

# Reuso e Software Livre

## Perspectivas e desafios

Gabriel Fontes & Prof. Dra. Elisa Yumi Nakagawa

# Quem sou



- Aluno de mestrado da Prof. Elisa
- Parte do GELOS<sup>1</sup>
- Envolvido na comunidade do NixOS<sup>2</sup>
- Trabalhando com DevOps/SRE (atualmente na Magalu Cloud<sup>3</sup>)



---

<sup>1</sup><https://gelos.club>

<sup>2</sup><https://nixos.org>

<sup>3</sup><https://magalu.cloud>

## Quem sou (ii)



Tenho 4 gatos e 1 catioro. Foto da Brasa:



Figura 1: Gato Brasa

Para mais fotos me chamar no telegram (último slide)

# O que é Software Livre?



- Também conhecido como Software Open Source
- Livre para usar, estudar, modificar, e **reusar** para qualquer propósito



## O que é Software Livre? (ii) ●●●●○○○○○○○○○○

Definição da FSF [1]:

0. Liberdade para executar/usar como desejar, para qualquer propósito
1. Liberdade para estudar como funciona, e modificar para que funcione como você deseje
2. Liberdade para redistribuir cópias
3. Liberdade para distribuir cópias modificadas

# Onde o SL entra em Reuso? ●●●●●○○○○○○○○○

- É um dos casos mais comuns de reuso hoje em dia
- Pacotes reusáveis, bibliotecas
- Práticas, frameworks...
- Ferramental para reuso (sistemas de módulos, gerenciador de pacotes)
  - ▶ A abundância de OSS catalisou esse progresso



## Onde o SL entra em Reuso? (ii) ●●●●●○○○○○○○

Você provavelmente já reusa  
Software Livre hoje.

# Quais as vantagens?



- Acesso gratuito a software de qualidade
  - Evita-se retrabalho
  - Qualidade maior (e.g. software de criptografia feito por *experts*)
  - Não “reinventar a roda”
- Interoperabilidade
  - Você pode fazer parte de uma comunidade
- Manutenibilidade facilitada

# E os riscos e desafios?



- Qualidade mesmo?



## E os riscos e desafios? (ii)

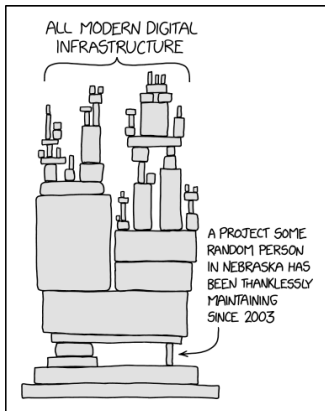


- Qualidade mesmo?
  - Quando se reusa software comercial, existem contratos
  - Maior parte de SL é sem garantias
  - Alguns projetos oferecem suporte (pago), mas a abundância de recursos gratuitos dificulta aceitar esses valores
- Auditoria
- Segurança
  - Ataques de cadeia de suprimentos
- Nem toda organização vê com bons olhos (*“Not Invented Here Syndrome”*)
- Excesso, principalmente de dependências externas, também pode gerar problemas
  - Caso *left-pad* [2]

## E os riscos e desafios? (iii)



- Sustentabilidade (eu pesquiso isso! ^v^)



XKCD #2347: Dependency [3]

# Por quê contribuir?



- Garantir sustentabilidade de Softwares que você utiliza
- Padronização
  - Entidades podem de-duplicar trabalho e melhorar interoperabilidade
- Custo de manutenção da sua feature é movido para upstream
- Algumas licenças exigem que mudanças sejam devolvidas
  - Copyleft (e.g. GPLv2 usada no kernel Linux)

# O que mais?



- Como tem sido pra você reusar software livre?
  - Quais suas dores?
  - O que gostaria de entender melhor?
- Quer trocar uma idéia? Meus contatos:
  - [g.fontes@usp.br](mailto:g.fontes@usp.br)
  - <https://m7.rs>
  - <https://linkedin.com/in/misterio7>
  - <https://github.com/misterio77>

## Bibliografia & Mais leituras ●●●●●●●●●●●●●●●●

- [1] «What is Free Software?». [Online]. Disponível em: <https://www.gnu.org/philosophy/free-sw.en.html>
- [2] Wikipedia contributors, «Npm left-pad incident — Wikipedia, The Free Encyclopedia». [Online]. Disponível em: [https://en.wikipedia.org/w/index.php?title=Npm\\_left-pad\\_incident&oldid=1304058784](https://en.wikipedia.org/w/index.php?title=Npm_left-pad_incident&oldid=1304058784)
- [3] «Dependency - xkcd.com». [Online]. Disponível em: <https://xkcd.com/2347/>
- [4] N. R. Mead, C. Woody, e S. Hissam, « Open Source Software: The Ultimate in Reuse or a Risk Not Worth Taking? », *Computer*, vol. 58, n.º 2, pp. 78–83, 2025, doi: [10.1109/MC.2024.3423908](https://doi.org/10.1109/MC.2024.3423908).