

# Apache Spark on K8s + HDFS Security



Ilan Filonenko (ifilonenko@bloomberg.net)

**Bloomberg**

# Agenda

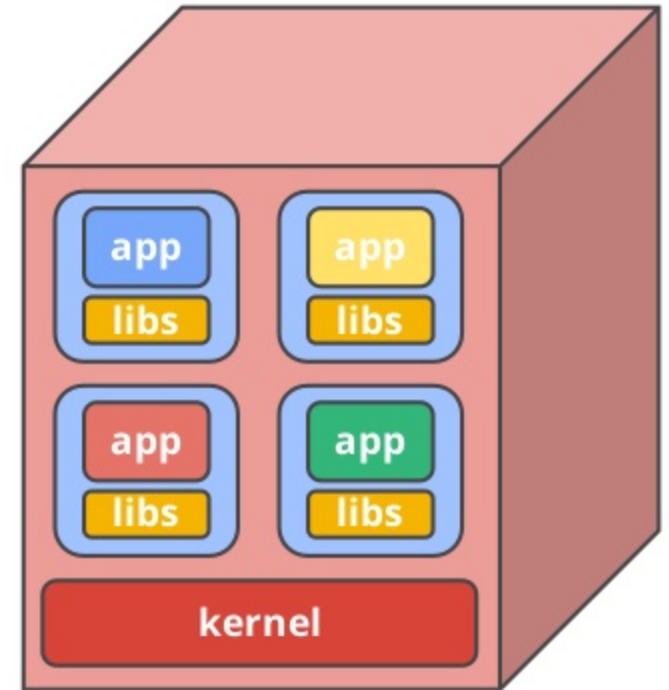
1. **Kubernetes intro**
2. **Big Data on Kubernetes**
3. Demo: Spark on K8s accessing secure HDFS
4. Secure HDFS deep dive
5. HDFS running on K8s
6. Data locality deep dive

# Kubernetes

“New” open-source cluster manager.

- [github.com/kubernetes/kubernetes](https://github.com/kubernetes/kubernetes)

Runs programs in Linux containers.



1600+ contributors and 60,000+ commits.



*"My app was running fine  
until someone installed  
their software"*

*- Jane Doe, Sr. Dev*

**DON'T  
TOUCH  
MY  
STUFF**

## More isolation is good

Kubernetes provides each program with:

- a lightweight virtual file system -- Docker image
  - an independent set of S/W packages
- a virtual network interface
  - a unique virtual IP address
  - an entire range of ports

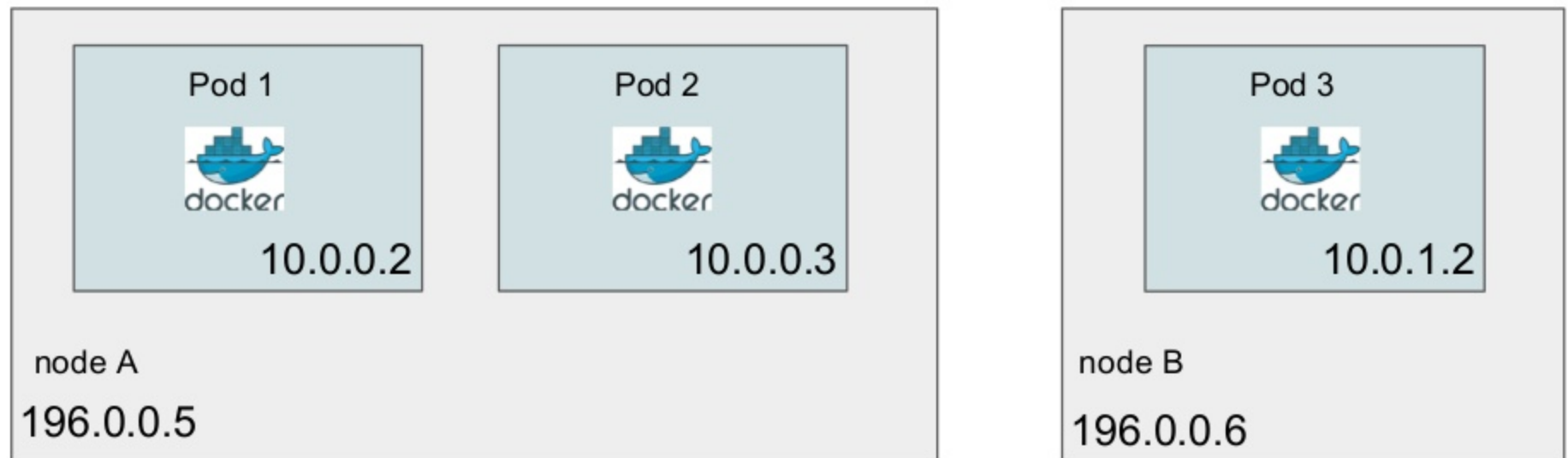
## Other isolation layers

- Separate process ID space
- Max memory limit
- CPU share throttling
- Mountable volumes
  - Config files -- ConfigMaps
  - Credentials -- Secrets
  - Local storages -- EmptyDir, HostPath
  - Network storages -- PersistentVolumes

# Kubernetes architecture

**Pod**, a unit of scheduling and isolation.

- runs a user program in a primary container
- holds isolation layers like a virtual IP in an infra container



## Big Data on Kubernetes

Since [Spark 2.3](#), the community has added features:

