



Keychecker

SSH Key based attack tool for DVCS Systems

By

Anant Shrivastava



DVCS Systems

Distributed Version Control System

- Github
- Gitlab
- Bitbucket
- Codeberg
- Gitea
- Forjego
- Huggingface



Access Methods

- Username and Password
 - Deprecated or discouraged
- API Keys
 - Control Plane Access
- SSH Keys
 - Data Plane Access

Ref : <https://cyfinoid.com/decouple-and-conquer-understanding-control-and-data-planes-across-technology/>

You found a key



API



github.com/cyfinoid/ghnavigator

SSH Key



github.com/cyfinoid/keychecker



SSH Key username enumeration

- Everyone knows that if you connect to ssh port of github via your valid ssh key you get your username

```
→ keychecker git:(main) ✗ ssh -i ./test_keys/github_anantshri git@github.com
PTY allocation request failed on channel 0
Hi anantshri! You've successfully authenticated, but GitHub does not provide shell access.
Connection to github.com closed.
```

What else



SSH Enumeration Continues

Step 1: Setup config file for SSH

```
Host enumeration
  Hostname github.com
  IdentifyFile <keyfile>
```

Step 2: Obtain or Generate List of Project Files

Step 3: Identify Different Orgs. Listed in Target User's GitHub Profile

```
curl https://github.com/<username> | rg 'avatar-group-item'
curl https://api.github.org/users/{username}/orgs
```

Step 4: Brute force The Repositories using a wordlist

```
git ls-remote git@enumeration:/<name_or_org>/<repo_projectname>.git
```

```
Repository Discovery Results:
Total attempts: 6
Accessible repositories: 1
Failed attempts: 5
```

```
👤 User: anantshri
📁 dotfiles
```

DEMO

```
→ keychecker git:(main) ✕ uv run keychecker
```

```
usage: keychecker [-h] [-i INPUT]
                  [--validate [{github,gitlab,bitbucket,codeberg,gitea,huggingface,dataops,assembla,boltic,sourcehut,notabug,azuredevops,framagit,gitverse,launchpad,gitee,coding,codeup,gitflic} ...]]
                  [--no-validate] [--discovery DISCOVERY] [--github-token GITHUB_TOKEN] [--no-progress] [--public-out PUBLIC_OUT]
                  [--no-banner] [--timeout TIMEOUT]
                  [--concurrency CONCURRENCY] [-v] [-V]
                  [key_file]
```

SSH private key fingerprinting and Git hosting repository discovery tool

positional arguments:

key_file Path to private key file

options:

```
-h, --help            show this help message and exit
-i, --input INPUT     Path to private key file (alternative to positional argument)
--validate [{github,gitlab,bitbucket,codeberg,gitea,huggingface,dataops,assembla,boltic,sourcehut,notabug,azuredevops,framagit,gitverse,launchpad,gitee,coding,codeup,gitflic} ...]
                        One or more servers to validate against (default: core providers). Optional/regional
                        providers: gitee(chinese), coding(chinese), codeup(chinese), gitflic(russian). When used with --discover-repos, specifies which server to use for
                        repository discovery.
--no-validate         Skip server validation (only analyze key locally)
--discovery DISCOVERY
                        File with candidate repository names (not usernames). When specified, enables repository discovery mode.
--github-token GITHUB_TOKEN
                        GitHub API token for enhanced organization discovery (or set GITHUB_TOKEN env var)
--no-progress         Disable progress bars during repository discovery
--public-out PUBLIC_OUT
                        Save derived public key to file
--no-banner           Suppress banner output
--timeout TIMEOUT     Per-connection timeout (default: 5s)
--concurrency CONCURRENCY
                        Parallel connections (default: 10)
-v, --verbose         Debug/trace logs
-V, --version         show program's version number and exit
```

Examples:

```
keychecker ~/.ssh/id_ed25519                # Analyze key + validate all servers
keychecker -i ~/.ssh/id_ed25519             # Analyze key + validate all servers
keychecker ~/.ssh/id_ed25519 --no-validate   # Analyze key only
keychecker ~/.ssh/id_rsa --validate github   # Validate against GitHub only
keychecker ~/.ssh/id_rsa --validate github --discovery repo_names.txt
keychecker ~/.ssh/id_rsa --public-out public_key.pub
keychecker --version                         # Show version information
```

KeyChecker: Fingerprinting the Blast Radius of SSH Keys

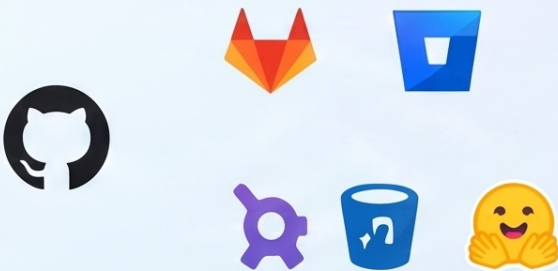
Tool Capabilities & Intelligence

Advanced Key Intelligence



Automatically detects key types, security weaknesses, passphrases, and extracts metadata like local hostnames.

Automated Account Discovery



Validates keys against 19+ platforms including GitHub, GitLab, Bitbucket, and Hugging Face.

Repository Probing



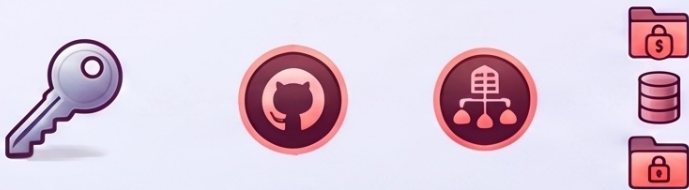
Uses safe, read-only "git ls-remote" probes to discover accessible private repositories without writing data.

Provider Comparison

Provider	Primary Feature	Access Level
 GitHub	Username & Org Discovery	SaaS
 Hugging Face	AI/ML Model Access	SaaS
 GitLab/Bitbucket	Key Confirmation	SaaS & Self-hosted

Impact for Security Teams

Red Team: Demonstrating Impact



Converts "stolen keys" into a clear narrative of the specific repositories and organizations compromised.

Blue Team: Gap Identification

Proactively validates the scope of exposure to build realistic threat models around key compromise.



The Telemetry Blind Spot

Addresses the lack of visibility in standard logs regarding failed or successful SSH attempts.



SSH Attempts (Blind Spot)

Thanks you for listening



- anant@cyfinoid.com
- @anantshri
- <https://github.com/cyfinoid/keychecker/>

