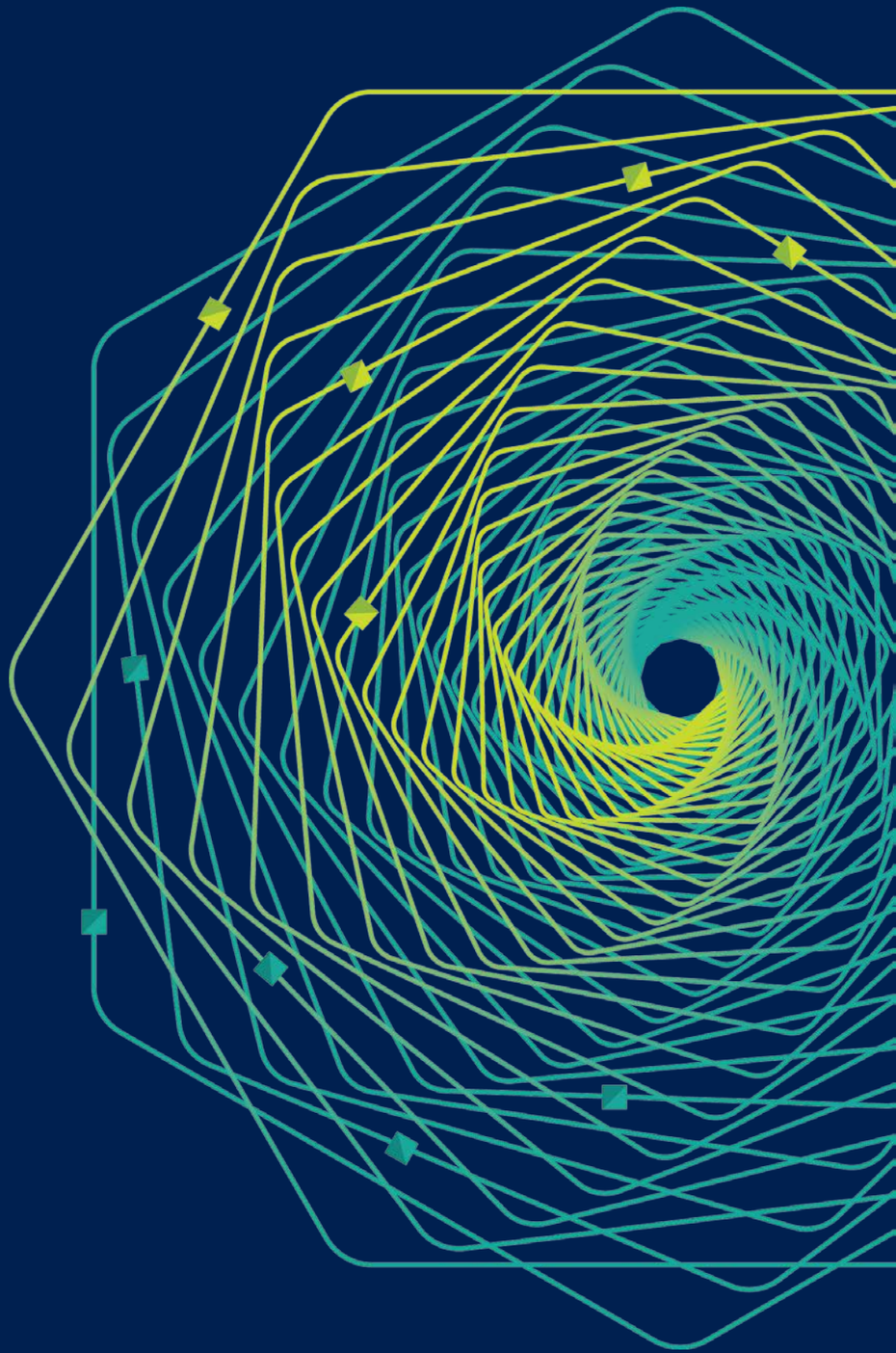
Microsoft

Research Faculty Summit 2018

Systems | Fueling future disruptions



Towards secure, practical confidential computing with open source secure enclave

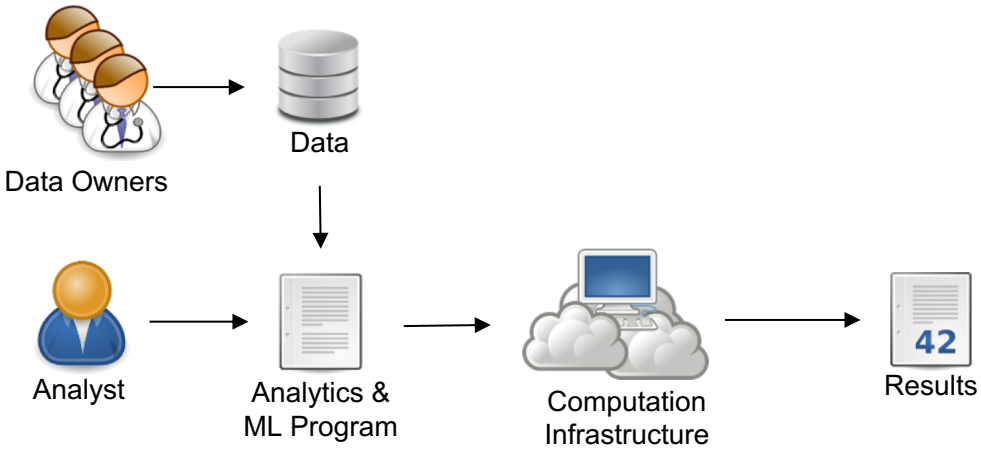
Dawn Song
Professor, UC Berkeley
Founder and CEO, Oasis Labs

The Value of Data Analytics and Machine Learning

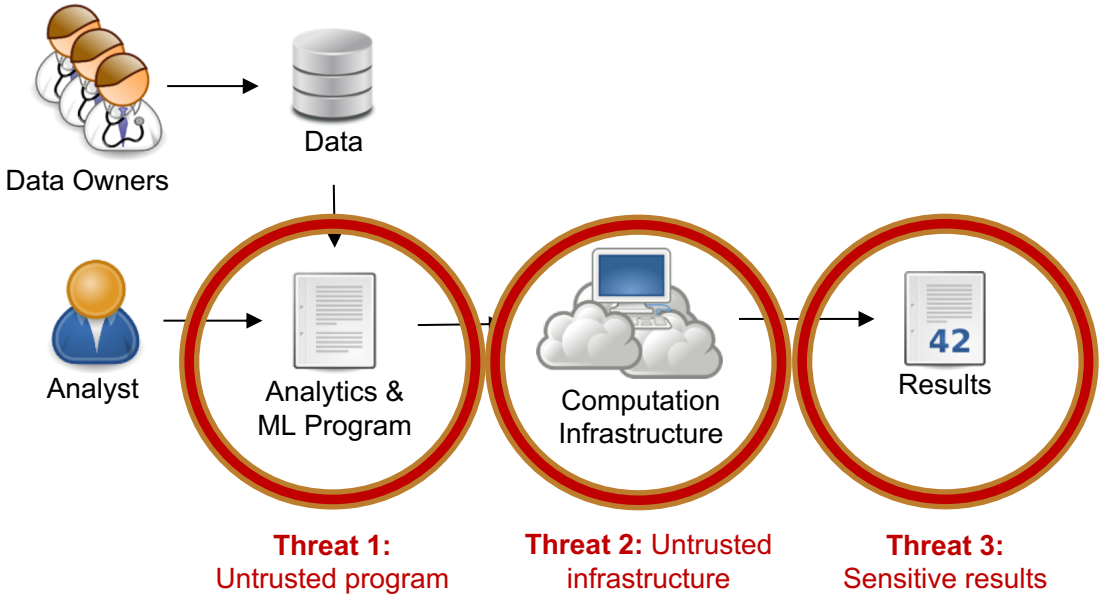
Data analysis and machine learning has many applications, huge potential impact

“Data is the New Oil”

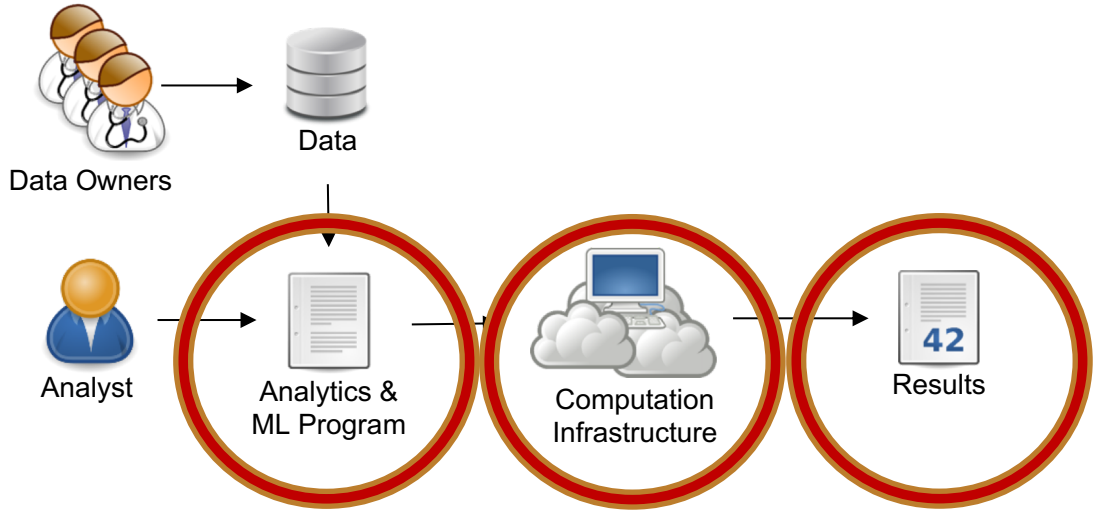
Current Frameworks for Data Analytics & Machine Learning



