摘要

本文主要对信息收集平台的功能进行研究，分析现阶段国内外已有的平台、工具、手段所采用的相关技术，总结自身大学所学知识，设计实现出一款简单实用的信息收集平台，主要面向渗透测试人员和网络安全的学习人员。信息收集平台在开发过程中主要采用了 Django、Python 以及 JS 技术等，实现了前后端的数据分离，主要的模块包括 Web 基础信息查询、端口扫描、旁站查询和信息泄露查询模块，在实际运用中能达到快速收集目标站点信息的目的，在准确收集目标信息的同时，也提高了信息收集的效率。

关键词：Django;Python;JS；信息收集

Abstract

This article mainly studies the function of the information collection platform, analyzes the relevant technologies adopted by the existing platforms, tools, and means at home and abroad, summarizes the knowledge learned by the university, and designs and implements a simple and practical information collection platform. For penetration testers and network security learners. In the development process of the information collection platform, Django, Python and JS technologies are mainly used to achieve data separation between the front and back ends. The main modules include Web basic information query, port scanning, side station query and information leakage query module. It can achieve the purpose of quickly collecting the target site information, while accurately collecting the target information, it also improves the efficiency of information collection.

Key words: Django;Python;JS;Information Collection

目　录

第一章 绪论

1.1 研究背景

当今社会得益于人们生活水平在逐步提升，互联网也随之快速的发展。我们的生活与互联网的联系也越来越密切，而 Web 网站作为当今互联网的主流业务随之暴漏出来的安全问题也接踵而至。

在深信服 2019 年上半年的网络安全态势报告[1]中的数据可以看到，目前 Web 网站被攻击的数量整体还是呈上升的趋势，全球网络态势依然严峻。而在国内，Web 网站被攻击的重灾区还是出现在了安全设施较为薄弱的科研教育和企业政府。

此外，随着 Web 网站与服务相关的漏洞细节被披露曝光，利用这些漏洞进行复现攻击的门槛也越来越低，倘若未能及时修复，就会造就了不必要的损失。根据 CNNVD2019 年上半年的监测数据显示[2]，国内出现的 0day 漏洞数量持续走高，利用这种漏洞很容易对一大批同类型的 Web 网站发起攻击，造成极大的损失。可见，Web 网站的安全问题依旧是形势严峻，不容忽视。

1.2 研究现状

网络安全在国家的战略地位中显得尤为的重要，加上国内的网络环境面临的威胁日益严重。国家开始对网络安全所面临的问题越来越重视，进而采取了一系列的净网护网等措施，网络安全上升到了一个全新的高度。

而 Web 渗透测试是检测 Web 网络安全的重要方法，渗透测试的标准有很多，包括 OWASP Testing Guide、OSSTMM、NIST 还有 PTES 等标准，这些都是目前安全行业公认的可以为渗透测试提供指导的标准，国内目前最常用到的渗透测试标准还属 PTES 标准作为指导进行渗透测试。

信息收集是渗透测试阶段中很关键的一步，在这一阶段中，需要通过信息收集去收集到更多与目标的相关信息，最后把收集到的信息结合起来制定相关的方案，才能有效的提高渗透测试的成功率，还可以降低渗透测试对目标系统业务带来的一些风险[3]。

在信息收集阶段，根据是否直接与目标系统产生接触，可以分为主动和被动信息收集。利用主动信息收集与目标系统产生直接接触，可以收集到目标系统

的 IP 及域名信息、端口信息、操作系统信息、CMS（content managenment system, 内容管理系统）信息[4]等等，利用被动信息收集不与目标直接接触，通过 OSINT（Open source intelligenc, 公开资源情报计划）和搜索引擎相互结合的方式获取目标系统的商业信息、社交信息、组织结构信息等等。只要是有利于渗透测试的信息就一点也不能落下[5]。

1.3 面临问题

目前业界在渗透测试、大范围网络空间终端设备的发现和识别还有其他的 Web 服务器等资产搜集发现方面，最为有代表性的工具还属 Zoomeye、Shodan、Fofa 三大搜索引擎，主要的功能就是对全网服务器、物联网设备进行基础信息的收集，只要是连接到网络的设备都会被轻易搜索到。

但是在业内渗透测试的前期信息收集工作主要还是通过使用一些工具、公开网站还有搜索引擎一点一点的去进行收集。虽然有三大搜索引擎，但是收集到的信息还是远远不够。在收集过程中使用到的既有主动式也有被动式的信息收集，两者相互结合起来获取更多的信息，更有利于后续的渗透工作进行。但是正是信息收集涉及的范围太大，就拿主动式信息收集来说，其中使用到的方法、工具、网站就千差万别，虽然最终实现的功能都是大体相同的，但是没有一家能结合百家之长，还是有很多的工具、网站如雨后春笋般的出现。

除此以外，测试人员热衷使用自己开发的程序进行信息收集。这就造成了同类功能的收集软件很多，但是使用的体验不尽相同，让人不知如何选择。所以，这样信息收集的方式还是不够的高效便捷，而且人工的信息收集可能会出现结果偏差的情况。综上所述，针对目前渗透测试缺乏一个综合的平台可以做到收集更全面、更直接的站点信息，并可以在信息收集结束后进行分类整合输出。

1.4 研究意义

本课题通过研究在渗透测试过程中最重要的一个环节——信息收集这一过程的具体工作方式，设计出一个可行的信息收集平台的工作框架，通过一个框架就能总结出在渗透测试过程中需要收集到的信息，进而减少重复的工作量和提高整体的工作效率。

在渗透测试中，前期的信息收集工作往往需要占用一半的时间以上，这也可以看出信息收集工作的困难程度之大。而收集到的信息越多，就越有利于之后的渗透测试

。但是往往在信息收集的过程中，涉及到的工具和途径是非常多的，这时我们就只能一类一类的去查询我们所需要的信息，在此过程中还得过滤筛选掉没用的信息，整理分类出有用的信息。这就导致了信息收集的工作复杂，投入的人力物力消耗巨大，效率十分低下的问题，而且最后的信息整合没有一个参考的模板。因此，信息收集的复杂程度之高已成为当前渗透测试的首要问题，对于渗透测试前期的信息进行收集整合和过滤，能在一定程度上提高信息收集的效率和降低人力资源的消耗，保证渗透测试的成功率。

