Secure Operating Systems

Nikos Tziritas

Cloud Computing (Part 2)

Cloud Computing Security Features (1/5)

- Advanced Perimeter Firewall
  - Most of the firewalls are simple because they just inspect the source and destination packets only.
  - There are some more advanced firewalls available that perform stable packet inspection
  - They check the file packets integrity to ensure the stability before approving or rejecting the packet

Cloud Computing Security Features (2/5)

- Intrusion Detection Systems with Event Logging
  - All IT Security compliance standards must involve the businesses to have a means, which can track and record all type of intrusion attempts
  - IDS event logging solutions are necessary for all businesses that want to meet the compliance standards
  - There are some cloud providers who offer IDS monitoring service and update the security rules for their firewalls in order to counter the great signals and malicious IP addresses, which are detected for all cloud users

Cloud Computing Security Features (3/5)

- Internal Firewalls for Each Application & Databases
  - Using a strong top-in-line perimeter firewall will block the external attacks only but internal attacks are still a major danger.
  - If there are no internal firewalls in infrastructures to restrict the sensitive data access and applications then there will be a security issue
  - What if an employee user account can allow the hackers to bypass the perimeter firewall completely?

Cloud Computing Security Features (4/5)

- Data-at-rest encryption
  - Data encryption is one of the effective methods to keep the most sensitive data stored in the cloud infrastructure safe and secure from the unauthorized user.
  - Strong type of encryption will minimize the chance of stolen data used for some purpose.

Cloud Computing Security Features (5/5)

- Tier IV Data Centers with Strong Physical Security
  - Last possible way for the hackers and the industrial spies is the physical hardware
  - If hackers get direct access to hardware they have free reign to steal the data or upload the malware directly to the local machine
  - Thus a user must use tier IV data centers that will protect the cloud environment and restrict the access to the physical systems
    - 24x7 CCTV monitoring
    - Controlled access checkpoints via biometric security controls
    - Armed security patrols

Privileged User Access

- Sensitive data processed outside the enterprise brings with it an inherent level of risk
- Get as much information as you can about the people who manage your data
- Ask providers to supply specific information on the hiring and oversight of privileged administrators and the controls over their access.

Data Location

- When you use the cloud, you probably won’t know exactly where your data is hosted
- In fact you might not know what country it will be stored in.
- Ask providers if they will commit to storing and processing data in specific jurisdictions, and whether they will make a contractual to obey local privacy requirements on behalf of their customers.

Data Segregation

- Data in the cloud is typically in a shared environment alongside data from other customers
- Encryption is effective but it is not a cure-all
- Find out what is done to segregate data at rest
- The cloud provider should provide evidence that encryption schemes were designed and tested by experienced specialists
- Encryption accidents can make data totally unusable

Recovery

- Even if you don’t know where your data is, a cloud provider should tell you what will happen to your data and service in case of a disaster
- Any offering that does not replicate the data and application infrastructure across multiple sites is vulnerable to a total failure
- Ask the provider if it has the ability to do a complete restoration, and how long it will take.

Investigative Support

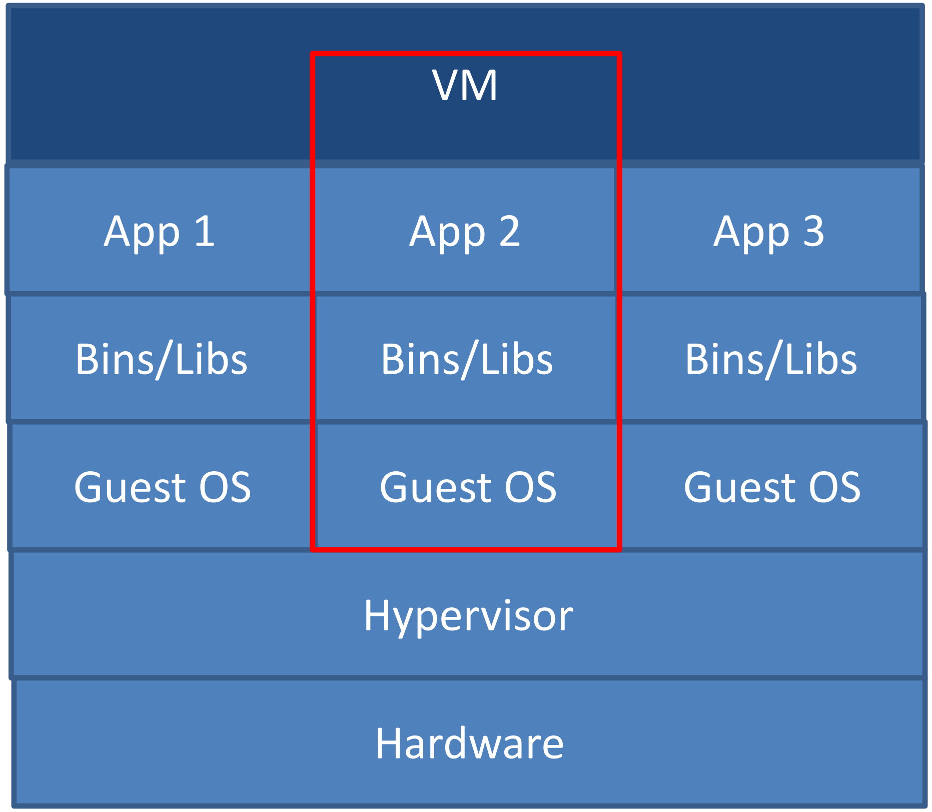
- Investigating inappropriate or illegal activity may be impossible in cloud computing.
- Cloud services are especially difficult to investigate, because logging and data for multiple customers may be co-located.
- If you cannot get a contractual commitment to support specific forms of investigation, then your only safe assumption is that investigation and discovery requests will be impossible.

