Clustering Samba with Zookeeper and Cassandra

Richard Sharpe

Outline

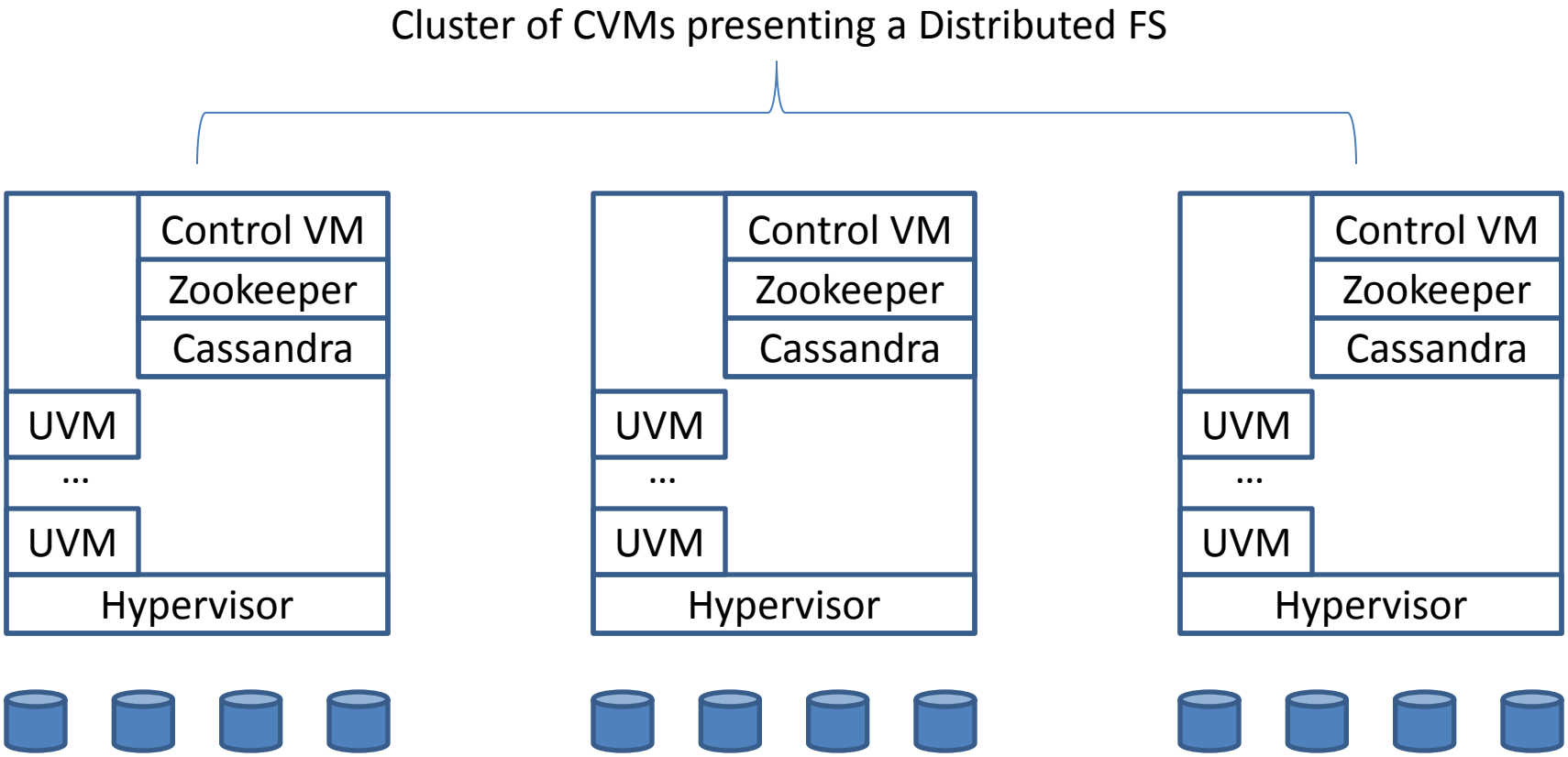
- What I’m doing
- Nutanix Environment
- Filer Need and Approach
- Sharding the file system
- Samba mods and system architecture
- Conclusions