1.5 章节安排

我把论文的整体划分为五个章节，每个章节的主要内容如下：

第 1 章：首先介绍了论文的研究背景和目前现状，然后又通过目前的网络安全背景情况，从而引入目前检测 Web 网站安全的方式——渗透测试，进而提出信息收集的基本概念和信息收集的范围，并分析了目前信息收集方式的问题，最后给出了本文的结构。

第 2 章：主要介绍了信息收集平台的主体框架，以及架构中每个模块实现的原理，还有各个模块实现的思路方法。

第 3 章：主要就是研究了信息收集平台各个模块所要用到的技术原理，包括介绍此次实践中所用到的一些理论方法还有基本概念，然后还对部分实现思路的代码进行了介绍。

第 4 章：对各个模块进行了功能上的验证和测试，包括对信息收集平台前后端实现的具体效果进行了测试和检验，还有前端各模块的显示细节，后端各模块的功能展示还有前后端的连接测试。

第 5 章：总结了此次毕业设计中所做的研究工作，包括提出了对此次设计的不足之处，还有对今后的人生道路进行了展望。

第二章 系统框架和模块设计

2.1 系统框架

2.1.1 基本概念

2.1.1.1 Web 渗透测试

Web 渗透测试（Penetration Test）是一种评估 Web 系统安全的方法。在此过程中，网络安全工作人员会从“逆向”的角度出发，用攻击的手段还有结合各种技术方法去寻找系统所存在的漏洞，最后确定被测试的 Web 系统的安全状况，同时提出 Web 系统中存在的缺陷，并给出相对应的修复方案[6]。

2.1.1.2 Web 渗透测试基本流程

PTES（The Pen-etration Testing Execution Standard, 渗透测试执行标准）是当前业内比较常用的渗透测试指导标准[7]，其中主要包括前期交互制订方案、收集目标相关信息、结合信息制订渗透方案、验证测试渗透方案、开始漏洞攻击、后渗透扩大范围、输出测试报告等环节。下面主要分析最基本的四个流程，分别是信息收集、漏洞分析、渗透实施和输出报告这四个基本阶段。

(1)信息收集阶段

在渗透测试阶段，最重要的工作就是前期的信息收集阶段。在渗透攻击前一般需要做大量的准备工作——信息收集，这个不管是个人还是团队在渗透测试前都是必须要做的工作。一般信息收集占总工作量的 50%，甚至 70%。在这个阶段，渗透测试者利用各种信息来源、搜索引擎还有工具，去获取到更多与目标相关的信息。在这个阶段使用到的方法一般包括 OSINT（Open source intelligenc, 公开资源情报计划）、各种搜索引擎、平台查询等等[8]。收集到的信息越多，模拟攻击的成功率越高，渗透测试可以发现的问题也更多。由此可见，前期信息收集工作的重要性。

（2）漏洞分析阶段

信息收集完毕以后，就需要对相关的信息进行整理分析，通过结合各类信息去分析判断出目标系统可以突破利用的点，最终确定出一套可行的渗透测试方案和实施渗透测试的进攻点。之后就是搭建起目标的网络环境，尽可能的模拟出一个最真实的环境进行测试，确定出最终的漏洞突破口。在这个阶段，通过对漏洞分析，模拟攻击，渗透测试者可以确定出可行的攻击方案，在确保攻击成功的同时也能保证规避风险以降低渗透测试对系统产生的影响。

（3）渗透实施阶段

在确定了可利用的漏洞点之后，就准备开始真正的渗透测试，获取目标系统的访问控制权。进一步获取目标系统里面更多的信息，如果是大型的渗透测试目标，还需要考虑横向移动、旁站爆破等等手段，拿下更多的访问控制权。还有就是需要对目标系统中的防御软件进行免杀、绕过甚至是干掉。最后拿到相关的权限和信息之后，需要清理入侵的痕迹，以避免造成目标系统安全响应团队的响应。

（4）输出报告阶段

完成渗透攻击，拿到目标的访问控制权限之后，需要对整体渗透测试流程的细节进行一个报告撰写。报告需要清楚的做好每个阶段的情况记录，其中包括渗透测试各个阶段中获取到的关键的目标信息、探测挖掘出的漏洞、成功实现渗透攻击的过程细节，以及通过暴漏出来的漏洞能对目标系统相关的业务造成的影响，最后是修复漏洞的方案建议，包括修复与升级的技术方案，以帮助客户修复安全防御体系中所存在的问题。

2.1.1.3 信息收集技术研究

根据前面对 Web 渗透测试的基本流程介绍可以知道，一般在 Web 渗透测试开始之前需要制订好初步的计划方案，然后开始信息收集工作，直至后面的几个流程步骤，整个渗透测试的工作都是根据这个标准流程化的去严格执行的。

信息收集阶段在整个渗透测试的工作中起着至关重要的作用。主要就是对目标系统进行一个初步的情况了解，尽可能的去收集更多与目标相关的信息，为下一步的渗透攻击做好相应的准备[9]。

其实信息收集简单来说就是利用各种各样的方法去获取我们所需要的信息。这是最关键的一个阶段，一次完美的信息收集带来的效果是巨大的，甚至是推动整个渗透测试的进行。如果尽可能的去获取到了各种与目标相关的信息，那么取得突破的点可能又多了一个，突破的概率可能又提高了一点。总之，收集的信息越多，就可以给后期的渗透测试提供更多的思路和指导[10]。

（1）信息收集的范围

信息收集的范围一般需要根据目标系统的复杂程度，再结合实际的情况出发进行划分，从而围绕目标系统进行信息收集。其中，需要收集与目标系统相关的信息主要包括网络拓扑信息、服务器信息、组织架构信息等等，除此之外，一些与目标相关的业务信息、邮箱信息，甚至人员信息也不能落下。在这个阶段需要收集的信息数目庞大、种类繁多，为了使信息收集更加具体细致，信息收集工作一般又可以分为以下几个部分，主要包括目标获取、主机探测、端口扫描、指纹识别、协议分析等等。

