

title DLS cockpit
subject ATN decoder - Installation manual

version/lang. 15.0 E
status Released
date of issue 2026-01-15
effective date
contact/title Daniel Schmutz / Operations COM GVA
doc. owner/title Luc Chevalley / Senior Project Manager
location only the current electronic version on the intranet is valid
file 53127859
pages 28
classification Restricted
for action
for information
template ID template ID / 1.0
abstract This is the installation manual of the ATN decoder, a component of the DLS cockpit solution.

legal notice The entire content of this publication is protected by copyright. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any electronic, mechanical, photocopying and recording means or otherwise, without the prior written permission of skyguide.

Table of contents	Page
1 History of changes	4
2 Solution overview	5
2.1 Product release	5
2.2 Principle	5
2.3 High level schema	5
2.4 Pre-requisites	6
3 Hardware and software	7
3.1 Hardware	7
3.2 Software	7
3.2.1 Operating System	7
3.2.2 ATN decoder software	7
3.2.3 Folder structure	8
3.2.4 File transfer	8
3.2.5 Hardening	8
3.2.6 Data file deletion	9
3.2.7 Log files rotation and deletion	9
4 Customization	10
4.1 Purpose	10
4.2 Items to be changed	10
4.2.1 Operating system	10
4.2.1.1 hostname file	10
4.2.1.2 hosts file	10
4.2.1.3 motd file	10
4.2.1.4 resolv.conf file	10
4.2.2 General parameter file	11
4.2.3 Filebeat configuration	12
4.2.4 NTP time synchronization (CentOS 7.2)	13
4.2.5 Chrony time synchronization (RHEL 8)	14
4.3 Checklist for ATN decoder customization	15
4.4 Physical connections	16
5 ATN decoder software installation and configuration	17
5.1 Installation	17
5.2 Cronjob	18
5.3 Parallel operations	20
5.4 Output file	20
6 System check	23
6.1.1 Filebeat	23
6.1.2 NTP (CentOS 7.2)	23
6.1.3 Chronyd (RHEL 8)	24
7 Useful commands	26
8 References	27
8.1 Software releases	27

9 Abbreviations, Terms and Definitions	27
9.1 Abbreviations	27
9.2 Terms and Definitions	28

1 History of changes

Changes				
Version	Status	Date of issue	Author	Details
0.1	Draft	2020-06-30	L. Chevalley	Document creation
1.0	Released	2020-07-01	L. Chevalley	Adaptations. Released version.
1.1	Proposed	2020-12-07	M. Rebmeister	Add package location
1.2	Proposed	2021-05-03	L. Chevalley	Alignment with VDL2 decoder documentation. Proposed version.
2.0	Released	2021-06-09	D. Schmutz, L. Chevalley	Final review. Released version.
3.0	Released	2021-09-07	L. Chevalley	Minor corrections following review. Released version.
4.0	Released	2021-09-28	L. Chevalley	Made separate section for software upgrade. Released version.
5.0	Released	2022-11-17	L. Chevalley	Amelioration of section 5.2 and addition of automatic NTP service launch through cronjob.
6.0	Released	2022-12-19	L. Chevalley	Cronjob example correction in section 5.2
7.0	Released	2023-03-20	L. Chevalley	Added remark under section 5.1 and adapted section 8.1.
8.0	Released	2023-03-30	L. Chevalley	Added references to ADS-C and IDRP which can now be decoded with PDEC. Added details for NTP synchronization under RHEL 8.
9.0	Released	2023-05-23	L. Chevalley	New PDEC C3p1 release. Added some indications under 5.1. Added some information in case of implementation on Ubuntu.
10.0	Released	2023-07-04	L. Chevalley	C3 release can be installed on Centos 7.2 with dedicated SW release provided by Airtel-ATN.
11.0	Released	2023-09-19	L. Chevalley	Clarified how to fix the PLM_SNIFFED_ADDRESS and PLM_SNIFFED_INTERFACE variables. New C4 release.
12.0	Released	2024-07-27	L. Chevalley	Renamed CPDLC cockpit into DLS cockpit and CPDLC decoder into ATN decoder.
13.0	Released	2024-08-22	L. Chevalley	Corrected the commands for filebeat tests under 6.1.1.
14.0	Released	2024-12-05	L. Chevalley	Added the implementation of the skyguide atsu.csv in procedure of section 5.1
15.0	Released	2026-01-15	L. Chevalley	Full review and corrections aligned with product release 3.6

2 Solution overview

2.1 Product release

This document describes the installation and maintenance procedures for the ATN decoder. The document is valid for DLS cockpit product release 3.6 and later.

2.2 Principle

The ATN decoder is part of the DLS cockpit product provided by skyguide and is powered by the AIRTEL-ATN PDEC (Protocol Decoder) product. The product release of a specific ATN decoder can be found as filename of the following empty file:

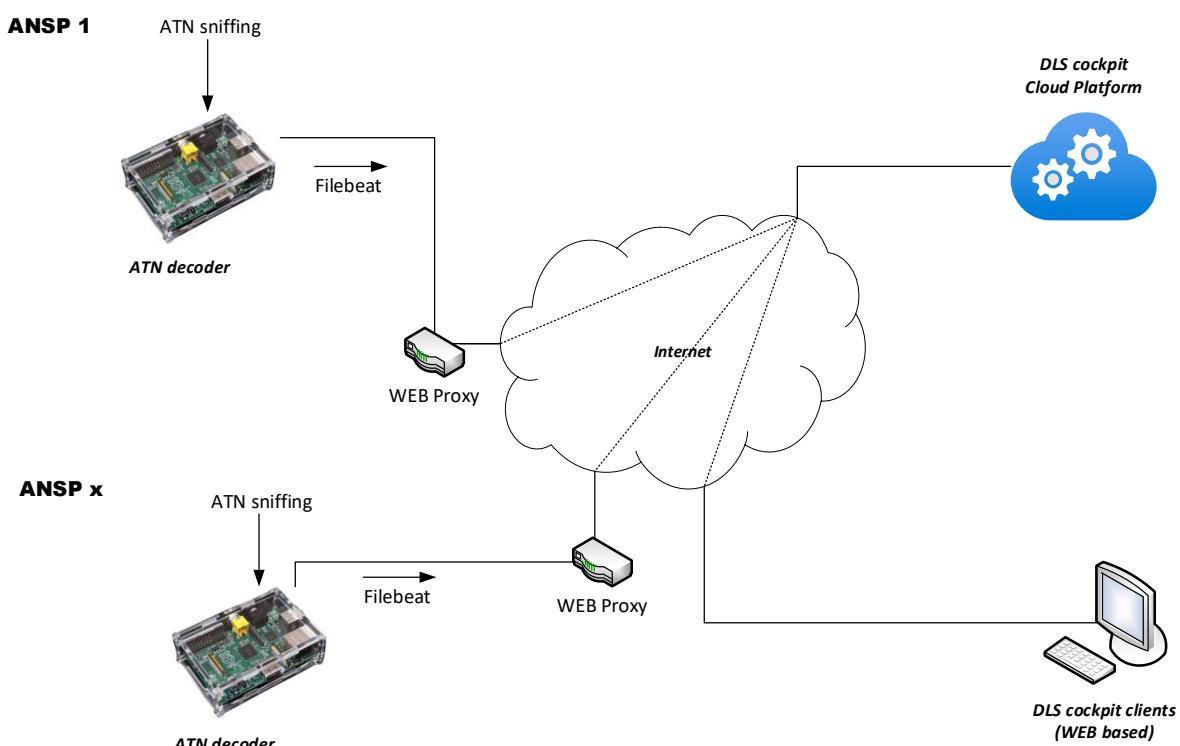
/opt/cpdlc/config/ATN-decoder_<n.m> (n.m = product release number as shown in section 2.1)

The ATN decoder is used to sniff ATN traffic and record CM, ADS-C, CPDLC, TP4 and IDRP data flows. The data is decoded and stored into local files and sent over the network to the Cockpit Cloud Platform.

The data can then be retrieved and displayed using a web browser.

2.3 High level schema

The following schema, gives a high level view of the solution. Technical details of the implementation, like internet access, firewall rules and so on, should be made available and maintained by each ANSP.



2.4 Pre-requisites

In order to have the solution working, the following prerequisites have to be satisfied:

On the client (ANSP/ACSP) side:

- Network connection to the Internet available
- Fixed IP address to integrate the station into the client LAN, also used for optional SSL certificates generation
- Access to NTP and DNS servers
- Access to the DLS cockpit cloud server granted over the internet (firewall rules, web proxy, etc.)
- Ethernet port providing a copy of the ATN traffic to be sniffed
- Uninterrupted power supply

On the DLS cockpit cloud server side:

- Logstash pipelines defined
- Kibana dashboards set up
- Filebeat access for data transfer
- SSL certificate generated for the station's specific IP address, if requested

3 Hardware and software

3.1 Hardware

Each station is based on a mini-PC or a server (e.g. Qotom mini-PC).

Description	Illustration
1x QOTOM Mini Pc with multi-Gigabit Ethernet ports 4GB of RAM is enough to run the ATN decoder 64GB SSD is enough as well. NB: Exact configuration can vary.	
Power supply	
Several cables	– 1-2x LAN

For the first boot, monitor and keyboard are required to configure the station. Afterwards the station can run headless and can be accessed over the network using SSH secured communication.

3.2 Software

3.2.1 Operating System

In order to run the ATN decoder and especially the PDEC software of Airtel-ATN, CentOS v7.2 needs to be installed or RHEL 8.4 or 8.6. An implementation was also done on Ubuntu LTS 22 but it requires some specific settings.

3.2.2 ATN decoder software

The system relies on the PDEC software of Airtel-ATN, which is based on the sniffing of ATN data through the tcpdump software. The PDEC package is provided by skyguide to the customers.

The PDEC software is recording the traffic heard at the defined Ethernet interface. CM, ADS-C, CPDLC, TP4 and IDRP data are captured and written in real time to the output file at

`/usr/PDEC/livemonitoring/logs/YYYYMMDD_live_capture.log`

in undecoded text format.

These logfiles are processed at regular intervals (defined in the crontab) through the automatic launching of the "decode_capture.sh" script. The decoded files contain only the new decoded data and are stored at

`/usr/PDEC/livemonitoring/data/YYYYMMDDhhmm_pdus.csv`

NB: The regularity set at the crontab level defines the refresh rate of the DLS cockpit.

The ATN decoder health is controlled using crontab and a specific script (`check_pdec_tcpdump.sh`). After reboot and every minute, the system verifies that the tcpdump runs properly and if not, the `start_capture.sh` is started.

`/usr/PDEC/livemonitoring/cron/start_capture.sh`

The latest ATN decoder release is 3.6 running the PDEC software C5p2.

3.2.3 Folder structure

The following folder structure has been set up:

Folder	Type of files
/usr/PDEC/livemonitoring/cron	Application scripts
/usr/PDEC/Airtel_PDEC_EXE_<version>_<OS>/bin	Application binaries
/usr/PDEC/Airtel_PDEC_EXE_<version>_<OS>/data	Application configuration files
/usr/PDEC/livemonitoring/data	Minute files (decoded outputs)
/usr/PDEC/livemonitoring/logs	Raw data files (undecoded)
/opt/cpdlc/certs	SSL certificates location, if used
/opt/cpdlc/config	Configuration files
/opt/cpdlc/scripts	Scripts
/opt/cpdlc/systemd	System units
/opt/data/xcpdlc	Daily files (decoded)
/opt/data/xcpdlc_archive	Archived daily files (decoded)
/var/log/ansplogs	log files

Configuration files are described later in this document (see 4).

3.2.4 File transfer

The decoded data (containing only the new data since the last decoding) are first appended to a daily file located at

/opt/data/xcpdlc

The content of the daily file is transferred to the DLS cockpit cloud server using "Elasticsearch Filebeat" utility which allows a nearly real time and optimized data flow. Every time new data are appended to the daily file, filebeat identifies the new lines and forwards them to logstash in the DLS cockpit cluster.

