Virtual Systems based training

Release 1

COLLABORATORS			
	TITLE : Virtual Systems based training		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY	Włodzimierz Wiszka	July 5, 2013	

REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME
1.00	20 Jul 2012	first release	ww
1.01	1 Aug 2012	chapter on classroom setup extended	ww
1.02	7 Aug 2012	a chapter on analyzing IP traffic added	ww
1.03	8 Aug 2012	remarks on using VMs added, corrections	ww
1.04	10 Sept 2012	flowchart on participants' WS IP setup added, corrections	ww
1.05	18 Sept 2012	references added, references grayed, minor corrections	ww
1.06	05 Jul 2013	english web addresses	ww

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Note:
This document concerns any IT training setup and is kept as universal as possible. All referred details are used for demonstration purposes only and are meant to be treated as examples. In particular ALDP stands for a non-existent Application Layer Diameter Protocol and denotes any TCP/IP based application layer protocol. An assumption has been made that all target systems are RedHat (RHEL) installations with production applications deployed. Please refer to the author for further clarifications.

Chapter 2

Training environment

2.1 Prerequisities

- **Location**
  - on-site or local
- **Number of participants**
  - 6-12, up to 3 per 1 target system (VM server)
- **Participants' workstations**
  - Windows XP/Windows 7 with office software
  - web browser, ssh client (eg. putty) installed
  - ALDP tools installed (for ALDP deployment)
  - network analyzer, e.g. wireshark, with ALDP protocol dictionary (for ALDP deployment)
  - controll over firewall
- **Participants' knowledge required:**
  - general Linux administration
  - RHEL network administration
  - IP networking (ideally on the level of CCNA)
- **Materials (soft copies):**
  - customer documentation
- **Materials (hard copies)**
  - exercises binder

2.2 General remarks on using VMs

2.2.1 Hosts's Operating System

While in the past some OS/virtualization combinations were known to introduce instabilities (e.g. with bridged networking), today to a high extent this is a matter of personal choice. Having said that, there are still features worth considering:

- Running additional services on the host (NTP, FTP)
- Memory management (e.g. Windows XP can only effectively manage 3GB)
- Dual display management for working with a projector
- Multiple workspaces allowing spanning information accross a bigger area

2.2.2 Memory

Make sure all declared guests' memory is mapped to host's physical memory. In case of memory limits on the host use fast (SSD) disks.

2.2.3 Filesystems

Use expandable virtual disks, so that not to eat up all the disk space at once. Please note they grow in size with time, even if effectively same disk space is being used. Some virtualization technologies offer tools to shrink virtual images (e.g. How To Shrink VMware Virtual Disk Files (.vmdk) .

Virtual disk images rarely fall below 4GB of size. Therefore either decide to split them into up to 2GB chunks or choose your filesystem accordingly.

2.2.4 Timekeeping practices

Depending on virtualization technology used (VMware/VirtualBox), host OS (Linux vs. Windows) and kernel, guest OS, kernel and distribution, additional kernel parameters may/may not be required. This can be checked by comparing system time with an external reference, e.g. a stopwatch:

```
$ sleep 10
```

If you notice significan't difference between actual and expected elapsed time of this command's execution you should take relevant measures. It is usually sufficient to add `divider=10` as a kernel parameter, e.g.:

```
# cat /boot/grub/grub.conf
          ...
          kernel /vmlinuz-2.6.9-89.ELsmp ro root=/dev/lvm01/root.vol console=tty0 divider=10
          ...
```

Please refer to your virtualization technology supplier for further instructions (e.g. VMware KB: Timekeeping best practices for Linux guests).

Please note that if the guest is running on a laptop, changing host's CPU frequency may affect guest's virtual clock. Therefore it is always recommended to disable any CPU speed scailing and/or power management processes and enforce highest CPU frequency on the host (it may be helpful to have laptop plugged into a power outlet).

Due to possible difficulties in timekeeping it is recommended to run an NTP server on the host, synchronizing all guests clocks.

