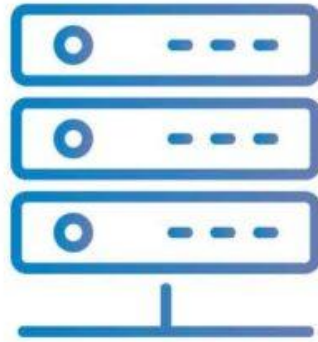


NPSS Server User Guide



By Michael Stich

Index

First Time Setup	3
Server Operation	11
Security Implications	14

First Time Setup Guide



By Michael Stich

In order to setup a NPSS Server for a target GitHub repository, follow these 5 steps:

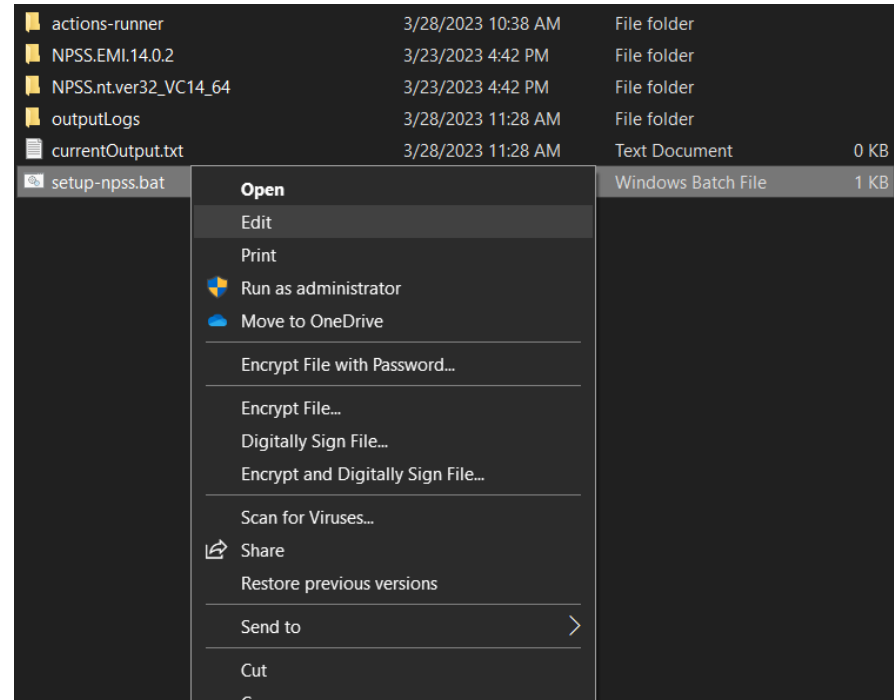
- 1. Find or allocate a machine that can natively run Window's PowerShell**
- 2. Download and extract the NPSS-Server Directory to your machine from this repository:**

<https://github.com/stichmc/npss-server>


3. Paste the NPSS and EMI package version you wish to use for the server into the NPSS-Server directory

actions-runner	3/30/2023 11:55 AM	File folder	
errorOutputLogs	4/25/2023 11:58 AM	File folder	
NPSS.EMI.yourVersionHere	3/23/2023 4:42 PM	File folder	
NPSS.nt.yourVersionHere	3/23/2023 4:42 PM	File folder	
currentOutput.txt	4/20/2023 2:49 PM	Text Document	0 KB
setup-npss.bat	3/28/2023 11:23 AM	Windows Batch File	1 KB

