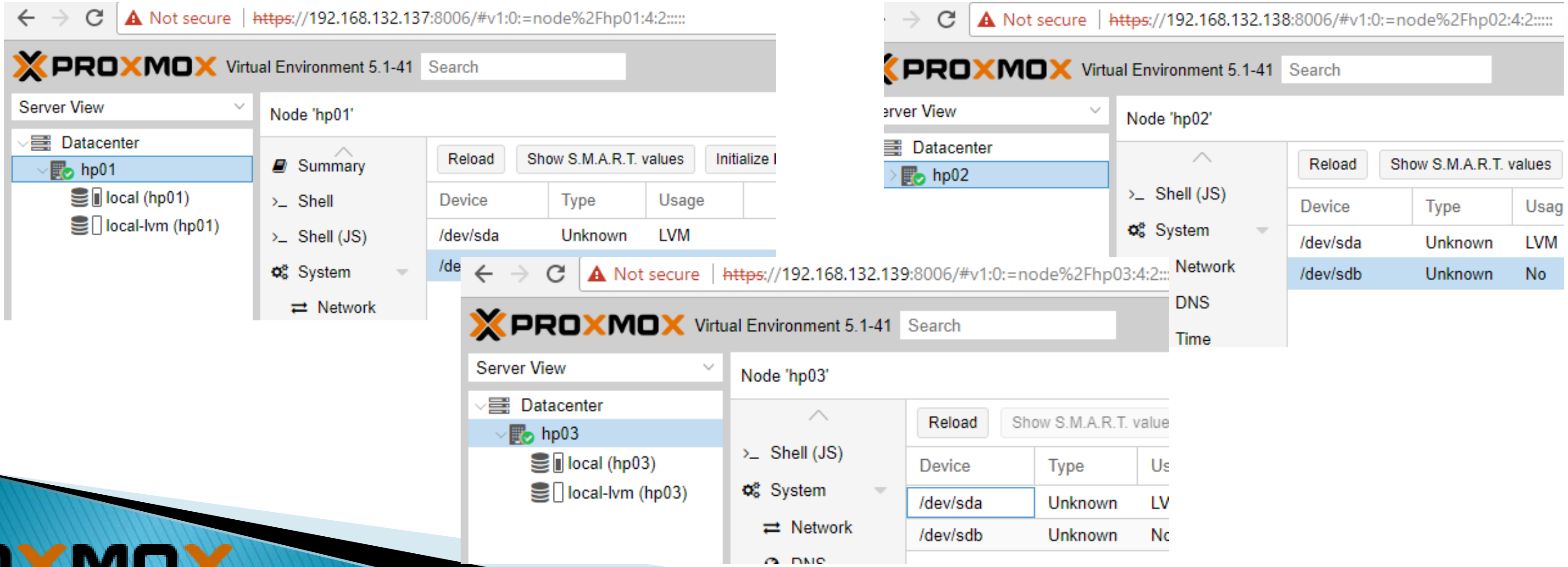
Proxmox VE 5

Lecture 10
Proxmox VE Cluster

Instructor: Hadi Alnabriss

Proxmox VE Cluster

- How can you manage multiple hosts in your datacenter?



Not secure | https://192.168.132.137:8006/#v1:0:=node%2Fhp01:4:2::::

Node 'hp01'

Device	Type	Usage
/dev/sda	Unknown	LVM
/de		

Not secure | https://192.168.132.138:8006/#v1:0:=node%2Fhp02:4:2::::

Node 'hp02'

