Exploration on Confidential Computing for Big Data & AI

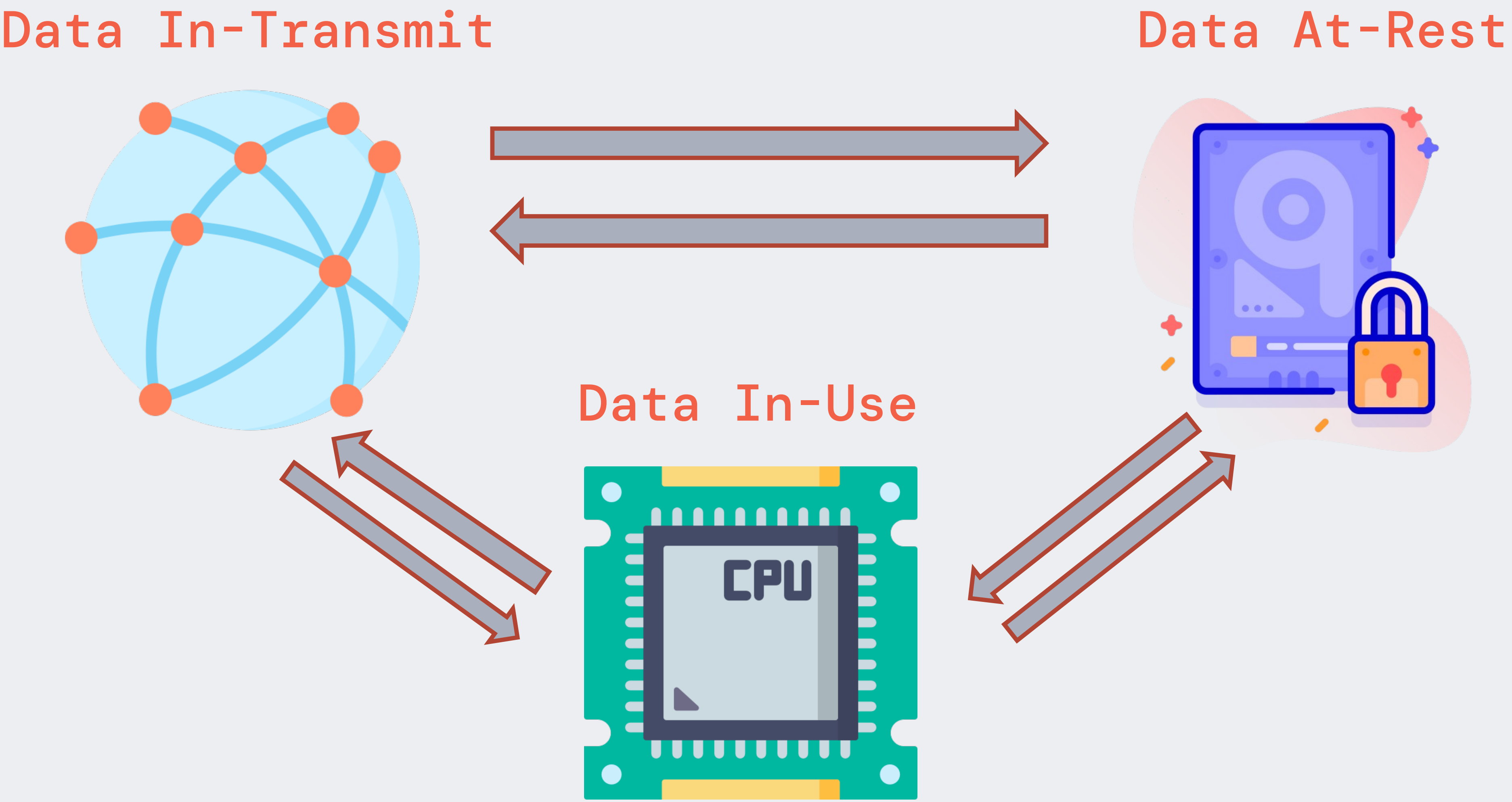
Chunyang Hui

Senior Engineer, Ant Group

Qiyuan Gong

Software Arch, Intel

States of Digital Data



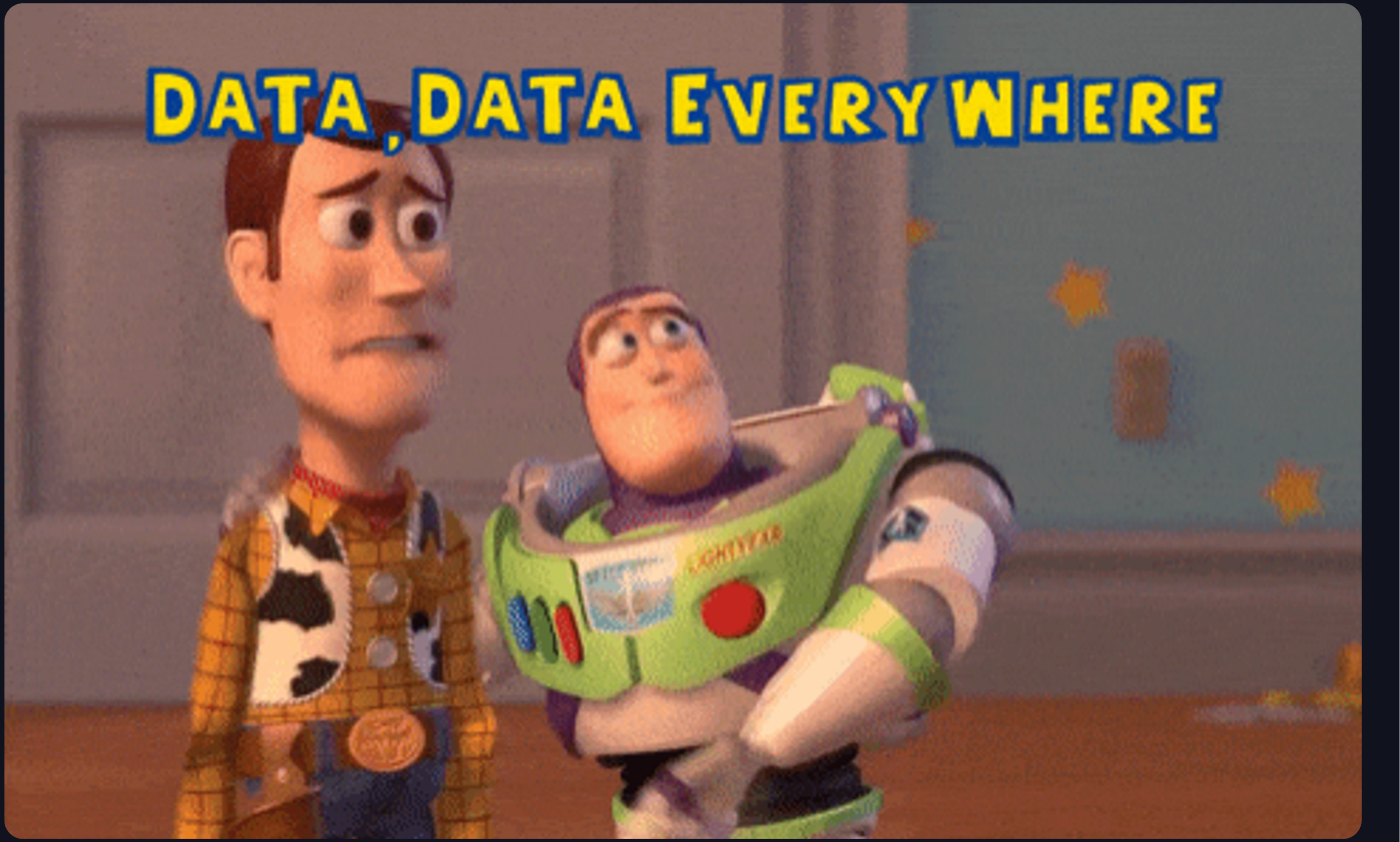
Data In-Transmit

Data At-Rest

Data In-Use

Confidential Computing

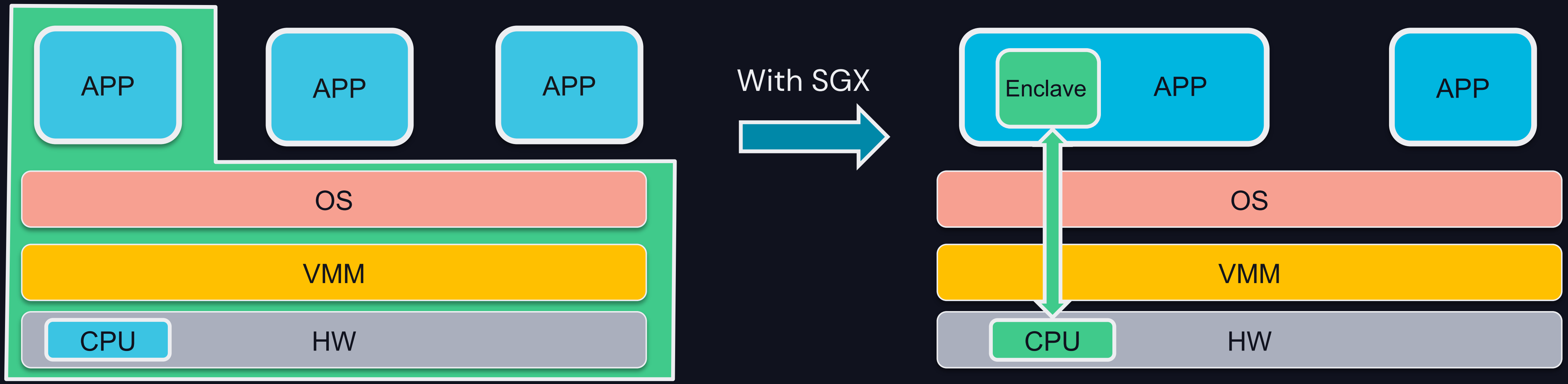
- Using hardware-based Trusted Execution Environments (TEE)
- Protect data in-use for data integrity, data confidentiality
- Only need to trust the hardware, small trusted computing base (TCB)
- Verifiable with Attestation



Intel® Software Guard Extension

An implementation of TEE technology

- mature, widely-used, protect users' sensitive data
- A set of CPU instructions to create and manage the hardware-protected memory (Enclave)
- Reduce the TCB to CPU + Enclave



Use Cases

Key Management Service

Keep private key protected in the Enclave

Multi-Party Computing

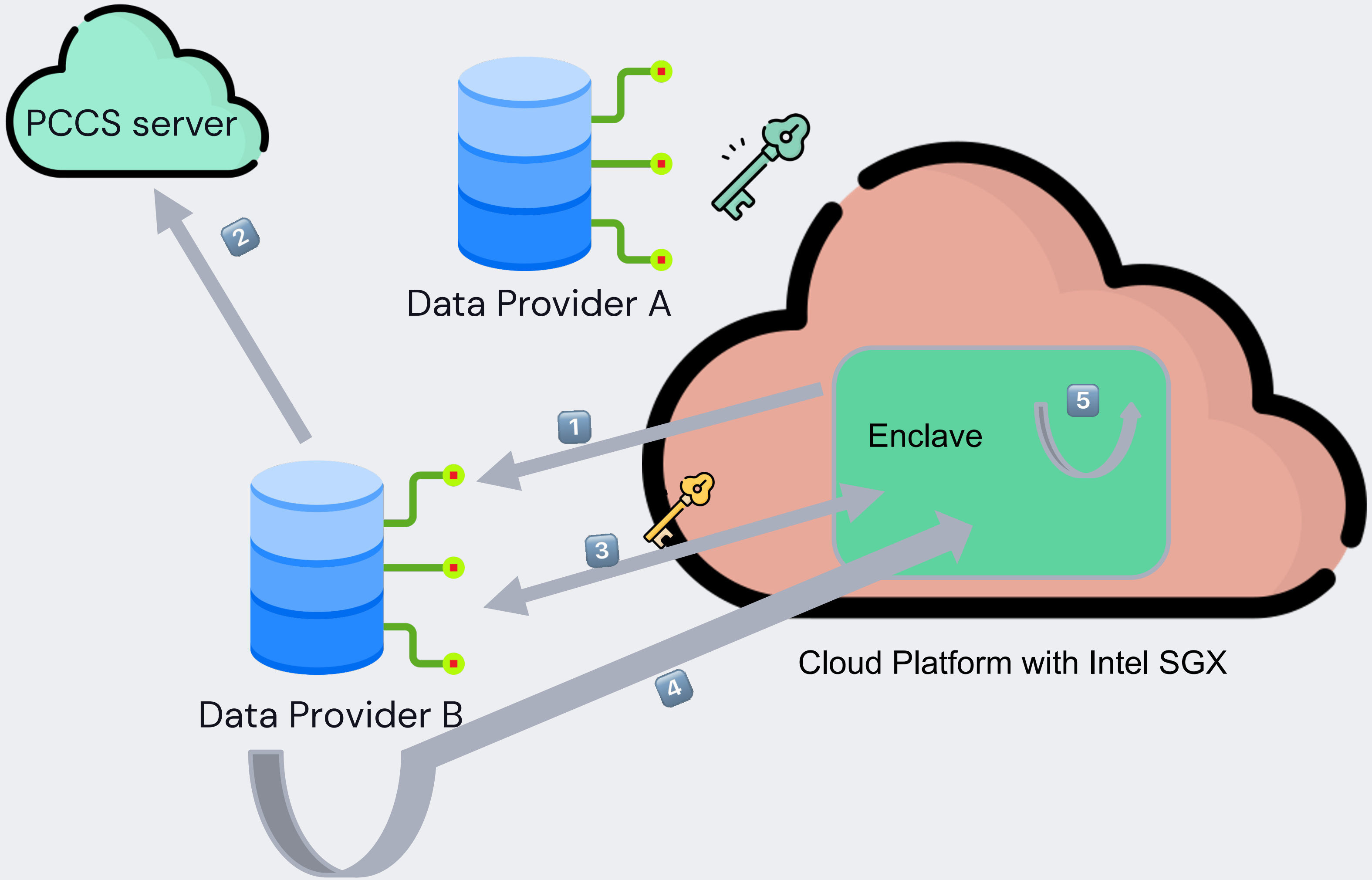
Keep private data protected while collaboration

Public Cloud Deployment

Keep critical app secure

Example

Multiple data holders train model on public cloud with TEE capabilities



Data Provider C

1. Get quote from the enclave
2. Verify the quote (EPID, DCAP)
3. Negotiate the key
4. Encrypt data and send to enclave
5. Decrypt the data and start training

These slides are open-source and may be freely distributed and copied

SGX SDK vs. Library OS

SGX SDK	Library OS
Re-Design Ring 3, no OS access Trusted, untrusted	Almost Full OS Accessibility
Re-Engineering Code change	No Code Change
Re-Compilation Extra SGX dependencies	No recompilation

Occlum

Empowering everyone to run every app in enclaves

Empowering Everyone to run every app in enclaves

- *Occlum: Secure and Efficient Multitasking Inside a Single Enclave of Intel SGX (ASPLOS' 20)*
- Created by Ant Group in 2019
- Donated to CCC (Confidential Computing Consortium of Linux Foundation) in 2021
- https://github.com/occlum/occlum

THE LINUX FOUNDATION PROJECTS

ANT GROUP

CONFIDENTIAL COMPUTING CONSORTIUM

Key Features

Efficient Multi-tasking

- Single-address-space architecture
- Multiple processes share the same enclave
- Super fast process startup and IPC

Memory Safety

- First SGX LibOS written in Rust
- Rust is designed to be memory safe. It does not permit null pointers, dangling pointers, or data races

