Ansible + Bash Automation Repo

A small automation toolkit that combines Ansible playbooks and Bash scripts to speed up common Linux admin tasks. This repo is built for repeatable execution, quick troubleshooting, and clean documentation of day-to-day work in a RHEL/CentOS-style environment.

What’s Inside

playbooks/

Infrastructure and configuration automation written in Ansible. Use these playbooks to apply consistent changes across multiple hosts, reduce manual steps, and standardize system state.

logs.sh

Utility script for collecting and reviewing logs quickly (useful for troubleshooting services, validating changes, and grabbing outputs for documentation

performance.sh

Basic performance and health checks to capture system utilization and runtime stats (helpful for baseline comparisons and quick diagnostics).

procoreed.sh

A helper script focused on environment-specific admin tasks and repeatable command workflows used in daily operations.

Typical Use Cases

Automate user creation, access changes, and system configuration with Ansible

Run quick shell scripts to validate system state, gather logs, and check performance

Keep a version-controlled record of changes and reusable automation patterns

Quick Start

```
# clone
git clone https://gitlab.com/edwardgarrido3066/ansible_bash_repo.git
cd ansible_bash_repo
# view content
ls -la
To run playbooks (example):
cd playbooks
ansible-playbook <playbook_name>.yml -i <inventory>
```

Notes

Designed for repeatable execution and easy expansion

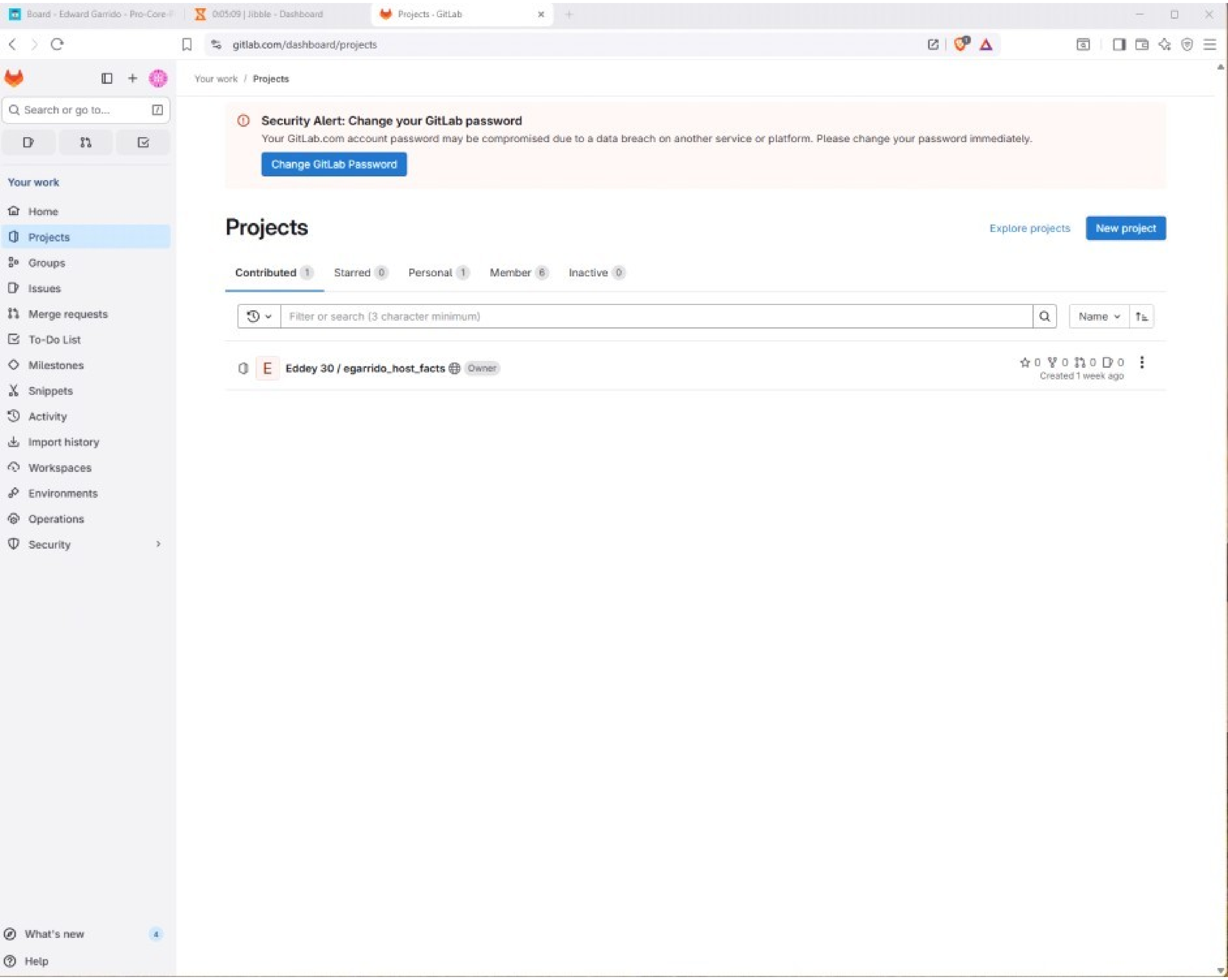
Add new scripts/playbooks as new tasks come up, keeping everything organized and version controlled

The GitLab dashboard is open on the Projects page, showing a security alert banner at the top prompting the user to change their GitLab password due to a potential compromise detected from another service. A blue Change GitLab Password button is visible within the alert.

The left sidebar displays the main navigation, including sections such as Home, Projects, Groups, Issues, Merge requests, To-Do List, Milestones, Snippets, Activity, Import history, Workspaces, Environments, Operations, and Security.

In the main content area, the Projects view is filtered to show contributed projects. One project is listed: egarrido_host_facts, owned by the user Eddy 30, marked as created about one week ago. The project row includes standard GitLab icons for visibility, issues, merge requests, pipelines, and settings, along with a three-dot menu for additional actions.

At the top right of the page, options to Explore projects and create a New project are available, indicating readiness to add or manage repositories.



gitlab.com/dashboard/projects

Security Alert: Change your GitLab password

Your GitLab.com account password may be compromised due to a data breach on another service or platform. Please change your password immediately.

Eddey 30 / egarrido_host_facts Owner

Created 1 week ago

The GitLab interface is open on the New project page with a security alert banner at the top warning that the account password may have been compromised and recommending an immediate password change. A prominent Change GitLab Password button appears within the alert.

The left navigation pane shows the standard GitLab menu, including Home, Projects, Groups, Issues, Merge requests, To-Do List, Milestones, Snippets, Activity, Import history, Workspaces, Environments, Operations, and Security, with Projects highlighted.

In the main content area, the Create new project section presents four large options laid out in a grid:

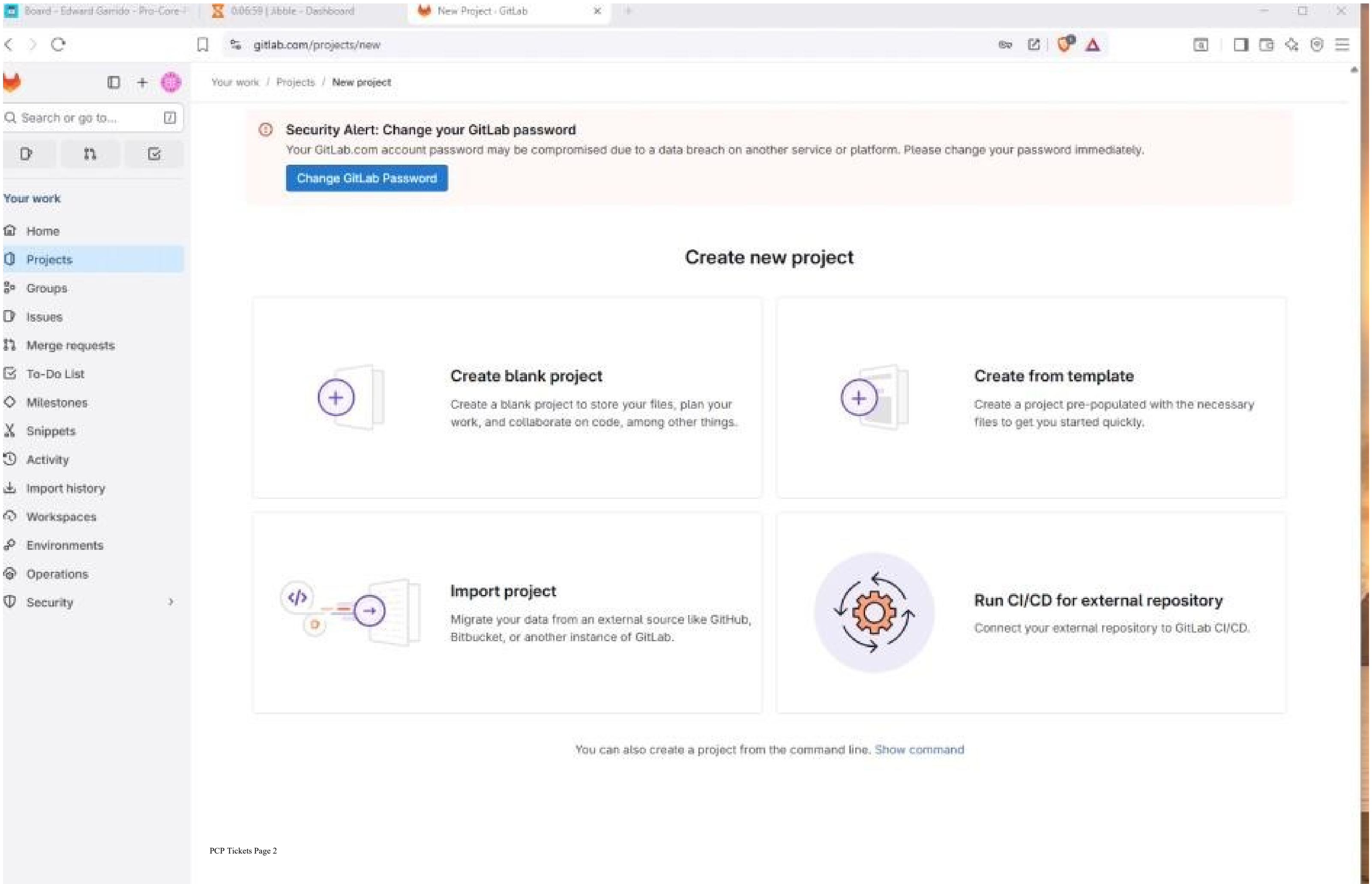
Create a blank project for starting a repository from scratch

Create from template to initialize a project with predefined files

Import project to migrate a repository from another source such as GitHub or Bitbucket

Run CI/CD for external repository to connect an external repository to GitLab CI/CD

Below these options, there is a note indicating that a project can also be created from the command line, with a link to show the relevant command.



Your work / Projects / New project

Security Alert: Change your GitLab password

Your GitLab.com account password may be compromised due to a data breach on another service or platform. Please change your password immediately.

Change GitLab Password

Create new project

Create blank project

Create a blank project to store your files, plan your work, and collaborate on code, among other things.

Create from template

Create a project pre-populated with the necessary files to get you started quickly.

Import project

Migrate your data from an external source like GitHub, Bitbucket, or another instance of GitLab.

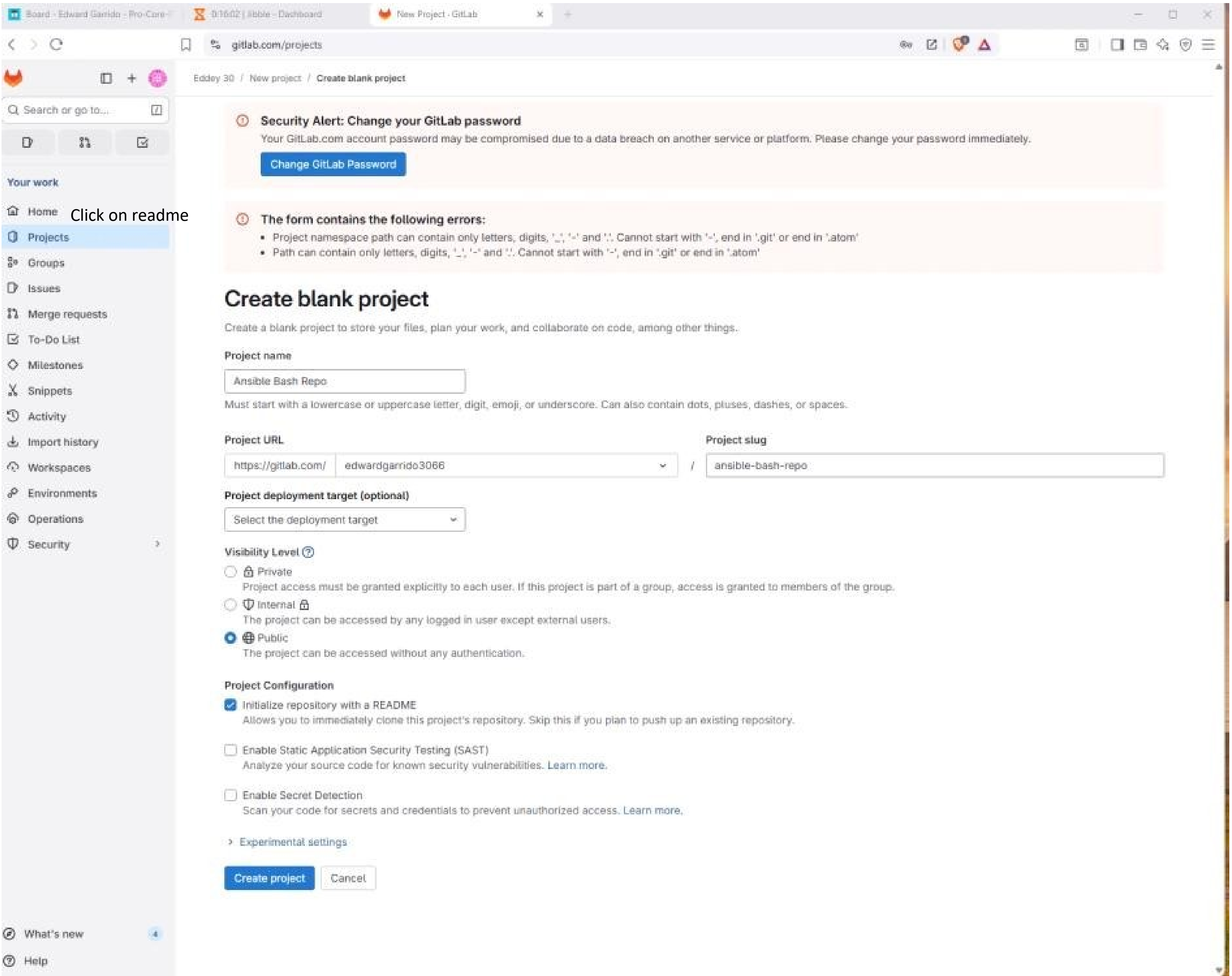
Run CI/CD for external repository

Connect your external repository to GitLab CI/CD.

You can also create a project from the command line. Show command

The Git Lab page shows the Create blank project form with a security alert at the top prompting a password change. The project name is set to Ansible Bash Repo, with the slug ansible-bash-repo under the user namespace edwardgarrido3066.

The project visibility is set to Public, the repository is configured to initialize with a README, and options for security scanning are visible but not enabled. At the bottom, buttons to Create project or Cancel are available.



Click on readme

Security Alert: Change your GitLab password
Your GitLab.com account password may be compromised due to a data breach on another service or platform. Please change your password immediately.