Device	Type	Usag
/dev/sda	Unknown	LVM
/dev/sdb	Unknown	No

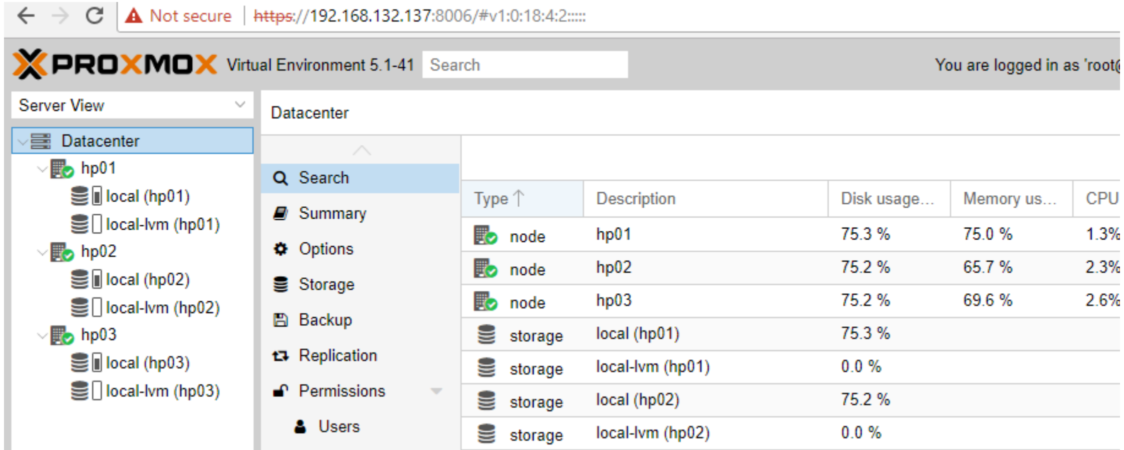
Not secure | https://192.168.132.139:8006/#v1:0:=node%2Fhp03:4:2::::

Node 'hp03'

Device	Type	Us
/dev/sda	Unknown	LV
/dev/sdb	Unknown	Nc

Clustering Advantages

- Grouping nodes into a cluster has the following advantages:
  - (1) Centralized, web based management
  - (2) Multi-master clusters: each node can do all management task



Type ↑	Description	Disk usage...	Memory us...	CPU
node	hp01	75.3 %	75.0 %	1.3%
node	hp02	75.2 %	65.7 %	2.3%
node	hp03	75.2 %	69.6 %	2.6%
storage	local (hp01)	75.3 %		
storage	local-lvm (hp01)	0.0 %		
storage	local (hp02)	75.2 %		
storage	local-lvm (hp02)	0.0 %		

Clustering Advantages

- Grouping nodes into a cluster has the following advantages:
  - (3) pmxcfs: database-driven file system for storing configuration files, replicated in real-time on all nodes using corosync.
  - (4) Easy migration of virtual machines and containers between physical hosts
  - (5) Fast deployment
  - (6) Cluster-wide services like firewall and HA

Requirements

- (1) All nodes must be in the same network
  - Because *corosync* uses IP Multicast to communicate between nodes.
  - Corosync uses UDP ports 5404 and 5405 for cluster communication.

- What is corosync?
  - It is the communication system for nodes in the cluster
  - Note : Check the corosync service status

Requirements

- (2) Time Synchronization.
- (3) SSH tunnel on TCP port 22 between nodes is used.

Additional Requirements

- If you are interested in High Availability, you need:
  - At least three nodes for reliable quorum.
  - All nodes should have the same version.

- Also, We recommend a dedicated NIC for the cluster traffic, especially if you use shared storage.

Preparing Nodes

- Install Proxmox VE 5 on 3 nodes (use 20GB for Disk, 1GB for RAM and NAT for NIC)
  - Make sure that each node is installed with the final hostname and IP configuration.
  - Note (1): Changing the hostname and IP is not possible after cluster creation.
  - Note (2): We will use the hosts file to define hostname and IP

Prepare hosts file

- On each node prepare the hosts file to recognize all the nodes in your cluster

node01	192.168.132.133
node02	192.168.132.134
node03	192.168.132.135

Create the Cluster using Cluster Manager

- *"pvecm"* can be used to:
  - Create a new cluster,
  - Join nodes to a cluster,
  - Leave the cluster,
  - Get status information
  - Other various cluster related tasks.

Create the Cluster

- No GUI
  - You cannot create the cluster from GUI

Create the Cluster

- On any node
  - *node01# pvecm create YOUR-CLUSTER-NAME*

- On the other two nodes
  - *node02#pvecm add* IP-ADDRESS-CLUSTER
  - *node03#pvecm add* IP-ADDRESS-CLUSTER

  - Note: use the IP address from an existing cluster node (node01 in our case or IP 192.168.132.133).

