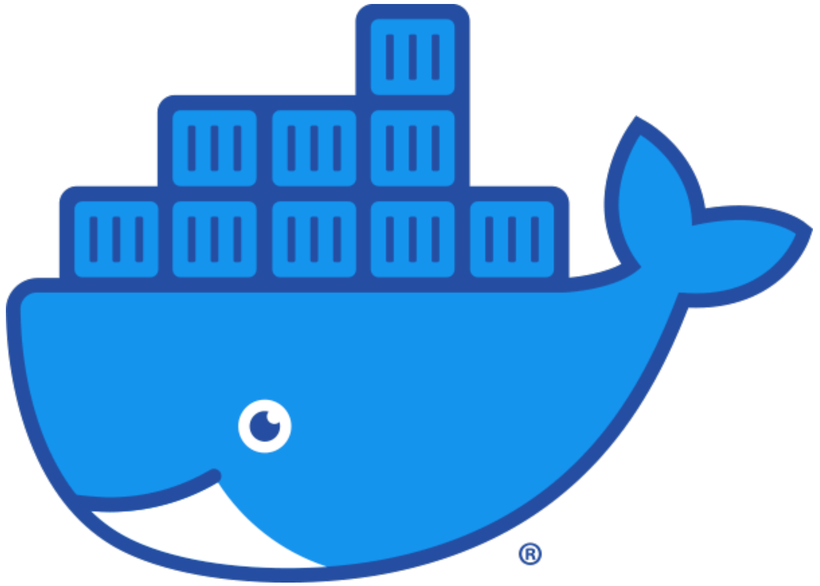
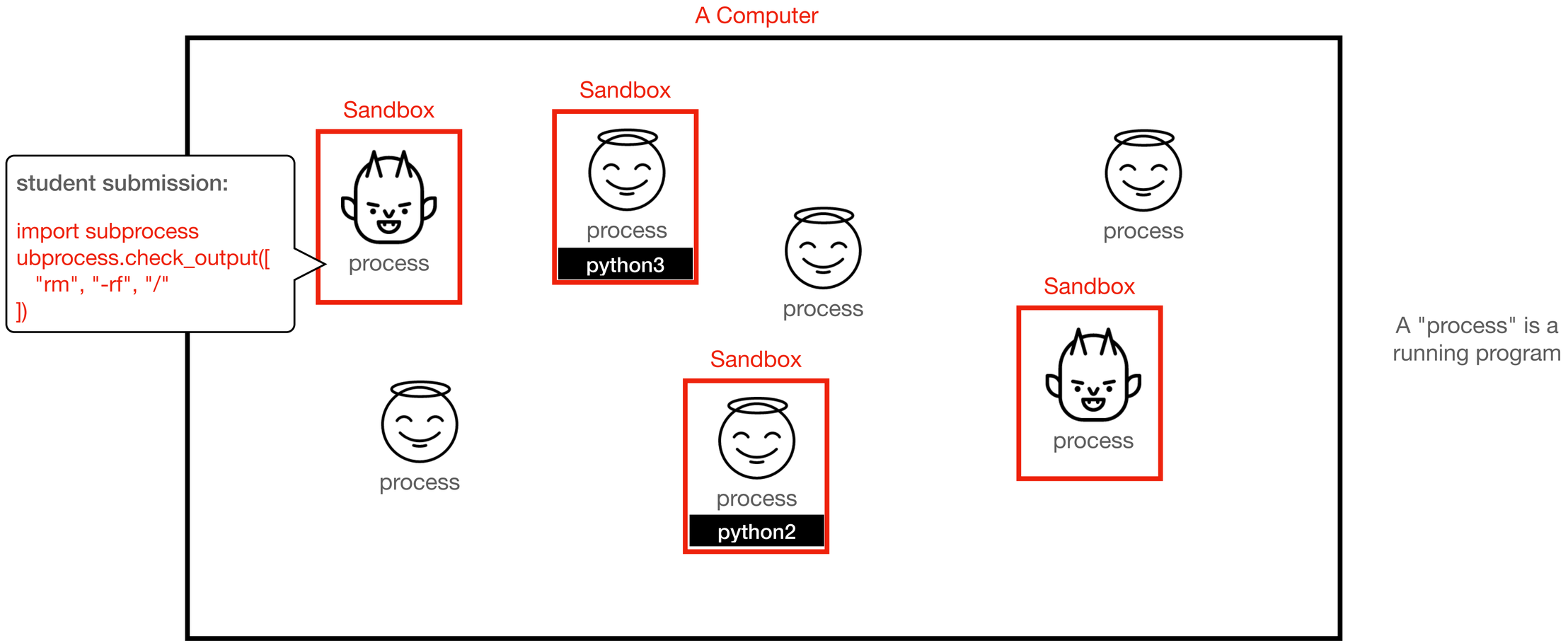
dotData Docker Workshop

