

RUST-VMM

Status Report - 2020

Andreea Florescu,
fandree@amazon.com

Intro

What is rust-vmm?

- Virtualization components
- Written in Rust
 - Components = package (Rust Crates)
- Open Source
- <https://github.com/rust-vmm>

Rust-vmm in Production

— — —

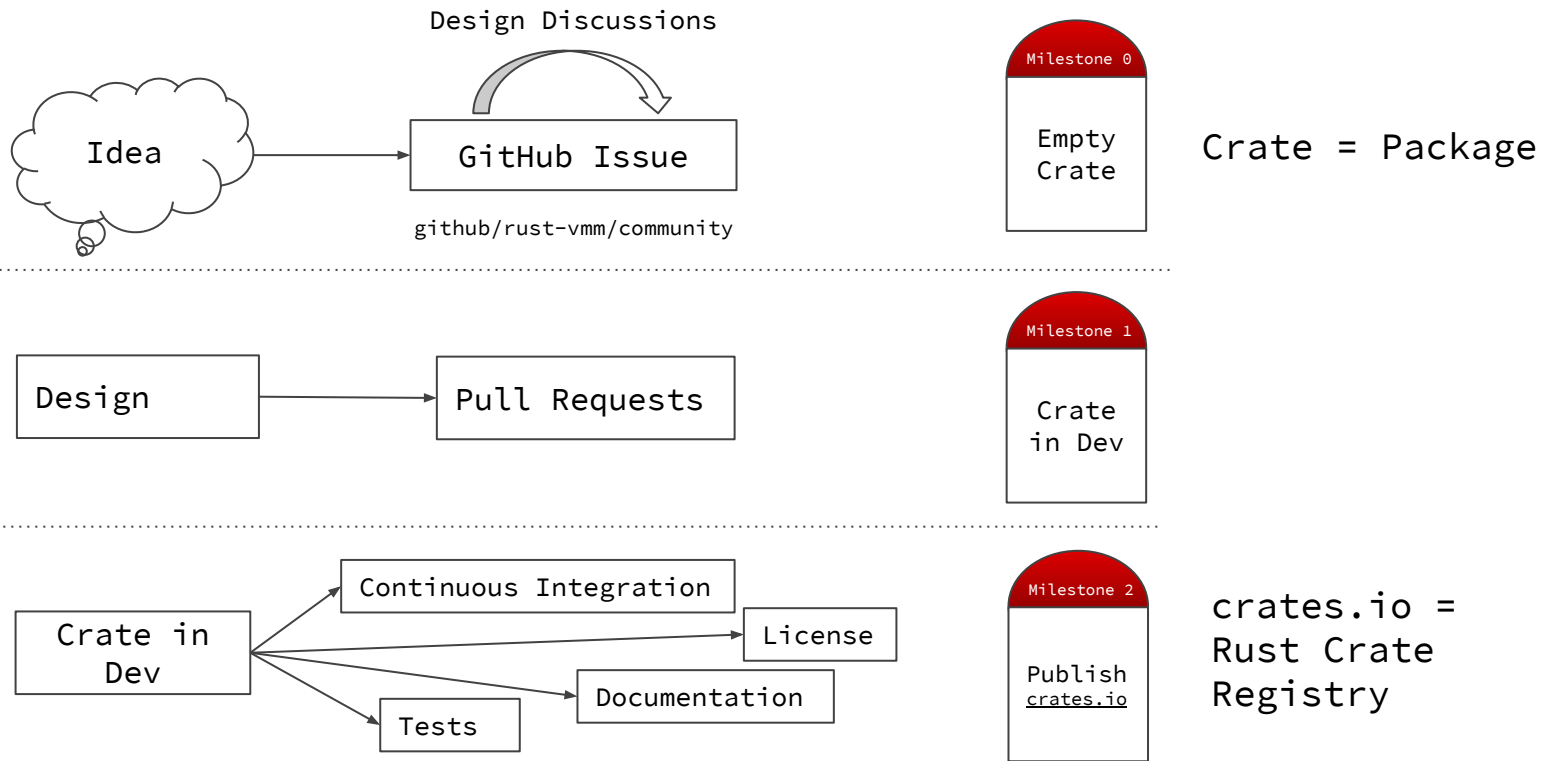
- [Firecracker](#)
- [Cloud Hypervisor](#)
- Alibaba Cloud Sandbox
- [Enarx](#)

Rust-vmm in Production

— — —

- [Firecracker](#)
- [Cloud Hypervisor](#)
- Alibaba Cloud Sandbox
- [Enarx](#)
- [libkrun](#)
- [Nydux: Dragonfly Container Image Service](#)
- ...