（2）目标获取

目标获取主要会针对域名解析的记录和子域名的枚举下手，获取目标的一个基本轮廓。域名解析可以使用 nslookup 等等工具来查询目标网站对应的域名信息，包括 A、NS、CNAME、MX 和 SOA 记录，通过这些手段就可以查询到网站的 DNS 服务器地址、邮件服务器地址等信息，甚至能暴露出目标站点的整体架构。

（3）主机探测

主机探测的目的在于判断目标主机的在线情况，只有存活的主机才是我们渗透攻击的目标，但是一般的网站都会自带防火墙或者是 WAF（网站应用级入侵防御系统）等防御系统。

此时我们需要使用一些工具去检测目标防御和在线的状态，也就是 WAF 探测，常用的 WAF 探测工具有 wafw00f、Nmap、Sqlmap 等工具。WAF 探测工具的原理是通过向目标系统故意发起一个错误的请求，此时分析返回的请求结果，然后再与正常请求的返回结果进行对比，就可以发现目标系统是否存在 WAF。前面提到的 Nmap 工具是一个强大的工具，利用它可以直接发起 WAF 探测，得到 WAF 存在的状态之后，可以直接添加指定参数，绕过 WAF 进行探测，从而发现主机存活的状态。

主机探测主要分为内网和外网两部分，内网主机探测通常使用 Arp 协议对内网主机 IP 地址、Mac 地址等信息进行探测。除此之外内网还可以通过 NetBIOS 服务进行探测，如果内网主机开启了此服务，可以利用 Nbtscan 获取主机的 IP、NetBIOS 名字和 Mac 地址等信息。外网主机的探测则可以通过直接 Ping 目标主机地址，然后观察返回来的结果就可以知道主机的在线状态。由于 Ping 命令是直接通过发送 ICMP 包给目标主机的方式来获取在线状态，所以现在很多主机为了安全性都会把 ICMP 进行一个过滤。

（4）端口扫描

在确定了目标主机的存活状态之后，此时就需要进一步对主机进行更详细的探测扫描，在这一步需要获取目标主机端口的开放情况、还有对应的服务情况等等信息。TCP Connect（）扫描是使用频率比较多的一种扫描技术，然后还会用到 TCP SYN 扫描等等扫描技术。此外还有一些更为隐蔽、特殊的端口扫描技术，这个得看具体的情况去使用。

TCP Connect（）扫描：这是最常用到的一种扫描技术，因为使用的方法简单粗暴。它会直接与目标主机建立起 TCP 连接，如果与目标主机可以成功建立三次握手，说明端口是开放的。这种扫描方式因为会产生完整的三次握手，所以被发现的风险是比较大的，但与此同时它得到的结果却是最准确的。而 SYN

扫描方式则是在确定了端口的开放情况之后，赶在建立三次握手之前，发送取消建立连接的数据包，提前结束连接状态来获取端口的开放情况。这种扫描方式因为没有建立起完全的三次握手连接，所以扫描的速度会更加快。除了这些探测技术以外，直接使用现成的工具也是一个不错的办法，上面提到的 Nmap 就是最常用的神器之一，集主机探测、端口扫描、WAF 探测、系统识别各种功能于一身。

（5）指纹识别

指纹识别主要是识别目标主机的操作系统信息、中间件信息还有 Web 容器的信息。根据识别的方式不同又可以分为以下几种：主动识别、被动识别还有 Web 指纹识别。

主动识别主要是通过抓取 Banner（服务标识）信息来实现对目标系统的识别。Banner 信息中一般包含着服务器、语言、容器还有版本等信息。通过直接与目标系统发起请求进行连接，在连接成功之后，就可以收到目标系统主机返回来的 Banner 信息。比如，对腾讯网首页直接发起请求进行连接，在发起请求之后连接成功，可以看到返回来的数据包中包含着“server：nginx”的信息。之后能这些信息经过解析之后，就可以获取目标系统所使用的容器、系统及编程语言等等相关的信息了。

被动识别则是不主动与目标系统产生直接的接触，通过被动监听目标系统中的数据流然后进行分析，所以隐蔽性极高，几乎很难被发现。最主要的是通过被动接收数据，可以获取到更多的数据包，进行仔细分析。

Web 指纹识别比较有代表性的工具包括 Wappalyzer、Whatweb、Fofa 等，通过工具已有的数据特征库可以快速的探测一些常见的 Web 服务版本、应用版本还有 CMS 版本等信息，这些工具自带庞大的数据特征库，扫描识别的速度快，得到的信息也比较准确。在收集到准确信息的同时，也提高了收集信息的效率。

2.1.2 系统框架

本文提出的信息收集平台设计模型如图所示，根据实际情况，我把主要结构分成了四个模块，分别是指纹信息模块、端口扫描模块、旁站发现模块还有信息泄露查询模块。（如图 2-1 所示）

此次信息收集平台的设计框架我选择了 Django 来实现，Django 与其他的 Web 框架相对比有着独特之处：比如框架本身就集成了 ORM，可以不用再去关注数据的操作，除此之外还有模型绑定等等诸多的功能[11]。Django 是一个开源的由 Python 开发的 Web 应用框架，它采用了 MTV 的软件设计模式，即 M（模型）、T（模版）和 V（视图）[12]。

该信息收集平台的工作流程设计的比较简洁方便，主要面向四大模块的信息收集内容，其中包括 Web 服务器的基本信息、指纹信息、端口信息、旁站信息还有信

息泄漏信息等。只要输入一个 URL

就可以获取大部分所需的信息，其工作流程按照顺序可分为四大步：第一步，使用 Web 服务器指纹信息模块，输入目标系统的网站地址进行解析，在获取目标 IP 真实地址的同时，模块中的爬虫函数会对 CDN、WAF、容器等信息进行探测，最后做一个整合进行输出。第二步，使用 Web 服务器端口扫描模块，利用第一步获取到的 IP 地址，读取定义好的端口范围，对目标系统的端口进行请求探测，根据返回的结果获取端口的开放情况。第三步，使用 Web 服务器旁站发现模块，请求接口，返回存在旁站的网站标题和 URL。最后一步，使用 Web 服务器信息泄漏模块，输入目标系统的 IP 地址，读取自定义的 URL 泄漏地址进行请求，根据返回结果对目标系统的泄漏情况作出判断。整体下来能收集到大部分关键的目标信息。