The form contains the following errors:

- Project namespace path can contain only letters, digits, '_', '-' and '.'. Cannot start with '-', end in '.git' or end in '.atom'
- Path can contain only letters, digits, '_', '-' and '.'. Cannot start with '-', end in '.git' or end in '.atom'

Create blank project

Create a blank project to store your files, plan your work, and collaborate on code, among other things.

Project name
Ansible Bash Repo
Must start with a lowercase or uppercase letter, digit, emoji, or underscore. Can also contain dots, pluses, dashes, or spaces.

Project URL
https://gitlab.com/ edwardgarrido3066 /

Project slug
ansible-bash-repo

Project deployment target (optional)
Select the deployment target

Visibility Level

○ Private
Project access must be granted explicitly to each user. If this project is part of a group, access is granted to members of the group.

○ Internal
The project can be accessed by any logged in user except external users.

● Public
The project can be accessed without any authentication.

Project Configuration

☑ Initialize repository with a README
Allows you to immediately clone this project's repository. Skip this if you plan to push up an existing repository.

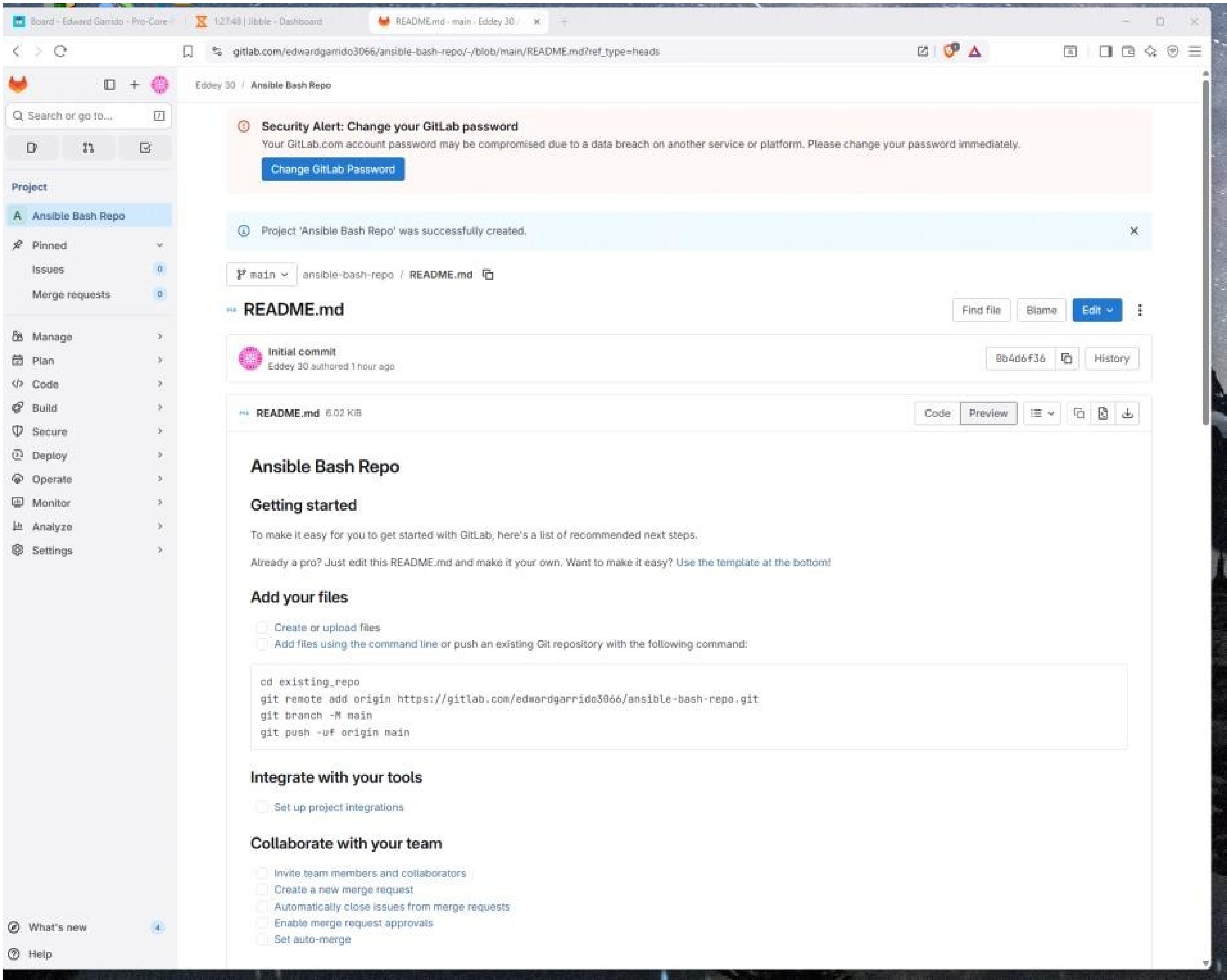
□ Enable Static Application Security Testing (SAST)
Analyze your source code for known security vulnerabilities. Learn more.

□ Enable Secret Detection
Scan your code for secrets and credentials to prevent unauthorized access. Learn more.

› Experimental settings

The GitLab project page for Ansible Bash Repo is open, showing a confirmation banner that the project was successfully created along with a security alert recommending a password change. The main view displays the default README.md file from the initial commit, including sections for getting started, adding files, and basic Git commands.

The left sidebar shows the project navigation with options like Project overview, Issues, Merge requests, CI/CD, and Settings, indicating the repository is ready for further configuration and content updates.



gitlab.com/edwardgarrido3066/ansible-bash-repo/-/blob/main/README.md?ref_type=heads

Security Alert: Change your GitLab password

Your GitLab.com account password may be compromised due to a data breach on another service or platform. Please change your password immediately.

Change GitLab Password

Project 'Ansible Bash Repo' was successfully created.

README.md

Initial commit
Eddey 30 authored 1 hour ago
8b4d6f36

README.md 6.02 KiB

Ansible Bash Repo

Getting started

To make it easy for you to get started with GitLab, here's a list of recommended next steps.

Already a pro? Just edit this README.md and make it your own. Want to make it easy? Use the template at the bottom!

Add your files

- □ Create or upload files
- □ Add files using the command line or push an existing Git repository with the following command:

```
cd existing_repo
git remote add origin https://gitlab.com/edwardgarrido3066/ansible-bash-repo.git
git branch -M main
git push -uf origin main
```

Integrate with your tools

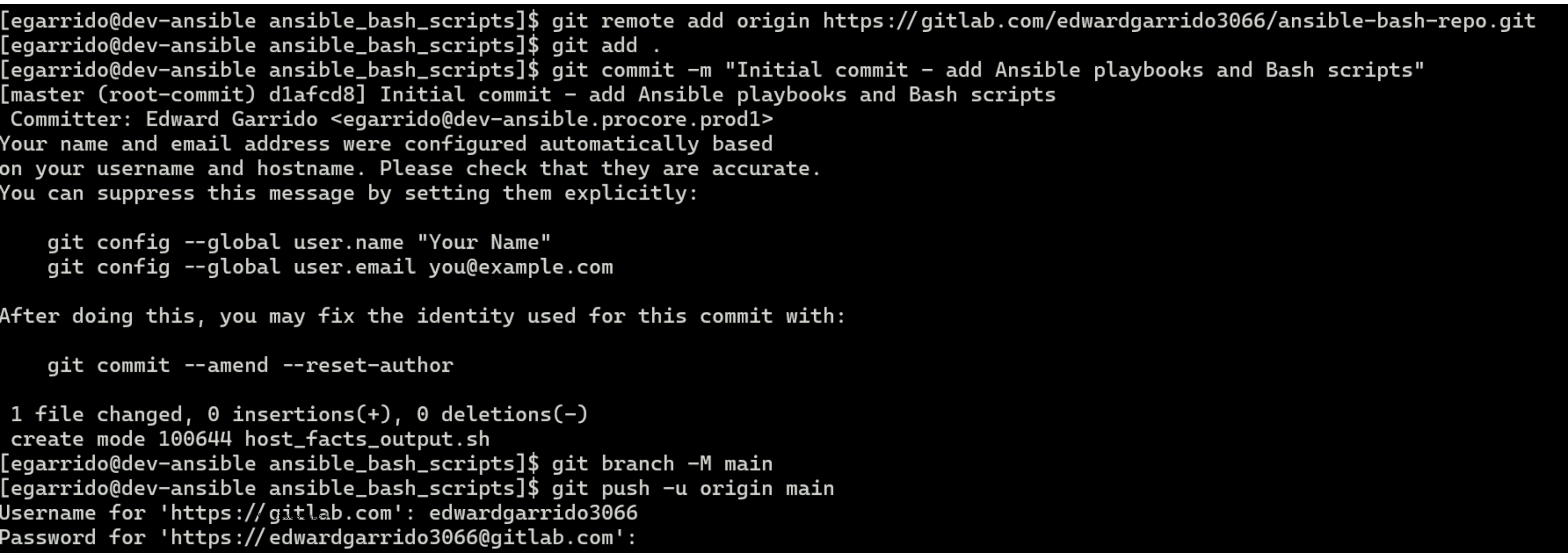
- □ Set up project integrations

Collaborate with your team

- □ Invite team members and collaborators
- □ Create a new merge request
- □ Automatically close issues from merge requests
- □ Enable merge request approvals
- □ Set auto-merge

A terminal session shows a local Git repository being prepared and pushed to GitLab. The remote origin is added using the GitLab HTTPS URL, all files are staged, and an initial commi is created with a message indicating the addition of Ansible playbooks and Bash scripts. Git reports the commit author as Edward Garrido and displays a notice that the user name and email were auto-configured.

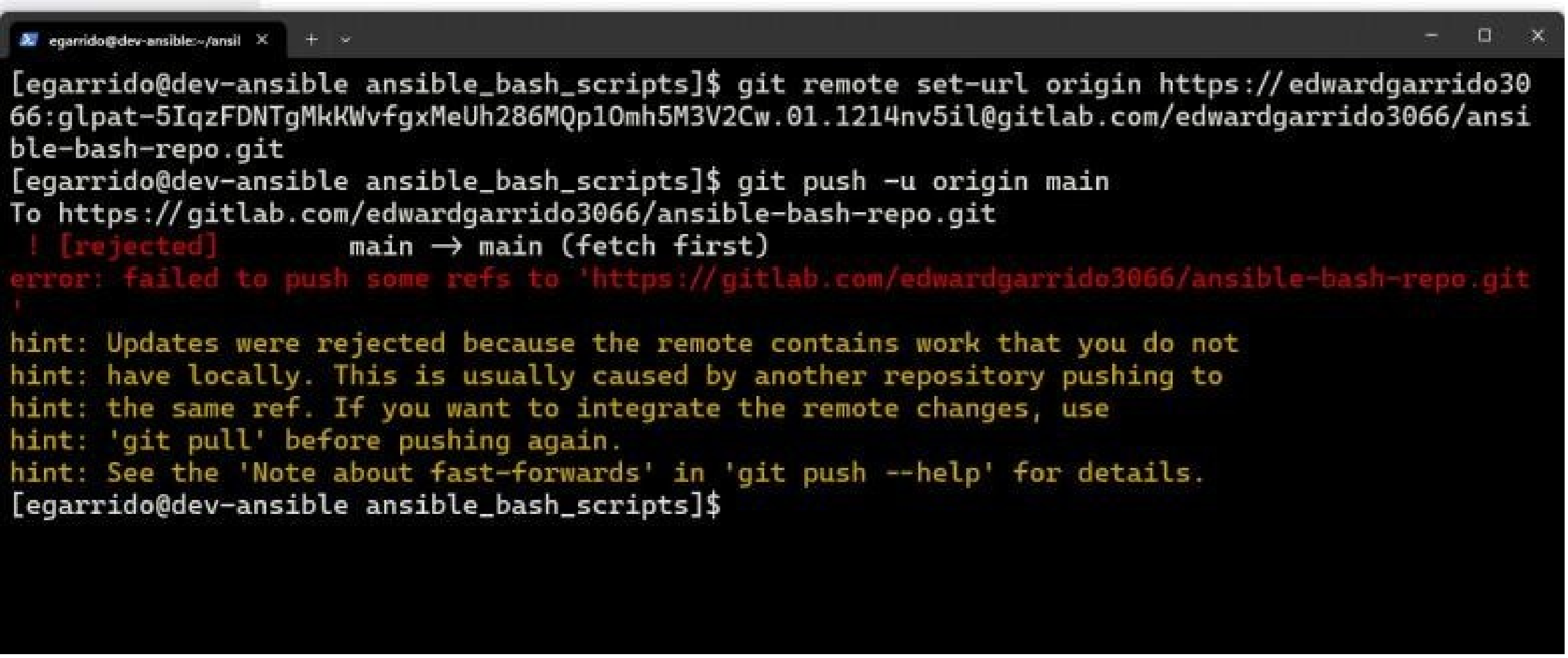
The branch is renamed from master to main, and a push is initiated to the remote repository. During the push, Git prompts for the GitLab username and password to authenticate the HTTPS connection.



```
[egarrido@dev-ansible ansible_bash_scripts]$ git remote add origin https://gitlab.com/edwardgarrido3066/ansible-bash-repo.git
[egarrido@dev-ansible ansible_bash_scripts]$ git add .
[egarrido@dev-ansible ansible_bash_scripts]$ git commit -m "Initial commit - add Ansible playbooks and Bash scripts"
[master (root-commit) d1afcd8] Initial commit - add Ansible playbooks and Bash scripts
 Committer: Edward Garrido <egarrido@dev-ansible.procore.prod1>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly:

    git config --global user.name "Your Name"
    git config --global user.email you@example.com

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 host_facts_output.sh
[egarrido@dev-ansible ansible_bash_scripts]$ git branch -M main
[egarrido@dev-ansible ansible_bash_scripts]$ git push -u origin main
Username for 'https://gitlab.com': edwardgarrido3066
Password for 'https://edwardgarrido3066@gitlab.com':
```

A terminal window shows an attempt to update the Git remote URL to use an HTTPS address with an embedded access token, followed by a push to the main branch. The push is rejected with a non-fast-forward error, indicating that the remote repository already contains commits not present in the local branch.

