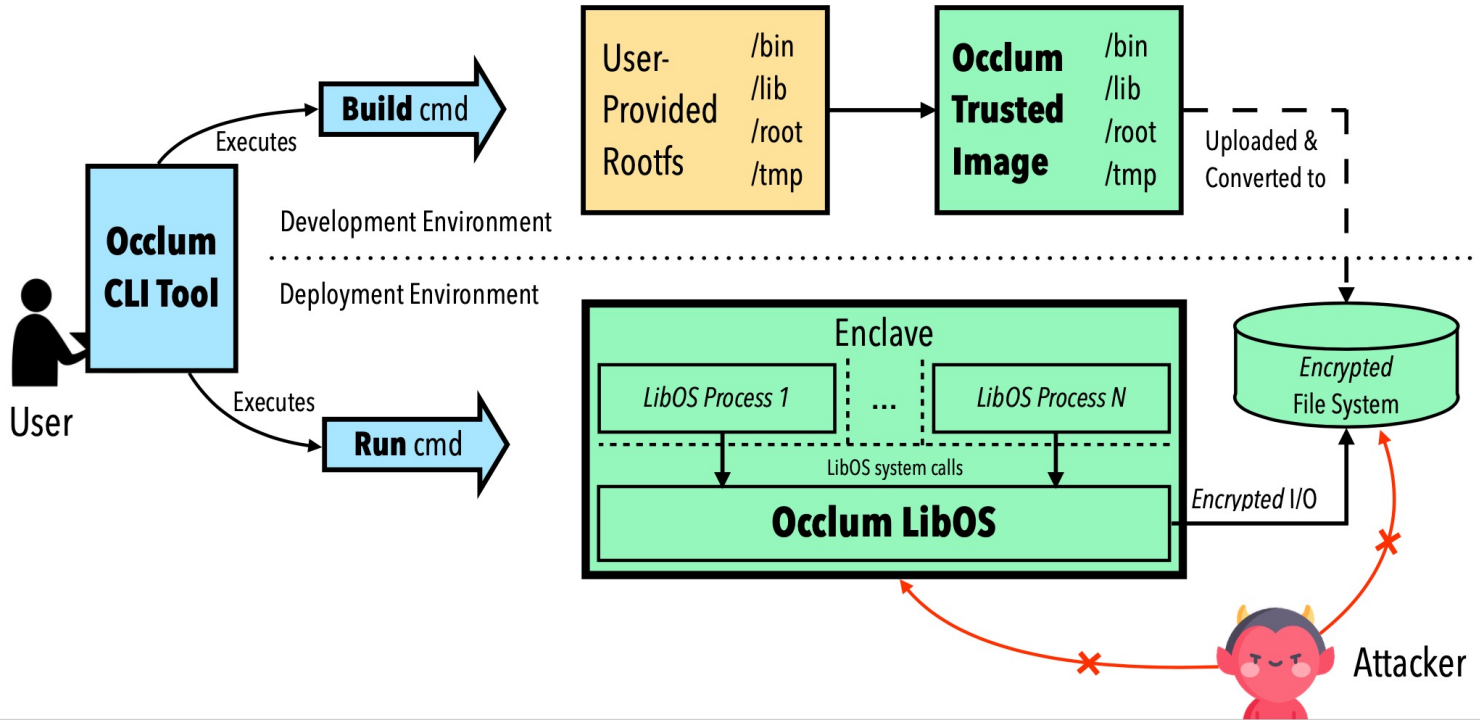
Occlum

A Memory-Safe, Multi-Process Library OS for Intel SGX

2021/11

Overview

A High-Level Overview



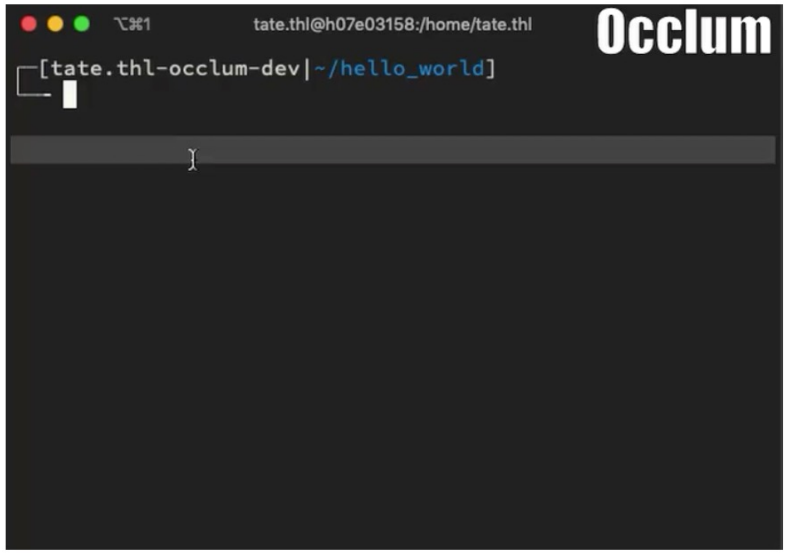
Major Features

- Usability
  - **Container-Inspired Interface**
  - **Automatic dependency analysis**
- Performance
  - **Efficient Multitasking**
  - **Async/await**
  - **IO_uring**
- Functionality
  - **Multi-stage boot**
  - **Multiple Types of File Systems**
- Security
  - **Memory Safety**
  - **Memory randomization**

