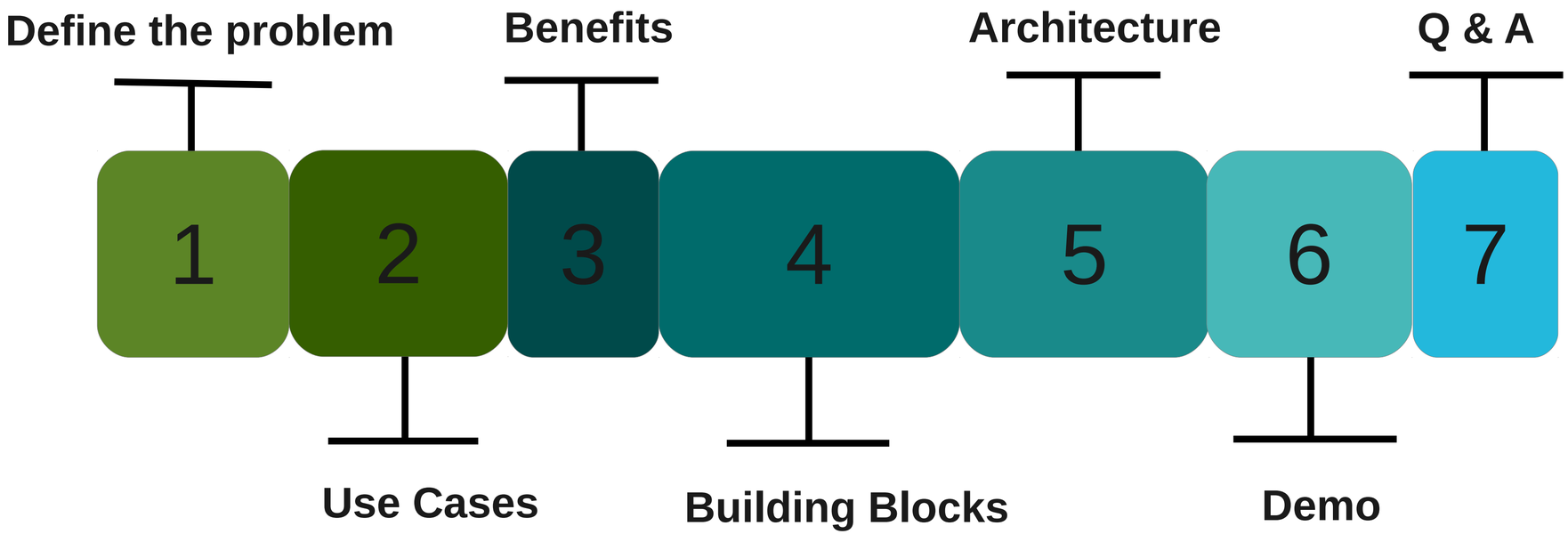
RED HAT SUMMIT
10 YEARS and counting
SAN FRANCISCO | APRIL 14-17, 2014

Linux Containers Roadmap
Red Hat Enterprise Linux 7 RC

Bhavna Sarathy
Senior Technology Product Manager, Red Hat

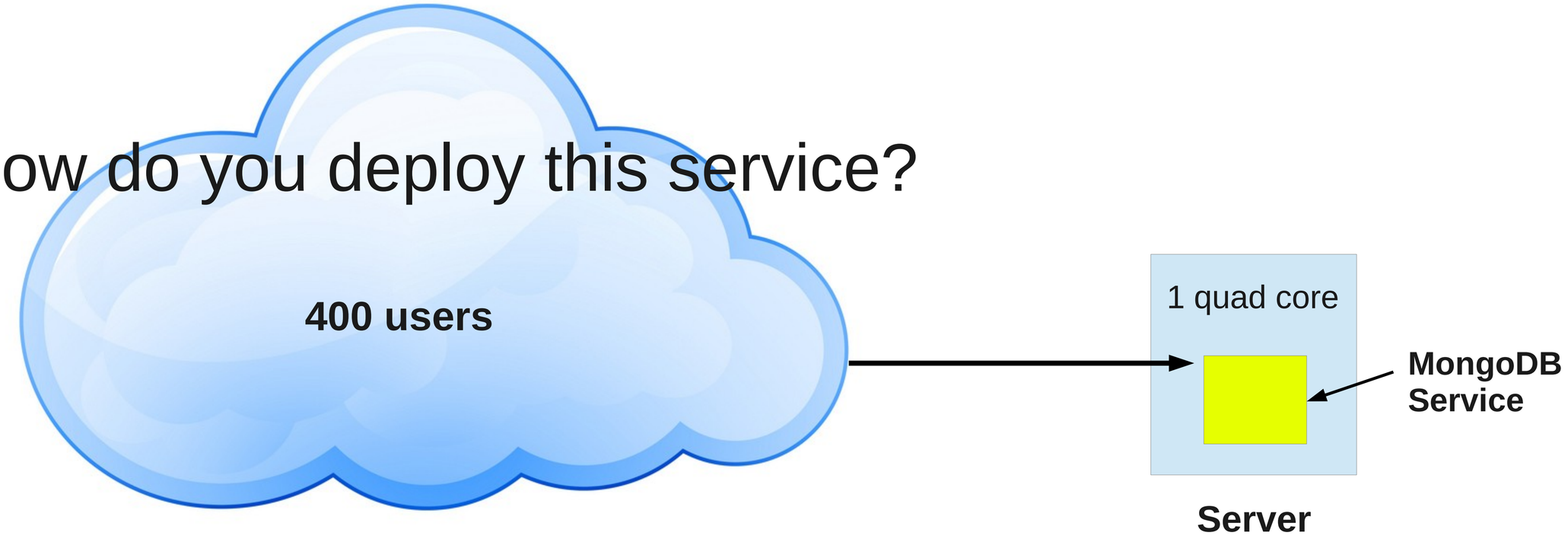
Linda Wang
Senior Eng. Manager, Red Hat

Bob Kozdemba
Principal Soln. Architect, Red Hat


Define the problem
Benefits
Architecture
Q & A
1
2
3
4
5
6
7
Use Cases
Building Blocks
Demo

Defining the problem space

How do you deploy this service?

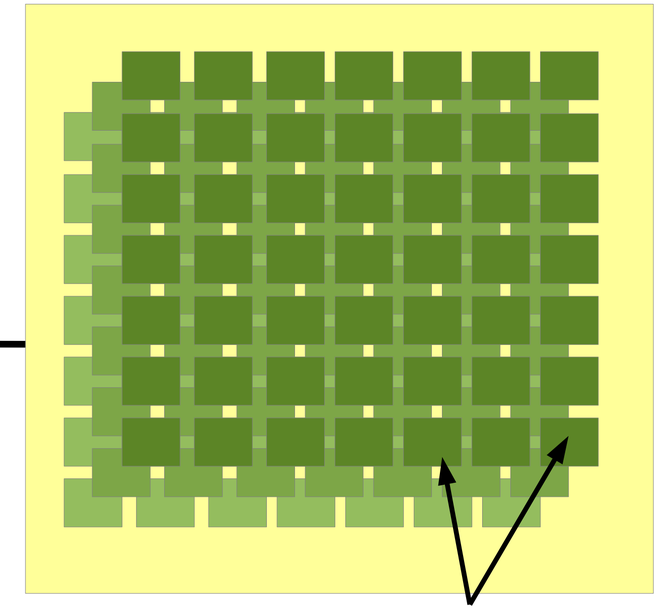


Defining the problem space

How do you deploy this service?

400,000 users

Server w/ 4000 cores



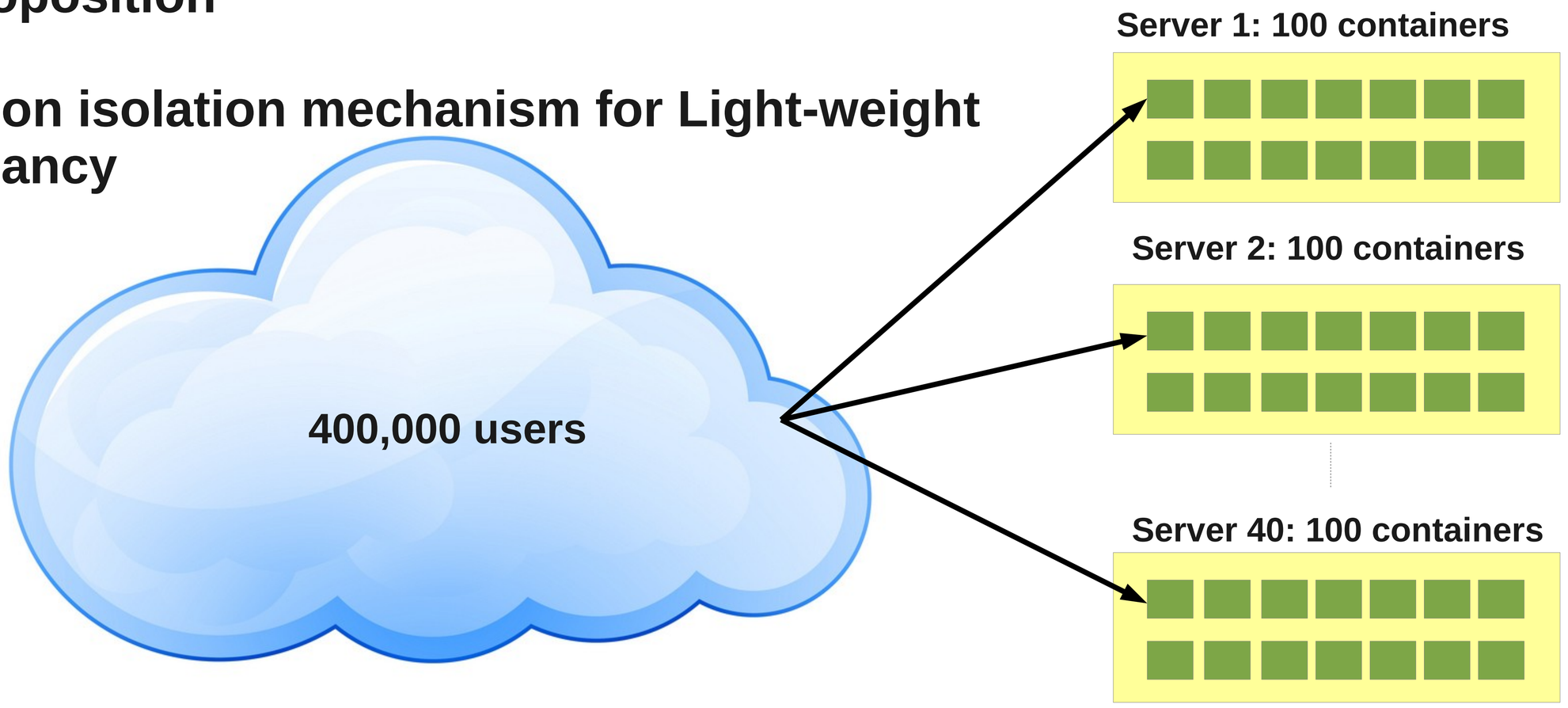
MongoDB Services

Containerize!