From idea to published crate



Component Status

Status 2019

vhost

vm-allocator

Milestone 0
Empty
Crate

vm-device

vm-virtio

Milestone 1
Crate
in Dev

linux-loader

vm-memory

virtio-bindings

kvm-bindings

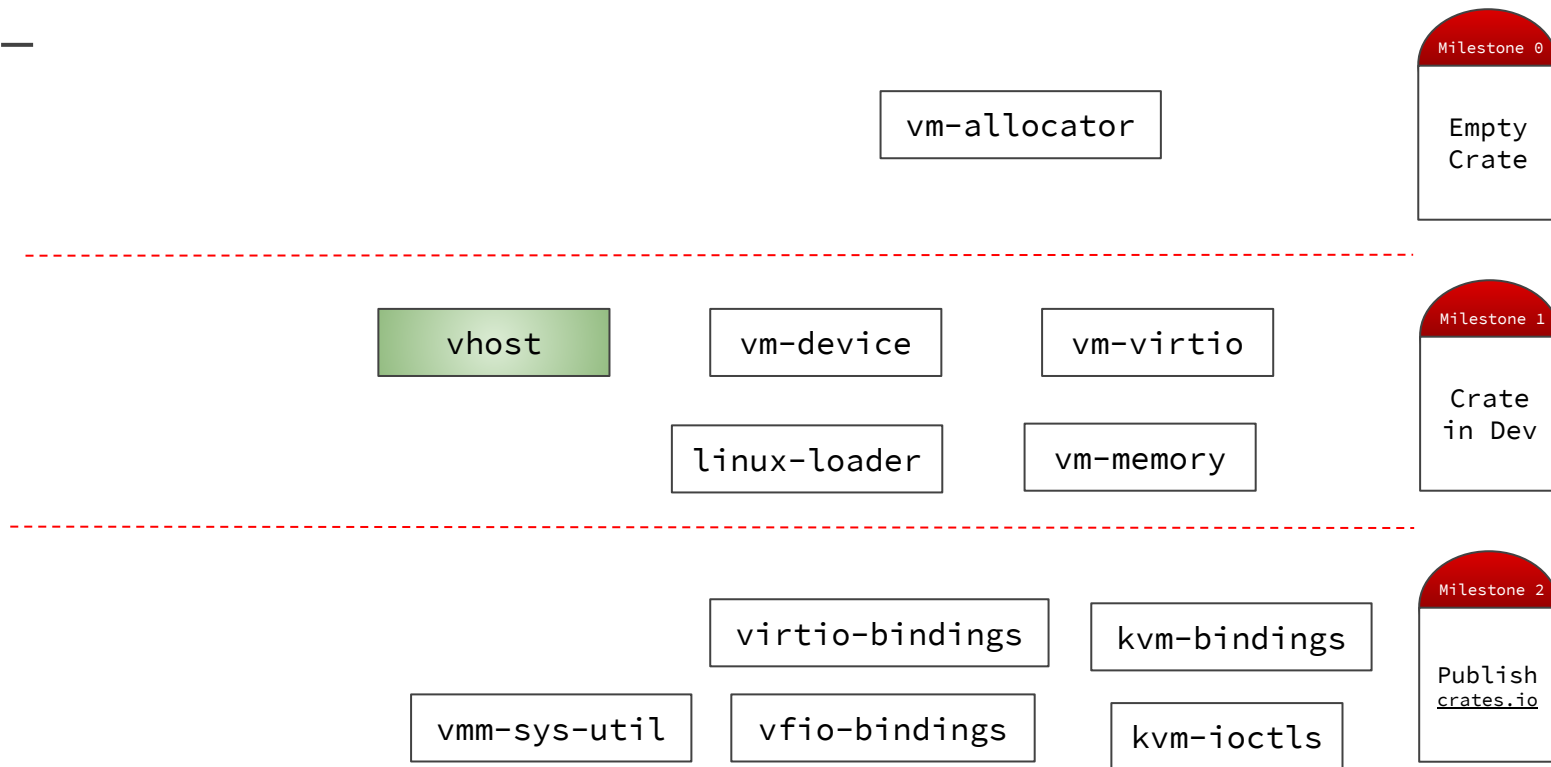
Milestone 2
Publish
crates.io