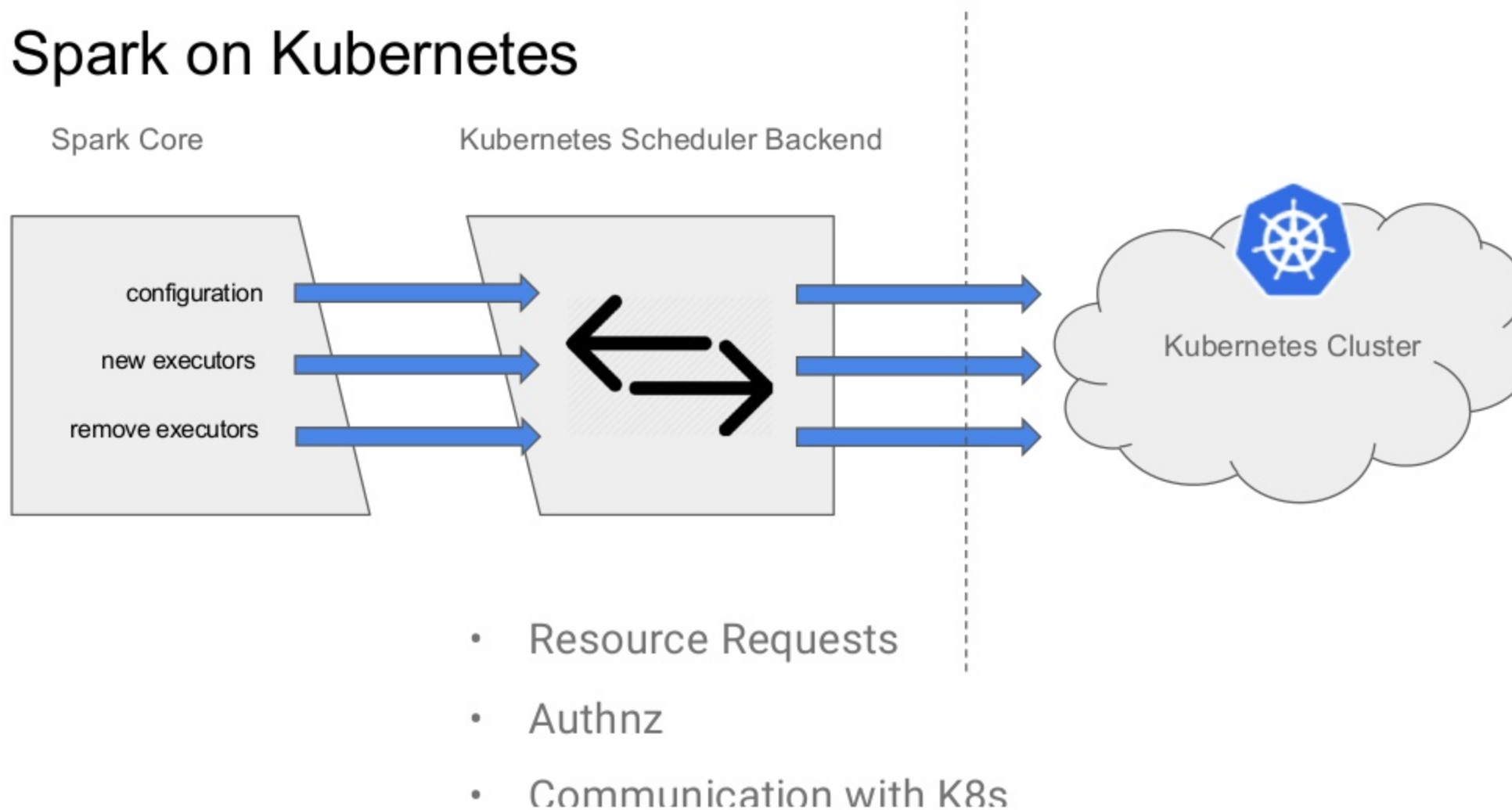
- non-JVM binding support and memory customization
- client-mode support for running interactive apps
- large framework refactors: rm init-container; scheduler

Talk: <https://conferences.oreilly.com/strata/strata-ca/public/schedule/detail/63855>

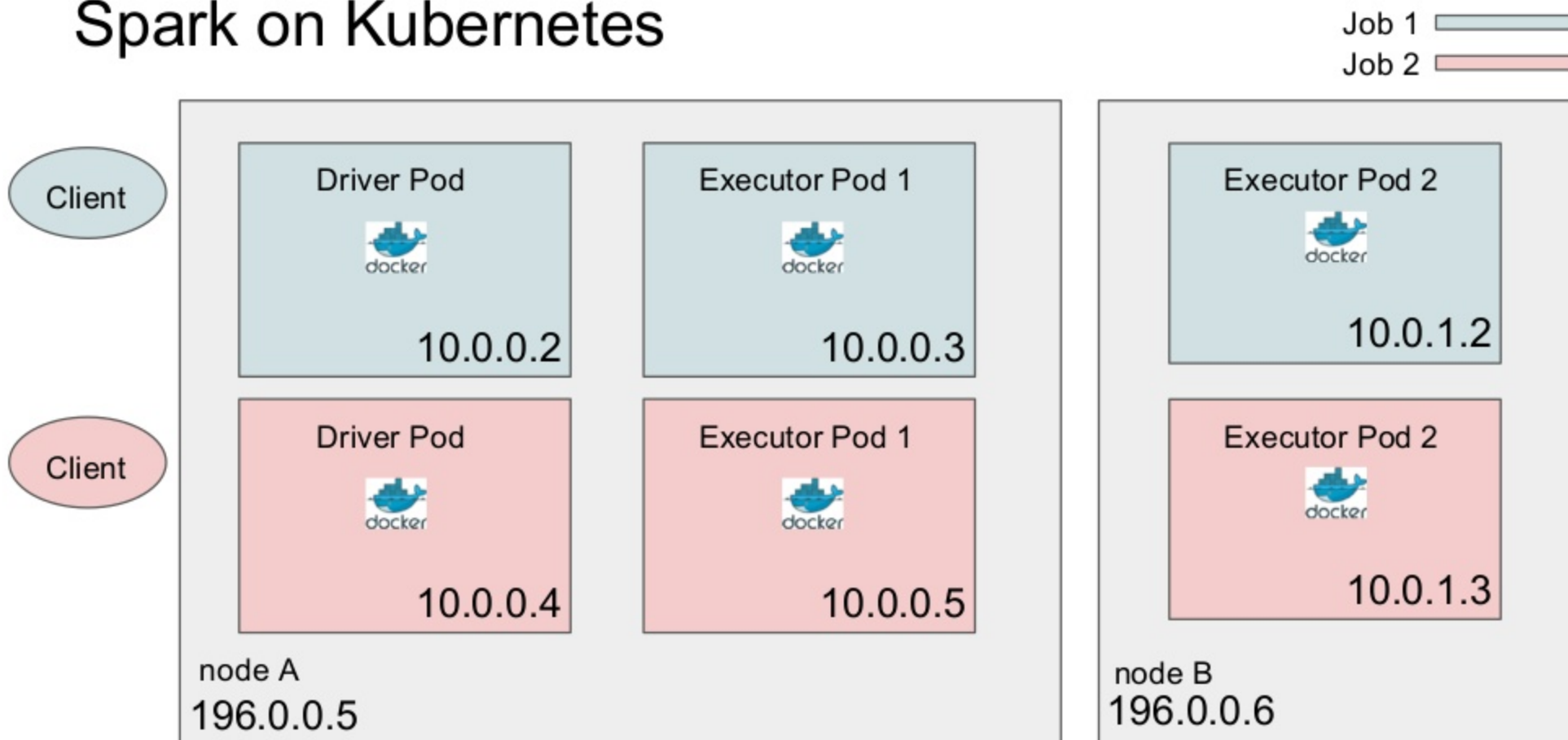
Kerberos work: <https://github.com/apache/spark/pull/21669>



