Networking

- Backplane Separation
  - Backplane Network Segmentation
  - IP Pools
- AHV Networks (acli)
- Open vSwitch
- IP / Routing & Crash Cart

Backplane Separation

Network Segmentation: CVM/host backplane and service traffic vom Management-Netz trennen, plus die IP-Pools, die dabei eine Rolle spielen.

Backplane Network Segmentation

What it is

By default all Controller VM (CVM) traffic shares one interface (CVM eth0): management (Prism, SSH, Rsyslog, SNMP, PE↔PC) and CVM↔CVM cluster-service traffic (Stargate, Cassandra, Zookeeper, Curator) all go out the same NIC and VLAN. **Network Segmentation** moves specific traffic types onto their own dedicated vNIC and VLAN, physically and logically separated from the management network.

Traffic types you can segment

- **Management** (default, eth0) — Prism, SSH, Rsyslog, SNMP, PE↔PC. Anything that needs the default gateway.
- **Backplane** (eth2) — CVM↔CVM, Host↔Host and Host↔CVM cluster-service traffic. Always an isolated, **non-routable** VLAN/subnet, kept separate from every other network in use.
- **Service** — an AOS feature you choose to move onto its own network, e.g. **Volumes** (iSCSI data services) or **DR/replication** (Cerebro). Unlike Backplane, this network is usually routable — DR traffic has to reach a remote cluster.
- **RDMA** — Stargate-only, uses physical NIC passthrough instead of a vNIC. Separate topic from the three above.

Prerequisites

- At least one spare physical NIC per node beyond what the management virtual switch (`vs0` / `br0`) already uses — backplane traffic needs its own uplinks, not a VLAN trunked over the management NICs.
- A second virtual switch (e.g. `vs1` / `br1`) built from those spare NICs, present on every node in the cluster.
- CVM and AHV host management interfaces (eth0) must stay on the same subnet/VLAN as each other — segmentation only ever pulls extra traffic off eth0, it never changes the management network itself.
- All hosts reachable and able to enter maintenance mode one at a time — segmentation is applied per host with a rolling CVM restart.

Enabling it (Prism Element)

Settings (gear icon) → Network Configuration → Internal Interfaces → "Backplane LAN" → Configure.

- VLAN ID — a VLAN not used anywhere else on the network.
- Subnet + gateway — Backplane itself never routes anywhere, but the wizard still wants a subnet/mask.
- Virtual switch to bind it to (`vs1` from the prerequisites above).
- IP address pool for the interface — see the *IP Pools* page in this chapter. Prism assigns one address per CVM (and per host, on current AOS) out of this range automatically.

Put hosts into maintenance mode one at a time before configuring, exit again once each CVM has restarted onto the new interface:

```
acli host.enter_maintenance_mode <host_ip>
# ... configure/verify backplane on this host ...
acli host.exit_maintenance_mode <host_ip>
```

Multiple clusters / Prism Central

For Service (DR) segmentation spanning a replication pair, Prism Central's Network Segmentation workflow can push the same VLAN/subnet config to several clusters at once — useful to keep both sides of the pair consistent.

IP Pools

Two different "IP pools" — don't mix them up

Nutanix reuses the term "IP pool" for two unrelated things. Both come up when you're doing backplane/segmentation work, but they address completely different traffic.

	Segmentation IP pool	AHV VM network IPAM pool
Addresses	CVM (and host) backplane/service vNICs	User VMs on an AHV-managed network
Configured in	Settings → Network Configuration → Internal Interfaces (or Prism Central Network Segmentation)	Prism → Network Config → the VM subnet, or `acli`
Routable?	No for Backplane, yes for Service/DR	Depends on the VM network's own VLAN
Assignment	Automatic, one IP per CVM/host, no DHCP involved	Nutanix-managed DHCP hands addresses to VMs on boot

Segmentation IP pool (Backplane / Service)

When you enable Backplane or Service segmentation, Prism asks for a start/end range, e.g.:

```
192.168.21.11 - 192.168.21.20
```

Size it for the cluster's current node count plus headroom for future expansion — one address is consumed per CVM (and, on current AOS, per AHV host) for each segmented traffic type. There's no reuse or lease/DHCP mechanism here: once an address is assigned it stays assigned until the interface is reconfigured.

AHV VM network IPAM pool

Separate from the above — this is Nutanix-managed DHCP for guest VMs on a given AHV network, already referenced on *Important Commands* in this book:

```
acli net.create <name> vlan=<vlan_id> ip_config=<subnet/prefix>
```

The command above only creates the network/subnet — the actual address pool for it is added separately:

```
acli net.add_dhcp_pool <name> start=<first_ip> end=<last_ip>
```

or in Prism: the VM network's config page → "IP Address Pools" → Create Pool.

Gotcha: `.254` in the subnet is reserved for the Acropolis DHCP server itself if you didn't explicitly set a DHCP server address when creating the network — don't include it in the pool range.

This pool has nothing to do with the CVM/host backplane above — it only ever hands out addresses to VMs that sit on this particular AHV network.

AHV Networks (acli)

AHV Networks (acli)

List all AHV networks

```
acli net.list
```

Create a new VLAN network

```
acli net.create <name> vlan=<vlan_id>
```

Create a network with IPAM (Nutanix-managed DHCP)

```
acli net.create <name> vlan=<vlan_id> ip_config=<subnet/prefix>
```

Create NIC with Intel E100:

```
acli vm.nic_create <VM_NAME> network=<NETWORK_NAME> mac=<NEW_MAC_ADDRESS> model=e1000
```

Open vSwitch

Open vSwitch (on the AHV host)

Show a host's uplink configuration

```
manage_ovs show_uplinks
```

Adjust uplinks

```
manage_ovs --interfaces eth0,eth1 update_uplinks
```

Show bridges and ports

```
ovs-vsctl show
```

Show bonding status

```
ovs-appctl bond/show <bond_name>
```

IP / Routing & Crash Cart

IP / Routing (CVM)

Default gateway and routing table on a CVM

```
netstat -rn
```

List ARP entries

```
arp -n
```

Delete a single ARP entry

```
arp -d x.x.x.x
```

Flush the entire ARP cache

```
ip -s -s neigh flush all
```

Host IP manuell setzen (Crash Cart)

Manuelle IP-Konfiguration pro AHV-Host, z. B. wenn der Host per Konsole/IPMI erreichbar ist aber (noch) kein Netzwerk hat

```
/usr/local/bin/network-crashcart
```