

Journey of Automation - Github, Galaxy, Fedora

Pavel Cahyna, Sergei Petrosian

Automation is more important than ever before in software project management

Learn how the Linux System Roles([url](#)) team leverages automation:

1. Automated Ansible role release and publish to Ansible Galaxy
2. Automated Fedora RPM build and publish with Packit

Automated GitHub Releases

1. A script executed manually to create a new tag in GitHub repository
 - a. Identify a new semantic version
 - b. Generate changelog using conventional commits
 - c. Create a PR with updated changelog
2. GitHub workflow that tags and releases GitHub repository once the changelog PR is merged
3. GitHub workflow that publishes repository into an upstream hub for a new release

Changelog Generation: Conventional Commits Format

Format:

```
<type><!>: PR Title
```

Example:

feat: Support custom data and logs storage paths

Figure out the new semantic version

- **!** - **MAJOR** bump
- **feat** - **MINOR** bump
- **fix**, **ci**, **test**,... - **PATCH** bump

Build changelog based on PR types

- `feat:` -> **New Features**
- `fix:` -> **Bug Fixes**
- `else` -> **Other Changes**

GitHub Release Process using Conventional PR Titles

1. A developer runs the script
2. This script collect merged PRs since last release
3. Using conventional commits format, identifies version and generates a new changelog
4. Pushes a PR with the updated changelog

Creating Releases

PR merge triggers workflow that creates GitHub tag and release, and publishes repository content to Ansible Galaxy

```
name: Tag, release, and publish role based on CHANGELOG.md push
on:
  push:
    branches:
      - main
    paths:
      - CHANGELOG.md
jobs:
  - name: Create tag...
  - name: Create Release...
  - name: Publish role to Galaxy...
```


Automated RPM Release with Packit