Enterprise-scale Computation Imaging

Charles Zhang
Cybersecurity Laboratory
X-Ray Imaging

• Tool for determining physical composition of objects
• Highly revolutionary
  • Pillar of modern medicine
  • Security and safety
  • Archeology
• Non-invasive therefore highly versatile
  • Human body + Animal + objects
Computation Imaging

• Tool for determining computational composition of software
• (Should be) Highly revolutionary
  • Health of software
  • Security and safety
  • Manufacturing governance
• Need to be non-invasive and highly versatile
  • Source code + bytecode + binary
• This is hardly a new idea, but not easy for enterprise-scale
Enterprise software: Big!
Enterprise software – Alive!

• Linux
  • 2021: 74,902 commits in 319 days ➔ 235 /day == 10 /hour
  • 2020: 90,421 commits in 366 days ➔ 247 /day == 10 /hour
  • 2019: 82,483 commits in 365 days ➔ 225 /day == 9 /hour

• Clang
  • 28,770 commits 319 day ➔ 90 /day == 4 /hour

• Tensorflow
  • 2021: 18,768 commits 319 days ➔ 59 /day == 2 /hour
Enterprise software – Mostly dark!

- Large-scale software supply chains (often in binary)
  - 15%-27% of code is third-party commercial software – so the source is often unavailable.
  - In-house supply chains across groups
- New trend: software evolves into cloud native
  - Amazon lambda deployment increases 200% in 2020\(^1\)
  - Code size including dependency < 25MB
- Most of the parts are “dark”
  - Not developed by you
  - Not directly examined (no source code, no documentation)

Figure: A graph showing the distribution of code origin for different classes of projects. Source “Software Assembly Practices Necessitate More Precautions” – VDC Research, 2016.

---

Enterprise software: Assembled control-flow!

• Componentized deployment
  • Serverless -- “trained developers to optimize Lambda functions for single, well-defined tasks with lower overall code sizes”
  • Micro-service -- “replace their large, cumbersome monolithic applications with microservices”

• Program dependencies cannot be locally reasoned
  • Assembled instead of self-contained
  • Inverted dependences through callbacks
  • Remote dependence via inter-process communication
  • Nobody knows how it works……
Enterprise-scale Computation Imaging

• High quality results:
  • Precise:
    • Balance between false and missing results
  • Fast:
    • Between editor feedback and nightly build

• Address the CODA challenges:
  • CONTINUOUS in time (incremental) and space (accumulative)
  • OPEN for customization through APIs and DSLs
  • Reasonable assumptions of the “DARK code”
  • Understanding of ASSEMBLED program dependency through callbacks or distributed computing

All need to be addressed simultaneously!
Introducing Clearblue Project

• To build open-source platform for non-invasive computation imaging technology

• Foundation
  • World-leading research results
  • Technology already commercialized and deployed in Huawei, Baidu, Alipay

• Goals
  • A general purpose language-based UI
  • A highly parallel and distributed composition analysis engine
  • A non-invasive software scanning apparatus based on binaries and texts