Long-term Viability

- Ideally, your cloud computing provider will never get acquired and swallowed up by a larger company
- But you must be sure that in such a case that data will remain available even after such an event

Popular Virtual Machines Providers

VM vs Container (1/2)



VM

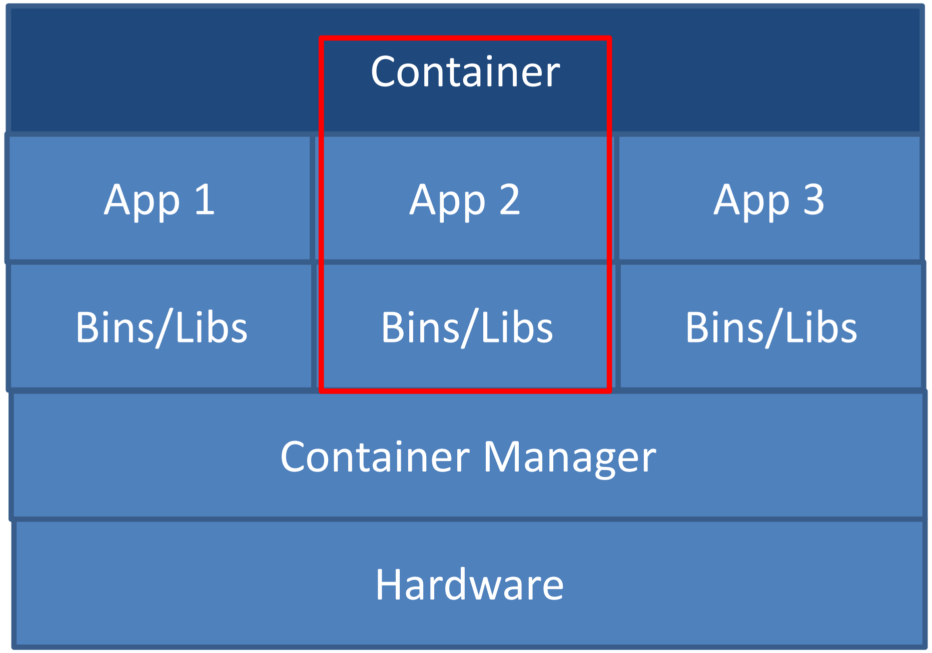
App 1 | App 2 | App 3

Bins/Libs | Bins/Libs | Bins/Libs

Guest OS | Guest OS | Guest OS

Hypervisor

Hardware



Container

App 1 | App 2 | App 3

Bins/Libs | Bins/Libs | Bins/Libs

Container Manager

Hardware

VM vs Container (2/2)

- Container is a self-contained package running a single application.
- Container is a light-weight architecture that can boot up much faster than a VM
- A container can migrate from a server to another one by consuming much less data against a VM
- A container needs less resources against a VM

Security Issues

- In a container a user with root access can have access to all of the containers hosted by the underlying OS.
  - Because a container can run anywhere, this poses some security vulnerabilities.
  - We need to be aware of how the container can impact the actual host
- VM approaches provide isolation from other VMs due to the hypervisor.

Gather Images Security Issues

- We must get the container image. The base image is important because it serves as a starting point to result in more enhanced images.
  - Trusted sources
- Adding an application is also a crucial task since if the application is not trusted then there can be security breaches.