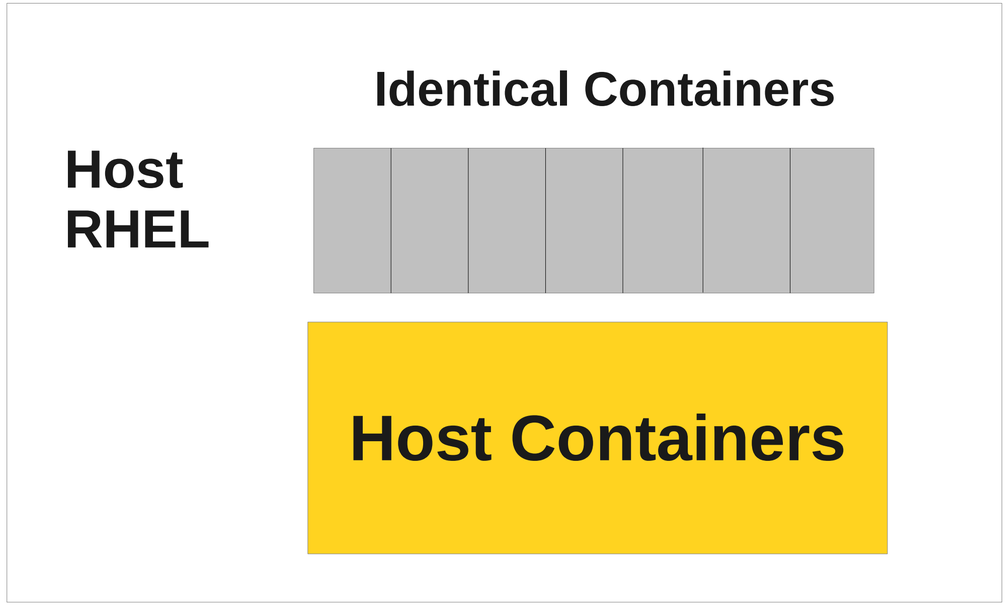
How about Red Hat Linux Containers?

Value Proposition

Application isolation mechanism for Light-weight multi-tenancy



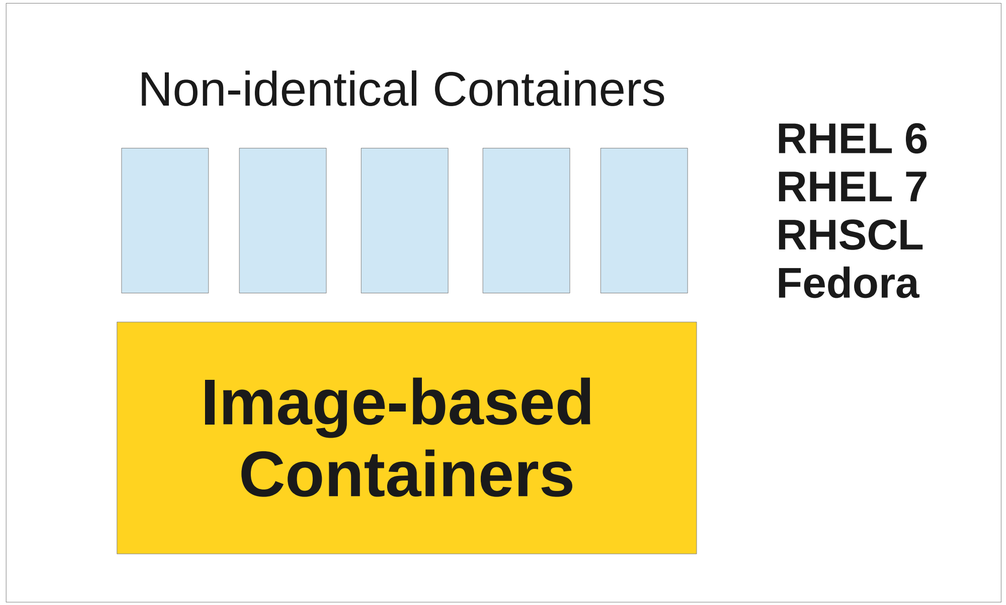
RHEL 7 Linux Containers Use Case 1
Host Containers



- RHEL 7 host carved into secure containers
- Each container running RHEL 7 userspace
- Pro : Security erratas can be applied easily with “yum update”
- Con : Limited to RHEL 7 runtimes

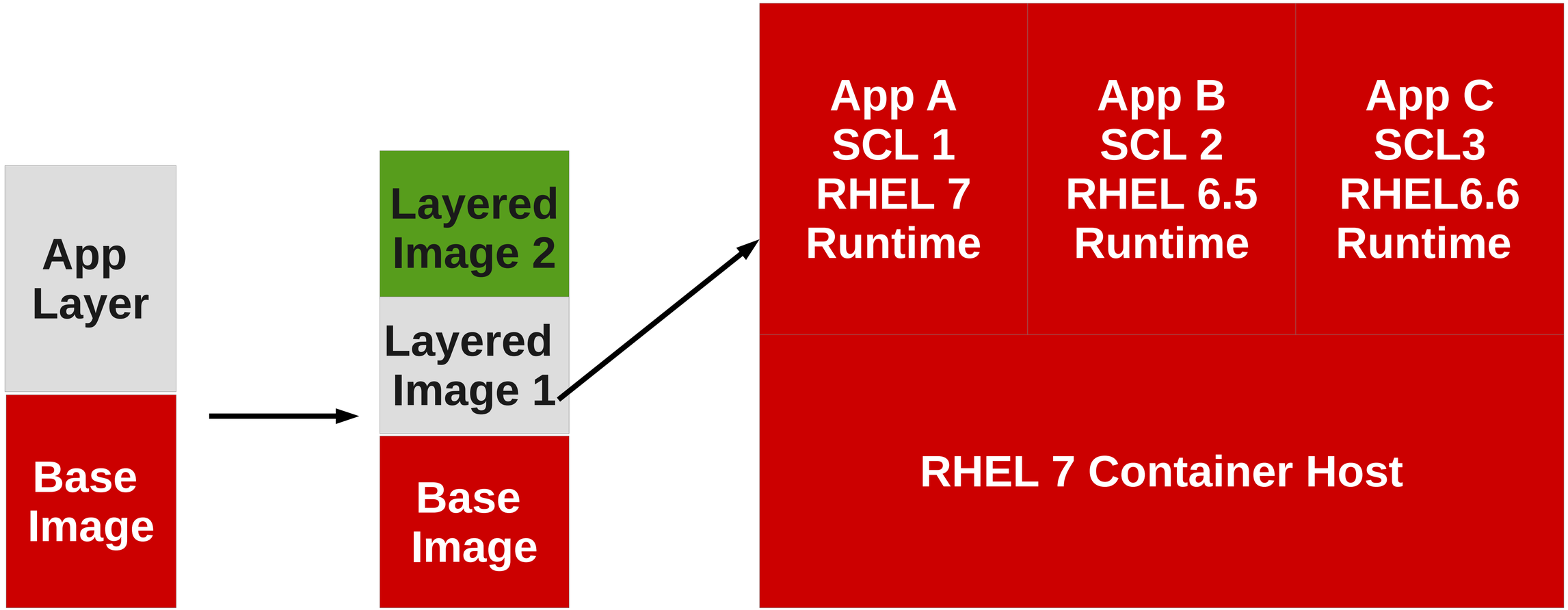
RHEL 7 Linux Containers Use Case 2
Image-based Containers

Docker format



- Docker builds on Linux Containers and provides format for content distribution
- Docker includes the userspace runtime of an application

Image-based Containers with Docker technology



docker

September 19, 2013

RED HAT AND DOCKER COLLABORATE

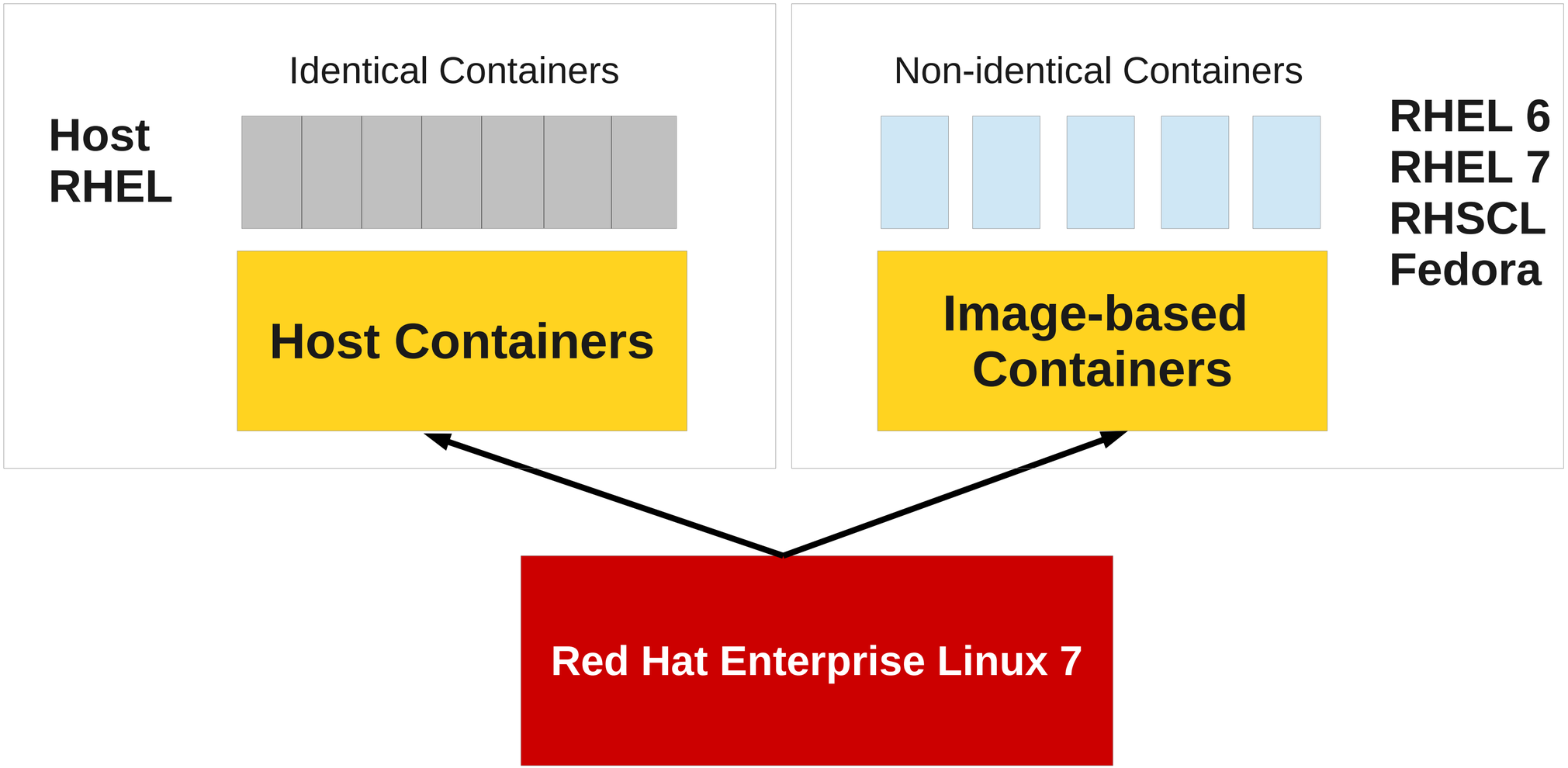
We are thrilled to announce the collaboration between Docker and Red Hat.

