DIGITAL AIR INTERFACE PROTOCOL PA

TRISTAR

Prepared by: Roberto Carlos Martín Pastor

Approved by: Óscar Casado Lobato

Authoritized by: Alfredo Martín Olalla

Code: GMV-TRISTAR-TN-033

Internal Code: GMV 22392/15 V3/15

Version 3.0

Date: 02/12/2015

GMV SISTEMAS S.A.U.
P.T.B. Parcela 101; Boecillo; Valladolid 47151
Tel. +34 983546554; Fax. +34 983546553
www.gmv.com

STATE SHEET

Version	Date	Pages	Changes
1	13/08/2015	23	Initial Version
2	20/11/2015	23	Clarification of scope and purpose of document in points 1.1, 1.2 and 1.6
3	02/12/2015	24	Clarification on logical layers, schedules definition and real time estimations in 1.1 paragraph

INDEX

1. INTRODUCTION 5
1.1. THIS DOCUMENT 5
1.2. EARLIER VERSIONS 5
1.3. CONTEXT 5
1.4. USE CASES 5
1.5. SCOPE 6
1.6. DEFINITIONS & ACRONYMS 6
1.6.1. DEFINITIONS 6
1.6.2. ACRONYMS 6
2. BASIS OF SPECIFICATION 8
2.1. ARCHITECTURE 8
2.2. MANDATORY AND OPTIONAL FIELDS 8
2.3. ACKNOWLEDGEMENTS 8
2.4. EFFICIENCY OF COMMUNICATION 9
2.5. FIELD CODING 9
3. MESSAGE BASICS 11
3.1. WRAPPER STRUCTURE 11
3.2. ACKNOWLEDGEMENT STRUCTURE 13
4. MESSAGE STRUCTURES 16
4.1. OVERVIEW 16
4.2. MESSAGE ACKNOWLEDGEMENT 16
4.3. LOG ON REQUEST – MESSAGE ID #138 16
4.4. LOG ON RESPONSE – MESSAGE ID #132 17
4.5. LOG OFF REQUEST – MESSAGE ID #11 18
4.6. PET POSITIONS – MESSAGE ID #139 19
4.7. PA ERROR (PA TO CENTRE) – MESSAGE ID #140 19
4.8. PET POSITION RESPONSE – MESSAGE ID #200 19
4.9. LOG ON RESPONSE – MESSAGE ID #132 20
4.10. ERROR RESPONSE (CENTRE TO PA) – MESSAGE ID #101 22
5. ANNEX I: SECURING COMMUNICATIONS 23

LIST OF TABLES & FIGURES

Tabla 1-1 Definitions 6
Tabla 1-2 Acronyms 6

No se **encuentran elementos de tabla de ilustraciones.**

1. INTRODUCTION

1.1. THIS DOCUMENT

1.1.1 This document has been produced by GMV as a specification for a set of messages that can be communicated between PA-mounted devices and office-based systems, using a digital open

Protocol, based on GMV experience, standards and best practices of RTIG

1.1.2 The status of this document is **released**, as of 20/11/2015**.** This issue is released as version

1.1.3 Document scope is the extension of TRISTAR project according to SWIZ, allowing third parties to integrate and extend TRISTAR system.

1.1.4 Reproduction and distribution of the document is allowed according to situations included in TRISTAR project SWIZ

1.1.5 Logical layer of protocol is described in paragraphs 1.5 and 2.1, referring to the Scope and Architecture of the protocol

1.1.6 6 Configuration of displays can be retrieved from information about topology and schedules defined in the interface documentation:

GMV-TRISTAR-TN-015	Web Services Schedules Integration
GMV-TRISTAR-TN-016	Database Views

1.1.7 Current Protocol includes real time information to be retrieved by the new devices, which together with the information defined in the interface documentation allows devices to calculate and estimate arrival and departures from bus stops.

1.2. EARLIER VERSIONS

1.2.1 This version includes clarifications about scope and authority of document

1.3. CONTEXT

1.3.1 Currently, real time systems use proprietary air interface protocols between PA and centre. Originally this practice was a result of the way that such systems were produced by the supply industry, but it was also necessary because of the need to ensure efficient use of limited radio spectrum. Spectrum efficiency remains a big concern with analogue Band III radio.