What I’m doing

- Leading a small team doing a scale-out filer at Nutanix
- Doing clustering in a different way
- CentOS 6.x
- ZFS
- Zookeeper and Cassandra

Nutanix Environment

- Hyper-converged platform



Nutanix Environment, cont

- 3-32 nodes today (larger works)
  - Storage
    - Rotating and SSD
  - Compute
  - Memory
- Distributed File System (NDFS)
  - Provides medium number of large objects
  - 10^5 to 10^6 objects
  - 10^9+ bytes
- Basic object is a vDisk

Nutanix Environment, cont

- RF 2 or RF 3 and erasure coding
  - Data automatically distributed/replicated
- Stores small objects in Cassandra
  - Cassandra mods to provide Strong Consistency
- Metadata in Cassandra
- Zookeeper for distributed configuration and clustering support

Nutanix Environment, cont

- Protobufs
  - C++
  - Python
  - Java
- Three hypervisors supported
  - ESX, KVM and Hyper-V
- Nodes ship with KVM
  - Because VMware stopped us from shipping ESX
  - A single installer VM image uses customer ISOs

Needed a Filer

- Customers ask for NAS support
  - Some want NFS
  - Most want CIFS/SMB
  - Crazies want shared NFS and CIFS
- NDFS optimized for vDisks
  - VMDKs, VHDs, etc
  - Not good at tens of millions of smallish files
- CVMs use port 445 for HyperV support

NAS Filer Goals

- Provide Scale-out service
  - Initially for homes and profiles shares (VDI workload)
  - Eventually for ordinary shares
- Multiple filers per cluster
- Cluster of VMs
  - Single AD machine account
- High Availability (better than VMware's HA)
- Disaster Recovery support

NAS Filer Goals, cont

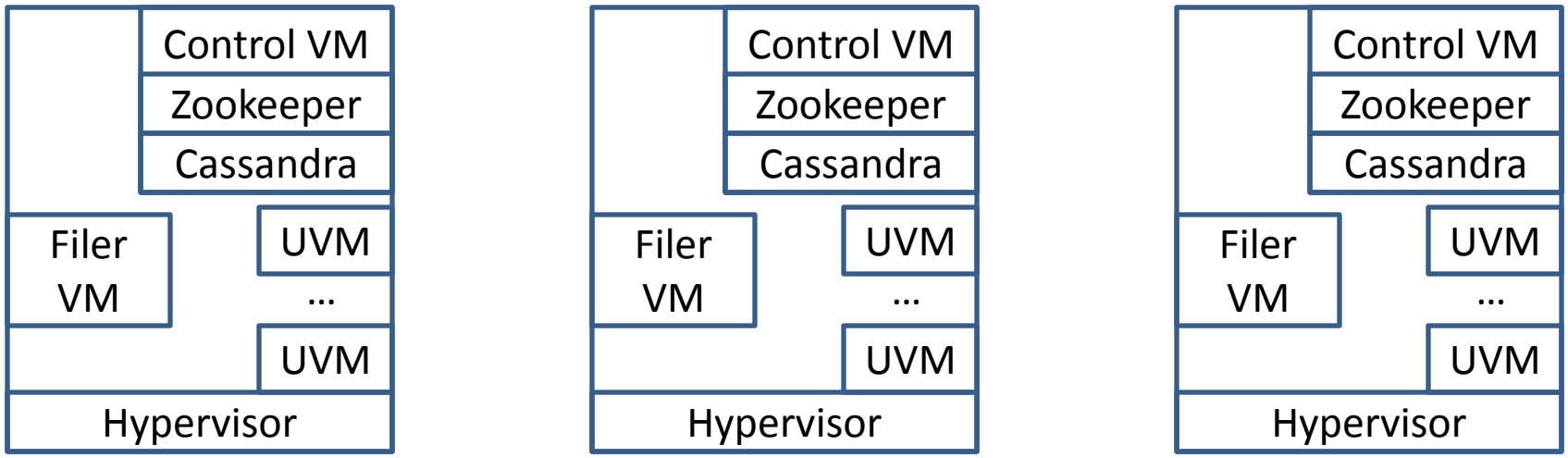
- Windows Previous Version
  - Three models controlled through config
    - Nobody (they use external backup/restore, eg NetBackup
    - BUILTIN/Administrators – Admin provided restore via WPV
    - Everyone – All users use WPV
  - Based around ZFS Snapshots