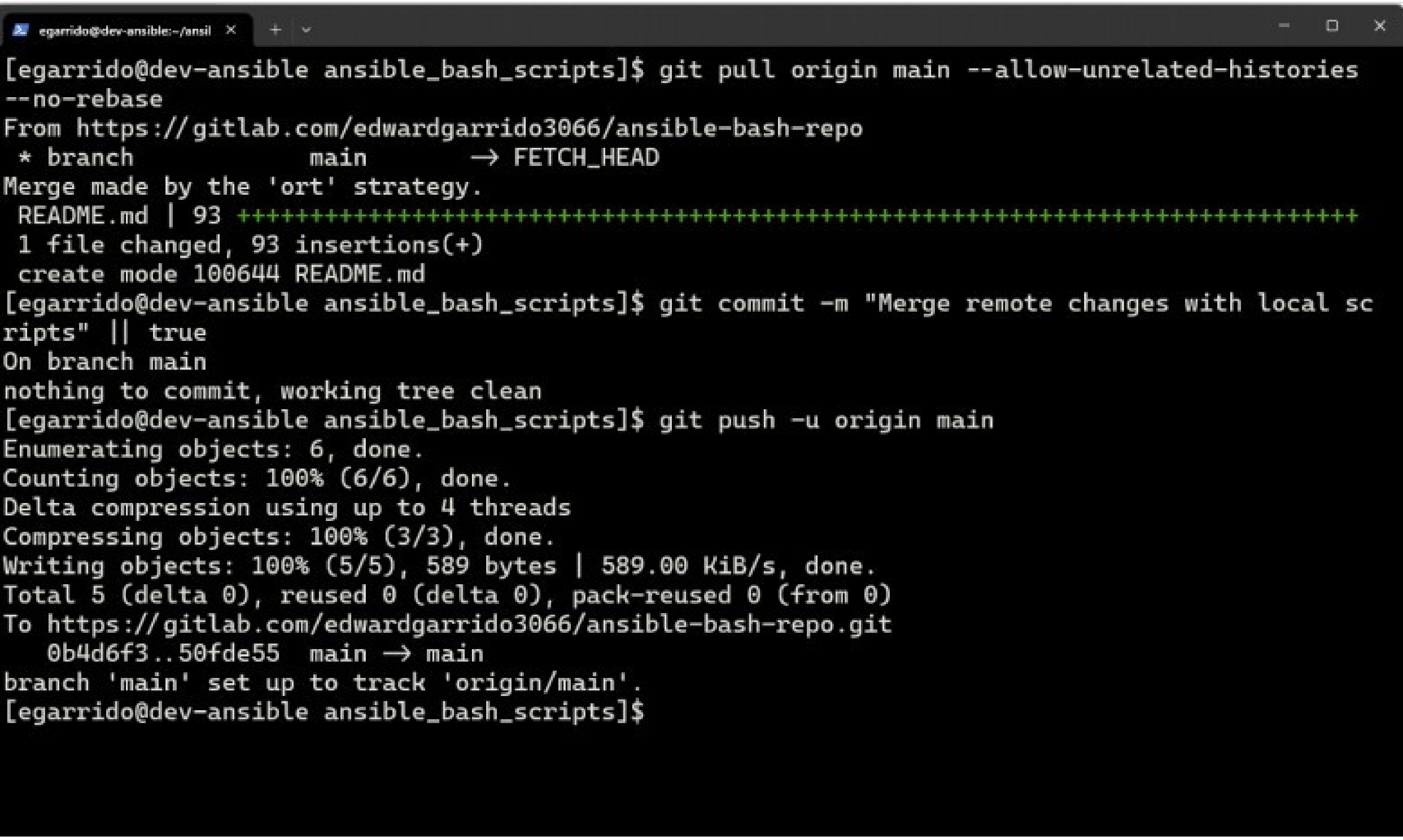
Git displays hints explaining that the rejection occurred because the histories differ and suggests pulling the remote changes first before pushing again. The session ends with the user returned to the shell prompt after the failed push.



```
[egarrido@dev-ansible ansible_bash_scripts]$ git remote set-url origin https://edwardgarrido30
66:glpat-5IqzFDNTgMkKWvfgxMeUh286MQp1Omh5M3V2Cw.01.1214nv5il@gitlab.com/edwardgarrido3066/ansi
ble-bash-repo.git
[egarrido@dev-ansible ansible_bash_scripts]$ git push -u origin main
To https://gitlab.com/edwardgarrido3066/ansible-bash-repo.git
 ! [rejected]        main -> main (fetch first)
error: failed to push some refs to 'https://gitlab.com/edwardgarrido3066/ansible-bash-repo.git
'
hint: Updates were rejected because the remote contains work that you do not
hint: have locally. This is usually caused by another repository pushing to
hint: the same ref. If you want to integrate the remote changes, use
hint: 'git pull' before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for details.
[egarrido@dev-ansible ansible_bash_scripts]$
```

The terminal output shows the local repository being synchronized with the remote GitLab repository after resolving a history mismatch. A pull is executed with the --allow-unrelated-histories option, successfully merging the remote README.md into the local branch using the default merge strategy.

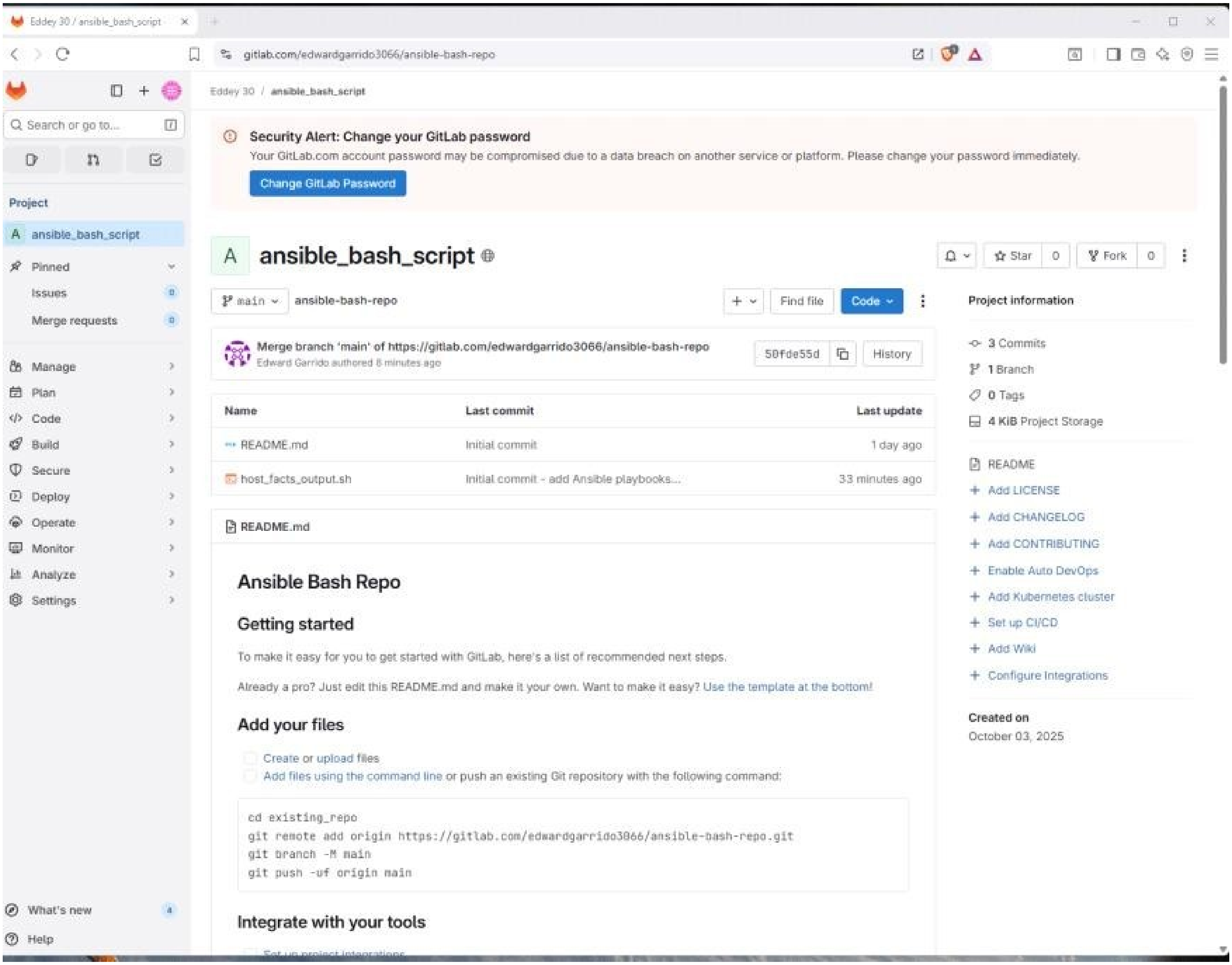
After confirming the working tree is clean, the changes are pushed to the main branch on GitLab. The push completes successfully, and the local main branch is set to track origin/main, indicating the repository is now fully aligned between local and remote.



```
[egarrido@dev-ansible ansible_bash_scripts]$ git pull origin main --allow-unrelated-histories
--no-rebase
From https://gitlab.com/edwardgarrido3066/ansible-bash-repo
 * branch            main       -> FETCH_HEAD
Merge made by the 'ort' strategy.
 README.md | 93 +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 1 file changed, 93 insertions(+)
 create mode 100644 README.md
[egarrido@dev-ansible ansible_bash_scripts]$ git commit -m "Merge remote changes with local sc
ripts" || true
On branch main
nothing to commit, working tree clean
[egarrido@dev-ansible ansible_bash_scripts]$ git push -u origin main
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (5/5), 589 bytes | 589.00 KiB/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://gitlab.com/edwardgarrido3066/ansible-bash-repo.git
   0b4d6f3..50fde55  main -> main
branch 'main' set up to track 'origin/main'.
[egarrido@dev-ansible ansible_bash_scripts]$
```

The GitLab repository page for ansible_bash_script is open, showing a security alert banner recommending a password change at the top. The project is on the main branch and lists two files in the repository: README.md and host_facts_output.sh, both associated with the initial commit and a recent merge into the main branch.

The README content is displayed below the file list, presenting the default Ansible Bash Repo introduction and setup guidance. The right sidebar shows project information such as the number of commits, branches, tags, and storage usage, indicating the repository is active and successfully synchronized with the remote.



Security Alert: Change your GitLab password

Your GitLab.com account password may be compromised due to a data breach on another service or platform. Please change your password immediately.

ansible_bash_script

Merge branch 'main' of https://gitlab.com/edwardgarrido3066/ansible-bash-repo
Edward Garrido authored 8 minutes ago

Name	Last commit	Last update
README.md	Initial commit	1 day ago
host_facts_output.sh	Initial commit - add Ansible playbooks...	33 minutes ago

README.md

Ansible Bash Repo

Getting started

To make it easy for you to get started with GitLab, here's a list of recommended next steps.

Already a pro? Just edit this README.md and make it your own. Want to make it easy? Use the template at the bottom!

Add your files

- □ Create or upload files
- □ Add files using the command line or push an existing Git repository with the following command:

```
cd existing_repo
git remote add origin https://gitlab.com/edwardgarrido3066/ansible-bash-repo.git
git branch -M main
git push -uf origin main
```

Integrate with your tools

- □ Set up project integrations

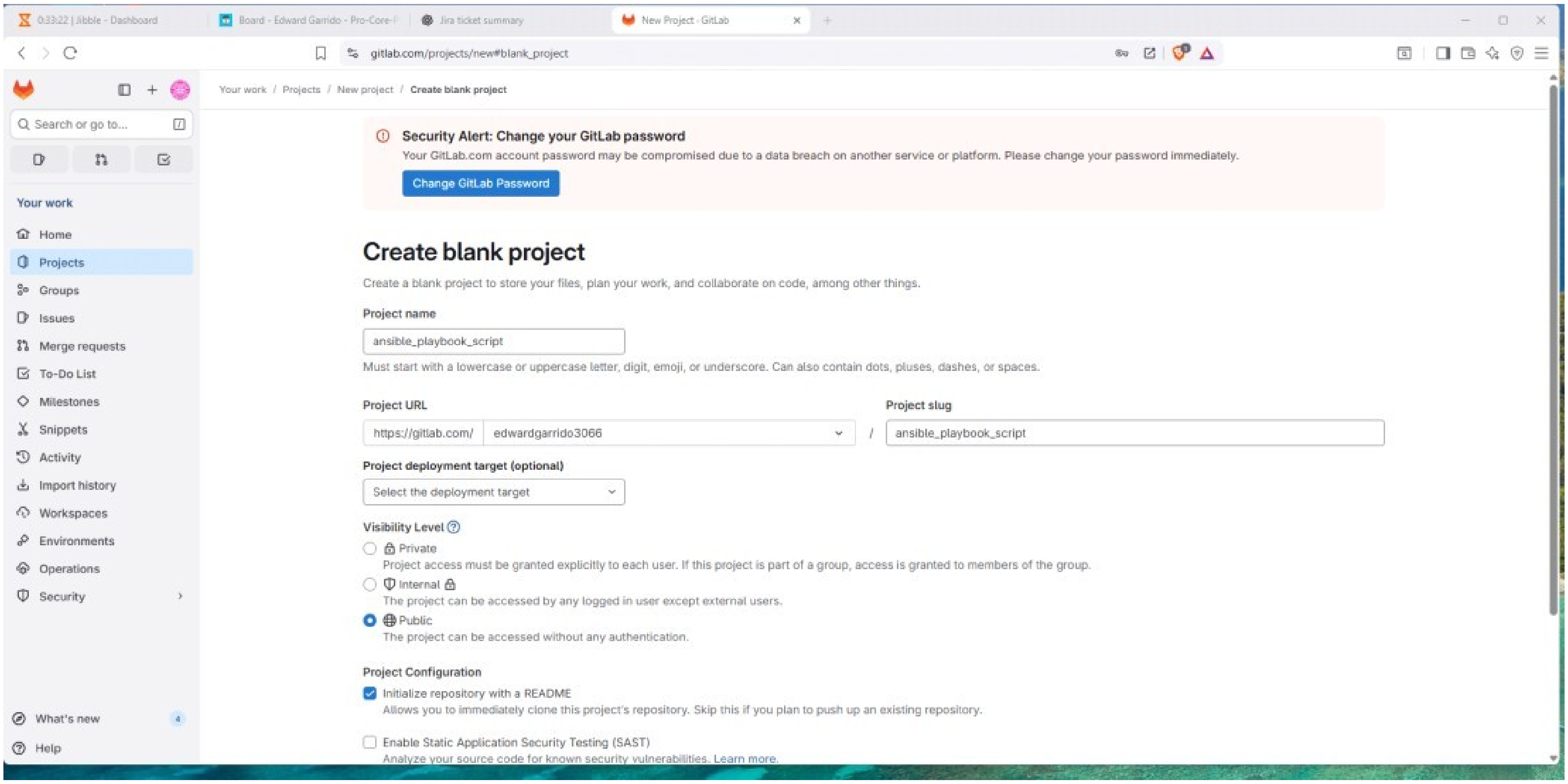
Project information

3 Commits
1 Branch
0 Tags
4 KiB Project Storage

Created on
October 03, 2025

The GitLab Create blank project page is displayed with a security alert banner at the top advising a password change. The project name field is filled in as ansible_playbook_script, with the project slug matching the same name under the edwardgarrido3066 namespace.

The visibility level is set to Public, the option to initialize the repository with a README is selected, and the form shows the standard project configuration options. The page is ready to create the new repository using the Create project button at the bottom.



Security Alert: Change your GitLab password

Your GitLab.com account password may be compromised due to a data breach on another service or platform. Please change your password immediately.

Create blank project

Create a blank project to store your files, plan your work, and collaborate on code, among other things.

Project name

ansible_playbook_script

Must start with a lowercase or uppercase letter, digit, emoji, or underscore. Can also contain dots, pluses, dashes, or spaces.

Project URL

https://gitlab.com/ edwardgarrido3066

Project slug

ansible_playbook_script

Project deployment target (optional)

Select the deployment target

Visibility Level

○ Private
Project access must be granted explicitly to each user. If this project is part of a group, access is granted to members of the group.

○ Internal
The project can be accessed by any logged in user except external users.

● Public
The project can be accessed without any authentication.