Current Frameworks Insufficient



Desired Solutions for Confidentiality/Privacy



Threats

Threat 1: Untrusted program

Threat 2: Untrusted infrastructure

Threat 3: Sensitive results

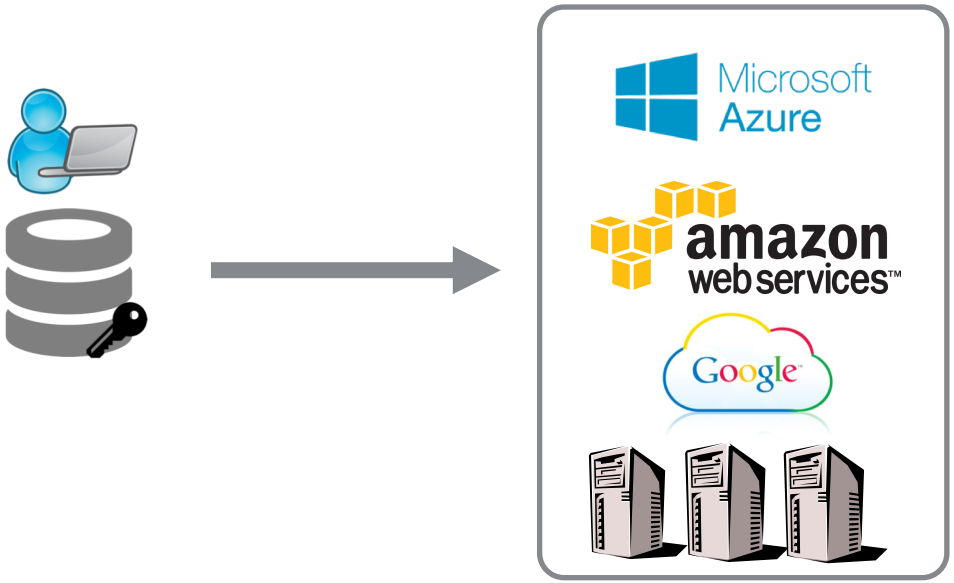
Desired Solutions

Program Rewriting & Verification

Secure Computation

Differential Privacy

Secure Computation: Simulating Trusted Third Party



- Does my secret data remain secret?
- Does the program execute as it is supposed to?
- Is the right program executed?

Secure Computation

- Example:
  - Build a word-embedding from everyone's text messages on their phones
- Challenge:
  - Text messages are highly sensitive
  - Computation infrastructure may not be trusted
- Solutions:
  - Crypto-based approach:
    - Non-interactive: Fully-homomorphic encryption (FHE)
    - Interactive: Multi-party computation (MPC)
  - Hardware-based approach:
    - Secure enclave provides isolation & remote attestation

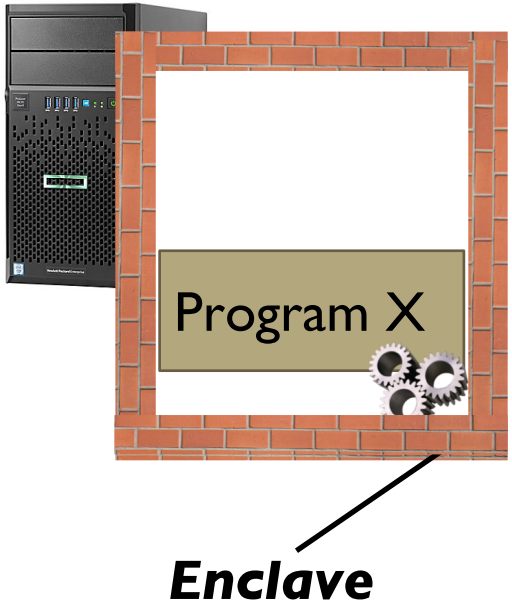
Crypto-based Secure Computation

- Fully-homomorphic encryption (FHE)
  - Given E(x), f, compute E(f(x))
  - Support general secure computation with strong security
  - High performance overhead: 10^6
  - Example: CryptoNet [Dowlin et al.]
    - Classification of an encrypted image using neural networks
    - On MNIST:
      - 51000 predictions per hour on a single PC
      - 579 seconds latency per image
- Multi-party computation (MPC)
  - Trust model: at most t out of k parties are malicious
  - Require many rounds of communication among different parties

Hardware-based secure computation

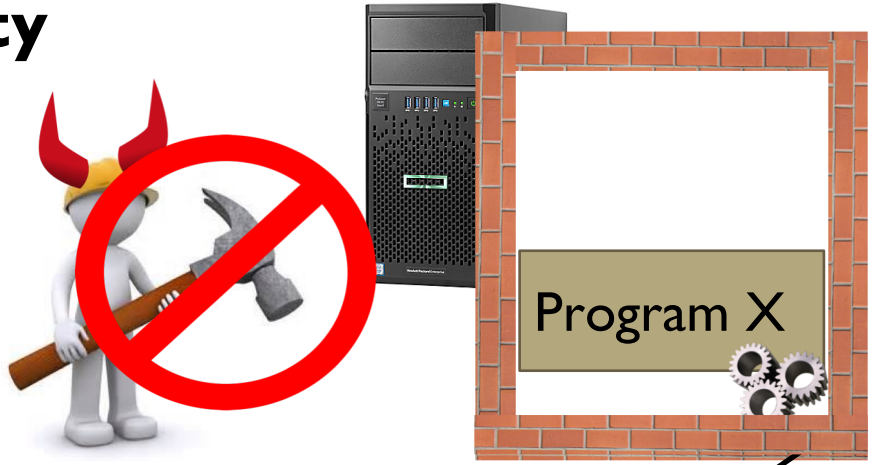
- Trusted Execution Environment (e.g., Intel SGX)
  - Secure enclave: isolation & attestation
    - Protect against malicious OS
  - Enable practical secure computation over encrypted data
    - In contrast to fully-homomorphic encryption (FHE) with 10^6 performance overhead
  - Many ecurity applications

Trusted Execution Environment (TEE)



Trusted Execution Environment (TEE)

Integrity



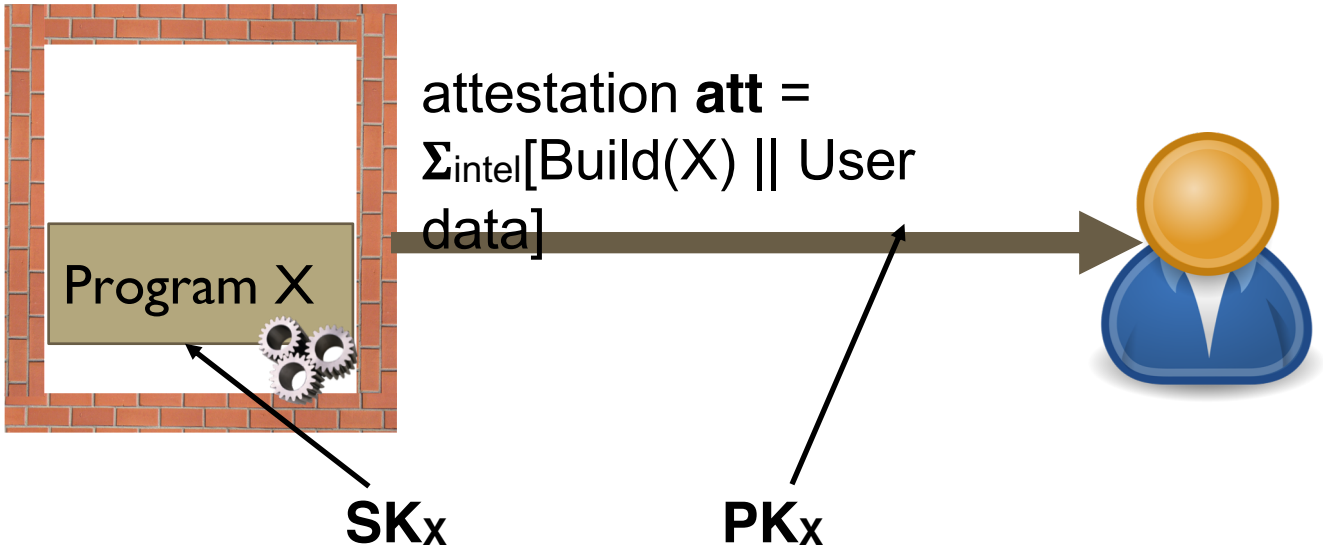
Enclave

Confidentiality

OS and other processes cannot tamper with execution of X.