The solution

- Cluster of VMs
- Samba for SMB 2.1+
- ZFS on Linux as file system
- iSCSI on multiple vDisks
  - A ZPool spans multiple vDisks
  - Thinly provisioned
  - Increase storage by adding more disks to a ZPool

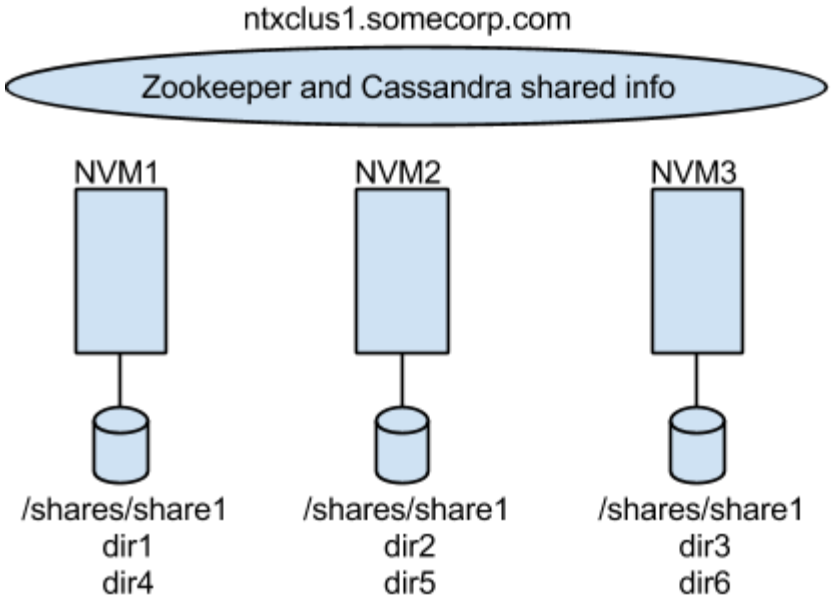
The solution, cont

- Add filer VMs to some nodes
- They form their own cluster

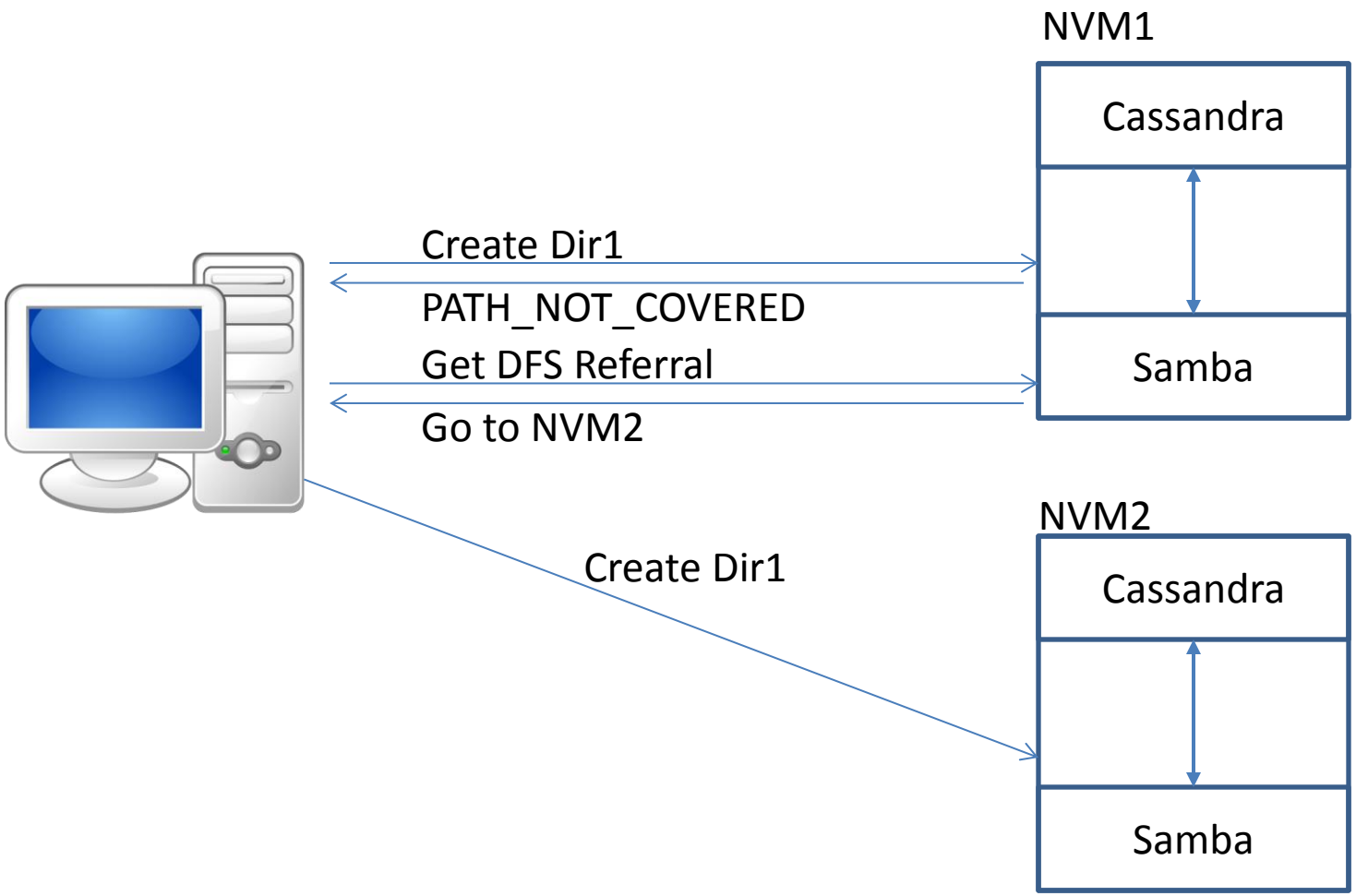


Basic Architecture

- Sharding of Shares across multiple nodes/VMs
  - Sharding at the root of shares only today
- Metadata in Cassandra, config in Zookeeper



Basic Sharding Approach



Benefits of sharding

- No need for a large scale shared file system
- Reduces need for shared locking information
  - Only needed at the sharding point
- Storage imbalance not really a problem
  - We have storage virtualization anyway
- Works well in VDI workloads
  - Homes and profiles directories close to VDI
- However, workload imbalance could happen

Why shard only at share root?

- Currently we only plan to shard at share root
- Simplifies the code
- Reduces the number of VFS referrals
  - Clients have limited cache size
  - Each referral increases CREATE latency
- Works well for VDI support

