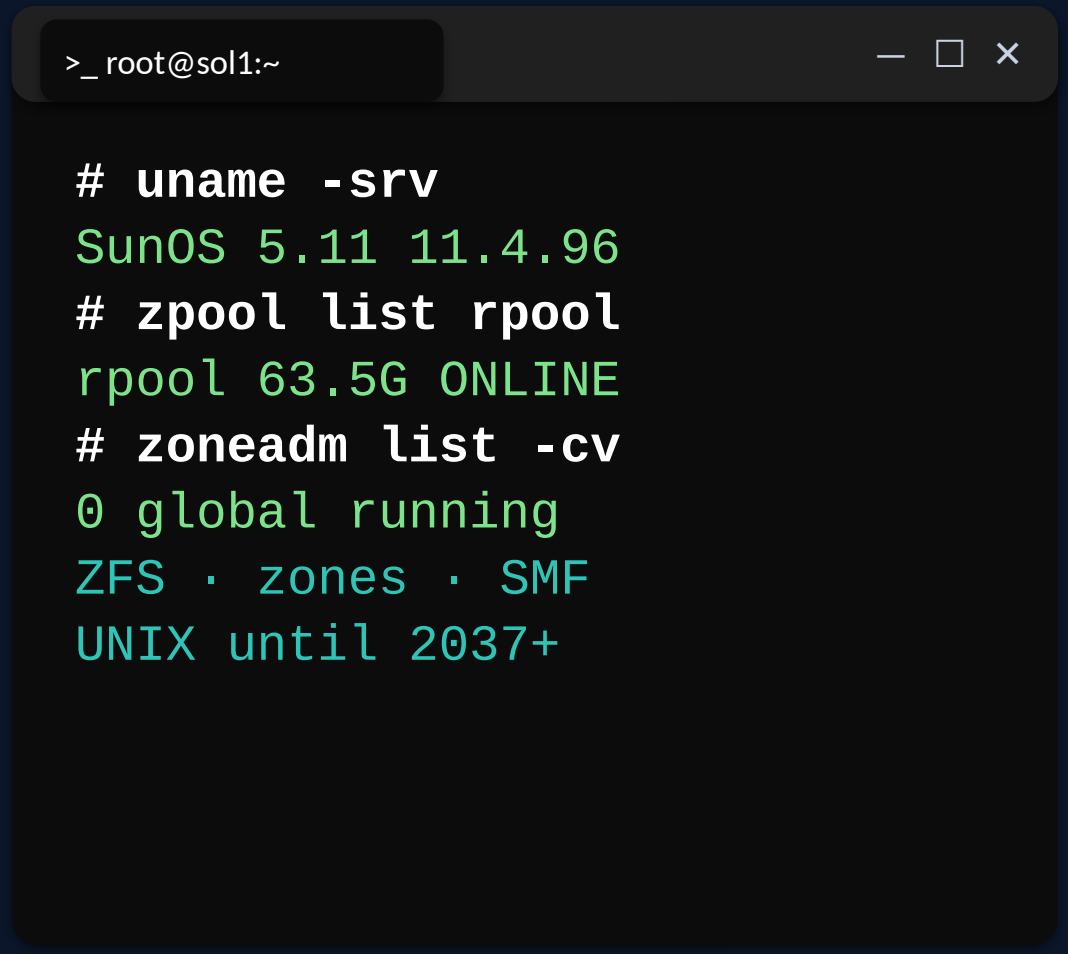


LESSON 5 OF 12

Oracle Solaris 11.4

ZFS · Zones · SMF · LDoms · OLVM

ZFS: Pools, Datasets, Snapshots & Replication

Full Course · Solaris 11.4 SRU96 · OVM (Nov 2026)

What You Will Learn in Lesson 5

- ZFS ideas: pooled storage, copy-on-write, checksums
- Pool layouts: mirror, raidz1/2/3, log, cache, spare
- zpool create, status, scrub, replace, monitor
- Datasets and properties: compression, quota, encryption
- Snapshots, rollback and clones
- zfs send / receive for backup and replication
- ZFS volumes, sharing (NFS/SMB) and the ARC

ZFS Ideas

Idea	What it means for you
Pooled storage	Disks form a pool; datasets share its space — no partitions to resize
Copy-on-write	Never overwrites live data — no fsck, always consistent
Checksums	Every block verified; bad copies healed from mirror/raidz
Snapshots	Instant, read-only, cost only changed blocks
Properties	Compression, quota, encryption, mountpoint per dataset
Boot environments	Root on ZFS makes lesson 3's safe updates possible

Pool Layouts

Layout	Survives / Notes
mirror (2–3 disks)	1–2 disk failures; best random I/O — rpool is a mirror
raidz1	1 failure per vdev; avoid with large disks
raidz2 / raidz3	2 / 3 failures per vdev — common for big data pools
log (SLOG)	Fast SSD for synchronous writes (NFS, databases) — mirror it
cache (L2ARC)	SSD read cache — after you add RAM
spare	Hot spare that FMA activates automatically
Rule	Redundancy lives in vdevs; a pool is lost if any vdev is lost