OS and other processes cannot learn state of X.*

Remote attestation



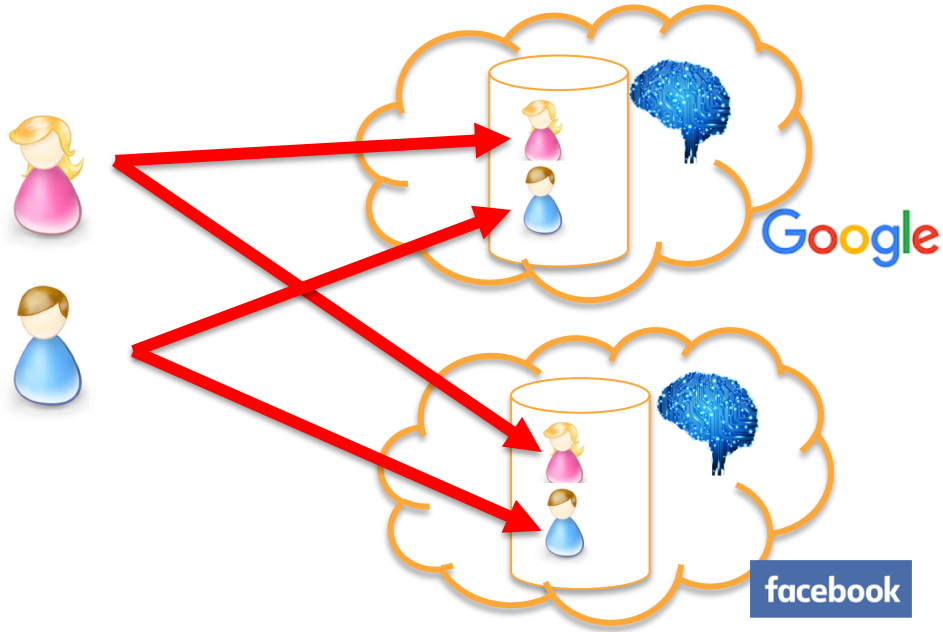
Secure Enclave as a CornerStone Security Primitive

- Strong security capabilities
  - Authenticate “itself (device)”
  - Authenticate “software”
  - Guarantee the integrity and privacy of “execution”
- Platform for building new security applications
  - Couldn’t be built otherwise for the same practical performance
  - Many examples
    - Haven [OSDI’14], VC3 [S&P’15], M2R[USENIX Security‘15], Ryoan [OSDI’16], Opaque [NSDI’17]
      - Single node or distributed computation using trusted hardware

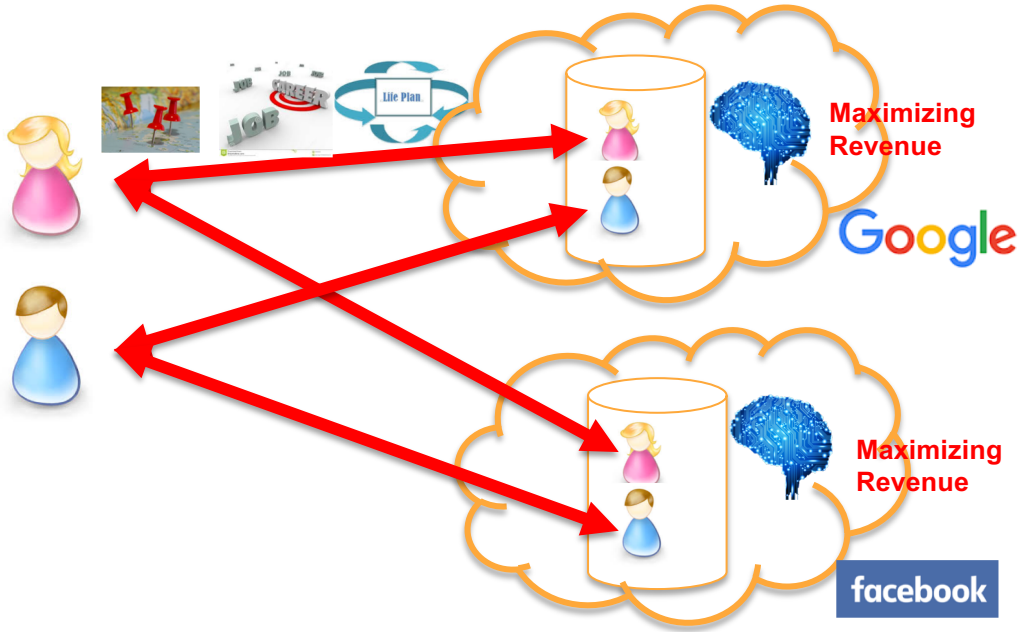
Whoever Controls & Leads in AI Will Rule the World

--Nation State Leaders

The Status Quo Today

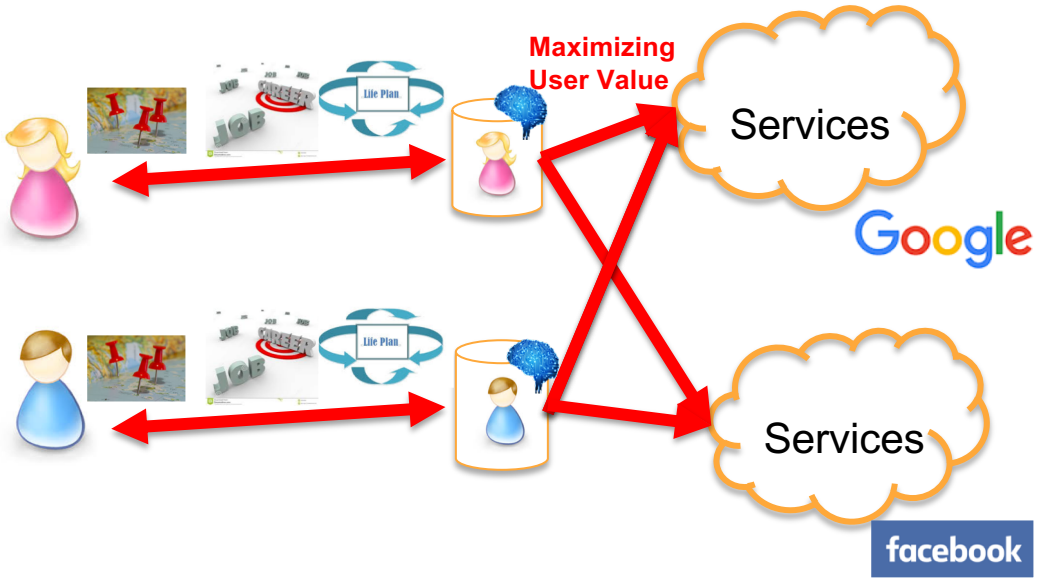


Who Will Be Running Our Lives?



Is there a different future?

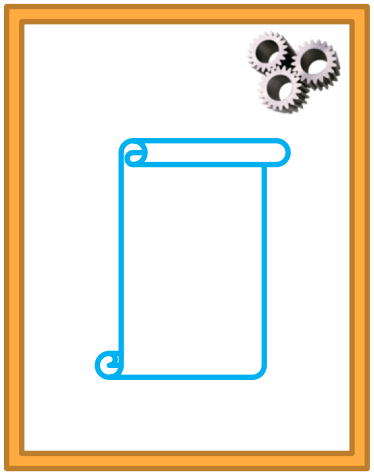
Intelligent Agent/Virtual Assistant under User Control