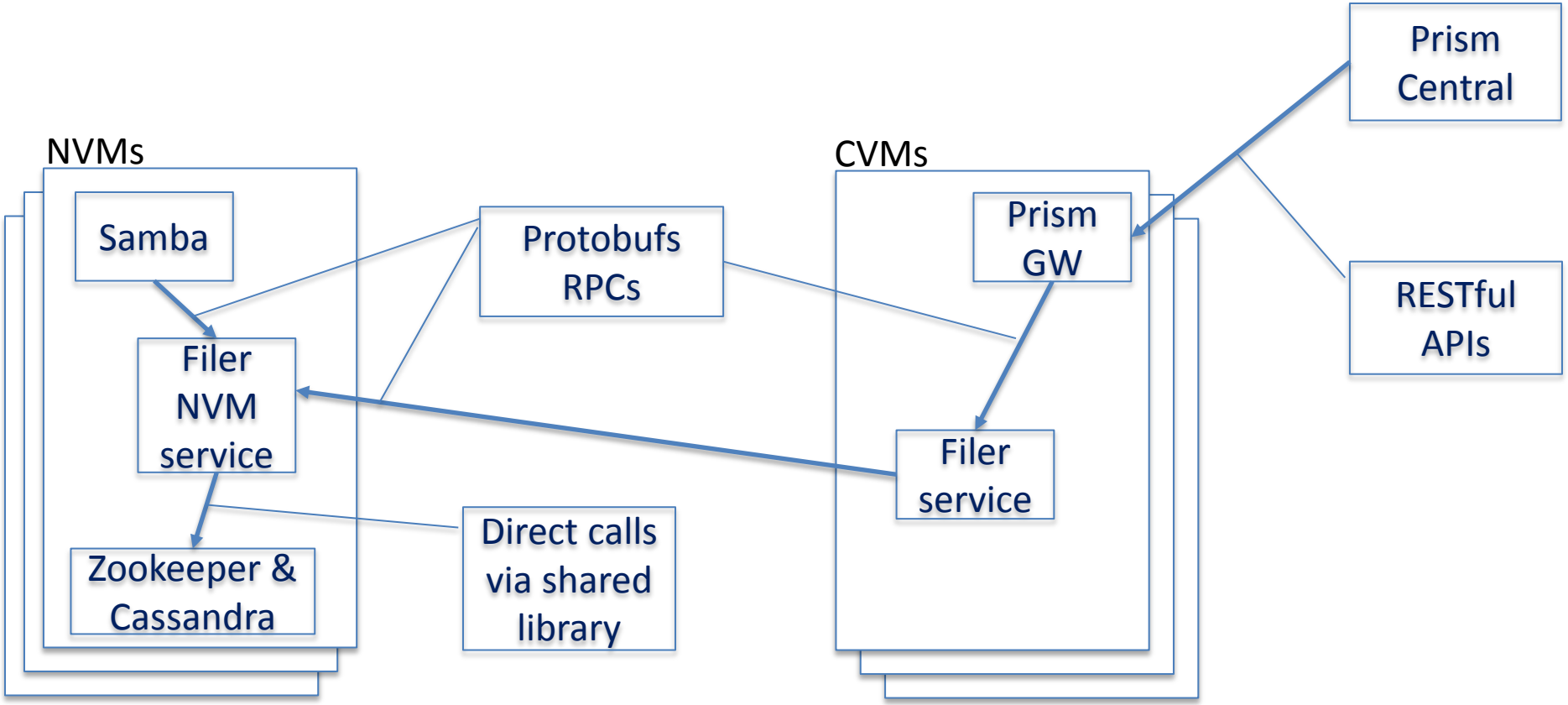
Shared information needed

- Still some shared information needed
- Configuration
- Secrets
- Metadata for the sharding point
  - Mappings
  - stat-like info
  - locking information
  - SD/ACL for root of share

Samba Config in Zookeeper

- All NVMs see the same config
- Similar to the current registry approach
- Already posted a config in Zookeeper patch
  - It has problems
    - Zookeeper client needs to reconnect across forks
    - When a change to the config changes smbds flood zookeeper with requests

