
OpenVMS network integration

OpenVMS Bootcamp 2015

Integrating OpenVMS systems as part of a heterogeneous networked infrastructure

Colin Butcher CEng FBCS CITP

Technical director, XDelta Limited

www.xdelta.co.uk



Agenda

- Multiple protocols – not just TCPIP V4
- Network connectivity: VLAN tagging, LAN failover
- LDAP / Windows AD authentication
- File sharing: FTP / SFTP, NFS, CIFS
- Terminal access: TELNET, SSH
- Web server: CSWS, TOMCAT
- Monitoring tools: WBEM, SMH, SNMP
- Management tools: Availability Manager

Personal background

- Systems architect specialising in mission critical systems
- Engineering background
- Wide range of experience (satellite flight control, air traffic monitoring, finance data, healthcare, etc.)
- Started XDelta in 1996

XDelta – what we do

- Lead mission-critical systems projects
- Deliver world class services in demanding environments
- Strategic planning, technical leadership and project direction with clarity of vision and an eye for detail
- Systems engineering for availability and performance
- Ensure long term success through skills transfer

OpenVMS networking: multiple protocols

- TCPIP V4
- TCPIP V6
- DECnet Phase IV
- DECnet-Plus and DECnet over IP
- SCS (use SCACP)
- Clustering over IP
- AMDS (Availability Manager)
- LAT / MOP / Remote Console (Terminal Servers)
- LAD / LAST (Infoserver)