4. Right click setup-npss.bat and select the “Edit” option

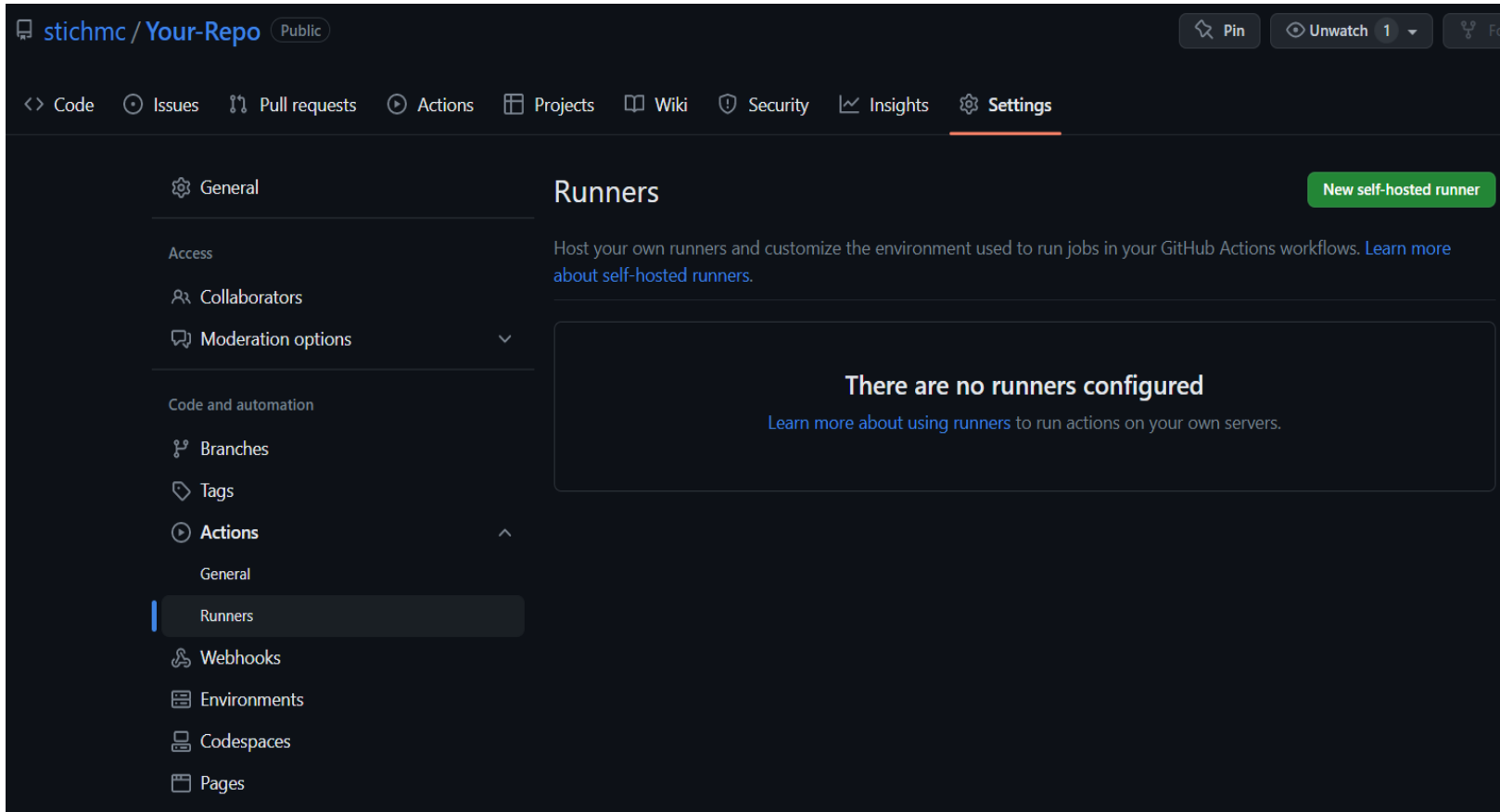


5. Inside the setup-npss.bat script, change “YOUR_VERSION_HERE” with the name of your NPSS and EMI package directories

 *setup-npss.bat - Notepad

```
File Edit Format View Help
@echo off
@SET _calldir=%CD%
@SET EMI_TOP=..\..\..\..\NPSS.EMI.YOUR_VERSION_HERE
@SET EMI_TOP=%EMI_TOP:"=%
set NPSS_CONFIG=nt
set NPSS_TOP=..\..\..\..\NPSS.nt.YOUR_VERSION_HERE|
set NPSS_DEV_TOP=%NPSS_TOP%\DLMdevkit
set NPSS_TEST_TOP=%NPSS_TOP%\Test
set MICODIR=%NPSS_TOP%
set DCLOD_PATH=%NPSS_TOP%\DLMComponents\nt
set PATH=%PATH%;%NPSS_TOP%\bin;%NPSS_TOP%\scripts
@SET PATH=%EMI_TOP%\scripts\nt;%EMI_TOP%\scripts\AutoDoc;%PATH%
@cd %_calldir%
@SET _calldir=
```

6. Navigate to your target GitHub Repository then: Open settings > Open Actions Dropdown > Open Runners



The screenshot shows the GitHub repository settings page for 'stichmc / Your-Repo'. The 'Settings' tab is selected, and the 'Runners' sub-tab is active. The page displays a message: 'There are no runners configured' with a link to 'Learn more about using runners'. A green button labeled 'New self-hosted runner' is visible in the top right corner of the Runners section.

