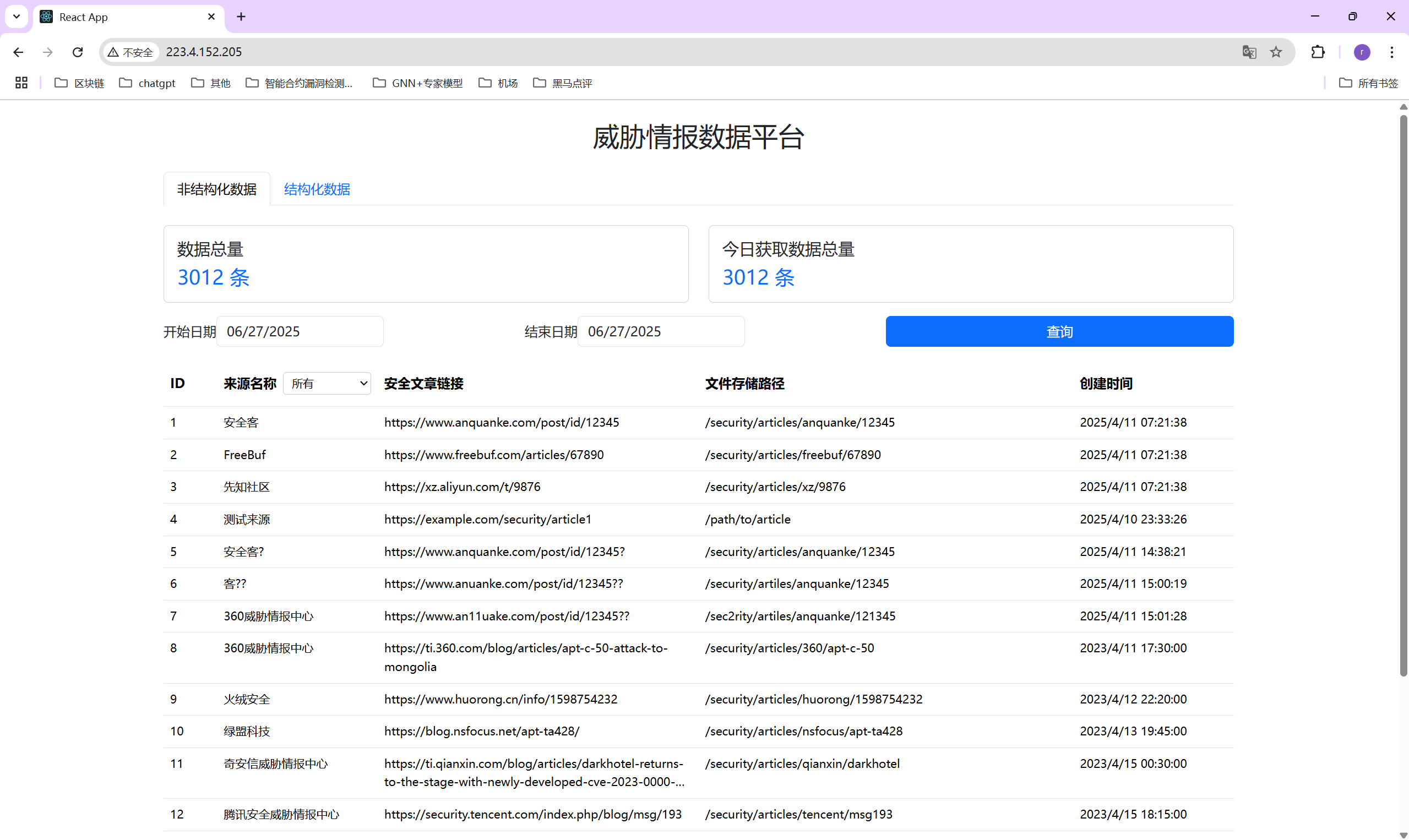
2025-06-27周报

马嘉润

1.将前后端项目部署到云服务器

将前后端项目部署到阿里云服务器，现在可以通过服务器的**公网IP地址**进行访问。

访问地址：http://223.4.152.205/



2.结构化数据关系扩充调研

核心思想是先定义好一个“图谱模板”，然后用从文本中抽出的关系三元组去填充它。这个模板就是预先定义好的节点类型和边类型

节点类型（6种）：

1. **Attacker (A)**: 攻击者。这可以是一个黑客组织的名字：如 IP 地址或 C2 服务器域名。

2. **Vulnerability (V)**: 漏洞。通常指一个 CVE 编号 (如 CVE-2017-0143 )。

3. **Device (D)**: 受影响的设备或软件。例如 Office 2012, Vista SP2。

4. **Platform (P)**: 更宏观的平台或操作系统。例如 Windows , Linux ,

5. **Malicious File (F)**: 恶意文件。可以是文件名 ( Android，SMB.bat ) 或文件哈希值 (MD5/SHA256)。

6. **Attack Type (T)**: 攻击类型。通常是标准的攻击分类，如 Remote Code Execution, Service , SQL Injection。

边类型（9种）：

连接不同类型节点的语义关系。连接、下载、属于、利用

R1: Attacker -exploit-> Vulnerability : 攻击者利用漏洞。（例如，"APT28 exploited CVE 2017-11882"）

R2: Attacker -invade-> Device: 攻击者入侵设备。（例如，"Lotus invaded Vista SP2 devices"） R3: Attacker -cooperate-> Attacker: 攻击者之间存在合作关系。（这个比较难直接抽取，可 能需要更高级的关联分析）

R4: Vulnerability -affect-> Device: 漏洞影响设备。（例如，"CVE-2017-0143 affects Windows 7"）

R5: Vulnerability -belong\_to-> Attack Type: 漏洞属于某种攻击类型。（例如，"CVE 2017-0143 is a Remote Code Execution vulnerability"）

R6: Vulnerability -include-> File: 漏洞涉及恶意文件。（例如，"CVE-2017-11882 involves the file EQNEDT32.EXE"）

R7: File -target-> Device: 恶意文件攻击特定设备。（例如，"This malware targets Office 2012"） R8: Vulnerability -evolve\_from-> Vulnerability : 漏洞的演进关系。（例如，"CVE-2018 0802 is an evolution of CVE-2017-11882"）