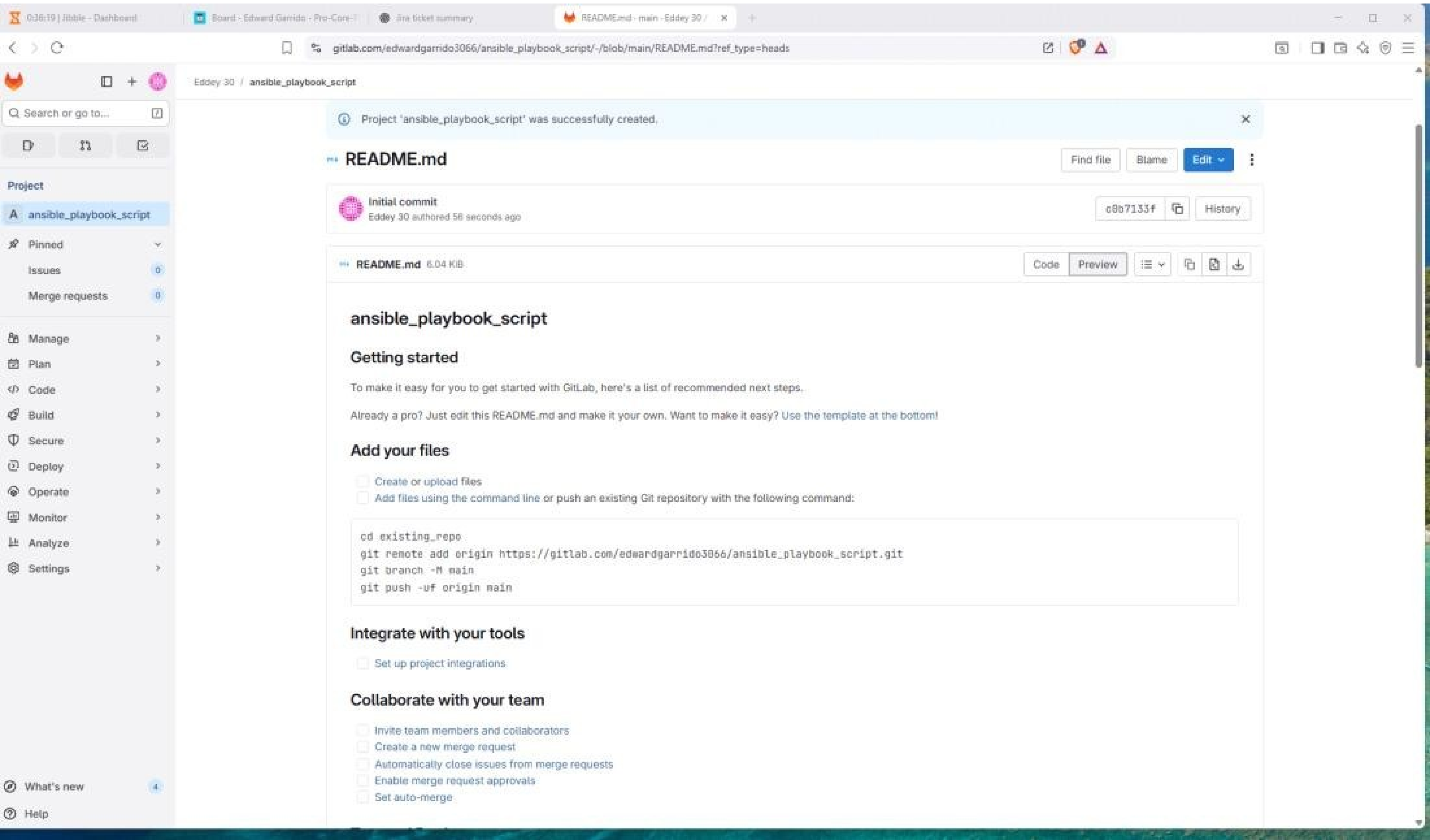
Project Configuration

☑ Initialize repository with a README
Allows you to immediately clone this project's repository. Skip this if you plan to push up an existing repository.

□ Enable Static Application Security Testing (SAST)
Analyze your source code for known security vulnerabilities. Learn more.

The GitLab project page for ansible_playbook_script is open, showing a confirmation banner that the project was successfully created. The repository is on the main branch and displays the default README.md from the initial commit.

The README contains the standard GitLab starter content with sections for getting started, adding files, and basic Git commands. The left sidebar shows the project navigation, indicating the repository is newly created and ready for further development.


0:36:19 | Jibble - Dashboard
Board - Edward Garrido - Pro-Core-
Jira ticket summary
README.md · main · Eddey 30 /
gitlab.com/edwardgarrido3066/ansible_playbook_script/-/blob/main/README.md?ref_type=heads
Search or go to...
Project
ansible_playbook_script
Pinned
Issues 0
Merge requests 0
Manage
Plan
Code
Build
Secure
Deploy
Operate
Monitor
Analyze
Settings
What's new 4
Help
Eddey 30 / ansible_playbook_script
Find file
Blame
Edit
History
Code
Preview

Project 'ansible_playbook_script' was successfully created.

README.md

Initial commit
Eddey 30 authored 56 seconds ago
c8b7133f

README.md 6.04 KiB

ansible_playbook_script

Getting started

To make it easy for you to get started with GitLab, here's a list of recommended next steps.

Already a pro? Just edit this README.md and make it your own. Want to make it easy? Use the template at the bottom!

Add your files

- □ Create or upload files
- □ Add files using the command line or push an existing Git repository with the following command:

```
cd existing_repo
git remote add origin https://gitlab.com/edwardgarrido3066/ansible_playbook_script.git
git branch -M main
git push -uf origin main
```

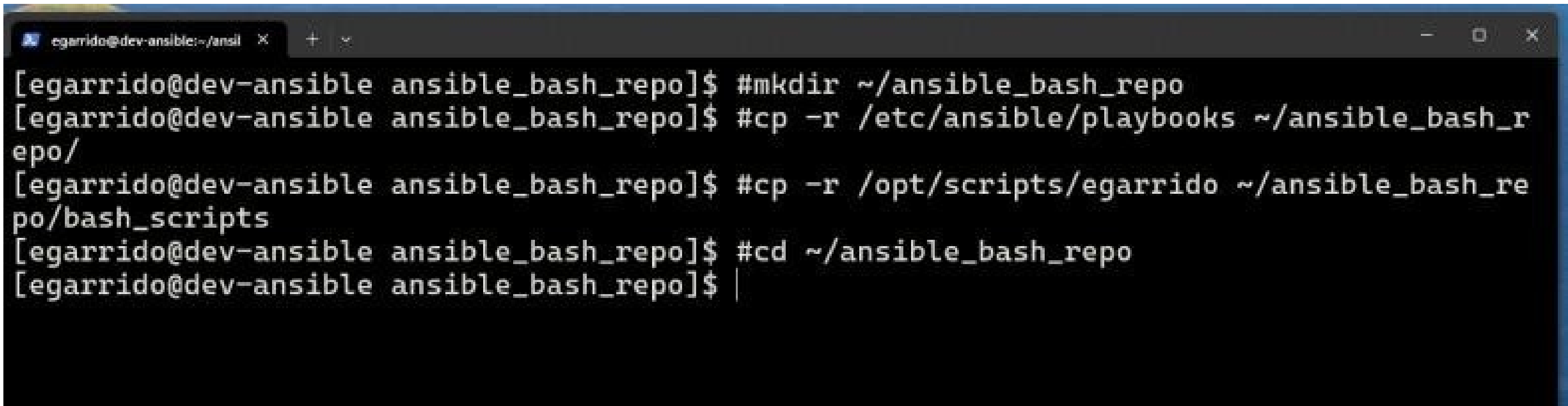
Integrate with your tools

- □ Set up project integrations

Collaborate with your team

- □ Invite team members and collaborators
- □ Create a new merge request
- □ Automatically close issues from merge requests
- □ Enable merge request approvals
- □ Set auto-merge

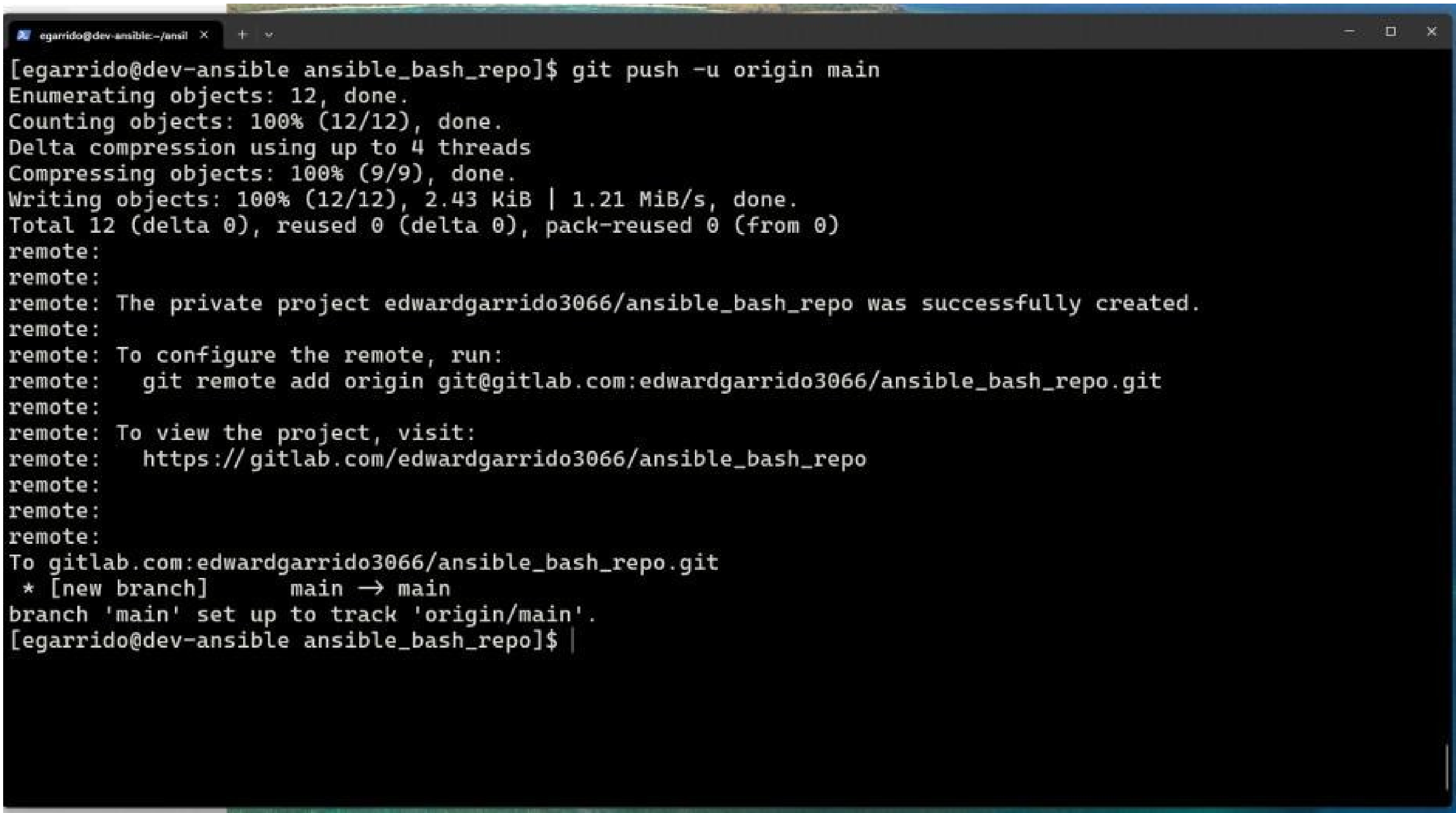
A terminal session shows a directory being prepared for a new repository. A folder named ansible_bash_repo is created in the home directory, Ansible playbooks are copied from /etc/ansible/playbooks, and Bash scripts are copied from /opt/scripts/egarrido into a bash_scripts subdirectory. The session ends with the user changing into the newly created ansible_bash_repo directory.



```
[egarrido@dev-ansible ansible_bash_repo]$ #mkdir ~/ansible_bash_repo
[egarrido@dev-ansible ansible_bash_repo]$ #cp -r /etc/ansible/playbooks ~/ansible_bash_r
epo/
[egarrido@dev-ansible ansible_bash_repo]$ #cp -r /opt/scripts/egarrido ~/ansible_bash_re
po/bash_scripts
[egarrido@dev-ansible ansible_bash_repo]$ #cd ~/ansible_bash_repo
[egarrido@dev-ansible ansible_bash_repo]$ |
```

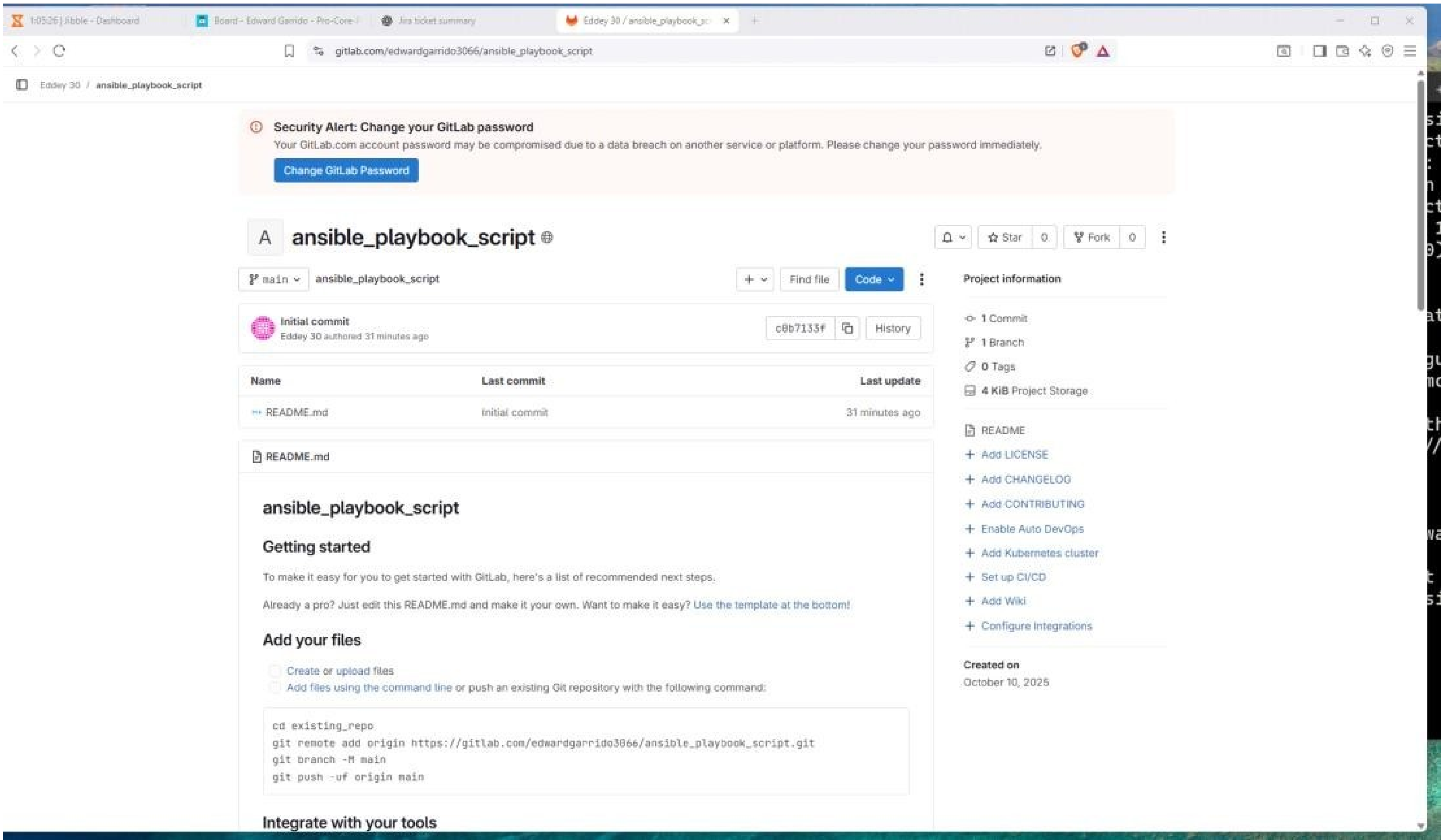
The terminal output shows a successful push of the local ansible_bash_repo repository to GitLab. All objects are enumerated, compressed, and written without errors, and the remote repository is automatically created under the user namespace edwardgarrido3066.