# Spark on Kubernetes



# Spark on Kubernetes

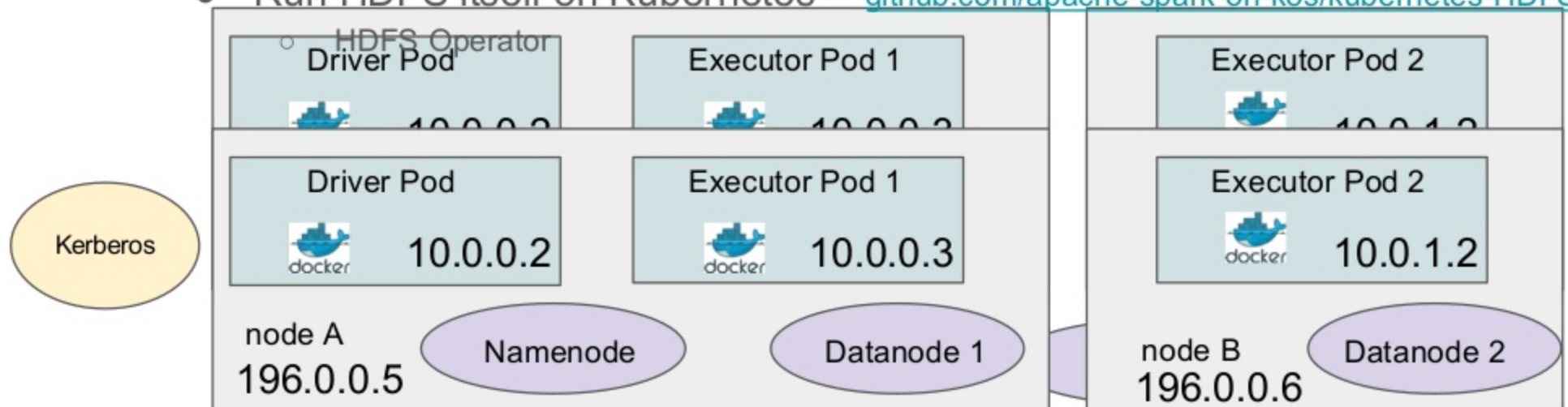


# What about storage?

Spark on Kubernetes supports cloud storages like S3.

Your data is often stored on HDFS:

- Access remote HDFS running outside Kubernetes
- Run HDFS itself on Kubernetes -- [github.com/apache-spark-on-k8s/kubernetes-HDFS](https://github.com/apache/spark-on-k8s/kubernetes-HDFS)



# Agenda

1. Kubernetes intro
2. Big Data on Kubernetes
- 3. Demo: Spark on K8s accessing secure HDFS**
- 4. Secure HDFS deep dive**
5. HDFS running on K8s
6. Data locality deep dive

# Demo: Spark k8s Accessing Secure HDFS

## Running a Spark Job on Kubernetes accessing Secure HDFS

Single-noded pseudo-distributed Kerberized Hadoop Cluster

<https://github.com/ifilonenko/hadoop-kerberos-helm>

Spark Submit with Kerberos Configs

<https://github.com/ifilonenko/secure-hdfs-test>

Keytab and \$kinit

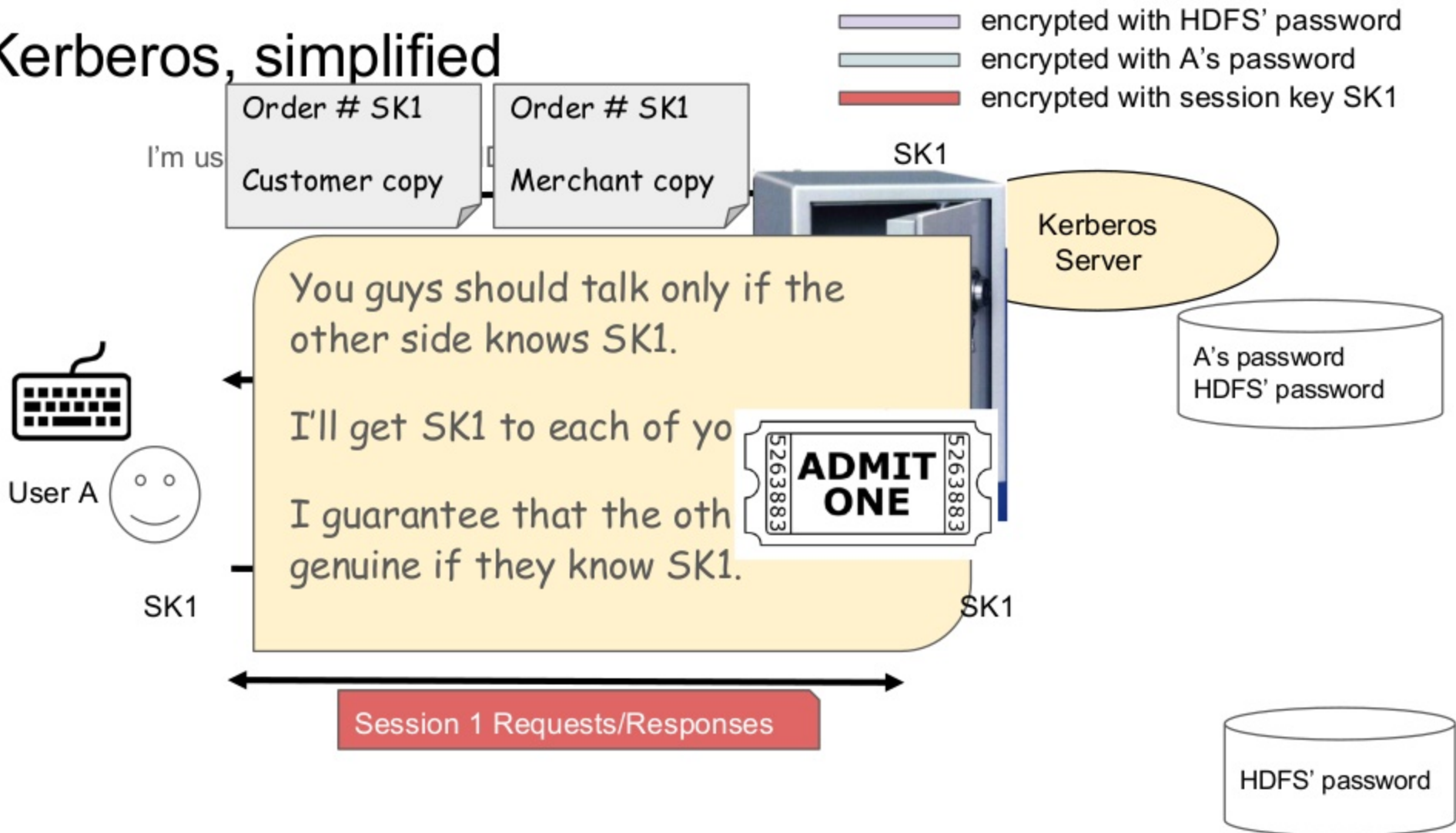
<https://asciinema.org/a/2vIJdw1N53Lo7LoSR09OMKdRH>

# Security deep dive

- Kerberos tickets
- HDFS tokens
- Long running jobs
- Access Control of Secrets



# Kerberos, simplified





# HDFS Delegation Token

Kerberos ticket, no good for executors on cluster nodes.

