Supply Chain Attacks via GitHub.com Releases

wwws.nightwatchcybersecurity.com/2021/04/25/supply-chain-attacks-via-github-com-releases/

nightwatchcyber April 25, 2021

Summary

Release functionality on **GitHub.com** allows modification of assets within a release by any project collaborator. This can occur after the release is published, and without notification or audit logging accessible in the UI to either the project owners or the public. However, some audit information may be available via the GitHub APIs. An attacker can compromise a collaborator's account and use it to modify releases without the knowledge of project owners or the public, thus resulting in supply chain attacks against the users of the project.

This issue was reported to the vendor – their response is that this is intended behavior and is an intentional design decision. While the vendor is planning improvements in this area, they are not able to provide additional details. GitHub.com paid plans and the GitHub enterprise server were not tested.

As a mitigation measure, project owners using GitHub.com are encouraged to use other methods for securing releases such as digitally signing releases with PGP. Users are encouraged to check digital signatures and use the GitHub.com release APIs to extract and verify release assets data.

Background

GitHub.com is a widely used tool for software development offering source code management (SCM) and other tools. It is used for hosting and distribution by many open source projects (OSS). The release functionality within GitHub.com offers a way to publish packaged software iterations as releases. These include a compressed snapshot of the source within the project as a .ZIP and .TAR.GZ file, as well as as additional binary assets. This functionality is a common way for open source projects to distribute their releases.

Vulnerability Details

The release functionality on **GitHub.com** allows modification of assets within a release by any project collaborator, after the initial release is published. An attacker can use this gap to modify releases without the knowledge of project owners by compromising an account of any project collaborator, thus resulting in supply chain attacks against those using the project. The following specific issues facilitate this:

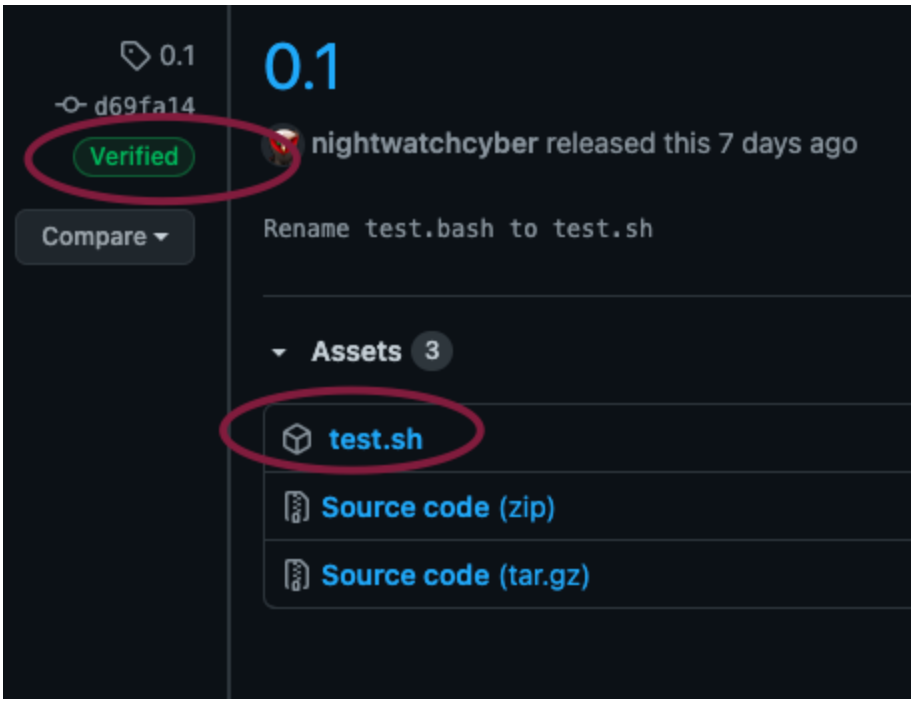
- **Release assets can be modified after initial publication** – except for the source code snapshots

- **Any project collaborator can modify a release** – there are no fine-grained controls to allow code access and not release access.
- **There is no notification or indication within the UI that a release was modified** – to either the project owners or other collaborators, or the public. However, some data is exposed via API.
- **A "verified" flag is displayed if the Git commit was verified** – but this only applies to the source code snapshot and not the other release assets

The releases API provided by GitHub does expose additional information about release assets, which could potentially be used to see if a release was modified. This information includes the username of the uploader and the timestamp when the upload took place. This can be compared to the main release metadata. An example of using APIs for checking releases can be found at our release_auditor project.

NOTE: Paid GitHub.com plans and the GitHub enterprise server were not tested.