The main branch is pushed to the remote, set to track origin/main, and confirmation messages indicate that the project was created successfully and is now accessible via the provided GitLab URL.



```
[egarrido@dev-ansible ansible_bash_repo]$ git push -u origin main
Enumerating objects: 12, done.
Counting objects: 100% (12/12), done.
Delta compression using up to 4 threads
Compressing objects: 100% (9/9), done.
Writing objects: 100% (12/12), 2.43 KiB | 1.21 MiB/s, done.
Total 12 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote:
remote: The private project edwardgarrido3066/ansible_bash_repo was successfully created.
remote:
remote: To configure the remote, run:
remote:   git remote add origin git@gitlab.com:edwardgarrido3066/ansible_bash_repo.git
remote:
remote: To view the project, visit:
remote:   https://gitlab.com/edwardgarrido3066/ansible_bash_repo
remote:
remote:
remote:
To gitlab.com:edwardgarrido3066/ansible_bash_repo.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
[egarrido@dev-ansible ansible_bash_repo]$
```

The GitLab repository page for ansible_playbook_script is open on the main branch. A security alert banner at the top recommends changing the account password. The repository currently contains only the default README.md from the initial commit, with project details shown in the right sidebar indicating one commit, one branch, and minimal storage usage.


1:05:26 | Jibble - Dashboard
Board - Edward Garrido - Pro-Core-
Jira ticket summary
Eddey 30 / ansible_playbook_sc
gitlab.com/edwardgarrido3066/ansible_playbook_script
Eddey 30 / ansible_playbook_script
Change GitLab Password
Star 0
Fork 0
main
Find file
Code
History

Security Alert: Change your GitLab password
Your GitLab.com account password may be compromised due to a data breach on another service or platform. Please change your password immediately.

ansible_playbook_script

main ansible_playbook_script

Initial commit
Eddey 30 authored 31 minutes ago
c0b7133f

Name	Last commit	Last update
README.md	Initial commit	31 minutes ago

README.md

ansible_playbook_script

Getting started

To make it easy for you to get started with GitLab, here's a list of recommended next steps.

Already a pro? Just edit this README.md and make it your own. Want to make it easy? Use the template at the bottom!

Add your files

- ○ Create or upload files
- ○ Add files using the command line or push an existing Git repository with the following command:

```
cd existing_repo
git remote add origin https://gitlab.com/edwardgarrido3066/ansible_playbook_script.git
git branch -M main
git push -uf origin main
```

Integrate with your tools

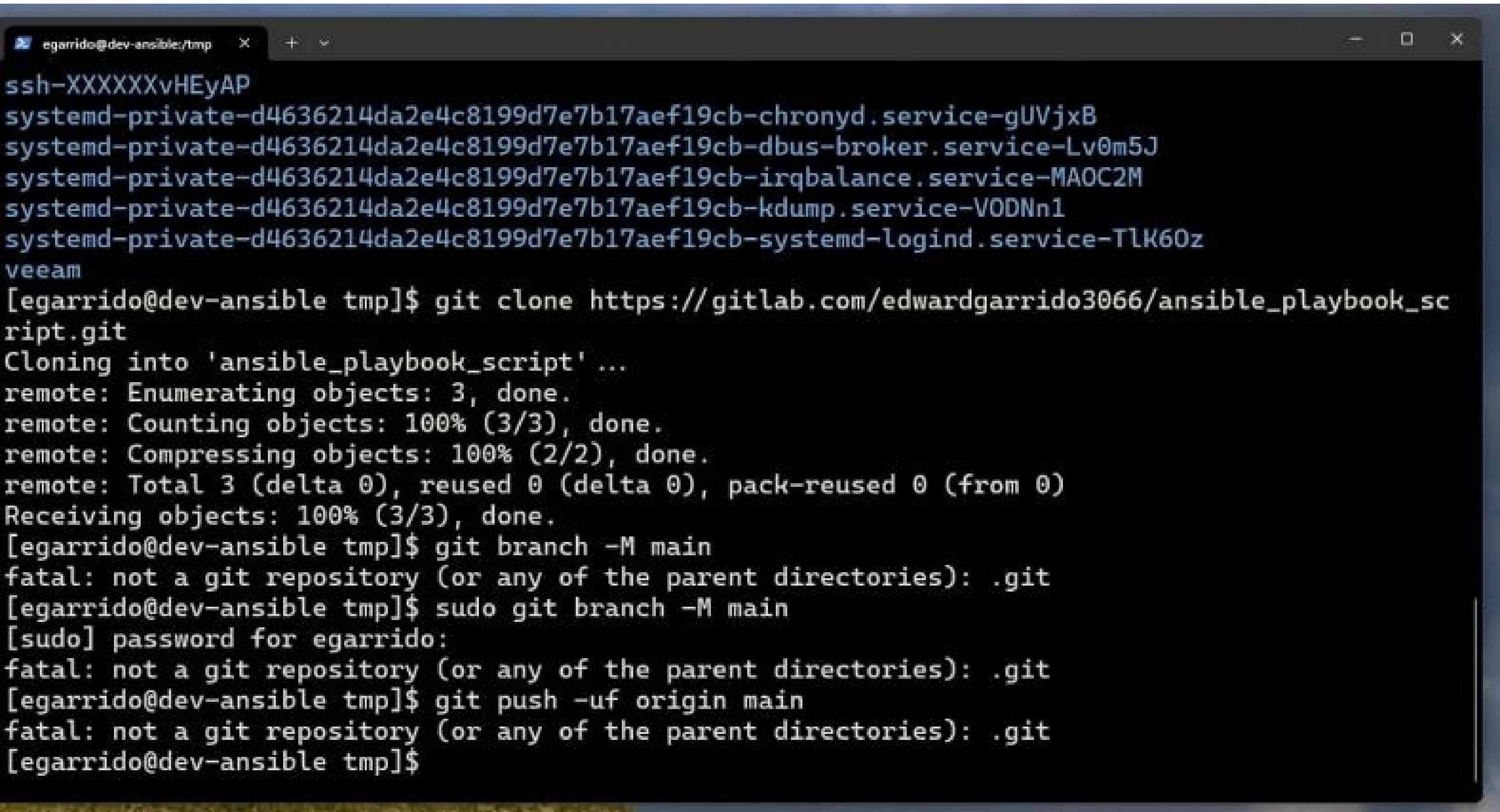
Project information

1 Commit
1 Branch
0 Tags
4 KiB Project Storage

README
+ Add LICENSE
+ Add CHANGELOG
+ Add CONTRIBUTING
+ Enable Auto DevOps
+ Add Kubernetes cluster
+ Set up CI/CD
+ Add Wiki
+ Configure Integrations

Created on
October 10, 2025

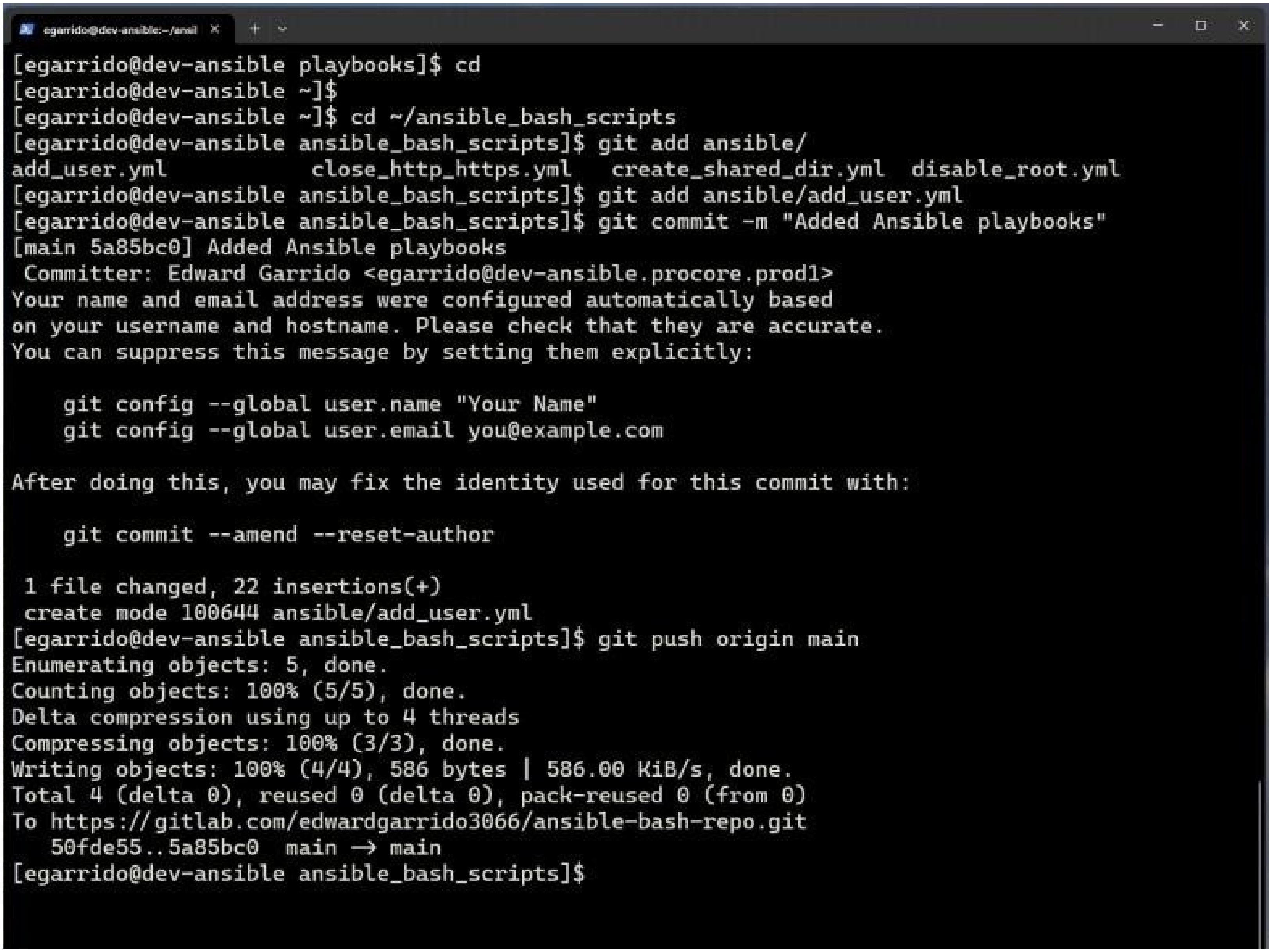
A terminal session shows a GitLab repository being cloned into a directory named ansible_playbook_script, with the clone completing successfully. After the clone, several Git commands are run from the parent /tmp directory rather than inside the repository directory, resulting in repeated errors stating that the location is not a Git repository. The session ends at the shell prompt after the failed Git operations.



```
ssh-XXXXXXvHEyAP
systemd-private-d4636214da2e4c8199d7e7b17aef19cb-chronyd.service-gUVjxB
systemd-private-d4636214da2e4c8199d7e7b17aef19cb-dbus-broker.service-Lv0m5J
systemd-private-d4636214da2e4c8199d7e7b17aef19cb-irqbalance.service-MAOC2M
systemd-private-d4636214da2e4c8199d7e7b17aef19cb-kdump.service-VODNn1
systemd-private-d4636214da2e4c8199d7e7b17aef19cb-systemd-logind.service-TlK6Oz
veeam
[egarrido@dev-ansible tmp]$ git clone https://gitlab.com/edwardgarrido3066/ansible_playbook_sc
ript.git
Cloning into 'ansible_playbook_script'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (3/3), done.
[egarrido@dev-ansible tmp]$ git branch -M main
fatal: not a git repository (or any of the parent directories): .git
[egarrido@dev-ansible tmp]$ sudo git branch -M main
[sudo] password for egarrido:
fatal: not a git repository (or any of the parent directories): .git
[egarrido@dev-ansible tmp]$ git push -uf origin main
fatal: not a git repository (or any of the parent directories): .git
[egarrido@dev-ansible tmp]$
```

A terminal session shows navigation into the ansible_bash_scripts directory, where Ansible playbook files are staged and committed to the local Git repository. The commit message indicates Ansible playbooks were added, and Git displays the author information along with a notice about auto-configured user identity.

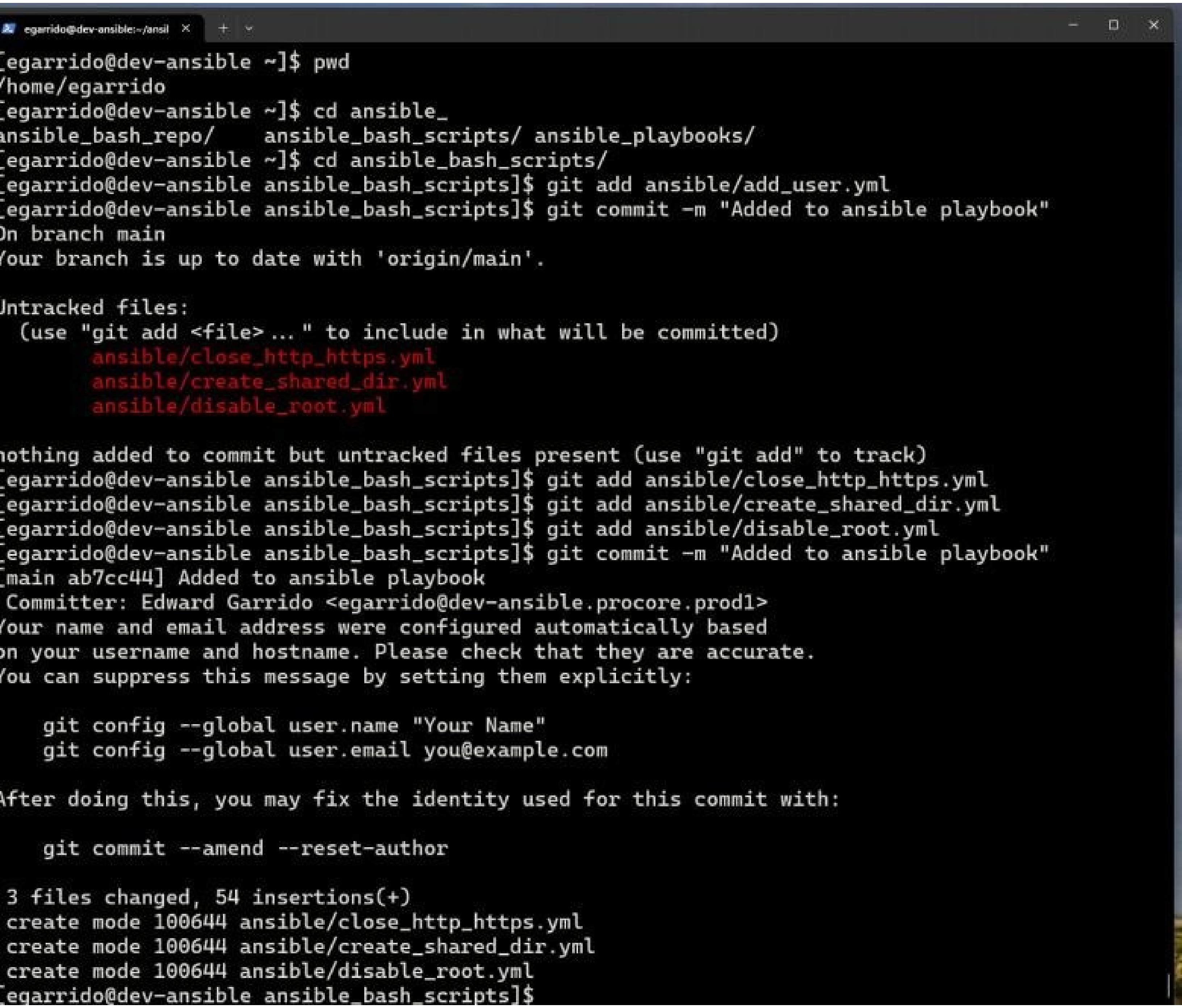
The changes are then pushed successfully to the main branch of the remote GitLab repository, confirming the new playbook file has been uploaded and the local and remote repositories are in sync.



```
[egarrido@dev-ansible playbooks]$ cd
[egarrido@dev-ansible ~]$
[egarrido@dev-ansible ~]$ cd ~/ansible_bash_scripts
[egarrido@dev-ansible ansible_bash_scripts]$ git add ansible/
add_user.yml           close_http_https.yml   create_shared_dir.yml  disable_root.yml
[egarrido@dev-ansible ansible_bash_scripts]$ git add ansible/add_user.yml
[egarrido@dev-ansible ansible_bash_scripts]$ git commit -m "Added Ansible playbooks"
[main 5a85bc0] Added Ansible playbooks
 Committer: Edward Garrido <egarrido@dev-ansible.procore.prod1>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly:

    git config --global user.name "Your Name"
    git config --global user.email you@example.com

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

 1 file changed, 22 insertions(+)
 create mode 100644 ansible/add_user.yml
[egarrido@dev-ansible ansible_bash_scripts]$ git push origin main
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 586 bytes | 586.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://gitlab.com/edwardgarrido3066/ansible-bash-repo.git
   50fde55..5a85bc0  main -> main
[egarrido@dev-ansible ansible_bash_scripts]$
```

The terminal session shows navigation into the ansible_bash_scripts repository and an attempt to commit changes to the main branch. An initial commit attempt reports untracked Ansible playbook files, prompting them to be explicitly added.

