

# safe.directory

## Symptoms

```
fatal: detected dubious ownership in repository at
```

```
fatal: unsafe repository ('/builds/rPrca3qv/0/group/project' is owned by  
someone else)
```

To add an exception for this directory, call:

```
git config --global --add safe.directory  
/builds/rPrca3qv/0/group/project
```

## Source of problem

Current user is not owner of git repository directory (.git).

- Version 2.30.5 Contains commit: [setup\\_git\\_directory\(\): add an owner check for the top-level directory](#), See release notes: [2.30.5.txt](#)
- More security checks were added with v 2.35.2 [Git security vulnerability announced](#)

## Workaround

Silence all warning (risky!):

```
git config --global --add safe.directory '*'
```

NOTE: \* is not glob pattern. It is only special value which turns off warning for all dirs. ([The command doesn't interpret the wildcard \\* as an operator](#))

[.gitconfig](#)

```
[safe]  
directory=*
```

Silence warning for specified directory:

```
git config --global --add safe.directory /home/john/project
```

NOTE1: Multiple config entries can be added to add more directories

NOTE2: safe.directory points only to one specified directory. It doesn't propagate to subdirectories.

## Workaround using env

Do not use environment **GIT\_CONFIG\_PARAMETERS**. It is only for internal git use, and format is not published.

For modern git (v2.31.0) it is possible to use new env config syntax:

```
GIT_CONFIG_COUNT=1
GIT_CONFIG_KEY_0=safe.directory
GIT_CONFIG_VALUE_0=*
```

See: [GIT\\_CONFIG\\_VALUE\\_](#)

GIT version changelog:

```
2.31.0:
    Two new ways to feed configuration variable-value pairs via
    environment variables have been introduced, and the way
    GIT_CONFIG_PARAMETERS encodes variable/value pairs has been tweaked
    to make it more robust.
```

Related GIT commit:

```
f9dbb64fadf599c588a39d2251bb3f9a2f7d572a 2021-01-12 13:27 +0100 Jeff King
config: parse more robust format in GIT_CONFIG_PARAMETERS
```

## Workaround for WSL

```
git config --global --add safe.directory
'%(prefix)///wsl$/Ubuntu-22.04/home/username/code/my-repo-name'
```

## Workaround for Yocto

Fix in Poky: [bitbake.conf: mark all directories as safe for git to read](#)

This variable can be added to `local.conf`, but it invalidates whole sstate. Simple solution is to fix one recipe:

```
do_compile_prepend() {
    git config --global --add safe.directory ${S}
}
```

## Workaround for Gitlab CI

```
git config --global --add safe.directory ${CI_PROJECT_DIR}
```

*# and if needed, for some submodules*

```
git config --global --add safe.directory ${CI_PROJECT_DIR}/bootloader
```

Workarounds:

- [Configure project clone directory as safe by default](#)
- [Mark project working directory as safe for Git](#)

**Best workaround:** [https://gitlab.com/gitlab-org/gitlab-runner/-/issues/29022#note\\_1356788508](https://gitlab.com/gitlab-org/gitlab-runner/-/issues/29022#note_1356788508)

[config.toml](#)

```
[[runners]]
  environment = ["GIT_CONFIG_COUNT=1",
"GIT_CONFIG_KEY_0=safe.directory", "GIT_CONFIG_VALUE_0=*",
"GIT_CONFIG_PARAMETERS='safe.directory=*'"]
```

or re-register runner with args:

```
gitlab-runner register \
  --env "GIT_CONFIG_COUNT=1" \
  --env "GIT_CONFIG_KEY_0=safe.directory" \
  --env "GIT_CONFIG_VALUE_0=*" \
  --env "GIT_CONFIG_PARAMETERS='safe.directory=*' "
```

**Note:** According to [bitbake.conf](#): mark all directories as safe for git to read :

This can be set globally via the internal environment variable `GIT_CONFIG_PARAMETERS`, we can't use `GIT_CONFIG_*_KEY/VALUE` as that isn't present in all the releases which have the ownership check.

From:

<https://niziak.spox.org/wiki/> - [niziak.spox.org](https://niziak.spox.org)

Permanent link:

<https://niziak.spox.org/wiki/git:issues:safedirectory>

Last update: **2023/07/17 21:17**