1.4. USE CASES

1.4.1 The messages outlined in this specification can be used by different stakeholders in a variety of ways. Local authorities as well as PA operators will have the opportunity to identify, locate and check on the status of their PAs throughout their period of service.

1.4.2 This specification could also assist PA operators improving the accuracy of PA service information. Operators themselves can benefit through the provision of detailed PA and driver information, while also providing service users with more accurate RTI data while a PA is in service, as well as improving the accuracy of PA timetabling.

1**.4.5** Note that this specification does not impose functional requirements on the system. Issues such as required location accuracy, communications availability etc must be specified separately.

1.5. SCOPE

1.5.1 This specification does not attempt to duplicate all possible functions of an RTI PA. It concentrates, rather, on the definition of a basic, common set of messages that would meet most common needs.

1.5.2 The core of the specification is presented in four message sets

- **Log On/Off PA**
  - **log on** (Log On Request and Log On Response messages): provides a notification that a PA is powering on and about to begin a period of service. This message contains details of the identity of the PA. Following the receipt of this Log On message by the appropriate centre, a unique identifier is issued to the PA to be used for that particular period of service.
  - **log off** (Log Off Request): provides a notification that a PA has ended a period of service and is powering down. This message contains the session identifier for the PA.
- **Pet Position**
  - **Pet Positions** (Positions Request message): This message is used by the PA to obtain the information of the position is of the vehicles, to calculate the same arrival time at your stop.
- **Errors**
  - This message is used to indicate the PA errors on it device Information.

1.5.3 The specification also includes a number of common message structuring, framing and coding rules

1.5.4 It is anticipated that guidance will be developed on operational aspects of using this protocol which are not suitable for standardisation at this point, eg how to manage update frequency and timeout thresholds.

1.5.5 These messages are designed for use in systems using IP-compatible communications, such as GPRS, 3G and WiFi..

1.6. DEFINITIONS & ACRONYMS

1.6.1. DEFINITIONS

The concepts and terms that have been used in the document and that was considered appropriate to define them, are in the following table

Tabla 1-1 Definitions

Concept/Term	Definition

1.6.2. ACRONYMS

The acronyms have been used in the document and it was considered appropriate to define them, are in the following table:

Tabla 1-2 Acronyms

Acronym	Definition
SAE	Fleet Management System

Acronym	Definition
OTP	Open Trip Planner
PA	User information panel
SWIZ	TRISTAR project User Requirements

2. BASIS OF SPECIFICATION

2.1. ARCHITECTURE

3.1.1 It is assumed that the device on PA sending data has intelligence and a knowledge of schedule information derived from a TransXChange source. Hence it can send information on the journey and operator. Therefore the production of a standard schedule configuration file for the PA is out of scope of this standard.

3.1.2 In the absence of this on PA intelligence only a basic feature set will be supported based on returning journey details as entered by the driver. Note also that the sending device will also have to be configured for operator, PA ID etc.

3.1.3 UDP/IP is used rather than TCP/IP for communications protocol as it has a lower overhead (28 bytes per message – 20 bytes IP header, 8 bytes UDP header). This may impose operational requirements for trapping non-communicating units.

3.1.4 PA IP addresses should be identified from the source IP address information in the IP header when a PA logs on. Once the PA session ID has been issued, this may be used to trap changes in IP address (eg if a PA falls out of GPRS coverage and is subsequently given a new dynamic IP address). Alternatively, the PA may be required to log in again, and be given a new session ID.

2.1.5 The same port should be used for sending and receiving UDP telegrams. The default port and IP address to be used needs to be configured into the PA.

2.2. MANDATORY AND OPTIONAL FIELDS

3.2.1 Within each message, a number of parameters have been outlined which will form the structure of the message itself. These parameters are described as either:

- Mandatory: parameters which must be included in messages for every RTPI scheme.
- Optional: parameters which may be included subject to the specification of the stakeholders involved.

3.2.2 In the case of the Log on, Journey Details and Positional Update messages, all parameters have been allocated a mandatory status in order to fix the length of these messages. This was carried out in order for a clear price structure for users. In situations where users do not use particular parameters within a message, many of the parameters can be populated with null values.

