Graphene:

Securing unmodified Linux Applications with Confidential Computing

Prof Don Porter, UNC, Chapel Hill
Mona Vij, Principal Engineer, Intel

The Confidential Computing Consortium

- Community focused on Open Source licensed projects securing DATA IN USE and accelerating the adoption of confidential computing through open collaboration
- Announced the intent to form in August at the Open Source Summit North America in San Diego, formally launched on 17 October 2019 with governance in place

Please visit https://confidentialcomputing.io

CONFIDENTIAL COMPUTING CONSORTIUM

Premier

accenture

蚂蚁集团 ANT GROUP

arm

facebook

Google

HUAWEI

intel

Microsoft

Red Hat

General

AMD

anjuna

Anqlave

appliedprivacy

ByteDance

cosmian

CYSEC

decentriq

EDGELESS SYSTEMS

Fortanix

iExec

MADANA

NVIDIA

PHALA NETWORK

r3.

swisscom

vmware

XILINX

Associate

IoTeX

MIT Connection Science

OASIS LABS

CONFIDENTIAL COMPUTING CONSORTIUM

Don Porter
[he/him]
Associate Professor
Computer Science
UNC Chapel Hill

Mona Vij
[she/her]
Principal Engineer
Intel Labs

Jesse Schrater
[he/him]
Director of Data
Center Security
Marketing

Confidential Computing

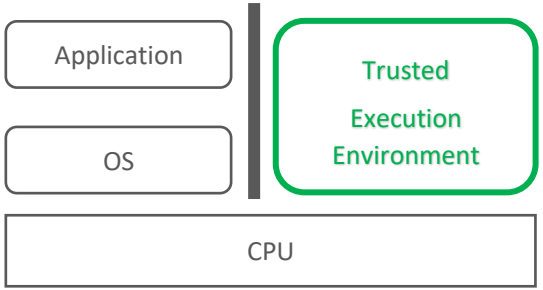
Confidential Computing Definition

Confidential Computing is the ***protection of data in use*** *by performing computation in a* ***hardware-based Trusted Execution Environment***

CONFIDENTIAL COMPUTING CONSORTIUM

Trusted Execution Environment (TEE)

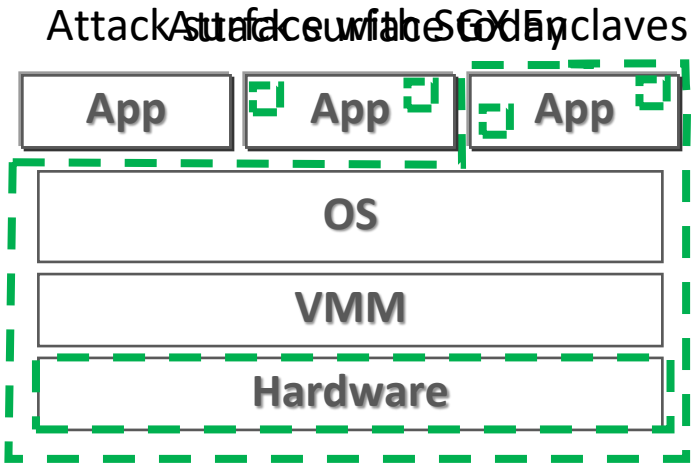
- A TEE is an environment that provides a level of assurance of three key properties
  - Data Confidentiality
  - Data Integrity
  - Code Integrity
- Hardware based TEE
  - Maps to a secure portion of memory
  - The code and data inside TEE can not be accessed from outside
  - Can prove to 3rd party its identity via attestation



- Most major hardware vendors support TEE
  - **Intel® SGX**, Intel TDX, AMD SEV, ARM TrustZone

Intel® Software Guard Extension (Intel® SGX)

- Intel® SGX provides a set of ISA extensions to build a protected enclave
- Application is split into untrusted app and trusted enclave
- Application builds trusted enclave with well-defined entry points
- Attest enclave fingerprint to 3rd parties for verification and secrets provisioning



