Uncovering OSINT Insights from 15TB of GitHub Logs

In our role as research cybersecurity experts, we are constantly on the lookout for valuable information hiding in everyday data. In our profession, what may appear as trivial data can often serve as the doorway to significant insights, with large data repositories like GitHub being no exception. In our recent blog post, we elaborated on how we used the impressive capabilities of Trickest workflows to process an enormous dataset of nearly 15TB from GitHub logs. Our mission was to mine the rich seam of public information available about all the users and repositories encapsulated within these logs.

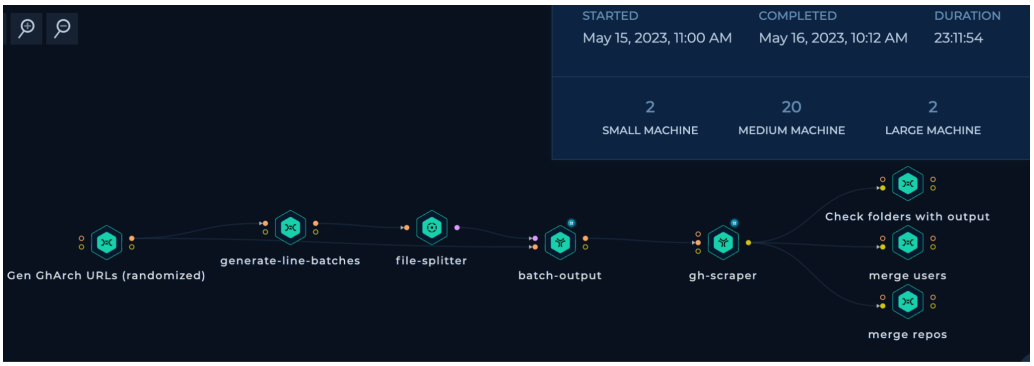
GitHub, a platform teeming with more than 56 million developers worldwide, houses an incredible array of public logs. While these logs are a treasure trove for researchers and developers, it can also potentially be exploited for nefarious purposes if not properly understood and handled. As experts in data security and privacy, we understand the implications of such openly available information and believe in the importance of scrutinizing this data to ensure better privacy practices.

In this report, we intend to take you on a deeper exploration of this data. **We aim to unearth intriguing patterns, identify potential security risks, and provide insights on how to better protect your data on such platforms**. Our team's extensive experience and expertise in data science and cybersecurity have equipped us with the tools to decipher these vast datasets and understand their implications comprehensively.

Recap

We built several workflows to parse and analyze almost 15TB of Github logs from GH Archive from 2015 till the present. After, the data was enhanced using the GitHub API to extract more information about the users and repositories. Our final result included two CSV files: one housing data on over 45 million users and the repositories to which they contributed, and the other containing data on over 220 million repositories.

Here is the Trickest workflow we used for data processing:



To extract valuable insights from these CSVs, we used gh-investigator tool, available in the Trickest library with 270+ open-source tools.

GitHub User Analysis

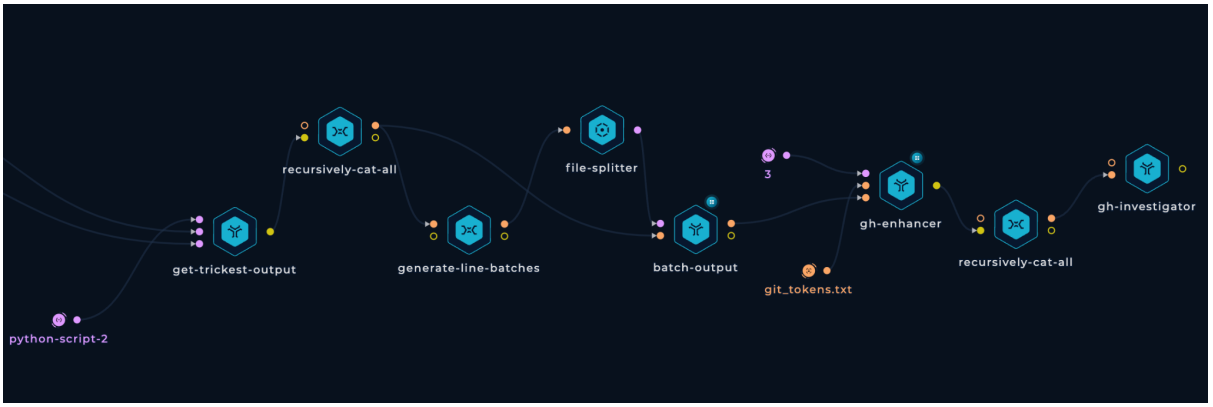
We extracted the following information for each user in our analysis:

- Username
- Repositories where the user has collaborated (capped at 500 to limit bot users who've collaborated on thousands of repositories)
- Deleted user status
- Site administrator status
- Hireability
- Availability of public email
- Information about the user's company
- Participation in the GitHub Stars program

Although we could extract much more information from the GitHub API, we opted to focus on the most relevant factors to manage the CSVs size.