Tyler Caraza-Harter



Sandboxing



A "process" is a running program

Sandboxing

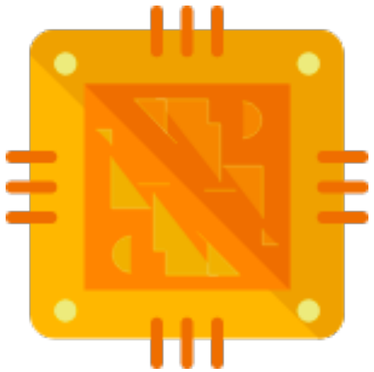
Most common sandboxing technologies:

1. Virtual Machines

2. Containers

But first, a little background...

CPUs, Operating Systems, and Kernels



a CPU's instruction set is the collection of operations it can perform

CPU Modes

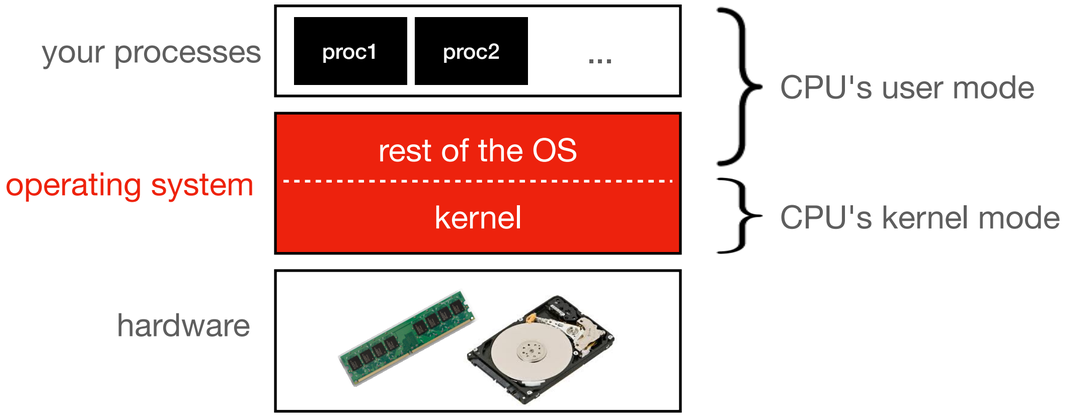
1. User mode (regular ops)
2. Kernel mode (privileged ops too)

Original 8086/8088 instruction set

Instruction	Meaning	Notes	Opcode
IN	Input from port	(1) `AL = port[imm];` (2) `AL = port[DX];` (3) `AX = port[imm];` (4) `AX = port[DX];`	0xE4, 0xE5, 0xEC, 0xED
INC	Increment by 1		0x40…0x47, 0xFE/0, 0xFF/0
INT	Call to interrupt		0xCC, 0xCD
INTO	Call to interrupt if overflow		0xCE
IRET	Return from interrupt		0xCF
Jcc	Jump if condition	`(JA, JAE, JB, JBE, JC, JE, JG, JGE, JL, JLE, JNA, JNAE, JNB, JNBE, JNC, JNE, JNG, JNGE, JNL, JNLE, JNO, JNP, JNS, JNZ, JO, JP, JPE, JPO, JS, JZ)`	0x70…0x7F, 0x0F80… 0x0F8F (since 80386)