vmm-sys-util

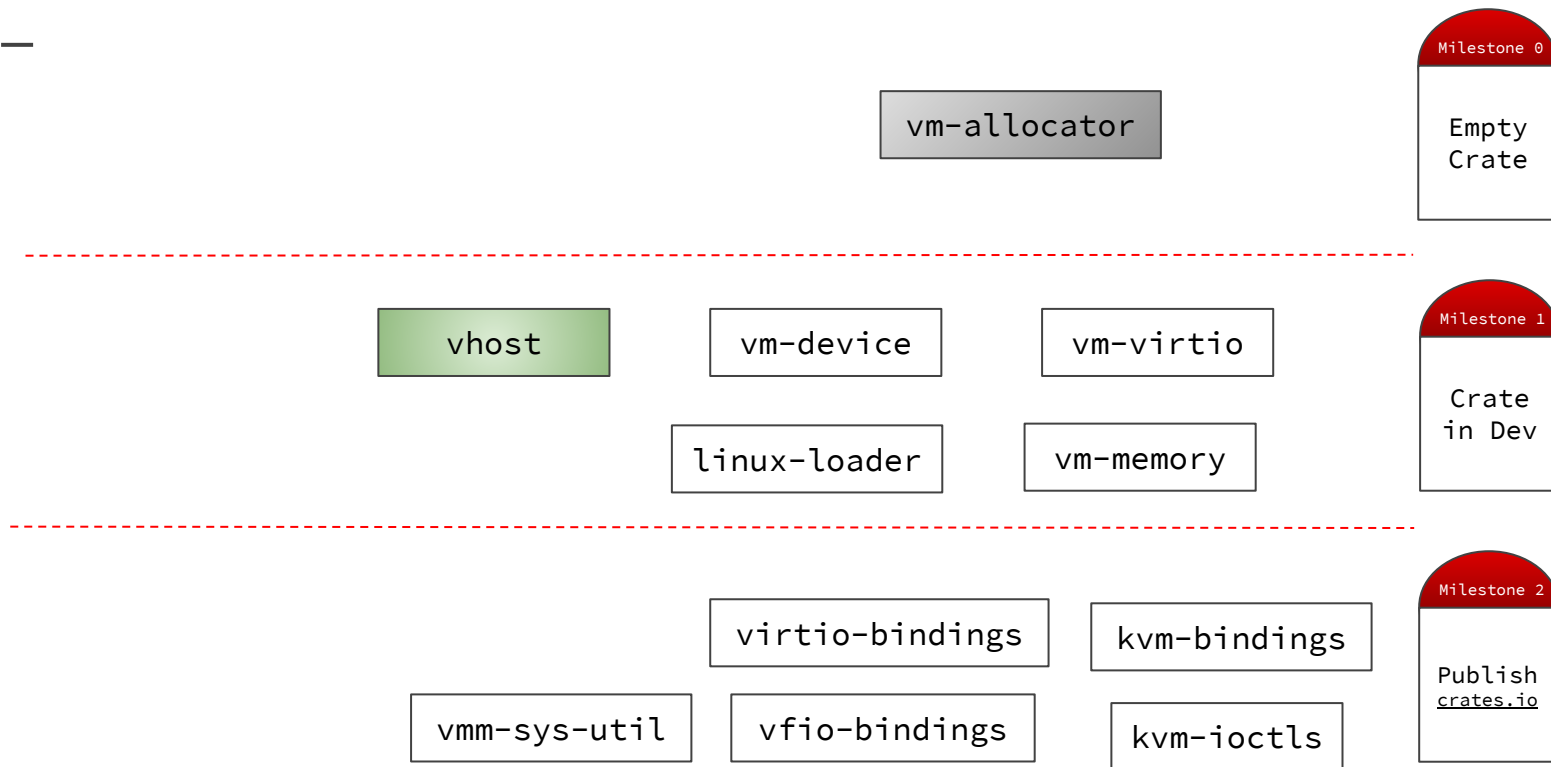
vfio-bindings

kvm-ioctls

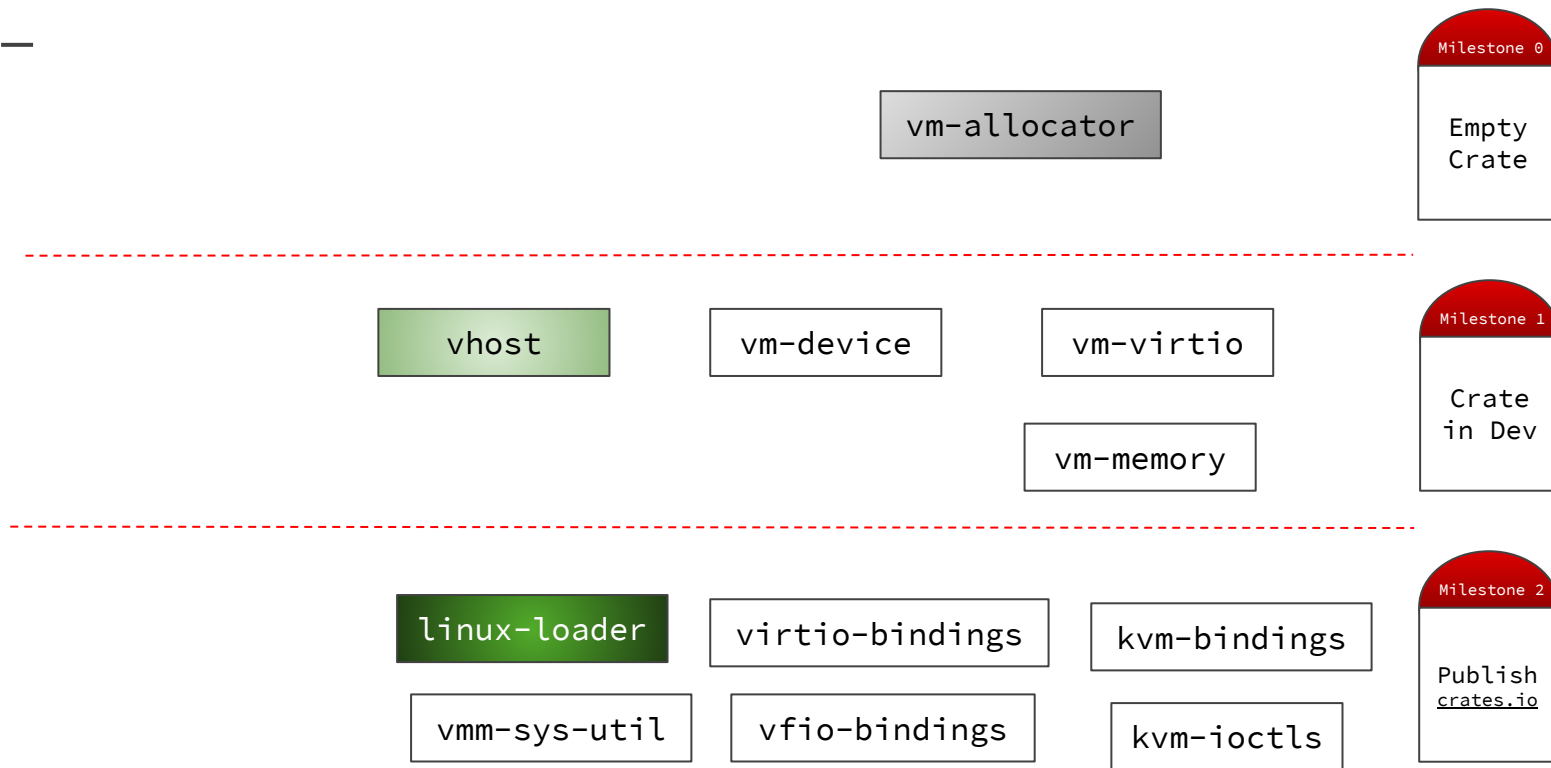
Status 2020



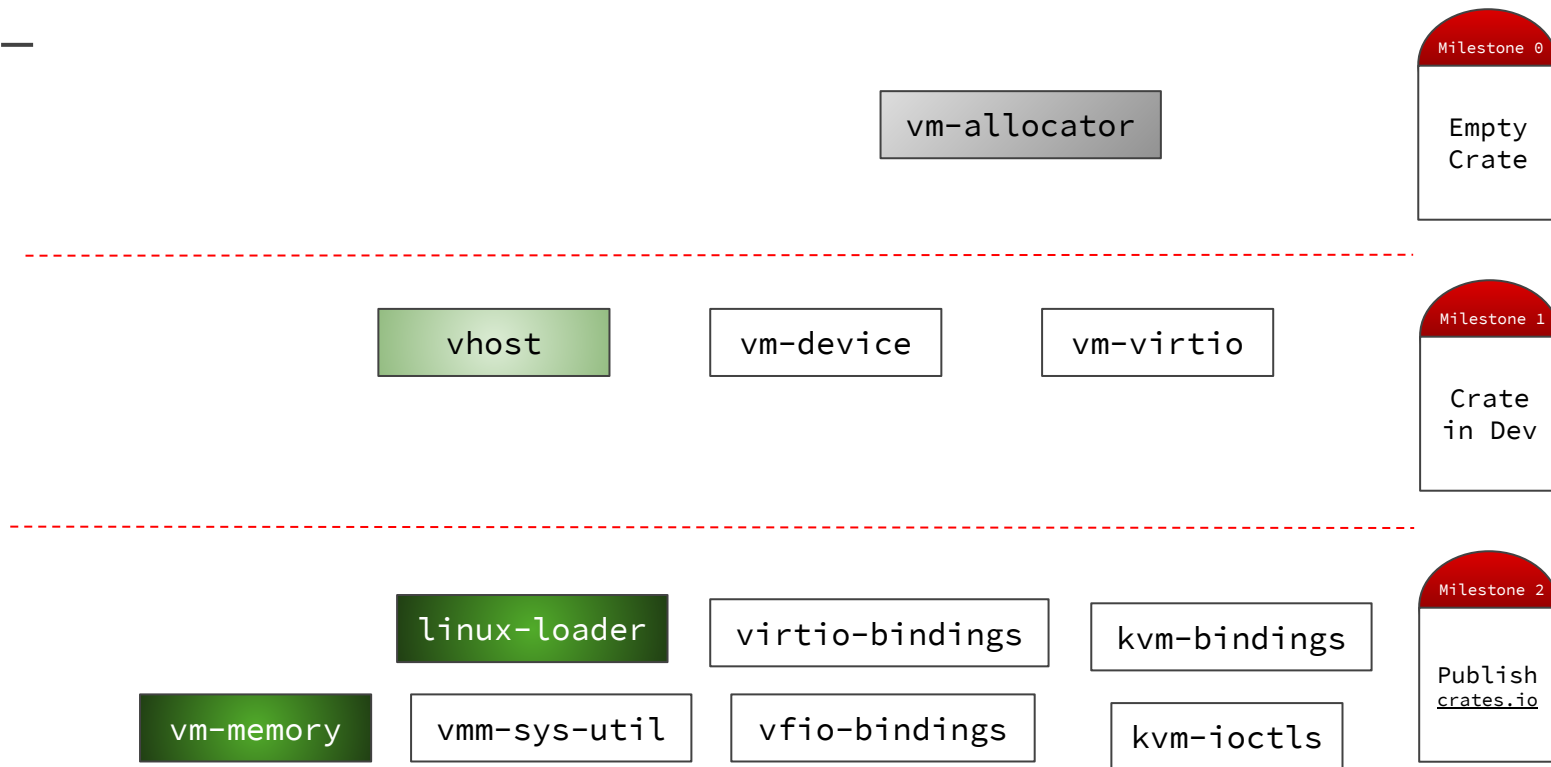
Status 2020



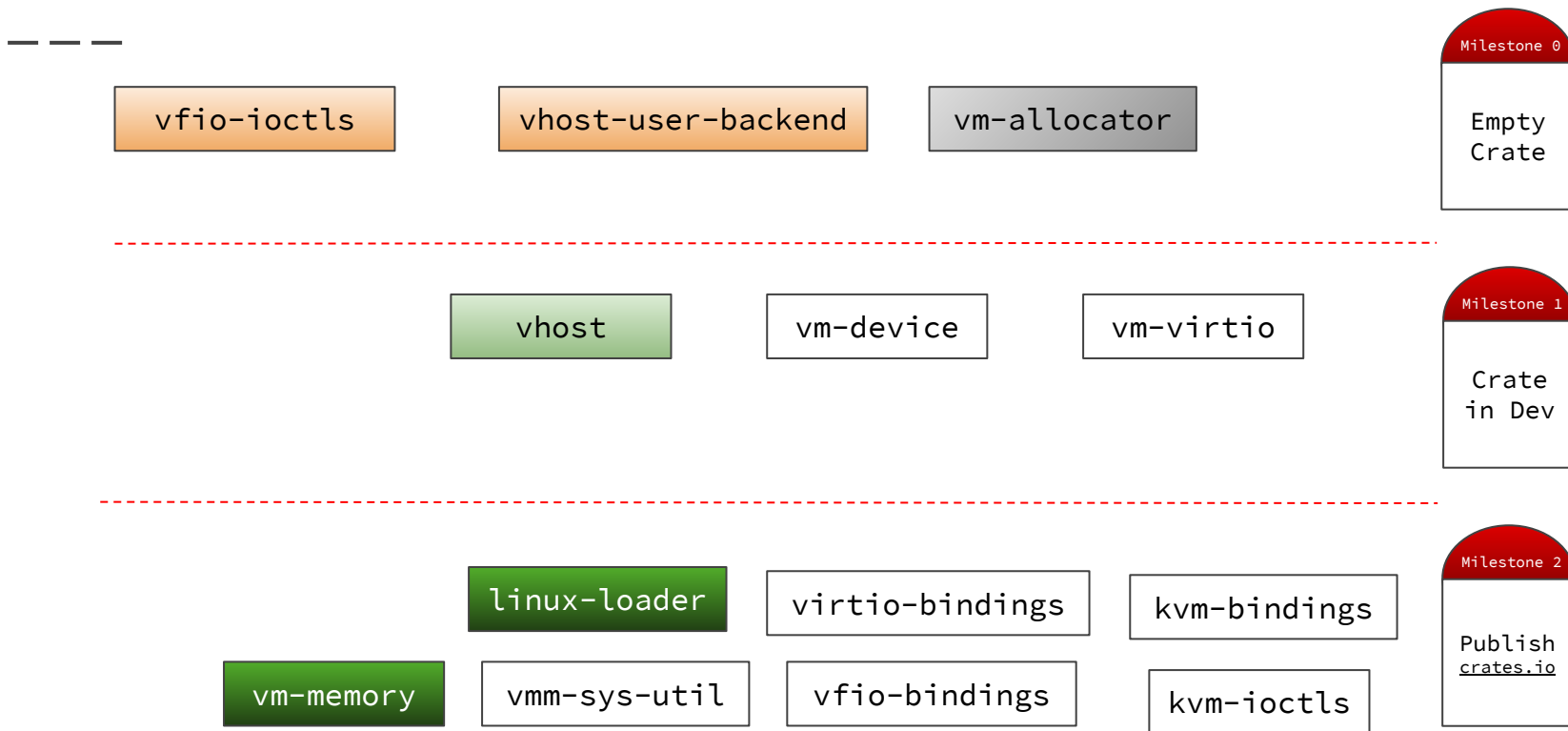
Status 2020



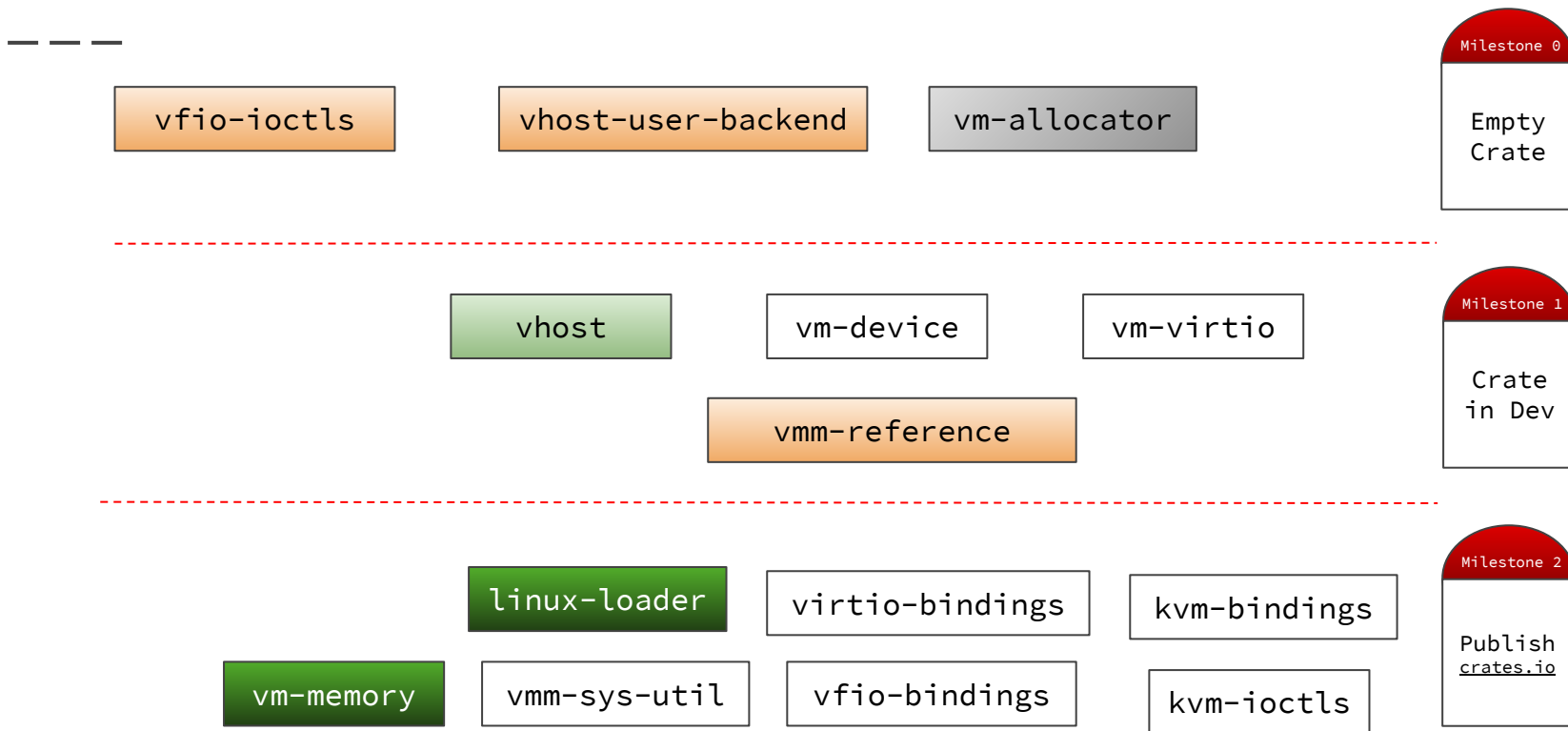