2.3. ACKNOWLEDGEMENTS

3.3.1 An acknowledgement for a message is requested by setting the Acknowledgment flag in the wrapper. Section 4 indicates the acknowledgment requirements for each message type – when sending a message of a given type, the Acknowledgment flag must be set accordingly.

3.3.2 Where more than one message is contained within a single wrapper, the Acknowledgement flag applies only to the first message – messages after the first can only be of a type which does

not require an acknowledgement.

2.4. EFFICIENCY OF COMMUNICATION

3.4.1 The efficiency of communicating these messages has been a major point of discussion when producing this specification. Due to their use of a narrower bandwidth, binary numbers are used where possible instead of ASCII characters. The result is a reduction in message size and therefore communications cost. A similar argument was also made for the use of numeric characters as opposed to alphanumeric ones in a bid to reduce the price (numeric codes can be squeezed two characters to a byte, where alpha codes need a byte per character).

3.4.2 A unique Session PA Identifier is used to improve the efficiency of the sending the messages contained within this specification. Instead of a PA being required to send its PA and operator ID for each message sent, this information will only be contained within the Log On message sent at the beginning of a period of service. Following the receipt of this Log On message by the appropriate centre, the unique identifier will be issued to the PA, to be used for any subsequent messages sent.

3.4.3 Aside from this coding efficiency, no provision has been made for the use of data compression algorithms. The rationale is that:

– The benefit of this last few percent of bandwidth efficiency is likely to modest.
– It would impose an avoidable processing overhead on systems.

3.4.4 While the specification allows for multiple messages to be combined in one wrapper, this may not always be possible because of acknowledgement requests. However, it is possible to combine multiple wrappers in the same UDP frame.

2.5. FIELD CODING

3.5.1 With the exception of Event Messages, all fields are fixed length. Fields may be one of the following Field Types:

– C = Alphanumeric character (ASCII, encoded one to a byte), with no leading spaces and with a terminating null (Hex 00) provided there is sufficient space[1]
– N = Nibble occupying 4 bits representing either a BCD numeric digit in the range 0-9 or as up to four individual binary flags; '0' or '1' (encoded two to a byte)
– X = Hexadecimal value representing a binary numeric value in the range 0-255, or up to eight individual binary flags '0' or '1' (encoded as one byte) [2]
– S = Binary Signed – 2's Complement Integer
– U = Binary Unsigned Integer

3.5.2 Where allowable and appropriate, null values will be as follows:

– For C fields, a sequence of Hex 00;
– For numerical N fields, a sequence of Hex F (a non-existent numeric)
– For numerical X fields, a sequence of Hex FF (equivalent to the value -1)
– For Signed Integer there is no Null Value an invalid value may however be defined for specific fields.
– For Unsigned Integer the maximum value is treated as NULL (e.g. 0xFFFF)

Bitfield (flag) fields, either N or X, are never permitted to be null.

3.5.3 The following special codes are used, as sequences (normally pairs) of N type (nibble) digits:

– h = Hour (24 hour clock)

– m = Minute
– s = Second
– D = Day
– M = Month
– Y = Year

3.5.4 Parameters are sent in normal reading order, ie first character or most significant byte first ("Big Endian"). For example:

– Public Service Code 15A, with format CCCCCC, would be transmitted as bytes 31, 35, 41, 00, 00, 00 (Hex).
– Format version 1.23, with format NNNN, would be transmitted as two bytes 01, 23 (Binary coded decimal).
– Direction – Inbound ('1'), with format X, would be transmitted as one byte, 01 Hex (or equivalently 00000001 binary).
– Format XXXX (0xFECD3523) would be transmitted as 4 bytes 0xFE, 0xCD, 0x35, 0x23.

3.5.5 Field values marked "not used" etc should not be used for vendor-specific extensions, to avoid difficulties arising with systems built to future versions of this specification.