Collaboration with Red Hat is important for a number of reasons, including:

- Driving compatibility with the most widely deployed Linux distributions
- Enabling integration with one of the most prominent and important PaaS solutions
- Collaborating with the most prominent, pure open source company

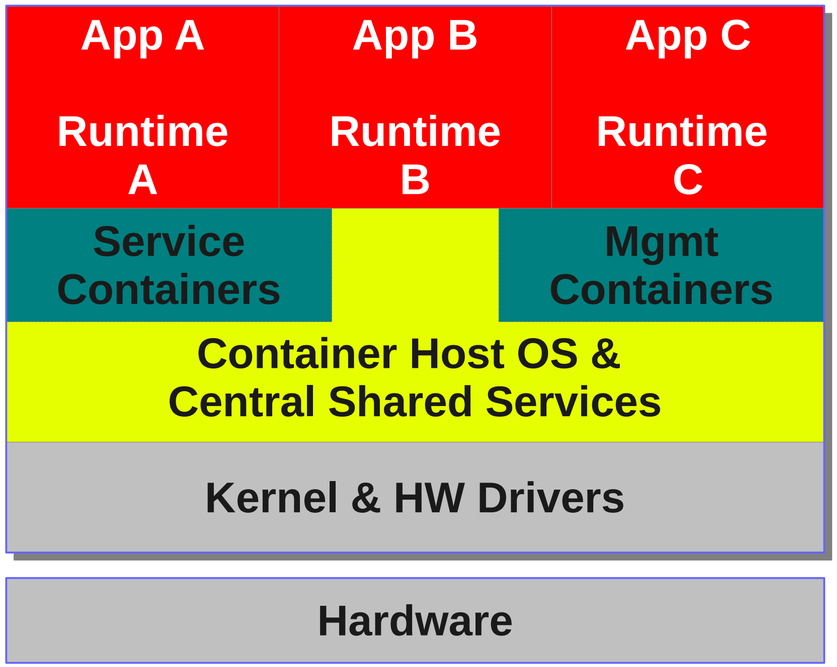
First, it is critically important for us to make Docker work seamlessly with Red Hat Enterprise Linux and related Linux distributions, such as Fedora. This is the #1 requested enhancement for Docker, and is obviously a major concern for people who want to deploy Docker in mainstream production environments. Our teams have been working together to package Docker for Fedora in time for the next release of Docker (0.7). Red Hat and dotCloud are planning to make Docker available for all Fedora users with upcoming releases,

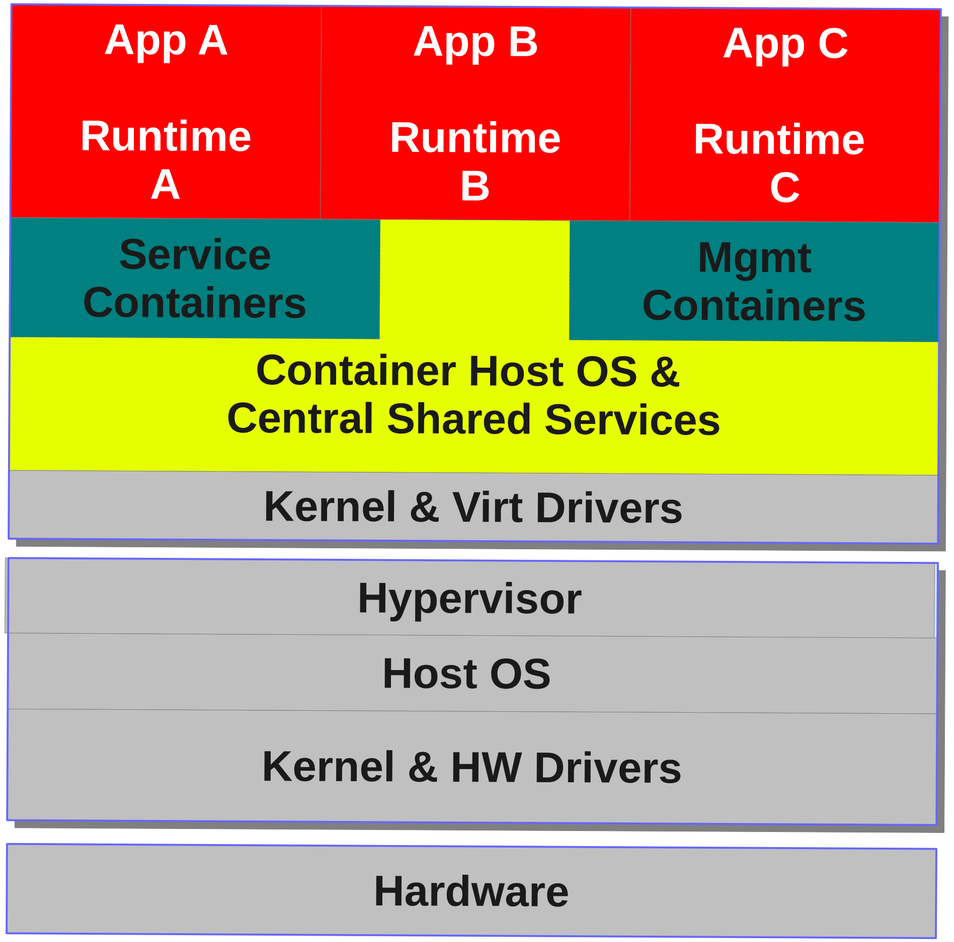
Linux Containers in Red Hat Enterprise Linux 7



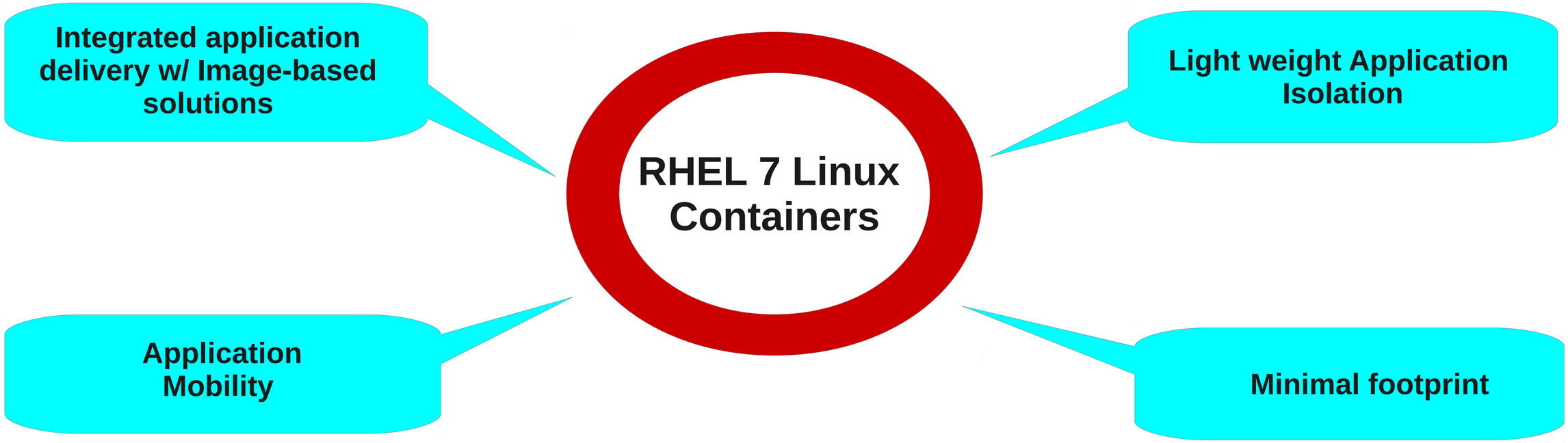
RHEL 7 Deployment Models

- Containers can be deployed in baremetal or virtual
- RHEL 7 supports both Virtualization with KVM and Linux Containers