Container-Inspired Interface

Container-Inspired Interface

- **Occlum core CLI commands**
  - `init`
  - `build`
  - `run`
  - `start`
  - `stop`
  - `exec`
  - `kill`



Automatic dependency analysis

Redis Demo

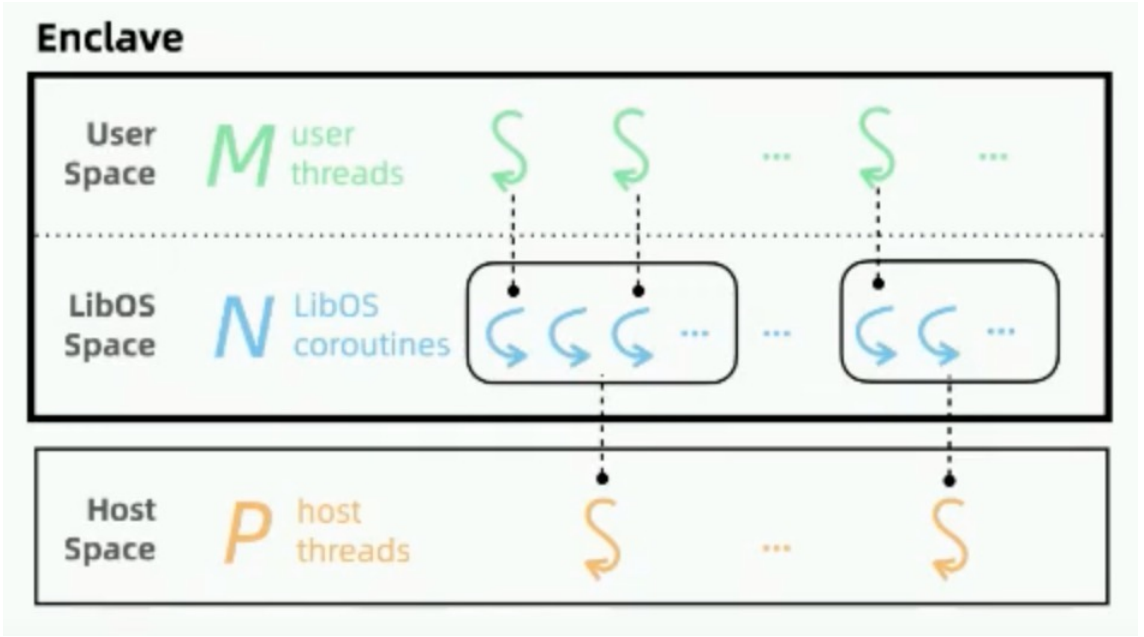
```
includes:
  - base.yaml
targets:
  - target: /bin
    copy:
      - files:
        - /usr/bin/redis-server
```

```
occlum new redis_instance
cd redis_instance
copy_bom -f ../redis.yaml --root image --include-dir /opt/occlum/etc/template
occlum build
```