- Stamped with the client IP.

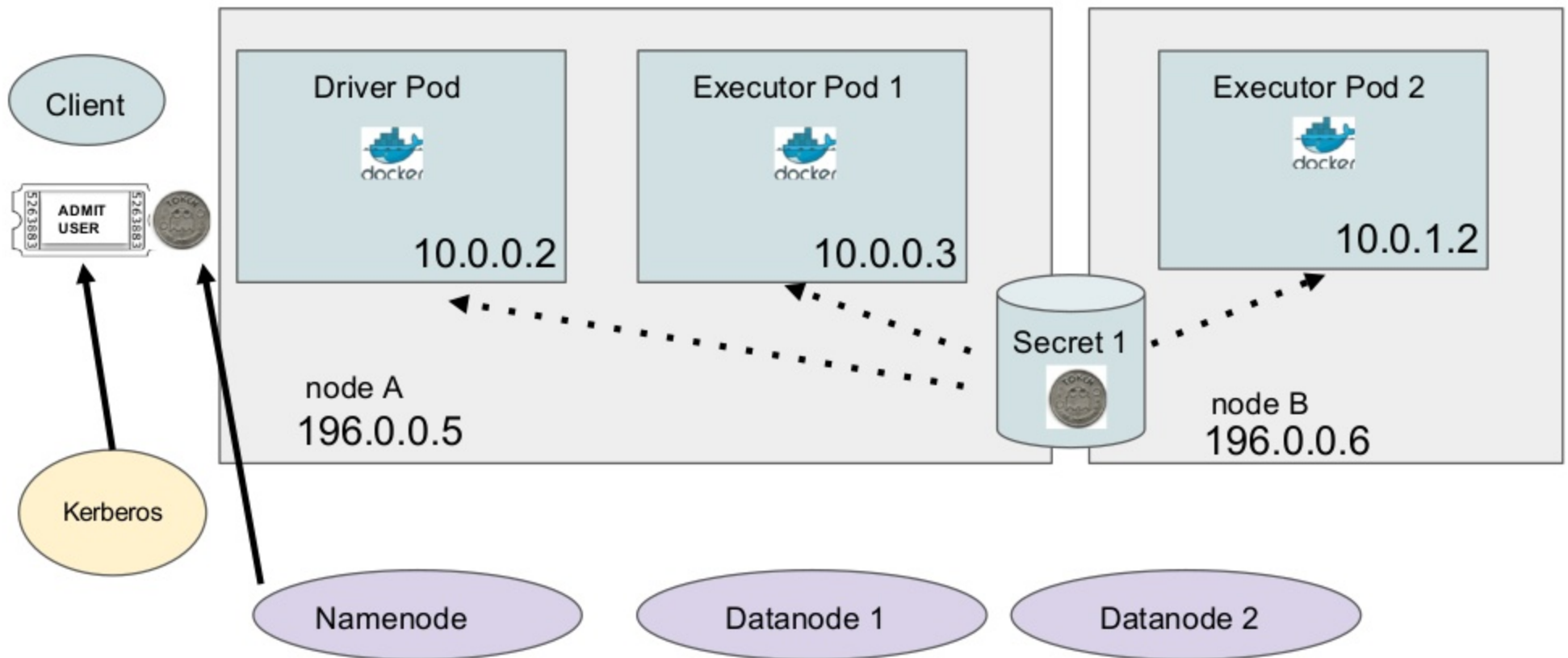
Give tokens to driver and executors instead.

- Issued by namenode only if the client has a valid Kerberos ticket.
- No client IP stamped.
- Permit for driver and executors to use HDFS on your behalf across all cluster nodes.