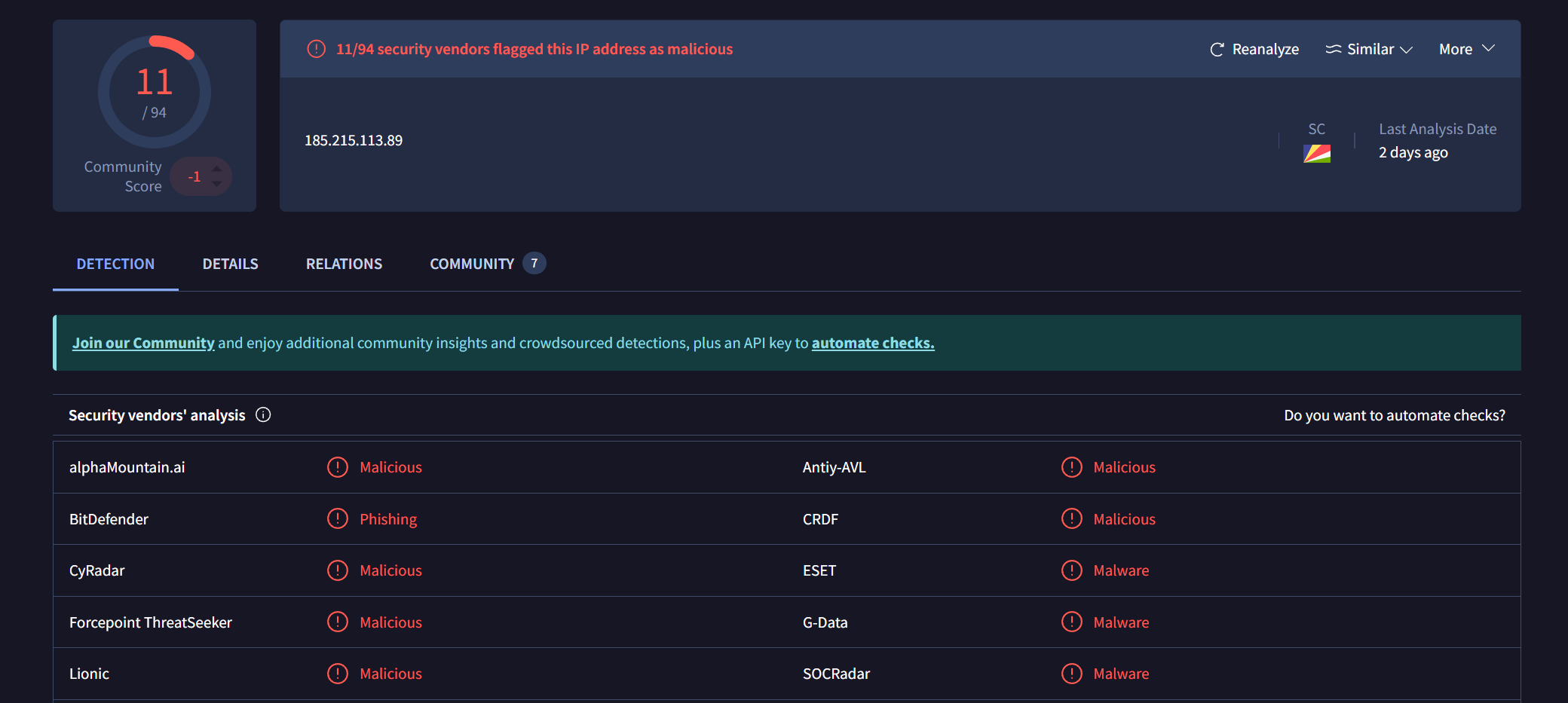
R9: Device -belong\_to-> Platform: 设备属于某个平台。（例如，"Office 2012 runs on Windows"）