Filebeat is started automatically at system boot time and the file transfer can be encrypted using SSL certificates. In this case, the certificates will be located in the following folder and owned by the "root" user in order to be accessed by filebeat:

/opt/cpdlc/certs

Filebeat configuration files are located in the following folder and make use of OS variables, which are defined in the `general_parameter_file` to customize the run options.

/opt/cpdlc/config

Filebeat start-up is controlled using the "systemctl" commands. Use the following command to enable the decoder to start-up automatically at boot time.

```
# sudo systemctl enable <unitname>
```

Filebeat is enabled/started by default. The unit name is often filebeat-xcpdlc.

3.2.5 Hardening

All Linux command have to be launched using the "sudo" prefix as the "admin" user is not allowed to use them or to access in write mode to the configuration files.

The password for the admin user should be changed according the existing security rules.

3.2.6 Data file deletion

The xcpdlc daily files are automatically archived and deleted by a Linux cron job according to the values defined in the `general_parameter_file`.

3.2.7 Log files rotation and deletion

All log files are written to the following folder.

`/var/log/ansplogs`

Log files are rotated every day and kept over five generations. File rotation is controlled using the following configuration file:

`/opt/cpdlc/config/ansplogs`

4 Customization

4.1 Purpose

This chapter explains how to set up a new station by customizing a standard distribution.

4.2 Items to be changed

All configuration files are located in the following directories:

- /usr/PDEC/Airtel_PDEC_EXE_<version>_<OS>/data
- /opt/cpdlc/config
- /opt/cpdlc/systemd
- /etc/systemd/system

Changes to be made are highlighted in the following paragraphs.

After all the operating system changes have been done, reboot the ATN decoder in order to activate them prior to connect the LAN cables.

4.2.1 Operating system

At operating system level, the following configuration file has to be adapted

4.2.1.1 hostname file

The hostname file can be found here:

/opt/cpdlc/config/hostname

Put the hostname of your machine in this file.

4.2.1.2 hosts file

Not used

4.2.1.3 motd file

The login banner is configured in the following file:

/opt/cpdlc/config/motd

```
*****  
DLS COCKPIT - ASTERIX and/or ATN decoder  
Unauthorized access is forbidden.  
*****
```

4.2.1.4 resolv.conf file

In order to resolve the IP address of the DLS cockpit cloud server from the configured node name, the DNS service can be configured according to the infrastructure used.

Configuration is done by adding the DNS server IP address into the following configuration file:

/opt/cpdlc/config/resolv.conf

The file should contain the DNS servers addresses:

```
nameserver <IP1>  
nameserver <IP2>
```

```
nameserver <IPn>
```

4.2.2 General parameter file

Most of the parameters have been centralized into a single file in order to keep the overview and simplify the application management.

For now, the following applications make use of this file:

- filebeat-xcpdlc
- scripts for archiving and purging the data and log files

This parameter file is the following

```
/opt/cpdlc/config/general_parameter_file
```

This file defines several options for the applications as well as some OS variables used to standardize some start up scripts and reduce the files to be modified to customize the system.

Parameters for each application are grouped into a single section and should not be modified unless requested by Skyguide. Variables that should or could be used by another script have to be exported to the operating system. To use them later in a script, the "general_parameter_file" has to be "sourced", using the following line at the beginning of the script:

```
. /opt/cpdlc/config/general_parameter_file
```

Here is an example of the general_parameter_file

```
#!/bin/bash
#
# general_paramter_file
#
# This file contain all the needed parameters and variables needed to run
# the decoder.
#
# Each section defines the variables for a specific component.
#
# This file should be kept as is and not be modified, except if asked by
# skyguide.
#
#####
#
# Options for Filebeat service
XCPDLC_OPTIONS="-e -c /opt/cpdlc/config/filebeat-xcpdlc.yml"
#
# Other variables
XCPDLC_ORIGIN=PDEC2
XCPDLC_NAME=XCPDLC-NAG
XCPDLC_IGNORE_OLDER=240h
XCPDLC_LOADBALANCE=false
XCPDLC_HOSTS="185.74.71.126:11531"
XCPDLC_PROXYURL=socks5://172.27.135.30:3128
XCPDLC_PATTERN='*_pdus.csv'
XCPDLC_DIR=/usr/PDEC/livemonitoring/data
XCPDLC_ARCHIVEDIR=/opt/data/xcpdlc_archive
XCPDLC_TRANSFERDIR=/opt/data/xcpdlc_transfer
XCPDLC_LOGRETENTION=20
XCPDLC_ARCHIVERETENTION=730
#####
#
```

4.2.3 Filebeat configuration

Data files are shipped using the Elasticsearch Filebeat utility. There is some configuration to be adapted in order to have the file transfer working correctly and to receive the data files at the right place.

Filebeat configuration is done using the following file:

/opt/cpdlc/config/filebeat-xcpdlc.yml

The following example shows the usage of OS variables to customize the configuration:

```
#===== Registry =====
filebeat.registry:
    path: xcpdlc
#===== Filebeat modules =====
filebeat.config:
    modules:
        path: /etc/filebeat/modules.d/*.yml
        reload.enabled: false
#===== Filebeat inputs =====
filebeat.inputs:
    - type: log
        enabled: true
        name: ${XCPDLC_NAME}
        fields: {Origin: "${XCPDLC_ORIGIN}"}
        fields_under_root: true
        paths: ["${XCPDLC_DIR}/${XCPDLC_PATTERN}"]
        ignore_older: ${XCPDLC_IGNORE_OLDER}
        multiline.match: after
        multiline.negate: true
        multiline.pattern: ^[0-9]{1,2}\/[0-9]{1,2}\/[0-9]{4}
        multiline.max_lines: 2000
        clean_removed: true
        close_inactive: 2s
        scan_frequency: 2s
        scan.sort: filename
        scan.order: asc
        harvester_limit: 1
#===== Filebeat output =====
output.logstash:
    enabled: true
    worker: 1
    loadbalance: ${XCPDLC_LOADBALANCE}
    hosts: '${XCPDLC_HOSTS}'
    proxy_url: ${XCPDLC_PROXYURL}
    ssl.enabled: true
    ssl.verification_mode: full
    ssl.certificateAuthorities: '${SSL_CERTIFICATE_AUTHORITIES}'
#===== General =====
path:
    config: /etc/filebeat
    data: '${XCPDLC_FILEBEAT_DATAPATH}'
    home: /usr/share/filebeat
    logs: /var/log/filebeat
#===== Processors =====
processors:
    - drop_fields:
        fields: ["input", "beat", "log", "host", "prospector", "agent"]
#===== Kibana =====
setup:
    kibana: null
```

Normally, there is no need to change anything in that file, as the specific values are given by OS variables.