Oasis: Privacy-preserving Smart Contracts at Scale

Our Solution

Privacy-preserving Smart contract

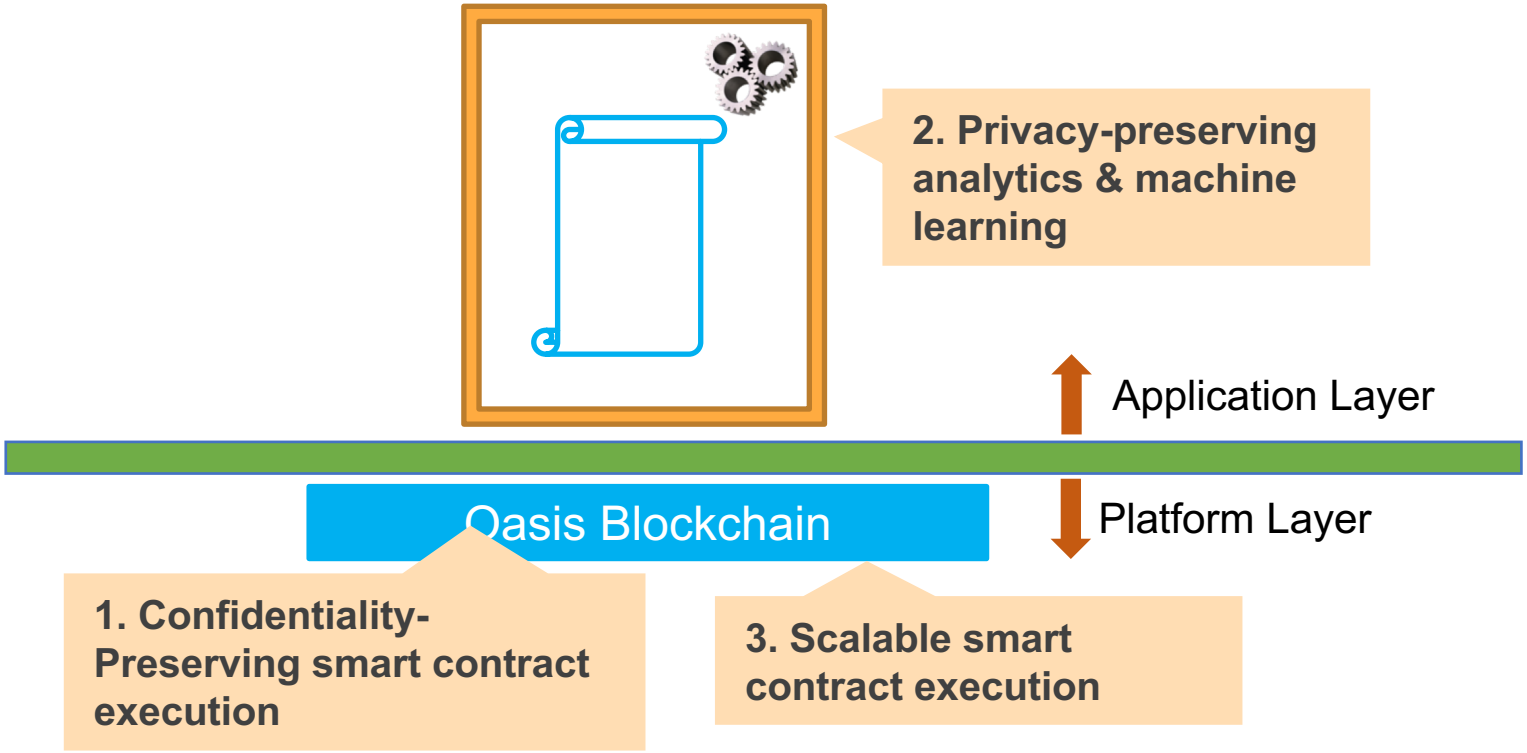


Oasis Blockchain

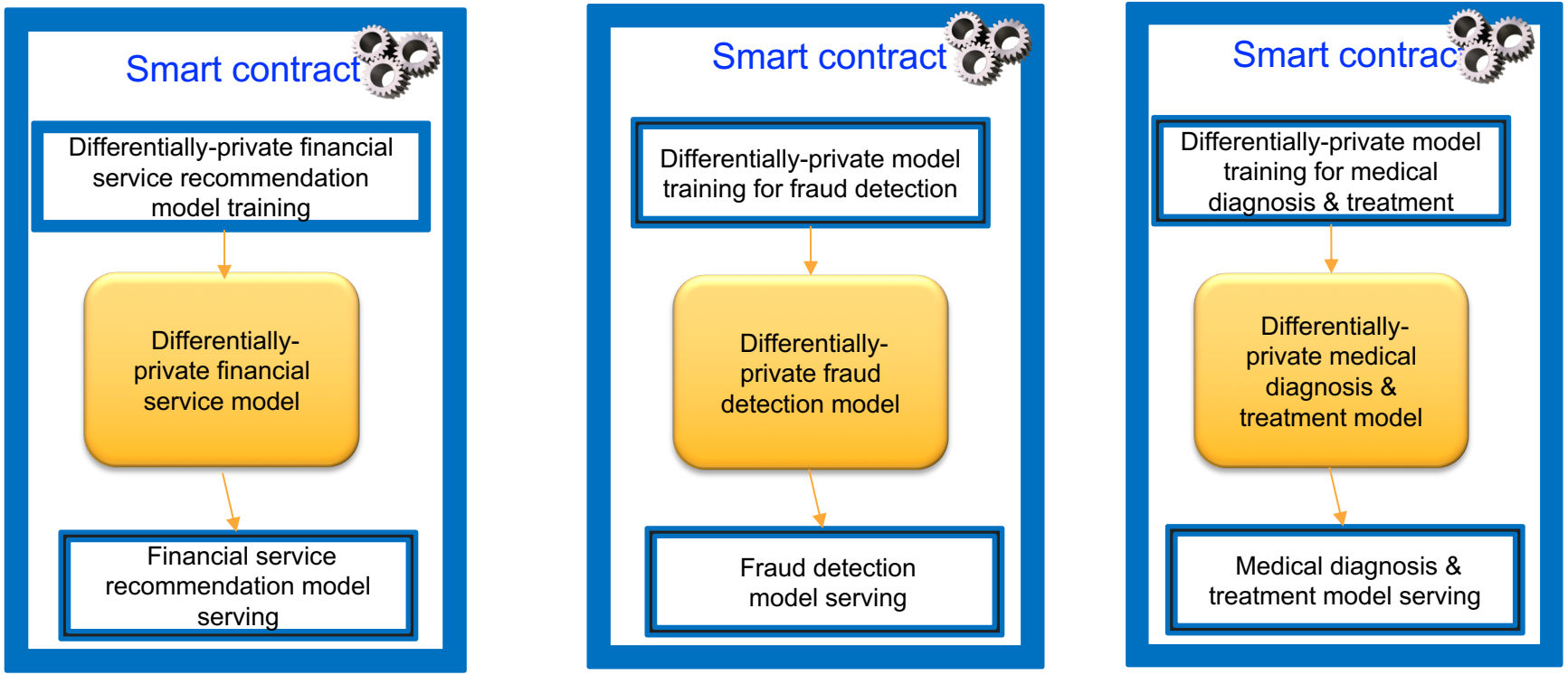
Properties of Our Solution

- Automatic enforcement of codified privacy requirements
- Without relying on any central party
- Scale to real-world applications including machine learning
- Easy to use for developers without privacy expertise

Privacy-Preserving Smart Contracts At Scale

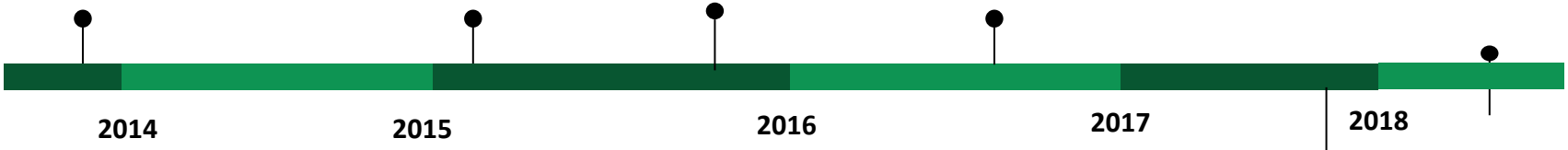


Democratization of AI: Blockchain of Intelligent Smart Contracts



Trusted hardware timeline

Closed source



2014 — ARM TrustZone

Hardware-based isolation for embedded devices

2015 — SGX: Software Guard Extensions

Built in to all Core™ processors (6th-generation and later)

2016 — Trusted Execution Environment

- Hardware-based isolation
- TLK: open-source stack for TEE

2017 — SEV: Secure Encrypted Virtualization

- Introduced in EYPC server processor line
- Provides confidentiality but not integrity

2018 — Intel SGX version 2

- In pipeline
- Drivers already available

Open source

Keystone: Open-source secure enclave

https://keystone-enclave.github.io

- Collaboration between Berkeley & MIT
- Remedies issues in previous secure hardware
- Can be publicly analyzed and verified
- Can be manufactured by any manufacturer
- First release: Fall 2018

Challenges in Secure Hardware

- How secure can it be? Under what threat models?

- What would you entrust with secure hardware?
  - Your bitcoin keys
  - Financial data
  - Health data

Grand Challenge

- Can we create trustworthy secure enclave as a cornerstone security primitive?
- Widely deployed, enable secure systems on top
- A new secure computation era

Path to Trustworthy Secure Enclave

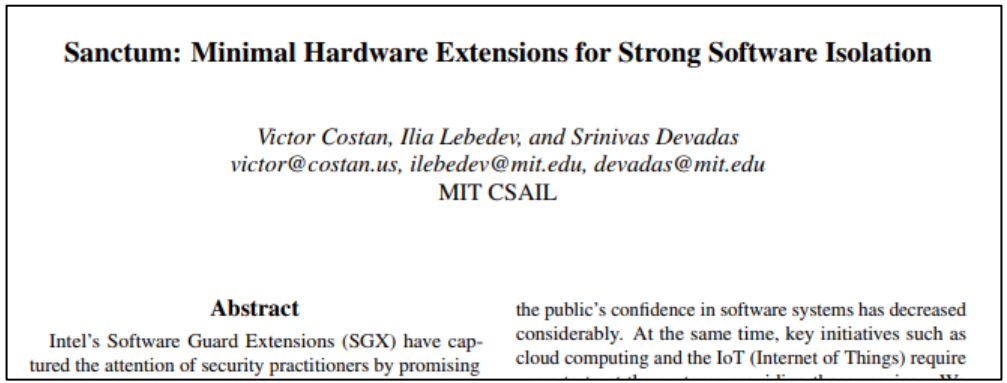
- Open source design
- Formal verification
- Secure supply-chain management

Importance of Open Source Secure Enclave Design

- None of the commercial TEE designs is opened to public
  - Cannot analyze/verify a hardware vendor' design in closed source
- No industry agreement on right solution for everything
- Open source provides transparency & enables high assurance
- Open source builds a community

Sanctum

- Secure processor design on RISC-V ISA
- Fully-isolated per-enclave page table
- Defending against cache-based side-channel attacks



Sanctum: Minimal Hardware Extensions for Strong Software Isolation

Victor Costan, Ilia Lebedev, and Srinivas Devadas
victor@costan.us, ilebedev@mit.edu, devadas@mit.edu
MIT CSAIL

Abstract

Intel's Software Guard Extensions (SGX) have captured the attention of security practitioners by promising the public's confidence in software systems has decreased considerably. At the same time, key initiatives such as cloud computing and the IoT (Internet of Things) require