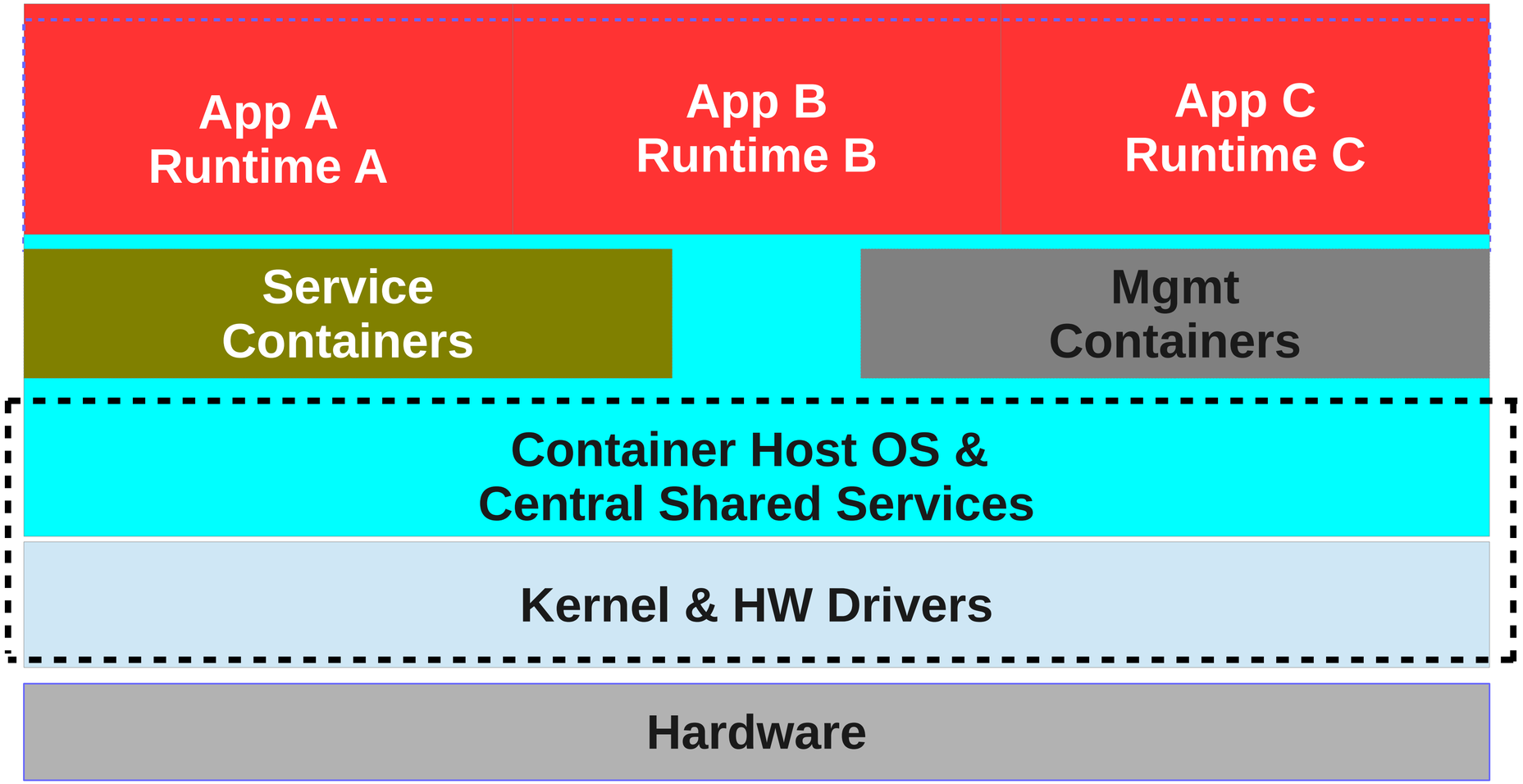


RHEL 7 Linux Containers Benefits

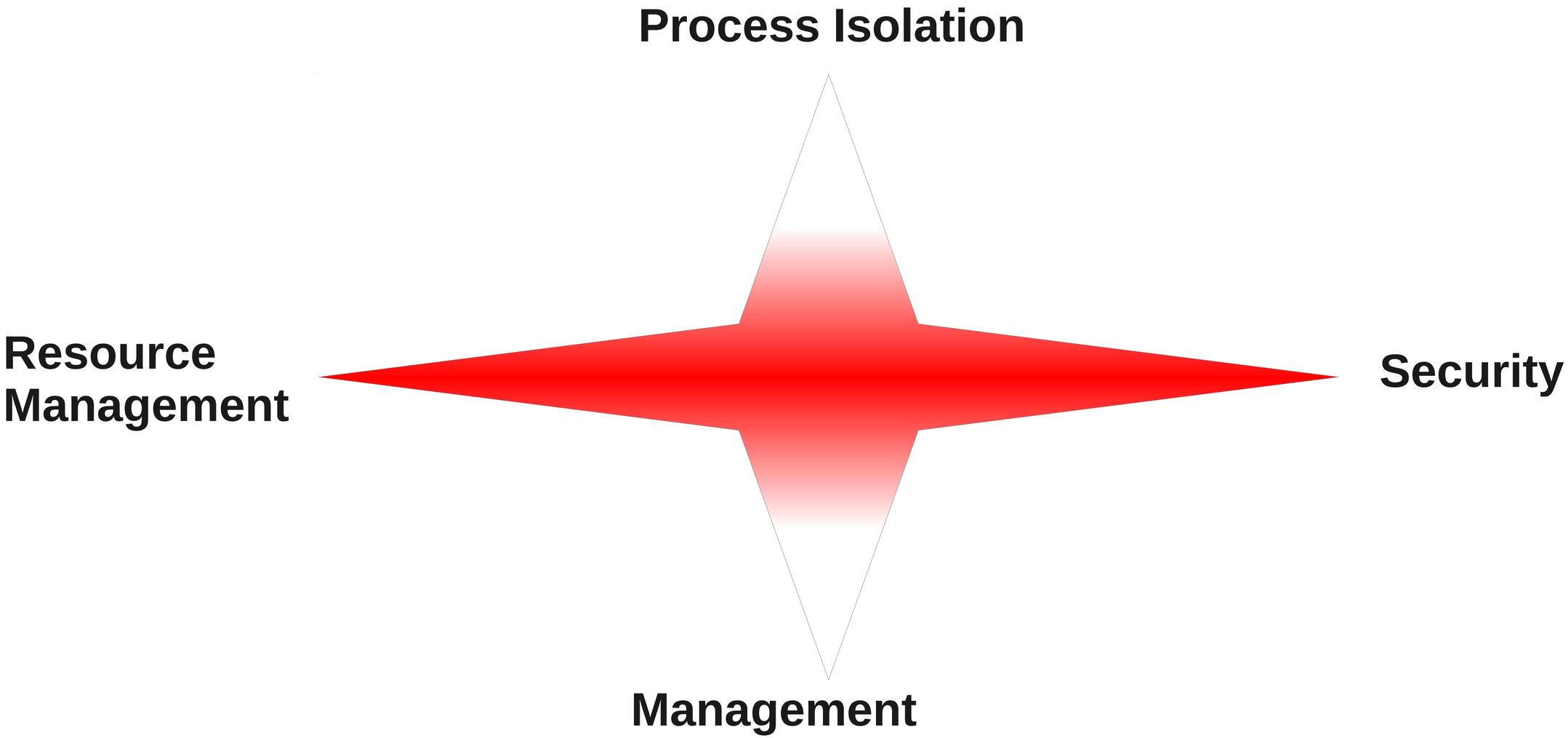


Simplified application delivery

RHEL 7 Container Host



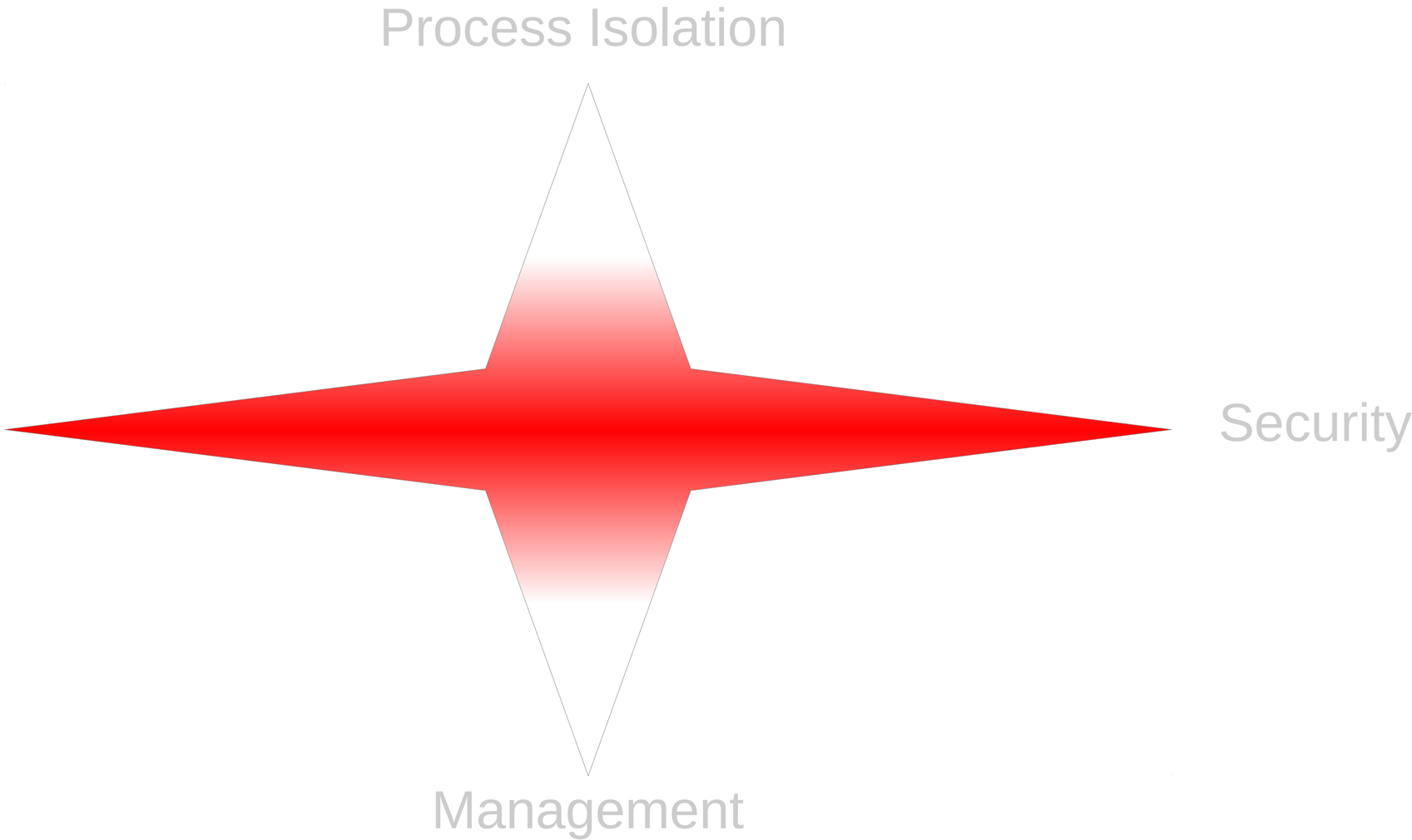
RHEL 7 Linux Containers - Building Blocks



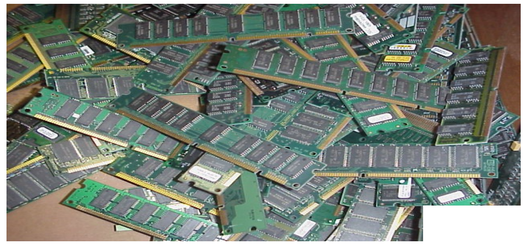
RED HAT SUMMIT
10 YEARS and counting
SAN FRANCISCO | APRIL 14-17, 2014

Linux Containers Features and Architecture

Linux Containers – Resource Management

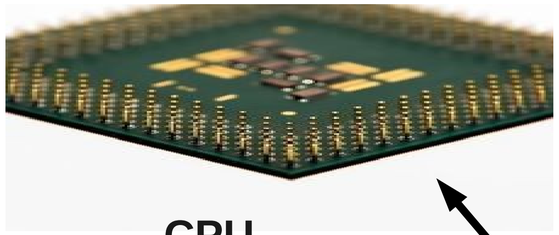


Resource Management – Control Groups



Memory

Network



CPU

Block IO

Cgroups

Linux Kernel

Hardware (Intel, AMD)

Control Groups – In a Container Environment

Control Groups – Resource Control

Control Groups - Usability Improvements

- RHEL6 Any privileged process can manage Cgroups
  - No coordination and with unexpected results
  - Kernel moving to single writer mode
  - Kernel does not enforce this yet...
- RHEL7 systemd will manage cgroups
  - Recommended to use systemd APIs in RHEL7
  - New concept of Scopes/Slices

https://www.youtube.com/watch?v=MSG4jW187Is

http://www.freedesktop.org/wiki/Software/systemd/ControlGroupInterface

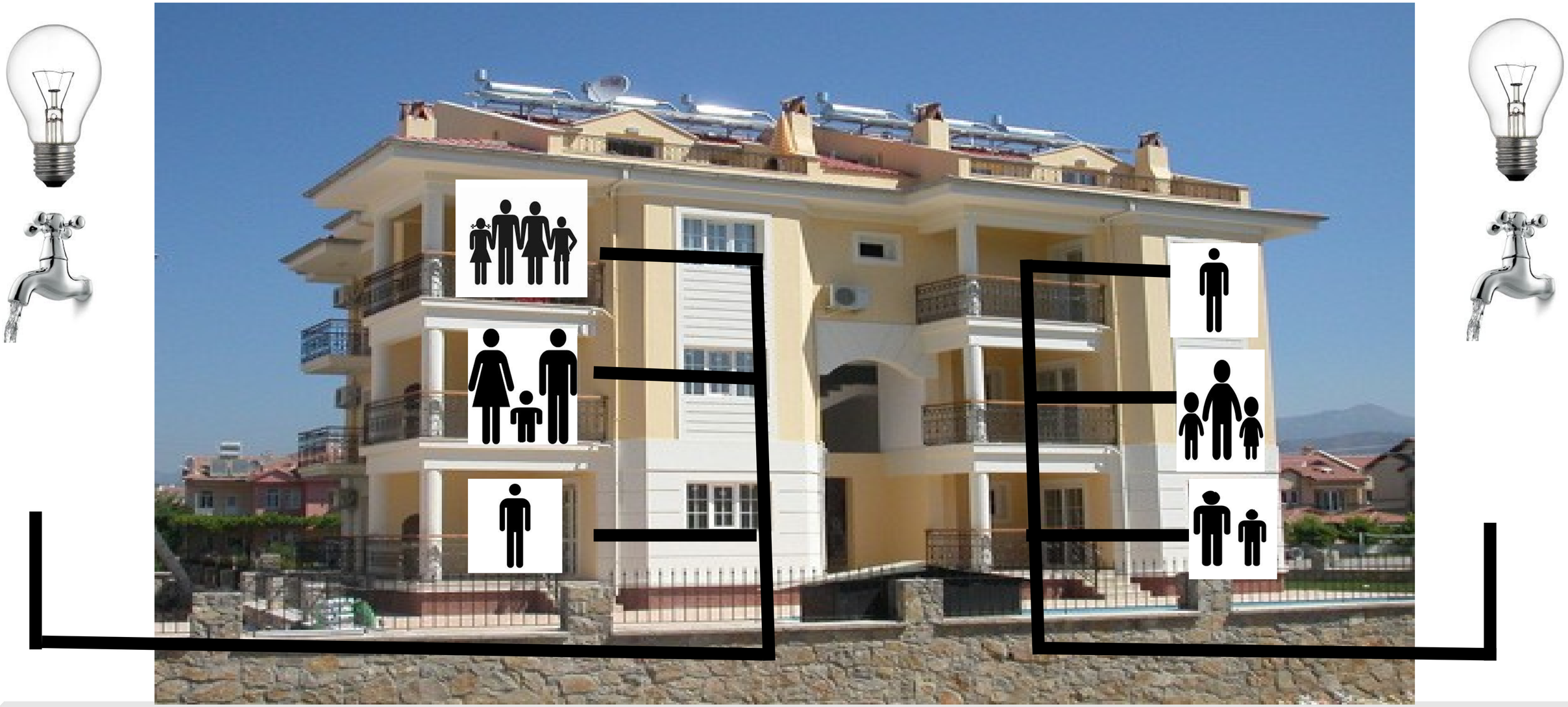
Control Groups - Usability Improvements: Scopes

Systemd puts all related worker PIDs into cgroup called a ‘scope’.