Certificates usage is not a “must have” and can be implemented on demand. If requested, copy the SSL certificates received from Skyguide into the following folder using the “root” account.

/opt/cpdlc/certs

If certificates are used, the filebeat configuration file has to be adapted (output/logstash section).

```
output.logstash:
  ssl.certificateAuthorities: ["/etc/ca.crt"]
  ssl.certificate: "/etc/client.crt"
  ssl.key: "/etc/client.key"
```

(Example taken from the filebeat documentation)

Socks proxy has to be configured according to the network implementation if requested or omitted if there is no proxy to use between the ATN decoder and the elasticsearch servers.

For all Filebeat flows, a separate instance should be set up. Each instance controlled by the `systemctl` command and configured using a specific `*.service` file located in the `/etc/systemd/system` folder.

4.2.4 NTP time synchronization (CentOS 7.2)

In order to have the system time synchronized, the NTP servers IP addresses have to be defined in the following configuration file:

`/etc/ntp.conf`

If NTP synchronization is used, the corresponding system unit (`ntpd.service`) has to be enabled and started.

Uncomment the `#NTP=` line and add the specific NTP server addresses separated by a blank, like:

```
# For more information about this file, see the man pages
# ntp.conf(5), ntp_acc(5), ntp_auth(5), ntp_clock(5), ntp_misc(5),
ntp_mon(5).

driftfile /var/lib/ntp/drift

# Permit time synchronization with our time source, but do not
# permit the source to query or modify the service on this system.
restrict default nomodify notrap nopeer noquery

# Permit all access over the loopback interface. This could
# be tightened as well, but to do so would effect some of
# the administrative functions.
restrict 127.0.0.1
restrict ::1

# Hosts on local network are less restricted.
#restrict 192.168.1.0 mask 255.255.255.0 nomodify notrap

# Use public servers from the pool.ntp.org project.
# Please consider joining the pool (http://www.pool.ntp.org/join.html).
server 172.24.185.14
server 172.24.185.30
server 172.24.185.46
server 172.25.185.14
server 172.25.185.30
server 172.25.185.46

#broadcast 192.168.1.255 autokey          # broadcast server
#broadcastclient                         # broadcast client
```

```
#broadcast 224.0.1.1 autokey          # multicast server
#multicastclient 224.0.1.1           # multicast client
#manyCastServer 239.255.254.254      # manyCast server
#manyCastClient 239.255.254.254 autokey # manyCast client

# Enable public key cryptography.
#crypto

includefile /etc/ntp/crypto/pw

# Key file containing the keys and key identifiers used when operating
# with symmetric key cryptography.
keys /etc/ntp/keys

# Specify the key identifiers which are trusted.
#trustedkey 4 8 42

# Specify the key identifier to use with the ntpdc utility.
#requestkey 8

# Specify the key identifier to use with the ntpq utility.
#controlkey 8

# Enable writing of statistics records.
#statistics clockstats cryptostats loopstats peerstats

# Disable the monitoring facility to prevent amplification attacks using
ntpdc
# monlist command when default restrict does not include the noquery flag.
# See CVE-2013-5211 for more details.
# Note: Monitoring will not be disabled with the limited restriction flag.
disable monitor
```

NB:

- Time is set as UTC
- NTP synchronization status can be seen using
ntpq -p
- Time should be checked and adapted, if required, using the following command.
date -u mmddHHMM

4.2.5 Chrony time synchronization (RHEL 8)

In order to have the system time synchronized, the NTP servers IP addresses have to be defined in the following configuration file:

/etc/chrony.conf

If chrony synchronization is used, the corresponding system unit (`chronyd`) has to be enabled and started.

Add the specific NTP server addresses on several lines, like shown in the example below:

```
# Use public servers from the pool.ntp.org project.
# Please consider joining the pool (http://www.pool.ntp.org/join.html).
#pool 2.rhel.pool.ntp.org iburst
# Record the rate at which the system clock gains/losses time.
driftfile /var/lib/chrony/drift

# Allow the system clock to be stepped in the first three updates
# if its offset is larger than 1 second.
makestep 1.0 3
```

```
# Enable kernel synchronization of the real-time clock (RTC).  
rtcsync  
  
# Enable hardware timestamping on all interfaces that support it.  
#hwtimestamp *  
  
# Increase the minimum number of selectable sources required to adjust  
# the system clock.  
#minsources 2  
  
# Allow NTP client access from local network.  
#allow 192.168.0.0/16  
  
# Serve time even if not synchronized to a time source.  
#local stratum 10  
  
# Specify file containing keys for NTP authentication.  
keyfile /etc/chrony.keys  
  
# Get TAI-UTC offset and leap seconds from the system tz database.  
leapsectz right/UTC  
  
# Specify directory for log files.  
logdir /var/log/chrony  
  
# Select which information is logged.  
#log measurements statistics tracking  
  
server 172.24.185.14  
server 172.24.185.30  
server 172.24.185.46  
server 172.25.185.14  
server 172.25.185.30  
server 172.25.185.46
```

NB:

- Time is set as UTC
- Check the system unit. It should be active (running)
systemctl status chronyd
- NTP synchronization status can be seen using
timedatectl
chronyc sources
- Time should be checked and adapted, if required, using the following command.
date -u mmddHHMM

4.3 Checklist for ATN decoder customization

Below is an overview of all the changes that need to be done to customize the ATN decoder for a specific location. There is no need to change any other file on the ATN decoder, except for troubleshooting or testing new functionalities.

Skyguide will only support the standard delivered configuration.