2.3 Classroom network setup

Directions on how to setup a classroom network for a VM-based training

- Target systems are run as VMs with bridged networking, exposing interfaces to the training LAN.
- Participants connect with client tools to target systems. This may require additional software installation on their workstations.
- In case TCP connections need to be made from target systems to tools running on participants' workstations, participants may be required to switch off all their firewalls or make relevant exceptions.
- Optionally additional (client) VMs may be run so that participants can use already prepared environment. In that case, in terms of TCP, participants only make outgoing RDP connections to these client VMs, and from there they access target systems, therefore no workstation's firewall adjustment is needed
- The model assumes participants are able to change their workstations' IP configuration or they get it from DHCP server. If this is not the case the IP layout may be adapted. In that case, however, rather use secondary network interfaces on target systems (server VMs) than change existing IP configuration (primary interfaces), as deployed applications may depend on that.
- When using 2 network interfaces the general rule is the last started takes over DNS and routing settings. In this case you should first start wired one for the training LAN and then wireless for accessing Internet, corporate network and alike.

Refer to flowchart on Figure 2.2 for further details on setting up participant's workstations.

parameter	**default (initial) value**
user root password	qwe123
user aldpadmin password	aldpadmin
web gui listening port	9000
web gui username/password	admin/admin

Table 2.1: Services configuration (same on all servers)

Private training network layout

- 2-6 virtual RHEL systems with ALDP services deployed
- 1 home router with DHCP, DNS forwarding
- 1 8-port switch