Keystone Enclave

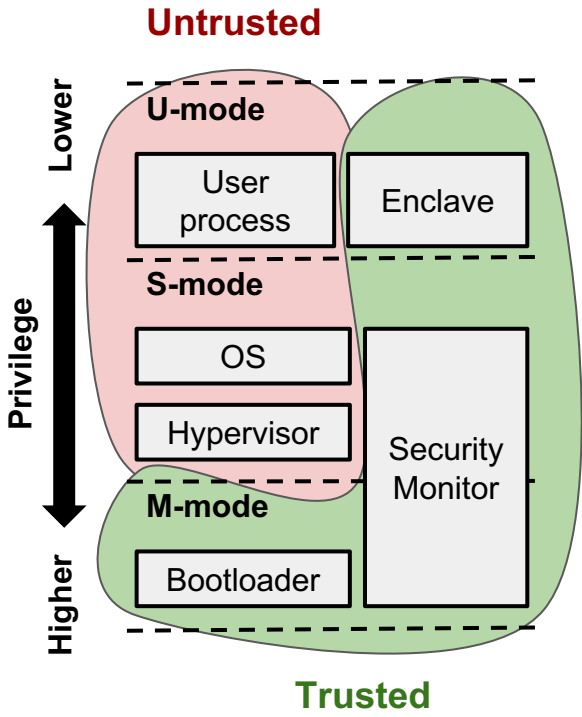
- What is the Keystone Enclave?
  - Open-source Trusted Execution Environment (TEE) based on RISC-V
- Strong Memory Isolation
  - ISA-enforced memory access management
  - Separate virtual memory management without relying on the OS
- Simple and Portable
  - Exploits standard RISC-V ISA primitives: PMP, TVM
- Remote Attestation
  - Extends MIT Sanctum’s remote attestation
- Open Source
  - Full software/hardware stack will be released
  - Run on many platforms: QEMU, Amazon AWS FPGA (FireSim), HiFive Unleashed, ...

Keystone Goals

1. Chain of Trust
   - Secure boot
   - Remote attestation
   - Secure key provisioning (PUF)
2. Memory Isolation
   - Physical memory protection
   - Page table isolation
3. Defense against Physical Attack
   - Memory encryption
   - Memory address bus encryption
4. Defense against Side-channel Attack
   - Isolated architecture
5. Formal Verification
6. Deployment
   - Amazon AWS FPGAs (FireSim)
   - HiFive Unleashed
7. Tape Out to Chip
8. Secure supply-chain management

Keystone & RISC-V

- Three privilege modes: **U**ser, **S**upervisor, **M**achine
- Security Monitor
  - Physical memory protection with **M-mode**
  - Virtual memory isolation with **S-mode**: minimize M-mode attack surface
- Keystone relies on Standard RISC-V Primitives
  - Simple & portable design by using RISC-V priv-1.10 spec
  - Physical Memory Protection (PMP): dynamically configurable memory access restriction
  - Trap Virtual Memory (TVM): intercepting supervisor virtual memory management

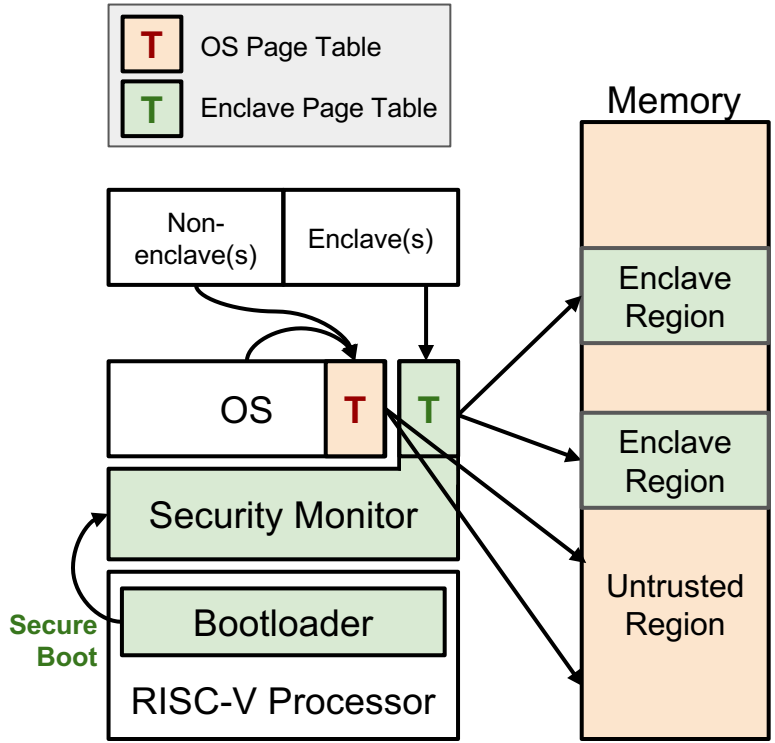


RISC-V Requirements

- Compatible ISA Subsets
  - Any subsets of RV32/RV64 are compatible
- RISC-V Priv-1.10
  - Hardware should support three software privilege modes as specified (M/S/U-mode)
  - Physical Memory Protection (PMP):
    Hardware should have PMP feature as specified and have more than 2 PMP registers
  - Trap Virtual Memory (TVM):
    Hardware should let M-mode intercept virtual memory management of the S-mode
- Additional Components
  - True Random Number Generator (TRNG) - Read-only register
  - Physical Unclonable Function (PUF) - Only readable by M-mode

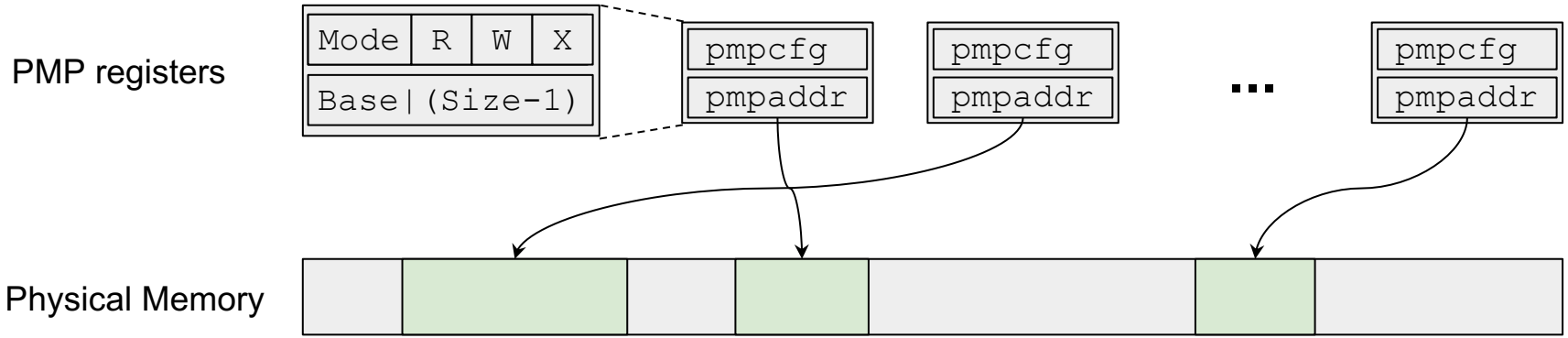
Keystone v1.0 Architecture

- Consists of Two Privileged Software Components
  - **Bootloader** (read-only, baked in CPU's boot ROM)
  - **Security monitor**
- Bootloader
  - Measures and signs the security monitor
- Security Monitor (SM)
  - Managing enclaves
  - Remote attestation
  - Memory isolation and VM management
  - Sanitizing interrupts
- Keystone Kernel Module
  - Provides Keystone API
  - Coordinates the OS and the SM

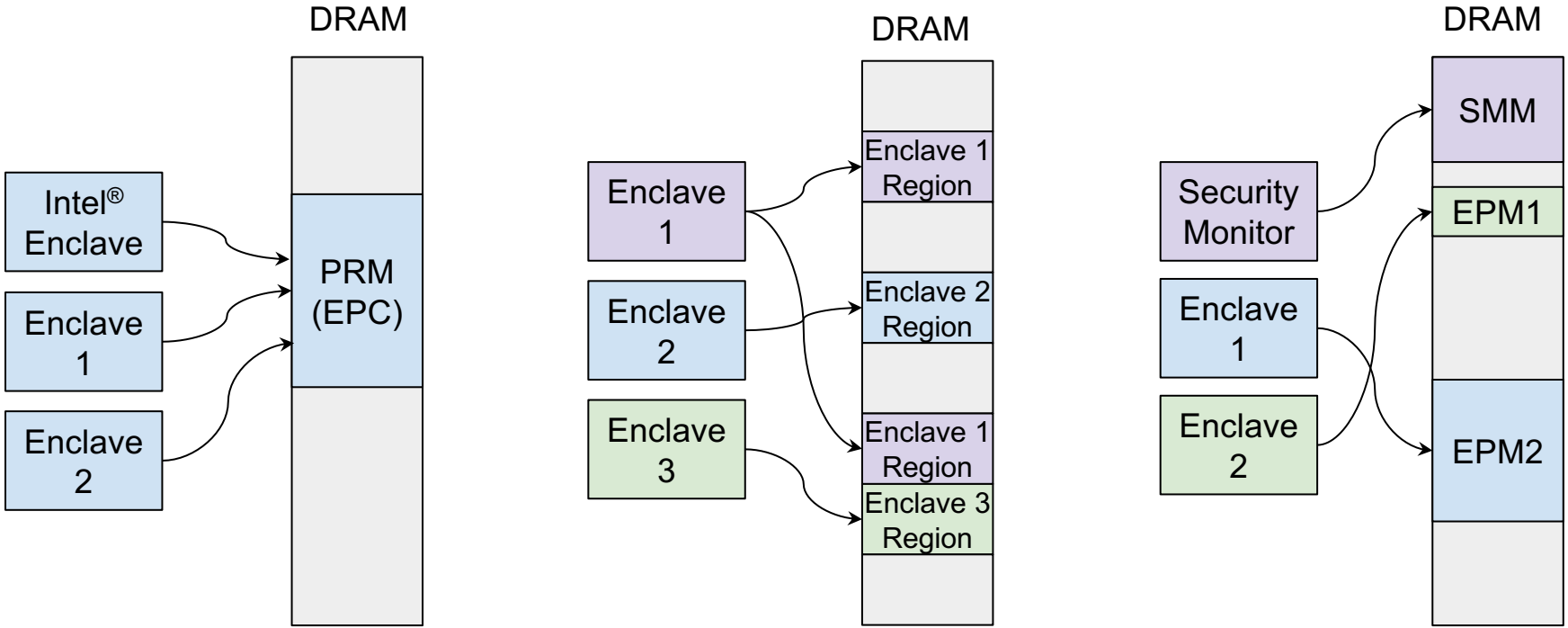


Physical Memory Protection (PMP)