With the help of *gh-investigator*, we generated several files, including lists of Github Star users, deleted users, hireable users, site administrators, users with public emails, and users with company information. **You can find the download links for these files at the conclusion of this report**.

Here is the Trickest workflow we employed to gather details about each identified user:



Deleted Users

These refer to user accounts that were active at one point but are no longer available. This unavailability could be a result of the user account being deleted or renamed. Regardless of the reason, it's crucial to note which users are no longer active, as it could pose an

impersonation risk. An attacker could create a new user account with the same name, attempting to impersonate the original user.

Please be aware that GitHub's Action feature allows workflows triggered by a `pull_request` from an external user to run automatically for that user's subsequent PRs once a PR is successfully merged. However, if a user account is deleted and a new one is created with the same name, these permissions ARE NOT preserved. You can find more information about this at GitHub Security.

Our data reveals 4,826,245 deleted usernames. You can access this list in `gh-investigator-users_deleted.csv`.

Site Administrators

According to GPT4:

> *A GitHub site administrator, often referred to as a "site admin," is a user with elevated privileges that allow them to manage all aspects of the GitHub instance. This term is generally used within the context of GitHub Enterprise, which is a self-hosted version of GitHub that an organization can run on its own infrastructure.*
> *The site administrator has the highest level of access and can control all the repositories, organizations, and users in the GitHub Enterprise instance. They can set up and modify accounts, handle security settings, and monitor system health among other tasks.*

In GitHub.com (*the public version of GitHub*), this level of control is held by GitHub's own staff and not by individual users or organizations.

This suggests that the 16 users listed in `gh-investigator-users_site_admin.csv`, namely `sentinel`, `GreCodes`, `dvelton`, `saquib-alam`, `accessibility-bot`, `docs-bot`, `iowannie`, `willf`, `NickLiffen`, `ShorukElHadad`, `Ellemmenno`, `davecheney`, `mcantu`, `JeffOgah`, `hubot`, `education-web-bot`, are highly privileged users.

Interesting observations include:

- `GreCodes` is open for employment opportunities.
- `NickLiffen` has made his email address public: `nickliffen@github.com`.
- `hubot` appears to be a bot user.
- `accessibility-bot`, `ShorukElHadad`, `Ellemmenno`, `education-web-bot` have not contributed to any repositories.

It has been confirmed that there are more GitHub staff members with `site_admin` set to True who do not appear in this list. This discrepancy may be due to the rate at which we queried the GitHub API for this data.

Hireable Users

These users are actively seeking employment opportunities. Such information could be beneficial to recruiters or potentially exploited by malicious social engineers who might take advantage of these users' job-seeking status.

We found 1,273,018 hireable users in the data. You can access the list in `gh-investigator-users_hireable.csv`.

Highly Collaborative Users

To maintain manageable CSV file sizes, we limited the number of repositories in which a user has collaborated to 500. We made this decision based on our discovery that certain bot users have contributed to thousands of repositories.

We identified 814 highly collaborative users (possibly bots) in the data. Here are 20 of them: `conda-forge-linter`, `conda-forge-curator[bot]`, `SimonCropp`, `SimenB`, `jjhelmus`, `fire`, `lineageos-gerrit`, `lindseyberlin`, `sharelatex-ci`, `sharelatex-github-sync-acceptance-tests`, `hotman663`, `graingert`, `grandroyalcasino`, `theapplegates`, `0xflotus`, `ThomaCheatham`, `fossabot`, `dalskar`, `SooluThomas`, `olamy`.

Notably, some of these users are also available for hire and even have public emails: `SimonCropp`, `0xflotus`, `olamy`, `echarles`, `tpgxyz`, `schollz`, `tkelman`, `szepeviktor`, `danielbachhuber`, `wizardforcel`, `ademaro`, `Klozz`, `hrbrmstr`...

Users with a Public Email or Company Configured

These are users who have made their email or company information public. This data might be of interest to social engineers looking to target a specific company.