Manage Access Security Issues

- We must use private registry to control access through role-based assignments
  - A registry is considered private if pulling an image requires authentication
- By having access through a private registry we can automate and assign policies (e.g., checking signatures, code scans) minimizing in that way human errors that can result in security breaches.

Integrate Security Testing to avoid Security Breaches

- Because patching containers is less safe than rebuilding them, we can integrate security testing such that to trigger automated container rebuilding when needed in case of updates.

Defend your Infrastructure

- We must take care when choosing a hosting operating system such that it provides maximum container isolation.
- Which containers need to access other containers?
- How will we control access and manage shared resources?

Kubernetes Container Orchestration

- Kubernetes cluster consists of a set of nodes (worker nodes)
  - Physical or virtual nodes
- These nodes host applications in the form of containers
- Somebody needs to load containers on the worker nodes.
  - Identify the right worker nodes
  - Monitor and track the containers running on nodes
- The container loading on worker nodes takes place by the master node
  - Manage
  - Plan
  - Schedule
  - Monitor
- To achieve the above master node uses some components called control plane components.

Key-Value Store (known as ETCD Cluster)

- ETCD is a distributed reliable key-value store that is simple, fast and secure.
- ETCD makes reference to distributing the “Unix/etc” directory.
  - Most global configuration files live across multiple machines
- Database Storing information in a key-value format.

Key-Value

No Key-Value

name	Age	location	Salary	Grade
Kostas	35	Athens	1000	
Manos	21	Thessaloniki		A
Maria	38	Karditsa	800	
Giorgia	20	Volos		B

Key-Value

Key	Value
Name	Kostas
Age	35
Salary	1000
Location	Athens

Key	Value
Name	Giorgia
Age	20
Grade	B
Location	Volos

Distributed Key-Value

- We keep a copy of the store inside each node
  - In that way it is more reliable since if some node fails then the rest nodes keep also an instance of the store
- The replicated stores must be consistent when updating information
  - There must be a leader to keep consistency of stores across nodes

Kube Scheduler

- A scheduler identifies the right node to load the container
- The above takes place based on the container demands and node available resources

Controllers

- Node controller is responsible for integrating new nodes on the cluster
  - Monitor when nodes are unavailable
- Replication controller ensures that the desired number of containers (replicas) are running across the cluster nodes

Kube-apiserver

- Kube-apiserver is responsible for orchestrating all operations within the cluster
  - ETCD Cluster
  - Kube-scheduler
  - Controllers
- Kube-apiserver is used by external users to perform management operations on the cluster

Container Runtime

- We need a container runtime to run containers across cluster nodes.
- Such a runtime for example is docker.
- Kubernetes supports also other runtimes except docker.