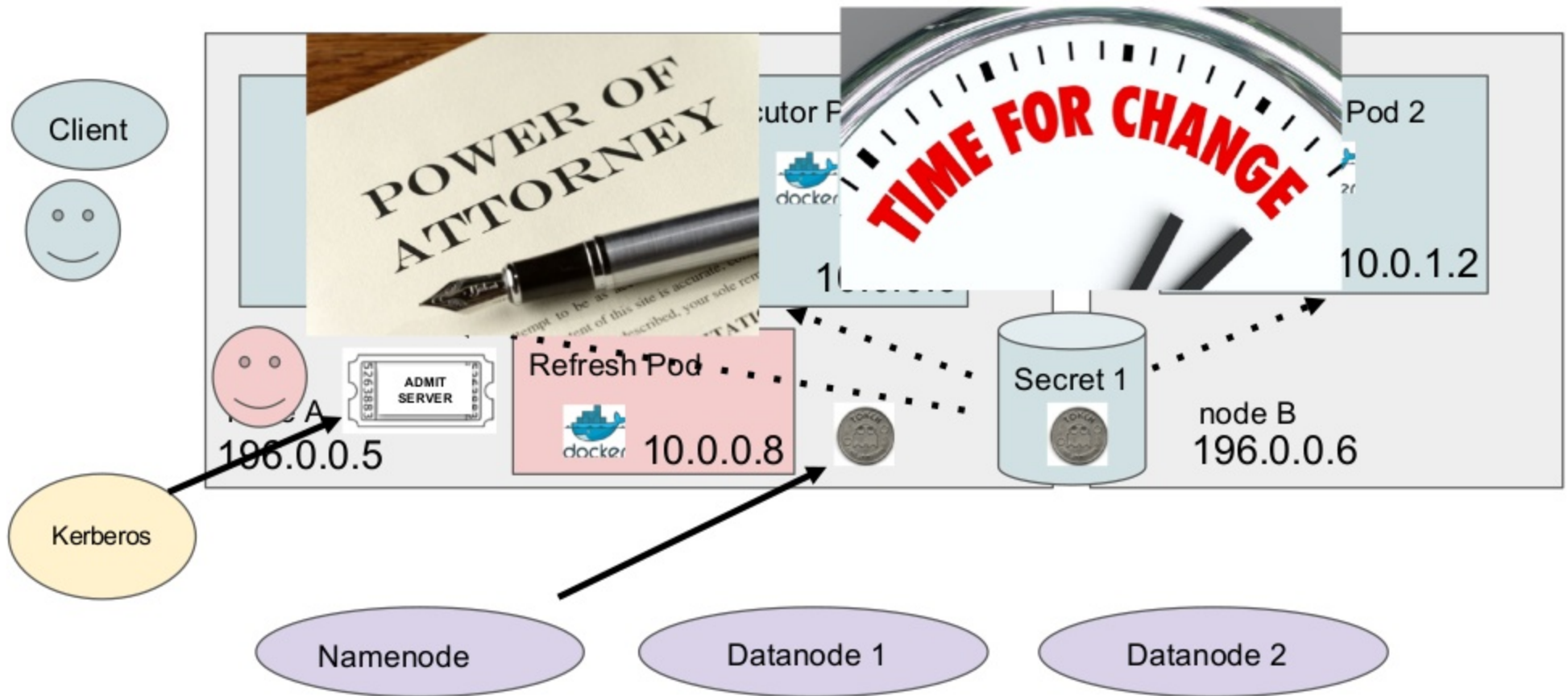




# Problem Statement: How to share secrets via K8s Secret

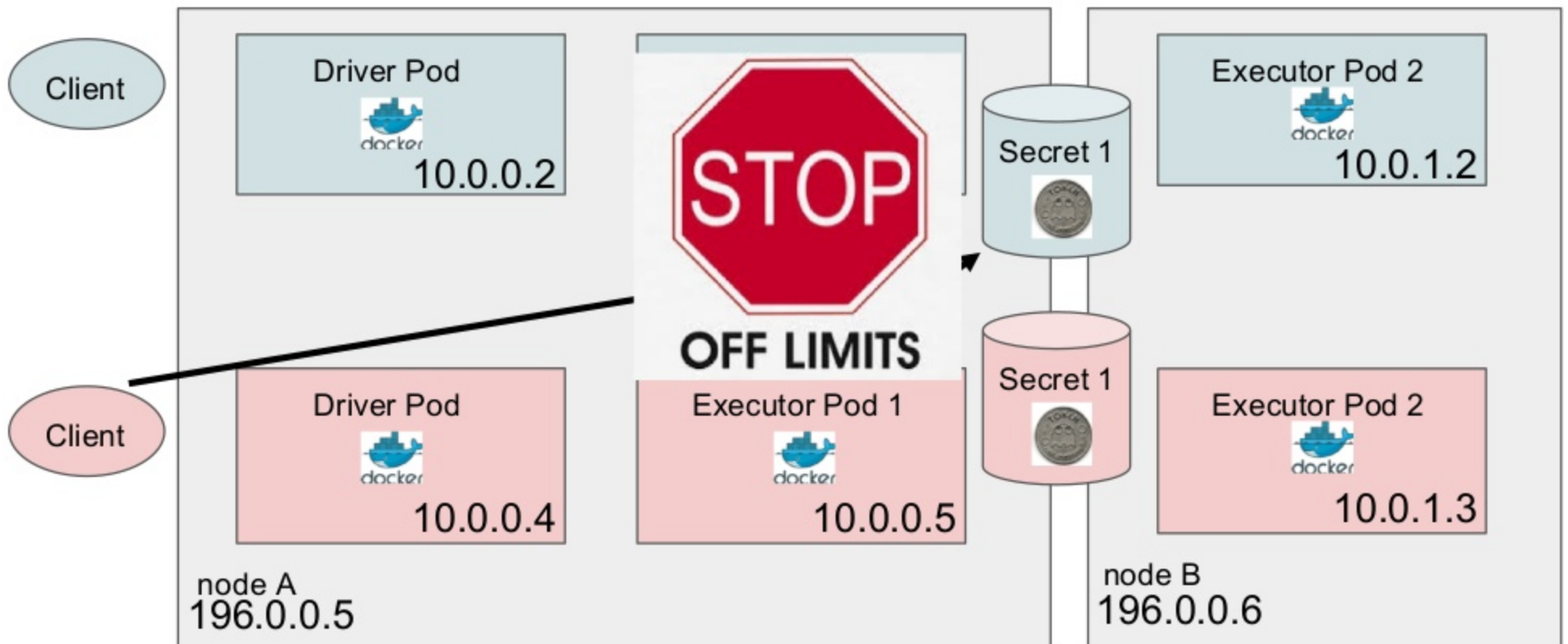


# When Refresh expires with K8s microservice



# Problem: Keep Secrets to yourself with RBAC

Job 1   
Job 2 



# Access Control of Secrets

HDFS DTs and renewal service keytab in Secrets

Admin can restrict access by:

1. Per-user AC, manual
2. Per-group AC, manual
3. Per-user AC (automated, upcoming)

	Job owner human user	Job owner's pods	Other human users	Other users' pods	Renew service pods
Access to the DT secret	create	get	<b>none</b>	<b>none</b>	get, update
Access to the renewal keytab secret	<b>none</b>	<b>none</b>	<b>none</b>	<b>none</b>	get

# Demo: Spark k8s Accessing Secure HDFS

## Running a Spark Job on Kubernetes accessing Secure HDFS

Single-noded pseudo-distributed Kerberized Hadoop Cluster

<https://github.com/ifilonenko/hadoop-kerberos-helm>

Spark Submit with Kerberos Configs

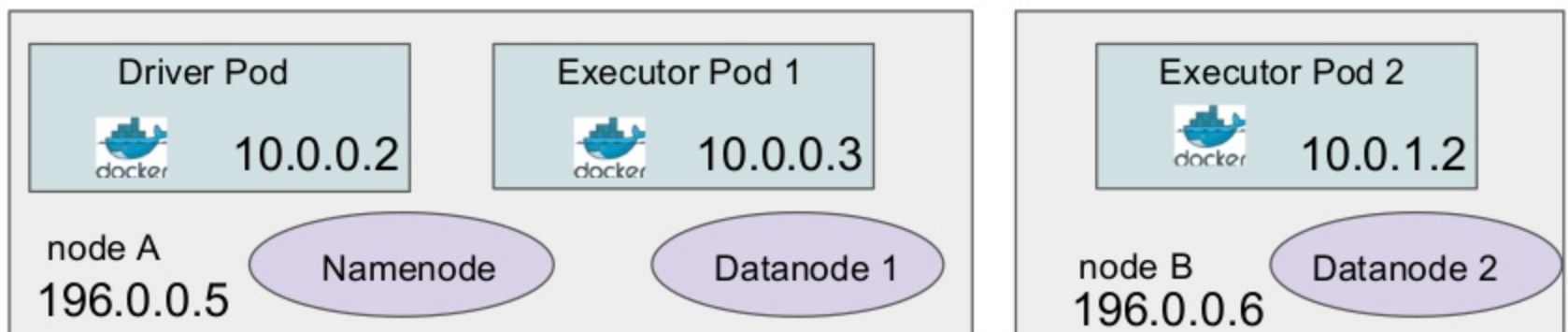
<https://github.com/ifilonenko/secure-hdfs-test>