We found 2,651,140 users with a configured email in the file `gh-investigator-users_email.csv`, **3,076,228 users with a configured company in the file** `gh-investigator-users_company.csv`, **and 912,592 users who have configured both**.

Open Source Intelligence (OSINT) on Companies

Using the collected data, it's possible to pinpoint users who are presently or were formerly associated with a specific company.

For instance, running the command:

```
cat all_user_info.csv | grep -i trickest | wc -l
```

allows us to identify all users linked to the company *Trickest* (31 users in this case). However, we can refine these results to minimize false positives:

- By seeking users with the term *Trickest* in their email or company details, we can reduce the number to 8: *gligaTrickest*, *popovicnenad*, *mhmdiaa*, *76creates*, *trickest-workflows*, *patman1970*, *nenadzaric*, *Banegaaa*.

- By searching for users who have contributed to Trickest company repositories using a simple command like `cat all_user_info.csv | grep -i "trickest/"`, we can identify the following users: *popovicnenad*, *c3l3si4n*, *PolovinaD*, *mhmdiaa*, *mihailotomic*, *76creates*, *trickest-workflows*, *nenadzaric*, *kljunowsky*.

GitHub Repository Analysis

We gathered the subsequent details for each repository:

- Owner/Repository
- Number of stars the repository has
- Number of forks the repository has
- Number of watchers of the repository
- Repository deletion status
- Whether the repository is private
- Whether the repository is archived
- Whether the repository is disabled

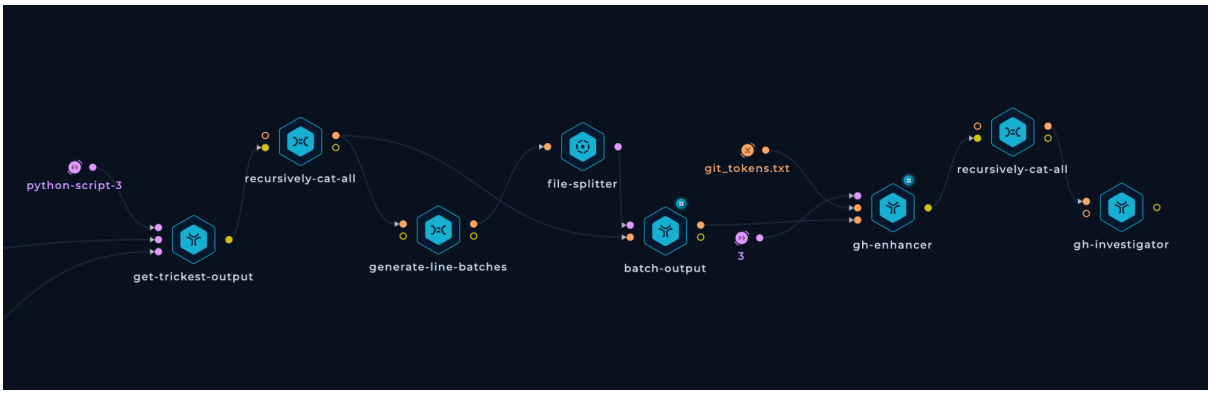
While the GitHub API can provide an array of additional data, we opted to concentrate on these points of interest to keep the CSV files manageable in size.

Our tool, gh-investigator, generated the following files:

- `gh-investigator-repos_sorted_stars.csv`: Repositories (with more than 100 stars) sorted by the number of stars.
- `gh-investigator-repos_sorted_forks.csv`: Repositories (with more than 50 forks) sorted by the number of forks.
- `gh-investigator-repos_sorted_watchers.csv`: Repositories (with more than 20 watchers) sorted by the number of watchers.
- `gh-investigator-repos_deleted.csv`: List of all deleted repositories.
- `gh-investigator-repos_private.csv`: List of all private repositories.
- `gh-investigator-repos_archived.csv`: List of all archived repositories.
- `gh-investigator-repos_disabled.csv`: List of all disabled repositories.

You can find the download links for these files at the conclusion of this report.

Here is the Trickest workflow we employed to obtain details about each identified repository:



Duplicate Repository Handling

The same repository might appear under different names due to reasons such as repository renaming or differences in case sensitivity between the secret, the user, and the generated CSV. After running a Python script to remove such duplicate repositories, the repository count was pared down from approximately 222 million to slightly over 220 million.

Moreover, among these, almost half a million repositories have been deleted, while approximately a million and a half have been archived. This indicates that roughly 218 million repositories remain active currently.