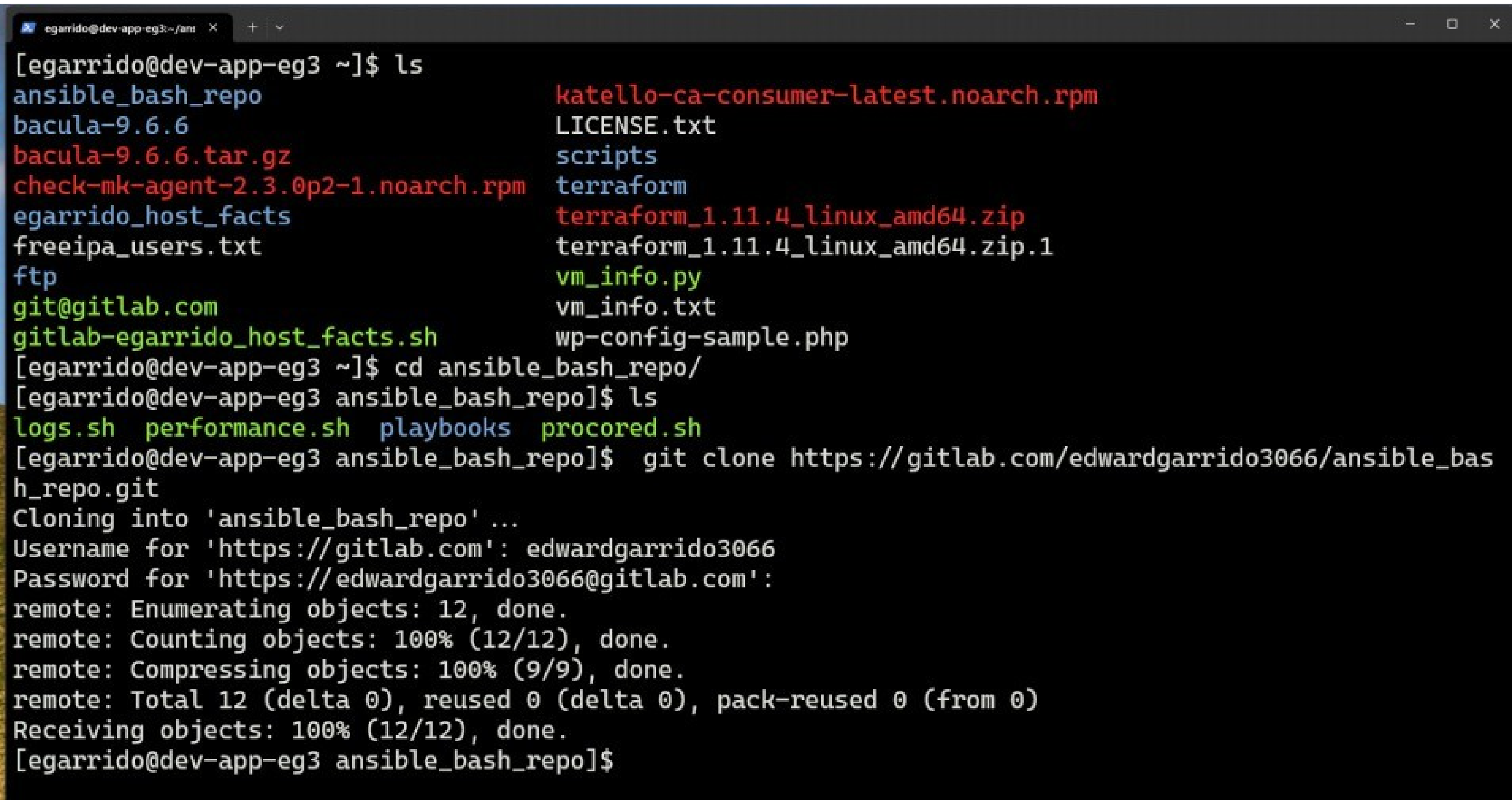
After staging the remaining playbooks, a successful commit is completed with a message indicating updates to the Ansible playbook content. Git reports the number of files created and inserted, confirming the new playbooks are now tracked in the repository.



```
[egarrido@dev-ansible ~]$ pwd
/home/egarrido
[egarrido@dev-ansible ~]$ cd ansible_
ansible_bash_repo/    ansible_bash_scripts/ ansible_playbooks/
[egarrido@dev-ansible ~]$ cd ansible_bash_scripts/
[egarrido@dev-ansible ansible_bash_scripts]$ git add ansible/add_user.yml
[egarrido@dev-ansible ansible_bash_scripts]$ git commit -m "Added to ansible playbook"
On branch main
Your branch is up to date with 'origin/main'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        ansible/close_http_https.yml
        ansible/create_shared_dir.yml
        ansible/disable_root.yml

nothing added to commit but untracked files present (use "git add" to track)
[egarrido@dev-ansible ansible_bash_scripts]$ git add ansible/close_http_https.yml
[egarrido@dev-ansible ansible_bash_scripts]$ git add ansible/create_shared_dir.yml
[egarrido@dev-ansible ansible_bash_scripts]$ git add ansible/disable_root.yml
[egarrido@dev-ansible ansible_bash_scripts]$ git commit -m "Added to ansible playbook"
[main ab7cc44] Added to ansible playbook
 Committer: Edward Garrido <egarrido@dev-ansible.procore.prod1>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly:

    git config --global user.name "Your Name"
    git config --global user.email you@example.com

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

 3 files changed, 54 insertions(+)
 create mode 100644 ansible/close_http_https.yml
 create mode 100644 ansible/create_shared_dir.yml
 create mode 100644 ansible/disable_root.yml
[egarrido@dev-ansible ansible_bash_scripts]$
```

A terminal session shows a home directory listing that includes various scripts, archives, configuration files, and an ansible_bash_repo directory. After navigating into ansible_bash_repo, the contents are listed, revealing shell scripts and a playbooks directory.

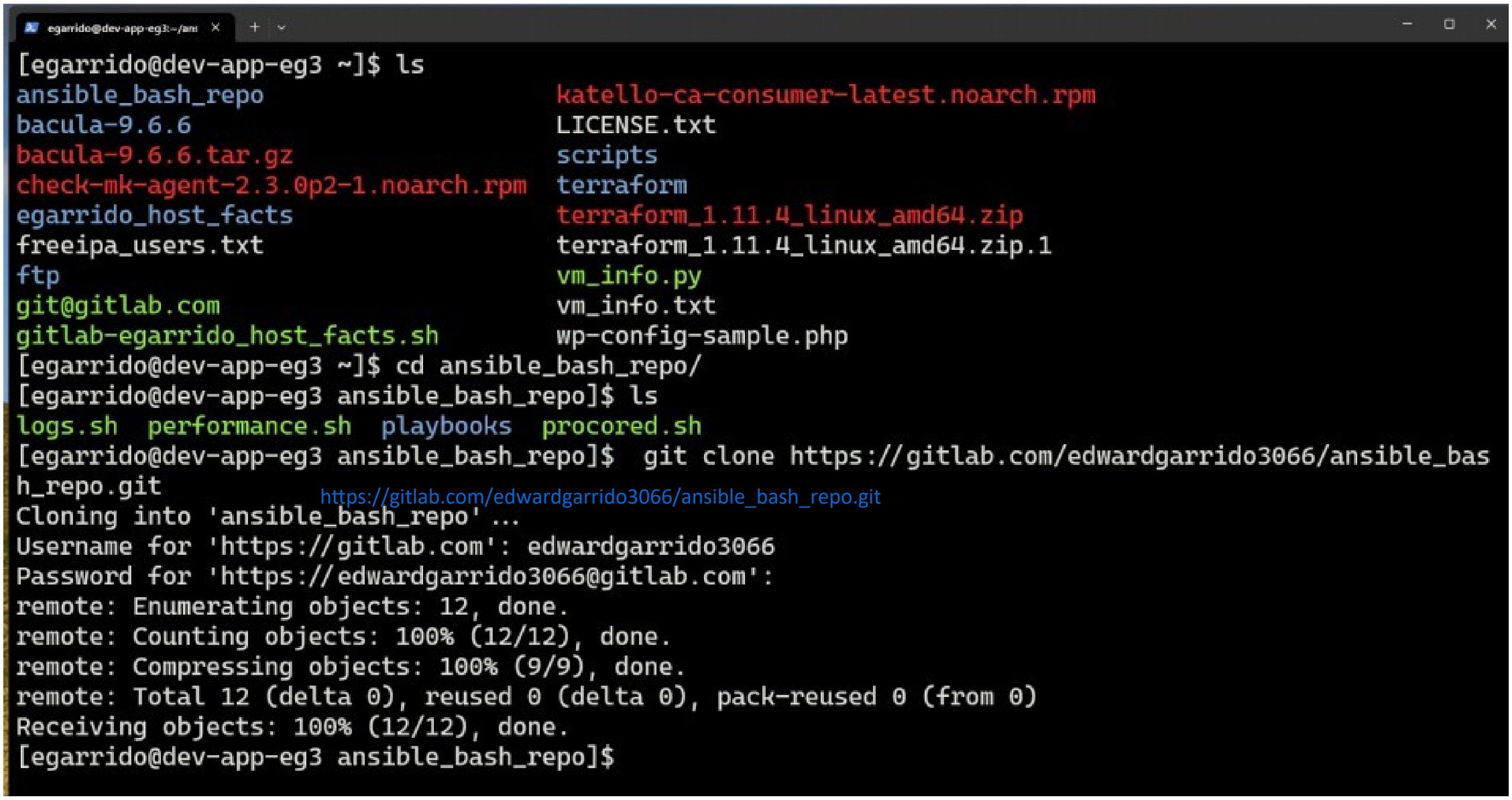
A Git clone command is then executed for the ansible_bash_repo repository from GitLab using HTTPS authentication. The clone completes successfully after prompting for the GitLab username and password, and the repository contents are downloaded into the local directory.



```
[egarrido@dev-app-eg3 ~]$ ls
ansible_bash_repo                    katello-ca-consumer-latest.noarch.rpm
bacula-9.6.6                         LICENSE.txt
bacula-9.6.6.tar.gz                  scripts
check-mk-agent-2.3.0p2-1.noarch.rpm  terraform
egarrido_host_facts                  terraform_1.11.4_linux_amd64.zip
freeipa_users.txt                    terraform_1.11.4_linux_amd64.zip.1
ftp                                  vm_info.py
git@gitlab.com                       vm_info.txt
gitlab-egarrido_host_facts.sh        wp-config-sample.php
[egarrido@dev-app-eg3 ~]$ cd ansible_bash_repo/
[egarrido@dev-app-eg3 ansible_bash_repo]$ ls
logs.sh  performance.sh  playbooks  procored.sh
[egarrido@dev-app-eg3 ansible_bash_repo]$  git clone https://gitlab.com/edwardgarrido3066/ansible_bas
h_repo.git
Cloning into 'ansible_bash_repo'...
Username for 'https://gitlab.com': edwardgarrido3066
Password for 'https://edwardgarrido3066@gitlab.com':
remote: Enumerating objects: 12, done.
remote: Counting objects: 100% (12/12), done.
remote: Compressing objects: 100% (9/9), done.
remote: Total 12 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (12/12), done.
[egarrido@dev-app-eg3 ansible_bash_repo]$
```

A terminal session lists the contents of the user’s home directory, showing various scripts, archives, RPM files, and an ansible_bash_repo directory. The session then navigates into ansible_bash_repo, where shell scripts and a playbooks directory are visible.