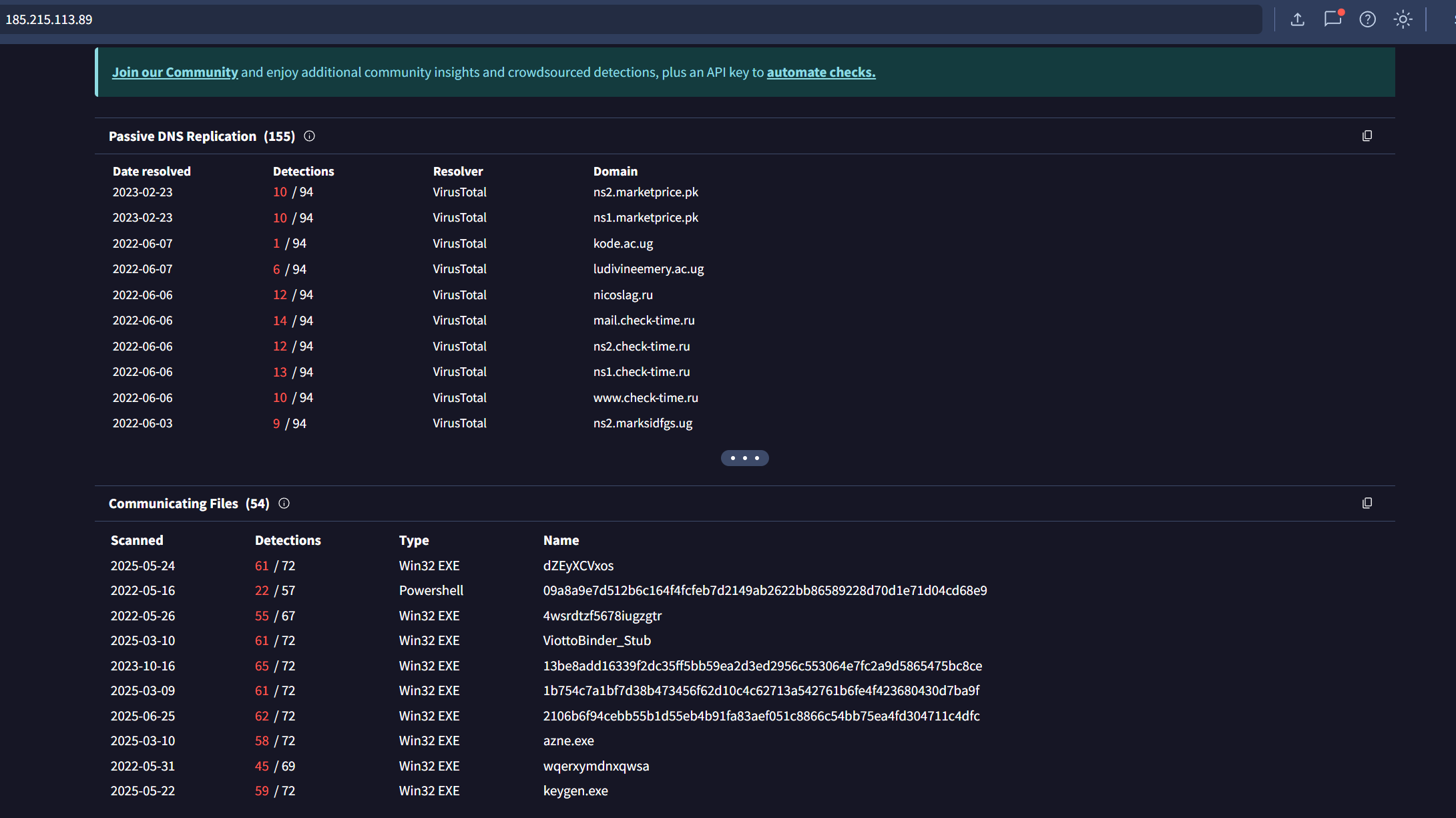
原文的数据源来源和关系的建立也是开发了爬虫系统，获取网页的html源码，然后xpath提取正文内容来解析。

结构化数据获取上下文方法：

（1）威胁情报平台 API:

VirusTotal，AlienVault OTX, ThreatConnect:用 API 提交MD5，ip等信息，VT 会返回极其丰富的信息，我们来构建md5域名的连接信息等等。





（2）之前爬取的安全报告和博客

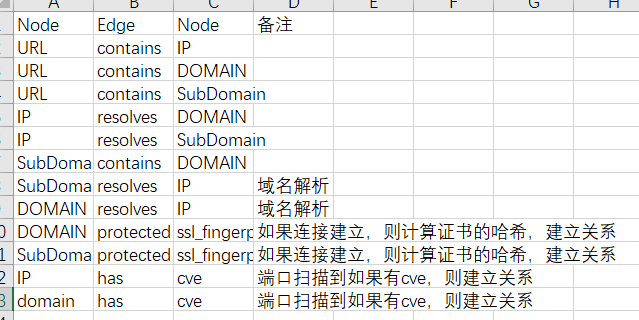
将IOC 中的ip，md5 作为关键词，在正文中检索。如果同时提到一个 IP 和一个 MD5，简单方法就是建立一条 related\_to 的边。