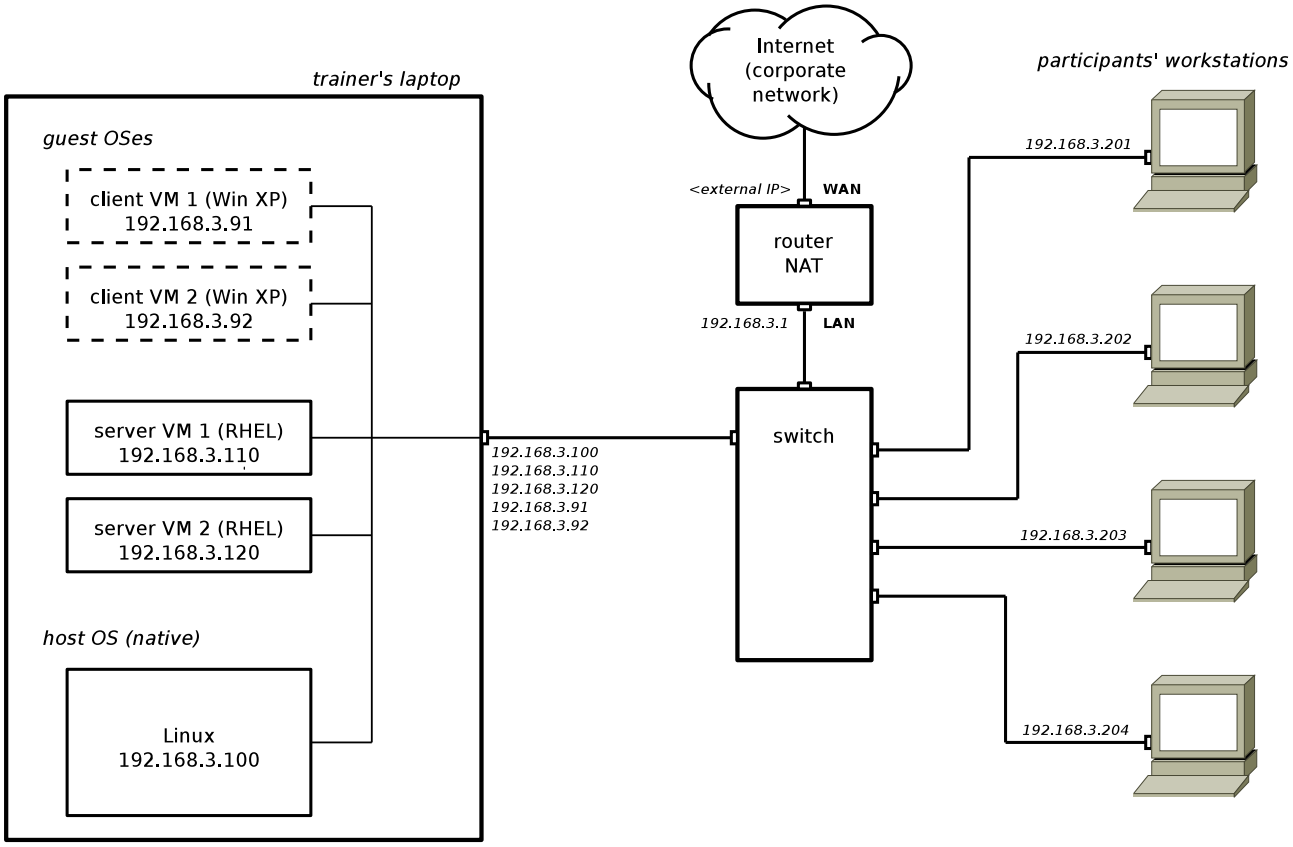

Figure 2.1: Example classroom training network layout

The flowchart below shows typical decisions to be made when setting up a training network. The requirement is participants' workstations (WSs) can access both training LAN and company's corporate network, so that e.g. to check emails, browse the Internet, access production systems and alike.

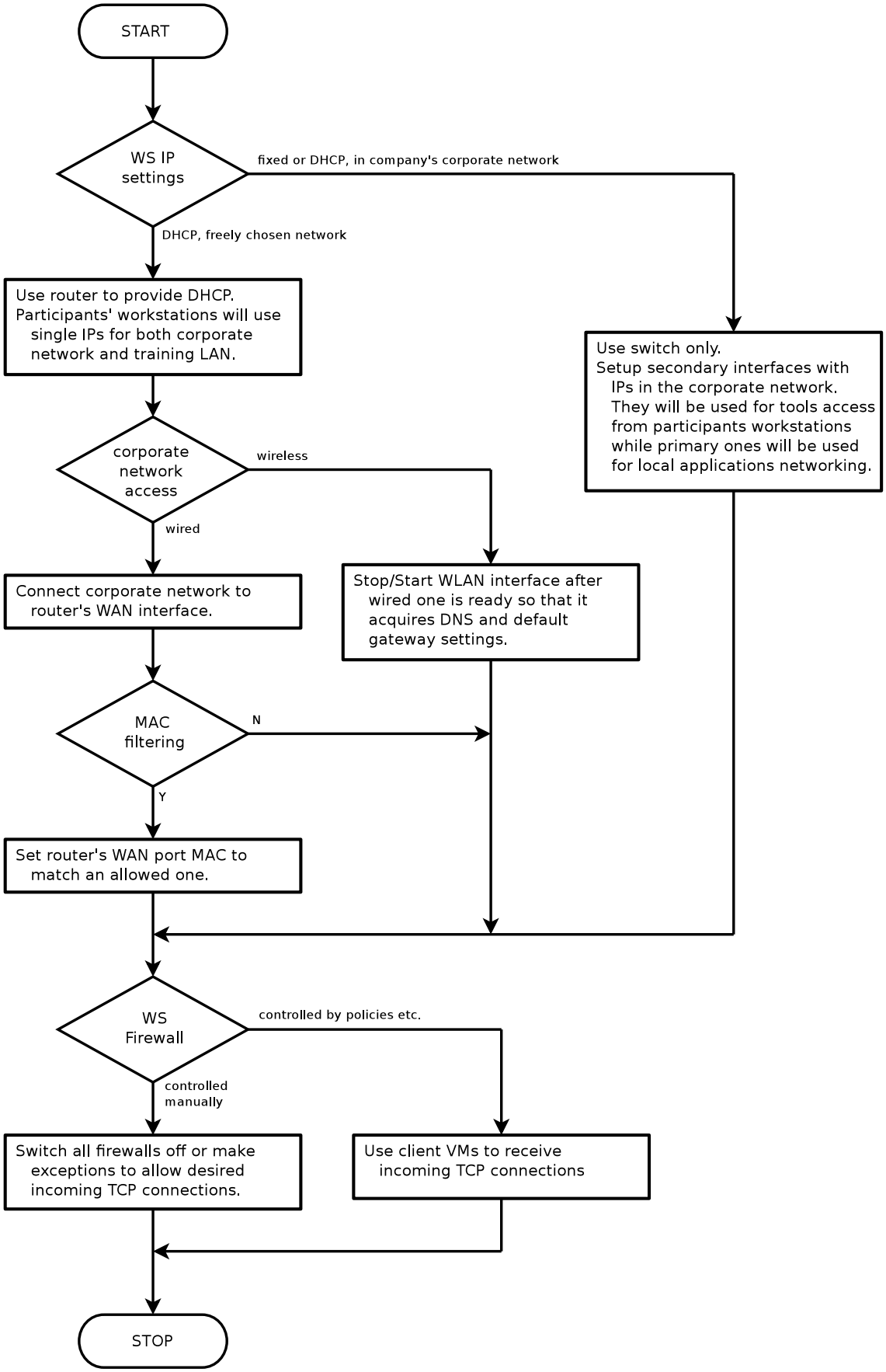

Figure 2.2: Common factors determining training network setup.