Status 2020



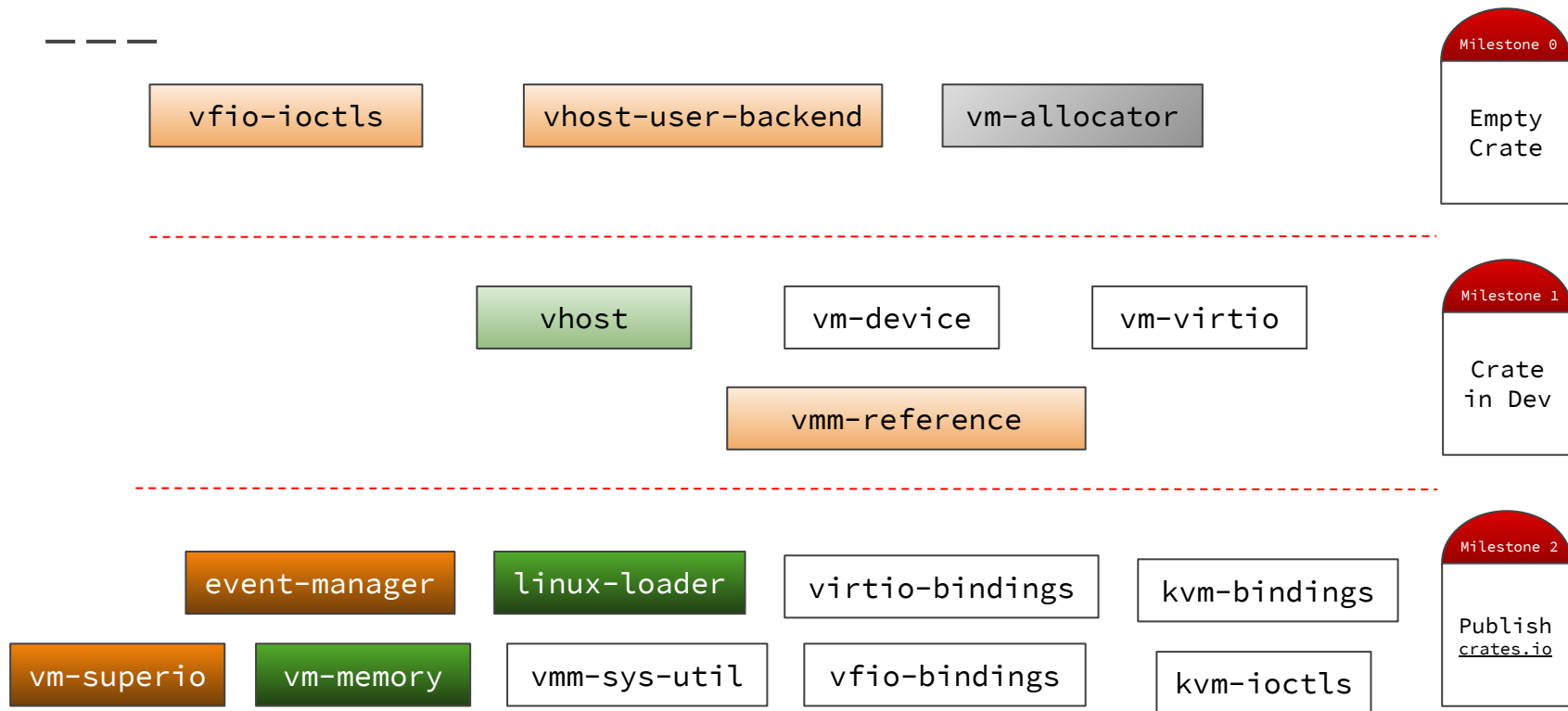
Status 2020 - New Development



Status 2020 - New Development



Status 2020 - New Development



Investment in Security & Testing

- Performance tests
 - Pipeline for running the tests