或者对正文进行句法依赖解析，构建更精确的关系类型，exploit 或 affect 。

曾家豪-本周工作

1. 调研结构化数据的边和节点，如下图所示

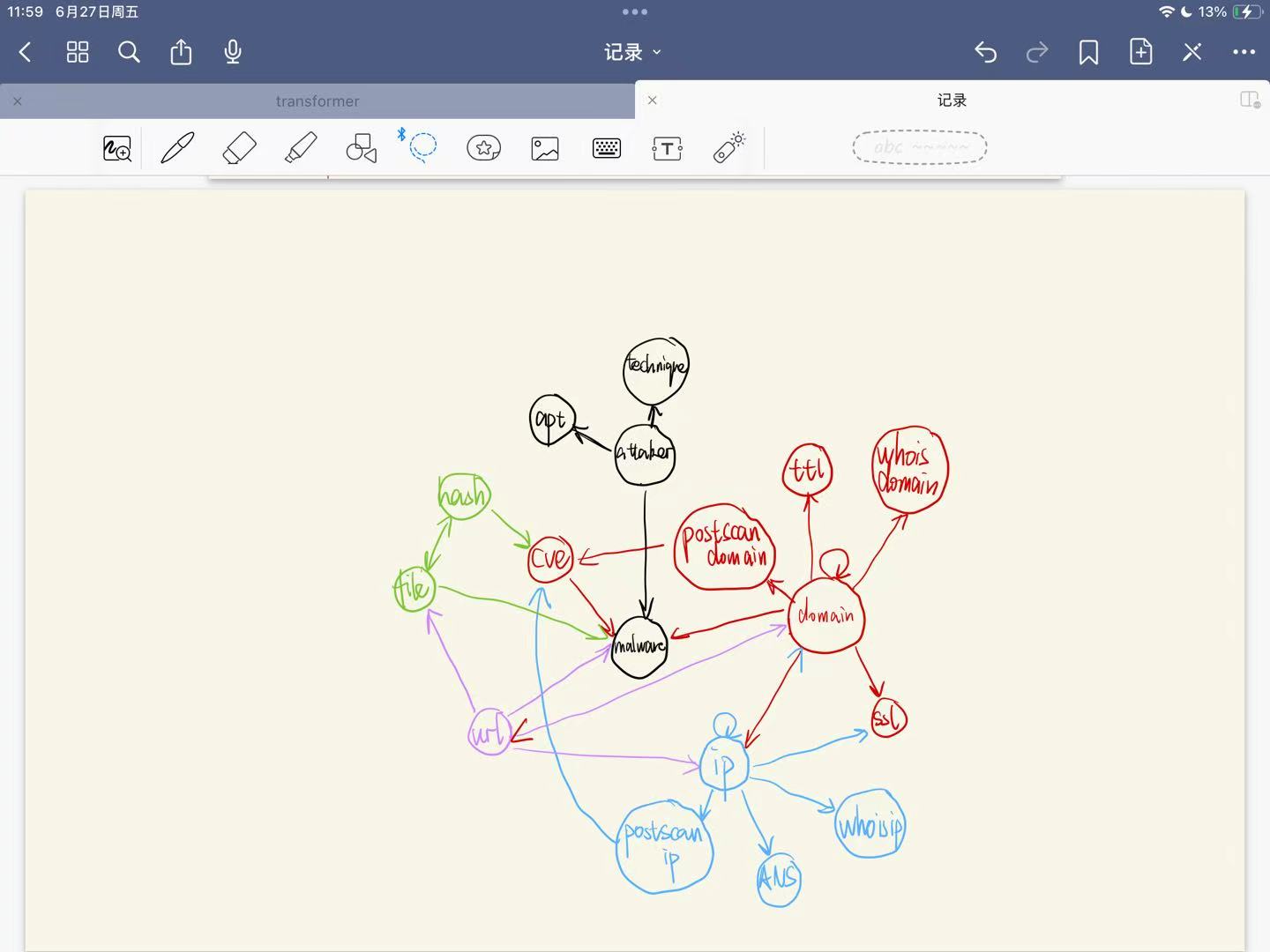
由于论文只详细讲了部分的三元组关系，有些没涉及，因此要获取剩下的关系得从它提供的github内获取相关的图数据。



