



Using MicroVMs to Test Infrastructure as Code

Light-weight virtualization using Firecracker and Ignite

Who is Talking

Per Böhlin

Chief Software Architect

per.bohlin@syntheticmr.com

SyntheticMR

SyntheticMR's mission is to make quantitative imaging solutions the standard of care to support more precise diagnosis, reliable monitoring and personalized treatment strategies. Built on sustainability, innovation and trust, we strive to improve patient care worldwide.

Software for MRI

We are hiring!

Come talk to me or send me an email if you would be interested

The Problem

Testing System Provisioning and IaC

- Staging environments are not scalable
- Traditional VMs are not fast enough
- Docker in Docker offers too poor isolation

The Solution

MicroVMs

- Virtualization running on developer's machines
- Fast
- Complete isolation

Firecracker & Ignite

What is it:

- KVM based lightweight VMM
- Look and feel of Docker
- Works on “regular” Linux
- Works on WSL2 with custom kernel
- Uses containerd for networking
- Can use docker images as basis for VM file-system

What we like:

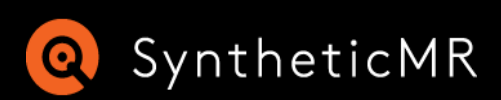
- Fast to run
- Fast to build VMs
- Simple to use

Live Demo

Show:

- Build VM using docker
- Run VM
- Run services
- Run automated tests





Questions?

Contact

per.bohlin@syntheticmr.com

www.syntheticmr.com
Linköping, Sweden

References

Firecracker & Ignite:

- <https://firecracker-microvm.github.io/>
- <https://github.com/weaveworks/ignite>

Quicker Ansible:

- https://mitogen.networkgenomics.com/ansible_detailed.html

Infrastructure Test Automation:

- <https://docs.pytest.org/en/7.2.x/>
- <https://testinfra.readthedocs.io/en/latest/>

WSL2 Setup

Requires Windows 11!

Build custom WSL2-kernel to enable features needed by ignite:

```
git clone git@github.com:ZenDevelopmentEcosystem/WSL2-Linux-Kernel.git
git checkout firecracker-ignite
make KCONFIG_CONFIG=Microsoft/config-wsl -j16
make KCONFIG_CONFIG=Microsoft/config-wsl -j16 bindeb-pkg
```

Save `arc/x86_64/boot/bzImage` to your home directory and reference it in `C:\Users\\.wslconfig`.

Turn on nested virtualization:

```
[wsl2]
kernel=C:\\Users\\<User>\\bzImage
nestedVirtualization=true
```

Install Debian packages using `dpkg -i ../...deb` if using a Debian-based WSL-distro.



We are hiring!

<https://syntheticmr.com/company/careers/>