- Services
  - Apache processes in same services/apache scope
  - Mysql processes in same services/Mysql scope
  - Apache/Mysql get an equal “slice” of the system
- Users accounts
  - All users get an equal “slice”
- Machines
  - All containers/VMs get an equal “slice”
- No service/user/machine can dominate system

Control Groups – Systemd's "Scope"

Control Groups – Systemd's "Slice"



Control Groups - Usability Improvements: Slices

Special unit file for assigning resource constraints

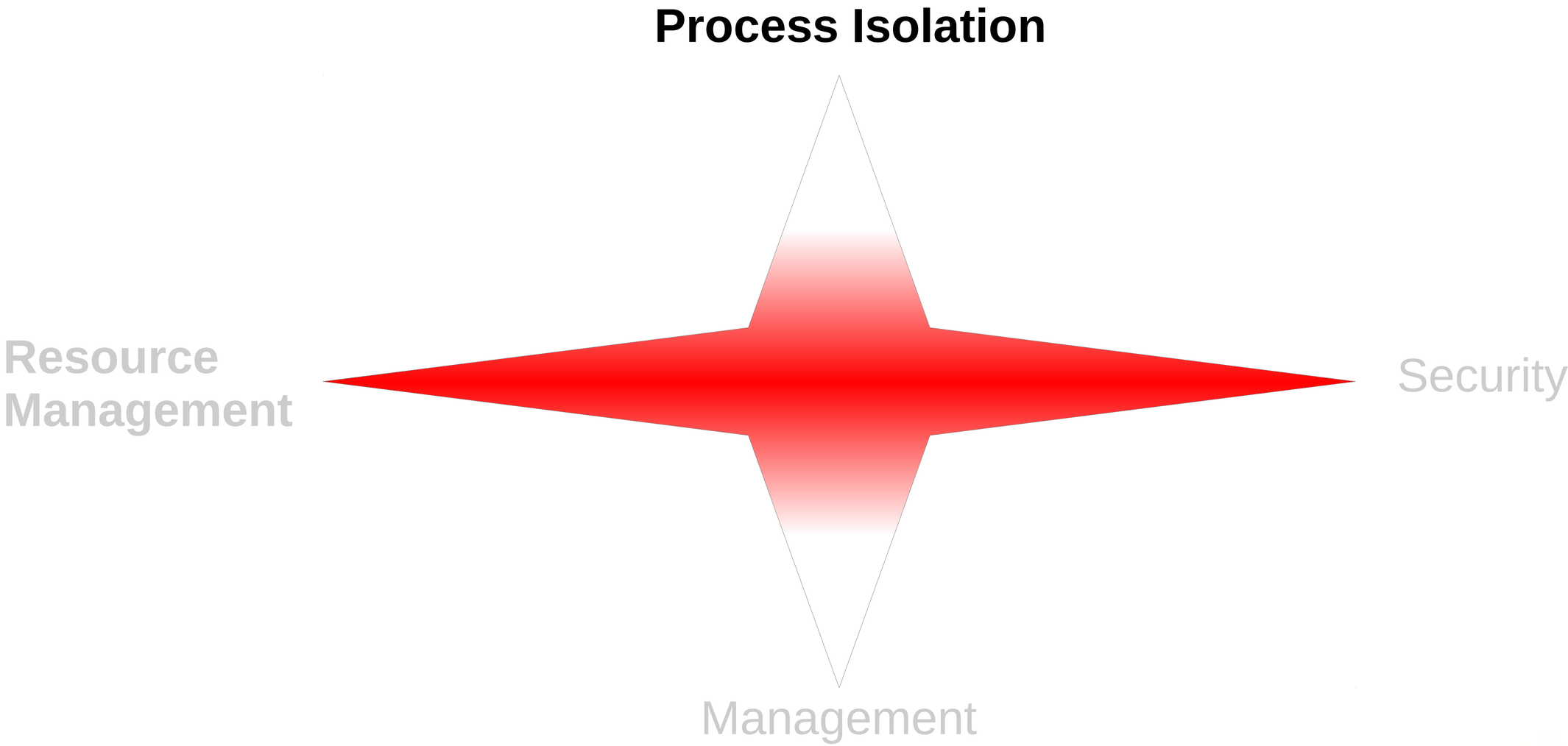
Slices get assigned to scopes

- Systemd automatically assigns services to system.slice
- You can override resource with Unit file configuration
  - MemoryLimit=1g
- Command Line

```
#> systemctl set-property httpd.service CPUShares=524 MemoryLimit=500M
```

- Systemd will assign Containers to machine.slice
  - You can override by editing
  - /etc/systemd/system/big-machine.slice

Linux Containers – Process Isolation



Process Isolation - Namespaces

Process Isolation - Namespaces

- Isolate processes
  - Create a new environment with a Subset of the resources
- Once set up, namespaces are transparent for processes
- Can be used in custom and complex scenarios
- Supported Namespaces
  - ipc, pid, mnt, net, uts
  - Future Red Hat Enterprise Linux 7: user namespace

Process Isolation - Namespaces

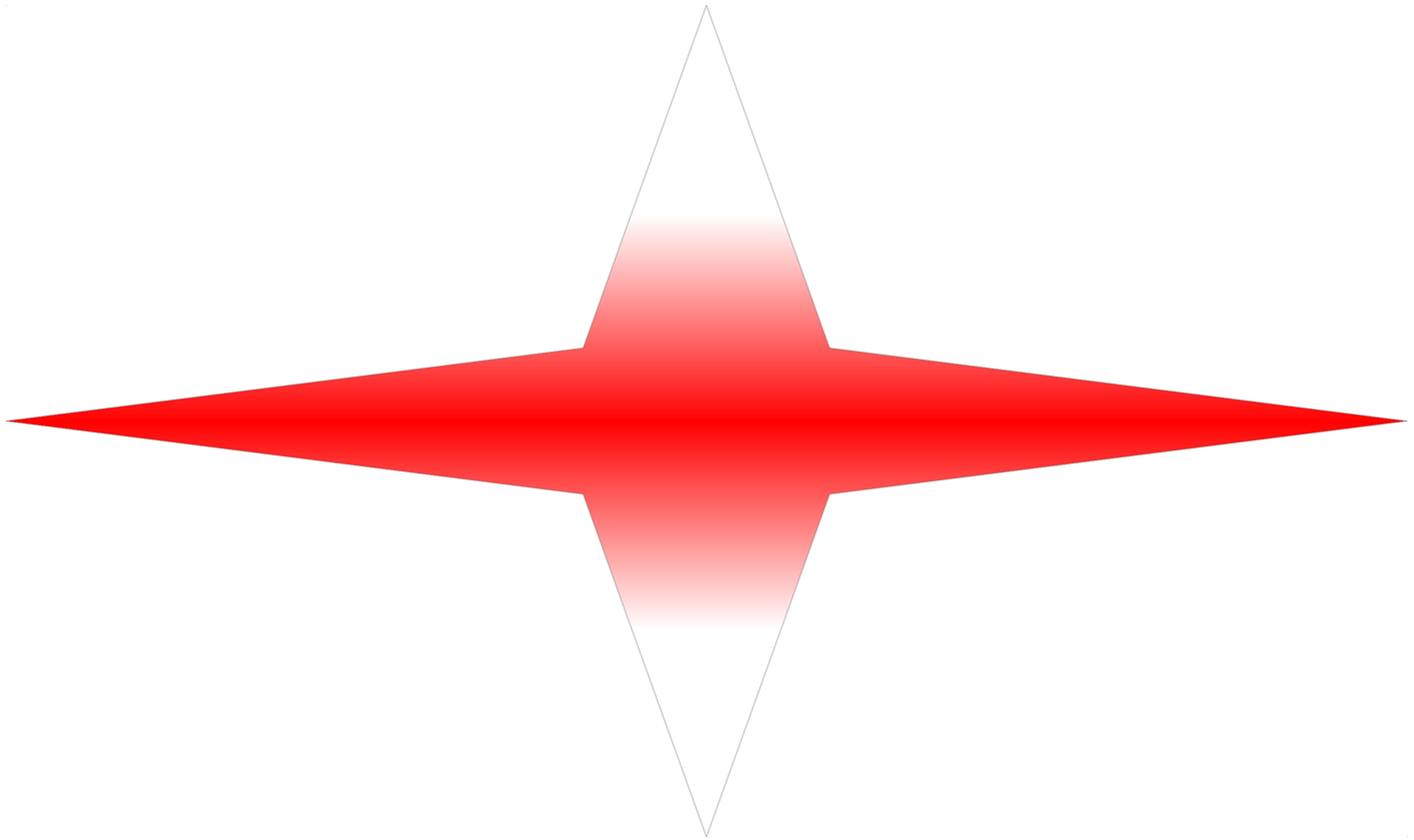
Namespaces	Functionality	What does it mean?
Mount	Isolate the set of FS mount points seen by processes	/tmp in container can be different in ns’ Remount ‘/’ read only within namespace
PID (process ID)	Process can have same PID in different NS (include PID1)	Process in NS can’t see/interact with process outside All processes are visable in ‘root’ PID NS
Network	Isolate the networking stack: ip addr, routes, netfilter iptable rules	Each NS has its own private loopback IF Commonly used with virtual ethernet IF pair
UTS	Set a different host and domain names for NS	No impact to the rest of the system Useful when combined with Network NS
IPC	Private inter-process communication environment: message queues, semaphores, shared memory	Resources are only accessible within the Namespace

Process Isolation - User Namespace

May be enabled in future releases...

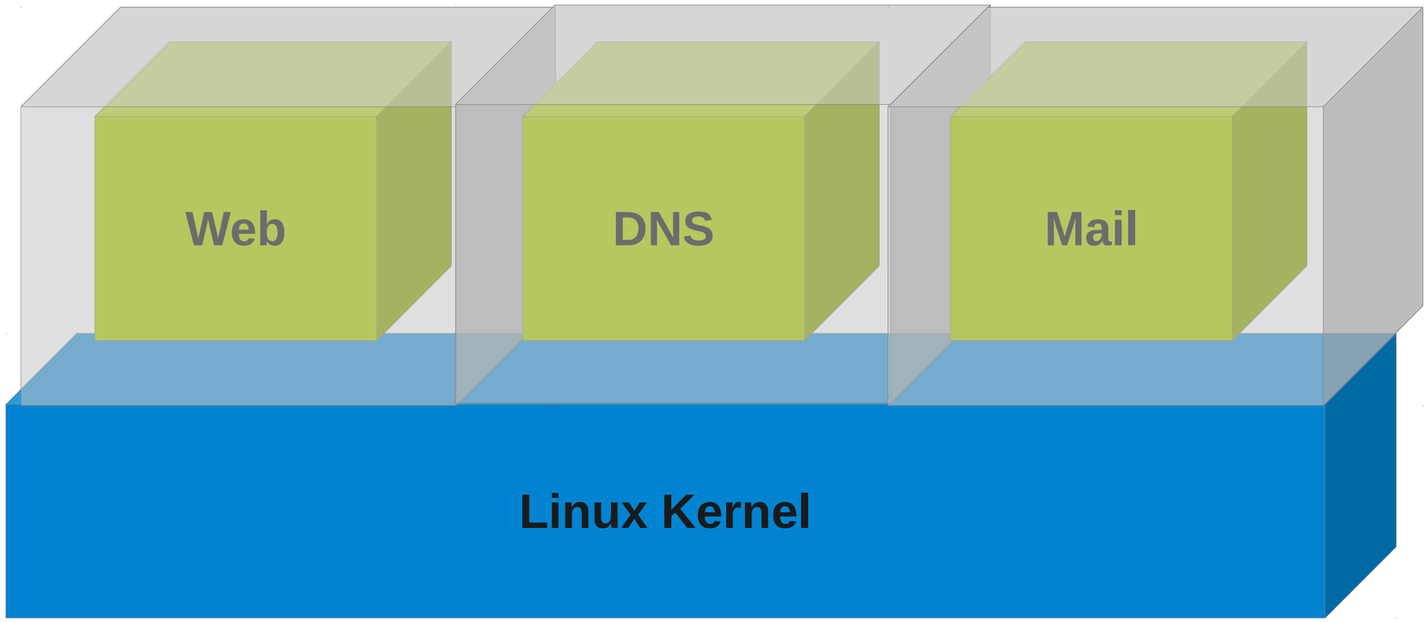
- Mapping between UIDs and GIDs
  - 5000-6100 on Root Namespace
  - 0-1100 in User Namespace
- Super user (UID 0) possible inside the Namespace
  - Configuring network on a network namespace
  - Binding to ports < 1024
  - Treated as unprivileged UID 5000 outside namespace

