并获取了服务器的控制权限。随后，我利用WebShell对系统进行进一步的渗透，扫描内网并发现了开放的SMB服务漏洞（如永恒之蓝）。通过该漏洞，我对内网主机进行了溢出攻击，成功获取了多台主机的控制权限，并进一步扩展攻击范围。最终，我实现了对内网系统的深度控制，模拟了一次完整的网络攻击过程，占比25%。

## 小组实验成员工作完成情况

#### （1）城市水务系统网络安全攻防虚拟仿真实验

#### （2）智慧校园网络安全实验

# 总结与思考

## 针对Metasploit渗透测试实验的改进意见与建议

在 Metasploit 渗透测试实验中，虽然提供了基本的工具使用和漏洞利用流程，但实验环境中的漏洞目标和攻击手段相对固定，缺乏对真实场景的多样性模拟。建议在实验中增加更多类型的漏洞靶机，比如结合 Web 应用、数据库和工业控制系统的攻击目标，同时引入动态防御机制（如防火墙或 IPS）以提高实验的复杂性。此外，可以增加攻击失败后的日志分析和溯源过程，帮助学员更深入地理解攻击行为的痕迹和防御措施。

## 针对实时流量分析实验的改进意见与建议

实时流量分析实验主要通过 Wireshark 抓包和分析流量实现，但在实际实验中，对于抓包结果的筛选和分析要求较高，易对新手造成困难。建议在实验指导中提供更多的流量分析模板和过滤条件示例，帮助学员快速定位特定数据包。同时，可以设计更加复杂的场景，例如模拟流量中包含恶意数据包，要求学员结合抓包工具和分析方法找到潜在威胁。此外，还可以引入其他流量分析工具（如 Zeek 或 Suricata），以拓展实验范围。

## 针对城市水务系统网络安全攻防虚拟仿真实验的改进意见与建议

城市水务系统实验对工业控制系统的安全问题有一定的模拟，但实验场景主要集中于 Modbus 协议的漏洞利用，范围较为单一。建议增加更多工控协议的模拟（如 DNP3 或 OPC UA），并引入物联网设备的攻击场景，强化学员对工控网络安全的理解。同时，实验环境中缺乏防御策略的实践内容，可以加入攻防对抗的环节，例如要求学员配置防火墙规则或启用 Modbus 加密机制进行防护，从而提升实验的综合性和实践价值。

## 针对智慧校园网络安全实验的改进意见与建议

智慧校园实验涵盖了 Web 漏洞挖掘、内网渗透和社会工程等内容，但实验流程较为标准化，实际操作的灵活性不足。建议增加模拟真实攻击场景的模块，例如模拟复杂的社会工程攻击（如钓鱼邮件）或弱口令暴力破解的多层次防御对抗。同时，可以扩展实验的防守部分，要求学员不仅完成攻击任务，还需要制定一套完善的安全防护策略，并通过实验验证其有效性。此外，实验可加入模拟攻击日志分析的环节，强化对攻击过程的追踪与溯源能力。

## 实验过程中遇到的难点及解决方案

在实验过程中，难点主要集中在以下几个方面：一是 Wireshark 流量分析中的数据包过滤规则设置复杂，初学者容易漏掉关键数据包；二是 Metasploit 渗透测试中漏洞利用不稳定，可能受实验环境配置影响；三是工控系统实验中 Modbus 通信抓包和重放的操作需要较高的准确性，容易导致实验失败。为解决这些问题，可以在实验文档中提供详细的排错步骤和案例；同时，在实验前期增加环境调试环节，确保实验环境稳定运行。如果问题仍未解决，可以计划开发交互式实验平台，集成常见错误提示与自动化环境检测功能，以降低实验操作难度。

