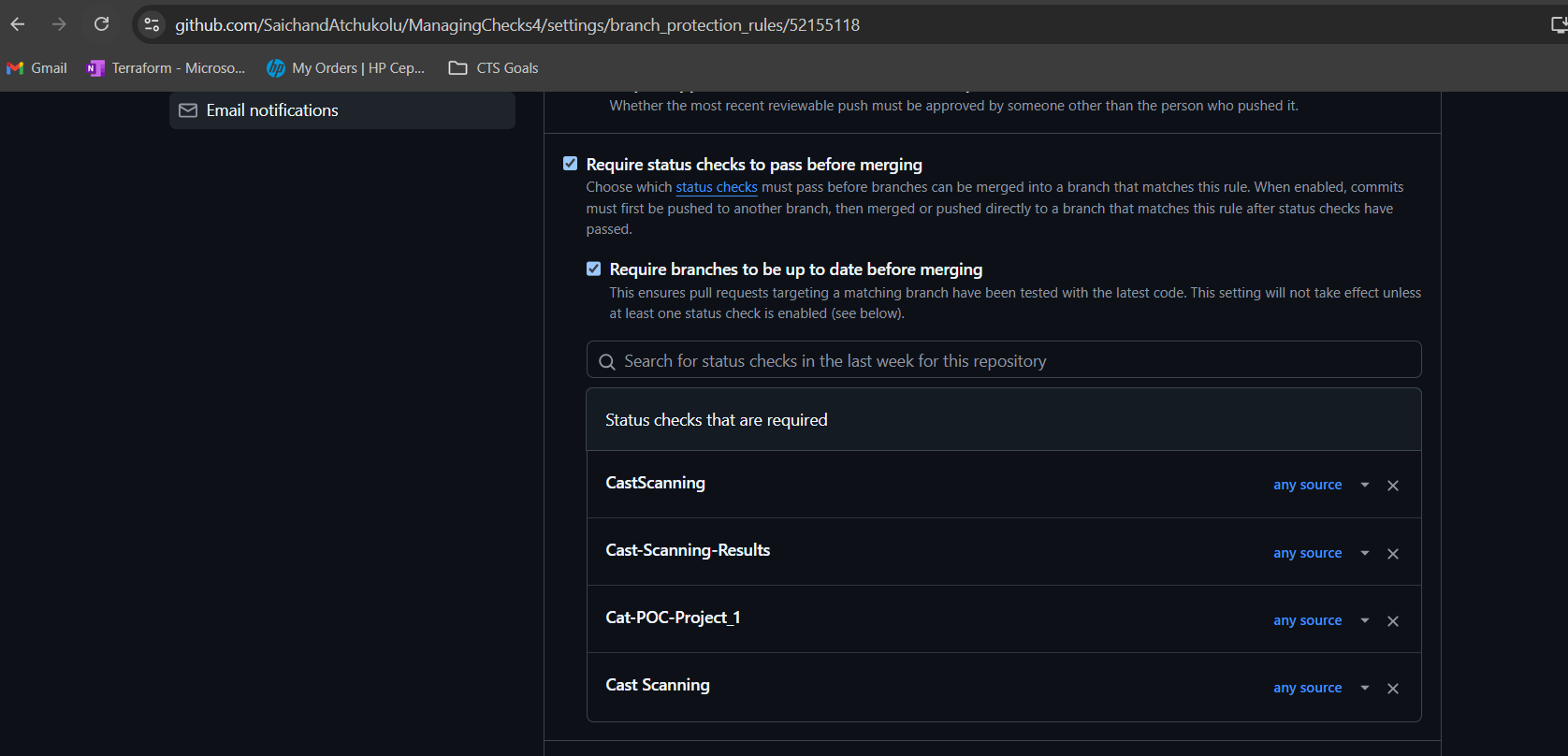
Below is the command to be enable the status checks in GitHub branch protection rules on main branch,

Command:



Screenshot from GitHub repo after execution:

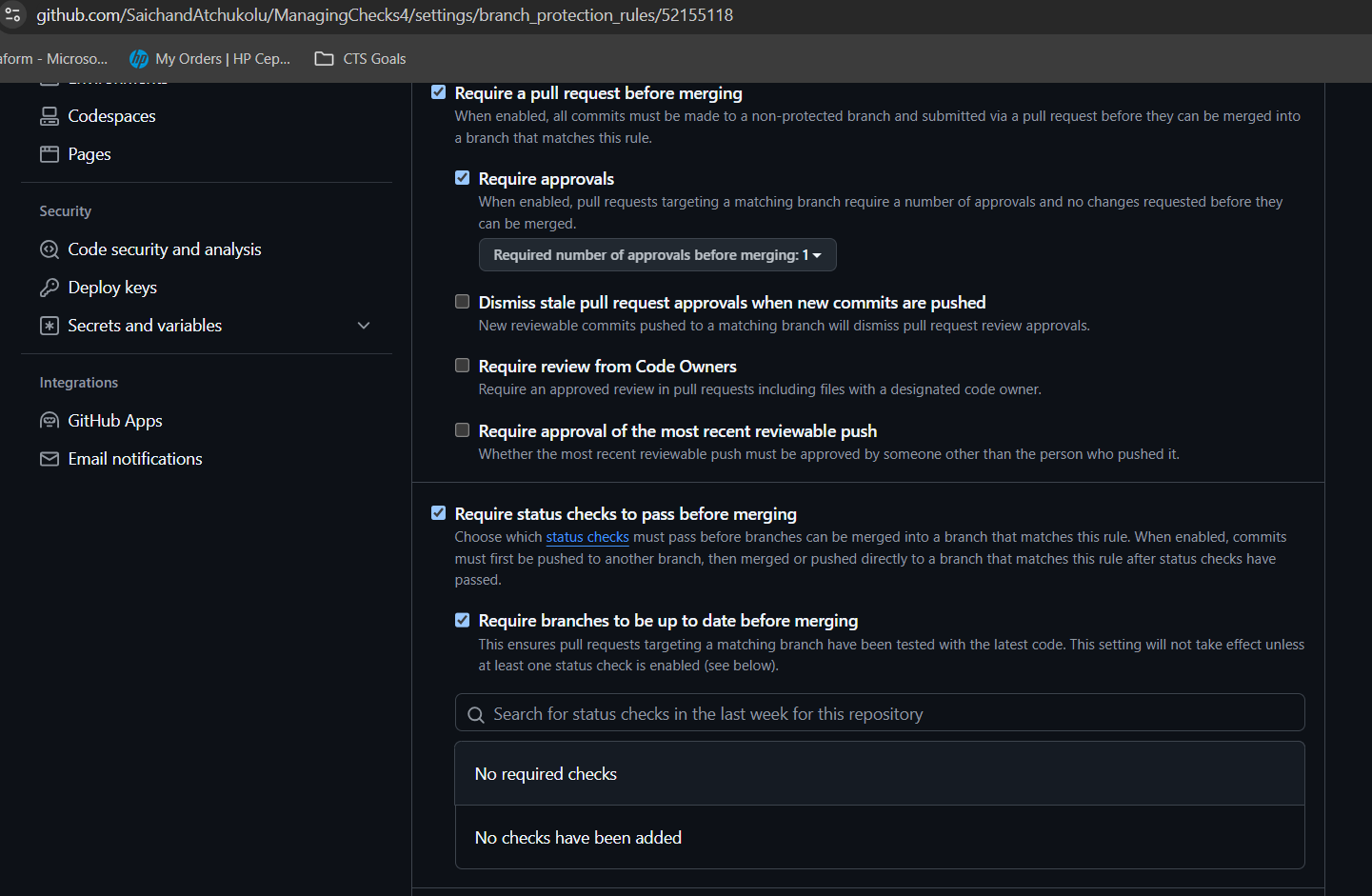


Below is the command to be disable the status checks in GitHub branch protection rules on main branch,

Command:



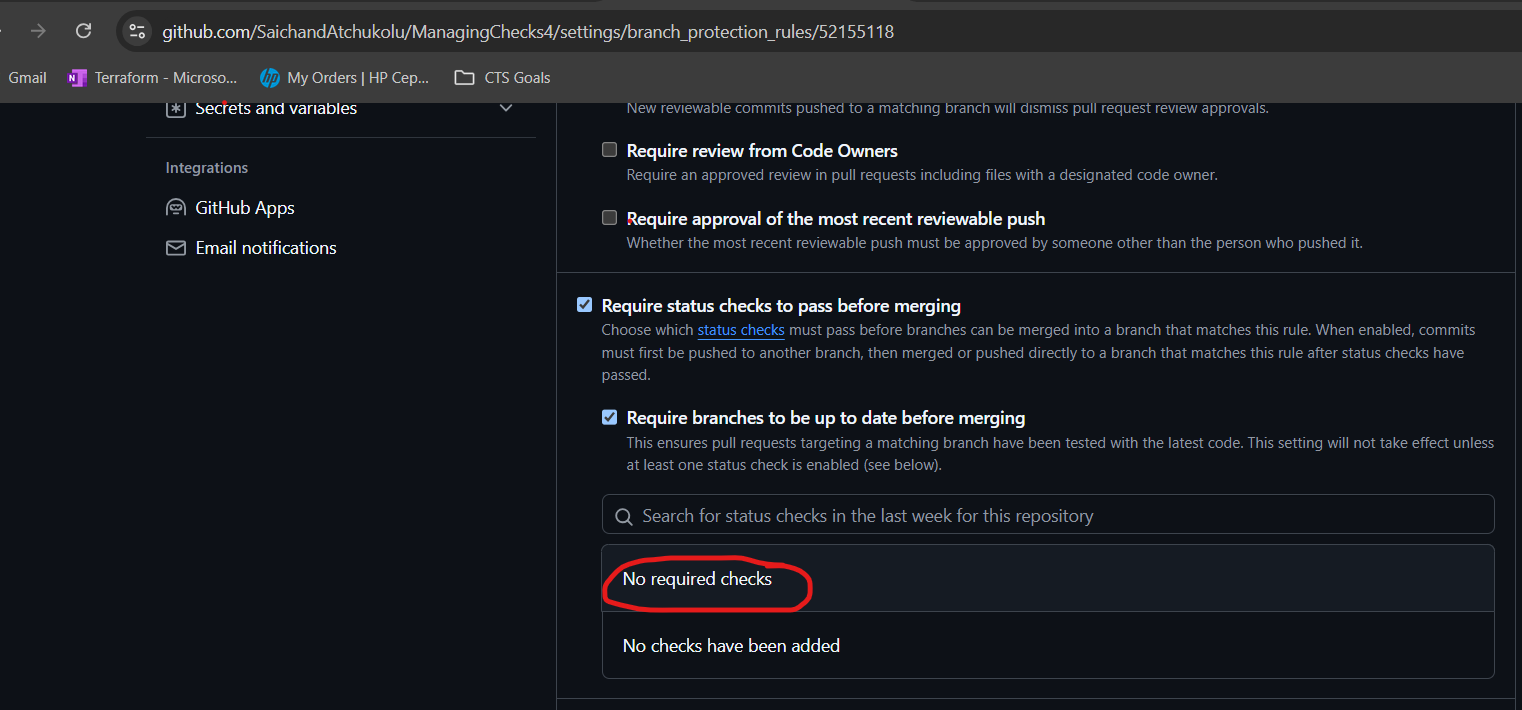
Screenshot from GitHub repo after execution:



Below is the command to be disable the status checks in GitHub branch protection rules on specific repo,



Screenshot from GitHub repo after execution:



Work instructions:

Work Instruction: Updating GitHub Branch Protection Rules Using PowerShell

1. Prepare the Necessary Files and Information

Ensure you have a GitHub Personal Access Token with the necessary permissions to modify repository settings.

Create a text file named repos.txt in the same directory as the script. This file should contain a list of repositories and branch details to be processed.

The format of each line in repos.txt should be:

php

Copy code

<Repository URL>,<Branch Name>,<Status Checks>

Example:

ruby

Copy code

https://github.com/example/repo1,main,check1|check2

https://github.com/example/repo2,dev,check3|check4

2. Open PowerShell

Open PowerShell on your computer.

3. Run the Script

Copy and paste the script content into your PowerShell window or save it as a .ps1 file (e.g., UpdateBranchProtection.ps1) and execute it.

4. Enter Your GitHub Personal Access Token

When prompted, enter your GitHub Personal Access Token.

This token will be securely stored and converted by the script for use.

5. Specify the Action and Repository Selection

When prompted, provide the action you want to perform:

Type enable to enable status checks on the repositories.

Type disable to disable status checks on the repositories.

Enter the repository selection:

Type all if you want to apply the action to all repositories listed in repos.txt.

Enter a comma-separated list of repository URLs or names if you want to apply the action to specific repositories only.

6. Wait for the Script to Process

The script will read each line in repos.txt and perform the specified action on each repository branch listed.

If successful, you will see a message confirming that the branch protection rules have been updated.

If there are any errors, the script will display an error message with details.

7. Review the Results

After the script completes, check the output messages to ensure all desired repositories were processed correctly.

If there are any errors or warnings, review the messages and make necessary adjustments.

8. Verify Changes on GitHub

Log in to your GitHub account and navigate to each repository to verify that the branch protection rules have been updated as expected.

9. Modify repos.txt if Needed

If you need to run the script again for different repositories or branches, update the repos.txt file accordingly.

By following these steps, you will effectively manage the branch protection rules for your GitHub repositories using this PowerShell script.