MILESTONE 3

Taja Hicks
Department of Computer Science
Texas A& M San Antonio University
thick02@jaguar.tamu.edu

Luis Morales
Department of Computer Science
Texas A& M San Antonio University
lmora027@jaguar.tamu.edu

Abigail Rodriguez Vazquez
Department of Computer Science
Texas A& M San Antonio University
arodr0378@jaguar.tamu.edu

ABSTRACT

This paper presents the goal, use cases, and results of the OSINT tool, Blackbird. Blackbird is a tool with key features including free AI-powered insights, metadata extraction, WhatsMyName integration, export options and customizable filters. Practical applications of Blackbird are digital investigations, social media research, compliance and verification, and data collection and analysis. This paper goes over the goal and purpose of Blackbird, why someone would want to use this tool, comparing similar tools against Blackbird, and our contribution to the tool Blackbird. This paper will explain how to install and use Blackbird, go over important code within

1 Introduction to BlackBird

Blackbird is an OSINT tool that combines username and email searches across more than 600 platforms [1]. This is done using AI to profile. By utilizing community-driven projects (like WhatsMyName), it lowers false positive rates and provides high-quality results. Some main Features include smart filters, polished PDF/CSV exports, and more [1].

1.1 Goal and Purpose of Blackbird

The goal of blackbird is to use AI to analyze the sites where a username or email is found and returns a behavioral and technical profile of the user. This Helps you understand more about someone whether it be what language their learning, what platforms they participate in. As well as some of Risk they might encounter, all with less effort [1] .

1.2 Why would you want to use Blackbird When it comes to Computer Security?

- Detecting Possible Data Breaches: You can use Blackbird to better locate the websites or apps you are associated with and help you see if you might have been impacted by a breach.
  - This can also help you ensure that you use different passwords for different accounts on different platforms.
- Becoming more Aware/Mindful: Since Blackbird returns a behavioral and technical profile of the user, we can become more aware of our online presence.
  - This awareness can help you better understand some of the vulnerabilities you may encounter and mitigate them.
  - In addition to being more mindful of our computer security and implementing more secure login systems, such as two-factor authentication.

1.2.1 Setting up Blackbird

Below we provide a link to demonstrate how to install and use Blackbird.

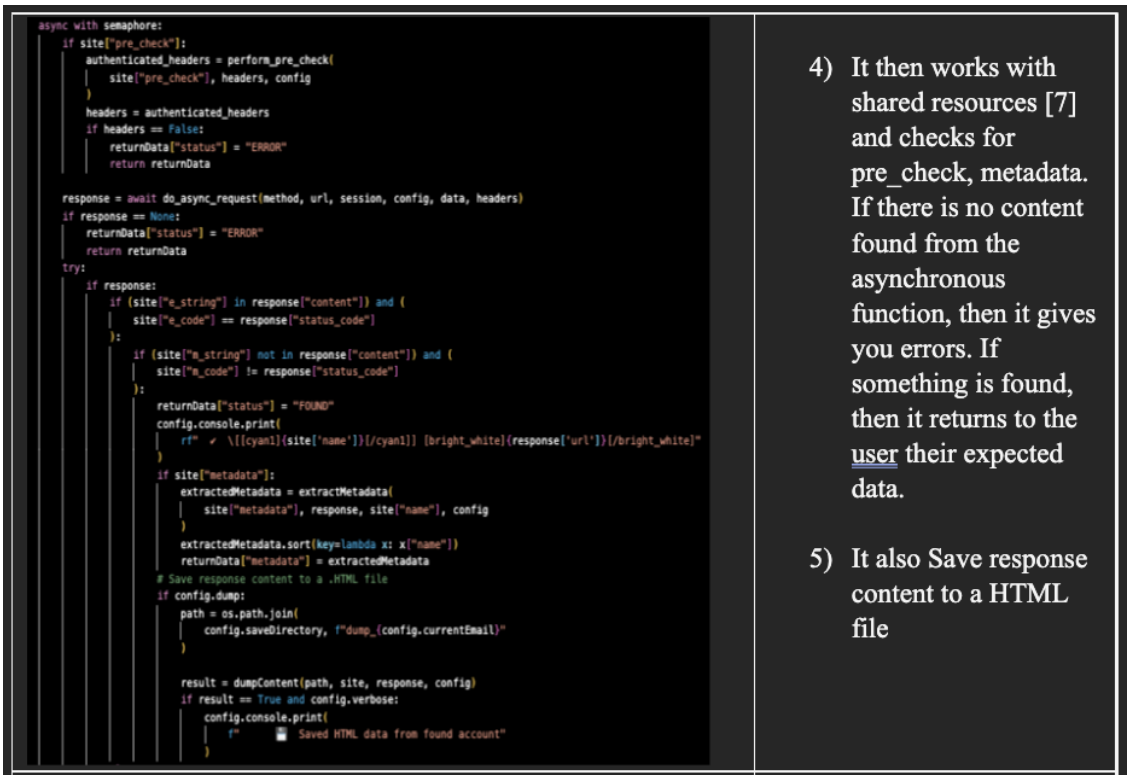
`https://colab.research.google.com/drive/1l0EaYnDW7ltHTDp0RUflfRGaGup5unUk?usp=sharing`