…

https://en.wikipedia.org/wiki/X86_instruction_listings



Sandboxing

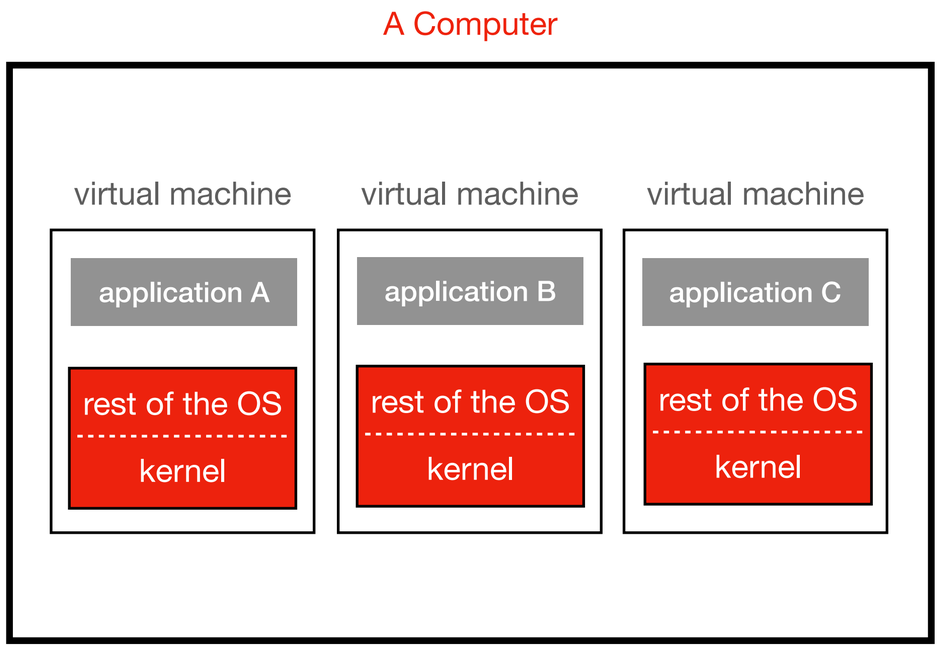
Most common sandboxing technologies:

1. Virtual Machines

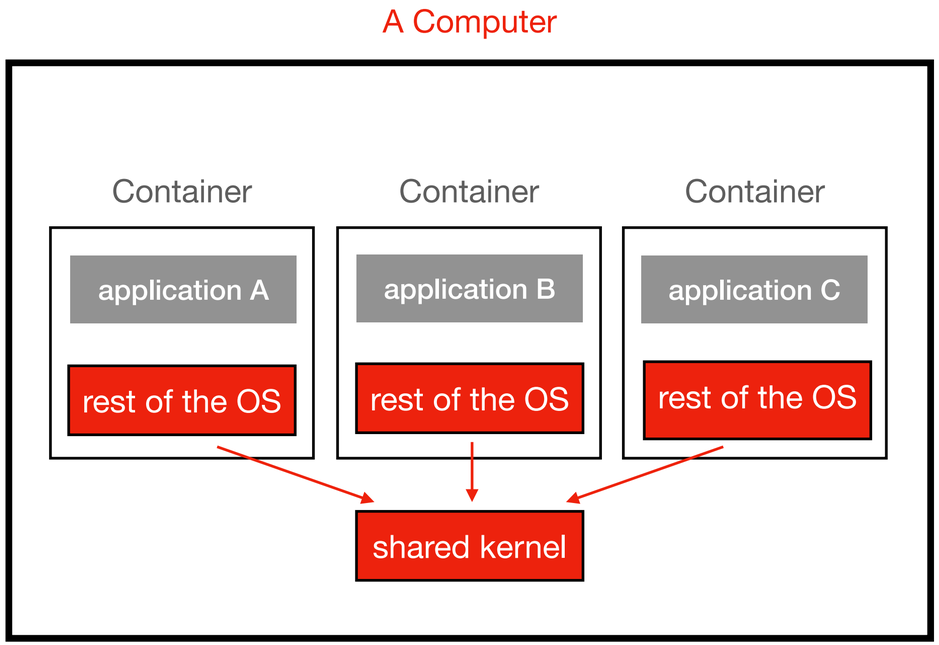
2. Containers

Now, with that quick background...