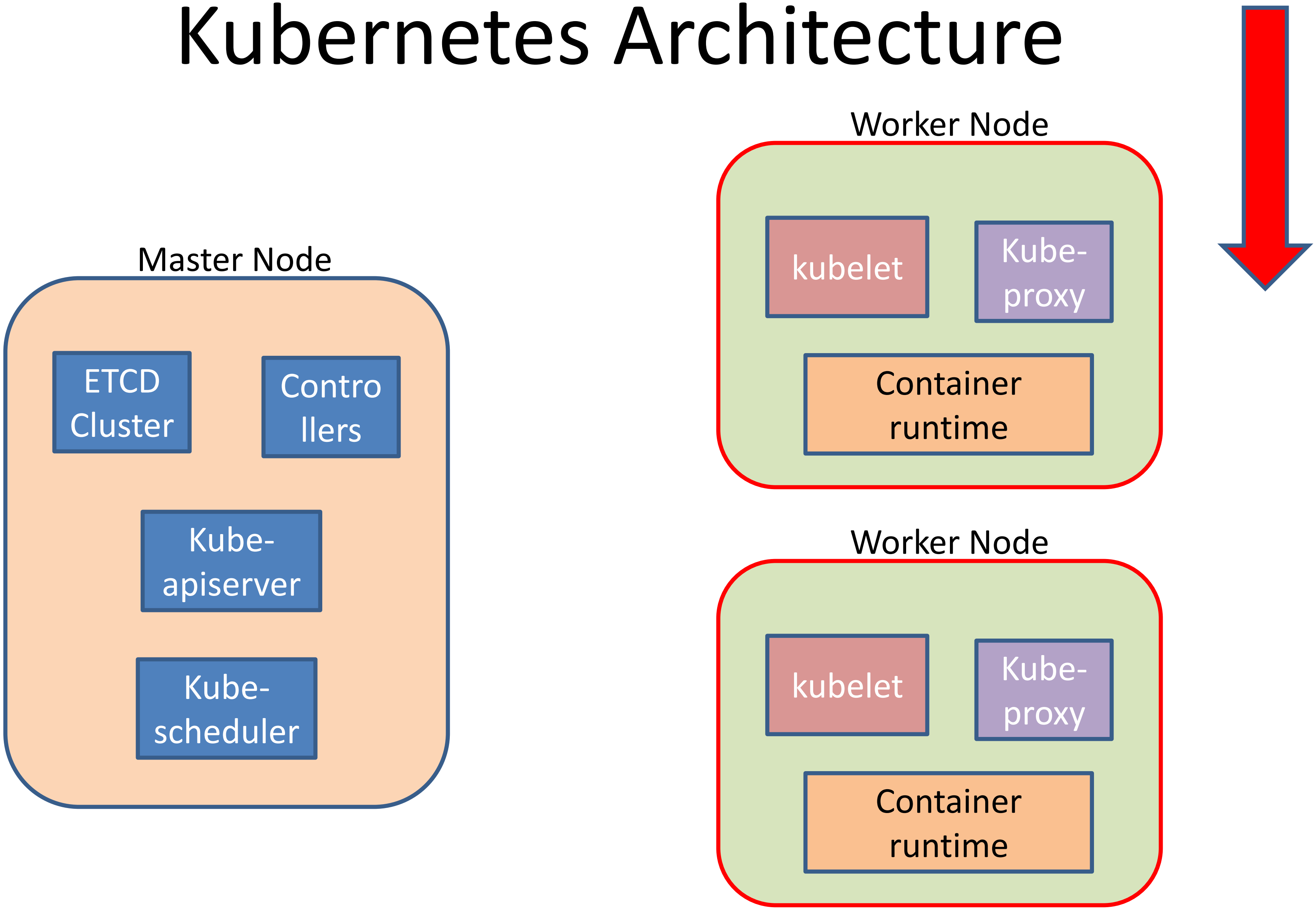
Kubelet

- Kubelet is an agent running on each worker node within the cluster.
- Kubelet is responsible for reporting back to the master node about the information of the containers loaded on the specific node as well as the state of the node.
- Kubelet is responsible for loading the appropriate containers on a node
- Kubelet listens for instruction from the kube-apiserver

Kube Proxy

- Kube proxy runs on each worker node within the cluster
- Its responsibility is the communication between worker nodes.
  - For example we can have in one worker node a web server and on another the database server. So these services needs to communicate each other.