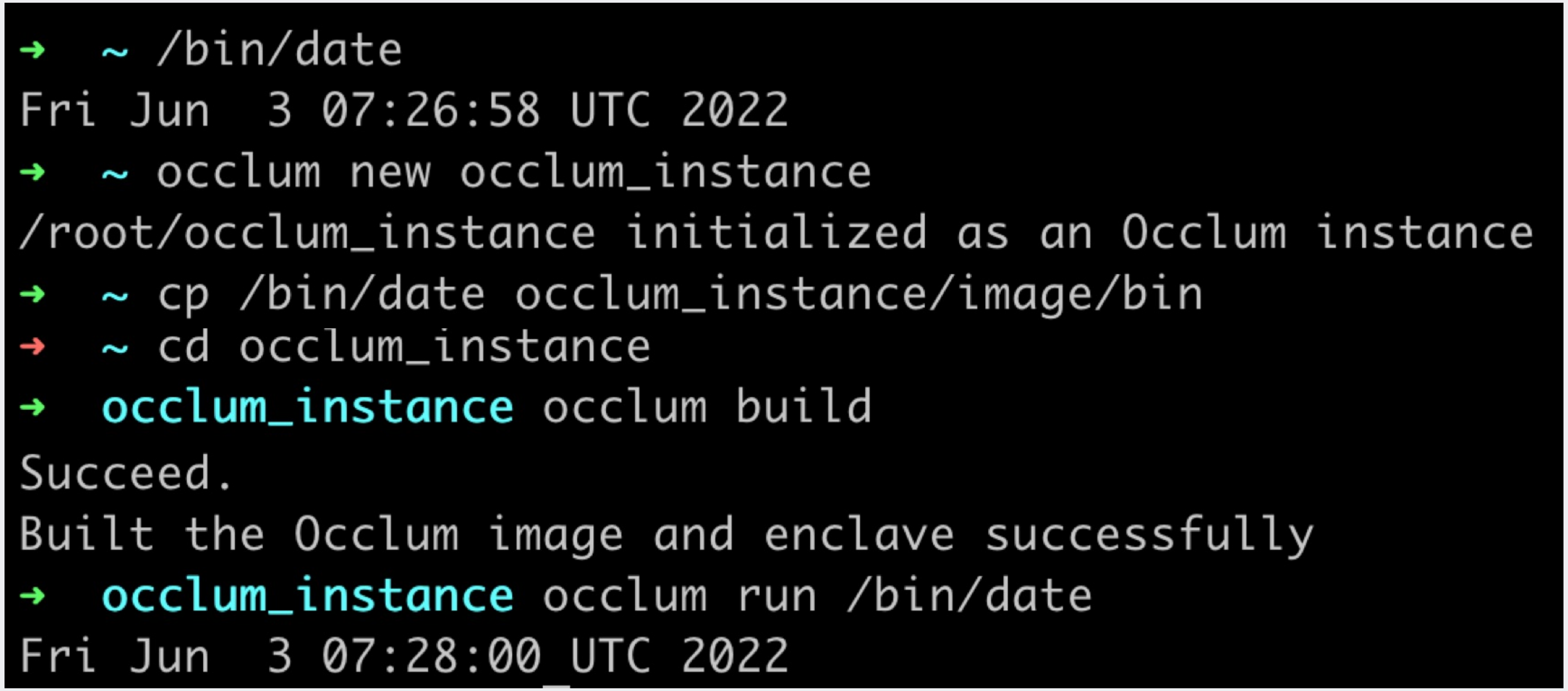
Ease of Use

- Empowering everyone to run apps in Enclave
- Similar user commands with Docker

Occlum Commands

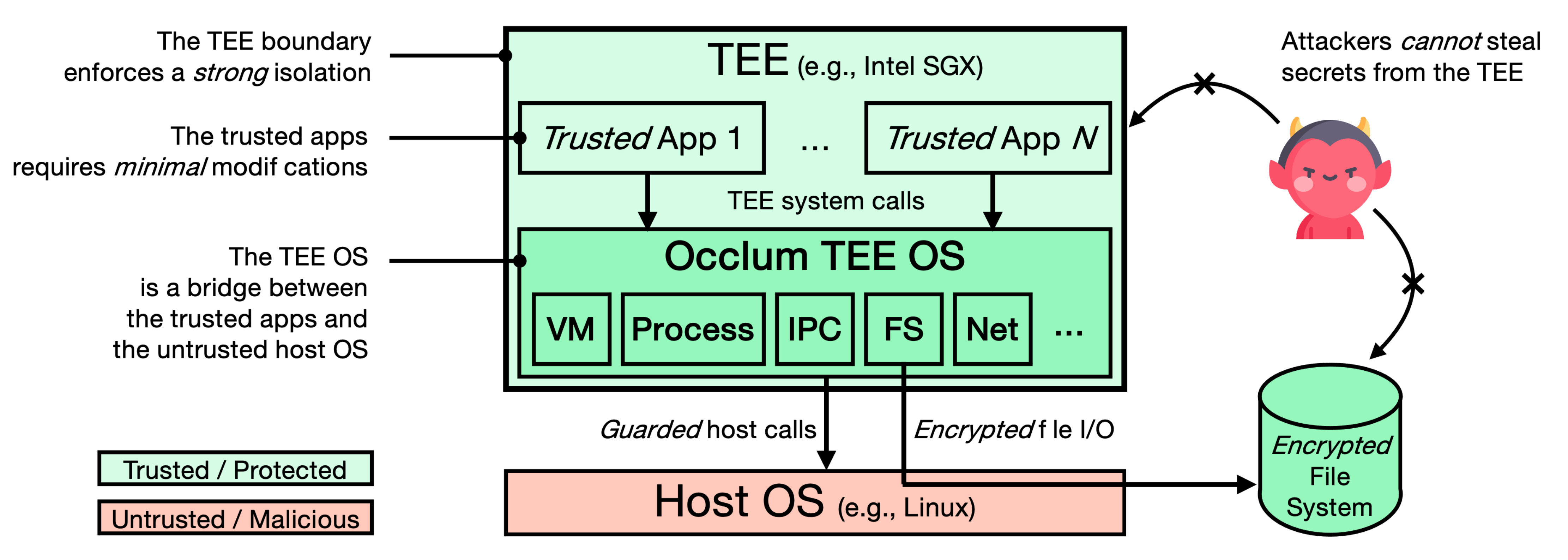
Ease of Use

- occlum new/init
- occlum build
- occlum run
- occlum start/exec



```
➜  ~ /bin/date
Fri Jun  3 07:26:58 UTC 2022
➜  ~ occlum new occlum_instance
/root/occlum_instance initialized as an Occlum instance
➜  ~ cp /bin/date occlum_instance/image/bin
➜  ~ cd occlum_instance
➜  occlum_instance occlum build
Succeed.
Built the Occlum image and enclave successfully
➜  occlum_instance occlum run /bin/date
Fri Jun  3 07:28:00 UTC 2022
```

Architecture



https://github.com/occlum/occlum

Use Cases

https://github.com/occlum/occlum/tree/master/demos

Programming Language
C/C++
JAVA
Python
Go
Rust
Shell Script (Bash, Fish)
…

Popular Applications
OpenVINO
PyTorch
Flink
Redis
SQLite
Vault
…

oneAPI Deep Neural Network Library (oneDNN)

Collaboration

Who is using Occlum

Azure

Alibaba Cloud

ANT GROUP

intel

EDGELESS SYSTEMS

[1] Azure: https://docs.microsoft.com/en-us/azure/confidential-computing/confidential-containers#occlum
[2] Alibaba Cloud: https://www.alibabacloud.com/blog/inclavare-confidential-computing-container-technology-for-cloud-native_596708
[3] Edgeless System: https://blog.edgeless.systems/marblerun-now-supports-occlum-even-more-confidential-computing-at-scale-2f6dd17e00c0
[4] Intel: https://community.intel.com/t5/Blogs/Tech-Innovation/Artificial-Intelligence-AI/Better-Together-Privacy-Preserving-Machine-Learning-Powered-by/post/1335716
[5] Ant: https://www.mo4tech.com/sofaenclave-the-next-generation-trusted-programming-environment-of-ant-financial-enables-confidential-computing-to-protect-financial-business-for-102-years.html