Chapter 3

Acquiring and analyzing IP traffic

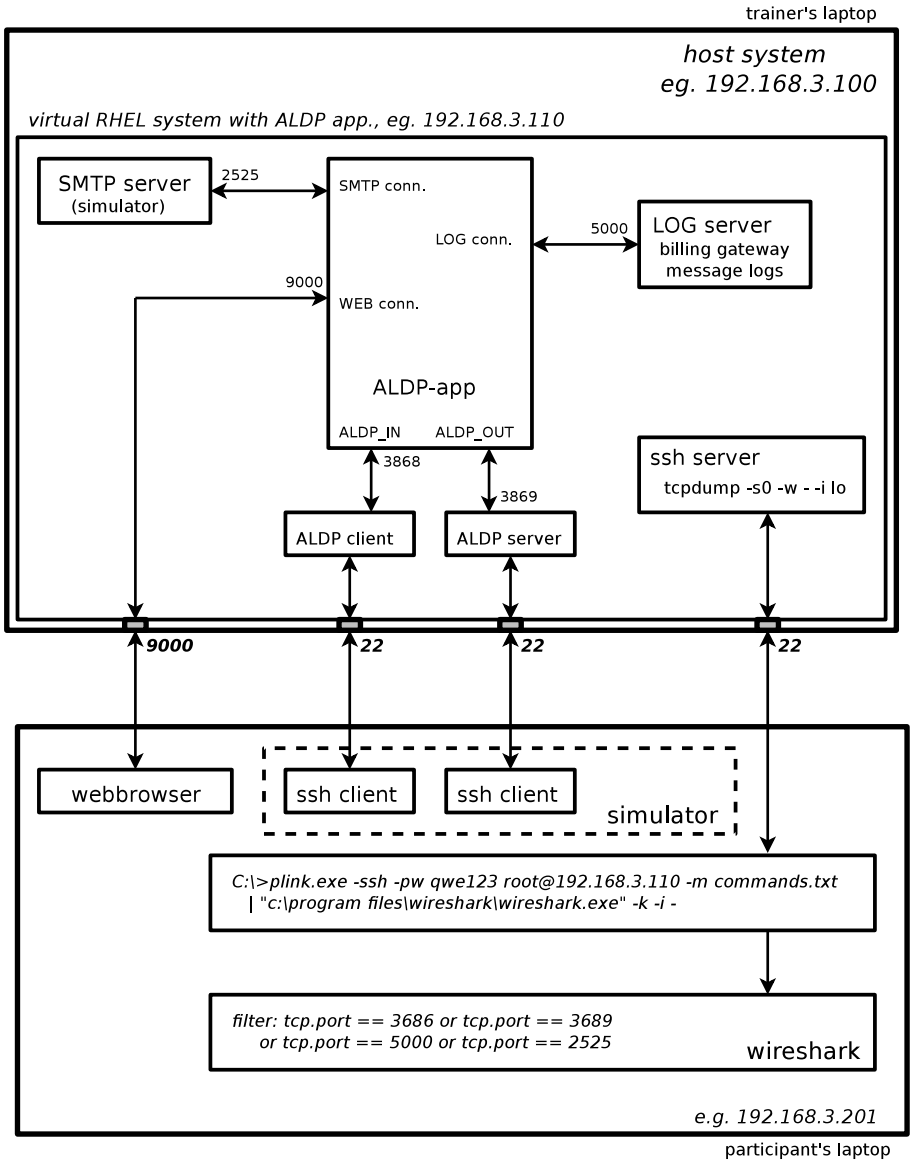

Figure 3.1: Example process/network model for ALDP call flow exercises

Testing environment

- Both ALDP server and client are being started locally on a target system, using localhost interface. In order to watch the corresponding traffic you need either to run `tcpdump` on target system, saving packets to a file, download it to your laptop and analyze with `wireshark`, or run ssh client on your laptop, invoking remote `tcpdump` session and redirecting it's output to the locally started wireshark. Both methods are described below.
- In order for the `wireshark` to be able to analyze ALDP packets a special ALDP dictionary is needed, which has to be placed in `wireshark`'s "Diameter" directory. Be advised to make a copy of the standard diameter dictionary first.
- Most of the exercises require multiple terminal (ssh) sessions, so that to run ALDP server and client and observe various application activities simultanuously. These are referred to as 1#, 2# etc. in the screen shots.