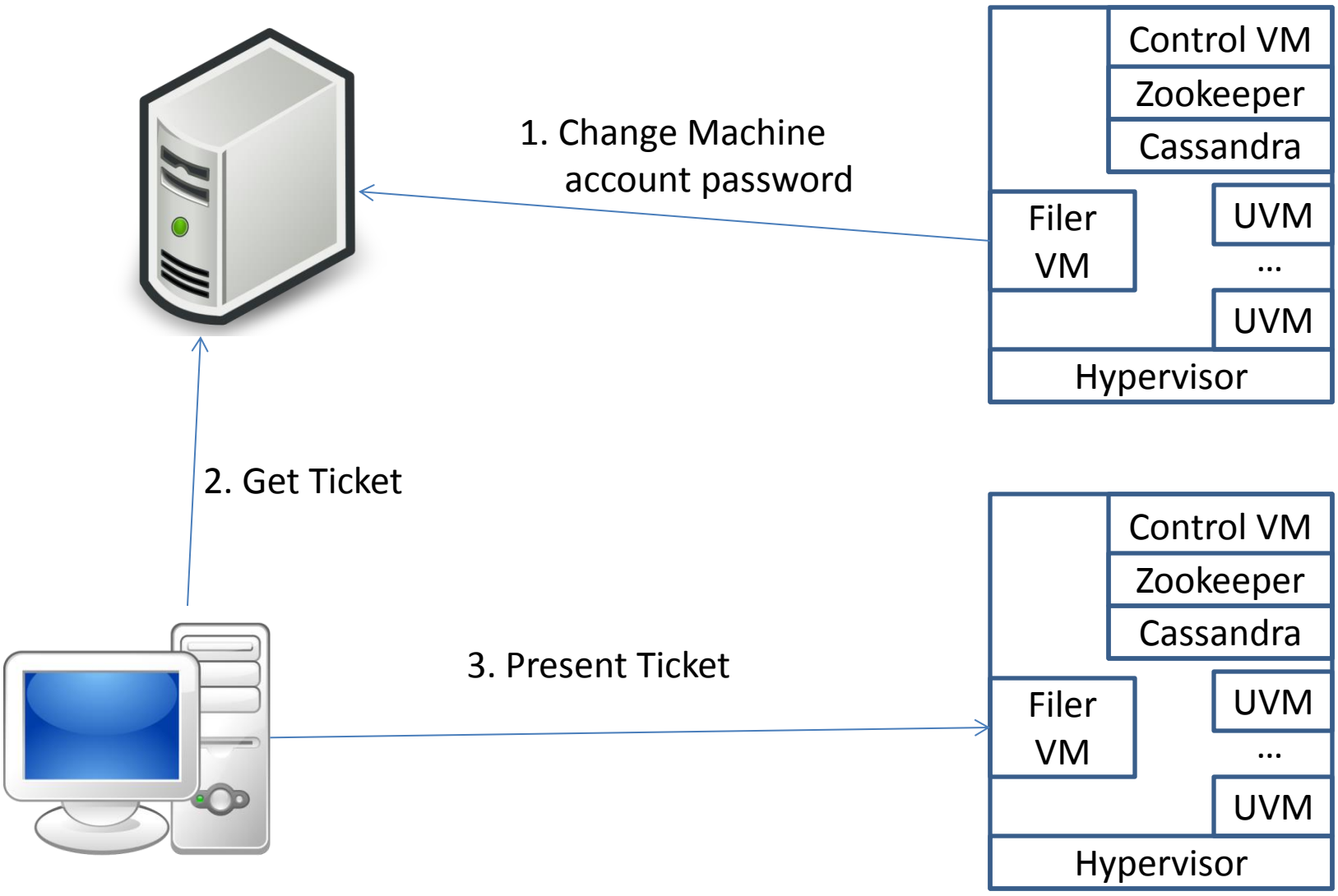
The approach



Secrets in Zookeeper

- Each NVM uses the same machine account
  - Add SPNs for each NVM as well as the cluster name SPN
    - Enabled single-sign-on with DFS referrals
- Will likely keep secrets in Zookeeper encrypted with a shared hash
- Have to deal with the races around changes to machine account password