3. MESSAGE BASICS

3.1. WRAPPER STRUCTURE

4.1.1 Messages using this specification shall have the following wrapper structure.

Parameter	**Mandatory/ Optional**	**Null Value Allowed?**	**Format (*Byte Aligned*)**
Format Version No	Mandatory	No	NNNN *Stored as Nibbles and refers to the* *version of this specification used,* *e.g. 0122 means Version 01.22* *and 0070 means Version 00.70* *Version numbers are regarded as* *right aligned, e.g v3.7 should be* *interpreted as Version 03.07 and* *coded as 0307*
Message Flags	Mandatory	No	X (bit mapped)
Message Counter	Mandatory	No	XX
Session PA Identifier	Mandatory	No	XX
Optional Data Fields	Mandatory	No	XX (bit mapped) *e.g. 1111 0110 0xxx xxxx, where x* *= Don't care*
Message Payload	Mandatory	No	Mandatory and Optional data fields
Message Time Stamp	Mandatory	No	XXXXXX Encoded as BCD: YYMMDDhhmmss *i.e. six bytes*

4.1.2 The Message Structure is applicable to Outgoing and Incoming messages (with respect to the PA-based device) and is structured in such a way as to allow for message enhancements and optional data fields.

4.1.3 Format Version Number : This parameter uniquely identifies the version number of this protocol (ie of this document) that is being used. The Format Version Number shall be amended to

reflect alterations or enhancements to the payload. The Format Version Number requires four Nibbles in BCD format representing two-digit major and two-digit minor version number in the form 00.01 to 99.99 with 00.00 being reserved.

4.1.4 Message Flags: The Message Flags shall be used to qualify the Message in regard of Acknowledgement Requests and Acknowledgements. The ability to identify Test Messages within a live system is supported. This parameter requires a single Nibble and is bit mapped as detailed below.

- Bit 0 – always 0, "This is a Message" (see **Acknowledgement Structure**)
- Bit 1 – Require Acknowledge Flag (If Bit 0 = 0)
  - 0 = No Acknowledgement required
  - 1 = Acknowledgement required
- Bit 1 – Message Acknowledgement Flag (If Bit 0 = 1)
  - 0 = Message is not Acknowledged (Error in message)
  - 1 = Message is Acknowledged
- Bit 2 – Reserved
- Bit 3 – Live or Test/Evaluation Message
  - 0 = Live Message
  - 1 = Test Message
- Bits 4-7 – Reserved.
  - Identifies Multiple Message transmission status:
  - 0000 - Multiple Message transmission **NOT** implemented
  - 0001 - 1111 – Message Wrapper contains between 1 (one) and 15 (fifteen) concatenated messages.

4.1.5 Message Counter: The Message Counter will be under the control of the Message Sender and will provide a unique Message Reference that may be used in subsequent messages if reference to previous messages is required. The Message Counter shall be in the range 0-65535 and will wrap to 0 (Zero) following terminal count increment. New sessions should be default initiate a new Message Counter sequence from 0.

4.1.6 Session PA Identifier : The Session PA Identifier parameter is derived from the Log On Response as a result of the Log On Request generated by the PA. For the initial Log On Request message, when this is clearly unavailable, this parameter shall be set to 0 (zero). This parameter requires two bytes in the range 1-65535 where 0 (zero) is reserved as described above.

4.1.7 Optional Data Fields: The Optional Data Fields parameter describes which if any of the Optional Parameters are contained within the Payload of the message. Where message types and corresponding message payload do not support Optional parameters, then this parameter shall be ignored. The Optional Data Fields parameter requires two bytes in 16 bit format where each bit corresponds to an optional data parameter. The 16 bits are ordered MSB to LSB where the most significant bit corresponds to the first optional parameter. Unused bits ("don't cares") shall be set to 0 (zero). Where an optional field is used, the message must be sent in its own message wrapper.

4.1.8 Message Payload: The Message Payload contains the Mandatory and Optional data fields for he given message type. The ability to accommodate Multiple Messages within the Message Payload is supported as defined by Message Flags bits 4 – 7. The message types are described below.

4.1.9 Message Time Stamp: The Message Time Stamp parameter shall be the GPS's NMEA 0183 representation of its Coordinate Universal Time (UTC). This parameter will prove useful when network delays are encountered.