Example of Code: Explaining whats happening

1.3 Email.py

- This takes place in the email.py in blackbird

Figure 1: **Email.py**

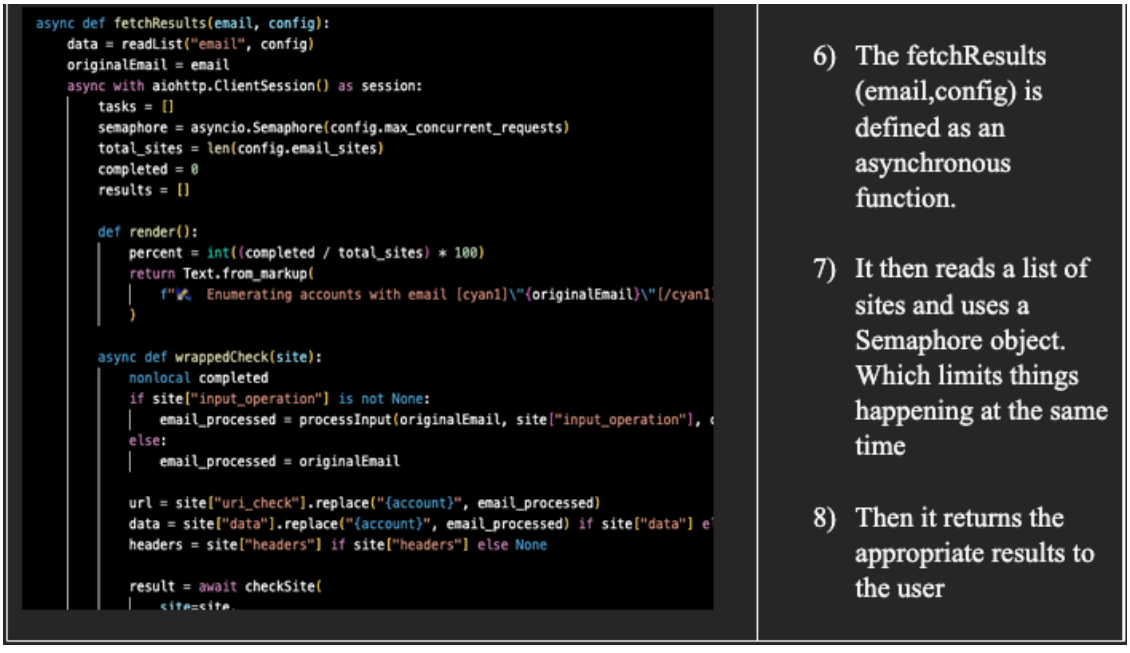


```
async with semaphore:
    if site["pre_check"]:
        authenticated_headers = perform_pre_check(
            site["pre_check"], headers, config
        )
        headers = authenticated_headers
        if headers == False:
            returnData["status"] = "ERROR"
            return returnData

    response = await do_async_request(method, url, session, config, data, headers)
    if response == None:
        returnData["status"] = "ERROR"
        return returnData
    try:
        if response:
            if (site["e_string"] in response["content"]) and (
                site["e_code"] == response["status_code"]
            ):
                if (site["m_string"] not in response["content"]) and (
                    site["m_code"] != response["status_code"]
                ):
                    returnData["status"] = "FOUND"
                    config.console.print(
                        rf"  ✓ \[[cyan1]{site['name']}[/cyan1]] [bright_white]{response['url']}[/bright_white]"
                    )
                    if site["metadata"]:
                        extractedMetadata = extractMetadata(
                            site["metadata"], response, site["name"], config
                        )
                        extractedMetadata.sort(key=lambda x: x["name"])
                        returnData["metadata"] = extractedMetadata
                    # Save response content to a .HTML file
                    if config.dump:
                        path = os.path.join(
                            config.saveDirectory, f"dump_{config.currentEmail}"
                        )

                        result = dumpContent(path, site, response, config)
                        if result == True and config.verbose:
                            config.console.print(
                                f"      💾 Saved HTML data from found account"
                            )
```

4) It then works with shared resources [7] and checks for pre_check, metadata. If there is no content found from the asynchronous function, then it gives you errors. If something is found, then it returns to the user their expected data.

5) It also Save response content to a HTML file

Figure 2: **Email.py**



```
async def fetchResults(email, config):
    data = readList("email", config)
    originalEmail = email
    async with aiohttp.ClientSession() as session:
        tasks = []
        semaphore = asyncio.Semaphore(config.max_concurrent_requests)
        total_sites = len(config.email_sites)
        completed = 0
        results = []

        def render():
            percent = int((completed / total_sites) * 100)
            return Text.from_markup(
                f"🔍  Enumerating accounts with email [cyan1]\"{originalEmail}\"[/cyan1
            )

        async def wrappedCheck(site):
            nonlocal completed
            if site["input_operation"] is not None:
                email_processed = processInput(originalEmail, site["input_operation"],
            else:
                email_processed = originalEmail

            url = site["uri_check"].replace("{account}", email_processed)
            data = site["data"].replace("{account}", email_processed) if site["data"] e
            headers = site["headers"] if site["headers"] else None

            result = await checkSite(
                site=site,
```

6) The fetchResults (email,config) is defined as an asynchronous function.