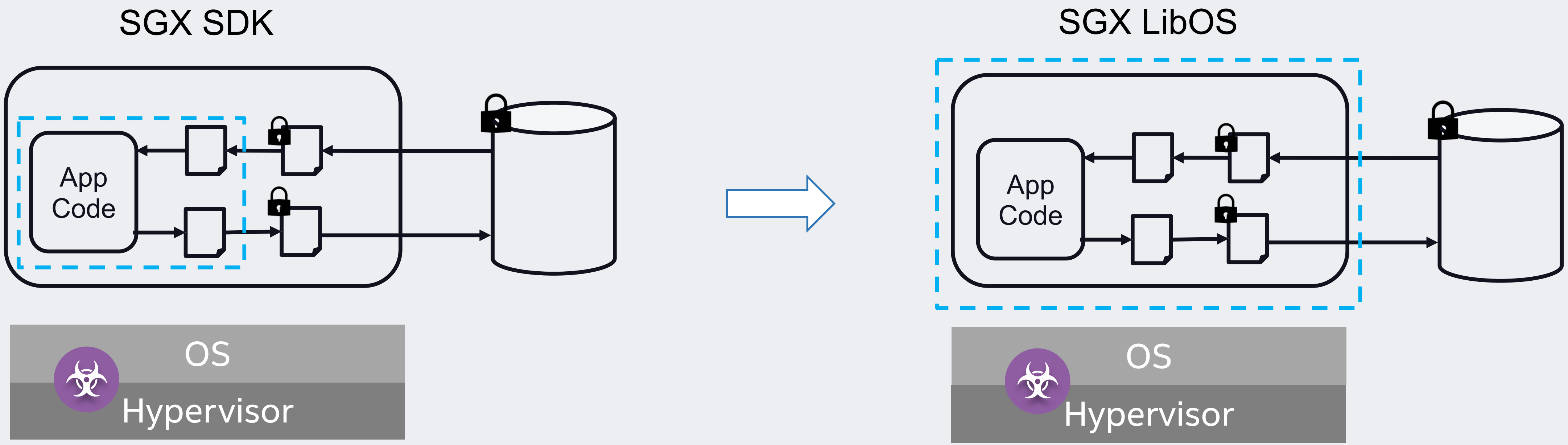
Future Work

- Add SGX EDMM support for higher memory performance

- Polish Next-Gen Occlum (NGO: https://github.com/occlum/ngo) for best performance and stability
  - Rust Async/Await
  - Linux io_uring

- Support a long list of frequently-used applications

SGX LibOS Secure Computation



Protect sensitive modules & plain text

Less SGX EPC requirement
Need to change app design

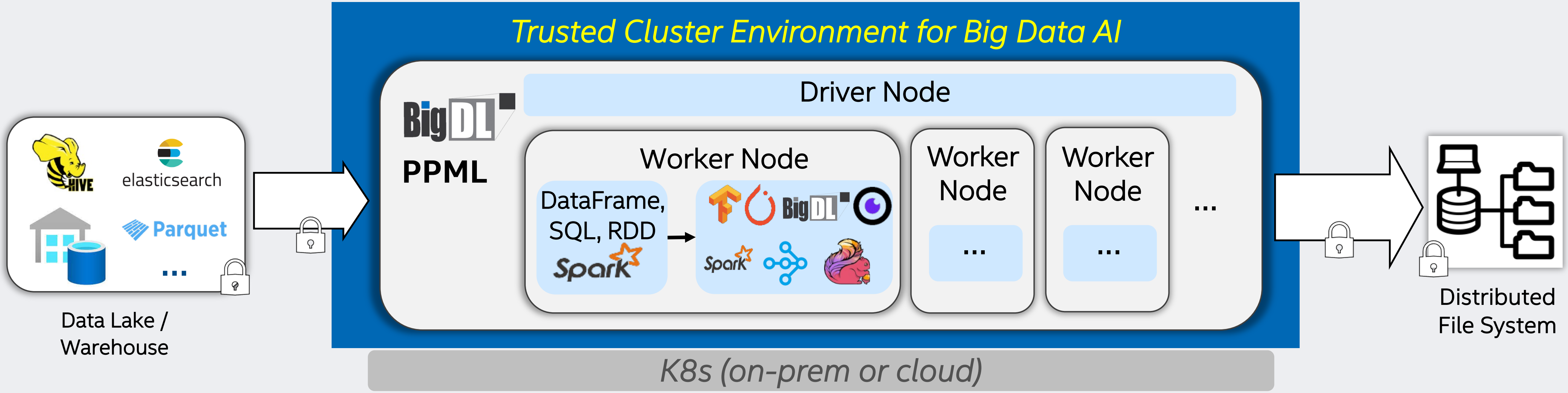
Protect entire apps

More SGX EPC requirement
Don’t have to change app design

⬚ Running in SGX

ML & Big Data Analytics in Privacy Way

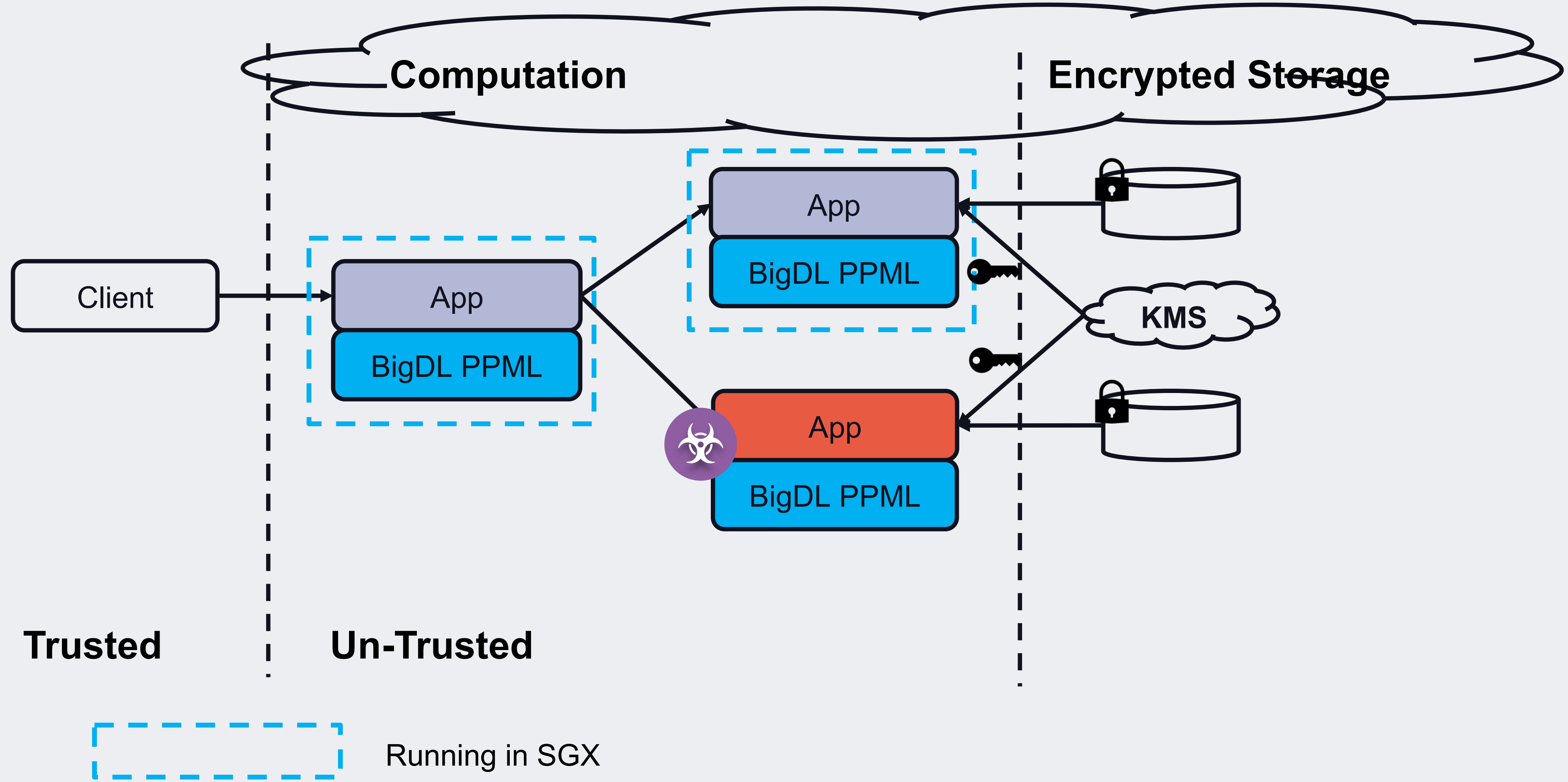
Secure & Trusted Big Data and AI, even on Untrusted env