Secrets in Zookeeper



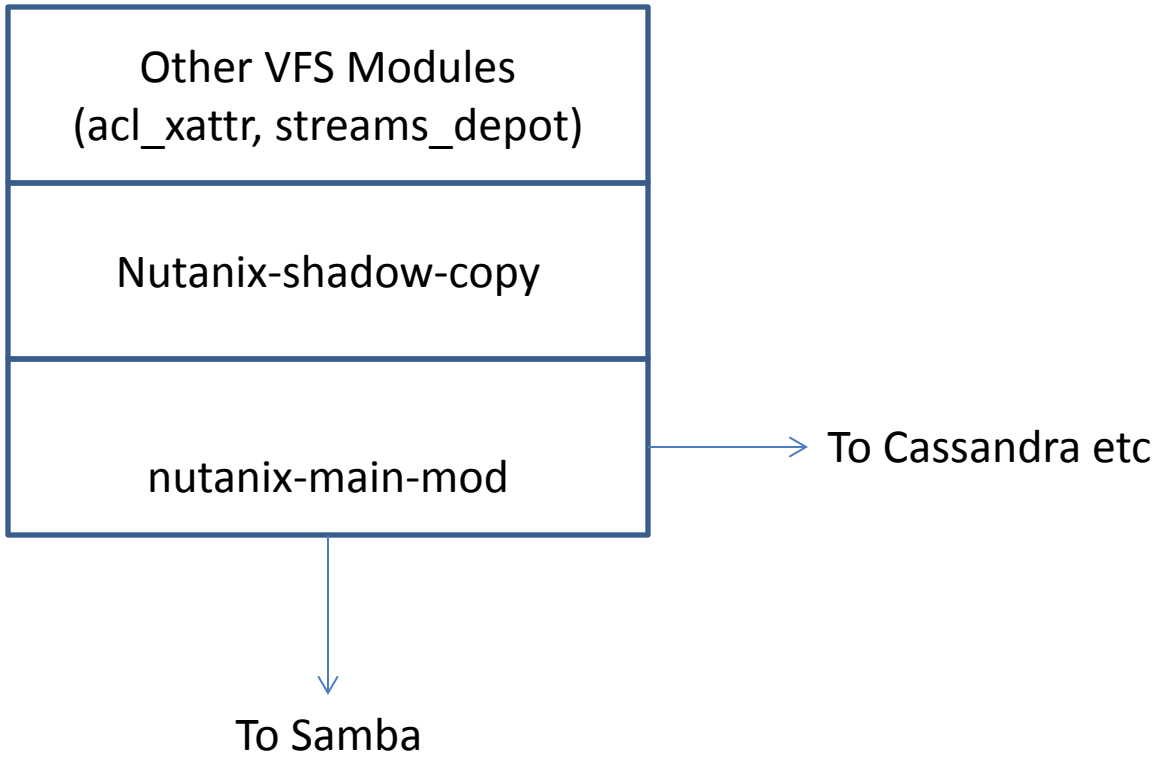
Metadata in Cassandra

- Need strong consistency
  - Nutanix has Multi-Paxos “tables”
- Mapping of object to its location
- Stat-info
- DOS attributes
- Locking info
  - Share-mode locks most important
- SD/ACL at the share root
- Share-level ACL

The VFS layer

- Most of our changes are in our VFS modules
- Realpath does heavy duty
- Stat just as important
- Must sit below other modules
- Can not let any calls through to Samba at the sharding point

The VFS layer, cont



Problems in the Samba VFS

- Lack of consistent error return codes
  - Some are UNIX, some are Windows
- Not all functions dealing with files get an FSP
  - Directory handling, for example
- Lack of information on when certain functions are called
  - REALPATH vs STAT