3.2. ACKNOWLEDGEMENT STRUCTURE

3.2.1 Acknowledgements using this specification shall have the following structure.

Parameter	**Mandatory/ Optional**	**Null Value Allowed?**	**Format (*Byte Aligned*)**
Format Version No	Mandatory	No	NNNN *Stored as Nibbles and refers* *to the version of this* *specification used, e.g. 0122* *means Version 01.22 and* *0070 means Version 00.70* *Version numbers are* *regarded as right aligned,* *e.g v3.7 should be* *interpreted as Version 03.07* *and coded as 0307*
Acknowledgement Flags	Mandatory	No	X (bit mapped)
Acknowledgement Counter	Mandatory	No	XX
Message Counter Reference	Mandatory	No	XX *Previous Incoming Message* *Ref.*
Session PA Identifier	Mandatory	No	XX
Message Time Stamp	Mandatory	No	XXXXXX Encoded as BCD: YYMMDDhhmmss *i.e. six bytes*
Error Number	Mandatory	No	X As defined below

4.2.2 The Acknowledgement Structure may be required in response to the previous Incoming Message generated by either the PA or the Central Communication Server and is structured as follows.

4.2.3 Format Version Number : This parameter uniquely identifies the version number of this protocol (ie of this document) that is being used. The Format Version Number shall be amended to reflect alterations or enhancements to the payload. The Format Version Number requires four Nibbles in BCD format representing major and two digit minor version number in the form 00.01 to 99.99 with 00.00 being reserved.

4.2.4 Acknowledgement Flags: The Acknowledgement Flags shall be used to qualify the Acknowledgement in relation to the previous Incoming Message. The ability to identify Test Messages within a live system is supported. This parameter requires a single Nibble and is bit mapped as detailed below.

- Bit 0 – always 1, “This is an Acknowledgement” (see **Message Structure**)
- Bit 1 – Message Acknowledgement Flag
  - 0 = Message is not Acknowledged (Error in message)
  - 1 = Message is Acknowledged
- Bit 2 – Reserved
- Bit 3 – Live or Test/Evaluation Message
  - 0 = Live Message
  - 1 = Test Message
- Bits 4 – 7 – Reserved.

4.2.5 Acknowledgement Counter: The Acknowledgement Counter will be under the control of the Acknowledgement Sender and will provide a unique message reference. The Acknowledgement Counter shall be in the range; 0-65535 and will wrap to 0 (Zero) following terminal count increment.

4.2.6 Message Counter Reference: The Message Counter Reference is used by the Acknowledgement to reference the previous Incoming Message from either the PA or the Central Communication Server. This is a Mandatory parameter and will be in the range 0-65535.

4.2.7 Session PA Identifier : The Session PA Identifier parameter is derived from the Log On Response as a result of the Log On Request generated by the PA. For the initial Log On Request message, when this is clearly unavailable, this parameter shall be set to 0 (zero). This parameter requires two bytes in the range 1-65535 where 0 (zero) is reserved as described above.

4.2.8 Message Time Stamp: The Message Time Stamp parameter may be derived from the GPS’s NMEA 0183 representation of its Coordinate Universal Time (UTC) (if sent by the PA) or the server time (if sent by the server). This parameter will prove useful when network delays are encountered.

4.2.9 Error Number: This field is used to send an error code to indicate the reason for a negative acknowledgement. **The field is ignored for a positive acknowledgement** The following are some standard codes, however it is expected that application specific codes may be required.

Number Meaning

0 No error code.
1 Unknown Sender – e.g. SVID is unknown.
2 Unknown Running Board
3 Unknown Operator Code
4 Unknown PA ID
5 Unknown Service Code
6 Unknown Journey Number
7 Unknown Event Received
8 Unknown Event data sent with known Event
9 Config Information not supported
10 Serial No Information not supported
11 Duplicate PA ID
12 No SVID available
13 Corrupt Message Received
14 .. 127 Reserved codes
128..255 Implementation defined codes

4. MESSAGE STRUCTURES

4.1. OVERVIEW

5.1.1 As displayed in the table below, a number of messages have been defined, with the vast majority allocated a 'frozen' status. It was deemed by document contributors that these frozen messages were in a suitable state to be used.

5.1.2 In addition, each message has been assigned a category: either Principal or Variant. The Principal messages are those containing all of the relevant parameters for a particular message.

The Variant messages provide different parameter combinations, enabling users to choose which set of parameters is most appropriate for them. For example, in the case of both the Journey Details and Position Update messages 'basic' messages have been used as Variant messages where only the minimum number of essential parameters are included.