Linux Containers - Security

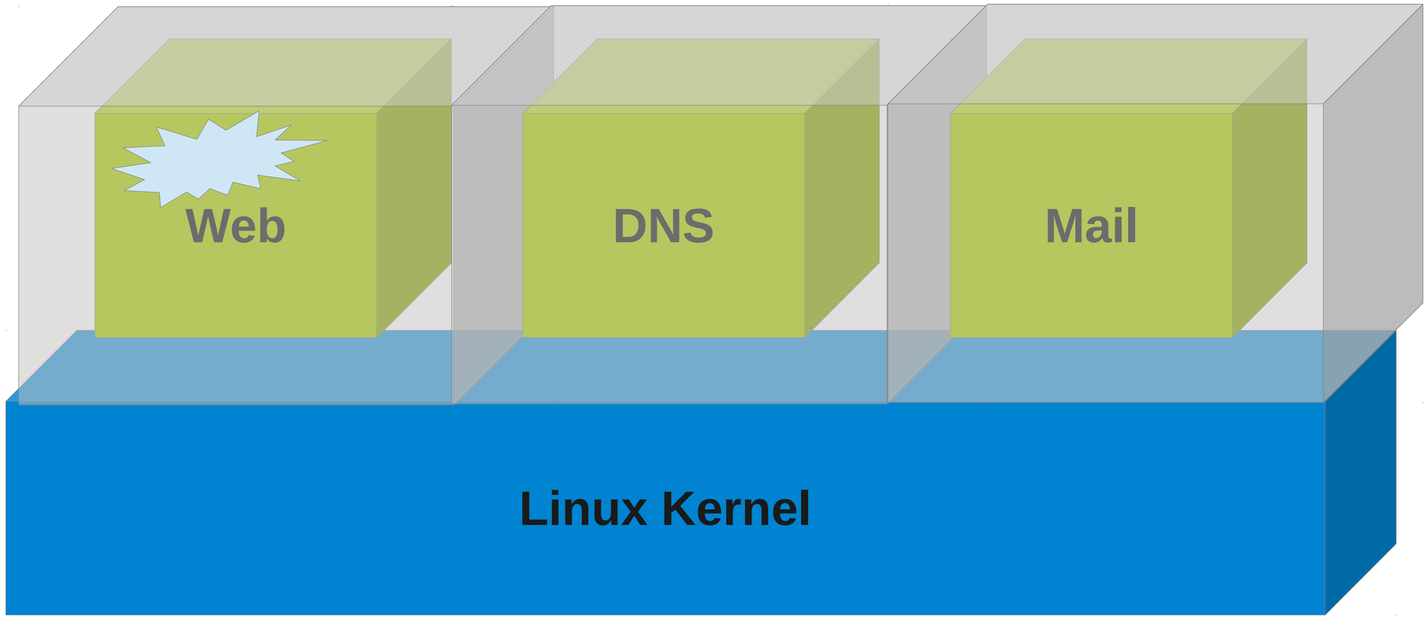


SELinux – Security

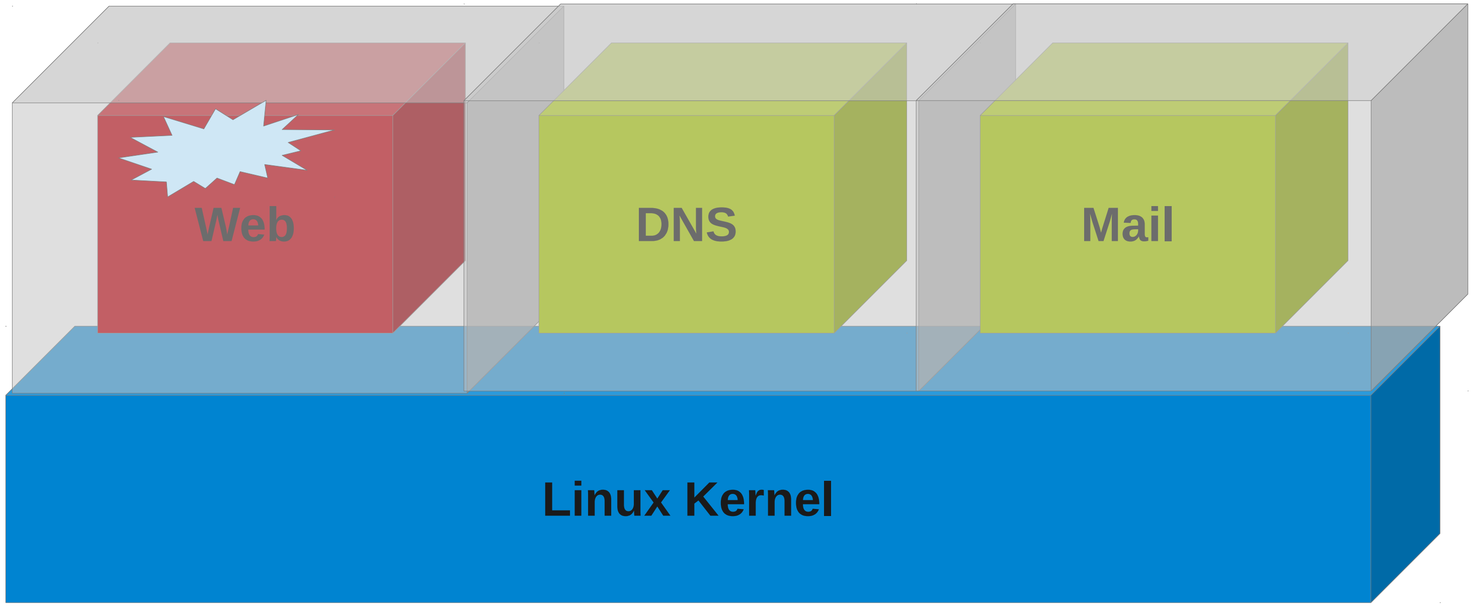
SELinux - Security



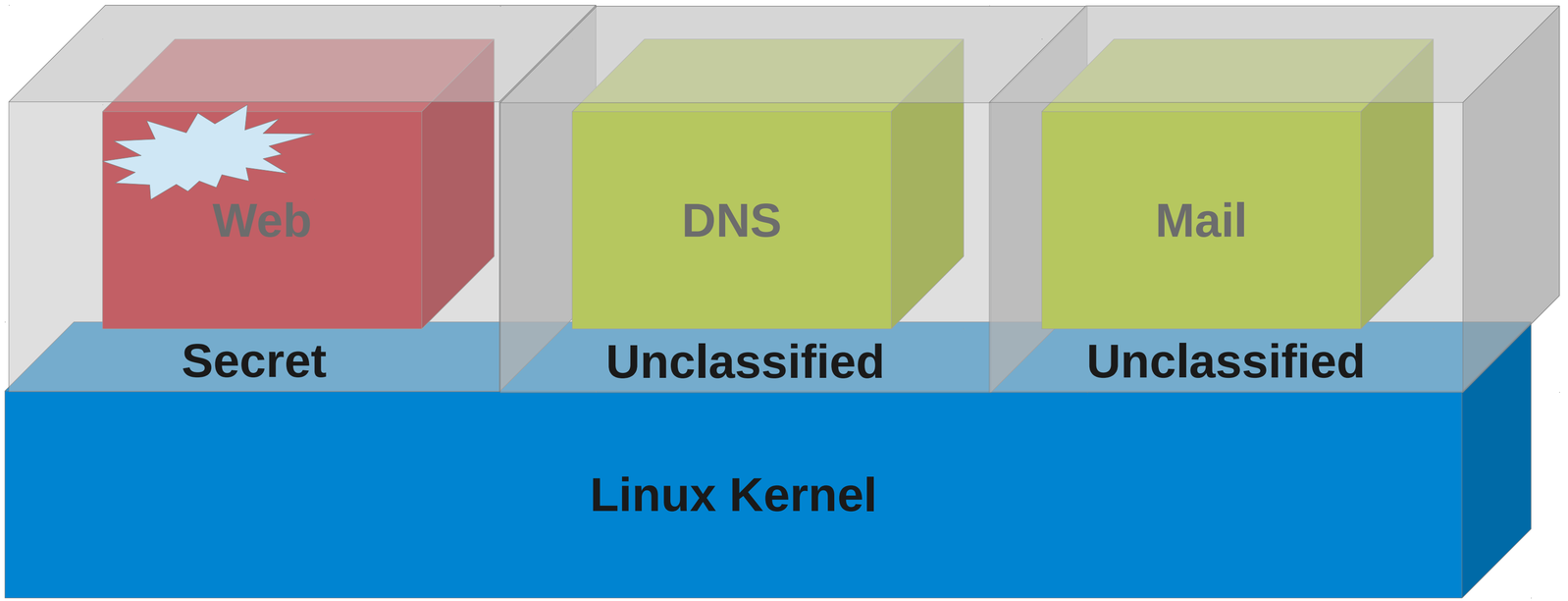
Each container process is confined in its own sandbox distinct from other the other processes



When a process is attacked.....

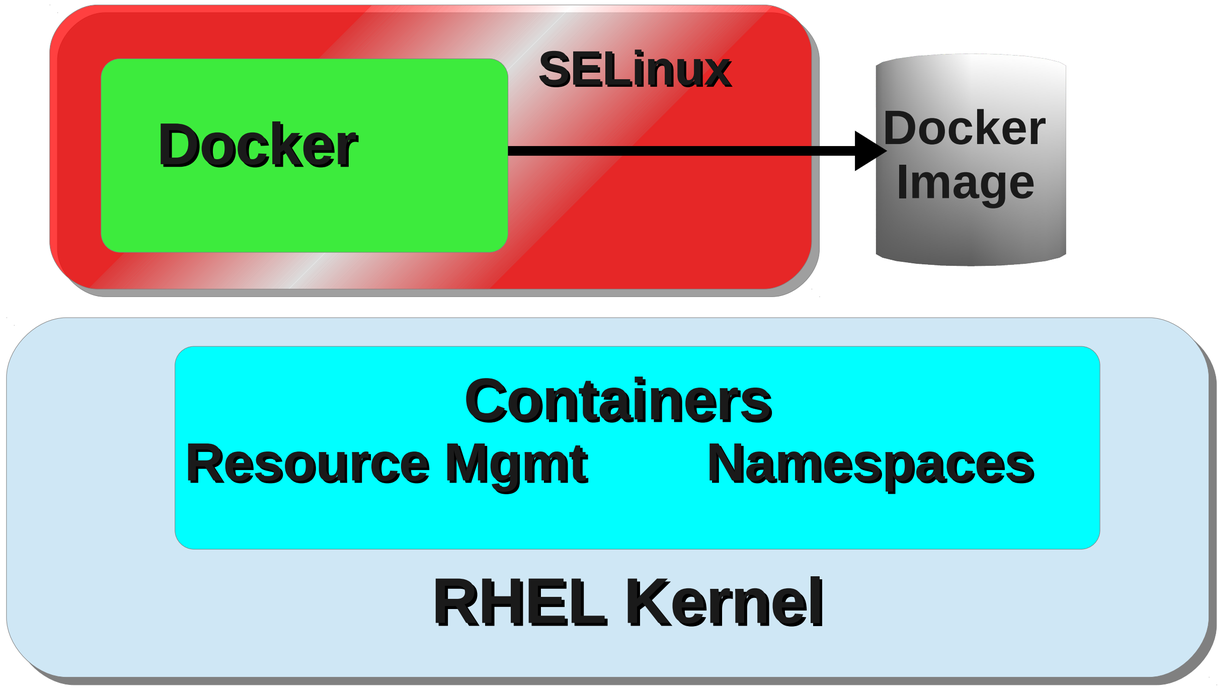


...and compromised, there is far less exposure. Only the container process is lost – lose the process not the system.

