Open Source is Eating the World

Tony Wasserman
Carnegie Mellon Silicon Valley

SWforum.eu
23 March 2021
Others have used the same “eating” phrase.....
Credit where credit is due

- Michael Skok, VC, April, 2013
- Johann Gyger, blog, August, 2018
- IBM Developer Blog, February, 2019
- Joseph Jacks, OSS Capital, March, 2020
“Software is Eating the World”

- Marc Andreessen

General Partner, Andreessen Horowitz
and
Co-developer, Mosaic browser
Modern Societies Depend on Software

- **Transportation**
  - Avionics
  - Air and rail traffic control
  - Autonomous vehicles
  - Urban traffic signals
- **Communication**
  - Telecom networks: voice, email, messaging
- **Business**
  - E-commerce
  - Accounting
  - Stock market
  - Resumé screening
Software is mostly reliable, most of the time

Saturday's app updates for my iPad

Daily bug fixes
People expect software to be reliable...

- Check into a hotel room
- Connect telephone calls correctly
- Have online grocery orders delivered properly
- Calculate physiological data (heart rate, etc.)
- Provide stable operating systems and productivity tools
- Maintain login security (passwords, 2FA, TouchID, FaceID)
and trustworthy....

- Microsoft Exchange Server
- Tesla Autopilot
- Boeing 737 MAX avionics
- IT monitoring services, e.g., SolarWinds
- Cloud and web hosting companies
What Do We Mean by “Trustworthy”?

• Is it safe for widespread use?
• Does it provide reliable and accurate results?
• Is it secure against hackers and organized attacks?
• Would you stake your life (or that of your family) on its reliable operation?
• Do we trust the organizations and people responsible for its development and ongoing maintenance?

Would you fly on a Boeing 737 MAX with MCAS? Would you trust Tesla’s Autopilot to drive you?
Are AI/ML Applications Trustworthy?

• From algorithms to heuristics
  – Toolkits, e.g., TensorFlow and PyTorch

• Machine learning failures
  – Facial recognition
  – Medical diagnosis
  – Hiring algorithms and discrimination

• Development teams dominated by white males
  – More diversity needed
Sometimes the software gets indigestion as it eats the world
How does free and open source software (FOSS) fit into this picture?

- Is it reliable?
- Is it trustworthy?
- Is it secure?
- How does FOSS compare to traditional proprietary software in these respects?
- Is it eating the world?
Open Source Eating the World Means That

- Businesses and organizations are evaluating both proprietary and open source systems concurrently and finding
  - Total cost of FOSS systems is lower
  - Quality is comparable
  - Commercial support and training is available
- Companies building software systems are including FOSS in their products
- Much leading-edge technology (containers, large data management, ML/AI, etc.) is *de facto* open source
Different types of free and open source

- **Commercial** open source (COSS) has management team and company support structure, much like proprietary software companies
- **Foundation-based** FOSS projects have foundation oversight, OSI-approved licenses, external funding
- **Community-based** FOSS is independently developed and maintained
- All may have available third-party support and training
Examples of Widely-Used FOSS

- Linux, including Ubuntu, RHEL, Debian (45% of all servers)
- Apache HTTP Server (27% of all websites)
- Android (75% of mobile phones)
- Angular
- WordPress (40% of all websites)
- OpenOffice/LibreOffice
- MySQL
- Hadoop
- Kubernetes
- Godot Game Engine
Synopsys Analyzes Commercial Applications

- Annual Study of Proprietary Applications
  - 1250 audited applications in 17 industries
- Most recent reports
  - 2020 Open Source Security and Risk Analysis
  - DevSecOps Practices and Open Source Management 2020
- 99% of analyzed codebases contained open source software
FOSS in almost every codebase

High-risk vulnerabilities found in 49% of codebases; license conflicts found widely plus unlicensed and outdated code

Source: 2020 Open Source Security and Risk Analysis, Synopsys
Use of Security Tools

Which, if any, of the following security tools does your team currently use?

- **45%** Web application firewall
- **38%** Software composition analysis (SCA)
- **37%** Dynamic application security testing (DAST)
- **37%** Intrusion/detection protection system
- **34%** Runtime application self-protection (RASP)
- **33%** Static analysis security testing (SAST)
- **33%** Interactive application security testing (IAST)
- **27%** Penetration testing
- **23%** Protocol or API fuzzing
- **21%** Container security
- **7%** None of the above

Development organizations need to increase their use of security tools.

Source: DevSecOps Practices and Open Source Management 2020, Synopsys
The Role of Development Processes

• Some development process is better than none
• Agile methods, e.g., Scrum, provide insight into development progress
• Organizations release software without testing for runtime errors and application vulnerabilities
  – Be aware of CI/CD processes
Organizational Management of FOSS

Open Source Project Office
- Establishment of open source strategy
- Inventory of FOSS software
- Managing open source license compliance
- Engaging with developer community
- Contributing back to projects

https://todogroup.org/guides/create-program/
#the-role-of-the-open-source-program-office
Three takeaways

• FOSS use is not just mainstream, but nearly universal, and requires organizational governance

• Trustworthiness is independent of proprietary vs. open source code

• All software projects should analyze their code to reduce static code problems and other vulnerabilities prior to deployment