 - Benchmarks in: event-manager, linux-loader, vm-memory, vm-virtio
 - Next: infrastructure for nightly runs
- Security
 - Code audit
 - Next: threat model

Security Vulnerabilities

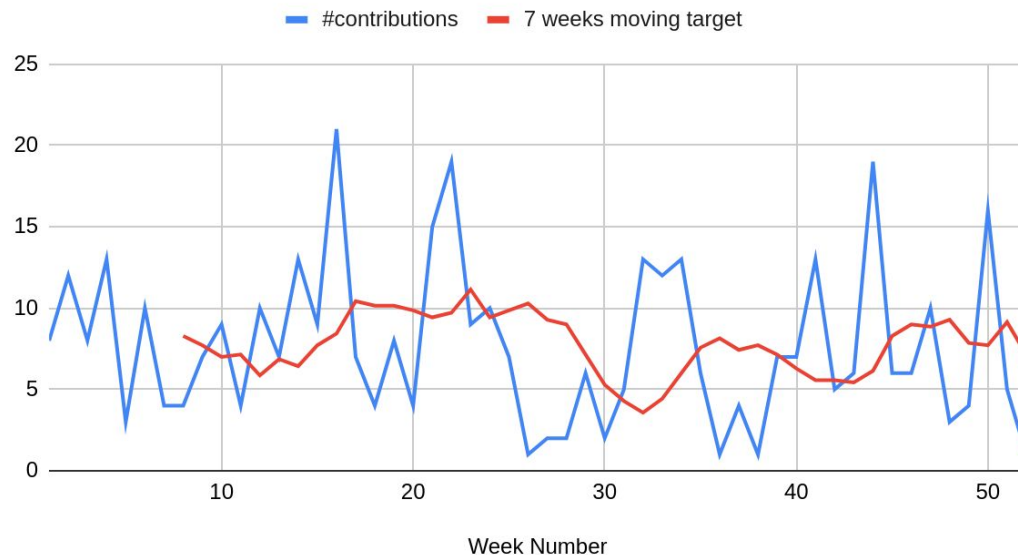
- `vm-memory`:
 - Non-atomic reads & writes
 - <https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2020-13759>
- `vm-superio`:
 - Serial console may allocate an unbounded amount of memory
 - CVE ID allocation in progress
 - <https://github.com/rust-vmm/vm-superio/issues/17>

Community

Code Contributions

- GitHub API
- Only code contributions :(
- All rust-vmm components

Contributions for last 52 weeks (as of 2020-10-12)



Start Contributing

- Find issues
 - [Label = “good first issue”](#)
 - [Label = “help wanted”](#)
 - [Issues without owners](#)
- Slack:
 - [rust-vmm channel](#)
- Email:
 - rust-vmm@lists.opendev.org

Future Investments

Community

- Gather feedback on contribution & consumption model
- Reporting security vulnerabilities

Virtualization Components

- vhost-user-backend
- Virtio
 - Block
 - Network
 - Vsock
- Vcpu Abstractions
- PCI
- VFIO
- Fuzzing for emulation code

Thank you!