图 2-1 系统框架

2.2 模块设计

按照前面提出的信息收集平台系统框架，本小节将对信息收集平台的指纹信息模块、端口扫描模块、旁站发现模块还有信息泄漏模块的设计思路进行逐一介绍。

2.2.1 Web 服务器指纹信息

指纹信息模块我主要分为三个部分去实现，分别是请求目标、请求接口和指纹识别（如图 2-2-1 所示）。直接请求目标是抓取 Banner 信息，里面包含着域名、语言、操作系统还有 Server 信息等等。而 CDN、WAF 还有 IP 信息等等则是通过网上平台已有的接口或者是自己写的爬虫进行查询，最后则是结合 GitHub 开源的指纹识别项目增加查询到的信息的准确性。

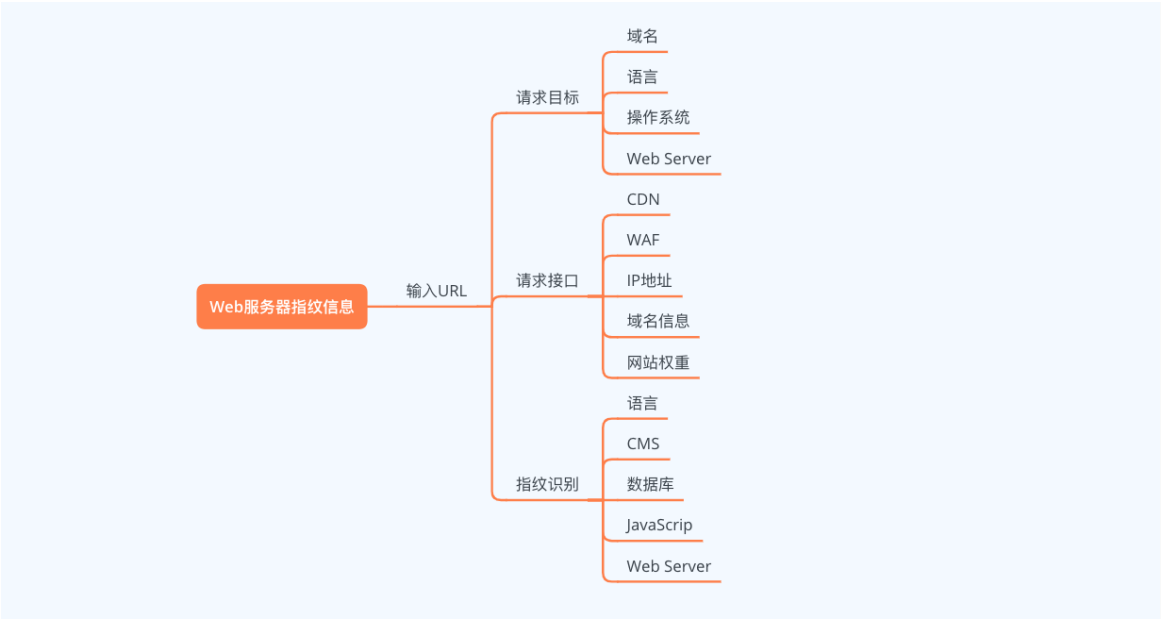

图 2-2-1 Web 服务器指纹信息框架

2.2.2 Web 服务器端口扫描

TCP Connect( )扫描是目前使用频率比较多的一种扫描技术，除了 TCP Connect( )扫描之外，还有 SYN 扫描、ACK、FIN 等等扫描技术，每一种扫描技术都存在着不同的优缺点。根据各个扫描方式的不同主要可以分成开放、半开放还有隐蔽扫描等扫描方式（如图 2.2.2 所示），这里我选择了 TCP Connect( )扫描技术进行端口探测。

图 2-2-2 端口扫描技术分类

2.2.3 Web 服务器旁站发现

假设目标服务器上只有一个网站，当服务器还存在其他的网站的时候，那么这个网站相对于第一个网站而言就可以被称为旁站[13]。这些服务器上共存着多个网站的目录，不同的网站使用着不同的域名、甚至不同的端口，但是它们的目录是相互独立不受影响的。

有时候在渗透测试的过程中，总会遇到无法突破的站点。此时可以利用旁站查询，查询服务器是否存在其他的旁站。倘若旁站的安全措施做的不好，此时就可以利用旁站拿下整台服务器的权限，直通目标站点的目录。所以很有必要对目标系统的旁站进行收集查询，检测服务器上的旁站是否存在安全漏洞。

在这个模块中，我直接使用了在线的平台查询进行爬虫查询。网上有很多在线的查询工具，比如：aizhan、站长之家、Webscan 等等在线查询平台。（如图 2.2.3 所示）