Lift and Shift Unmodified Applications

- In un-trusted cloud and edge deployments, there is a strong desire to shield the whole application from rest of the infrastructure
- Developers want end-to-end secure solutions with “push-button” approach
- Graphene supports lift and shift paradigm for unmodified application binaries for CC with Intel SGX

Graphene Overview

Graphene Project Summary

- Technical acceptance by CCC - pending legal review for trademark/naming

- Graphene runs unmodified Linux Applications on other platforms
  - Current focus on Intel® SGX
- Community maintained Open-Source (LGPL) project hosted on Github
- Well defined testing and validation criteria with CI/CD (Jenkins)
- Project maintenance is governed via a well-defined governance criteria
- Cloud deployment with Azure Kubernetes Service
- Very actively being developed towards production readiness in Q2'21

Growing Community

AI/ML

Databases

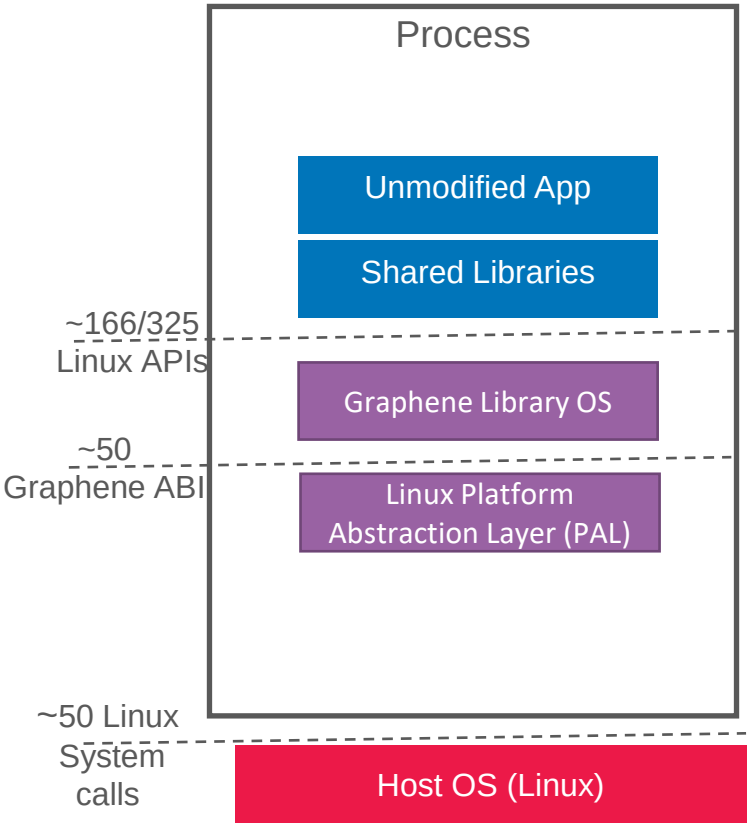
Web Servers

Languages

Misc

Graphene Architecture

Graphene Library OS



SUNY Stony Brook
Graphene [EuroSys'14]

Cooperation and Security Isolation of Library OSes for Multi-Process Applications

Chia-Che Tsai Kumar Saurabh Arora Nehal Bandi Bhushan Jain William Jannen
Jitin John Harry A. Kalodner† Vrushali Kulkarni Daniela Oliveira† Donald E. Porter
Stony Brook University †Bowdoin College
{chitsai,karora,nbandi,bpjain,wjannen,jijohn,vakulkarni,porter}@cs.stonybrook.edu
{hkalodne,doliveir}@bowdoin.edu

Intel Labs and SUNY Stony Brook
Graphene-SGX [ATC'17]