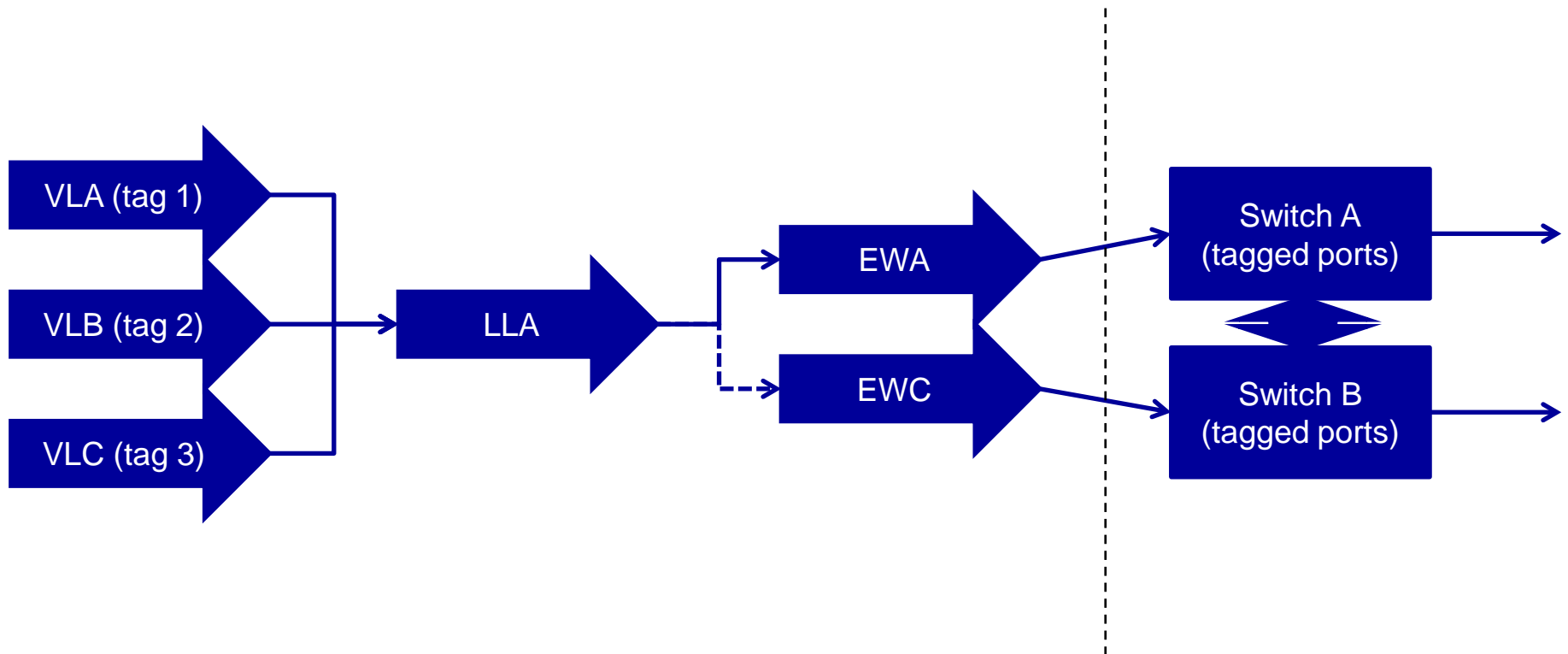
OpenVMS networking: multiple addresses

- DECnet naming is “per node”
- TCPIP addressing is “per interface”
- TCPIP allows multiple addresses per interface – and they can move between interfaces
- TCPIP - use “service addresses” that can be moved and enabled / disabled as needed
- TCPIP - use multiple subnets
- SCS, AMDS, LAT etc. are layer 2 non-routable

OpenVMS networking: jumbo frames

- Performance is better
- All intervening devices in the network infrastructure must have jumbo frame support enabled
- LAN_FLAGS bit 6 (64) enables jumbo frame support
- LANCP /JUMBO qualifier enables / disables jumbo frame support on a per device basis

OpenVMS networking: connectivity



OpenVMS networking: LAN failover

- Group NICS into LAN failover sets
- LLdriver presents “logical lan” devices

```
$ lncp define device lla/enable/failover=(ewa,ewc)
```

```
$ lncp define device llb/enable/failover=(ewb,ewd)
```

- One device in the failover set gives a layer of indirection

```
$ lncp define device lld/enable/failover=(ewd)
```

OpenVMS networking: VLAN tagging

- VLdriver applies 802.1Q tags to packets

```
$ lncp define device vla/tag='scs_tag'/vlan_device=lla
$ lncp define device vlb/tag='decnet_tag'/vlan_device=lla
$ lncp define device vlc/tag='userip_tag'/vlan_device=llb
$ lncp define device vld/tag='sysip_tag'/vlan_device=llc
$ lncp define device vle/tag='bkpip_tag'/vlan_device=lld
$ lncp define device vlf/tag='amds_tag'/vlan_device=lld
```

- Configure the switch ports to accept tagged packets

LDAP / Windows AD - authentication

- ACME
- LDAP
- Configure LDAP / Windows AD to allow access from OpenVMS LDAP authentication agent
- Swap loginout image etc. (requires reboot – security)
- Configure ACME server to use LDAP agent
- Configure LDAP agent on VMS to have access to LDAP directory service / Windows Active Directory
- Configure “mapping of principals” if needed
- Set “extauth” flag in UAF

LDAP / Windows AD – agent config (1)

```
sys$startup:LDAPACME$CONFIG-STD.INI
```

```
server = xd01w2k3.lab1.xdelta xd02w2k3.lab1.xdelta
```

```
port = 389
```

```
!
```

```
bind_dn = cn=svc_openvms_extauth,
```

```
        ou=OpenVMS ExtAuth,
```

```
        ou=XDelta Limited,
```

```
        dc=lab1,dc=xdelta
```

```
bind_password = <password_to_gain_access_to_ldap>
```

```
bind_timeout = 2
```

```
port_security = starttls
```

```
!ca_file = [directory]cacert.pem
```

```
!
```

LDAP / Windows AD – agent config (2)

```
<cont...> sys$startup:LDAPACME$CONFIG-STD.INI
```

```
password_type = active-directory
```

```
password_update = replace
```

```
!
```

```
base_dn = ou=Users,ou=XDelta Limited,dc=lab1,dc=xdelta
```

```
login_attribute = sAMAccountName
```

```
scope = sub
```

```
filter = objectclass=*
```

```
!
```

```
!mapping = <local / global> (or none if not defined)
```

```
!mapping_file=SYS$STARTUP:LDAP_LOCALUSER_DATABASE.TXT
```

```
!mapping_attribute = description
```

```
!mapping_target=VMSUsers.hp.com
```

LDAP / Windows AD - example

SYSTEM on RX2660 \$ set h 0

OpenVMS IA64 V8.4 on node RX2660

Username: butcher_ca

Password:

Last interactive login on Thursday, 16-APR-2015 19:46:32.29

**** Logon authenticated by LDAP ****

BUTCHER_CA on RX2660 \$

File sharing – CIFS (SAMBA) and NFS

- CIFS V1.2-ECO01 PS2_13 is latest from HP
- TCPIP V5.7-ECO05A is latest from HP
- NFS V3 client
- Pathworks (Advanced Server) to CIFS migration