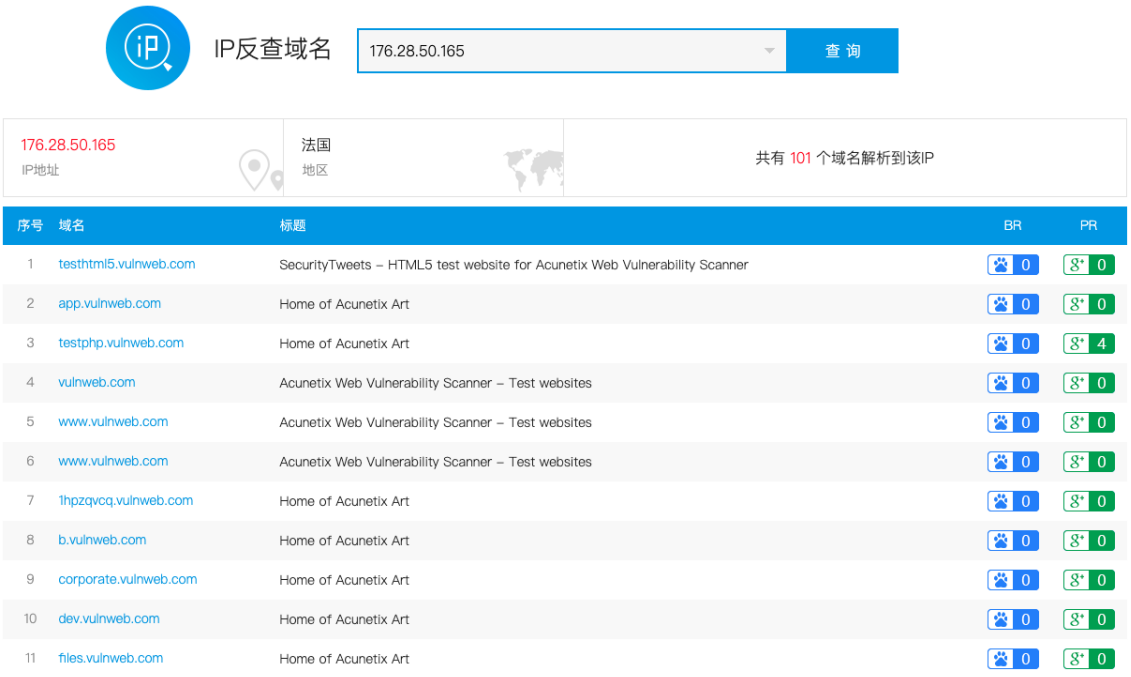

图 2-2-3 aizhan 在线旁站查询

2.2.4 Web 服务器信息泄露

什么是 Web 服务器信息泄漏呢？Web 敏感信息泄漏包括软件信息泄漏、文件包含泄漏还有配置错误信息泄漏等等（如图 2.2.4 所示）。Web 敏感信息泄漏，简单来说就是网站管理员把服务器不该公开的信息，不小心暴露出来了，给予了每个人都可以访问到的机会，从而造成了网站服务器相关的配置信息泄漏。

只要泄漏的数据对于攻击者有参考作用的信息，都属于敏感信息泄漏。此模块主要是通过收集了常见的敏感信息泄漏的路径，如：WEB-INF 泄漏、Wp-config.php 泄漏还有备份文件泄漏等等的路径，然后逐一尝试访问，根据返回的结果来判断服务器是否存在信息泄漏的地方。

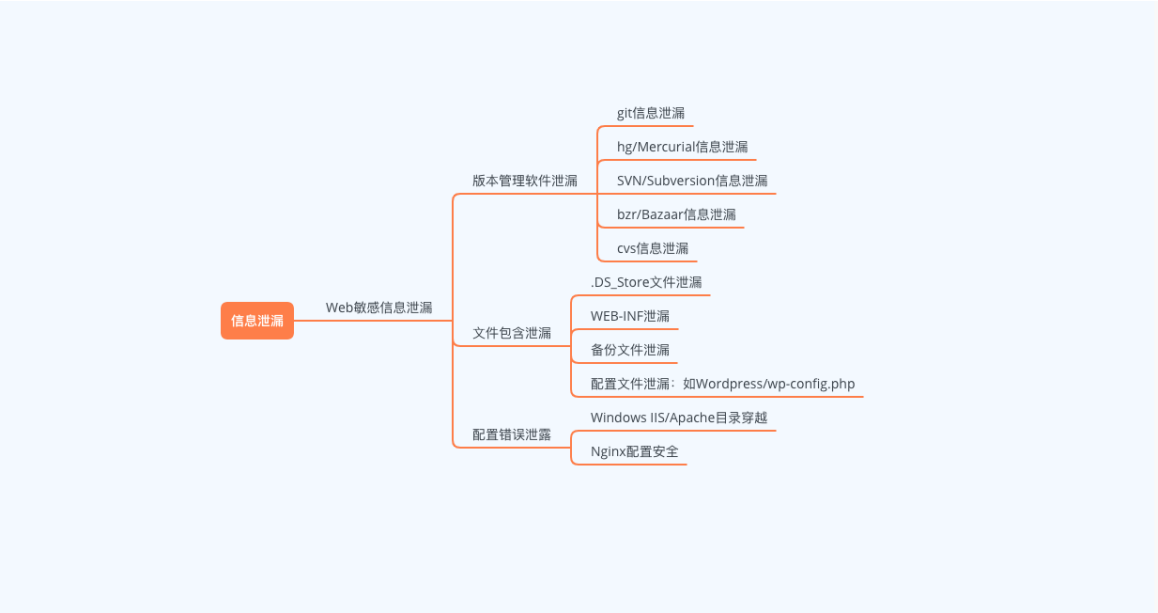

图 2-2-4 Web 敏感信息泄漏分类

2.3 本章小结

本章对信息收集平台的整体框架进行了介绍，一开始，先是对相关的概念进行了基本的介绍，然后提出了该信息收集平台的整体框架结构，最后从各个模块的设计思路出发，进行了讲解，为下文的具体内容做好铺垫打好基础。

第三章 技术要点研究

3.1 Web 服务器指纹识别技术研究

3.1.1 技术概述

在这个模块中，我主要从三个方面进行研究，首先是请求目标，主要研究通过直接与目标系统发起请求建立连接，从而抓取目标系统的 Banner 信息，进而对目标系统的相关信息作出判断；然后是请求接口，主要是专门对 IP 信息、CDN、WAF 等信息作出查询判断，通过已有的在线平台还有爬虫接口去获取目标系统的相关信息；最后是指纹识别框架，这部分主要是利用了开源的成熟框架，进行正则匹配，从而返回目标系统所使用的框架、组件和容器等等信息。通过这三个部分的信息获取结合，得到一个较为准确的结果。

3.1.2 指纹识别技术研究

（1）请求目标

通过 requests 函数直接对目标发起请求，然后抓取目标返回来的结果进行信息查找，也就是 Banner 信息抓取。Banner 信息中一般包含着服务器、语言、容器还有版本等基本信息[14]。通过直接与目标系统发起请求进行连接，在连接成功之后，就可以收到目标系统主机返回来的 Banner 信息。在返回的结果中就可以看到包含着目标的 IP、域名、Server、还有语言等相关信息。（如图 3.1.2.1 所示）