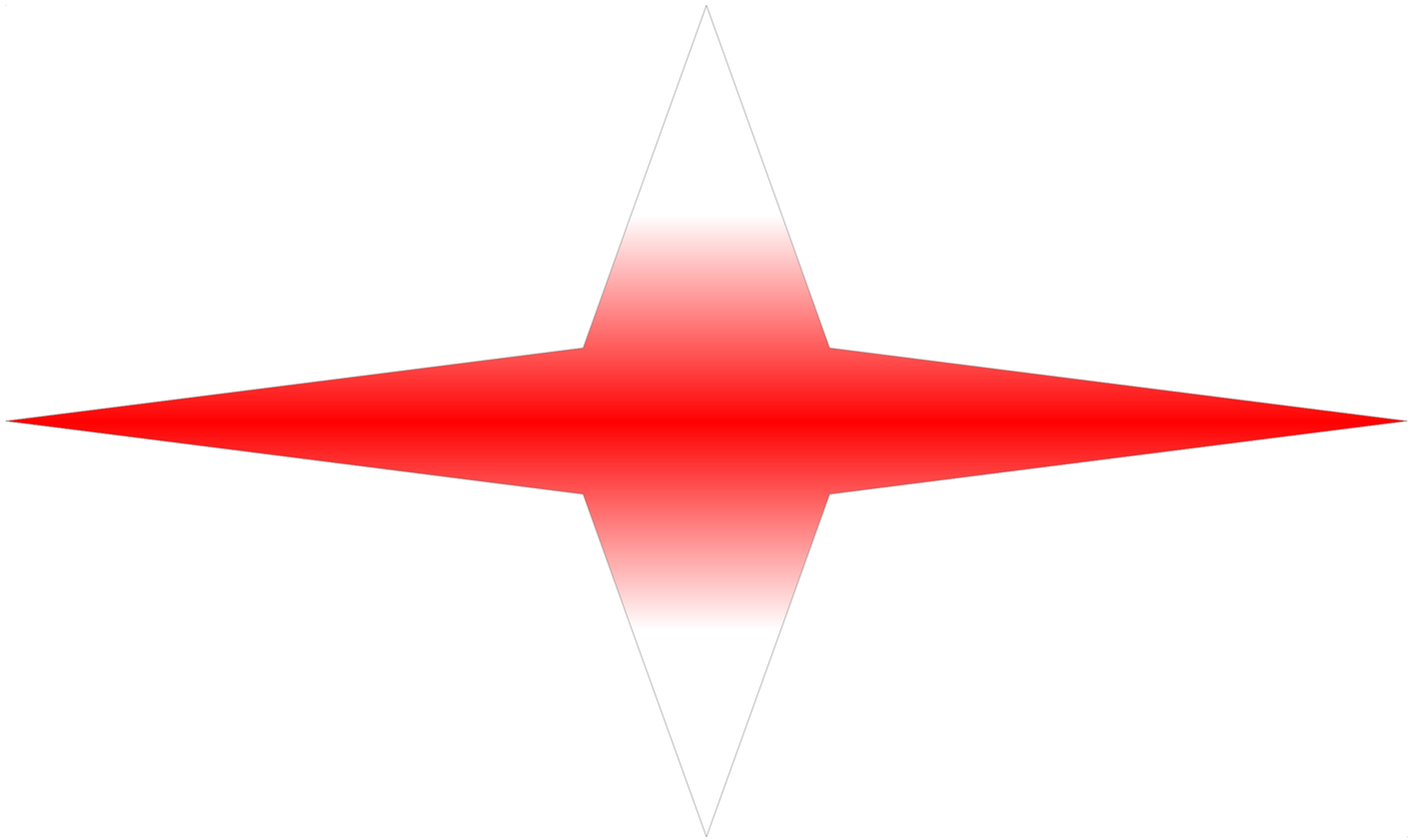


Label the container process with multiple level using SELinux Multi Level Security (MLS)

SELinux and Docker



Linux Containers - Management



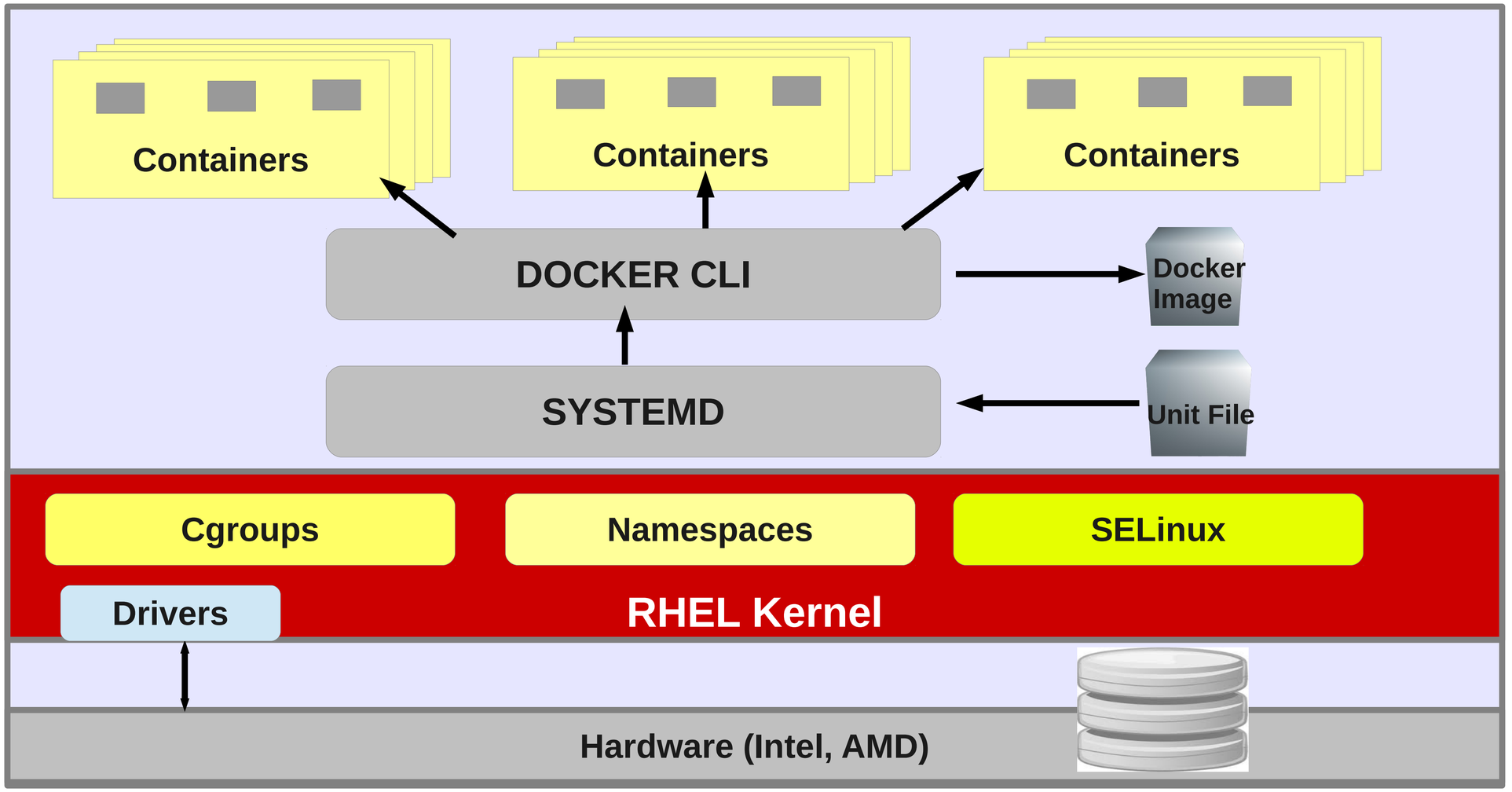
System and Container Management

Docker CLI

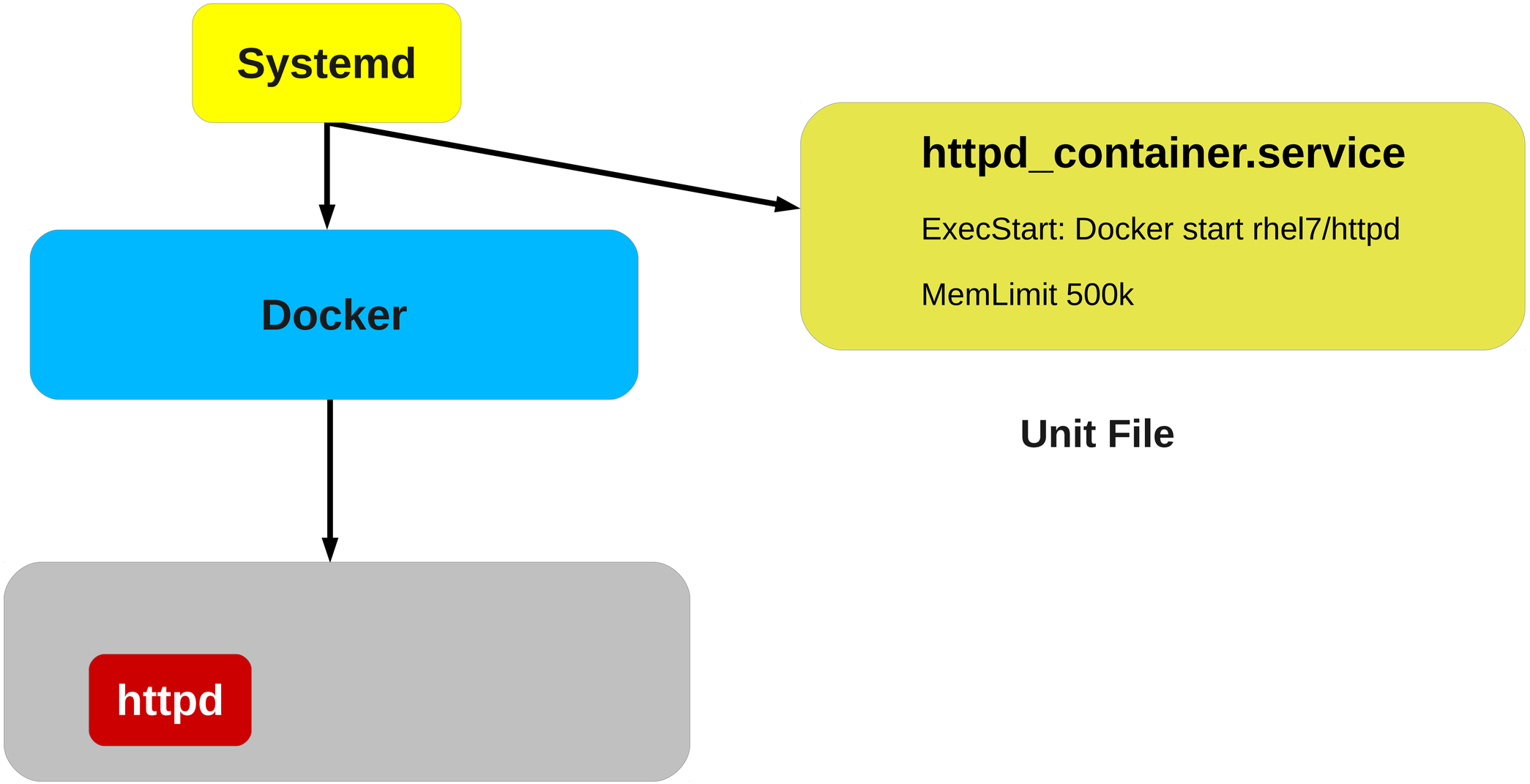
Docker format

- Tool to package an application and its runtime dependencies for deployment into a Linux Container
- Docker 0.9 includes libContainer, native LXC implementation