5.1.3 The following message types are currently defined.

Message	ID Message	Type Category	From	Message Status
138	Log On Request	Principal	PA	Frozen
11	Log Off Request	Principal	PA	Frozen
139	Pet Position	Principal	PA	Frozen
140	Error	Principal	PA	Frozen
132	Log On Request Resp	Principal	Server	Frozen
200	Pet Position Response	Principal	Server	Frozen
101	Error Response	Principal	Server	Frozen

4.2. MESSAGE ACKNOWLEDGEMENT

5.2.1 The structure of the Acknowledgement Message was defined.

5.2.2 A Message Acknowledgement may be sent by either the Server or the PA in response to a previously received Message where the associated Message Flags specifically request a response.

5.2.3 A Message Acknowledgement does not itself require an acknowledgement and thus the Acknowledgement Flags do not support this requirement.

4.3. LOG ON REQUEST – MESSAGE ID #138

5.3.1 Lhe Log On Request message relates to a PA logging on

This Message Type does not require an acknowledgement

Parameter	Mandatory/ Optional	Null Value Allowed?	Format	Length (bytes)

Message ID	Mandatory	No	No 138	1
PA ID	Mandatory	No	CCCCCCCCC *O-licence number*	9

5.3.2 Message ID: This parameter uniquely defines the message type. The Message ID requires a single byte capable of describing 255 (1-255) message types with 0 (Zero) being reserved.

5.3.3 PA ID: The PA ID parameter is derived from the Operator O-Licence number and forms part of a PA's unique identifier along with the PA ID parameter.

4.4. LOG ON RESPONSE – MESSAGE ID #132

This Message Type does not require an acknowledgement

Parameter	**Mandatory/ Optional**	**Null Value Allowed?**	**Format**	**Length (bytes)**
Message ID	Mandatory	No	132	1
Session PA Identifier	Mandatory	No	XX A returned value of 0 (zero) indicates an invalid Log on	2
Error Number	Mandatory	No	X See Acknowledgement Error code	1
Server Address	Optional	No	XXXX IP address of the server to which data is to be addressed rather than the default address	4
Server Port	Optional	No	UU IP Port on the server to send information to.	2

5.4.1 Message ID: This parameter uniquely defines the message type. The Message ID requires a single byte capable of describing 255 (1-255) message types with 0 (Zero) being reserved.

5.4.2 Session PA Identifier: The Session PA Identifier parameter is in response to the Log On Request message generated by the PA and managed by the Scheme centre. This parameter will be used in all subsequent messages as a means of identifying specific PAs. This parameter requires two bytes and the returned value will be in the range 1-65535. It must also be noted that multiple IP addresses can be used within a single session.

5.4.3 Error Number: Although this message is not an "acknowledgement", it is convenient to include an error field that uses the same conventions as acknowledgement errors (see Section 3.2.9).

5.4.4 Server Address: the address to which the server would prefer the mobile to send data. The intention of this is to allow the server to perform load balancing across the multiple servers. If the Server Address is omitted or is not supported by the PA then the original configured base address to which the log on requires was sent should continue to be used by the PA.

5.4.5 Server Port: the port on [Server Address] to which data is to be sent from the PA.

5.4.6 The SVID is allocated by the server on the first logon on request received. This should be reused for the same PA (as identified by PA id , operator id) on each subsequent logon request received before a logoff message from the PA is received or the server "times-out" the session id. The message counter should not be reset if the same PA requests a log on when an SVID is already allocated to it. The optional Serial Number can be used to validate the uniqueness of the PA requesting a connection and allows the server to reject duplicated PA IDs.

5.4.7 If this message is received by the PA and the SVID in the telegram is different to the SVID the PA currently has then the PA should reset the message counter and use the new SVID in all subsequent communications.

4.5. LOG OFF REQUEST – MESSAGE ID #11

This Message Type does require an acknowledgement

Parameter	Mandatory/ Optional	Null Value Allowed?	Format	Length (bytes)
Message ID	Mandatory	No	11	1
Session PA Identifier	Mandatory	NO	XX	2

5.5.1 Message ID: This parameter uniquely defines the message type. The Message ID requires a single byte capable of describing 255 (1-255) message types with 0 (Zero) being reserved.