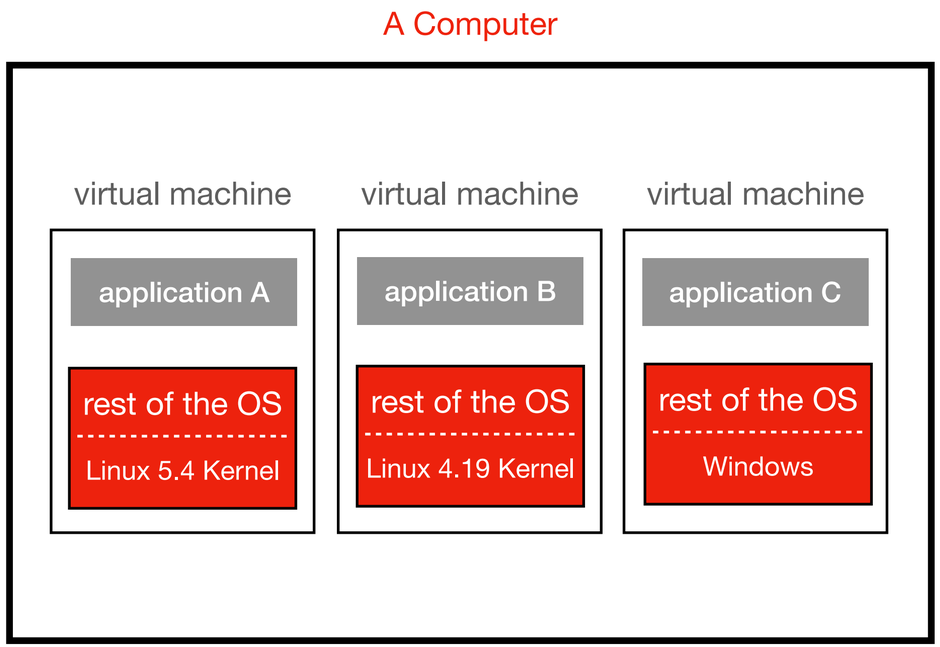
Virtual Machines



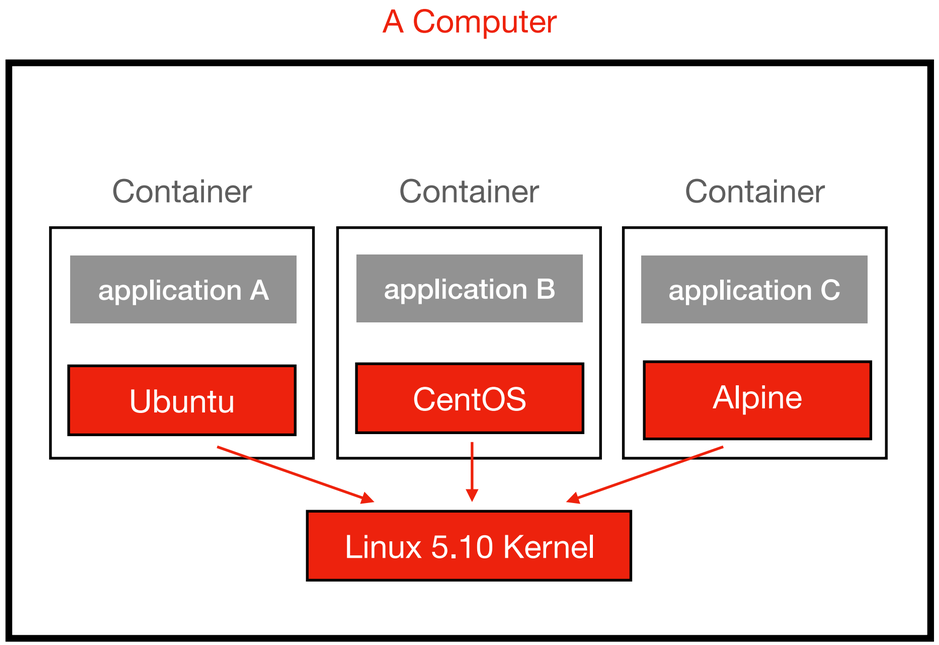
Containers



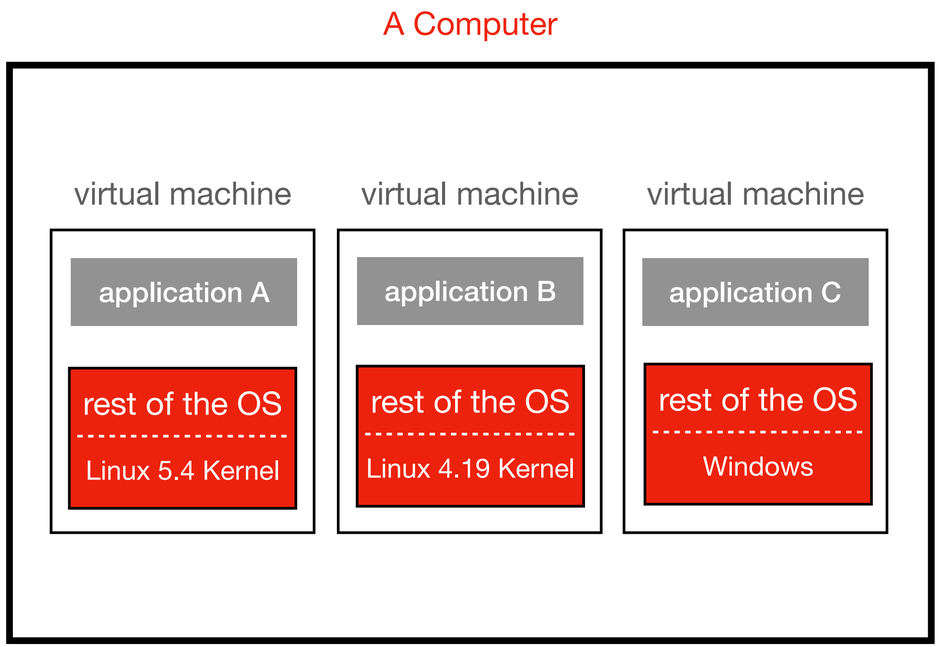
Virtual Machines



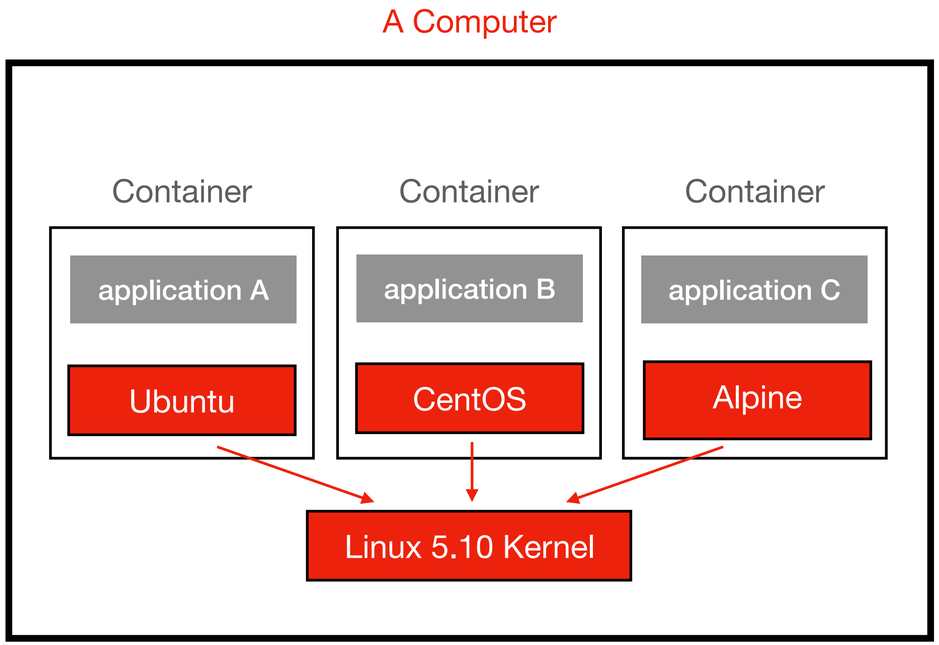
Containers