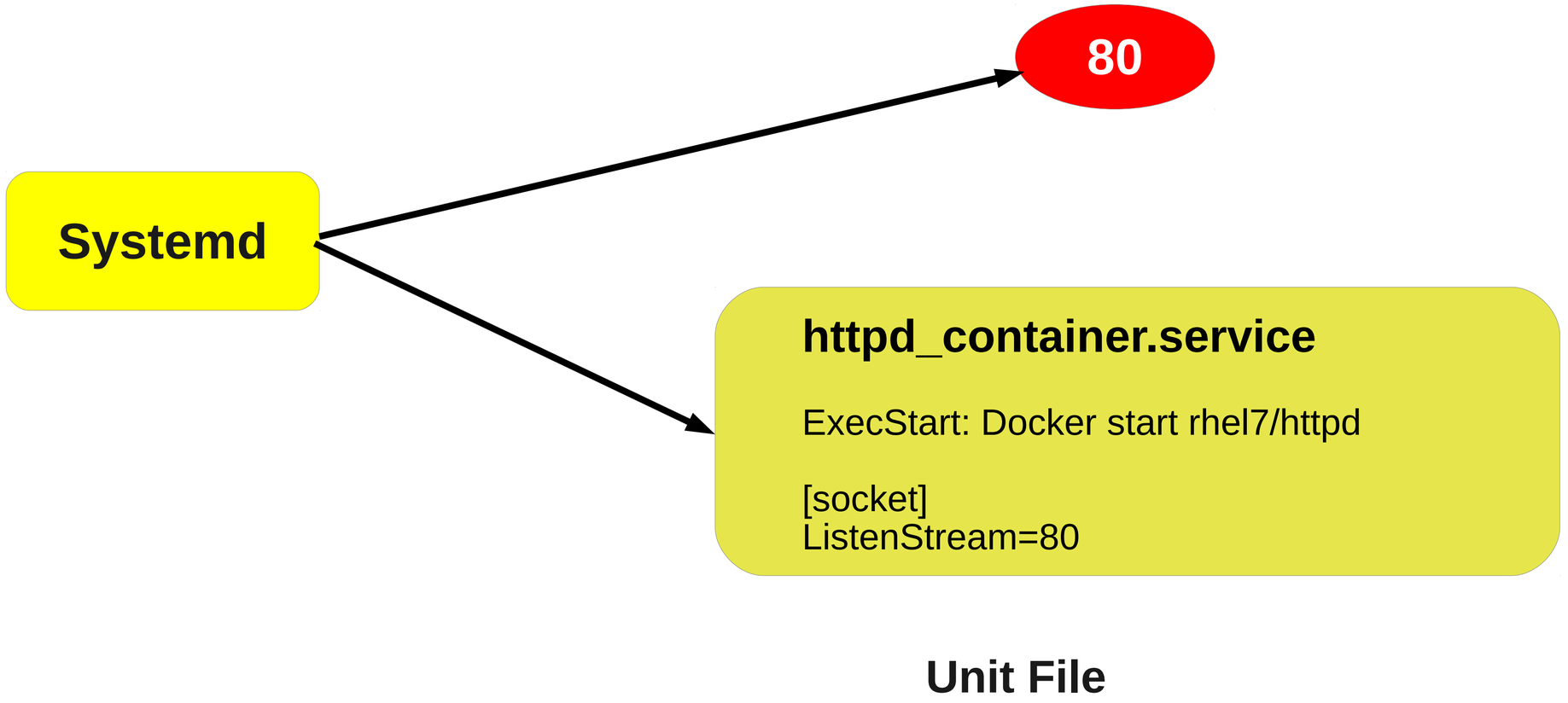
Red Hat Enterprise Linux 7 Containers Architecture with Docker CLI



Systemd Cgroup Configuration passed to Docker

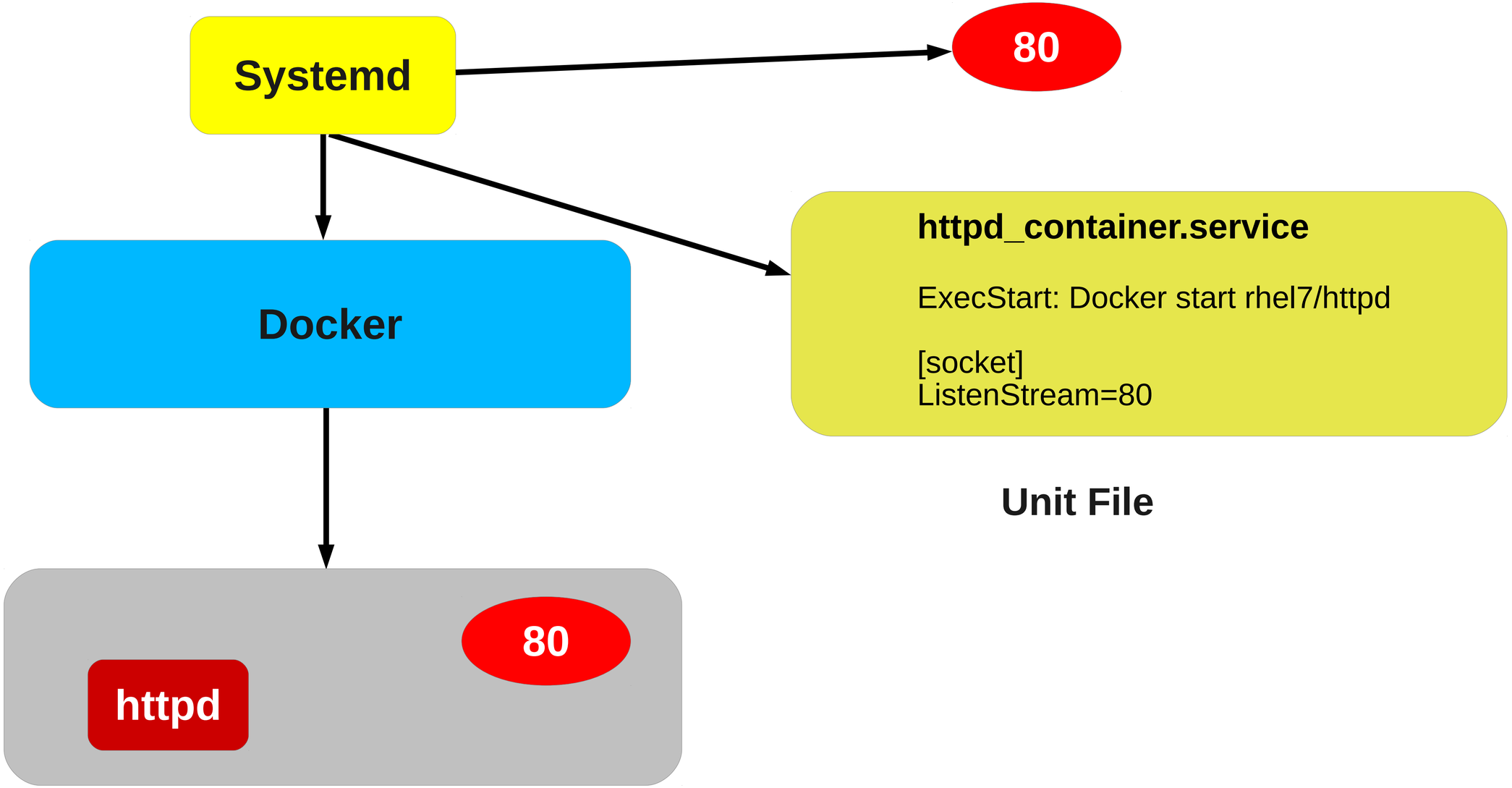


Systemd Socket Activation of Docker Containers

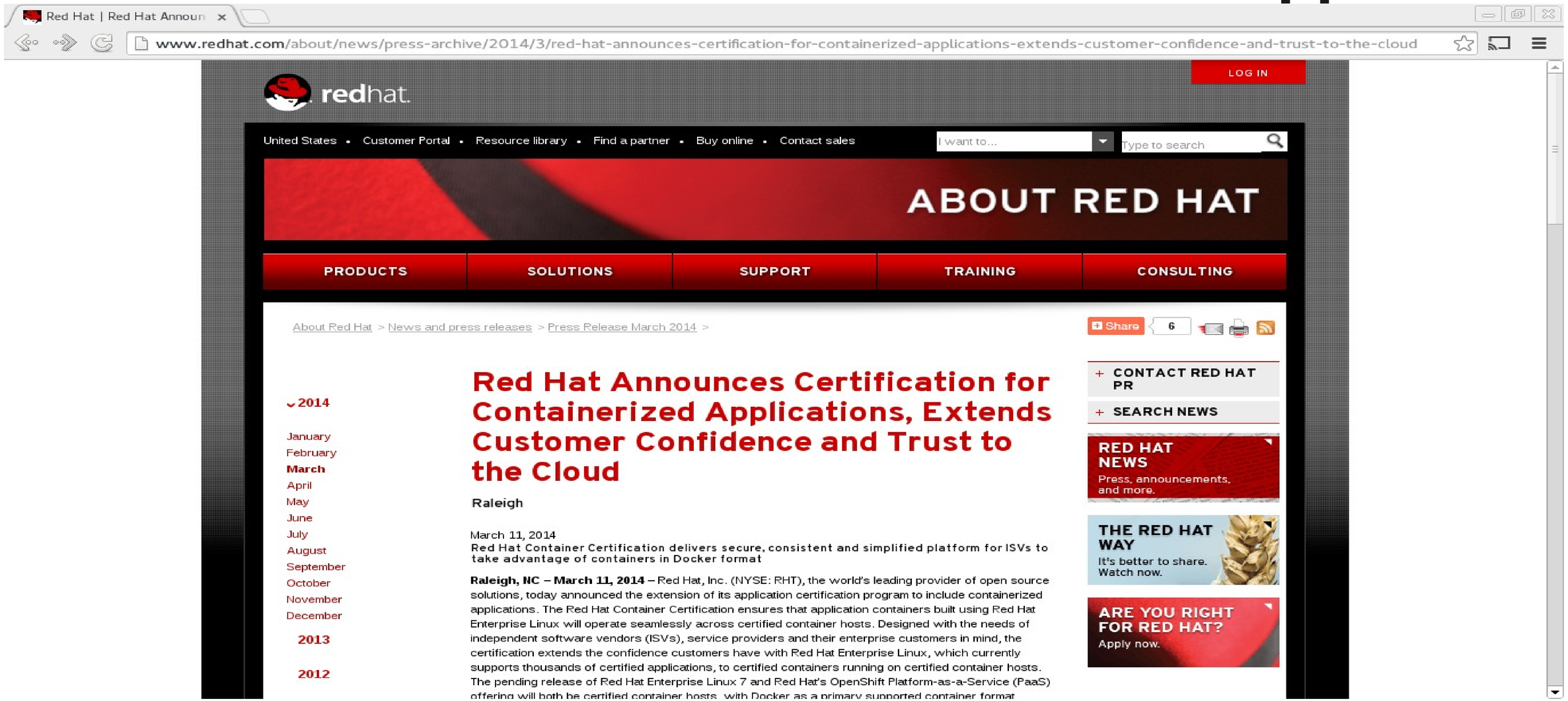


Unit File

Systemd Socket Activation of Docker Containers



Red Hat Certification for Containerized Apps



http://www.redhat.com/about/news/press-archive/2014/3/red-hat-announces-certification-for-containerized-applications-extends-customer-confidence-and-trust-to-the-cloud

Demo

Linux Containers in RHEL 7 - Key Takeways

- Application isolation mechanism for Light-weight multi-tenancy
- Application centric packaging w/ Docker image-based containers
- Linux Containers Productization
  - Key kernel enablers – full support in RHEL 7 GA
  - Docker 1.0 – shipped with RHEL 7 GA
- Linux Container Certification
- Red Hat and Docker partnership to build enterprise grade Docker containers

Linux Container Sessions, Demos and Labs

Time	Title	Venue
Wed 1:20 PM – 2:20 PM	Portable, lightweight and interoperable Docker containers across Red Hat Solutions	Room 236
Wed 3:50 PM – 5:50 PM	Containers & resource management in Red Hat Enterprise Linux 7 Beta (Hand-on Lab)	Room 105
Wed 11:00 AM – 1:00 PM	Linux Containers and Application Isolation Demo	Red Hat Booth (Infrastructure Pod 1)
Wed 1:20 PM – 3:20 PM	Implementing & managing OpenShift Enterprise	Labs II
Wed 11:30 AM – 2:00 PM	Next Generation Container Management Demo	Emerging Technologies Red Hat Booth (Pod 31)

Questions?