## 对关键信息基础设施安全保护的理解

通过这些实验，我深刻认识到关键信息基础设施的安全性直接关系到国家的经济运行、社会稳定以及人民生活的正常秩序。关键信息基础设施（如工业控制系统、电力能源系统、金融系统、交通运输系统和医疗系统）是社会运行的核心支柱，这些系统一旦遭受网络攻击、物理破坏或内部威胁，不仅可能导致严重的经济损失，还可能对国家安全造成不可逆转的影响。例如，工业控制系统被攻击可能导致大规模停电，金融系统漏洞可能引发资金被盗或金融市场的混乱，医疗系统的网络攻击则可能直接威胁到人类生命安全。

因此，在保护关键信息基础设施的过程中，需要从多个维度入手，实施全面的防护策略。技术层面，应重点加强系统的漏洞管理，定期扫描并修复已知漏洞，同时完善访问控制策略，确保只有授权用户能够访问敏感资源。此外，还应部署入侵检测与防御系统，通过实时监测和快速响应阻止潜在威胁。数据层面，需要对核心数据实施严格的加密与备份策略，防止数据泄露、篡改或丢失。同时，对通信协议进行加密和认证，保障数据传输的安全性。物理层面，应加强对重要设施的物理保护，如视频监控、门禁系统和警报机制，防范外部入侵者破坏。

在技术防护之外，管理层面的安全措施同样不可忽视。需要建立健全的安全管理制度，明确岗位职责和应急响应流程，以提高整个组织的安全防护能力。此外，还需注重人员培训，提升操作人员和管理人员的网络安全意识与技能，减少由于人为失误导致的风险。例如，通过开展网络安全演习或攻防演练，提高团队对突发安全事件的应急处理能力。

与此同时，关键信息基础设施的安全保护不能局限于单一机构或企业，而是需要各行业、各部门乃至国际间的协作与信息共享。例如，建立跨行业的威胁情报共享机制，可以帮助各方及时掌握最新的攻击手段和防御方法，从而形成更强大的整体安全防线。总之，关键信息基础设施的保护是一项系统工程，需要技术、管理和意识层面的协同配合，才能确保国家和社会的长久安全与稳定。

## 其他感想

通过这些实验，我不仅学到了许多专业的网络安全知识和实践技能，也加深了对关键信息基础设施保护重要性的理解。这是我本科四年的最后一个实验，完成之后让我感慨万千。在实验过程中，从一开始的生疏和困难，到逐渐掌握技术、理解原理并成功完成任务的过程，让我真正体会到了学习的成就感和成长的喜悦。

回顾本科四年，每一次实验和项目不仅丰富了我的知识储备，也锻炼了我的动手能力和团队协作精神。这些实验让我逐步从一个对专业知识一无所知的新生，成长为能够独立思考、解决问题的网络安全实践者。我尤其感激在这段时间里，老师的细心指导和同学们的互相帮助，是这些支持让我在每一次遇到困难时都能坚持下来并不断进步。

作为最后一个实验，它标志着我本科阶段学习的结束，同时也预示着新的起点。站在这个节点，我更加坚定了未来继续深耕网络安全领域的决心。我希望能将本科期间积累的知识与技能运用到未来的工作和学习中，继续提升自己，为社会的网络安全和技术发展贡献一份力量。这个实验带给我的不仅是技术层面的成长，更是一次对自我突破和专业热爱的总结和升华。感谢这段充满收获和意义的旅程，我会铭记这些宝贵的经验与感悟！

|  |  |
| --- | --- |
| 原创性声明 |  |
| 本人郑重声明本报告内容，是由作者本人独立完成的。有关观点、方法、数据和文献等的引用已在文中指出。除文中已注明引用的内容外，本报告不包含任何其他个人或集体已经公开发表的作品成果，不存在剽窃、抄袭行为。  已阅读并同意以下内容。  判定为不合格的一些情形：  （1）请人代做或冒名顶替者；  （2）替人做且不听劝告者；  （3）实验报告内容抄袭或雷同者；  （4）实验报告内容与实际实验内容不一致者；  （5）过度使用AI技术者。  **作者签名：** |  |

签名：

日期：2024.12.27