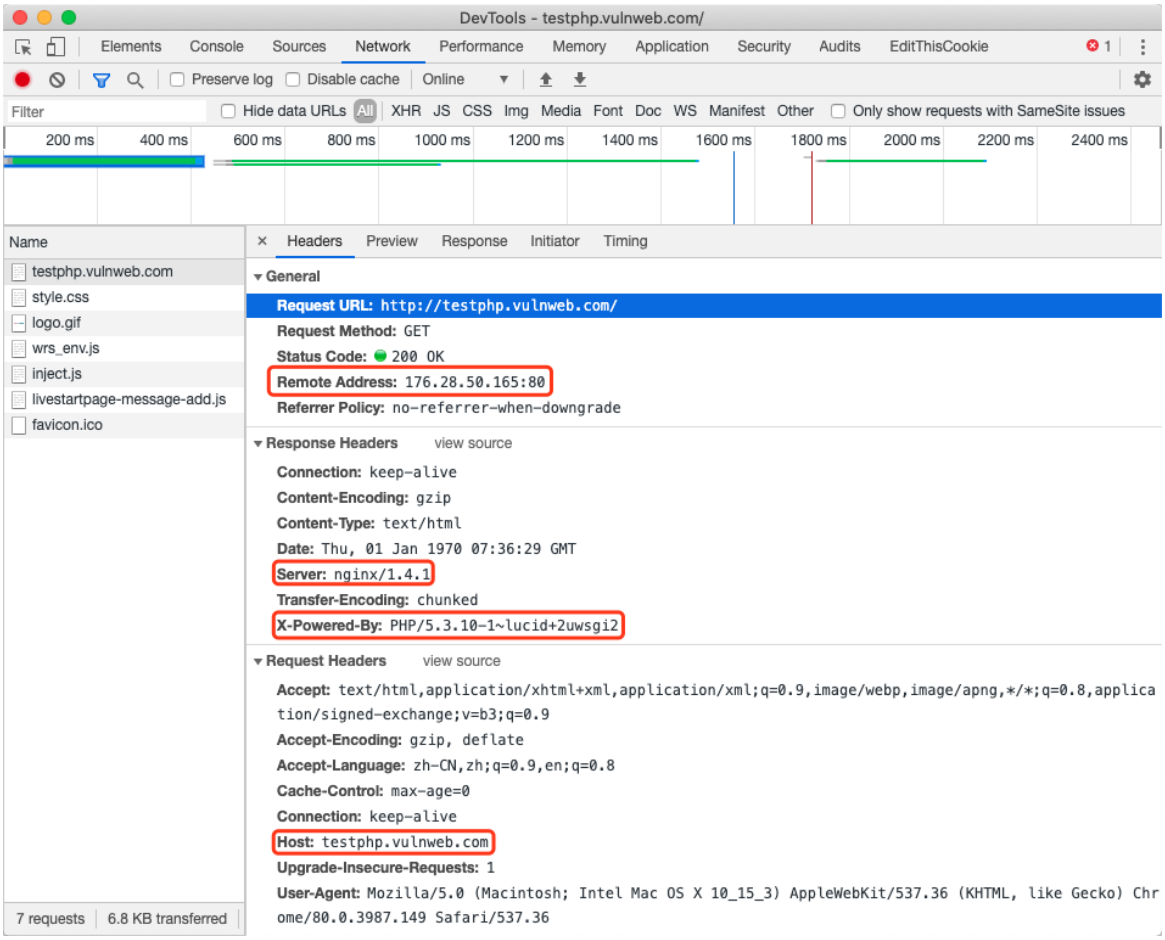

图 3-1-2-1 向目标发起请求

（1）请求接口

除了直接请求目标之外，还应该使用一些接口，专门负责获取目标系统的 IP 地址信息、whois 信息、域名信息、CDN 信息还有 WAF 信息。（如图 3.1.2.2 所示）

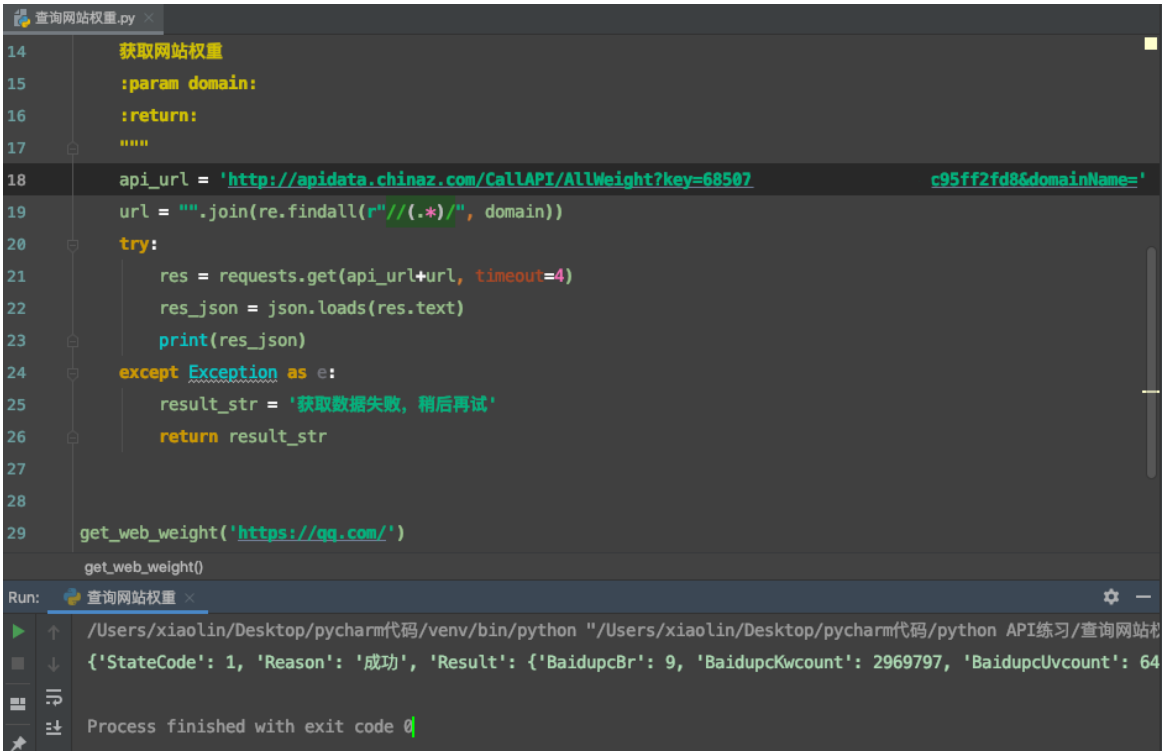

图 3-1-2-2 请求接口查询网站权重

（1）指纹识别

识别网页中的标题信息还有网站中的静态文件 MD5 是目前比较常见的 Web 指纹识别方式。在这里我使用了 Wappalyzer 的识别框架（如 3.1.2.3 所示）