- Standard, distributed AI applications on encrypted data
- Hardware (Intel SGX/TDX) protected computation (and memory)
- End-to-end security enabled for the entire workflow

Powered by oneAPI

Attack on distributed applications

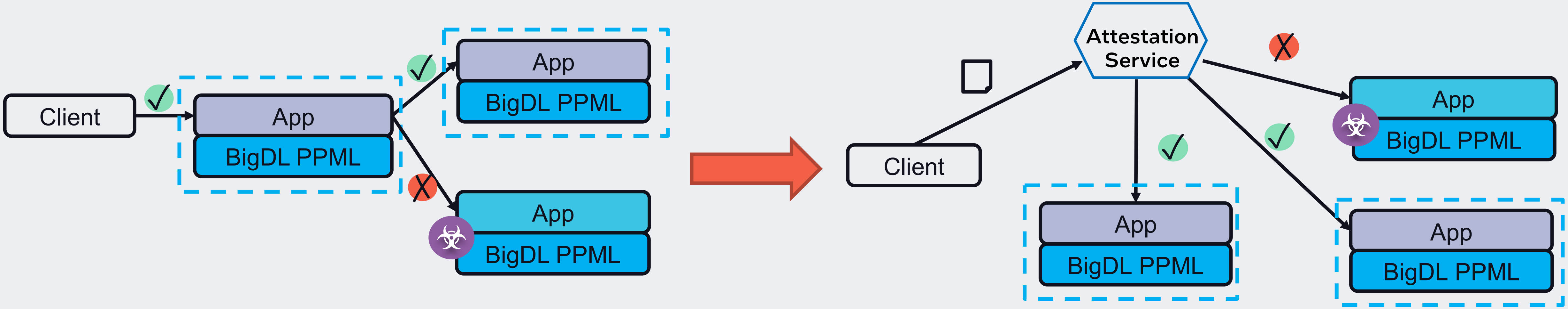


Ensure Integrity with SGX Attestation

Security & Privacy in E2E is never an easy job

P2P or leveled Attestation

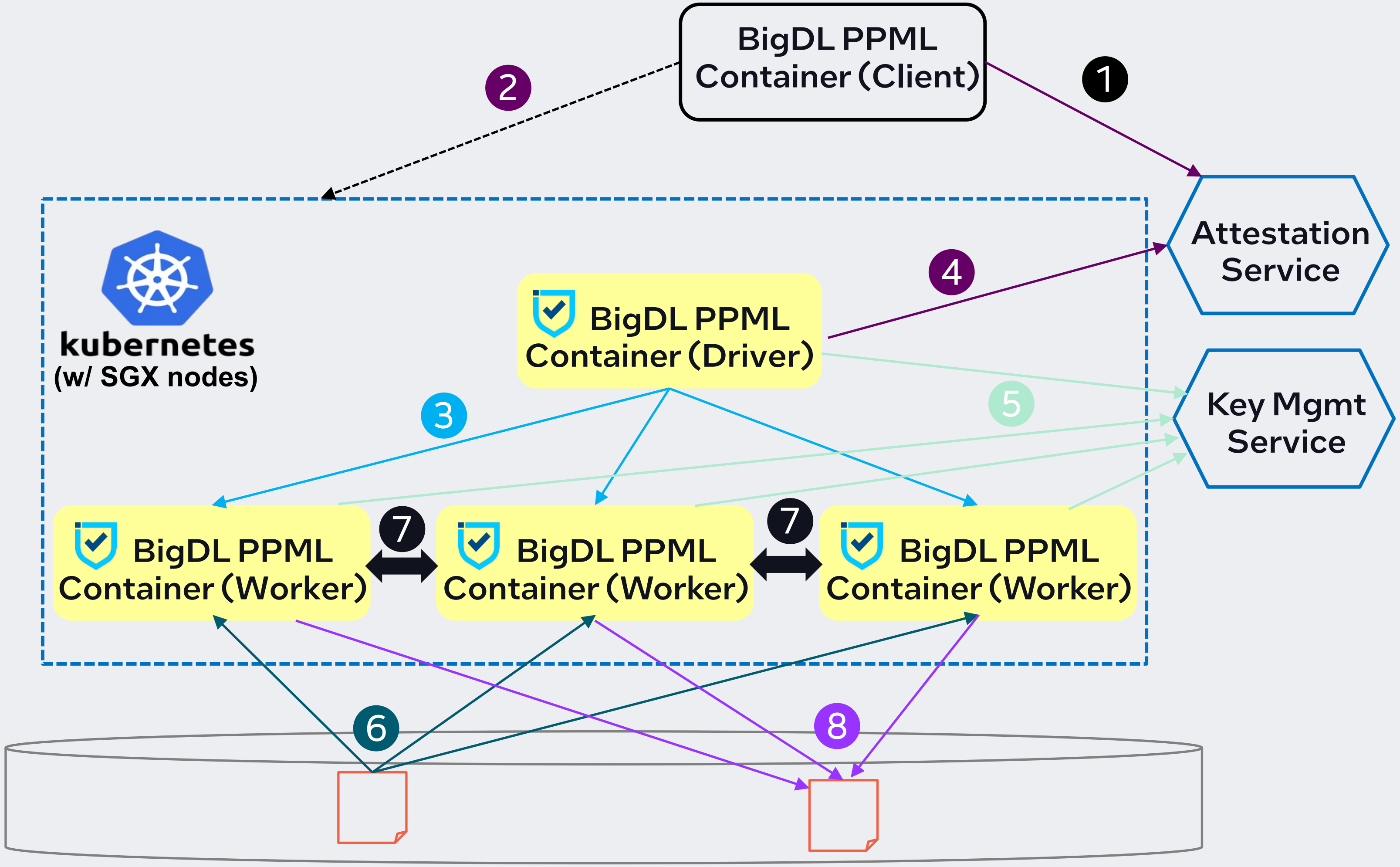
Centralized Attestation



✓ Looks Good (Pass attestation)

✗ Not Good (Fail on attestation)