7. Click the “New self-hosted runner” button

8. Select “Windows” and “x64”

Runners / Create self-hosted runner

Adding a self-hosted runner requires that you download, configure, and execute the GitHub Actions Runner. By downloading and configuring the GitHub Actions Runner, you agree to the [GitHub Terms of Service](#) or [GitHub Corporate Terms of Service](#), as applicable.

Runner image

☐ macOS

☐ Linux

☒ Windows

Architecture

x64

9. Without closing GitHub, open a new PowerShell **X86 32-bit** instance and navigate to your NPSS-Server directory

 Windows PowerShell (x86)

```
PS C:\Users\mstich> cd C:\Users\mstich\Desktop\NPSS-Server
PS C:\Users\mstich\Desktop\NPSS-Server>
```


10. Input the following instructions, (that you should see on GitHub), in order into your machine's PowerShell console:

```
# Create a folder under the drive root
$ mkdir actions-runner; cd actions-runner

# Download the latest runner package
$ Invoke-WebRequest -Uri https://github.com/actions/runner/releases/download/v2.303.0/actions-runner-win-x64-2.303.0.zip -OutFile actions-runner-win-x64-2.303.0.zip

# Optional: Validate the hash
$ if((Get-FileHash -Path actions-runner-win-x64-2.303.0.zip -Algorithm SHA256).Hash.ToUpper() -ne '2368cd782c5b0dba1af8f90e1cc1c294cce2d165ed24f026577304b66440b31e'.ToUpper()){ throw 'Computed checksum did not match' }

# Extract the installer
$ Add-Type -AssemblyName System.IO.Compression.FileSystem ;
[System.IO.Compression.ZipFile]::ExtractToDirectory("$PWD/actions-runner-win-x64-2.303.0.zip", "$PWD")
```

Configure

```
# Create the runner and start the configuration experience
$ ./config.cmd --url https://github.com/stichmc/fitgpt --token AXMFAAMSW5P7QYM3G0LSDGDEEW3XK
```

If you run the validate hash command and nothing prints to the console, the correct files have been installed.

12. Copy and upload the files in this repository to your GitHub repository:

https://github.com/stichmc/NPSS-Power-System-Library/tree/npss_runner_and_tests

13. Go to your repository's Settings tab -> Branches. Under branch protection rules, add this rule for the branch/branches you will be working on.

Protect matching branches


☐ **Require a pull request before merging**
When enabled, all commits must be made to a non-protected branch and submitted via a pull request before they can be merged into a branch that matches this rule.

☒ **Require status checks to pass before merging**
Choose which [status checks](#) must pass before branches can be merged into a branch that matches this rule. When enabled, commits must first be pushed to another branch, then merged or pushed directly to a branch that matches this rule after status checks have passed.

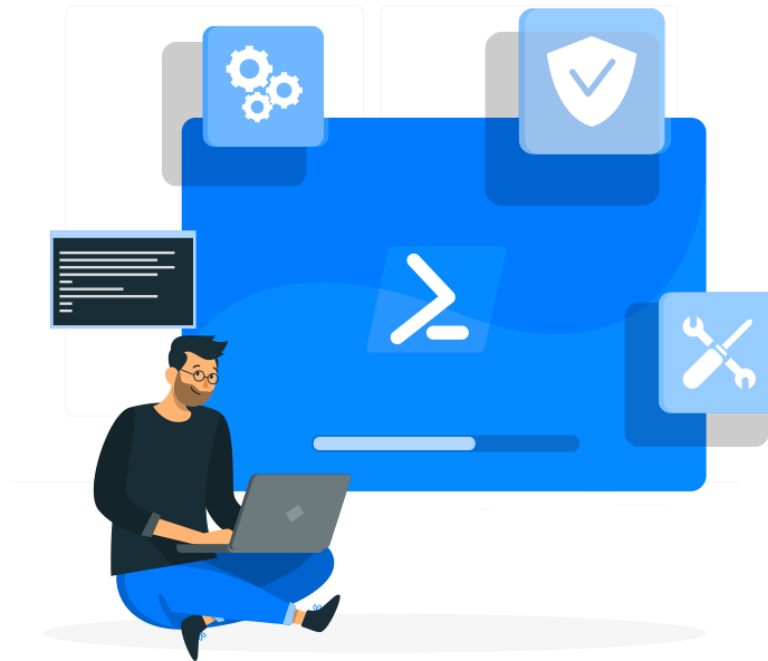
☐ **Require branches to be up to date before merging**
This ensures pull requests targeting a matching branch have been tested with the latest code. This setting will not take effect unless at least one status check is enabled (see below).