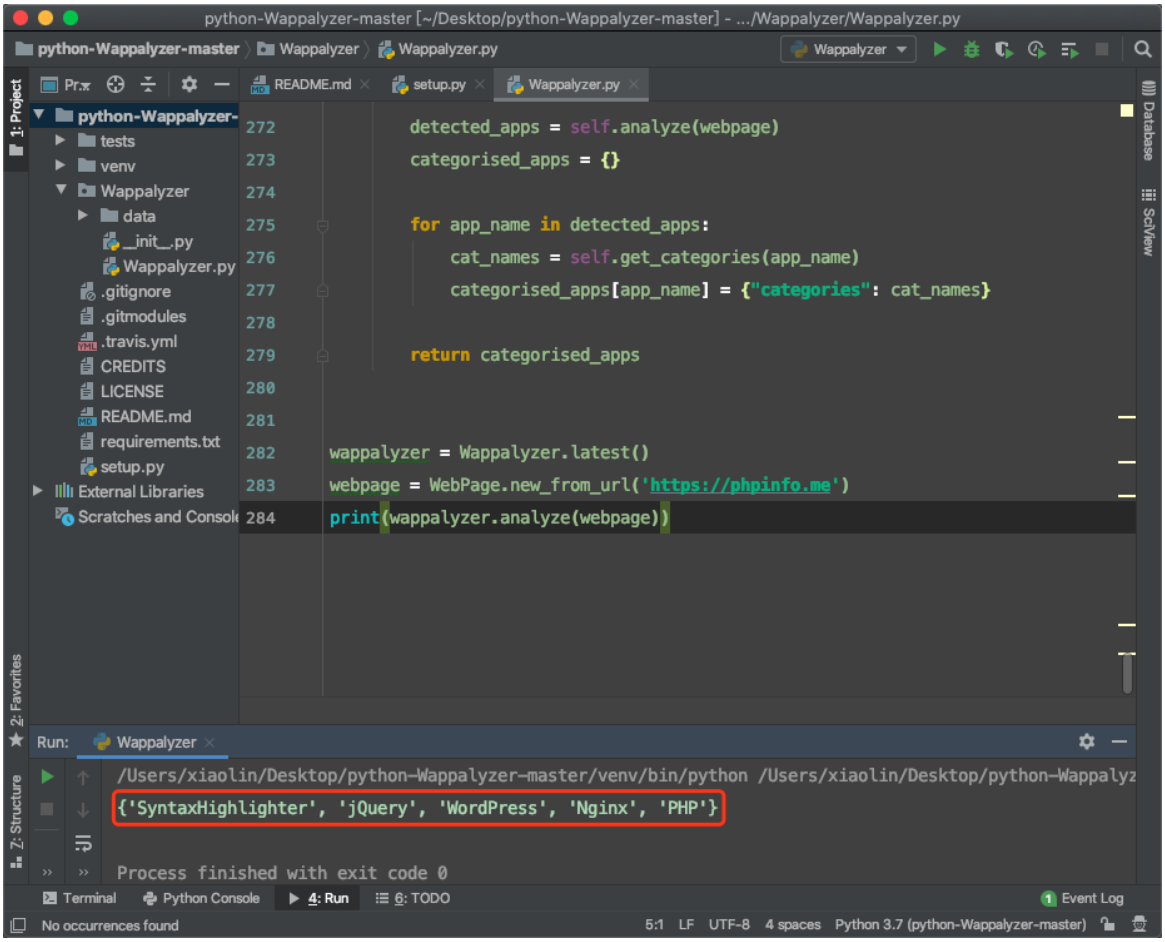

图 3-1-2-3 Wapplyzer 框架指纹识别

3.2 Web 服务器端口扫描技术研究

3.2.1 技术概述

Socket（套接字）接口是操作系统中的一种接口技术[15]，它的出现主要是为了解决目前计算机之间的通信而物理接口不够用的问题。如今有了 Socket 接口，计算机之间就可以利用 Socket 接口进行连接。所以，接下来我们要说到的端口也就是 Socket 接口。

TCP/IP 协议在平常的使用中十分广泛，比如 Telnet、SMTP、HTTP 还有 SSL 等等都是基于 TCP 协议的，端口扫描技术也是基于 TCP 协议进行工作的[16]。每一种扫描技术都存在着各自不同的优缺点，而目前比较常用到的端口扫描技术主要有 TCP Connect( )扫描和 SYN 扫描等。在实际的渗透测试过程中，会根据实际的情况再决定使用不同的扫描方式。这里我主要选择了 TCP Connect( )扫描技术对端口探测进行研究，主要是看中了它结果准确的优点。

3.2.2 端口扫描技术研究

TCP Connect( )扫描是最常见的一种扫描方式，在扫描的过程中主要使用到了TCP/IP 协议的三次握手连接机制。具体的表现为，我们直接利用 Connect( )发起连接请求，如果目标主机有反应并成功建立起三次握手，则表示该端口是开放的；如果失败，则表示该端口关闭。

该过程可以表示为，客户端 A 调用 Connect( )向服务器 B 发送 SYN 报文，之后等待服务器 B 端口返回 SYN 和 ACK 报文。在收到返回来的报文后，客户端 A 会继续向服务器 B 发送 ACK 报文确定建立连接。简单来说就是 A 向 B 发送“你在吗”的消息，B 收到之后会做出回应“我在的”，A 收到之后就会正式开始建立连接，这就是著名的“三次握手”（如图 3.2.2.1 所示）。如果目标端口是关闭的状态，则目标主机会返回 RST 的响应。