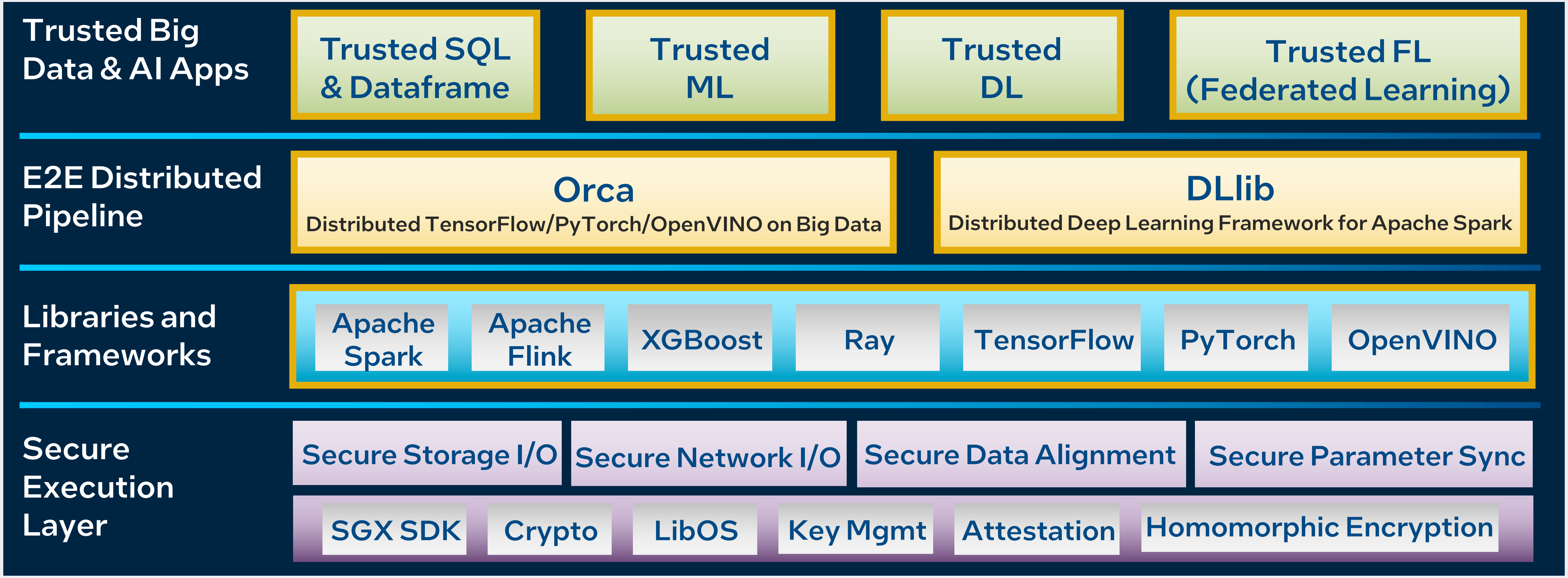
End-to-End Architecture of BigDL PPML



1. User submits Policy
2. User submits job to K8s (using BigDL PPML CLI), which creates the driver node
3. Driver creates more worker nodes
4. AS attests Driver/executor
5. Driver and workers request keys from KMS
6. Workers read and decrypt input data
7. Workers run distributed Big Data, ML and DL programs
8. Workers encrypt and write output data

BigDL PPML (Privacy Preserving ML)

Secure, Trusted Big Data and AI, even on Untrusted Cloud (using SGX)



Trusted Big Data & AI Apps

- Trusted SQL & Dataframe
- Trusted ML
- Trusted DL
- Trusted FL (Federated Learning)

E2E Distributed Pipeline

- Orca: Distributed TensorFlow/PyTorch/OpenVINO on Big Data
- DLlib: Distributed Deep Learning Framework for Apache Spark

Libraries and Frameworks

- Apache Spark
- Apache Flink
- XGBoost
- Ray
- TensorFlow
- PyTorch
- OpenVINO

Secure Execution Layer

- Secure Storage I/O
- Secure Network I/O
- Secure Data Alignment
- Secure Parameter Sync
- SGX SDK
- Crypto
- LibOS
- Key Mgmt
- Attestation
- Homomorphic Encryption

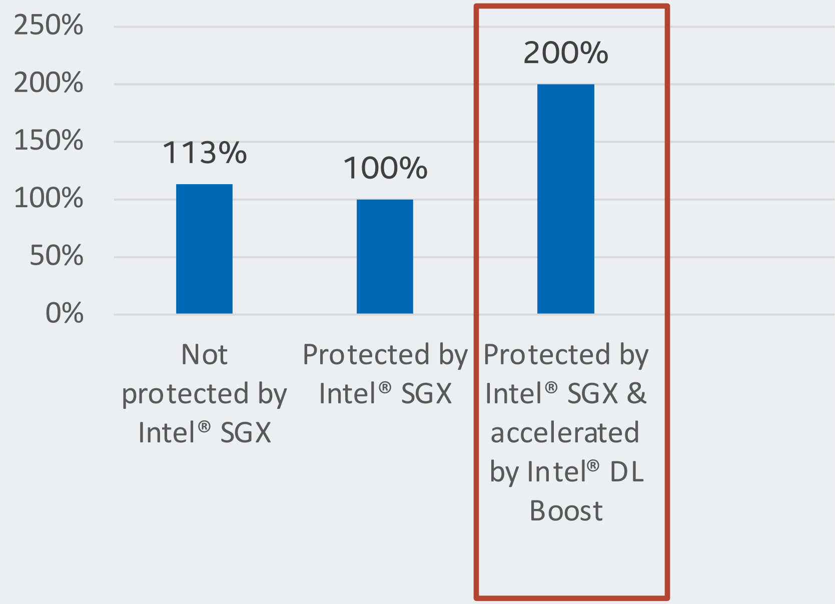
Intel SGX on kubernetes

Powered by oneAPI

Privacy Preserving Model Serving

Distributed & Secured Big Data and ML/DL Pipelines

BigDL PPML Inference Pipeline Performance



Application

- Secure & distributed inference solution build with BigDL, protected by Intel® SGX 2.0 and Occlum, and accelerated by Intel® DL Boost

Benefit

- The end-to-end distributed inference pipeline is protected by Intel® SGX 2.0 and Occlum (backed by Ant Group)
- 2.1X better inference throughput using Intel® DL Boost with Int8 compared to fp32

Performance Drivers

- Intel® DL Boost with Int8
- **oneAPI Deep Neural Network Library (oneDNN)**

At a Glance

Intel Architecture + Adjacencies

3rd Gen Intel® Xeon® Scalable Processor

Feature Enabling

Intel® SGX 2.0

Intel® DL Boost (Int8)

Intel Software Tools/Libraries

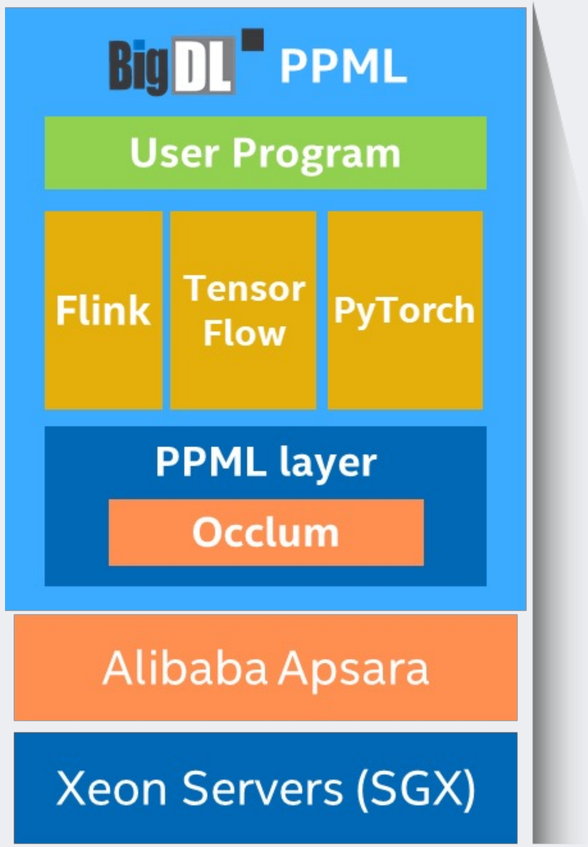
BigDL

oneAPI Deep Neural Network Library

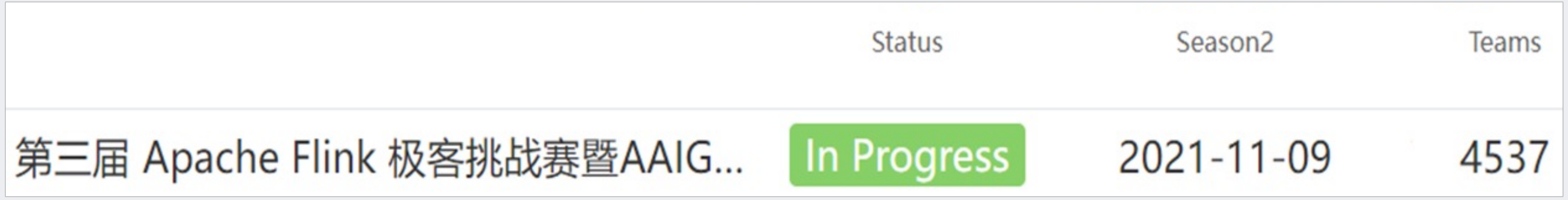
https://www.intel.com/content/dam/www/central-libraries/us/en/documents/alibaba-ppml-ai-blog-pdf.pdf

Privacy Preserving ML in Alibaba

Distributed & Secured Big Data and ML/DL Pipelines BigDL



(-)阿里云 / intel / Apache Flink® / AAIG / Occlum
电商推荐"抱大腿"攻击识别
第三届 Apache Flink 极客挑战赛暨AAIG CUP
Apache Flink®
Alibaba Cloud



	Status	Season2	Teams
第三届 Apache Flink 极客挑战赛暨AAIG...	In Progress	2021-11-09	4537

Alibaba, Intel and Occlum community co-host Kaggle-like PPML competition for spam detection in online e-commence recommendation.