Pre-defined Secrets

<https://asciinema.org/a/6YzzS6cP392iO3PnVo07yhHYk>

# Agenda

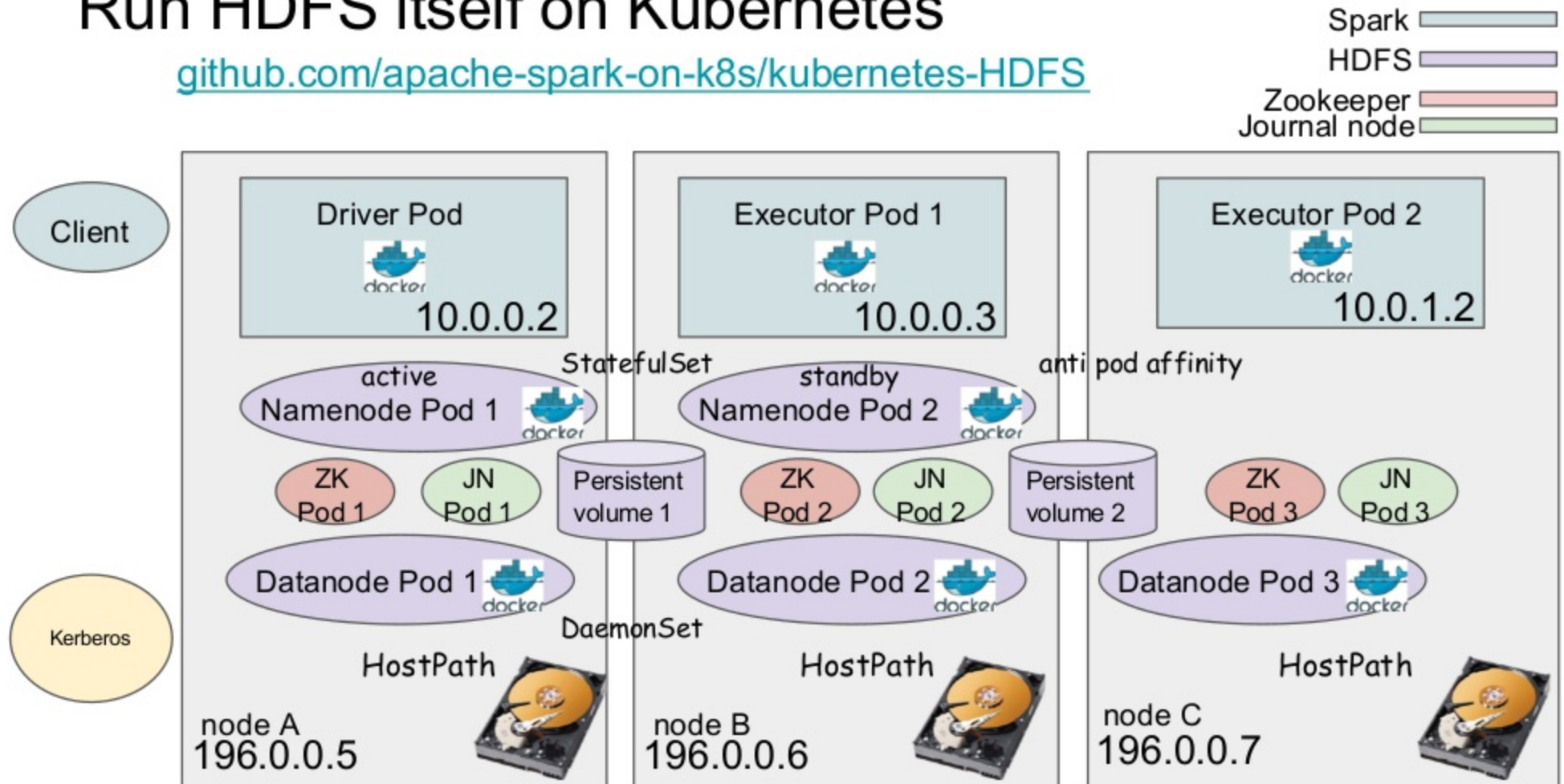
1. Kubernetes intro
2. Big Data on Kubernetes
3. Demo: Spark on K8s accessing secure HDFS
4. Secure HDFS deep dive
- 5. HDFS running on K8s**
- 6. Data locality deep dive**





# Run HDFS itself on Kubernetes

[github.com/apache-spark-on-k8s/kubernetes-HDFS](https://github.com/apache-spark-on-k8s/kubernetes-HDFS)