Performance

Enable Rust Async/await mechanism in Occlum

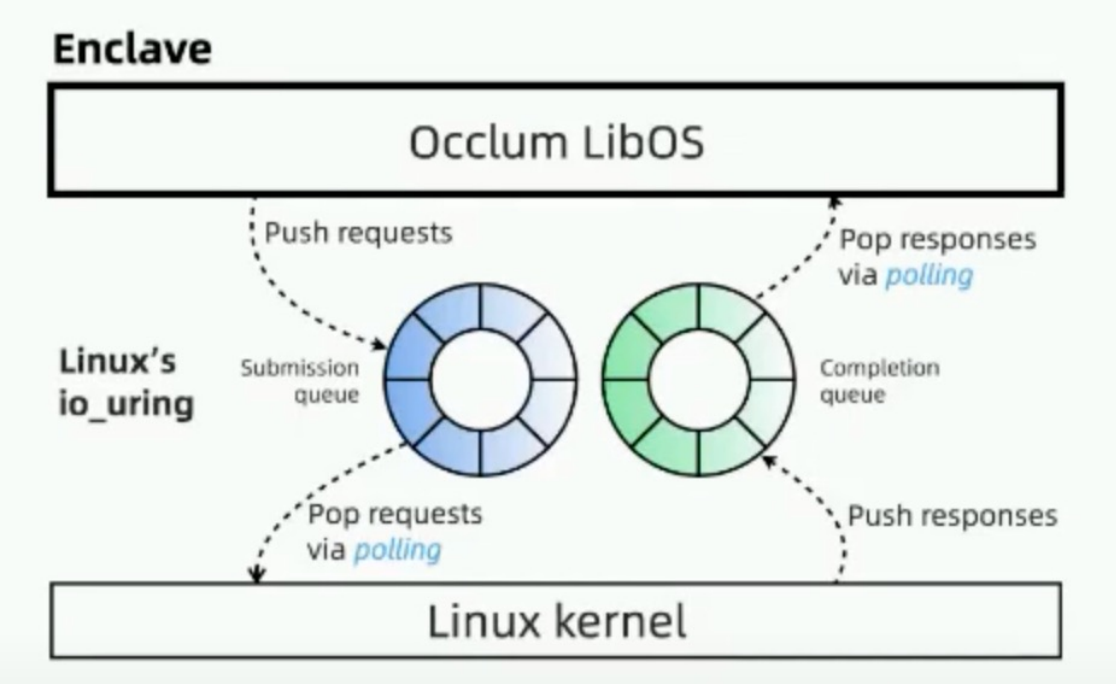
1 host thread: 1 SGX TCS : N Rust tasks : N application threads



Performance

Enable IO_uring in Occlum

Occlum handle network without Ocall



Functionality

- **Multi-stage boot**
  - The first stage is similar to the Linux initramfs. User could do some initialization tasks in this stage.
  - The second stage could be the unmodified application.
- **Multiple Types of File Systems**
  - SEFS is read-only and integrity guaranteed file system
  - Union-FS is similar to the overlay file system, which combine an read-only base SEFS and a writable empty SEFS
  - Encrypting SEFS