- Limit the physical access permissions for each privilege mode
- Fine-grained range granularity as small as 4 bytes (up to entire DRAM)
- Dynamically configurable by writing to *PMP registers*
- RISC-V standard - No additional hardware



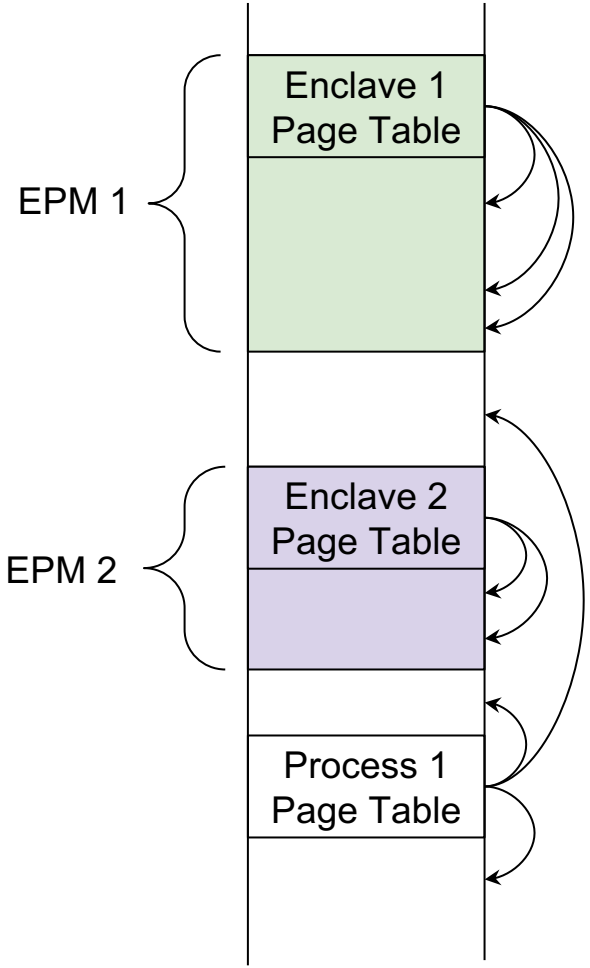
Memory Layout: SGX vs. Sanctum vs. Keystone



Intel SGX **Sanctum** **Keystone**

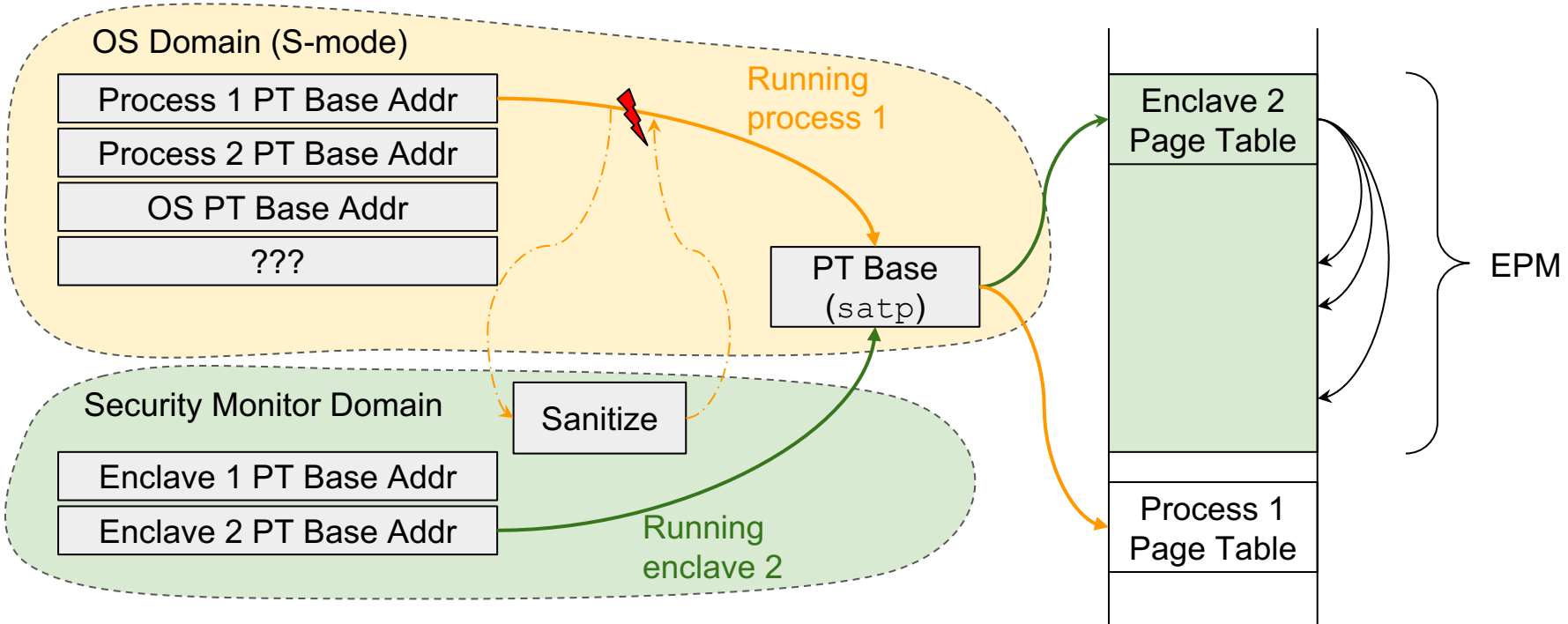
Page Table Isolation

- For a complete memory isolation, enclave VM must not rely on the OS's VM management [Xu et al., Oakland'15]
- Per-Enclave Page Table
  - Each enclave has its own page global directory (pgd)
  - Initialized & managed by security monitor
  - Enclaves run on U-mode, so they can see only their own private virtual address space.
- Page Table Handler
  - Handled by security monitor but in S-mode
- OS also manages VM; how to isolate?
  - Trap Virtual Memory (TVM) in RISC-V



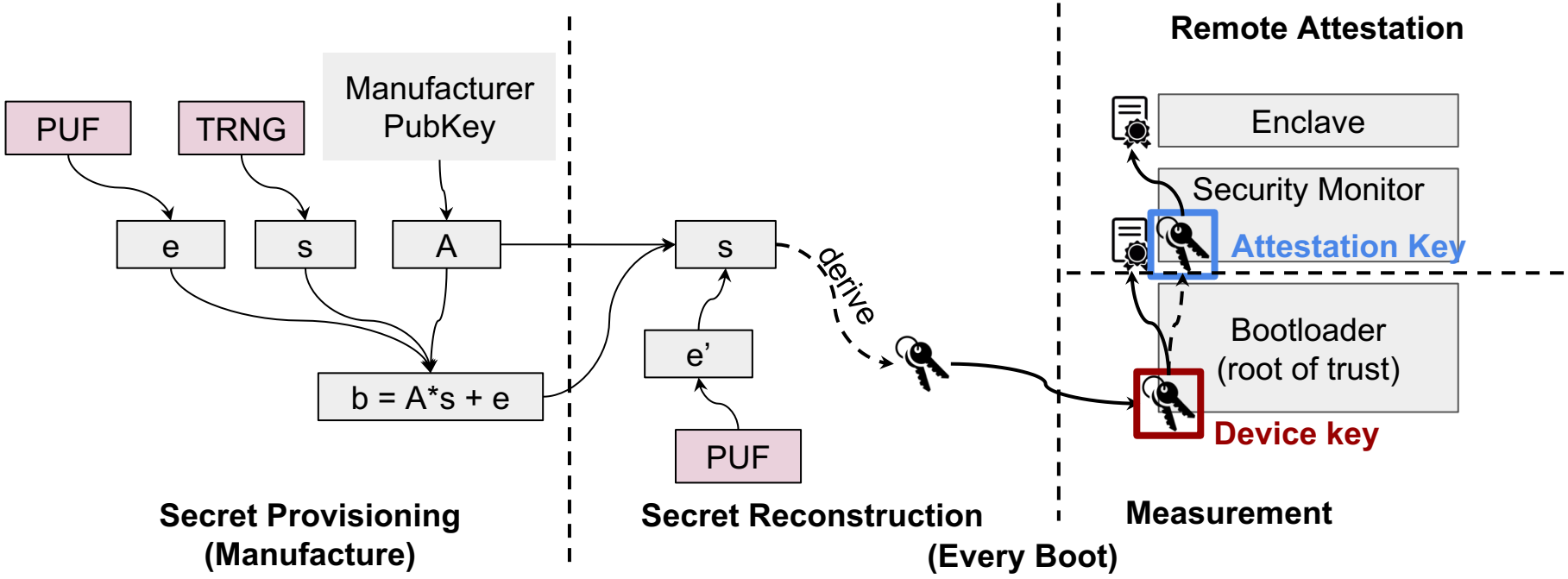
Trap Virtual Memory (TVM)