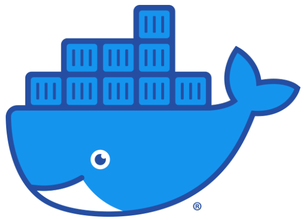


Virtual Machines



Containers





Docker makes it very easy to use these container features

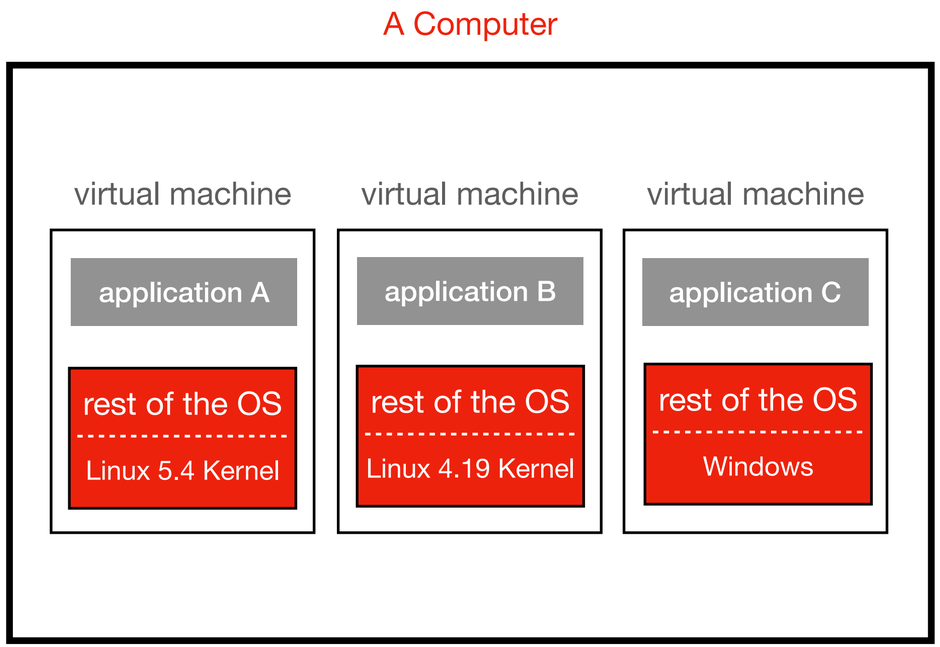
Pros

1. really fast
2. low memory overhead