7) It then reads a list of sites and uses a Semaphore object. Which limits things happening at the same time

8) Then it returns the appropriate results to the user

Figure 3: **Email.py**

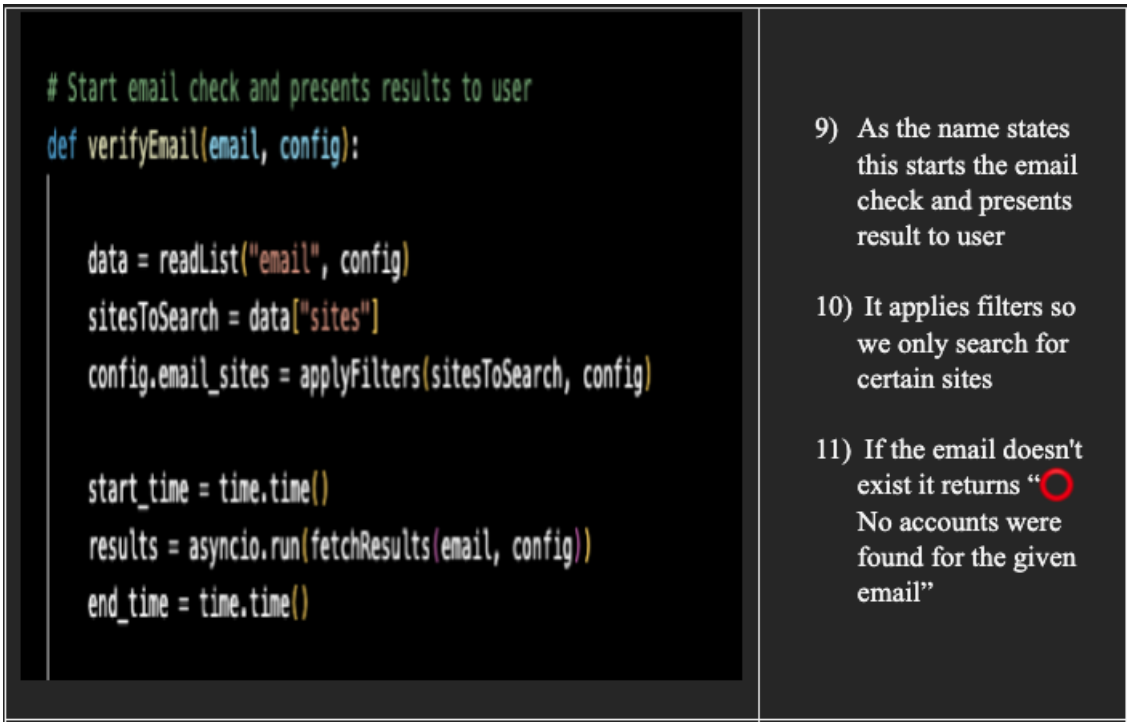


Figure 4: **Email.py**

1.4 Username.py

- This takes place in the user.py in blackbird

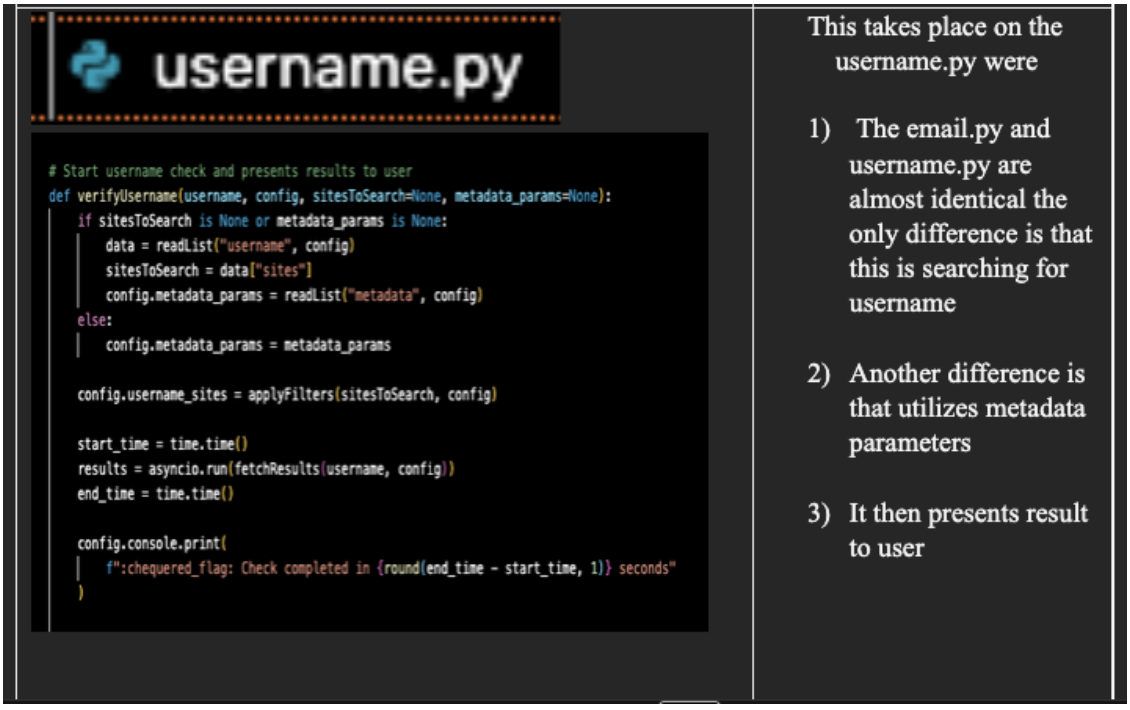


Figure 5: **User.py**

1.5 Pdf.py

- This takes place in the Pdf.py in blackbird

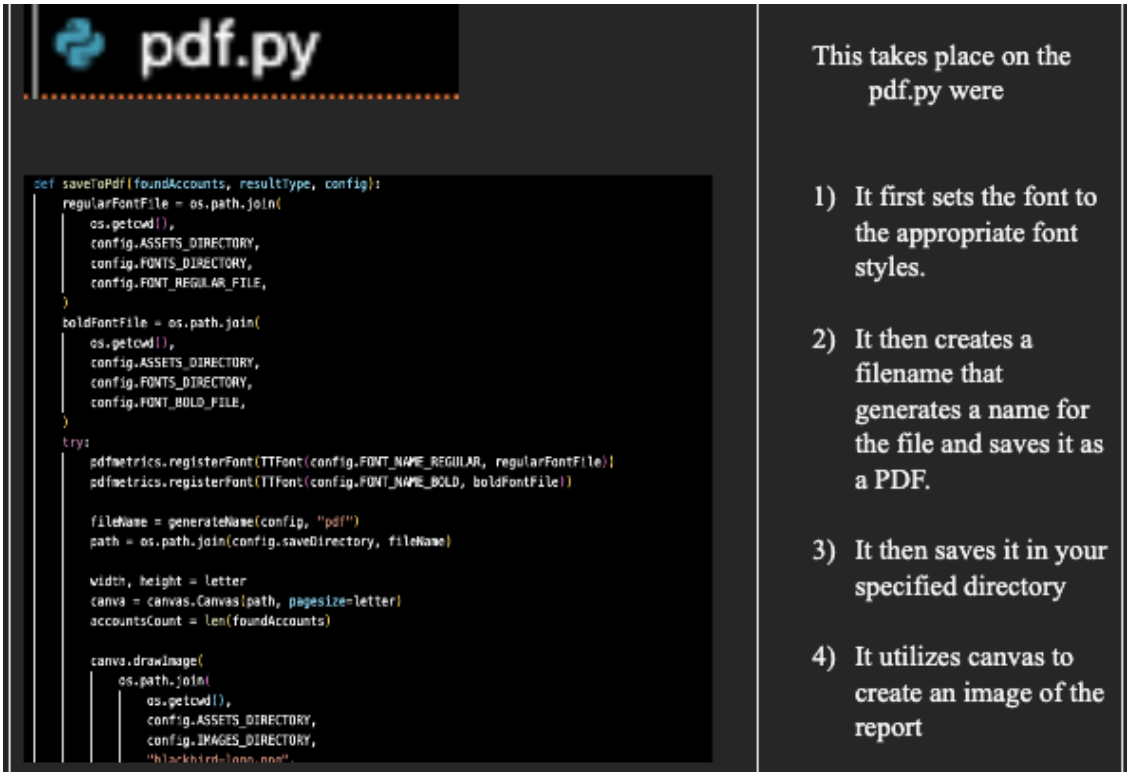


Figure 6: **Pdf.py**

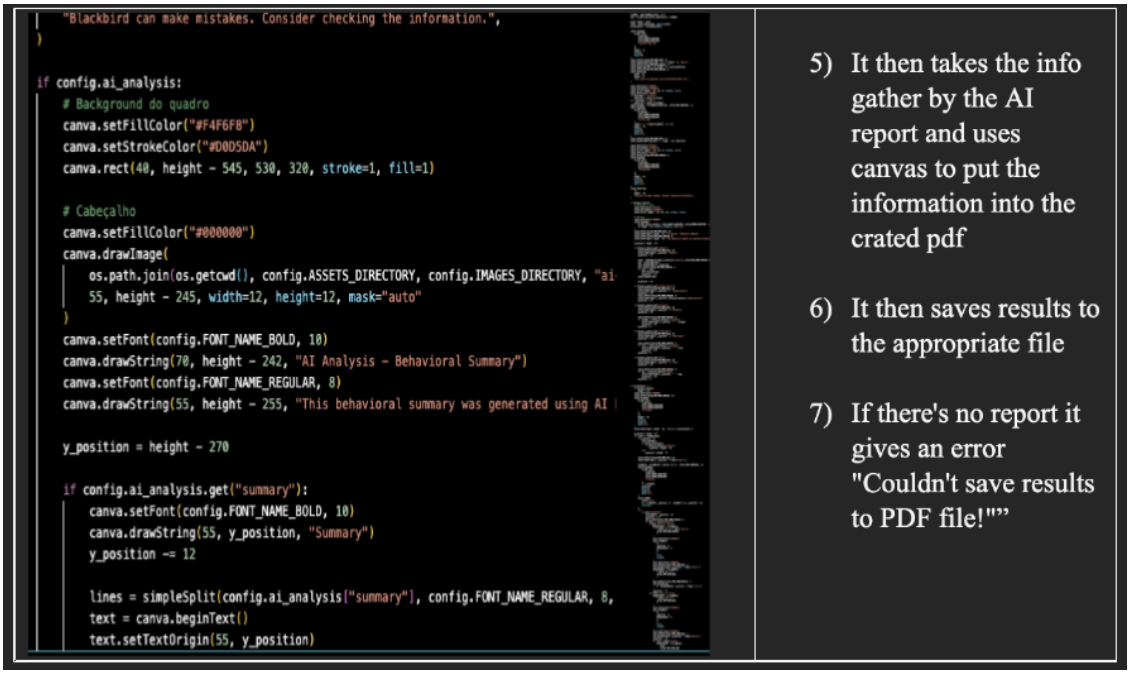


Figure 7: **Pdf.py**

1.6 Data.json

- This takes place in the email-data.json in blackbird

email-data.json

```
"categories": [
    "archived",
    "art",
    "blog",
    "business",
    "coding",
    "dating",
    "finance",
    "gaming",
    "health",
    "hobby",
    "images",
    "misc",
    "music",
    "news",
    "political",
    "search",
    "shopping",
    "social",
    "tech",
    "video",
    "xx NSFW xx"
]
```

This is a list of all available categories to both categorize sites and be used for filtering search results.

Figure 8: **Email-data.json**

```
"name": "Notion",
"uri_check": "https://www.notion.so/api/v3/getLoginOptions",
"data": "{\"email\":\"{account}\"}",
"method": "POST",
"headers": {
    "Content-Type": "application/json"
},
"e_code": 200,
"e_string": "\"hasAccount\":true",
"m_string": "\"hasAccount\":false",
"m_code": 404,
"known": [],
"cat": "misc",
"input_operation": null,
"metadata": null,
"pre_check": null
```

- This object defines how to interact with Notion's login API to check if an account (email) exists. The name of the service being checked — in this case, Notion.
- The API endpoint used to verify login options for a given email. This is where the POST request will be sent.
- The payload sent in the POST request. It's a JSON string with an email field. {account} is a placeholder that will be replaced with the actual email being tested.
- Specifies the HTTP method used to send the request — here, it's a POST.