Status checks that are required.

YAML File Checker

 **GitHub Actions** ▾ ×

Server Operation Guide

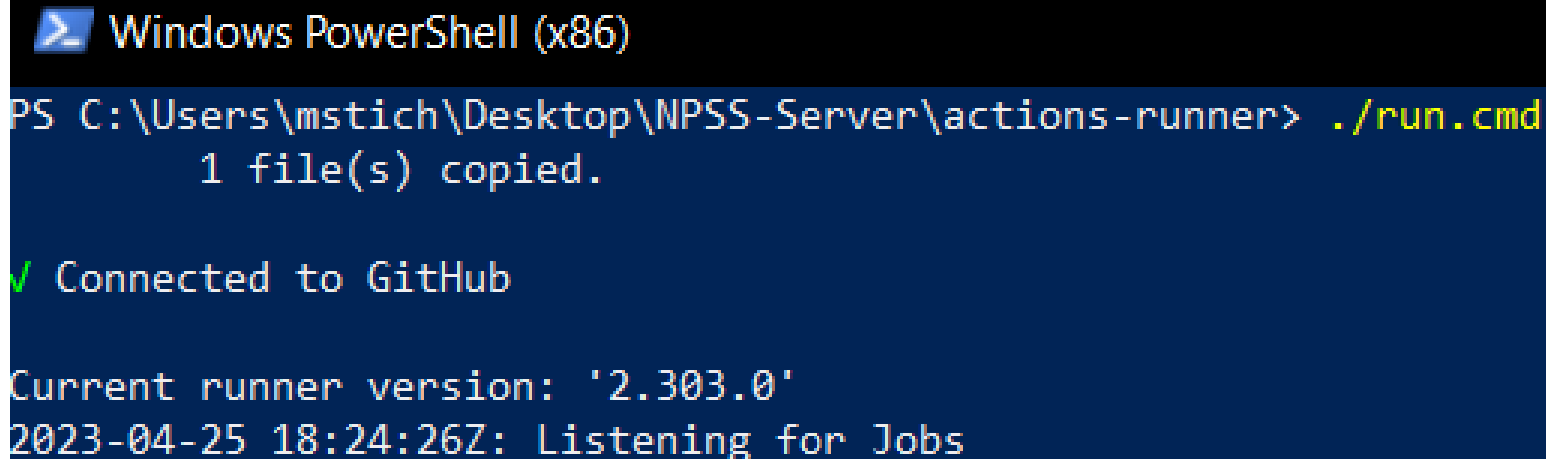


By Michael Stich

Starting The Runner

To startup your runner:

- Type `./run.cmd` into your PowerShell X86 32-bit console inside the actions-runner directory
- After a few moments, you should see this output to your console window:



```
> Windows PowerShell (x86)
PS C:\Users\mstich\Desktop\NPSS-Server\actions-runner> ./run.cmd
    1 file(s) copied.

✓ Connected to GitHub

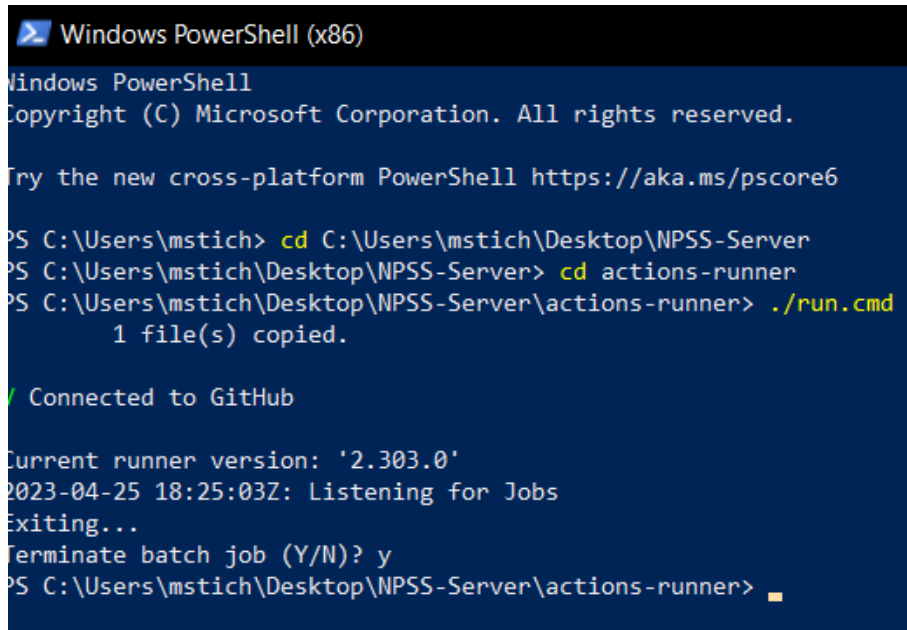
Current runner version: '2.303.0'
2023-04-25 18:24:26Z: Listening for Jobs
```