3.1 ALDP application configuration files

Check IP/port numbers in the ALDP connectors settings in the `aldp.properties` file:

```
# su - aldpadmin
> vi /opt/aldp/etc/aldp.properties
          ...
          ## ALDP Management configuration
          management.period=40
          ...
          ## ---- ALDP server connector ----
          aldp.server.service.id=ALDP_in
          # listen on all network interfaces
          aldp.server.hostname=0.0.0.0
          aldp.server.port=3868
          aldp.server.protocol=ALDP
          #-----------------------
          # [WW] ALDP --> ALDP
          aldp.server.bearer=SMS
          aldp.server.version=1
          # [WW] ALDP --> MM7
          # aldp.server.bearer=MMS
          # aldp.server.version=2
          #-----------------------
          aldp.server.component.id=ALDP_IN_1
          aldp.server.management.enable=true
          ...
          ## ---- ALDP client connector ----
          aldp.client.service.id=ALDP_out
          aldp.client.hostname=127.0.0.1
          aldp.client.port=3869
          aldp.client.protocol=ALDP
          aldp.client.bearer=SMS
          aldp.client.version=1
          aldp.client.component.id=ALDP_OUT_1
          aldp.client.management.enable=true
          ...
```

Check ALDP connector presence in the `msg_endpoints.xml` file:

```
# su - aldpadmin
> less /opt/aldp/etc/msg_endpoints.xml

          ...
          <MsgEndpoints>
            <endpoint id="ALDP_OUT_1">
```

```
            <name>Message_Send_ALDP_OUT_1</name>
          </endpoint>
        <MsgEndpoints>
        ...
```

3.2 Remote packet sniffing

3.2.1 Sniffing to a file and post-processing

- Start `tcpdump`, observe SYN packets, when ALDP app. tries to connect.

```
1#  tcpdump -i lo port 3868 or port 3869 or port 5000 or port 2525 -w packets.pcap -s0
```

- Run call flows
- Transfer `packets.pcap` with an SCP client (e.g. WinSCP) to your workstation
- Analyze packets, following directions from Section 3.2.2

3.2.2 Sniffing live traffic

Acquire live traffic by redirecting remotely started `tcpdump`'s output to locally started `wireshark`.

- Download plink.exe form putty download site
- Create a file `commands.txt` with the following content

```
      tcpdump -s0 -w - -i lo not port 22
```

- At the Windows command prompt run the following command

```
C:\> plink.exe -ssh -pw qwe123 root@192.168.3.110 -m commands.txt | "c:\program files\ ↩
    wireshark\wireshark.exe" -k -i -
```

- Download ALDP protocol dictionary, place it in relevant wireshark's diameter directory
- Make sure both 3868 and 3869 ports are specified for Diameter protocol.
- Set wireshark to filter ALDP, LOG and SMTP protocols: **`tcp.port == 3868 or tcp.port == 3869 or tcp.port == 2525 or tcp.port == 5000`**.
- Make wireshark interpret port 2525 activity as SMTP.
- Run call flows and observe traffic. You can colorize ALDP commands: CER-CEA, DWR-DWA, NPR-NPA, MPR-MPA.

If on Linux you can simply run

```
# ssh root@192.168.3.110 "tcpdump -w - -s0 not port 22" | wireshark -k -i -
```

3.3 ALDP connect/reconnect

Start several terminal (putty) windows so that to watch multiple processes. They will be refered to as 1#, 2# etc. in the screen shots below.