Figure 9: **email-data.json**

1.7 Wmn-data.json

- This takes place in the wmn-data.json in blackbird

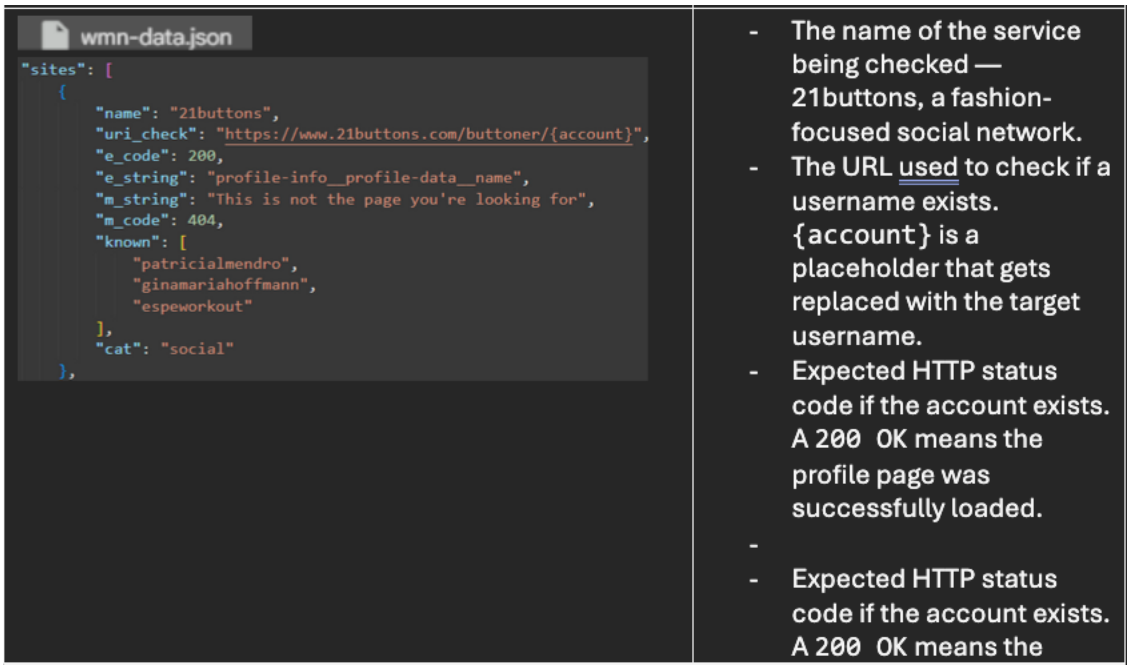


Figure 10: **Wmn-dat**

1.8 Blackbird.py

- This takes place in the Blackbird.py in blackbird

Figure 11: **Blackbird.py**

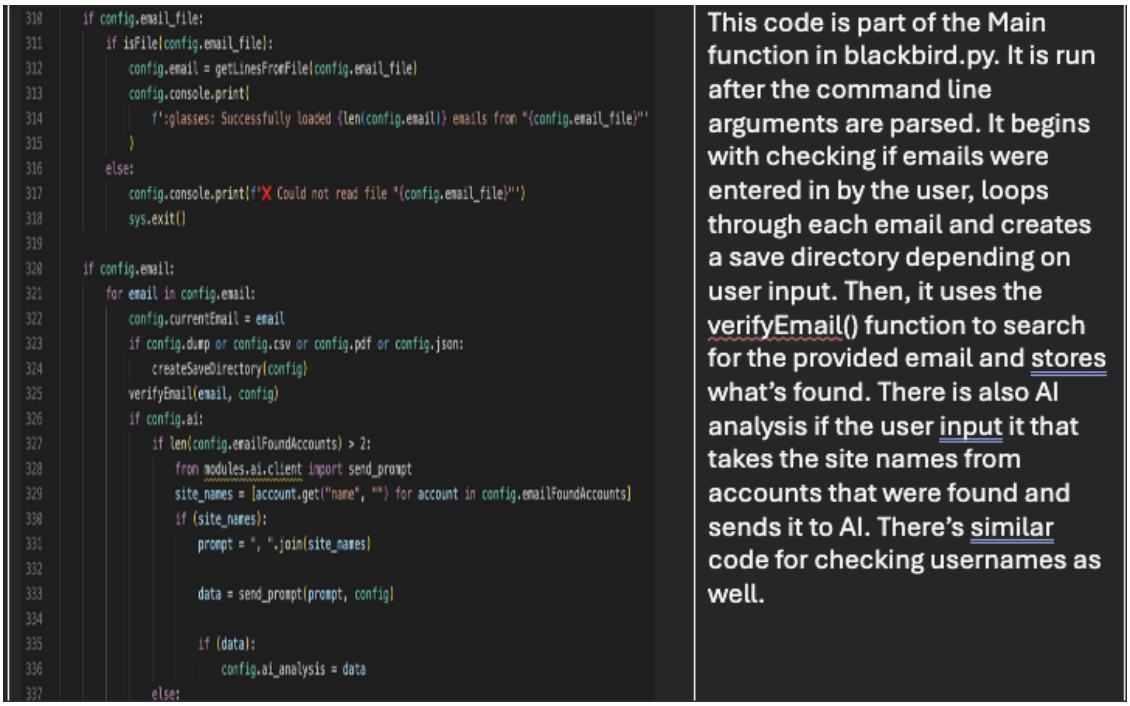


Figure 12: **Blackbird.py**

1.9 Cleint.py

- This takes place in the AI directory in blackbird

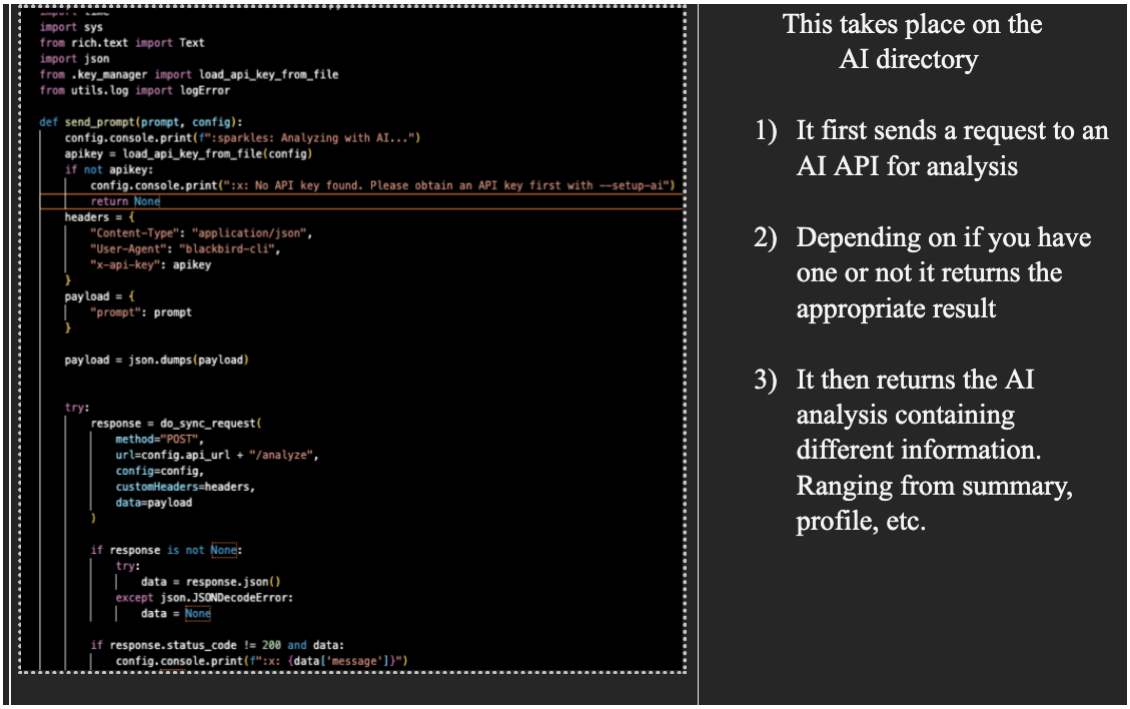


Figure 13: **AI directory**

FEEDBACK :My only comment will be on the last third related to code output or usage highlight and scenarios, to add more, thank you

2 Tools Comparison: Ghost-Track, Nexfil, BlackBird

Below we provide a link for Sample demos we made of how each tool is used (practical part)

Ghost-Track

`https://colab.research.google.com/drive/1vUeIXphxsZVIiFxSkOFtulSmAh-bKR_-?usp=sharing`

Nexfil

`https://colab.research.google.com/drive/1Q8DRwMWuybYTtYhVuObIqMOrn-WCz-xd?usp=sharing`

2.1 Strengths and weaknesses of tools:

2.2

Strengths for Blackbird

- Blackbird has a broader range of platforms, totaling 600.
- This provides a more in-depth look into the user, offering an analysis of the user's technical profile. As [9] mentions, this helps you understand more with less effort.
- It also includes polished PDF/CSV exports, making reading the information easy to read.