Graphene-SGX: A Practical Library OS for Unmodified Applications on SGX

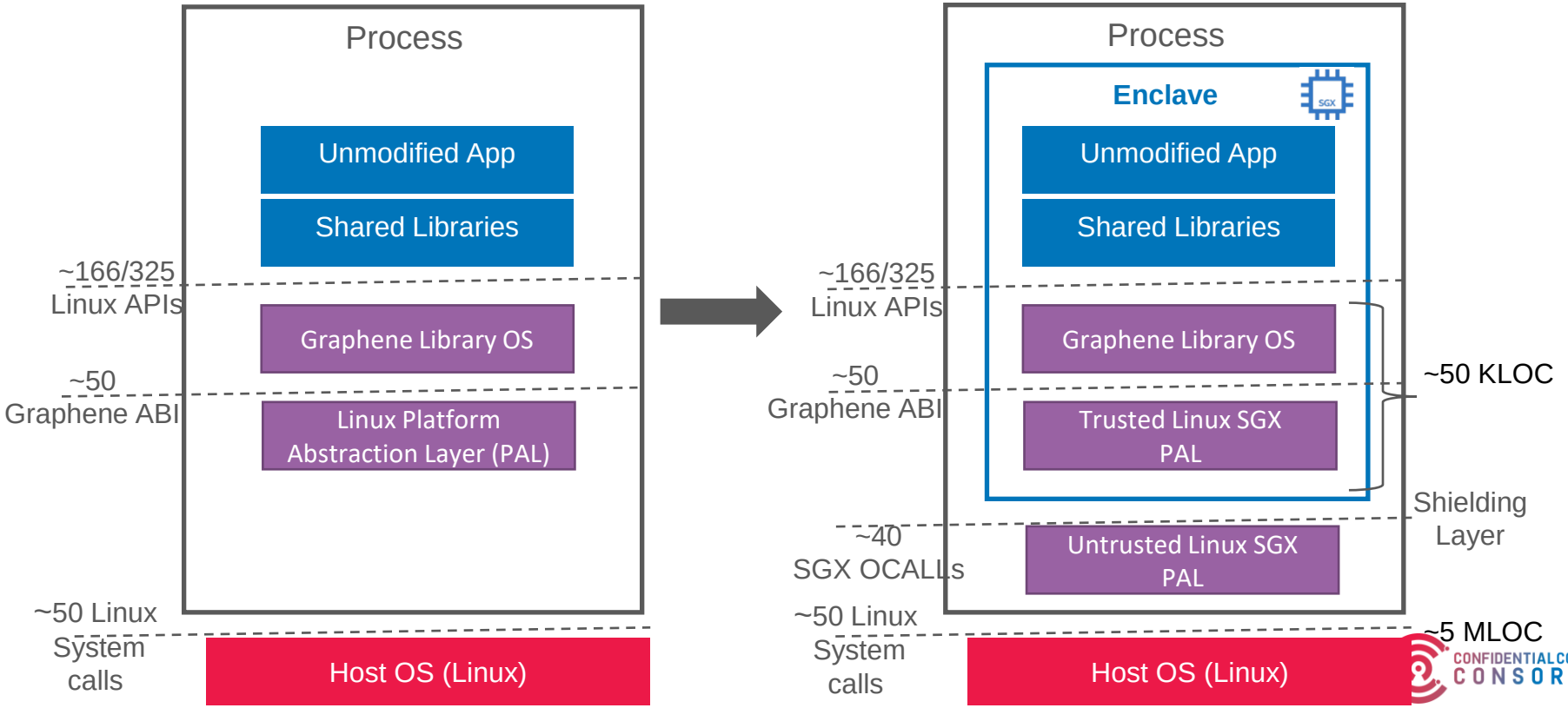
Chia-Che Tsai
Stony Brook University

Donald E. Porter
University of North Carolina at Chapel Hill and Fortanix

Mona Vij
Intel Corporation

CONFIDENTIAL COMPUTING CONSORTIUM

Library OS architecture is very suitable for Intel® SGX



15

Graphene Shielding Layer

- All security-critical paths are hardened against eavesdropping/attacks
  - Integrity of the loadable libraries is verified via checking against valid hash values
  - Transparent File encryption is supported via protected FS for “protected” files”
  - Migration of checkpointed state from parent to the child process is transparently encrypted for secure fork
  - All Pipes, messages and signals among Graphene processes are encrypted
  - Additional checks in SGX-PAL prevent against Iago attacks
- All network communication is assumed to be SSL/TLS-protected by the app itself