1. 正在部署论文的noe4j图数据库的相关环境，来获取所有的三元组图数据

冯世玲——本周周报

|  |  |
| --- | --- |
| DNS查询 | domain解析ip  domain共址关系  获取ttl |
| 证书分析 | 获取domain/ip证书指纹  domain共址关系 |
| Whois查询 | domain共址关系 |
| Web 指纹 | domain共址关系 |
| HTTP 状态码重定向 | Domain重定向至url  Url重定向至url |
| 端口扫描 | 扫描主机上的漏洞 |
| 被动DNS/主机记录分析 | ip托管的domain |
| 流量通信日志分析 | 查询ip间的通讯关系和活跃状态 |
| ASN查询 | IP归属自治系统 |
| 域名提取 | url提取domain |
| 下载行为分析 | 恶意文件落地链接 |
| cve数据库对比 | 确定Hash和cve之间的关系 |
| 攻击者分析 | 攻击者使用的工具、国家/黑客组织归属、MITRE ATT&CK中的攻击技术 |
| malware分析 | 查找关联的domain、url、cve |



|  |  |  |  |
| --- | --- | --- | --- |
| 节点 | 关系 | 节点 | 获取信息 |
| domain | SHARED | dmain | 1. DNS查询：IP地址共用、ASN 相同、CNAME 重合 2. 证书分析：TLS/SSL证书共用 3. Whois查询：注册信息相同或关联（如邮箱、注册商） 4. Web 指纹：HTTP 响应特征相同 |
|  | RESOLVES | ip | DNS查询：将domain解析成一个或多个ip |
|  | ttl\_info | ttl | DNS查询：恶意域名往往设置较低 TTL以实现**快速变换IP地址** |
|  | POINTS\_TO | url | HTTP 状态码重定向（状态码30x）：Domain重定向至url |
|  | Has\_scan | portscandomain | 端口扫描：通过端口扫描，可以检测目标主机上的开放端口，并确定这些端口上运行的服务及其版本。如果相关服务和版本报告了CVE（计算机安全漏洞），则可以指出被扫描主机上的漏洞。 |
|  | HAS\_SCAN\_RESULT | cve | 端口扫描 |
|  | PROTECTED | ssl | 证书分析：系统会尝试连接到首选的HTTPS端口，如80、443、2082、4443、5000、8008、8080、8443、9443和10443。如果成功建立连接，将计算证书的指纹。 |
|  | LOOKUP | whoisdomain | whois查询 |
|  | ASSOCIATED\_WITH | malware | 域名被用于传播或C2通信 |
| ip | HOSTS | domain | 被动DNS/主机记录分析 |
|  | communicate | ip | 流量通信日志分析：Netflow、pcap等提取IP对,查询ip间的通讯关系和活跃状态 |
|  | asn\_info | ans | ASN查询：IP归属自治系统 |
|  | PROTECTED | ssl | 证书分析 |
|  | Has\_scan | portscanip | 端口扫描 |
|  | HAS\_SCAN\_RESULT | Cve | 端口扫描 |
|  | LOOKUP | whoisip | whois查询 |
|  | ASSOCIATED\_WITH | malware | 与恶意软件通信或投递地址 |
| url | HOSTED\_BY | domain | 域名绑定URL |
|  | HOSTED\_ON | ip | HTTP请求IP反查 |
|  | REDIRECTS\_TO | url | HTTP 状态码重定向 |
|  | ASSOCIATED\_WITH | malware | URL为恶意负载承载或回传地址 |
|  | CONTAINS\_FILE | file | 下载行为分析：恶意文件落地链接 |
| file | IDENTIFIES | hash | Hash为文件指纹 |
|  | ASSOCIATED\_WITH | malware | 文件归属某恶意家族 |
| hash | EXPLOITS | cve | cve数据库对比 |
| attacker | USES\_MALWARE | malware | 攻击者使用的工具 |
|  | ATTRIBUTED\_TO | apt | 国家/黑客组织归属分析 |
|  | HAS\_TACTIC | technique | 使用MITRE ATT&CK中的攻击技术 |