Discovery of Leaked Secrets

It is a well-known issue that secrets (passwords, API keys, etc.) can be inadvertently leaked on GitHub. However, it was surprising to find that secrets had been included in unexpected places like repository names, such as this example: https://github.com/IHATELAGHong/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. There were also instances where secrets were found in company info like: `yasir-javed-58,yasir-javed-58/app1,0,0,0,,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,0`.

In total, **we uncovered 50 working GitHub tokens and 4 consumed OpenAI API Keys**.

When it comes to uncovering leaks in unexpected places, we have room to broaden our search parameters. For example, we could modify the Trickest workflows we ran to include issues and repository comments, thereby enabling a more extensive search for possible leaks.

Top Starred Repositories & Users

The 10 repositories with the highest star counts (at the time of our research) are:

Ranking 🏆	GitRepo	Stars ⭐
1	freecodecamp/freecodecamp	368,107
2	ebookfoundation/free-programming-books	281,193
3	996icu/996.icu	265,975
4	jwasham/coding-interview-university	258,516
5	sindresorhus/awesome	256,262
6	public-apis/public-apis	241,602
7	kamranahmedse/developer-roadmap	240,843
8	donnemartin/system-design-primer	221,307
9	facebook/react	208,321
10	vuejs/vue	203,866

Tools such as PEASS-ng and Hacktricks are in positions 3080 and 7080 respectively.

The top 10 users with the most stars across repositories (at the time of our research) are:

Ranking 🏆	GitRepo	Stars ⭐
1	microsoft	2,064,346
2	google	1,629,138
3	facebook	1,036,020
4	apache	893,299
5	sindresorhus	787,971
6	alibaba	768,376
7	vuejs	629,758

8	facebookresearch	534,802
9	airbnb	511,260
10	github	507,995

Most Forked Repositories & Users

The top 10 repositories with the most forks (at the time of our research) are:

Ranking 🏆	GitRepo	Forks 🔱
1	jtleek/datasharing	242,064
2	rdpeng/programmingassignment2	141,642
3	octocat/spoon-knife	133,886
4	smartthingscommunity/smartthingspublic	89,134
5	github/gitignore	80,632
6	pierian-data/complete-python-3-bootcamp	78,312
7	twbs/bootstrap	75,833
8	tensorflow/tensorflow	71,431
9	nightscout/cgm-remote-monitor	68,563
10	jwasham/coding-interview-university	62,440

The top 10 users with the most forks across repositories (at the time of our research) are:

Ranking 🏆	GitRepo	Forks 🔱
1	learn-co-students	2,379,786
2	learn-co-curriculum	1,266,966
3	lambdaschool	556,978
4	microsoft	438,847

5	apache	403,773
6	bloominstituteoftechnology	373,029
7	google	292,006
8	jtleek	245,331
9	rdpeng	243,450
10	mate-academy	225,070

Most Watched Repositories & Users

The top 10 repositories with the most watchers (at the time of our research) are:

Ranking 🏆	GitRepo	👀 Watchers
1	vhf/free-programming-books	9,674
2	ebookfoundation/free-programming-books	9,673
3	jwasham/coding-interview-university	8,602
4	freecodecamp/freecodecamp	8,440
5	torvalds/linux	8,167
6	tensorflow/tensorflow	7,710
7	sindresorhus/awesome	7,512
8	kamranahmedse/developer-roadmap	6,858
9	twbs/bootstrap	6,811
10	codehubapp/codehub	6,662

The top 10 users with the most watchers across repositories (at the time of research) are:

Ranking 🏆	GitRepo	👀 Watchers
1	learn-co-students	5,507,591
2	devexpress-examples	520,709
3	openmandrivaassociation	306,367
4	learn-co-curriculum	229,677
5	microsoft	216,571
6	textcreationpartnership	215,542
7	gitenberg	181,827
8	jenkinsci	135,572
9	conda-forge	130,212
10	uber	128,394

Conclusion

The sheer volume of accessible information pertaining to GitHub users and repositories remains astonishing. Such data can be harnessed for a myriad of purposes, both benevolent and malicious. However, the most critical takeaway from our exploration should be the heightened awareness of the transparency of our digital footprints. As individuals and organizations, it's crucial to recognize that our information is readily available and can be scraped for sensitive content. Vigilance and proactive measures in protecting our data should be prioritized in our increasingly digitized world.

If you want to **build workflows similar to these**, or use pre-built workflows for Attack Surface Management, Vulnerability Scanning, Threat Intelligence, Content Discovery or other common use cases, **fill out the form on our website** to get access to Trickest.

You can access all the research data directly in our GitHub repository.