4500 Teams

Building PPML Applications

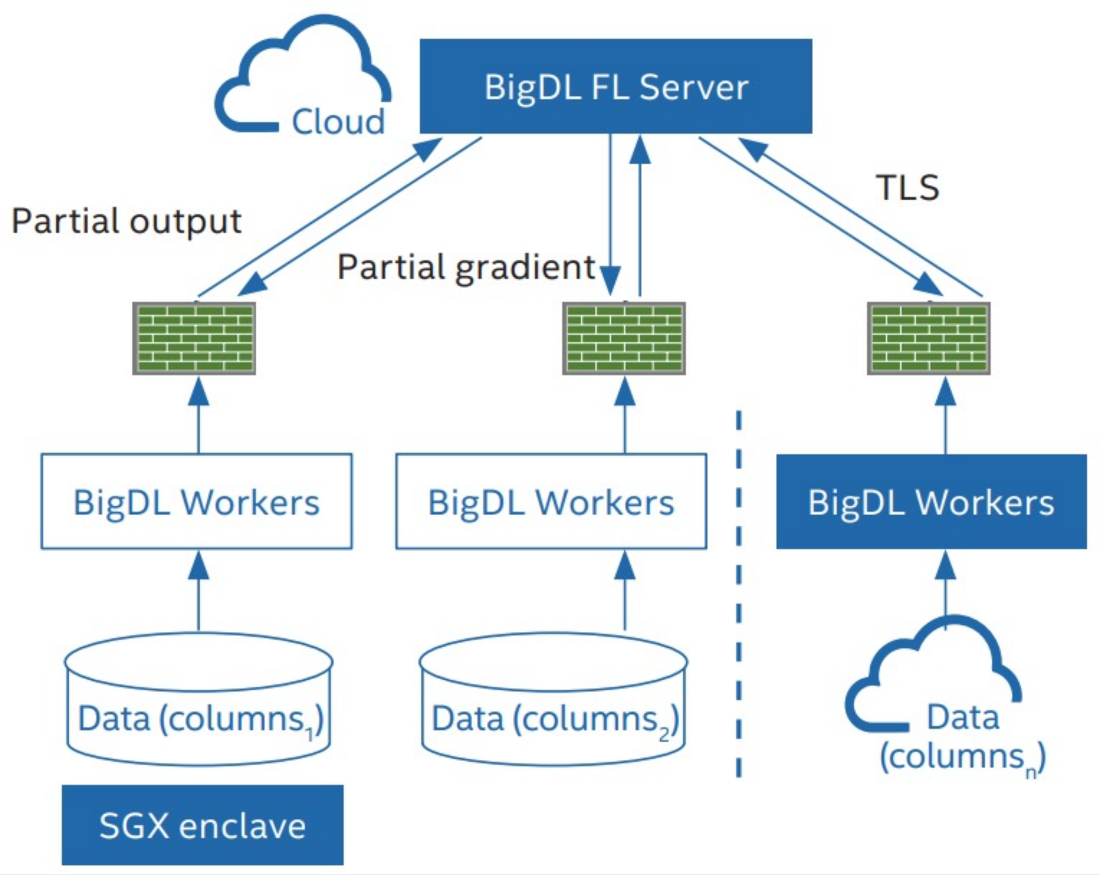
100+ IceLake Instances

Deployed on Alibaba Cloud

https://tianchi.aliyun.com/competition/entrance/531925/introduction

Trusted Federated Learning in Finance

Distributed & Secured Big Data and ML/DL Pipelines



Trusted Federated Learning

- Build united model across different parities
    - Training data remain local
    - Aggregation temp/partial results
- Secured computation environment with SGX

Win-Win for all parties

- End users
- Enterprises
- Cloud Service providers

Thank you

Hui, Chunyang

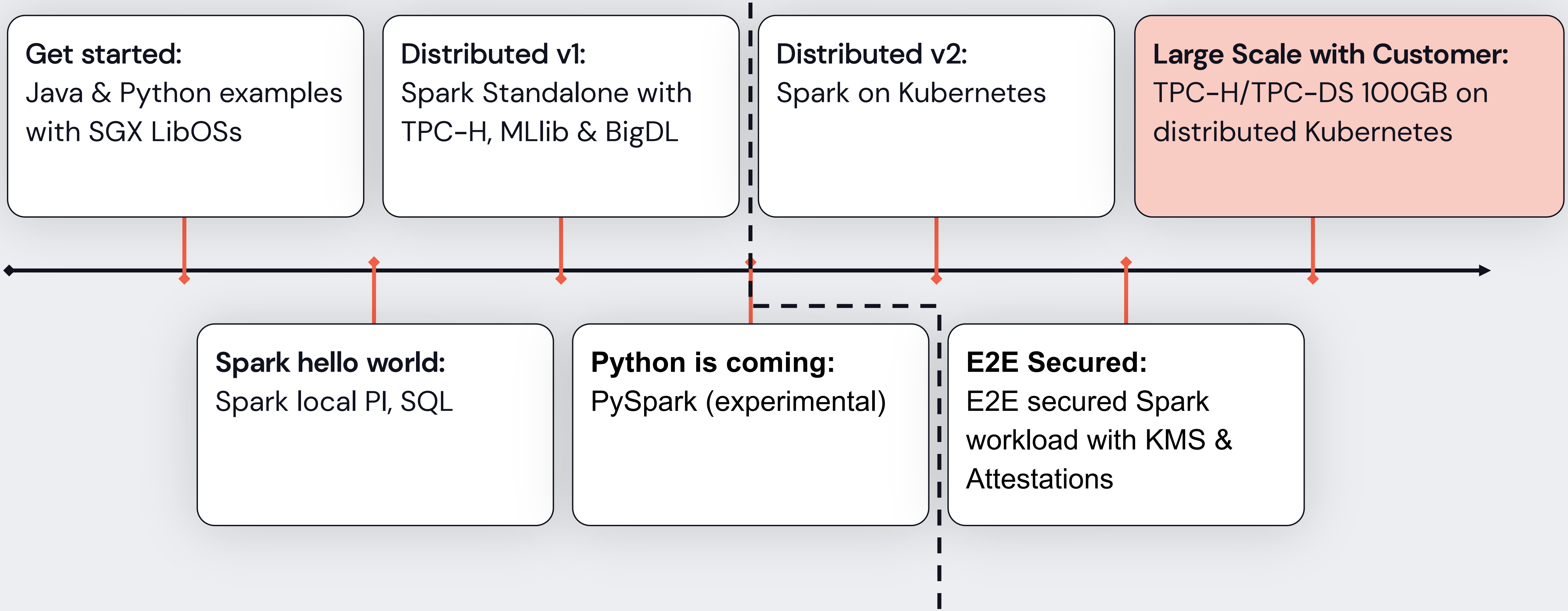
Senior Engineer, Ant Group

Qiyuan Gong

Software Arch, Intel

Timeline: Put Apache Spark in SGX

A long and exiting Journey of BigDL PPML



We are here!