Enabling app in Graphene requires a manifest file

redis.manifest (TOML format)

```
libos.entrypoint = “file:/usr/bin/redis”
loader.env.LD_LIBRARY_PATH = “/lib”

fs.mount.lib.type = “chroot”
fs.mount.lib.path = “/lib”
fs.mount.lib.uri = “file:/graphene/Runtime”

sgx.enclave_size = “1024M”
sgx.thread_num = 8

sgx.trusted_files.libc = “file:/graphene/Runtime/libc.so.6”
…
```

Executable to load into SGX enclave

Environment variables are overwritten

Restrict mount points to a small subset of host-OS directories

SGX-specific limits like the maximum enclave size and number of threads

SGX-specific trusted files (securely hashed and verified at load time)

Graphene Features for SGX Infrastructure

- SGX Attestation
  - Supports both EPID and DCAP/ECDSA SGX attestations
- Protected Files
  - Automatically encrypt/decrypt specified files in the manifest
- Asynchronous System Calls
  - Exit-less support as a performance enhancement feature
- Multi-process support
  - Fork and secure comm between parent and child process via encrypted IPC
- Docker Integration
  - Automatically convert Docker images to Graphene images

Graphene Remote Attestation

Graphene Remote Attestation

There level approach to Attestation

- Remote/Local Attestation Support:
  - Exposed via `/dev/attestation` pseudo-filesystem
  - Integrates with multiple backends under the hood including Intel DCAP
- Secure Channel Establishment
  - Constructed using RA-TLS (Remote Attestation integrated with Transport Layer Security)
- Secret Provisioning
  - Built using secret provisioning libraries