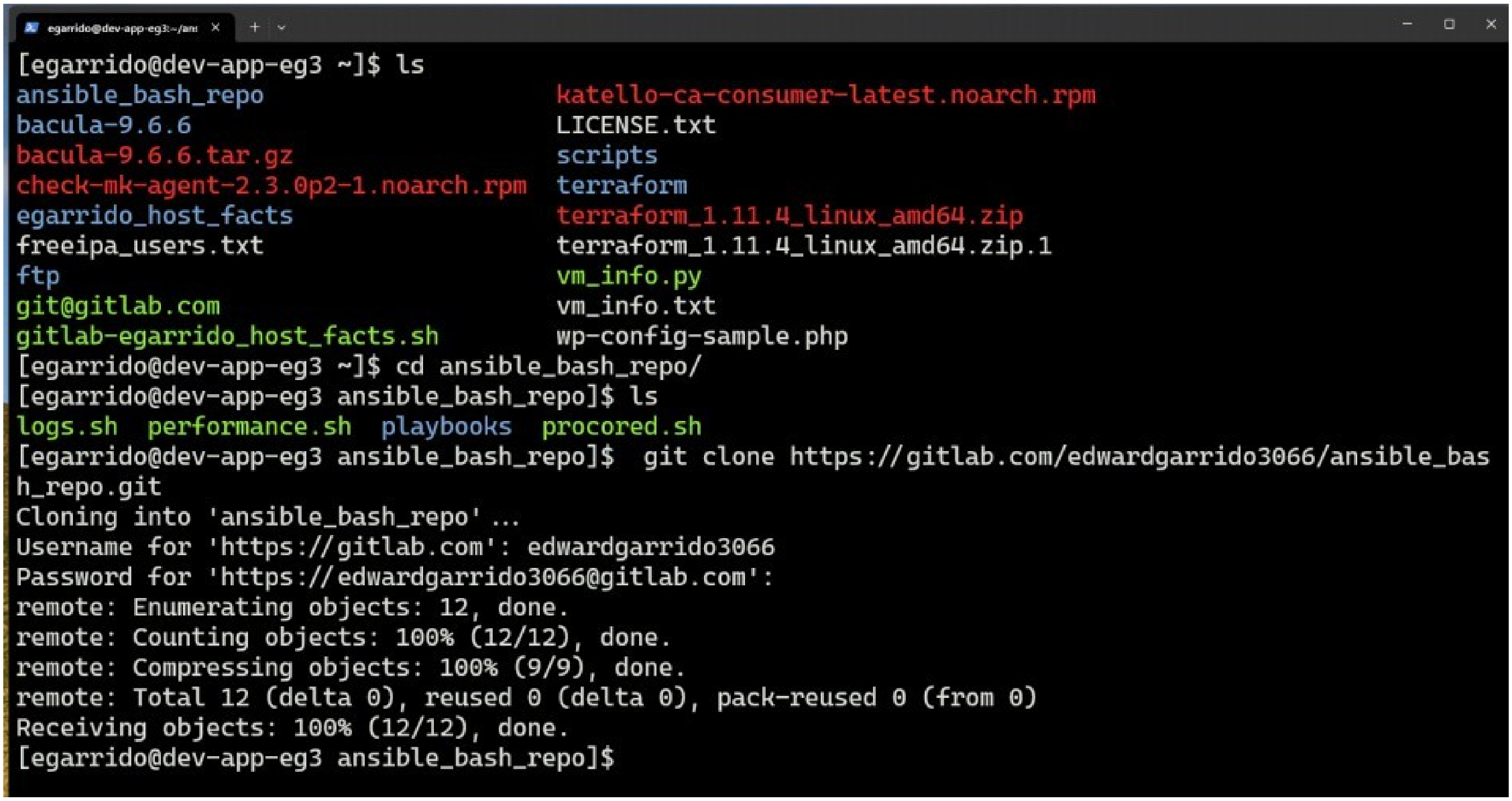
A Git clone command is executed to clone the ansible_bash_repo repository from GitLab using HTTPS authentication. After entering the GitLab username and password, the repository is cloned successfully, with all objects received and the prompt returning inside the repository directory.



```
[egarrido@dev-app-eg3 ~]$ ls
ansible_bash_repo                    katello-ca-consumer-latest.noarch.rpm
bacula-9.6.6                         LICENSE.txt
bacula-9.6.6.tar.gz                  scripts
check-mk-agent-2.3.0p2-1.noarch.rpm  terraform
egarrido_host_facts                  terraform_1.11.4_linux_amd64.zip
freeipa_users.txt                    terraform_1.11.4_linux_amd64.zip.1
ftp                                  vm_info.py
git@gitlab.com                       vm_info.txt
gitlab-egarrido_host_facts.sh        wp-config-sample.php
[egarrido@dev-app-eg3 ~]$ cd ansible_bash_repo/
[egarrido@dev-app-eg3 ansible_bash_repo]$ ls
logs.sh  performance.sh  playbooks  procored.sh
[egarrido@dev-app-eg3 ansible_bash_repo]$  git clone https://gitlab.com/edwardgarrido3066/ansible_bas
h_repo.git
                       https://gitlab.com/edwardgarrido3066/ansible_bash_repo.git
Cloning into 'ansible_bash_repo'...
Username for 'https://gitlab.com': edwardgarrido3066
Password for 'https://edwardgarrido3066@gitlab.com':
remote: Enumerating objects: 12, done.
remote: Counting objects: 100% (12/12), done.
remote: Compressing objects: 100% (9/9), done.
remote: Total 12 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (12/12), done.
[egarrido@dev-app-eg3 ansible_bash_repo]$
```

A terminal session shows the contents of the user’s home directory, which includes various RPM packages, tar archives, scripts, Terraform files, and an existing ansible_bash_repo directory. The user navigates into ansible_bash_repo and lists its contents, revealing shell scripts and a playbooks directory.

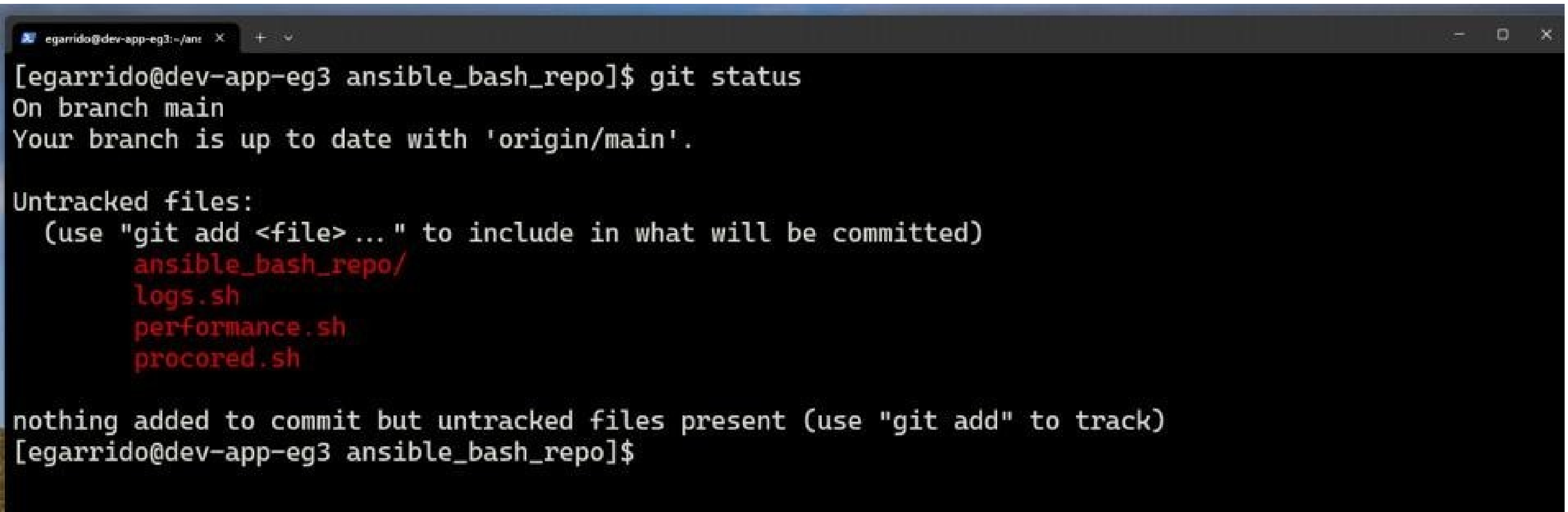
A Git clone command is then run to clone the ansible_bash_repo repository from GitLab using HTTPS authentication. After entering the GitLab username and password, the clone completes successfully, and all repository objects are downloaded.



```
[egarrido@dev-app-eg3 ~]$ ls
ansible_bash_repo                    katello-ca-consumer-latest.noarch.rpm
bacula-9.6.6                         LICENSE.txt
bacula-9.6.6.tar.gz                  scripts
check-mk-agent-2.3.0p2-1.noarch.rpm  terraform
egarrido_host_facts                  terraform_1.11.4_linux_amd64.zip
freeipa_users.txt                    terraform_1.11.4_linux_amd64.zip.1
ftp                                  vm_info.py
git@gitlab.com                       vm_info.txt
gitlab-egarrido_host_facts.sh        wp-config-sample.php
[egarrido@dev-app-eg3 ~]$ cd ansible_bash_repo/
[egarrido@dev-app-eg3 ansible_bash_repo]$ ls
logs.sh  performance.sh  playbooks  procored.sh
[egarrido@dev-app-eg3 ansible_bash_repo]$  git clone https://gitlab.com/edwardgarrido3066/ansible_bas
h_repo.git
Cloning into 'ansible_bash_repo'...
Username for 'https://gitlab.com': edwardgarrido3066
Password for 'https://edwardgarrido3066@gitlab.com':
remote: Enumerating objects: 12, done.
remote: Counting objects: 100% (12/12), done.
remote: Compressing objects: 100% (9/9), done.
remote: Total 12 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (12/12), done.
[egarrido@dev-app-eg3 ansible_bash_repo]$
```

The terminal displays the result of a git status command run inside the ansible_bash_repo repository. The repository is on the main branch and is fully up to date with origin/main.

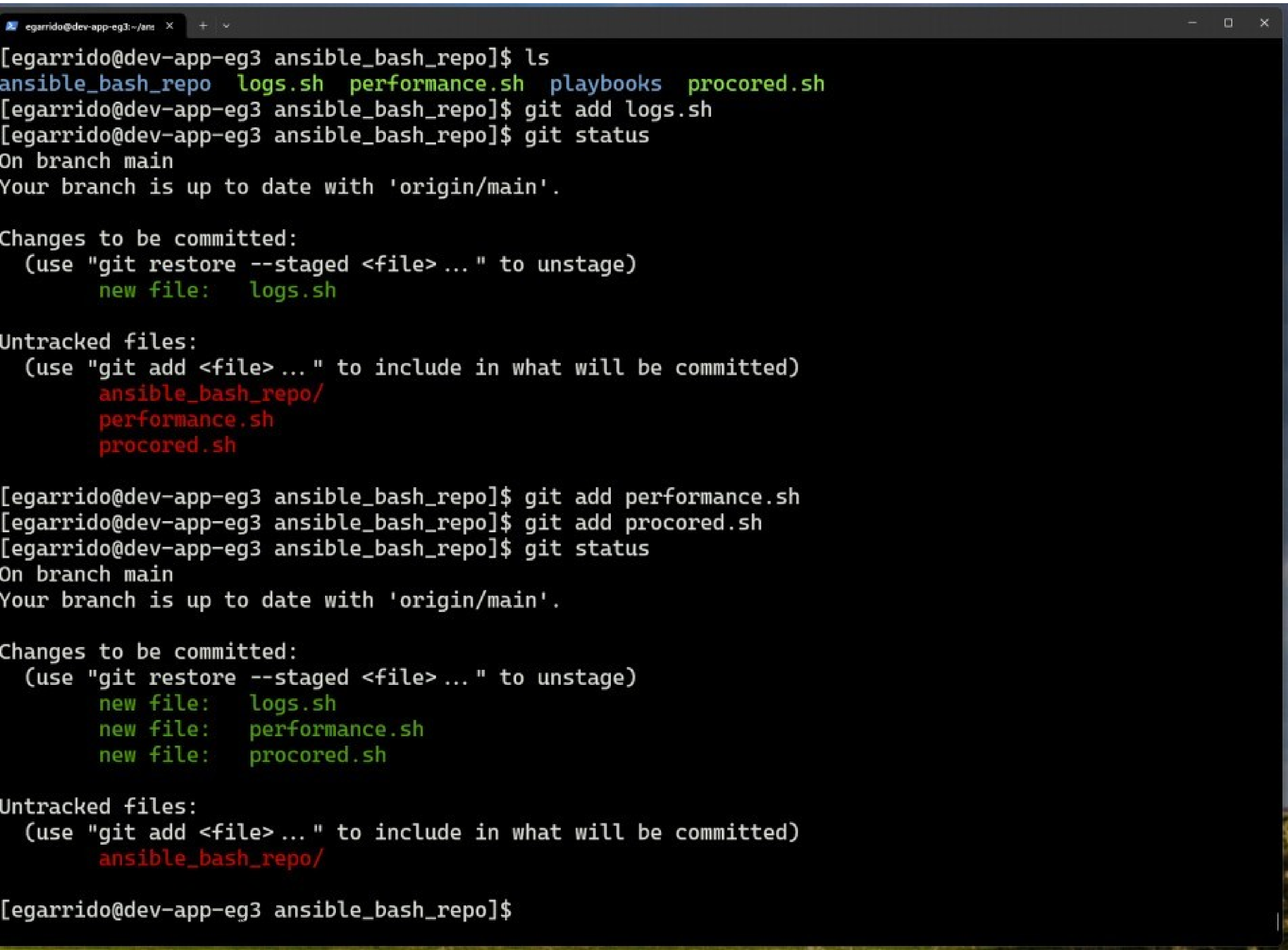
Git reports several untracked files, including a nested ansible_bash_repo directory and the scripts logs.sh, performance.sh, and procoreed.sh. No changes are staged or committed, and the output indicates that files must be added before they can be tracked by Git.



```
[egarrido@dev-app-eg3 ansible_bash_repo]$ git status
On branch main
Your branch is up to date with 'origin/main'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        ansible_bash_repo/
        logs.sh
        performance.sh
        procored.sh

nothing added to commit but untracked files present (use "git add" to track)
[egarrido@dev-app-eg3 ansible_bash_repo]$
```

The terminal shows files in the ansible_bash_repo directory being staged for commit. Individual scripts (logs.sh, performance.sh, and procoreed.sh) are added one by one, and git status confirms they are staged as new files ready to be committed.

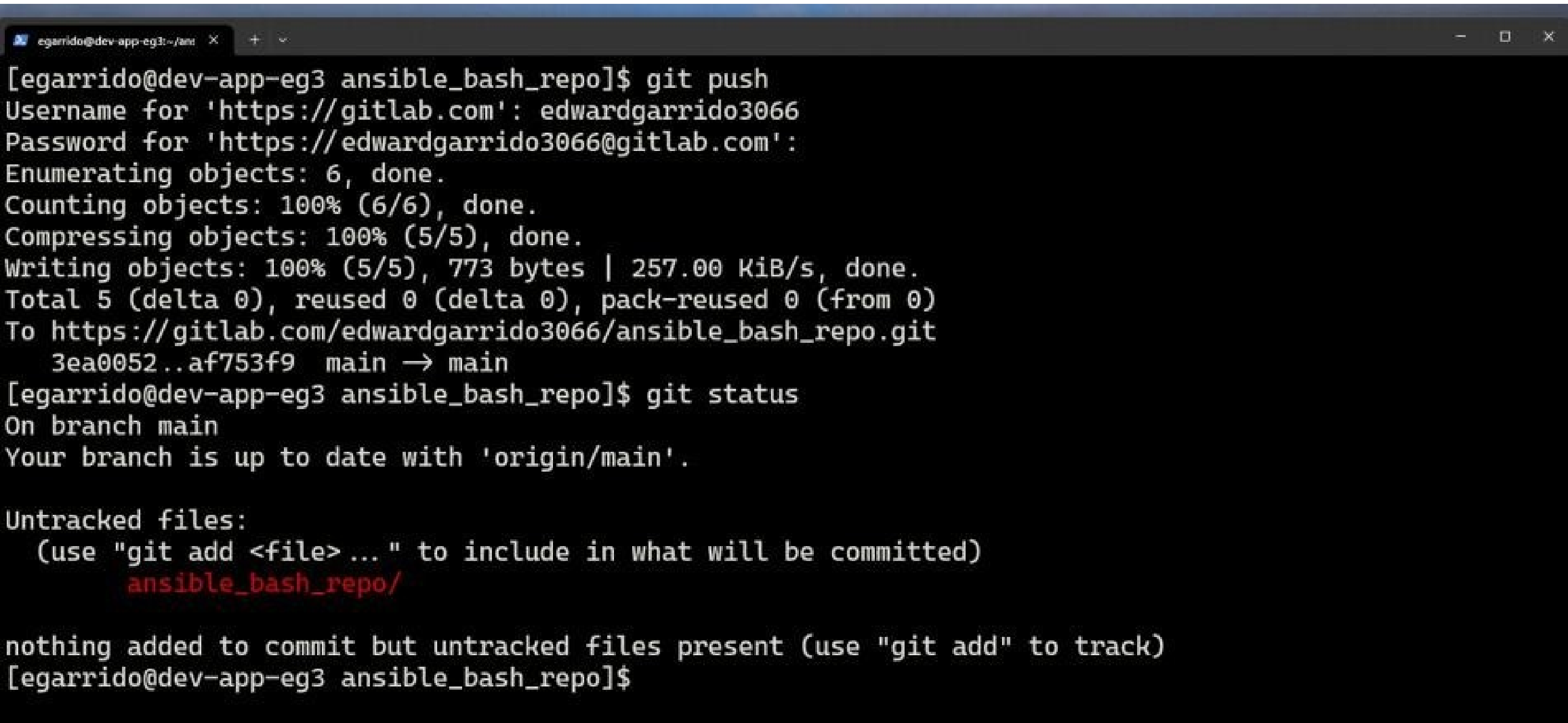
A nested ansible_bash_repo directory remains untracked, indicating it has not yet been added to version



```
[egarrido@dev-app-eg3 ansible_bash_repo]$ ls
ansible_bash_repo  logs.sh  performance.sh  playbooks  procored.sh
[egarrido@dev-app-eg3 ansible_bash_repo]$ git add logs.sh
[egarrido@dev-app-eg3 ansible_bash_repo]$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   logs.sh

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        ansible_bash_repo/
        performance.sh
        procored.sh

[egarrido@dev-app-eg3 ansible_bash_repo]$ git add performance.sh
[egarrido@dev-app-eg3 ansible_bash_repo]$ git add procored.sh
[egarrido@dev-app-eg3 ansible_bash_repo]$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   logs.sh
        new file:   performance.sh
        new file:   procored.sh

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        ansible_bash_repo/

[egarrido@dev-app-eg3 ansible_bash_repo]$
```