Changes to be done	Files to be adapted	See
Hostname	/opt/cpdlc/config/hostname	4.2.1.1

Welcome message	/opt/cpdlc/config/motd	4.2.1.3
General configuration	/opt/cpdlc/config/general_parameter_file	4.2.2
ATN decoding software	/usr/PDEC/livemonitoring/cron /usr/PDEC/Airtel_PDEC_EXE_<version>_<OS>/bin	5
File transfer via Filebeat	/opt/cpdlc/config/filebeat*.yml	4.2.3
PKI Certificates	/opt/cpdlc/certs	4.2.3
NTP	/etc/ntp.conf (CentOS 7) /etc/chrony.conf (RHEL 8)	4.2.4 4.2.5

Skyguide recommends to customize the ATN decoder in the following order:

1. Boot the ATN decoder without any connection (no network nor receivers)
2. Make all the requested configuration changes, according to the preceding table
3. Set date and time (always use UTC as timezone)
4. Remove all data and archived files
5. Reboot the ATN decoder
6. Check that all system units are up and running
7. Connect the network cable
8. Check if the network is up and running and if the data server can be reached
9. Check that the data files under /usr/PDEC/livemonitoring/logs are generated and updated using the tail -f command

4.4 Physical connections

- Power supply has to be connected to an UPS protected source
- LAN has to be connected to the defined switch port which provides
 - A copy of the ATN data
 - Internet access to send the data to the DLS cockpit cluster in the cloud

NB: According to the configuration, the ATN sniffing can be done over one Ethernet port (e.g. eth0) and the data can be sent to the cloud over another Ethernet port (e.g. eth1).

5 ATN decoder software installation and configuration

5.1 Installation

The PDEC is delivered with the following files

- Airtel_PDEC_<version>_<ANSP>_livemonitoring.tar.gz
- Airtel_PDEC_EXE_<version>_<OS>.tar.gz

The following procedure needs to be followed for the installation:

Step	Description	Remarks
1.	<p>Check that the PLM_SNIFFED_INTERFACE and PLM_SNIFFED_ADDRESS are set properly:</p> <pre># env</pre> <p>If the result is not as expected, modify the .bashrc file:</p> <pre># cd # vi .bashrc</pre> <p>Add the following lines at the end of the file with the appropriate information:</p> <pre>export PLM_SNIFFED_INTERFACE=xxxx export PLM_SNIFFED_ADDRESS=xxx.xxx.xxx.xxx</pre> <p>Close the terminal window that you're using, open a new terminal window and check that both variables are properly set using the <code>env</code> command.</p>	<p>The variables shall appear with their appropriate values (e.g. eth0 and 156.135.249.20). The PLM_SNIFFED_ADDRESS should be e.g. the address of the ATN GGR which communicates with SITA.</p> <p>The .bashrc file is a hidden file and cannot be seen with the <code>ls -l</code> command</p>
2.	<p>Stop the decoding and moving of minute files to daily files in the crontab by commenting the following line (add a # at the beginning of the line):</p> <pre>* 1-23 * * * sleep 45 && /usr/PDEC/livemonitoring/cron/decode_capture.sh && sleep 30 && /opt/cpdlc/scripts/purge_old_files.sh >> /var/log/ansplogs/purge_old_files.log 2>&1</pre>	<p>As from this pt, the data flow to the cockpit is interrupted.</p> <p>Only raw data continue to be sniffed.</p>
3.	Wait 2-3 minutes to make sure that all minute files have been produced and moved to the daily file.	
4.	<pre># mv livemonitoring livemonitoring.<previous_version></pre>	Move previous version. The sniffed data continue to be logged in this folder.
5.	<pre># tar xvfz Airtel_PDEC_<version>_SKYGUIDE*</pre>	
6.	<pre># tar xvfz Airtel_PDEC_EXE_<version>_<OS>.tar.gz</pre>	A new livemonitoring folder is created.
As of next step, the steps should be done quickly to avoid losing too many data. The commands can be launched at once using the command provided under step 10 (NB).		
7.	<pre>/usr/PDEC/livemonitoring/cron/stop_capture.sh</pre>	Stop tcpdump
8.	<pre>mv /usr/PDEC/livemonitoring.<previous_version>/logs/YYYYMM</pre>	Recuperate the live_monitoring file

	MDD_live_capture.log /usr/PDEC/livemonitoring/logs	from the previous monitoring folder
9.	mv /usr/PDEC/livemonitoring.<previous_version>/data/pdus.csv /usr/PDEC/livemonitoring/data	Recuperate the pdus.csv file from the previous monitoring folder
10.	/usr/PDEC/livemonitoring/cron/decode_capture.sh	Decode the raw data recuperated from the previous monitoring folder
11.	/usr/PDEC/livemonitoring/cron/start_capture.sh	Restart tcpdump

NB: The previous commands (steps 7 – 11) can be launched at once with the following one:

```
/usr/PDEC/livemonitoring/cron/stop_capture.sh && mv
/usr/PDEC/livemonitoring.<previous_version>/logs/YYYYMMDD_live_capture.log
/usr/PDEC/livemonitoring/logs && mv
/usr/PDEC/livemonitoring.<previous_version>/data/pdus.csv
/usr/PDEC/livemonitoring/data &&
/usr/PDEC/livemonitoring/cron/decode_capture.sh &&
/usr/PDEC/livemonitoring/cron/start_capture.sh
```

The data are now sniffed again and put in the respective folders of the new livemonitoring folder.

12.	Reactivate the decoding and moving of minute files to daily files in the crontab by de-commenting the following line (remove the # at the beginning of the line): <code>* 1-23 * * * sleep 45 && /usr/PDEC/livemonitoring/cron/decode_capture.sh && sleep 30 && /opt/cpdlc/scripts/purge_old_files.sh >> /var/log/ansplogs/purge_old_files.log 2>&1</code>	As from this pt, the data flow to the cockpit is reactivated. The first move of minute file will move all data captured during the interval
13.	Check that the raw data are properly recorded under: <code>/usr/PDEC/livemonitoring/logs</code>	
14.	Check that the data are properly decoded under: <code>/usr/PDEC/livemonitoring/data</code>	
15.	Check that the decoded data are moved regularly according to crontab to the daily file under: <code>/opt/data/xcpdlc</code>	

5.2 Cronjob

The regular generation of the decoded csv file is managed through the "start_capture.sh" script which is called at midnight by cronjob to produce a new live_monitoring daily file in /usr/PDEC/livemonitoring/logs. These files contain raw, undecoded data.