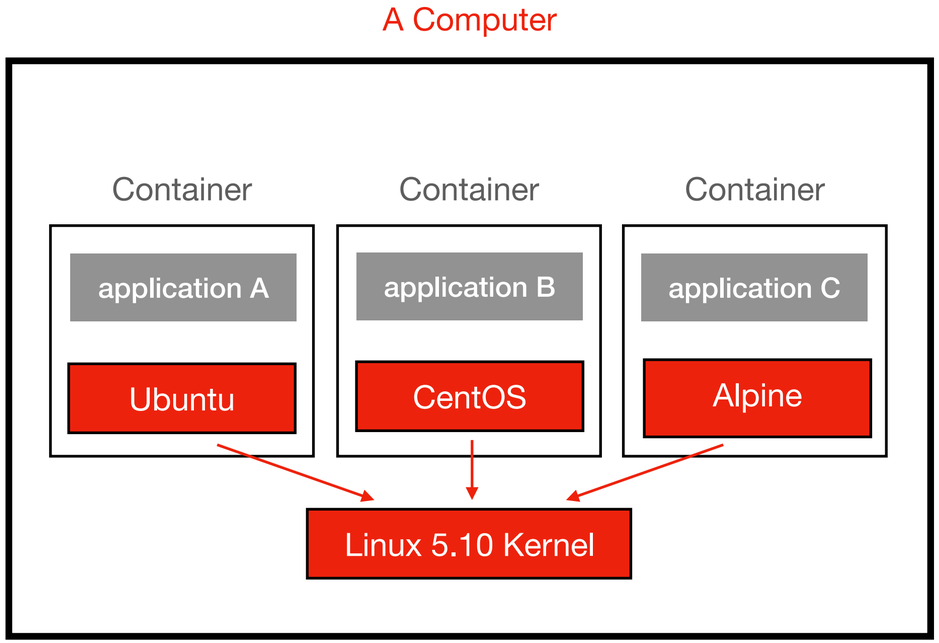
Cons

1. inflexible (all containers have same kernel version
2. kernel needs to support container features (**Linux** has cgroups, namespaces, seccomp, etc)
3. sharing kernel is a security risk