Creating and Caring for a Pool

```
root@sol1:~# format < /dev/null                # list disks: c2t1d0 c2t2d0 ...
root@sol1:~# zpool create datapool mirror c2t1d0 c2t2d0 mirror c2t3d0 c2t4d0
root@sol1:~# zpool add datapool spare c2t5d0
root@sol1:~# zpool status datapool
  pool: datapool
 state: ONLINE
  scan: none requested
root@sol1:~# zpool list ; zpool iostat -v datapool 5 2
root@sol1:~# zpool scrub datapool              # verify every block
root@sol1:~# zpool monitor -t scrub datapool 5  # progress (SRU 95 feature)
root@sol1:~# zpool attach rpool c1t0d0 c1t1d0   # turn rpool into a mirror
```

Replacing a Failed Disk

```
root@sol1:~# zpool status -x
  pool: datapool
 state: DEGRADED
status: One or more devices has been removed by the administrator.
        c2t3d0  REMOVED   0     0     0
        c2t5d0  INUSE (spare activated by FMA)
root@sol1:~# zpool replace datapool c2t3d0 c2t6d0
root@sol1:~# zpool monitor -t resilver datapool
root@sol1:~# zpool status datapool | grep state
 state: ONLINE
root@sol1:~# fmadm faulty      # confirm FMA closed the case
```

Datasets and Properties

```
root@sol1:~# zfs create -o compression=lz4 datapool/projects
root@sol1:~# zfs create -o quota=20G -o reservation=5G datapool/projects/web
root@sol1:~# zfs create -o encryption=on -o keysource=passphrase,prompt \
      datapool/secret
Enter passphrase for 'datapool/secret': ********
root@sol1:~# zfs set mountpoint=/srv/web datapool/projects/web
root@sol1:~# zfs get compression,compressratio,used,available datapool/projects
root@sol1:~# zfs set dedup=off datapool        # dedup needs lots of RAM
root@sol1:~# zfs set atime=off datapool/projects
root@sol1:~# zfs list -o name,used,avail,quota,mountpoint -r datapool
```

Snapshots, Rollback and Clones

```
root@sol1:~# zfs snapshot -r datapool/projects@before-upgrade
root@sol1:~# zfs list -t snapshot -o name,used,creation
root@sol1:~# ls /srv/web/.zfs/snapshot/before-upgrade/   # browse old files
root@sol1:~# cp /srv/web/.zfs/snapshot/before-upgrade/index.html /srv/web/
root@sol1:~# zfs rollback datapool/projects/web@before-upgrade
root@sol1:~# zfs clone datapool/projects/web@before-upgrade datapool/web-test
root@sol1:~# zfs diff datapool/projects/web@before-upgrade
root@sol1:~# zfs destroy datapool/projects/web@before-upgrade
# Automatic snapshots: svc:/system/filesystem/zfs/auto-snapshot (Time Slider)
```

Replication with send and receive

```
# Full copy to sol2 (192.168.56.52) over SSH
root@sol1:~# zfs snapshot datapool/projects@rep1
root@sol1:~# zfs send -R datapool/projects@rep1 | \
      ssh student@192.168.56.52 pfexec zfs receive -F backup/projects
# Next day: only the changes (incremental)
root@sol1:~# zfs snapshot datapool/projects@rep2
root@sol1:~# zfs send -R -i @rep1 datapool/projects@rep2 | \
      ssh student@192.168.56.52 pfexec zfs receive -F backup/projects
# Backup to a file
root@sol1:~# zfs send datapool/projects@rep2 | gzip > /backup/projects-rep2.zfs.gz
```

Volumes, Sharing and the ARC

Feature	Command
ZFS volume (zvol)	zfs create -V 20G datapool/vol1 → /dev/zvol/dsk/datapool/vol1
Swap / dump	rpool/swap and rpool/dump are zvols
NFS share	zfs set share.nfs=on datapool/projects
SMB share	zfs set share.smb=on datapool/projects (needs smb/server)
ARC (RAM cache)	kstat -p zfs:0:arcstats:size ; tune with user_reserve_hint_pct
Pool upgrade	zpool upgrade -v — only after all BEs can read the new version

Summary and Homework

- Pool = vdevs (mirror, raidz2…); redundancy lives in each vdev
- zpool create / status -x / scrub / replace / monitor
- Datasets with properties: compression, quota, encryption
- Snapshots, .zfs/snapshot, rollback, clone, diff
- zfs send | receive (full and -i incremental) for replication
- Homework: add 4 small VirtualBox disks, build two mirrors, snapshot, send to sol2 or a file

Thank You! 🙏

Free videos & PDFs: www.MyWebUniversity.com

Slides for this lesson: MyWebUniversity.com/presentations

YouTube: @MyWebUniversity · @MyWebUniversity-Dari

Wahidullah Lutfy — Founder and Director of IT of MyWebUniversity, and CS teacher

Book: The Direct Path to Linux Ubuntu (Amazon · Barnes & Noble)

Learn it, then pass it on.