To produce decoded files every 1 minute, the decode_capture scripts have to be invoked every minute. Ideally, in crontab, the script is invoked and immediately followed with the purge_old_files.sh script which will append the decoded minute file to the daily file under /opt/data/xcpdlc.

Add the following to the user crontab (crontab -e):

```
@reboot sleep 60 && PLM_SNIFTED_ADDRESS=156.135.249.28
PLM_SNIFTED_INTERFACE=eth0 /usr/PDEC/livemonitoring/cron/start_capture.sh
```

```

0 0 * * * PLM_SNIFFED_ADDRESS=156.135.249.28 PLM_SNIFFED_INTERFACE=enp0s31f6
/usr/PDEC/livemonitoring/cron/start_capture.sh

0 0 * * * sleep 45 &&
/usr/PDEC/livemonitoring/cron/decode_capture_midnight.sh && sleep 30 &&
/opt/cpdlc/scripts/purge_old_files.sh >>
/var/log/ansplogs/purge_old_files.log 2>&1

1-59 0 * * * sleep 45 &&
/usr/PDEC/livemonitoring/cron/decode_capture_inc_yesterday.sh && sleep 30 &&
/opt/cpdlc/scripts/purge_old_files.sh >>
/var/log/ansplogs/purge_old_files.log 2>&1

* 1-23 * * * sleep 45 && /usr/PDEC/livemonitoring/cron/decode_capture.sh &&
sleep 30 && /opt/cpdlc/scripts/purge_old_files.sh >>
/var/log/ansplogs/purge_old_files.log 2>&1

```

Notes:

- cron job output is logged in the system log (/var/log/messages)
- The variables `PLM_SNIFFED_ADDRESS` and `PLM_SNIFFED_INTERFACE` have to be set otherwise the default values set in the `/usr/PDEC/livemonitoring/cron/start_capture.sh` will be used.
- `decode_capture_midnight.sh` is used to decode the log at 00:00 including the data from the previous day (in that case from 23:59-00:00)
- `decode_capture_inc_yesterday.sh` is used to decode the logs at the start of a new day between 00h01 and 00h59, including the previous day log, to keep CM information knowledge
- The `sleep 45` and `sleep 30` elements in the crontab are only there for ATN decoders which provide ATN data that will need to be enriched with SATCOM data. The additional delay is to make sure that the SATCOM data have been processed by the DLS cockpit cloud before the ATN data are processed.

Cronjob is also used for the following other purposes:

- Automatic restart of the NTP service after boot (only necessary with CentOS 7)
- Automatic restart of the PDEC capture after boot
- Archive and purge data including logging in case of issues
- Automatic relaunch of tcpdump if tcpdump crashes.

Here is the typical cronjob configuration (with data produced every minute).

```

# RTCD and Archive and purge
@reboot sleep 60 && PLM_SNIFFED_ADDRESS=156.135.249.28
PLM_SNIFFED_INTERFACE=enp0s31f6
/usr/PDEC/livemonitoring/cron/start_capture.sh
0 0 * * * PLM_SNIFFED_ADDRESS=156.135.249.28 PLM_SNIFFED_INTERFACE=enp0s31f6
/usr/PDEC/livemonitoring/cron/start_capture.sh
0 0 * * * sleep 45 &&
/usr/PDEC/livemonitoring/cron/decode_capture_midnight.sh && sleep 30 &&
/opt/cpdlc/scripts/purge_old_files.sh >>
/var/log/ansplogs/purge_old_files.log 2>&1
1-59 0 * * * sleep 45 &&
/usr/PDEC/livemonitoring/cron/decode_capture_inc_yesterday.sh && sleep 30 &&
/opt/cpdlc/scripts/purge_old_files.sh >>
/var/log/ansplogs/purge_old_files.log 2>&1
* 1-23 * * * sleep 45 && /usr/PDEC/livemonitoring/cron/decode_capture.sh &&
sleep 30 && /opt/cpdlc/scripts/purge_old_files.sh >>
/var/log/ansplogs/purge_old_files.log 2>&1

```

```

0 1 * * * /usr/PDEC/livemonitoring/cron/generate_lisat.sh
#
# RTCD Health Check
#
* * * * sleep 30 && PLM_SNIFTED_ADDRESS=156.135.249.28
PLM_SNIFTED_INTERFACE=enp0s31f6 /opt/cpdlc/scripts/check_pdec_tcpdump.sh >>
/var/log/ansplogs/check_pdec.log
#
# DF dump restart (to generate new logfile for each day)
#
0 0 * * * /usr/bin/killall dumpDF

```

5.3 Parallel operations

It is not allowed with a single license to run the PDEC on 2 different hardware and sniff the ATN in parallel. However a cold standby solution is allowed within the same license price.

5.4 Output file

The output file is in .csv format with comma separator (",") and contains the following information (ref.:

Field	Type	Description	Example(s)
SYSTEMDATE	Date	The date when the data was captured by PDEC	19/03/2022
SYSTEMTIME	Time	The timestamp when the data was captured by PDEC (including ms)	01:47:03.164
ACC	Text	The ACC code (up to 8 letters)	LSAZ or XXXX
FlightIdentifier	Text	The Flight Identifier of the flight (recovered from CM Logon, empty if there is no CM Logon)	EDW399Z
AirframeIdent	Text	The 24bits address of the aircraft	0x4B18BB
Msg_Date	Date	The date embedded in the CPDLC data and ADS Report.	18/08/2022
Msg_Time	Time	The timestamp embedded in the CPDLC data and ADS Report (does not contain ms)	08:52:14
Air_Gnd	Text	The source of the data (air originated or gnd originated)	ATN_GND
MSG_Type	Text	Possible values provided in the columns on the right. In case of TP4 DT, the CM, CPDLC or ADS-C message type is provided instead of TP4 DT:	CM Logon Resp. (+ve) CPDLC end-response ADS Report Periodic TP4 DR IDRP Update
UM/DM	Integer	The UM/DM number (integer)	62
UM/DM CONCAT	Integer	In case of CPDLC message concatenation, the second	98

		element (integer)	
MIN	Integer	The MIN of the CPDLC message	1
MRN	Integer	the MRN of the CPDLC message	2
ACK	Integer	The ACK of the CPDLC message	0
Description	Text	Details according to the message type provided in MSG_Type	ADS (v1),CPDLC (v1), CPDLC (v2) UM20 CLIMB TO 390 LEVEL FL epp-flight-plan-change HOLD_TIMER_EXPIRED NO_SUBCODE
ADEP	Text	Departure Airport. Info recovered from the CM Logon information if available. If there is no CM Logon, the values are empty.	LSGG
ADES	Text	Destination Airport. Info recovered from the CM Logon information if available. If there is no CM Logon, the values are empty.	KJFK
LI	Integer	TP4 Length Indicator	10
CDT	Integer	TP4 credit	0
DST-REF	Hex	TP4 destination reference	0xC8B7
SRC-REF	Hex	TP4 source reference	0x0040
CLASS/EOT/REASON	Integer	TP4 Class/EOT/Reason	1
TPDU-NR	Integer	TP4 TPDU number	1
NTPDU-NR	Integer	TP4 next TPDU number	2
DUID	Integer	PDU ID in decimal	420
SrcNsap	Text	The source NSAP address possibly replaced with up to 10-letters code from atsu.csv	EDYY or 47002741444C48003C6481000053414142000001
DstNsap	Text	The destination NSAP address possibly replaced with up to 10-letters code from atsu.csv	EDYY or 47002741444C48003C6481000053414142000001
ContractNumber	Integer	ADS Contract Number	1
Position	Text	ADS Report Position, format: "<latitude> <longitude> <level>", with level optional	52:33:580 -10:11:430 level(300)
FOM	Text	ADS Report FigureOfMerit, format: "<estimated-position-uncertainty> <multiple-navigational-units-operating> <aais-availability>"	8 TRUE TRUE

Rate	Text	ADS Periodic Report Rate	00:00:30
IDRP_SEQ	Integer	IDRP Sequence Number	6
IDRP_ACK	Integer	IDRP ACK Number	2
IDRP_CDT_OFFRD	Integer	IDRP Credit Offered	3
IDRP_CDT_AVAIL	Integer	IDRP Credit Available	1
CircuitID	Text	IP address of the remote router sending or receiving the data.	57.77.110.33
Version	Text	CM/CPDLC/ADS version	v1
ASN1Message	Text	The full ASN.1 decoding of the PDU	<pre>"ADSRequestContract { periodic-contract { contract_number = 1 reporting_rate { reporting-time-minutes-scale = 4 } extended_projected_profile_modulus { modulus = 1 epp_window { number-of-way-points = 70 } } speed_schedule_profile_modulus = 1 } }"</pre>
CLNP-PDU	Text	The full hexadecimal value of the CLNP PDU	0x814E01289C0...

6 System check

To ensure the system is running as requested, there are some available checks that are launched at the user login or that can be launched manually.

6.1.1 Filebeat

A. Check the filebeat configuration:

If there is more than one filebeat instance running on the system, use the right configuration file name as parameter to check the instance.

```
#/usr/share/filebeat/bin/filebeat test config -c
/opt/cpdlc/config/filebeat-xcpdlc.yml
Config OK
#
```

This output means that the filebeat configuration syntax is correct, which doesn't mean it will work to 100%.

B. Check the filebeat connection:

```
#/usr/share/filebeat/bin/filebeat test output -c
/opt/cpdlc/config/filebeat.yml -e -E XCPDLC_LOADBALANCE=false -E
XCPDLC_HOSTS=xxx.xxx.xxx.xxx:yyyy -E
XCPDLC_PROXYURL=socks5://xxx.xxx.xxx.xxx:yyyy
logstash: node15638-enaire-cpdlc-cockpit.jcloud.ik-server.com:11107...
connection...
    parse host... OK
    dns lookup... OK
    addresses: 185.74.71.126, 185.74.71.125
    dial up... OK
TLS...
    security: server's certificate chain verification is enabled
    handshake... OK
    TLS version: TLSv1.2
    dial up... OK
    talk to server... OK
#
```

This output means that the filebeat configuration is working and that all network partners are reachable.

C. Check the running filebeat log:

```
#sudo tail -f /var/log/filebeat/filebeat
```

6.1.2 NTP (CentOS 7.2)

The status of the NTP client can be displayed using one of the following commands:

```
# systemctl status ntpd.service
# ntpq -p
```

```
# timedatectl status
```

```
[admin@gwcpcdcli-pdec2 ntp]$ systemctl status ntpd.service
● ntpd.service - Network Time Service
  Loaded: loaded (/usr/lib/systemd/system/ntp.service; enabled; vendor preset: disabled)
  Active: active (running) since Mon 2021-03-08 15:24:18 GMT; 1 months 25 days ago
    Main PID: 2500 (ntpd)
      CGroup: /system.slice/ntp.service
              └─2500 /usr/sbin/ntpd -u ntp:ntp -g

Warning: Journal has been rotated since unit was started. Log output is incomplete or unavailable.
[admin@gwcpcdcli-pdec2 ntp]$
```

```
[admin@gwcpcdcli-pdec2 ~]$ ntpq -p
      remote          refid      st t when poll reach   delay    offset  jitter
=====+
*172.24.185.14 .PZF.        1 u 49 1024 377  0.879  -0.280  0.253
+172.24.185.30 .GPS.       1 u 1039 1024 377  1.002  -0.151  0.329
+172.24.185.46 .GPS.       1 u 192 1024 377  1.000  -0.169  0.254
-172.25.185.14 .PZF.       1 u 955 1024 377  6.350  -1.424  0.305
+172.25.185.30 .PZF.       1 u 1083 1024 377  4.552  -0.414  0.364
-172.25.185.46 .GPS.       1 u 473 1024 377  6.414  -1.045  0.317
[admin@gwcpcdcli-pdec2 ~]$
```

```
[admin@gwcpcdcli-pdec2 ~]$ timedatectl status
    Local time: Mon 2021-05-03 15:06:52 BST
    Universal time: Mon 2021-05-03 14:06:52 UTC
        RTC time: Mon 2021-05-03 14:06:52
        Time zone: Europe/London (BST, +0100)
    NTP enabled: yes
    NTP synchronized: yes
    RTC in local TZ: no
        DST active: yes
Last DST change: DST began at
                  Sun 2021-03-28 00:59:59 GMT
                  Sun 2021-03-28 02:00:00 BST
Next DST change: DST ends (the clock jumps one hour backwards) at
                  Sun 2021-10-31 01:59:59 BST
                  Sun 2021-10-31 01:00:00 GMT
[admin@gwcpcdcli-pdec2 ~]$
```