Other issues in Samba

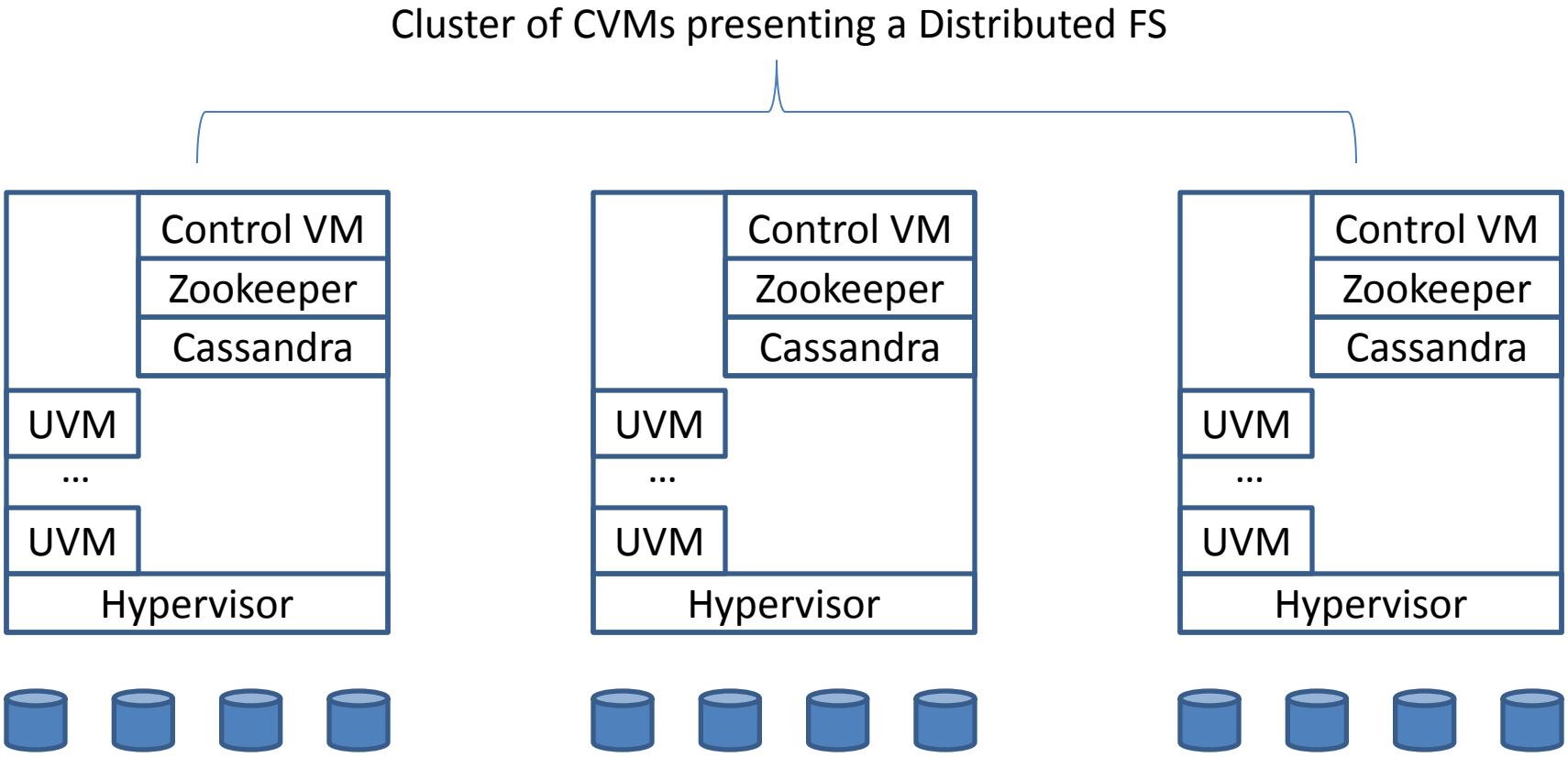
- Lack of exposed interfaces
  - Locking (Share modes and byte-range locks)
  - Secrets
  - Samba config
  - Share-level ACLs

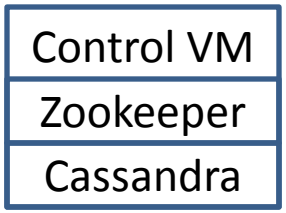
Problems with this approach

- Rename of objects at sharding point
- Delete of objects at the sharding point
- Current Windows clients won't do it
- There is a work-around
  - Go directly to the location of the object

Conclusions

- An interesting approach to a scale-out NAS
- Samba makes things easy
- Having fun again


Cluster of CVMs presenting a Distributed FS
Control VM
Zookeeper
Cassandra
UVM
...
UVM
Hypervisor
Control VM
Zookeeper
Cassandra
UVM
...
UVM
Hypervisor
Control VM
Zookeeper
Cassandra
UVM
...
UVM
Hypervisor


Control VM
Zookeeper
Cassandra