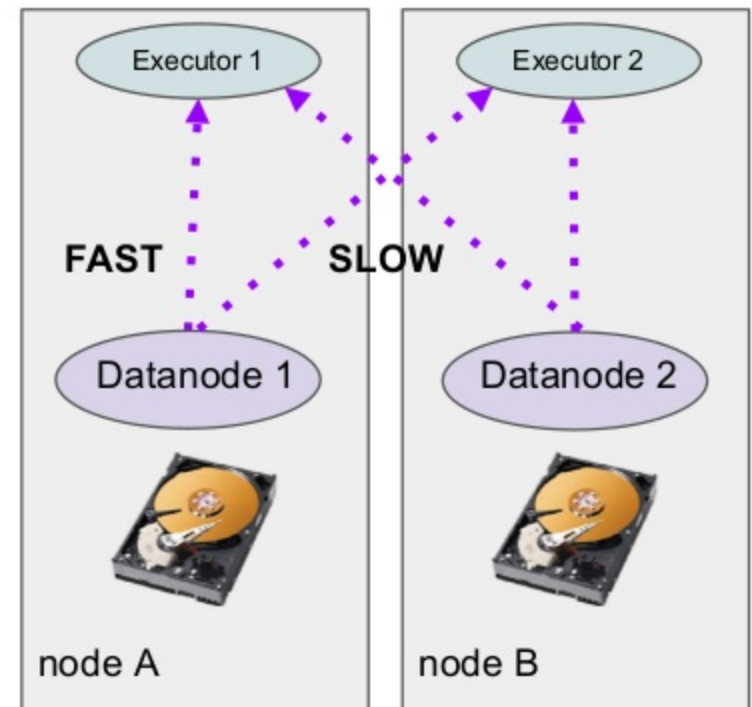


# Locality deep dive

Send compute to data

- Node locality
- Rack locality
- Where to launch executors

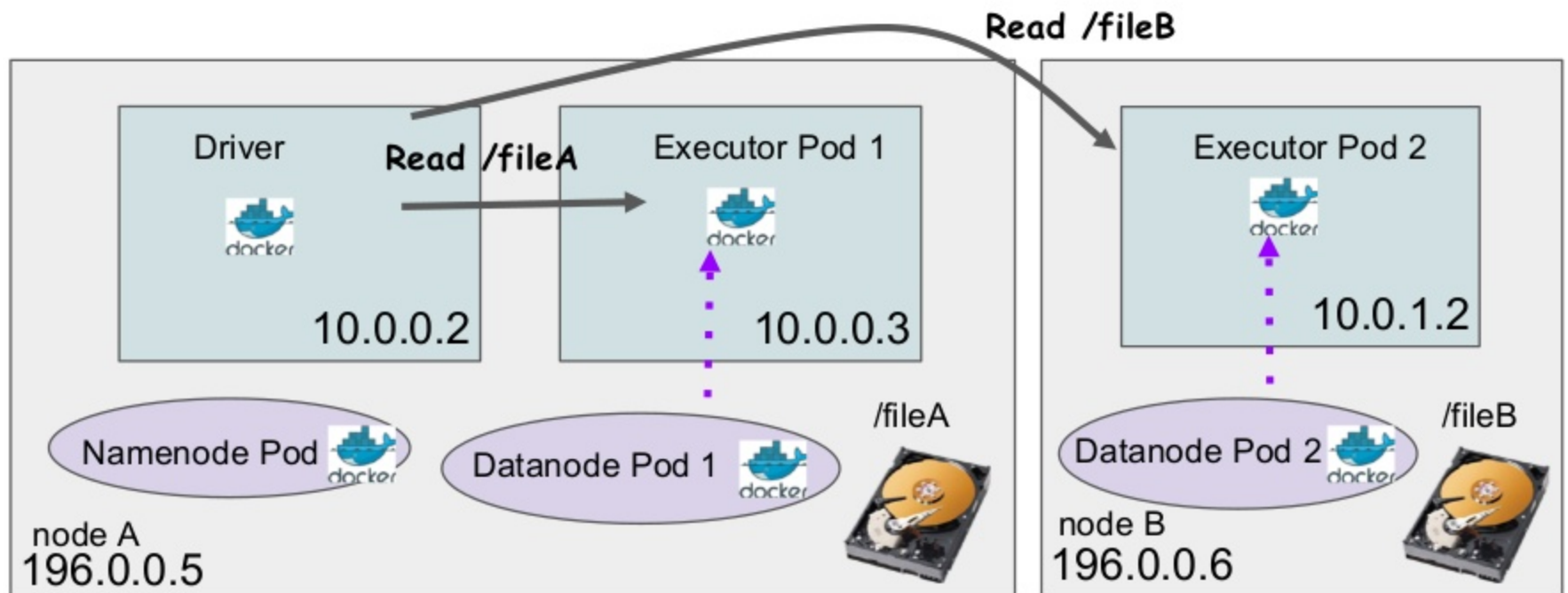
Spark on K8s had to be fixed





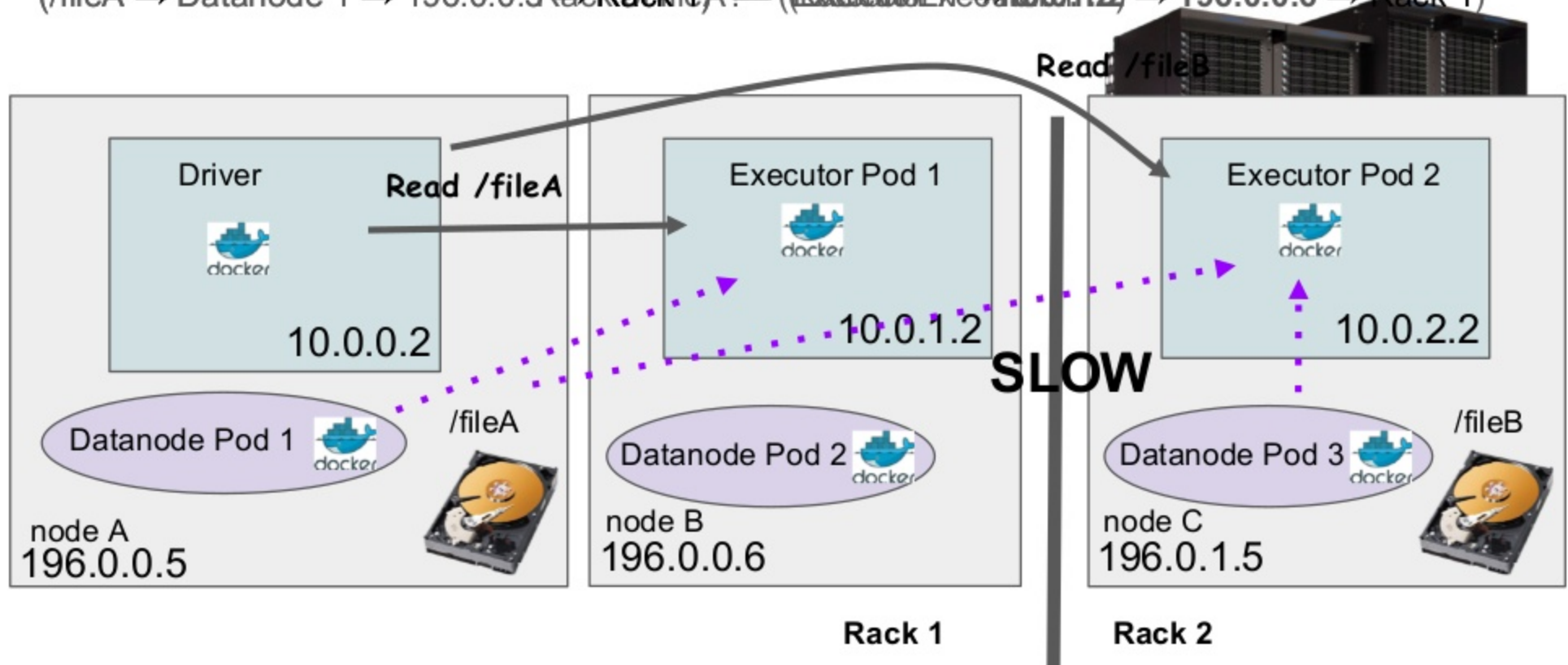
# Problem: Node locality broken with virtual pod IPs

$(/fileA \rightarrow \text{Datanode Location } 196.0.0.5) \neq (\text{Location of Executor } 10.0.0.3) \rightarrow 196.0.0.5)$