Weakness for Blackbird

- Although Blackbird has many advantages, it does not provide access to users' IP addresses or location. This information can be useful in some cases.
- It's also slower because of the number of results from queries

2.3

Strengths for Ghost-Track

- Ghost-track is very useful when it comes to tracking location or mobile numbers.
- Helping you gather information about a user, whether it be what country they live in or their capital.
- Ghost-track also tracks the username and checks what websites the user is on
- Also useful in physical threat scenarios

Weakness for Ghost-Track

- While Ghost-track also tracks username, there is a smaller range of platforms compared to the likes of Blackbird
- The information is good, but also very minimal.

2.4

Strengths for Nexfil

- Fast lookups that can be completed under 20 seconds.
- Allows files containing usernames to be searched .

Weakness for Nexfil

- Smaller range of platforms (over 300) when compared to Blackbird
- Limited to saving results to txt file only.
- Limited features

3 Related work

FEEDBACK :Related work, is OK but some papers seem not to be related to the focus need to pick which are relatable

3.1 Literature Review

Open-Source Intelligence (OSINT) tools have become indispensable in cybersecurity, digital forensics, and ethical hacking. They automate the collection of publicly available data, enabling investigators to uncover digital footprints

across social media, geolocation services, and online platforms. This review analyses the OSINT tool available on GitHub — Blackbird (Our Program), and places it in the context of broader academic and technical fields. We then compare their usage of Blackbird and other OSINT tools to our usage.

This paper [2] was written to undertake a study on how CITNT conducts activities to encourage citizens to participate in contributing to intelligence activities and decision-making processes. As it seems, the goal of this is more of a discussion on how we can integrate CITNT into our daily lives and government. In addition to exploring the benefits that would come from doing so. Another thing to note is that this research is primarily taking place in Finland, which means it is targeting a specific demographic. They do mention that some of the tools they utilize are Ukrainian Delta and Blackbird. From the article, it can be inferred that these tools are used to better understand and involve civilians in intelligence efforts, ultimately aiming to create a more informed and engaged public.

In our team's case, we had a completely different goal when it came to utilizing these tools. Our goals are to discuss possible computer security vulnerabilities, to uncover digital footprints, and to encourage safer online presence. We're attempting to answer queries about how we can better secure ourselves if we utilize various platforms with the same usernames. Some answers we found are integrating more secure login techniques, such as two-factor authentication. Our demographic is also broader than [2] since we are not really considering only one place.

Another paper we came across that used Blackbird was [3]. GoodFatR is a tool introduced in this article to collect threat reports and extract their Indicators of Compromise (IOCs). It appears that they deployed Blackbird to do an alternate validation, giving them user credentials tied to threats discovered by their tool. This takes a different approach because they created their own tool and just used Blackbird as supplementary support. The key tool for this research is Blackbird, which is used to identify potential computer security flaws. GoodFatR appears to be an effective tool to find indicators of compromise. We can investigate if it's possible to integrate certain features of this tool into our project and provide an IoC report on potential username input.

OSINT tools have many uses, and one of the uses we'll explore is its use in detecting and preventing organized crime. [4] is a paper that goes over organized crime and how those involved in it use information technology systems in their work. CAPER has six analysis modules, but the one this paper focuses on is the CAPER Facebook analysis submodule. This paper also focuses on the interactions between users through posts and comments [4]. CAPER integrates many things such as text, images, or video to detect connections or patterns related to crime. It's on a much larger scale when compared to Blackbird. CAPER is used in this paper to analyze names, locations, organizations, and the relationships between them to create a user-friendly network graph [4].

Facebook is a social media platform used by many to post photos, videos, and information about things like daily life. There is a lot of information to be found on Facebook because of this including personal information such as name, relatives, friends, general location, and more. The CAPER Facebook module performs tasks including gathering information from Facebook pages, groups, events, and users [4]. Specific information can include the date the post was created, likes and comments on the post as well as mentions [4]. All of this data can be gathered and used for various purposes and in the case of the paper, used to detect and prevent organized crime. Blackbird is smaller scaled and performs fast checks for usernames with cyber security analysts being the main audience. The main audience for CAPER would include professional intelligence analysts and the police. We used blackbird and we are limited to a smaller number of features compared to CAPER.

Another paper we will explore is [8] and its use of the OSINT tool, Maltego. Maltego is an OSINT tool, like Blackbird but with a broader scope. Unlike Blackbird, Maltego uses interactive graphs for visualization, offers free and paid tiers for more results, features and more, and is popular in penetration testing. In [8], Maltego is used to conduct security testing and focuses on examining the website of X Company to collect information to be used to help test its security. The information on vulnerabilities found is used to give recommendations to X company so that its website can be more secure [8]. In the case of [8], Maltego was used to collect and analyze information for intelligence purposes compared to Blackbird which has features limited to username and email lookup across different sources. Because of the difference in features, Blackbird is limited to certain usage unlike Maltego.