Get started:
Java & Python examples with SGX LibOSs

Spark hello world:
Spark local PI, SQL

Distributed v1:
Spark Standalone with TPC-H, MLlib & BigDL

Python is coming:
PySpark (experimental)

Distributed v2:
Spark on Kubernetes

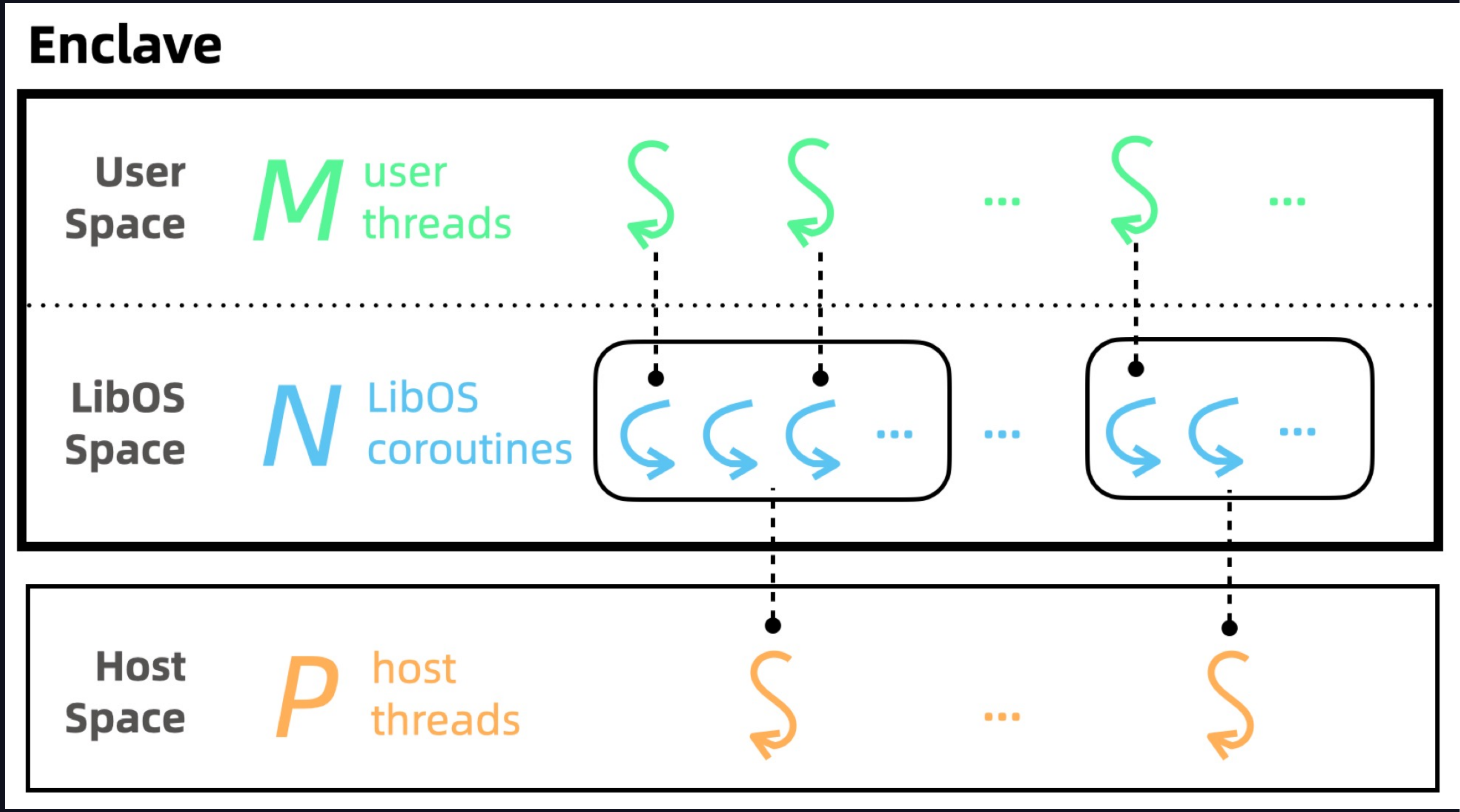
E2E Secured:
E2E secured Spark workload with KMS & Attestations

Large Scale with Customer:
TPC-H/TPC-DS 100GB on distributed Kubernetes

Next Generation Occlum

In-Enclave Scheduling

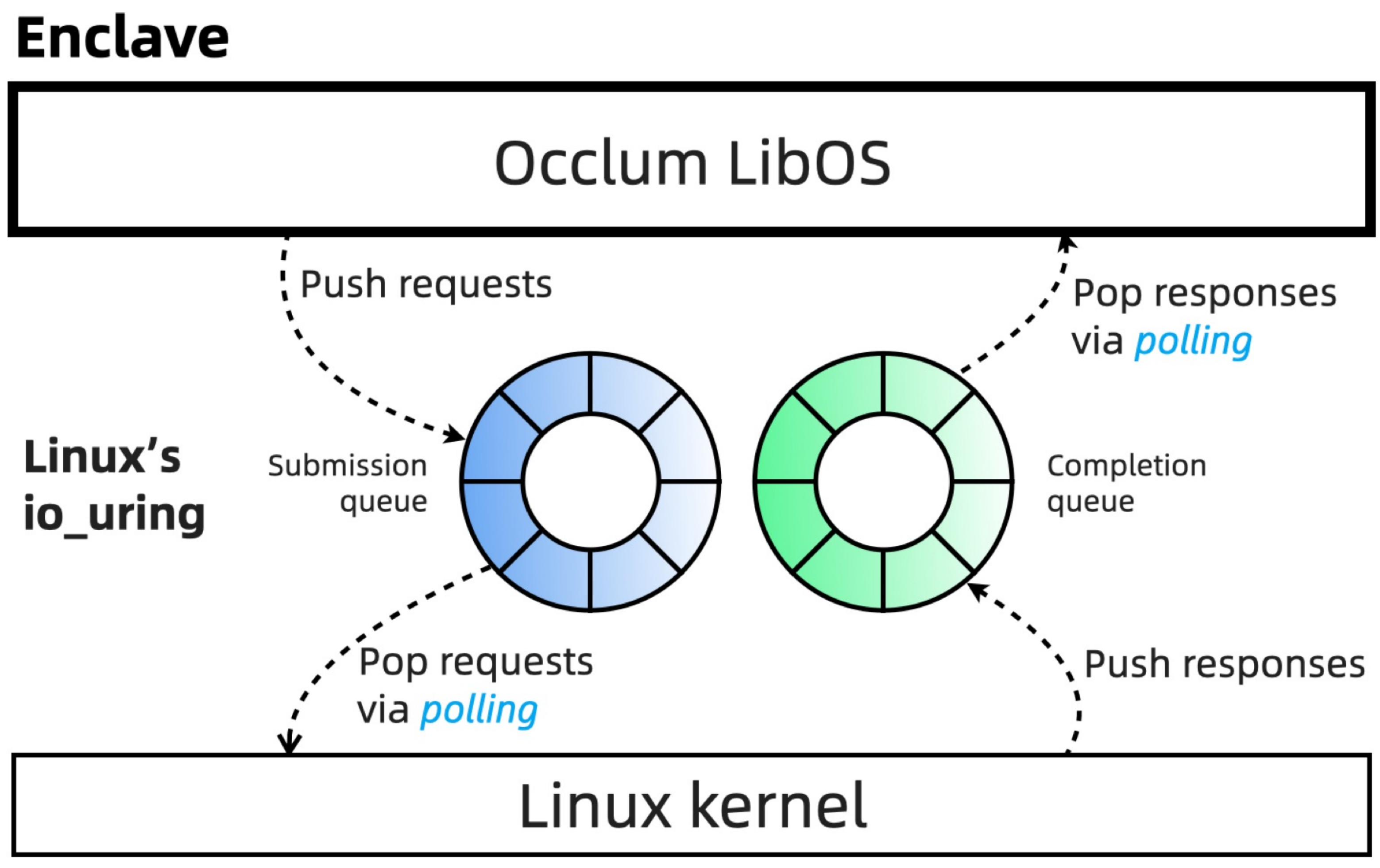
- Coroutine based
- Supports tons of user threads



Next Generation Occlum

Switchless Async IO

- Based on Linux io_uring
- Two ring buffers shared by the kernel and applications
- Very efficient for large IO throughput


Enclave
Occlum LibOS
Push requests
Pop responses via polling
Linux's io_uring
Submission queue
Completion queue
Pop requests via polling
Push responses
Linux kernel