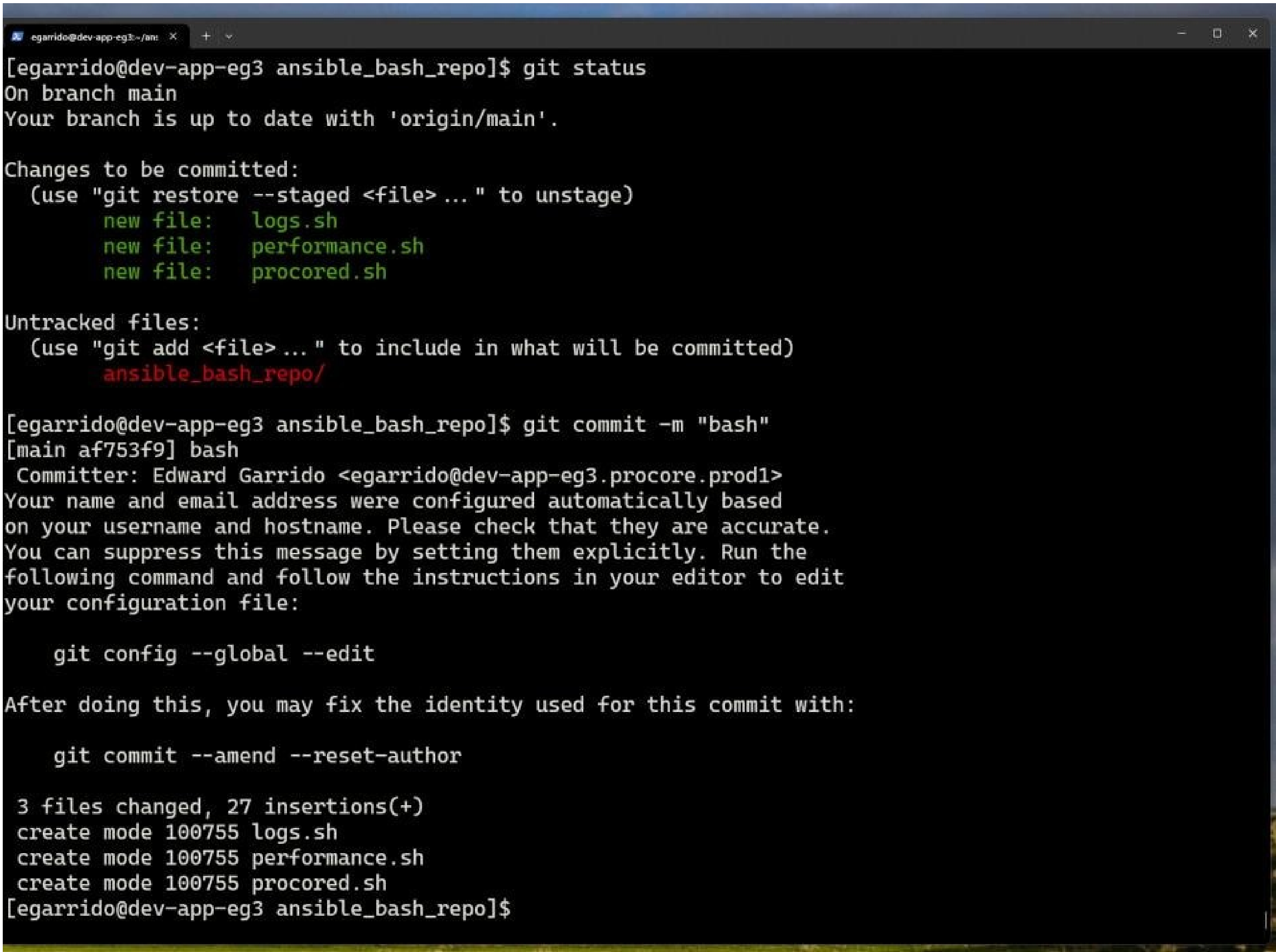
The terminal output shows a successful push of staged files to the main branch on GitLab after authenticating with the user account. Git confirms that all objects are written and the remote branch is updated.

A subsequent git status check shows the local branch is fully up to date with origin/main, with only a nested ansible_bash_repo directory remaining untracked and no pending changes staged for commit.



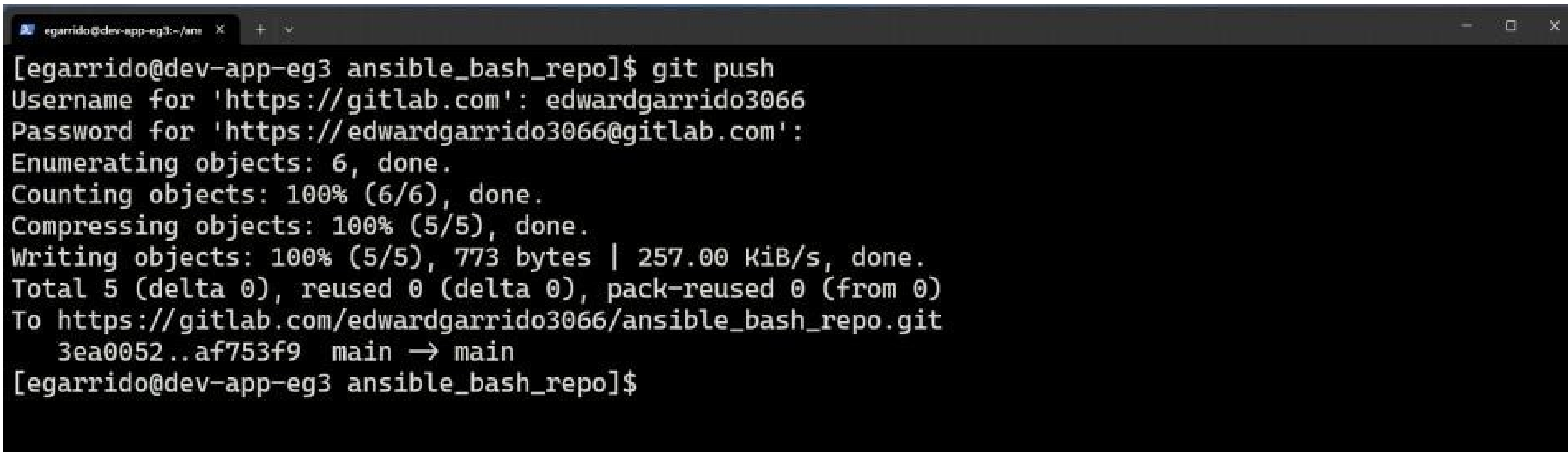
```
[egarrido@dev-app-eg3 ansible_bash_repo]$ git push
Username for 'https://gitlab.com': edwardgarrido3066
Password for 'https://edwardgarrido3066@gitlab.com':
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 773 bytes | 257.00 KiB/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://gitlab.com/edwardgarrido3066/ansible_bash_repo.git
   3ea0052..af753f9  main -> main
[egarrido@dev-app-eg3 ansible_bash_repo]$ git status
On branch main
Your branch is up to date with 'origin/main'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        ansible_bash_repo/

nothing added to commit but untracked files present (use "git add" to track)
[egarrido@dev-app-eg3 ansible_bash_repo]$
```

The terminal shows the repository on the main branch with three staged Bash scripts ready to be committed. A commit is created with the message “bash”, recording the addition of logs.sh, performance.sh, and procoreed.sh. Git confirms the commit details, including file creation and insertion counts, and notes that the author identity was auto-configured.



```
[egarrido@dev-app-eg3 ansible_bash_repo]$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   logs.sh
        new file:   performance.sh
        new file:   procored.sh

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        ansible_bash_repo/

[egarrido@dev-app-eg3 ansible_bash_repo]$ git commit -m "bash"
[main af753f9] bash
 Committer: Edward Garrido <egarrido@dev-app-eg3.procore.prod1>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

    git config --global --edit

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

 3 files changed, 27 insertions(+)
 create mode 100755 logs.sh
 create mode 100755 performance.sh
 create mode 100755 procored.sh
[egarrido@dev-app-eg3 ansible_bash_repo]$
```

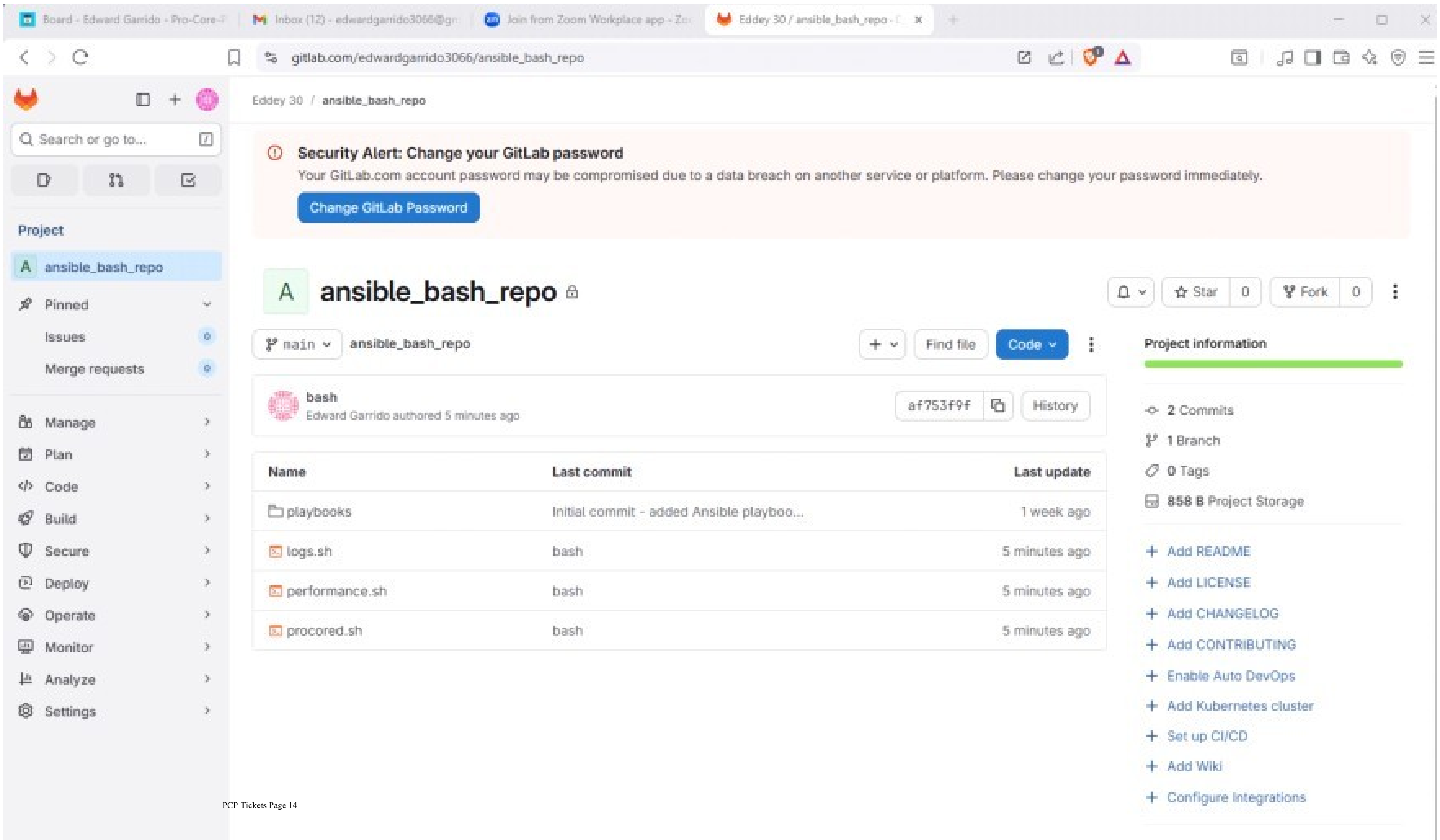
The terminal output shows a successful push of the latest commit to the main branch on GitLab. After authenticating with the GitLab account, all objects are enumerated, compressed, and written without errors, and the remote branch is updated to reflect the new commit.



```
[egarrido@dev-app-eg3 ansible_bash_repo]$ git push
Username for 'https://gitlab.com': edwardgarrido3066
Password for 'https://edwardgarrido3066@gitlab.com':
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 773 bytes | 257.00 KiB/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://gitlab.com/edwardgarrido3066/ansible_bash_repo.git
   3ea0052..af753f9  main -> main
[egarrido@dev-app-eg3 ansible_bash_repo]$
```

The GitLab repository page for ansible_bash_repo is open on the main branch. A security alert banner at the top advises changing the account password. The file list shows a playbooks directory and three Bash scripts—logs.sh, performance.sh, and procoreed.sh—all recently committed.

The commit history indicates two commits, with the most recent labeled “bash” authored a few minutes ago. The right sidebar displays basic project details, including branch count, storage usage, and options to add documentation, CI/CD, or integrations, confirming the repository is active and up to date.


Board - Edward Garrido - Pro-Core-
Inbox (12) - edwardgarrido3066@gr
Join from Zoom Workplace app - Zo
Eddey 30 / ansible_bash_repo -
gitlab.com/edwardgarrido3066/ansible_bash_repo
Search or go to...
Project
ansible_bash_repo
Pinned
Issues 0
Merge requests 0
Manage
Plan
Code
Build
Secure
Deploy
Operate
Monitor
Analyze
Settings
Eddey 30 / ansible_bash_repo

Security Alert: Change your GitLab password

Your GitLab.com account password may be compromised due to a data breach on another service or platform. Please change your password immediately.

bash
Edward Garrido authored 5 minutes ago
af753f9f

Name	Last commit	Last update
playbooks	Initial commit - added Ansible playboo...	1 week ago
logs.sh	bash	5 minutes ago
performance.sh	bash	5 minutes ago
procored.sh	bash	5 minutes ago

Project information

- 2 Commits
- 1 Branch
- 0 Tags
- 858 B Project Storage

The most recent work items focus on building, automating, and validating a Linux-based infrastructure using Ansible, Bash, and supporting tooling. Tasks include creating and organizing Ansible inventories, playbooks, roles, and execution environments; configuring repositories, packages, services, users, permissions, cron jobs, and storage; and validating results across multiple host groups. Additional work covers troubleshooting SSH and sudo behavior, resolving Podman and Ansible Navigator execution issues, managing Git/GitLab repositories from creation through push and merge, and documenting outputs for repeatability. Together, these efforts demonstrate end-to-end automation, version control discipline, and practical system administration workflows in a RHEL/CentOS-style environment.