InfoHound is an OSINT tool that reminded us that, every time something is published on the internet it gets indexed by many services [5]. InfoHound retrieves all publicly available information given a domain name. It will then help visualize which degree of exposure a web domain has on the internet. With the proliferation of users spread across public platforms, the need for tools that can efficiently collect, correlate, and analyze publicly available data has grown significantly. To this point, Blackbird, builds on the principles established by previous Open Source Intelligence (OSINT) applications. Offering both a modular and passive-first OSINT framework focused on identity resolution, account discovery, and analyst usability. Blackbird bridges gaps left by other tools by emphasizing identity enrichment and cross-platform account discovery. Instead of just listing emails or usernames, it tries to correlate them with actual profiles, services, and leaked credentials. This tactic shows a shift from simple enumeration towards actionable

intelligence. By querying hundreds of platforms for the presence of a given username, Blackbird can map digital identities across services. This feature allows for enhanced profiling, fraud detection, and adversary attributes(attacker's motivations, capabilities, and methods).

While Blackbird and similar tools have made a lot of progress, there is room for improvements. In the case of Blackbird relying on username matching alone, the potential for false positives is present. Connecting Blackbird to a visualization platform could potentially enhance its investigative uses [5]. Having a modular framework allows such enhancements and expansions on its utility. While also making it a strong candidate for further research and development in OSINT tooling. "The rapid development of CITINT processes and organizations places specific demands on legislative work and the definitions contained in international agreements" [9]. As non-state actors increasingly engage in intelligence gathering—often using sophisticated tools and platforms—the boundaries between civilian and combatant roles become blurred, challenging traditional legal frameworks.

4 Experiments and analysis

FEEDBACK:section 3 is not clear to me at all, I am not sure really what you have from others vs what you are trying to do, details are not enough to tell

5 Summary and Conclusion

Your conclusion here

Acknowledgments

This was supported in part by.....

References

[1] L. Antoniaci, BlackBird, 2020. [Online]. Available: `https://p1ngul1n0.gitbook.io/blackbird` .

[2] A. Franchino, S. V. W. Virdone, and F. De Marco, "Integrating Political Science and Artificial Intelligence: The Future of Political Analysis," Frontiers in Political Science, vol. 6, no. 1379789, 2024. [Online]. Available:`https://www.frontiersin.org/articles/10.3389/fpos.2024.1379789/full`

[3] J. Caballero, G. Gomez, S. Matic, G. Sánchez, and S. Sebastián, "GoodFATR: A Platform for Automated Threat Report Collection and IOC Extraction," 2022. [Online]. Available: `https://carlesi.vg/wp-content/uploads/2022/11/goodfatr.pdf`

[4] C. Aliprandi et al., "CAPER: Crawling and analysing Facebook for intelligence purposes," 2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2014), Beijing, China, 2014, pp. 665–669, doi: 10.1109/ASONAM.2014.6921656 .

[5] prajjqv, "Python async," GeeksforGeeks, 23-Jul-2025. [Online]. Available: `https://www.geeksforgeeks.org/python/python-async/`

[6] W3Schools, "Python Dictionaries," W3Schools Online Web Tutorials. [Online]. Available: `https://www.w3schools.com/python/python_dictionaries.asp`

[7] P. S. Foundation, "Synchronization primitives," Python Documentation. [Online]. Available:`https://docs.python.org/3/library/asyncio-sync.html`

[8] I. P. A. E. Pratama and A. A. B. A. Wiradarma, "Open Source Intelligence Testing Using the OWASP Version 4 Framework at the Information Gathering Stage (Case Study: X Company)," Int. J. Comput. Netw. Inf. Secur. (IJCNIS), vol. 11, no. 7, pp. 8–12, Jul. 2019, doi: 10.5815/ijcnis.2019.07.02.

[9] P. Carlesi, "Blackbird," GitHub. [Online]. Available: `https://github.com/p1ngul1n0/blackbird.`

[10] Xavier Marrugat Plaza, "InfoHound Tool – Improving OSINT open source CyberArsental for good," 2023.[Online]. Available: `https://openaccess.uoc.edu/items/9f1c6491-f7c2-4cda-b8a2-9ea7161ab83f`

[11] p1ngul1n0, "Blackbird: An OSINT tool to search for accounts by username and email in social networks," GitHub repository, GitHub, Inc. [Online]. Available: `https://github.com/p1ngul1n0/blackbird`

[12] thewhiteh4t, "Nexfil: OSINT tool for finding profiles by username," GitHub repository, GitHub, Inc. [Online]. Available: https://github.com/thewhiteh4t/nexfil

[13] Iikka Pietilä, Katleena Kortesuo, Ulla Pohjanen, Mikko Tuominen, "Kansalaisten tekemä CITINT ja sen vertautuminen valtiolliseen tiedusteluun," [Online]. Available: https://www.researchgate.net/profile/Katleena_Kortesuo2/publication/383941320_Kansalaisten_tekema_CITINT_ja_sen_vertautuminen_valtiolliseen_tiedusteluun/links/66e15aca2390e50b2c7e9849/Kansalaisten-tekemae-CITINT-ja-sen-vertautuminen-valtiolliseen-tiedusteluun.pdf#page=51