Cluster Status

- To check the cluster status
  - pvecm status
- To see nodes in the cluster
  - pvecm nodes

Delete Node

- A cluster includes the nodes:
  - node01
  - node02
  - node03
  - node04
- To delete node04
  - power off node04
  - from (node01, node02 or node03):
    - pvecm delnode node04

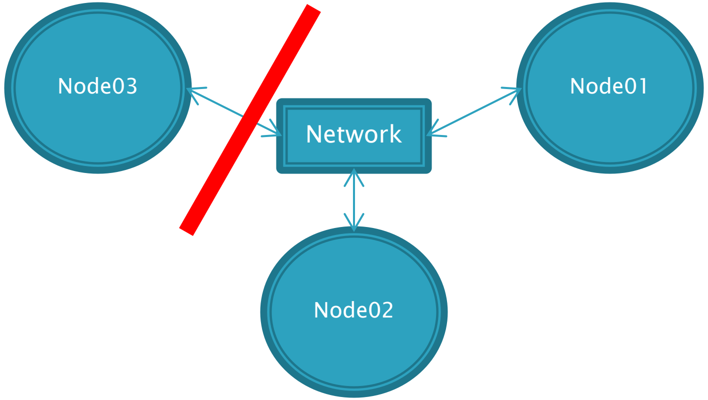
Quorum

- Proxmox VE use a quorum-based technique to provide a consistent state among all cluster nodes.

Quorum

- A quorum is the minimum number of votes that a distributed transaction has to obtain in order to be allowed to perform an operation in a distributed system.

- In case of network partitioning, state changes requires that a majority of nodes are online. The cluster switches to read-only mode if it loses quorum.



Cluster Network

- The cluster network is the core of a cluster.
- All messages sent over it have to be delivered reliable to all nodes in their respective order.
- In Proxmox VE this part is done by corosync, an implementation of a high performance low overhead high availability development toolkit.
- It serves our decentralized configuration file system (pmxcfs).

Cluster Network

- This needs a reliable network with latencies under 2 milliseconds (LAN performance) to work properly.
- While corosync can also use unicast for communication between nodes its **highly recommended** to have a multicast capable network.
- The network should not be used heavily by other members, ideally corosync runs on its own network.

Corosync Configuration

- The /etc/pve/corosync.conf file plays a central role in Proxmox VE cluster.
- It controls the cluster member ship and its network.
- For safety: use the *pvecm* command to configure your cluster

Cluster File System (pmxcfs)

- The Proxmox Cluster file system (“pmxcfs”) is a database-driven file system for storing configuration files.
- Files are replicated in real time to all cluster nodes using corosync.
- We use this to store all PVE related configuration files.
- The file system is mounted at /etc/pve

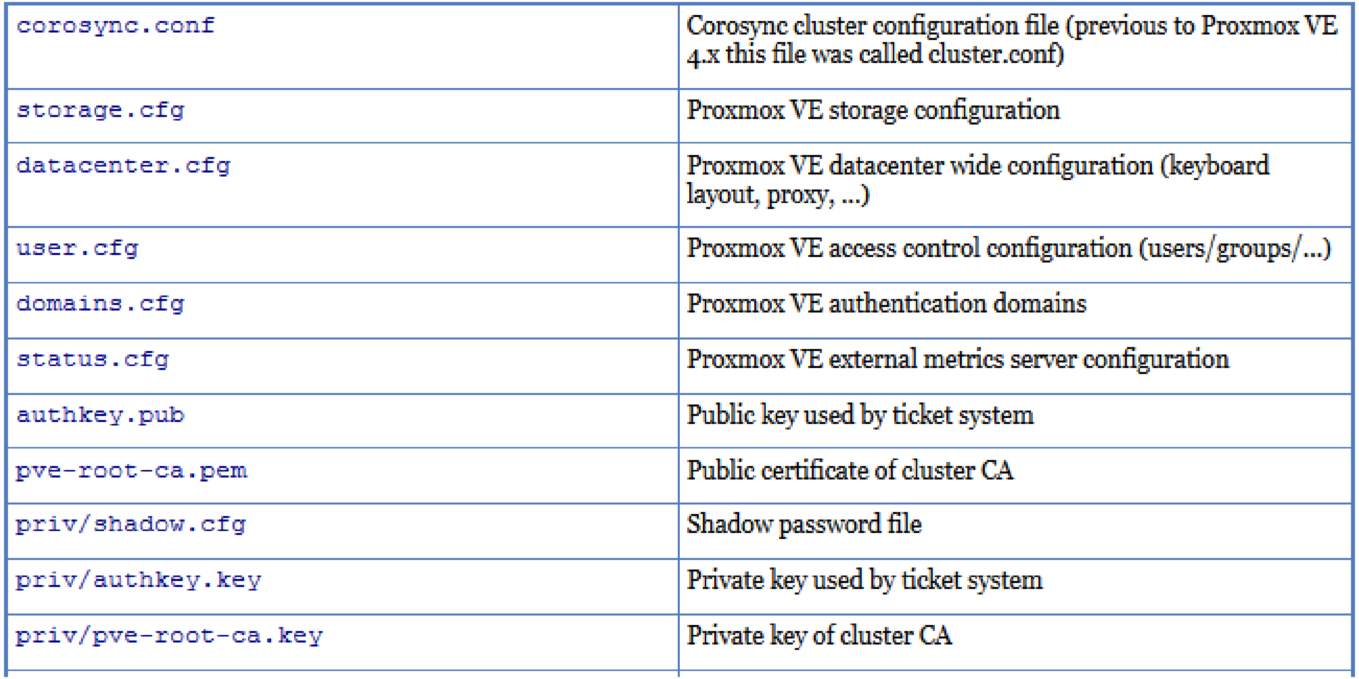
Cluster File System (pmxcfs)