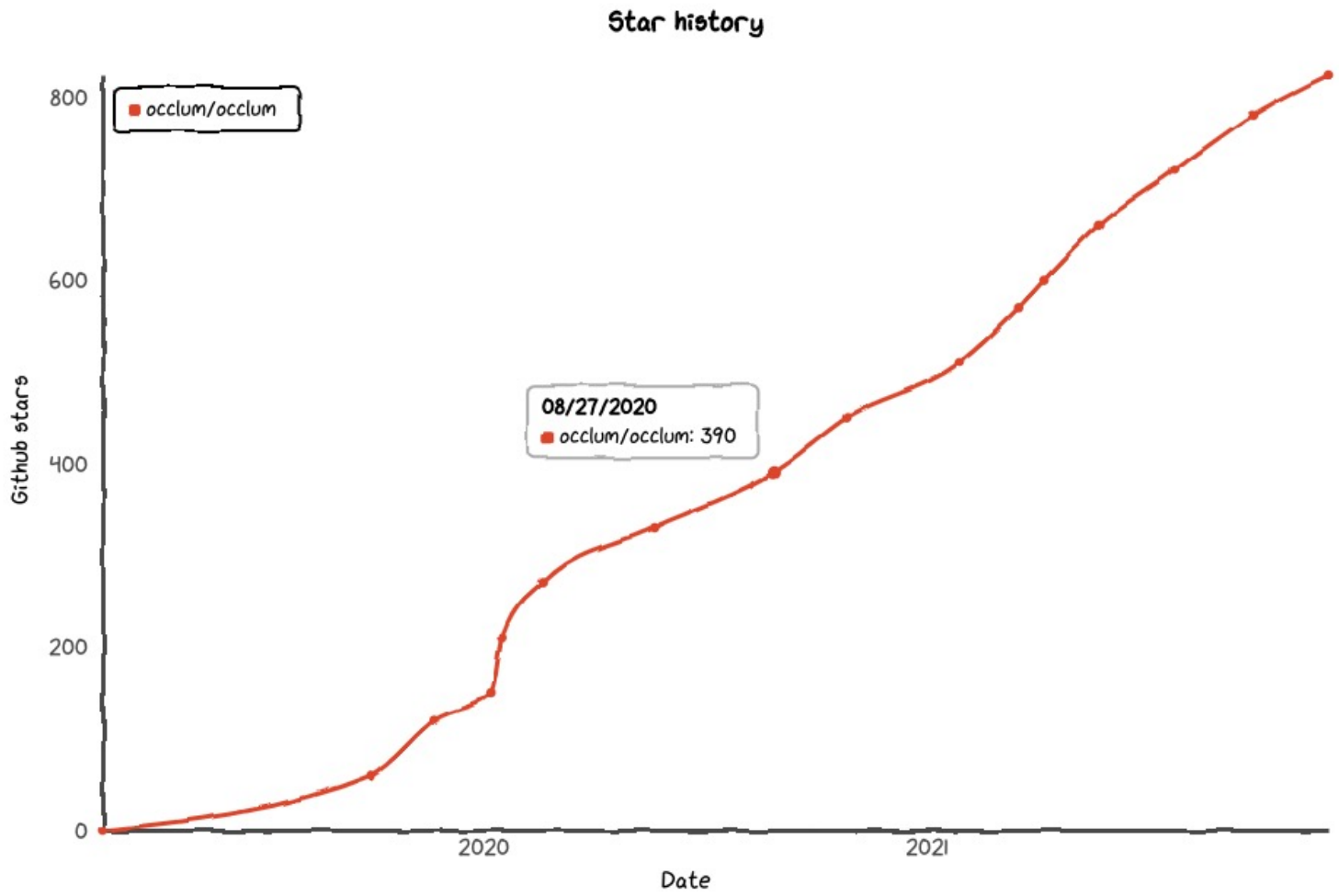
Security

- By default SGX EPC is encrypted
- Occlum enables memory random allocation
  - Attacker is hard to know the data address inside Occlum
  - Efficient use of memory

The supported usages

- bash/fish (C/C++)
- Flink (Java)
- Pytorch (python)
- sqlite (C/C++)
- tensorflow (python/C/C++)
- openvino (C/C++)
- xgboost (C/C++)
- redis (C/C++)
- Vault (golang)

Occlum Star number is 800+



Dependency

- New dependencies
  - atomic = "0.5"
  - memoffset = "0.6.1”
  - resolv-conf
  - Itertools
  - ctor = "0.1”
  - intrusive-collections = "0.9”
  - io-uring
  - futures