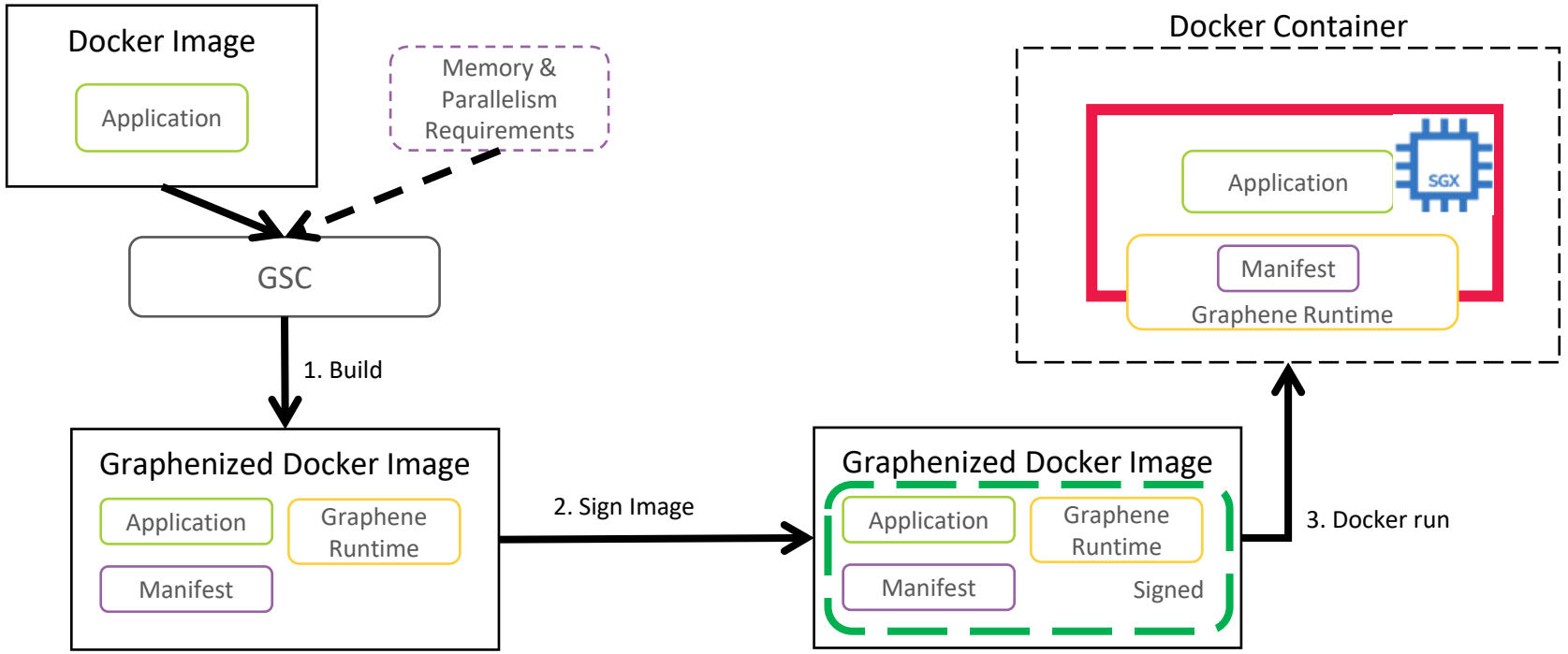
Docker Container Integration

Containers are deployment model for cloud

- Containers have become the defacto deployment model besides VMs
  - Small memory footprint
  - Fast startup

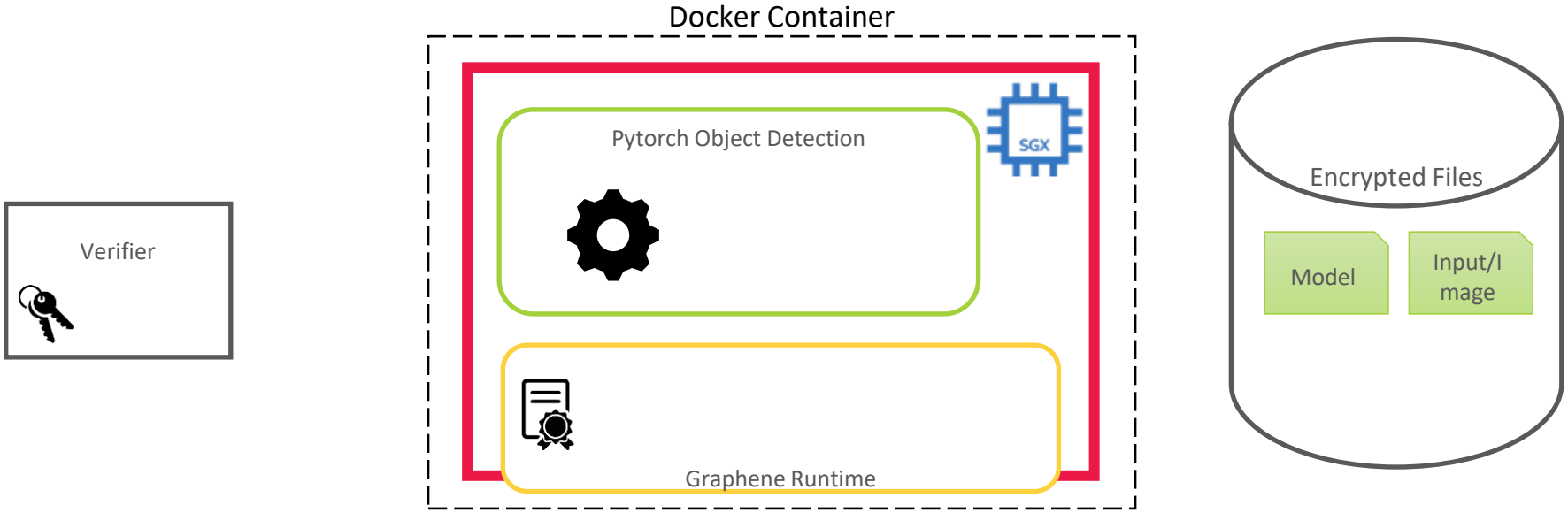
- Goals:
  - Provide an interface for automatically securing Docker container applications with SGX with Graphene
  - Minimize burden to craft application manifest