Shutting Down The Runner

To shut down your runner:

- Navigate to your PowerShell X86 32-bit console running the runner
- Left click inside the console and press ENTER followed by CTRL-C and input “Y”

**Do not shut the runner
down any other way or
you will need to reinstall
your runner**



```
Windows PowerShell (x86)
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\mstich> cd C:\Users\mstich\Desktop\NPSS-Server
PS C:\Users\mstich\Desktop\NPSS-Server> cd actions-runner
PS C:\Users\mstich\Desktop\NPSS-Server\actions-runner> ./run.cmd
1 file(s) copied.

/ Connected to GitHub

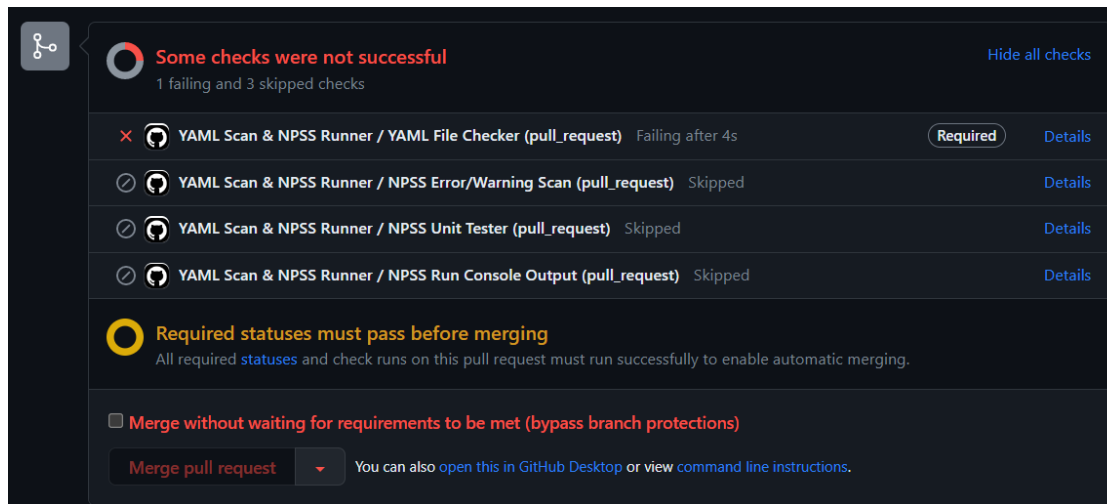
Current runner version: '2.303.0'
2023-04-25 18:25:03Z: Listening for Jobs
Exiting...
Terminate batch job (Y/N)? y
PS C:\Users\mstich\Desktop\NPSS-Server\actions-runner> █
```

Security Implications



Here are some security implications with the NPSS Server that would allow bad actors to take advantage of the system the runner is operating on.

1. YAML file modifications could potentially grant root access to the windows PowerShell kernel on the machine the runner is operating on. The included YAML file combats alterations to itself every time an NPSS run is requested. So it is **IMPERATIVE** whoever you allow direct commit access is trustworthy. If you receive a pull request and see the following error message:



DO NOT accept the pull request. The incoming code could potentially contain malicious YAML code.

2. The NPSS code being sent to the runner could be potentially malicious. There are operations in NPSS that could potentially interact maliciously with the computer hosting the runner. To circumvent these vulnerability, I recommend the use of a virtualization. VM's can add a layer of virtualization that would prevent any potential bad actors from using the NPSS code to connect to other computers on the network the host OS is on.

I recommend:



ORACLE[®]
VM
VirtualBox