- If enabled, M-mode intercepts S-mode virtual memory management:
  - Write to `satp` (page table base register) or `sfence.vma` (TLB flush)
- Security monitor sanitizes OS page table management

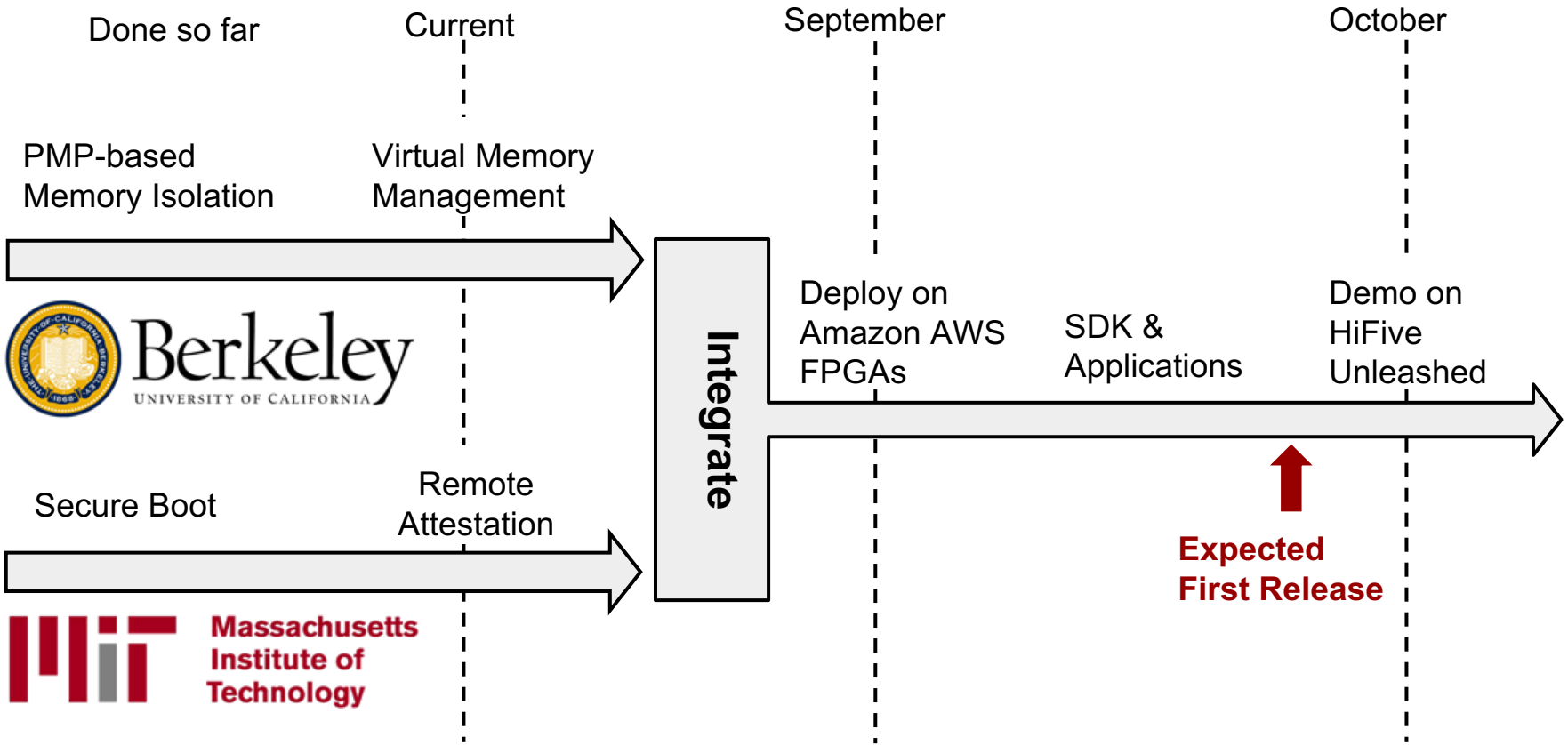


Secure Boot and Remote Attestation

- Keystone Inherits Remote Attestation from Sanctum
- Secure Boot and Key Provisioning using LPN PUF
  - Lebedev et al., “Secure Boot and Remote Attestation in the Sanctum Processor”, IACR’18

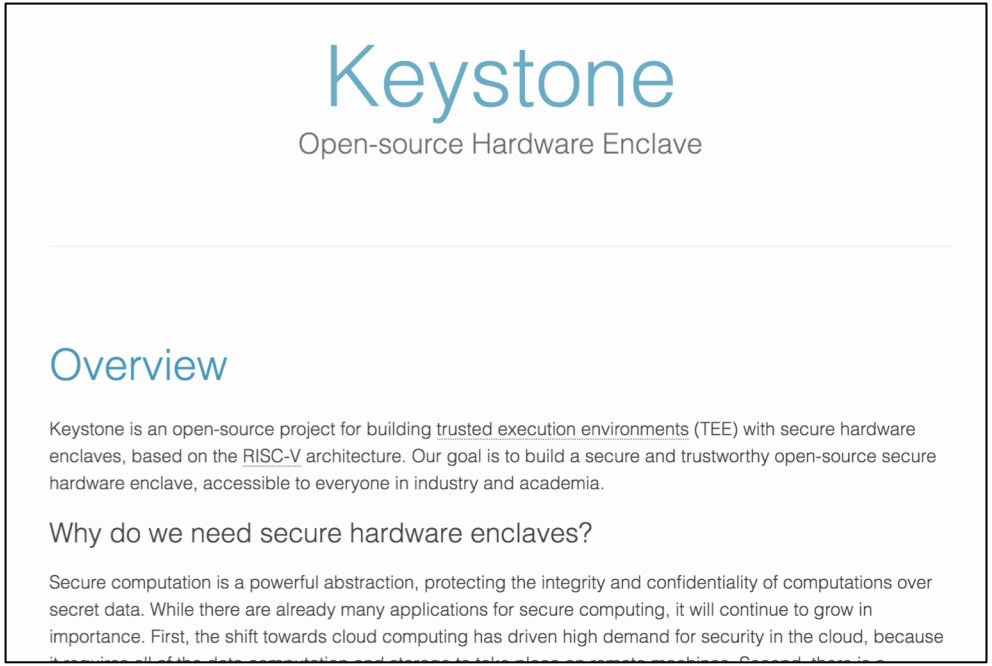


Timeline


Done so far
Current
September
October
PMP-based
Memory Isolation
Virtual Memory
Management
Berkeley
UNIVERSITY OF CALIFORNIA
Integrate
Deploy on
Amazon AWS
FPGAs
SDK &
Applications
Demo on
HiFive
Unleashed
Secure Boot
Remote
Attestation
Expected
First Release
Massachusetts
Institute of
Technology