Start ALDP server, observe how the application connects.


```
2# cd /opt/test/bin/
2# cat ./aldp.server.sh
2# less ../etc/aldp.server.xml

          ...
          <define entity="channel"
              name="channel-1"
              protocol="aldp-v1"
              transport="trans-1"
              <!-- Edit this to the real IP and port used by your server -->
              open-args="mode=server;source=127.0.0.1:3869">
          </define>
          ...

2# ./aldp.server.sh
3# netstat -an|grep 3869
```

Stop ALDP server, observe how the application disconnects. Then observe how it is trying to reconnect.

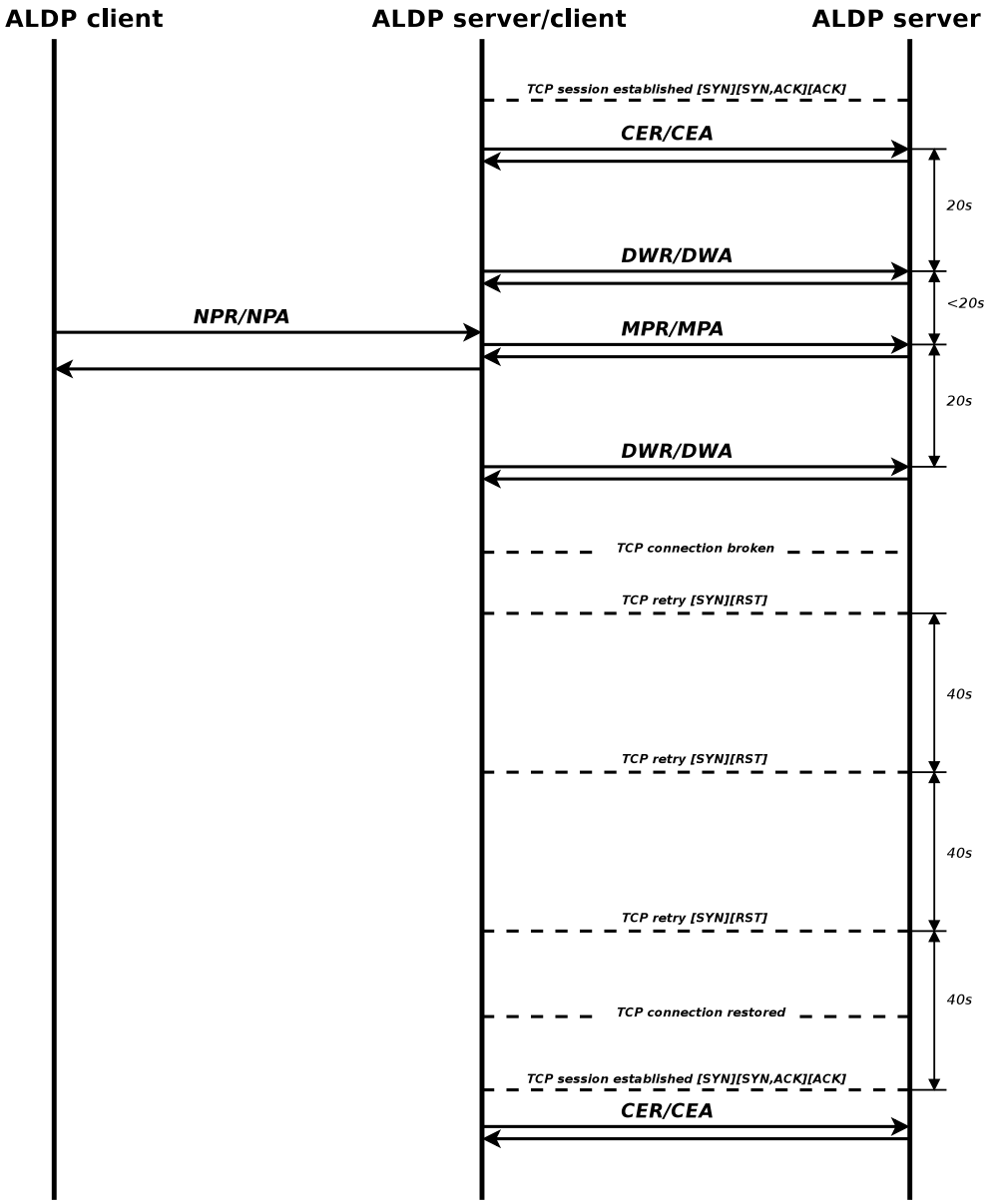

Figure 3.2: Example ALDP application connect/reconnect call flow