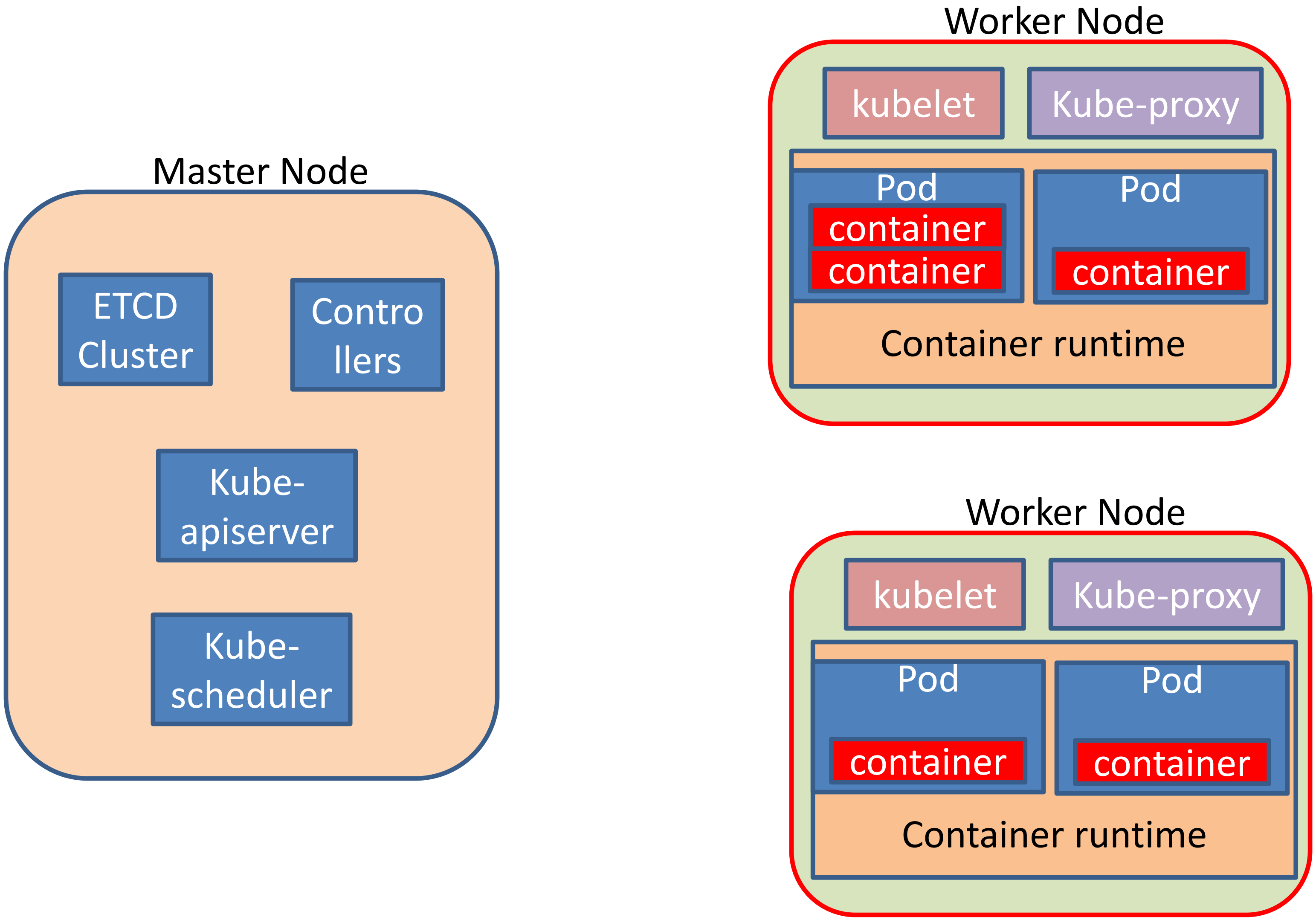
Kubernetes Architecture



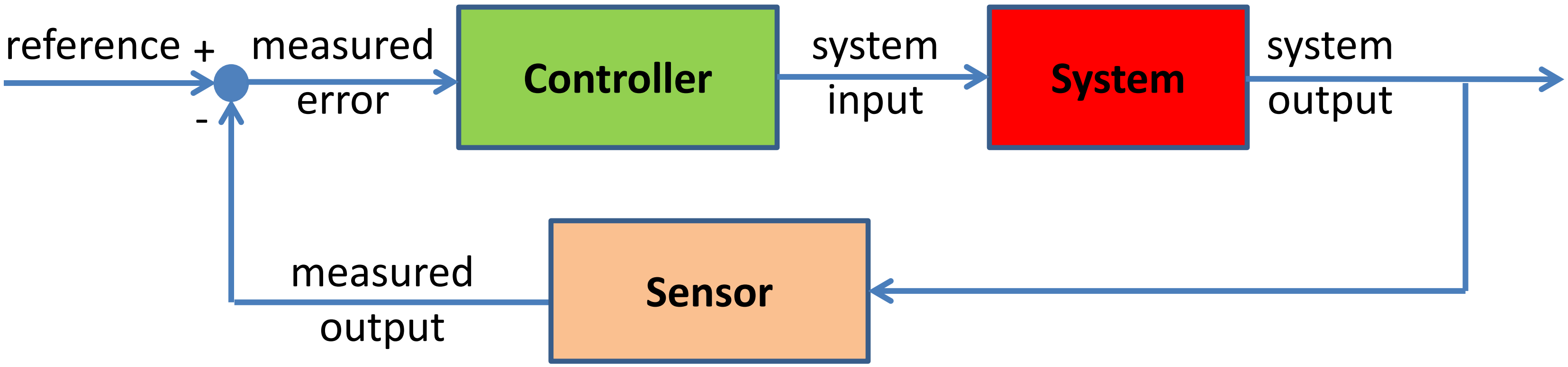
Pods

- Pod is the lowest level resource within a Kubernetes cluster
- A pod usually consists of a single container. Some times a pod may contain a couple of containers where they form a service
- Limits are set for Pods when designing a cluster. These limits are about memory and CPU.

Kubernetes Architecture with Pods



Kubernetes as a Control Loop



Container Orchestration Summary

- Allocation of resources between containers
- Scaling up/down containers to load balance application load across nodes
- Container deployment on nodes
- Container migration across nodes
- Monitoring of containers and nodes
- Connecting containers with external world
- Service discovery among containers