- Although the file system stores all data inside a persistent database on disk, a copy of the data resides in RAM.
- That imposes restriction on the maximum size, which is currently 30MB.
- This is still enough to store the configuration of several thousands of virtual machines.

Cluster File System (pmxcfs) Advantages

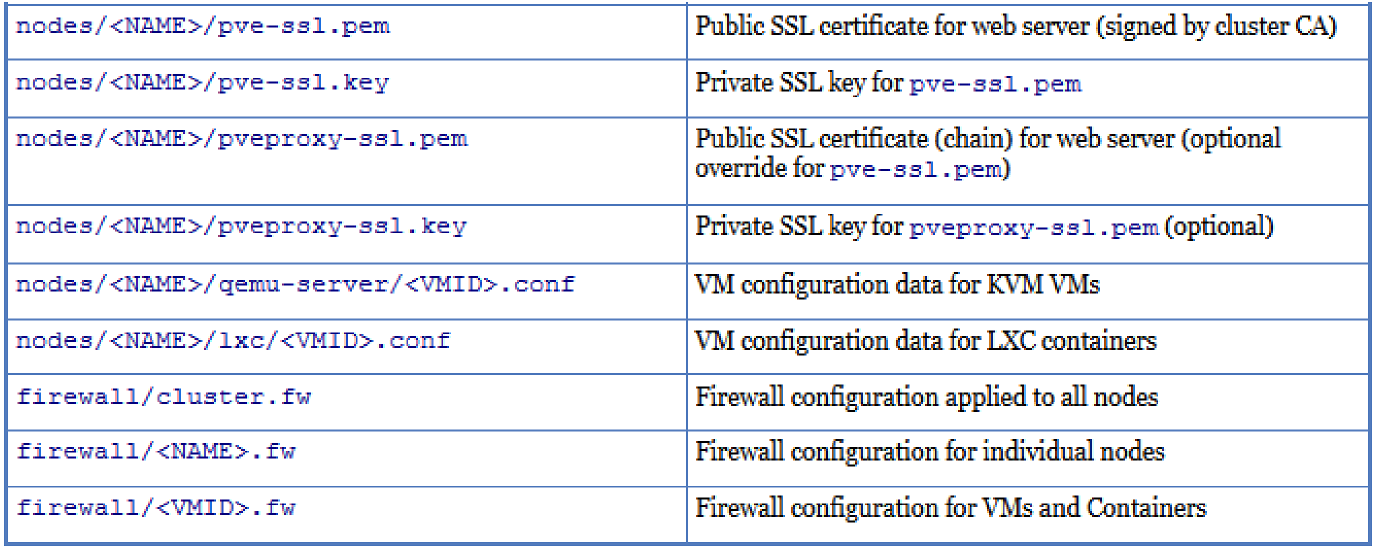
- This system provides the following advantages:
  - seamless replication of all configurations to all nodes in real time
  - provides strong consistency checks to avoid duplicate VM IDs
  - read-only when a node loses quorum
  - automatic updates of the corosync cluster configuration to all nodes
  - includes a distributed locking mechanism