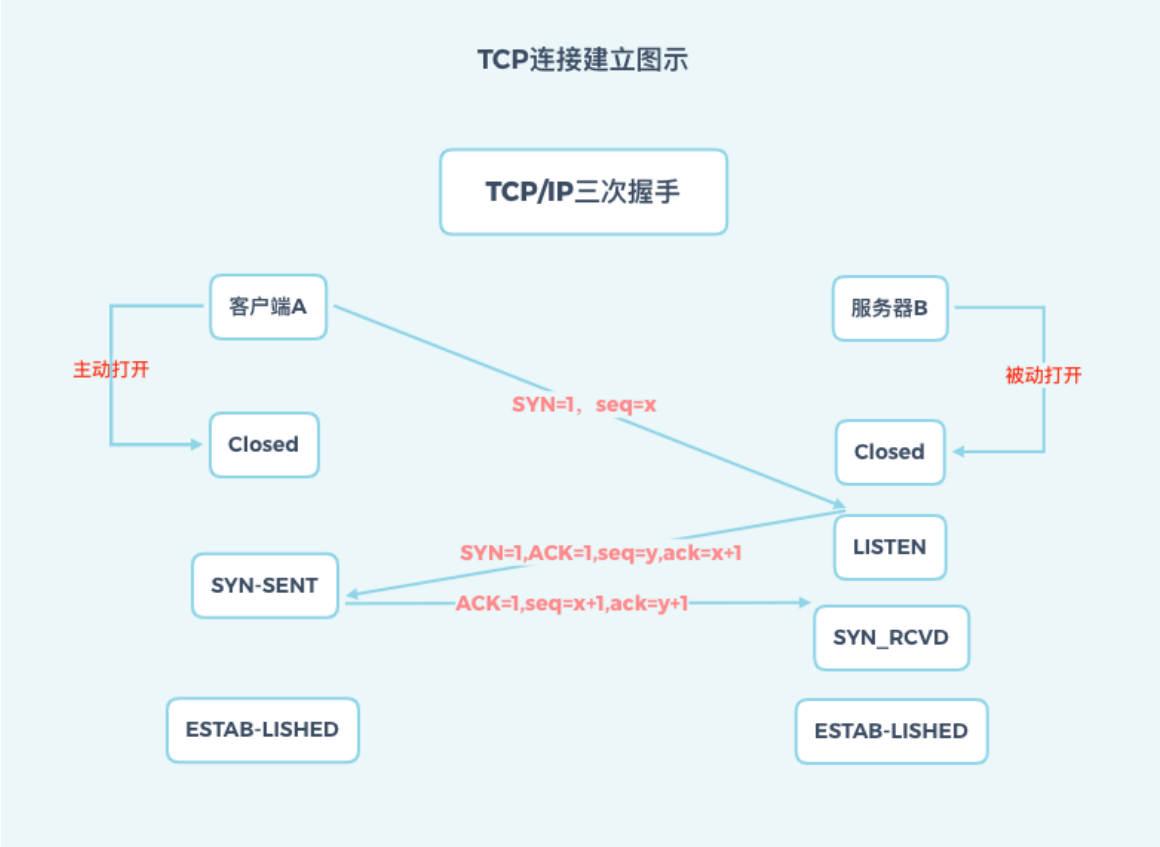

图 3-1-2-3 TCP/IP 三次握手

与 TCP Connect( )相比较之下，TCP SYN 扫描则是在客户端 A 与服务器 B 建立起三次握手之前，只完成了前两次握手，而不是建立起完整的连接。当我们向目标主机的端口发起连接请求的时候，如果目标主机返回来的数据包中包含有 RST 的响应，则说明这一端口是关闭的；如果目标主机返回来的数据包中包含有 SYN 和 ACK

数据，则说明这一端口是打开的，此时客户端会赶在建立连接前回包告诉目标主机端口已关闭，此时当然无法建立连接。因为这种扫描方式比较特别，只完成了前两次的握手状态，并没有建立完整的全连接，所以这种 TCPSYN 扫描技术也称之为半开放扫描。

具体的过程为，客户端 A 向服务器 B 发送伪造的 SYN 控制报文，如果此时收到返回的响应报文为 RST 报文，说明服务器 B 的端口为关闭状态；如果返回的是 SYN 和 ACK 报文，说明服务器 B 的端口为开放状态。随后客户端 A 会发送一个 RST 报文，告诉服务器 B 端口已关闭，此时服务器 B 端口接受不到客户端 A 的数据返回，所以无法建立三次握手连接。这样服务器 B 完全没有察觉，因为扫描前后的状态都没有改变。

SYN 扫描的优点在于这种扫描方式不需要建立起完整的三次握手，速度会相对较快。但是缺点就是这种构造 SYN 报文的扫描方式，通常需要超级用户。所以我选择了 TCP Connect（）扫描进行探测，这种方法的优点是所需要的权限较低，实现相对简单，而且可以使用多线程技术。虽然缺点是建立起全连接的三次握手会有很多记录，但是在网络与目标主机会产生大量连接记录的情况下，这点记录是相对来说不用担忧的。

3.3 Web 服务器旁站发现技术研究

3.3.1 技术概述

前面有提到旁站的相关概念，就是假设目标服务器上只有一个网站，当服务器还存在其他的网站的时候，那么这个网站相对于第一个网站而言就可以被称为旁站。如果在渗透测试的过程中，遇到了目标站点无法拿到权限的困难，那么可以对服务器的旁站进行探测。此操作可以查看旁站的存在情况，从而可以利用安全验证可能较弱的旁站进行突破，拿下整台服务器的权限。

一般个人站点或者是中小型的网站会存在比较多这样的情况，直接一台服务器上放置多个网站目录，通常这是为了节约成本。另外也有一些服务器出租的交易，就是与他人公用一台服务器，这样可以节约很多开销，但是安全的问题得不到保障。一些企业有时候为了贪图省事、节约成本，也会将官网、系统甚至是内部的网站放置在同一台服务器上。

这样的做法是很不妥当的，存在着很大的问题，虽然主站点设计的很安全，无法被入侵者突破拿下权限，但是旁站未必可以做到很安全。

3.3.2 旁站发现技术研究