Keystone Website and the Roadmap

Website: https://keystone-enclave.org



Keystone

Open-source Hardware Enclave

Overview

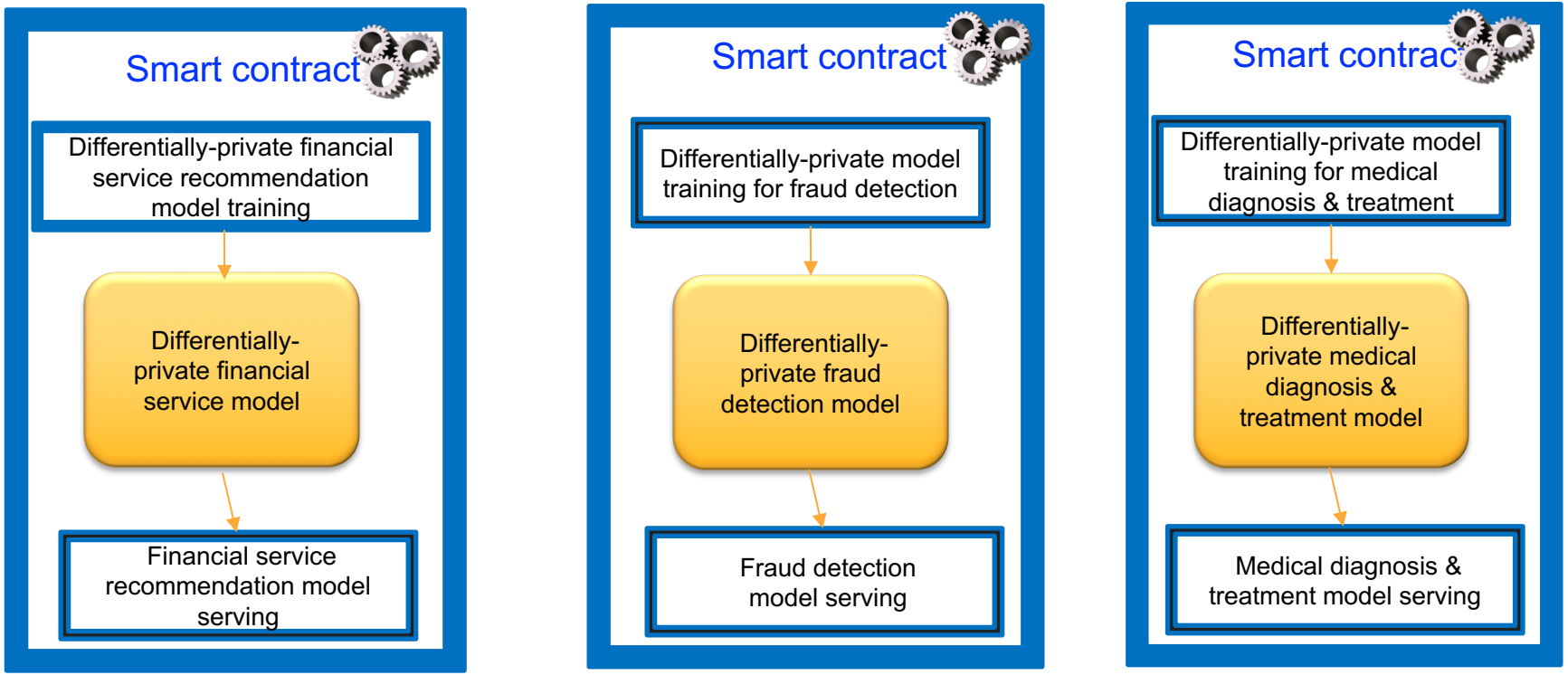
Keystone is an open-source project for building trusted execution environments (TEE) with secure hardware enclaves, based on the RISC-V architecture. Our goal is to build a secure and trustworthy open-source secure hardware enclave, accessible to everyone in industry and academia.

Why do we need secure hardware enclaves?

Secure computation is a powerful abstraction, protecting the integrity and confidentiality of computations over secret data. While there are already many applications for secure computing, it will continue to grow in importance. First, the shift towards cloud computing has driven high demand for security in the cloud, because

1. Chain of Trust
   - Secure boot
   - Remote attestation
   - Secure key provisioning (PUF)
2. Memory Isolation
   - Physical memory protection
   - Page table isolation
3. Defense against Physical Attack
   - Memory encryption
   - Memory address bus encryption
4. Defense against Side-channel Attack
   - Isolated architecture
5. Formal Verification
6. Deployment
   - RISC-V QEMU
   - Amazon AWS FPGAs (FireSim)
   - HiFive Unleashed
7. Tape Out to Chip
8. Secure supply-chain management

Democratization of AI: Blockchain of Intelligent Smart Contracts



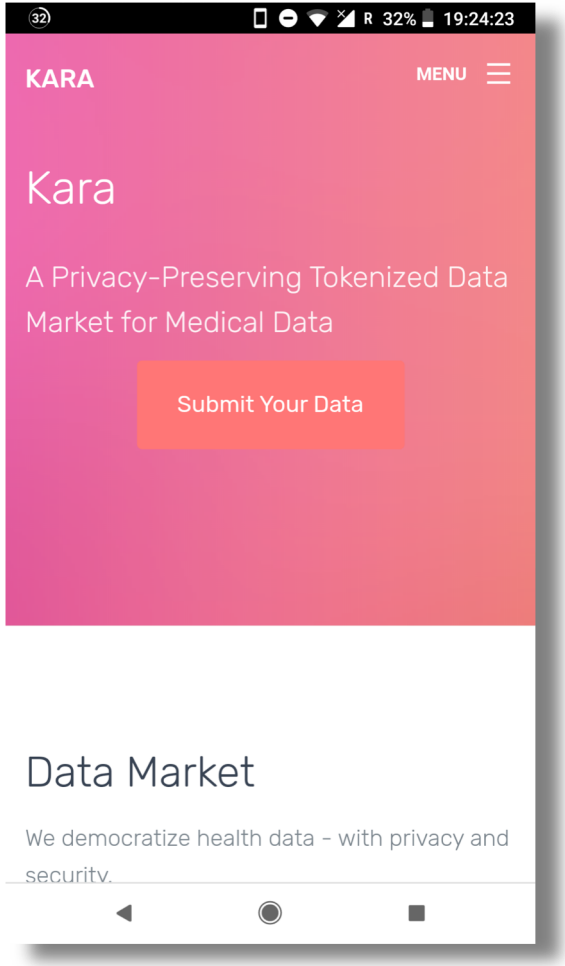
Kara

A Privacy-Preserving Tokenized Data Market for Medical Data

Oasis Labs Berkeley UNIVERSITY OF CALIFORNIA ETH Zürich Stanford HEALTH CARE

Medical data is locked in “Data Silos”.

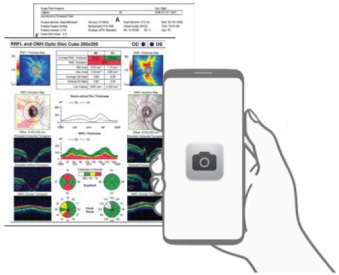
Goal: Incentivize doctors and patients to share data and improve medical research



How it works

Doctors / Patients

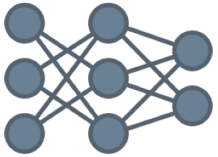
Researchers



Kara App

2. App automatically processes data and stores it to Oasis

Private Training



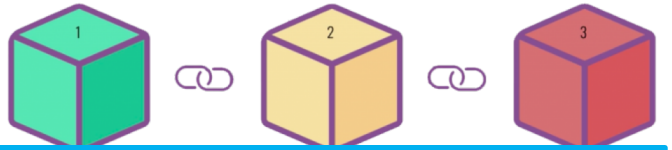
1. Patients take picture of their medical data

Data Market

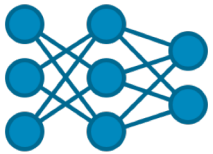
3. Researchers pay for model training

Untrained Model

Token



Oasis Blockchain



4. Receive **payments**

Privacy-preserving smart contracts serve as business logic

4. Receive **trained** model

Nick Hynes, Raymond Cheng, Noah Johnson, David Dao, Dawn Song. "**A Demonstration of Sterling: A Privacy-Preserving Data Marketplace**" in VLDB'18 (Demo Track)

David Dao, Dan Alistarh, Claudiu Musat, Ce Zhang. "**DataBright: Towards a Global Exchange for Decentralized Data Ownership and Trusted Computation**"

Oasis Labs Just Launched!


MIT Technology Review

Log in / Create an account Search

Topics+ The Download Magazine Events More+

Meet Oasis Labs, the blockchain startup Silicon Valley is buzzing about


Forbes

Big Hitter Crypto Funds Pile Into Privacy-Enhanced Smart Contract Startup Oasis Labs


THE WALL STREET JOURNAL.

U.S. Edition | July 10, 2018 | Today's Paper | Video

Oasis Labs Building Cloud Computing on Blockchain With $45 Million

Backers include a16zcrypto, Accel Partners, Binance, Polychain, Metastable


WIRED

BUSINESS CULTURE

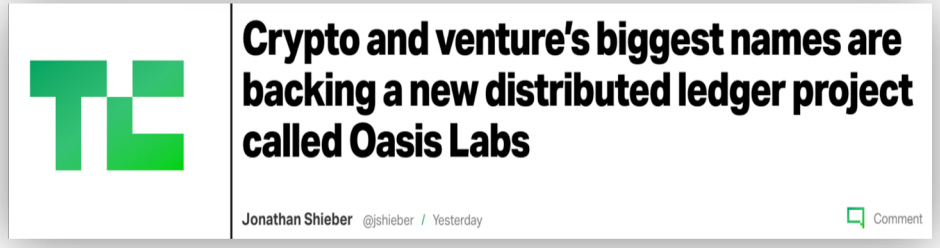
TOM SIMONITE BUSINESS 07.11.18 08:00 AM

HOW A STARTUP IS USING THE BLOCKCHAIN TO PROTECT YOUR PRIVACY


VB CHANNELS EVENTS NEWSLETTERS

Oasis Labs raises $45 million for 'privacy first' cloud on blockchain

DEAN TAKAHASHI @DEANTAK JULY 9, 2018 3:00 AM


Crypto and venture's biggest names are backing a new distributed ledger project called Oasis Labs

Jonathan Shieber @jshieber / Yesterday

Comment

Oasis Testnet

Interested in building an application on Oasis?

Join our private testnet!
https://www.oasislabs.com/developers

Oasis Labs

Building a privacy-first, high performance cloud computing platform on blockchain.

We're hiring!

www.oasislabs.com

Keystone Team

Ilia Lebedev
MIT
ilebedev@mit.edu

Srini Devadas
MIT
devadas@mit.edu

Dayeol Lee
UC Berkeley
dayeol@berkeley.edu

Krste Asanović
UC Berkeley
krste@berkeley.edu

Dawn Song
UC Berkeley
dawnsong@berkeley.edu

Massachusetts Institute of Technology

Berkeley
UNIVERSITY OF CALIFORNIA

Building Trustworthy Secure Hardware

More resources needed for research & development.

It requires community effort.

Let’s tackle the big challenges together!

Thank you!

Microsoft