Virtual Machines



Containers



Pros

1. really fast
2. low memory overhead

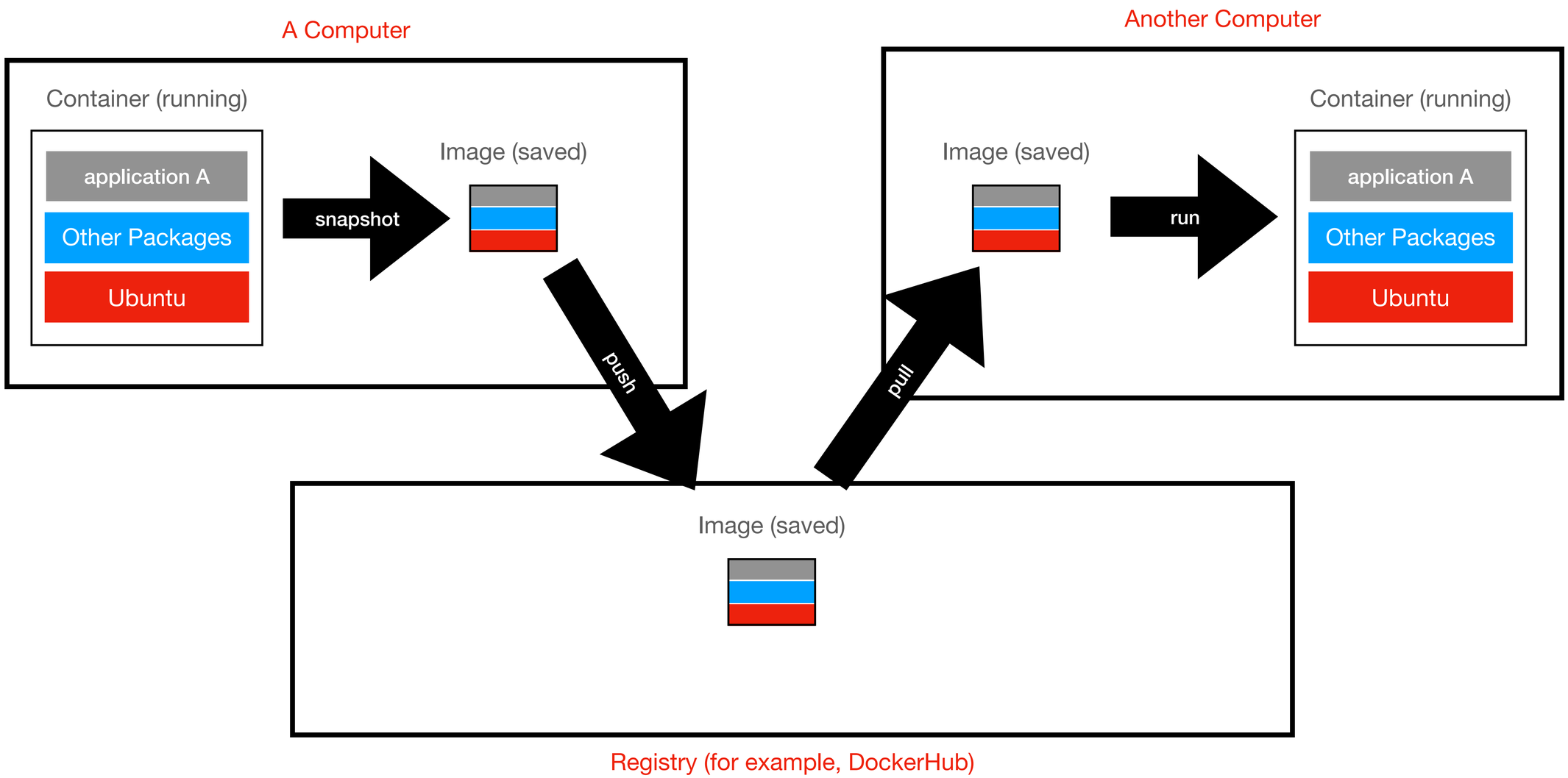
Cons

1. inflexible (all containers have same kernel version
2. kernel needs to support container features (**Linux** has cgroups, namespaces, seccomp, etc)
3. sharing kernel is a security risk