CIFS (SAMBA)

- Install and do initial configuration of CIFS
- Edit SMB.CONF to add shares and set values for your site
- Connect to domain (net rpc join)
- Start CIFS
- SWAT (if used) needs local webserver (CSWS)

CIFS (SAMBA)

- File formats:
 - File format conversion may be needed:
\$ convert/fdl="record; format stream_lf" <infile> <outfile>
- Security:
 - File ownership and protection mapping

CIFS (SAMBA) – joining a domain

Must have SMB.CONF correctly set up with domain data (domain, security, servers, etc.) in [global] section of file

Join domain:

```
$ net rpc join --User=<domain admin> (not VMS username)
<password>
Joined domain LAB1_XDELTA
```

Check the join:

```
$ net rpc testjoin
Join to 'LAB1_XDELTA' is OK
```

Hint: SMB.CONF - require “strongkey = yes”

CIFS (SAMBA) – shares – SMB.CONF

Share specific sections in file: [sharename]

[xdelta]

comment = XDelta working area

path = xd_tree_toplevel:

writable = yes

read only = no

printable = no

browseable = yes

inherit owner = yes

vfs objects = varvfc

vms rms format = streamlf

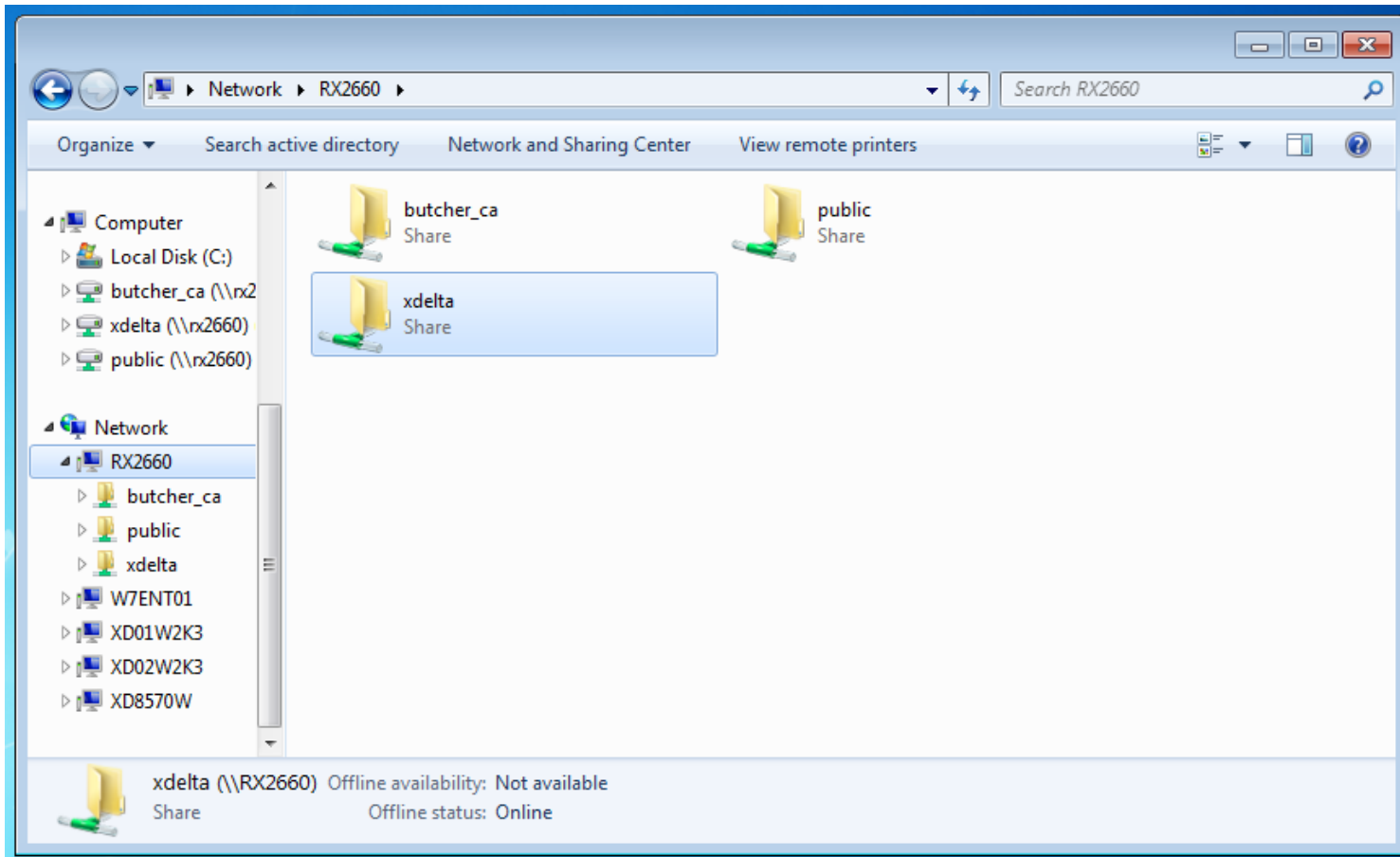
vms ods5 volume = yes

CIFS (SAMBA) – [homes] share

Special case: [homes] share

- Per-user “home” directories on server
- Needs domain access on both client and server
- WINBIND and username mapping
- See OpenVMS Technical Journal V14 for additional information

CIFS (SAMBA) – the view from Windows 7



CIFS - Alternative

- Use intermediate file server in Windows domain, so Windows clients work seamlessly
- Use NFS client on VMS to access file server
- Push files from VMS to file server (may need format conversion to stream_lf)
- Lab example, just to prove the point:
OpenVMS <NFS> Synology file server <SMB> Windows
Separate traffic using VLANs for security

NFS server and NFS client

- NFS V2 server
- NFS V2 and V3 client
- File format conversion may be needed:
\$ convert/fdl="record; format stream_lf" <infile> <outfile>
- TCPIP V5.7-ECO05A fixes a lot of NFS problems

NFS server

TCPIP V5.7-ECO04 added some useful features:

- Identifier based access:

```
$ define /system TCPIP$UCP_LOAD_RESOURCE_ID_PROXY 1
```

- Read-only exports

```
$ tcpip add export "/test" /host=* /options=(read_only)
```

- Improved control of OPCOM logging
- See release notes for details (now ECO05A)

NFS client

- Always use a separate copy of the XQP for NFS mounts, e.g.:

```
$ tcpip mount dnfsNNN: /host=<hostname / ip_address>/path="<path>" -  
  /uid=XXX/gid=YYY -  
  /structure=5/convert/system -  
  /processor=unique/acp=(dump,buffer_limit=1000000,page_file=500000) -  
  /transport=tcp/version=3 -  
  /sync=file_sync  
  /noadf -  
  <volume_name>
```

Terminal access

- SSH (and TELNET)
 - Terminal emulator products:
 - PuTTY (freeware, with SSH)
 - Attachmate reflection
 - Ericon Powerterm (version shipped with VMS does not have SSH)
 - Etc.
 - ILO on Integrity Servers
- Serial port access (useful for ILO setup etc.):
 - Console servers (SSH and TELNET)
 - DECservers (no SSH / SSL)

Web servers

- CSWS (Apache)
- WASD
- OSU (Ohio State University)

- CSWS used for SMH (systems management home page), SWAT (SAMBA configuration web pages), etc.

- CSWS has sub-kits for PERL, PHP and TOMCAT (Java)

Monitoring

- TCPIP V5.7 provides SNMP monitoring capability
- Tools such as PTRG (Paessler) will collect and display SNMP data graphically over time
- T4 can monitor and record traffic flows per network interface (or pseudo-interface)
- SCACP
- LANCP
- DECnet-Plus
- TCPIP (*nix like commands)

Availability Manager

- Uses AMDS protocol to interact with OpenVMS at driver level
- Can diagnose and bypass / fix a range of problems (with care!)
- New: VSI Availability Manager on 64bit platform

Future plans (VSI)

- New TCPIP stack with NFS V4, IPSEC, etc.
- Up-to-date SSL
- Up-to-date CIFS (SAMBA)
- Up-to-date CSWS (Apache), PHP, PERL, etc.
- Up-to-date JAVA and TOMCAT
- If you have a wish list, let us know!

OpenVMS network integration

OpenVMS Bootcamp 2015

Thank you for your participation

Colin Butcher CEng FBCS CITP

Technical director, XDelta Limited

www.xdelta.co.uk