Cluster Filesystem Files



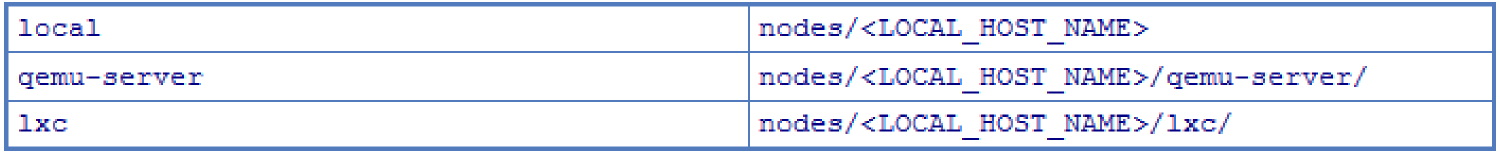
`corosync.conf`	Corosync cluster configuration file (previous to Proxmox VE 4.x this file was called cluster.conf)
`storage.cfg`	Proxmox VE storage configuration
`datacenter.cfg`	Proxmox VE datacenter wide configuration (keyboard layout, proxy, ...)
`user.cfg`	Proxmox VE access control configuration (users/groups/...)
`domains.cfg`	Proxmox VE authentication domains
`status.cfg`	Proxmox VE external metrics server configuration
`authkey.pub`	Public key used by ticket system
`pve-root-ca.pem`	Public certificate of cluster CA
`priv/shadow.cfg`	Shadow password file
`priv/authkey.key`	Private key used by ticket system
`priv/pve-root-ca.key`	Private key of cluster CA

Cluster Filesystem Files



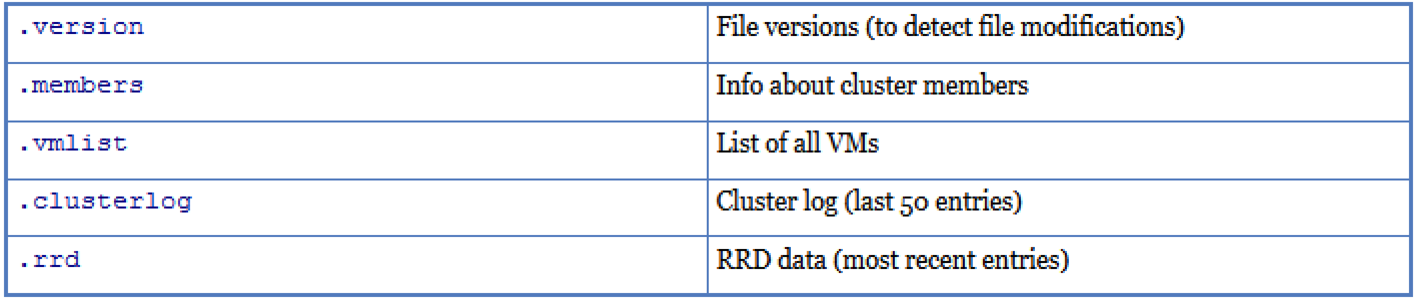
`nodes/<NAME>/pve-ssl.pem`	Public SSL certificate for web server (signed by cluster CA)
`nodes/<NAME>/pve-ssl.key`	Private SSL key for `pve-ssl.pem`
`nodes/<NAME>/pveproxy-ssl.pem`	Public SSL certificate (chain) for web server (optional override for `pve-ssl.pem`)
`nodes/<NAME>/pveproxy-ssl.key`	Private SSL key for `pveproxy-ssl.pem` (optional)
`nodes/<NAME>/qemu-server/<VMID>.conf`	VM configuration data for KVM VMs
`nodes/<NAME>/lxc/<VMID>.conf`	VM configuration data for LXC containers
`firewall/cluster.fw`	Firewall configuration applied to all nodes
`firewall/<NAME>.fw`	Firewall configuration for individual nodes
`firewall/<VMID>.fw`	Firewall configuration for VMs and Containers

Symbolic links



local	nodes/<LOCAL_HOST_NAME>
qemu-server	nodes/<LOCAL_HOST_NAME>/qemu-server/
lxc	nodes/<LOCAL_HOST_NAME>/lxc/

Special status files for debugging (JSON)



`.version`	File versions (to detect file modifications)
`.members`	Info about cluster members
`.vmlist`	List of all VMs
`.clusterlog`	Cluster log (last 50 entries)
`.rrd`	RRD data (most recent entries)

Conclusion

- Now you must be:
  - Able to create PVE cluster (add and delete nodes)
  - Understand Quorum
  - Understand Proxmox Cluster Filesystem