Even if you use "Docker for Mac" or "Docker for Windows", you're really using Linux

https://nickjanetakis.com/blog/should-you-use-the-docker-toolbox-or-docker-for-mac-windows

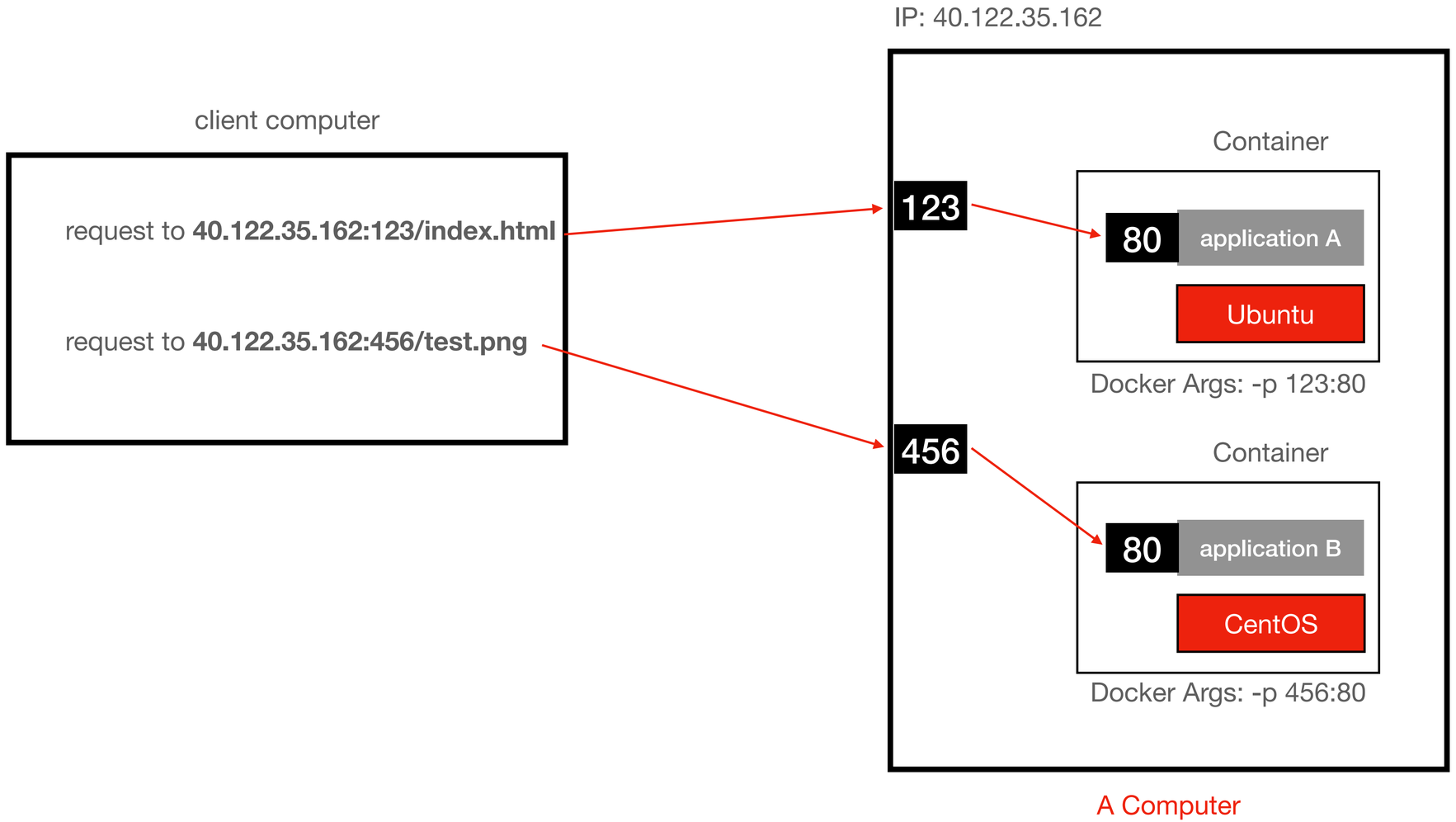
Docker Makes Reproducibility Much Easier



Demos...

https://tyler.caraza-harter.com/workshops/docker-s21/overview.html

Ports Mapping



Thank You!
Any questions?