Example of a release (see here):



Example of API response exposing asset data:

```
{
  "url": "https://api.github.com/repos/nightwatchcyber/gh_release_test/releases/41605278",
  "assets_url": "https://api.github.com/repos/nightwatchcyber/gh_release_test/releases/41605278/assets",
  "upload_url": "https://uploads.github.com/repos/nightwatchcyber/gh_release_test/releases/41605278/assets{?name,label}",
  "html_url": "https://github.com/nightwatchcyber/gh_release_test/releases/tag/0.1",
  "id": 41605278,
  "author": {
    "login": "nightwatchcyber",
    "id": 16509274,
    "node_id": "MDQ6VXNlcjE2NTA5Mjc0",
    "avatar_url": "https://avatars.githubusercontent.com/u/16509274?v=4",
    "gravatar_id": "",
    "url": "https://api.github.com/users/nightwatchcyber",
    "html_url": "https://github.com/nightwatchcyber",
    "followers_url": "https://api.github.com/users/nightwatchcyber/followers",
    "following_url": "https://api.github.com/users/nightwatchcyber/following{/other_user}",
    "gists_url": "https://api.github.com/users/nightwatchcyber/gists{/gist_id}",
    "starred_url": "https://api.github.com/users/nightwatchcyber/starred{/owner}{/repo}",
    "subscriptions_url": "https://api.github.com/users/nightwatchcyber/subscriptions",
    "organizations_url": "https://api.github.com/users/nightwatchcyber/orgs",
    "repos_url": "https://api.github.com/users/nightwatchcyber/repos",
    "events_url": "https://api.github.com/users/nightwatchcyber/events{/privacy}",
    "received_events_url": "https://api.github.com/users/nightwatchcyber/received_events",
    "type": "User",
    "site_admin": false
  },
  "node_id": "MDc6UmVsZWFzZTQxNjA1Mjc4",
  "tag_name": "0.1",
  "target_commitish": "main",
  "name": "0.1",
  "draft": false,
  "prerelease": false,
  "created_at": "2021-04-18T13:07:33Z",
  "published_at": "2021-04-18T13:08:13Z",
"assets": [
  {
    "url": "https://api.github.com/repos/nightwatchcyber/gh_release_test/releases/assets/35776063",
    "id": 35776063,
    "node_id": "MDEyOlJlbGVhc2VBc3NldDM1Nzc2MDYz",
    "name": "test.sh",
    "label": null,
    "uploader": {
      "login": "yakovsh",
      "id": 667009,
      "node_id": "MDQ6VXNlcjY2NzAwOQ==",
      "avatar_url": "https://avatars.githubusercontent.com/u/667009?v=4",
      "gravatar_id": "",
      "url": "https://api.github.com/users/yakovsh",
      "html_url": "https://github.com/yakovsh",
      "followers_url": "https://api.github.com/users/yakovsh/followers",
      "following_url": "https://api.github.com/users/yakovsh/following{/other_user}",
      "gists_url": "https://api.github.com/users/yakovsh/gists{/gist_id}",
      "starred_url": "https://api.github.com/users/yakovsh/starred{/owner}{/repo}",
      "subscriptions_url": "https://api.github.com/users/yakovsh/subscriptions",
      "organizations_url": "https://api.github.com/users/yakovsh/orgs",
      "repos_url": "https://api.github.com/users/yakovsh/repos",
      "events_url": "https://api.github.com/users/yakovsh/events{/privacy}",
      "received_events_url": "https://api.github.com/users/yakovsh/received_events",
      "type": "User",
      "site_admin": false
    },
    "content_type": "application/x-sh",
    "state": "uploaded",
    "size": 32,
    "download_count": 0,
    "created_at": "2021-04-25T03:11:32Z",
    "updated_at": "2021-04-25T03:11:32Z",
    "browser_download_url": "https://github.com/nightwatchcyber/gh_release_test/releases/download/0.1/test.sh"
  }
```


Steps to Replicate

The following steps can be used to replicate this issue:

1. Alice creates a public repository on GitHub.com, and adds some code including a shell script “test.sh”.
2. Alice invites Bob as a collaborator on this repository.
3. Alice publishes a release including the shell script “test.sh” as a separate asset.

4. Bob accesses the release, and modifies the “test.sh” script within the release.
5. When viewing the release via GitHub.com UI, there is no indication the script was modified. Downloading the script shows that it is different from what Alice published.

NOTE: Paid GitHub.com plans and the GitHub enterprise server were not tested.

Vendor Response and Mitigation

The issue was reported to the vendor via their bounty program. **Their response is that this is intended behavior and is an intentional design decision. While the vendor is planning improvements in this area, they are not able to provide additional details.**

GitHub.com paid plans and the GitHub enterprise server were not tested.

As a mitigation measure, project owners using GitHub.com are encouraged to use other methods for securing releases such as digitally signing releases with PGP. Users are encouraged to check digital signatures and use the GitHub.com release APIs to extract and verify release assets data.

An example of using APIs to check releases can be found in our release_auditor project.

References

Example repository: https://github.com/nightwatchcyber/gh_release_test
GitHub.com docs: here, here and here
HackerOne report # 1167780
release_auditor: see here

Credits

Advisory written by Y. Shafranovich

Timeline

2021-04-18: Initial report submitted to the vendor
2021-04-20: Automated response received
2021-04-21: Vendor response received, intended behavior
2021-04-21: Request to disclose sent
2021-04-23: Vendor ok with disclosure
2021-04-25: Public disclosure – added a link to the OSS project