NB: Not synchronized time will not affect the ATN decoder but will insert data with wrong timestamps in the DLS cockpit and **deteriorate its accuracy!**

6.1.3 Chronyd (RHEL 8)

The status of the NTP client can be displayed using one of the following commands:

```
# systemctl status chronyd
# chronyc sources
# timedatectl
```

```
[root@gsvdls01 etc]# systemctl status chronyd
● chronyd.service - NTP client/server
  Loaded: loaded (/usr/lib/systemd/system/chronyd.service; enabled; vendor preset: enabled)
  Active: active (running) since Wed 2023-01-25 01:32:40 UTC; 2 months 3 days ago
    Docs: man:chronyd(8)
          man:chrony.conf(5)
 Process: 820 ExecStartPost=/usr/libexec/chrony-helper update-daemon (code=exited, status=0/SUCCESS)
 Process: 782 ExecStart=/usr/sbin/chronyd $OPTIONS (code=exited, status=0/SUCCESS)
 Main PID: 793 (chronyd)
   Tasks: 1 (limit: 49987)
  Memory: 820.0K
    CGroup: /system.slice/chronyd.service
            └─793 /usr/sbin/chronyd
```

```
[root@gsvdls01 etc]# chronyc sources
210 Number of sources = 6
MS Name/IP address      Stratum Poll Reach LastRx Last sample
=====
^* 172.24.185.14          1 10   377   734   +282us[ +270us] +/- 1688us
^+ 172.24.185.30          1 10   377   905   -246us[ -258us] +/- 1783us
^+ 172.24.185.46          1 10   377   276   -345us[ -345us] +/- 1595us
^+ 172.25.185.14          1 10   377  1011   +771us[ +759us] +/- 4343us
^+ 172.25.185.30          1 10   377  1137   -91us[ -102us] +/- 3456us
^+ 172.25.185.46          1 10   377   405   +664us[ +664us] +/- 4361us
```

```
[root@gsvdls01 etc]# timedatectl status
           Local time: Thu 2023-03-30 07:52:17 UTC
           Universal time: Thu 2023-03-30 07:52:17 UTC
                 RTC time: Thu 2023-03-30 07:52:17
                  Time zone: Etc/UTC (UTC, +0000)
System clock synchronized: yes
          NTP service: active
    RTC in local TZ: no
```

NB: Not synchronized time will not affect the ATN decoder but will insert data with wrong timestamps in the DLS cockpit and **deteriorate its accuracy!**

7 Useful commands

Some useful commands are provided hereafter. They can be used to monitor and administer the ATN decoder:

# date -u mmddHHMM	Sets the system date in UTC format
# systemctl status <unit> e.g. # sudo systemctl status rtcd.service	Show the unit status, e.g.
# systemctl stop <unit>	Stop the specified unit
# systemctl start <unit>	Start the specified unit
# systemctl restart <unit>	Restart the specified unit
# top	Displays some information about the resources utilization
# df -kH	Displays the amount of space used on the hard disk in a human readable format
#timedatectl status	Displays date, time and the status of the NTP time synchronization.
#ntpq -p	Displays NTP servers synchronization status (CentOS 7)
#chronyc sources	Displays NTP servers synchronization status (RHEL 8)

8 References

8.1 Software releases

Software	Release	Comment
CentOS	7.2	GUI installed but not started automatically Can be used for all PDEC releases.
RHEL	8.4 or 8.6	GUI installed but not started automatically Only compatible with PDEC C3 release or later
PDEC	C5p2	Stable release (incl. CPDLC v2 and ADS-C). Can be installed Ubuntu LTS 22 (see note below) C5p2 can also be installed on CentOS 7.2 but it necessitates specific software provided by Airtel ATN.
Filebeat	7.5.2	Can also be a later release

Note:

For an Ubuntu implementation, the following aspects need to be taken into consideration:

- Use the RHEL8 distribution
- Use the gawk distribution and not mawk
- The PLM_AWK environment variable shall point to gawk
- The PLM_TCPDUMP environment variable shall point to tcpdump

9 Abbreviations, Terms and Definitions

9.1 Abbreviations

Abbreviation	Full expression
ACSP	Aeronautical Communication Service Provider
ADS-C	Automatic Dependant Surveillance Contract
ANSP	Air Navigation Service Provider
ATN	Aeronautical Telecommunication Network
CM	Context Management
CPDLC	Controller Pilot Data Link Communication
DNS	Domain Name System
IDRP	Inter Domain Routing Protocol
LAN	Local Area Network
LTS	Long Term Support
NTP	Network Time Protocol

RAM	Random Access Memory
RHEL	Redhat Enterprise Linux
SSD	Solid State Drive
SSL	Secure Socket Layer
TP4	Transport Protocol 4
UTC	Coordinated Universal Time

9.2 Terms and Definitions

Fill in the link to common glossary where the terms and definitions used in this document are defined or define it when appropriate

Term	Definition
DLS cockpit	Data Link Services Cockpit product of skyguide
PDEC	Protocol Decoder software from Airtel-ATN
RTCD	Real-Time Collecting and Decoding software from Airtel-ATN