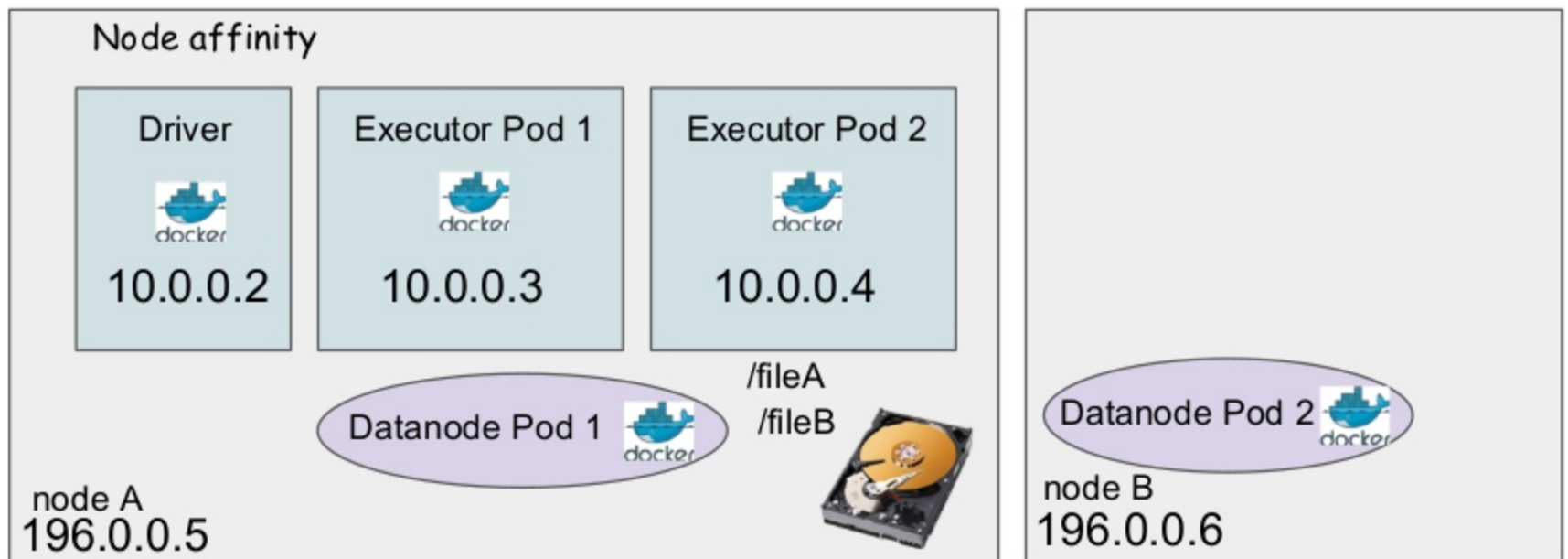
# Broken Rack locality with virtual pod IPs

(/fileA → Datanode 1 → 196.0.0.5 → Rack 1) ≠ (Executor 1 → 10.0.1.2 → 196.0.0.6 → Rack 1)



# Solved: Node preference

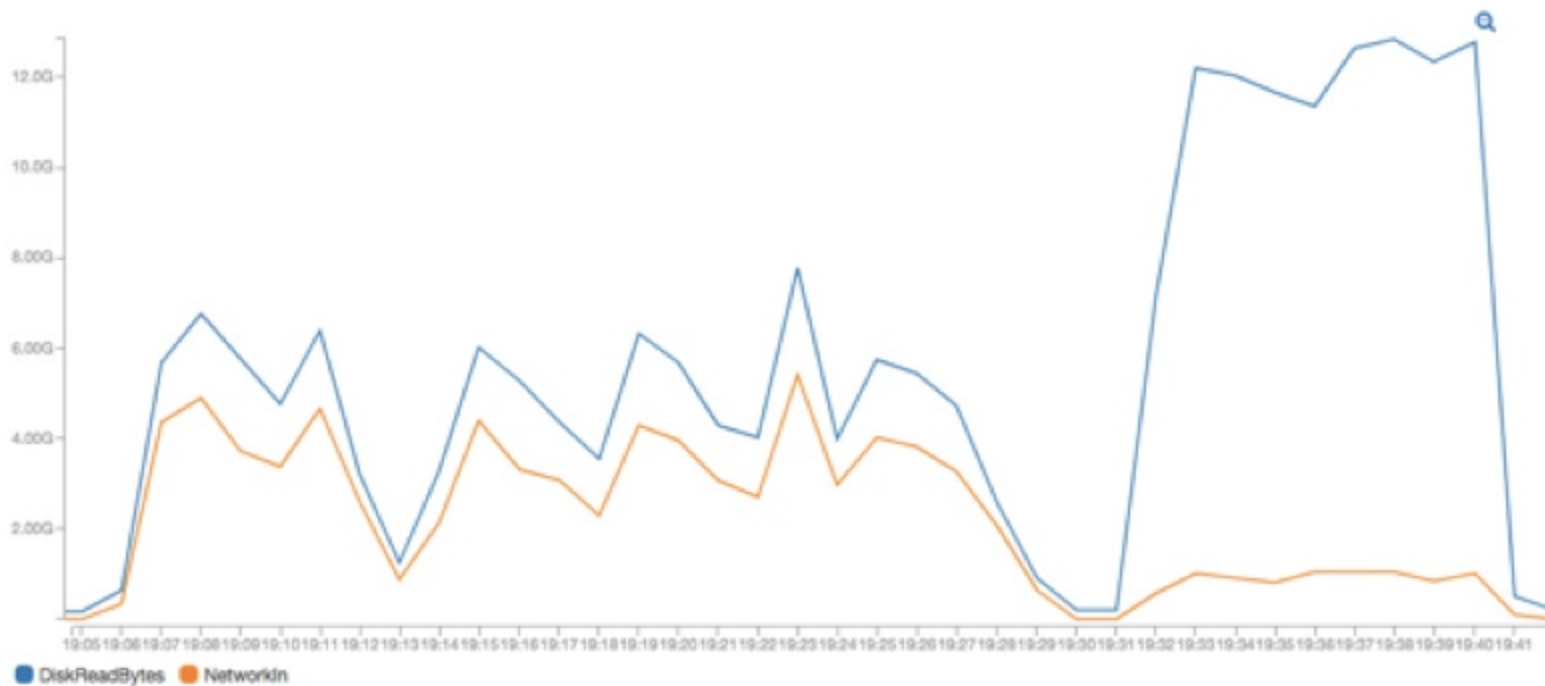
Hey K8s, I'd like node A much more for my executors



# Rescued data locality!

without data locality fix  
- duration: 25 minutes

with data locality fix  
- duration: 10 minutes



# Thank you!



Ilan Filonenko (ifilonenko@bloomberg.net)

## Bloomberg