Graphene Shielded Containers (GSC) in a Nutshell

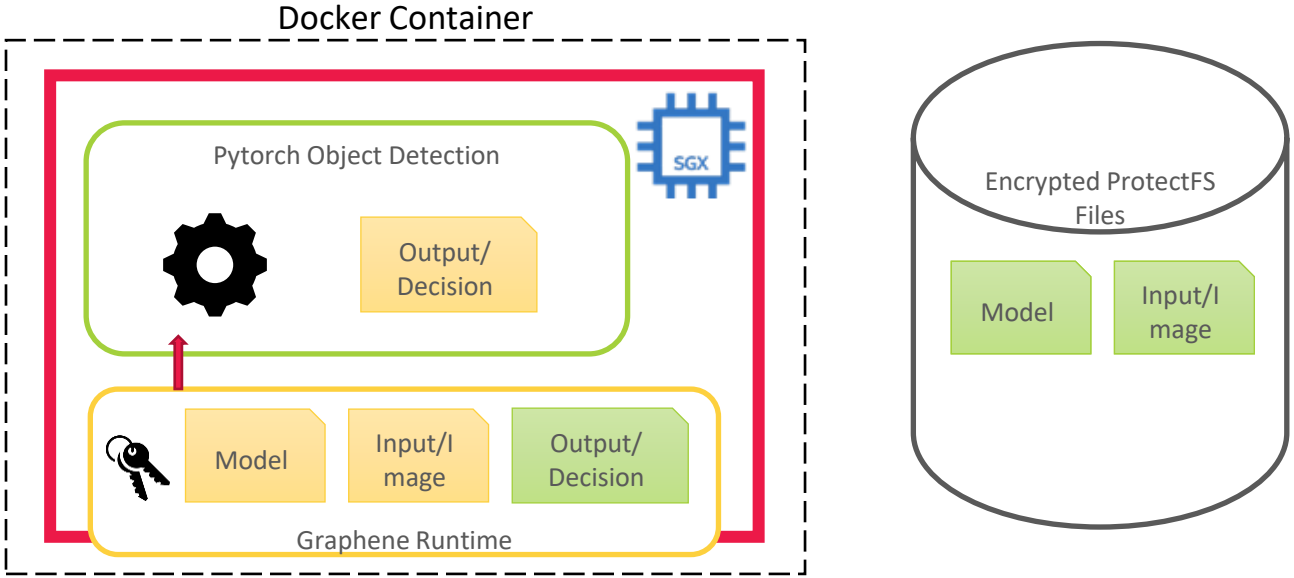


Securing an unmodified workload

End-to-End Secure Machine Learning with Pytorch



End-to-End Use Case using Pytorch



[PyTorch PPML Framework Tutorial]

CONFIDENTIAL COMPUTING CONSORTIUM

Graphene is actively evolving

- Initial SGX port released in 2017
- Open-source community established in Dec 2018
- First major release was in September 2019
- Difference between current version and that first release (~1.5 years):
    - 1,179 files changed
    - 87,890 insertions, 86,287 deletions
    - 1,143 commits from 34 authors
    - Memory management, threads, processes, signals 99% reworked

Graphene Project Future Plans

- Continue the development towards production readiness in Q2’21
- Improve tooling and documentation
- Integration with future cloud-based container deployments
- Grow supported runtimes and workloads
- Extend to secure communication with accelerators
- Explore addition of future TEE backends

Join our community!

- We welcome contributions of all type from the community!
  - Code
  - Issues
  - Documentation
  - Fruit baskets…
- Try deploying your unmodified Docker workloads with Graphene
  - In your datacenter or cloud
- We want to hear from your experience of using Graphene
  - Email or github issues welcome

Graphene project:
http://www.grapheneproject.io

GitHub repo:
https://github.com/oscarlab/graphene

Graphene Documentation:
https://graphene.readthedocs.io/en/latest/

CONFIDENTIAL COMPUTING CONSORTIUM

@ConfidentialC2
confidential-computing
confidentialcomputing.io

CONFIDENTIAL COMPUTING
CONSORTIUM