5.5.2 Session PA Identifier : The Session PA Identifier parameter is derived from the Log On Response as a result of the Log On Request generated by the PA. This parameter requires

two bytes in the range 1-65535 where 0 (zero) is reserved as described above.

5.5.3 When this message is received at the control centre, this PA is to be recorded as logged off and no further communication is to be received against this SVID until some other login results in the SVID being reallocated. Receipt of a message using an unallocated SVID and requiring acknowledgement should result in the Server sending a NACK to the PA with the Error Number 1. The PA should then follow the log on process.

4.6. PET POSITIONS – MESSAGE ID #139

This Message Type does require an acknowledgement

Parameter	Mandatory/ Optional	Null Value Allowed?	Format (*Byte Aligned*)	Length (bytes)
Message ID	Mandatory	No	139	1
Lines Number	Mandatory	No	n	1
Lines	Mandatory	No	n	n

5.6.3 Message ID: This parameter uniquely defines the message type. The Message ID requires a single byte capable of describing 255 (1-255) message types with 0 (Zero) being reserved.

5.6.4 Lines Number: Number of lines for which the information is sought its buses

5.6.5 Lines List: Unique identifiers of lines requested.

4.7. PA ERROR (PA TO CENTRE) – MESSAGE ID #140

This Message Type does require an acknowledgement

Parameter	Mandatory/ Optional	Null Value Allowed?	Format	Length (bytes)
Message ID	Mandatory	No	140	1
Error ID	Mandatory	No	X	1

5.11.1 Message ID: This parameter uniquely defines the message type. The Message ID requires a single byte capable of describing 255 (1-255) message types with 0 (Zero) being reserved.

5.11.2 Error ID: Error identifier enabled on the PA

4.8. PET POSITION RESPONSE – MESSAGE ID #200

This Message Type does require an acknowledgement

Parameter	Mandatory/ Optional	Null Value Allowed?	Format	Length (bytes)
Message ID	Mandatory	No	200	1
Session PA Identifier	Mandatory	No	XX	2
Line ID	Mandatory	No	X	1
Vehicles Number	Mandatory	No	X	1
Vehicles CheckCond	Optional	No	X	4
Position – latitude	Mandatory	No	X	4
Position – longitude	Mandatory	No	X	4
Rx Time	Mandatory	No	X	4

5.13.1 Message ID: This parameter uniquely defines the message type. The Message ID requires a single byte capable of describing 255 (1-255) message types with 0 (Zero) being reserved.

5.13.2 Session PA Identifier : The Session PA Identifier parameter uniquely identifies the specific PA. This parameter requires two bytes and will be in the range 1-65535.

5.13.3 Line ID: Unique identifiers of line requested.

5.13.4 Vehicles Number: Number of vehicles assigned to that line.

5.13.5 Vehicles CheckCond: Vehicles Code check (not in use)

5.13.6 Position: The Position parameter is derived from WGS84 latitude and longitude values and, dependent on operator preferences, is calculated to varying degrees of accuracy.

5.13.5 Rx Time: The Message Time Stamp parameter shall be the GPS's NMEA 0183 representation of its Coordinate Universal Time (UTC). This parameter will prove useful when network delays are encountered.

4.9. LOG ON RESPONSE – MESSAGE ID #132

This Message Type does not require an acknowledgement

Parameter	Mandatory/ Optional	Null Value Allowed?	Format	Length (bytes)

Message ID	Mandatory	No	132	1
Session PA Identifier	Mandatory	No	XX A returned value of 0 (zero) indicates an invalid Log on	2
Error Number	Mandatory	No	X See Acknowledgement Error code	1
Server Address	Optional	No	XXXX IP address of the server to which data is to be addressed rather than the default address	4
Server Port	Optional	No	UU IP Port on the server to send information to.	2

5.4.1 Message ID: This parameter uniquely defines the message type. The Message ID requires a single byte capable of describing 255 (1-255) message types with 0 (Zero) being reserved.

5.4.2 Session PA Identifier: The Session PA Identifier parameter is in response to the Log On Request message generated by the PA and managed by the Scheme centre. This parameter will be used in all subsequent messages as a means of identifying specific PAs. This parameter requires two bytes and the returned value will be in the range 1-65535. It must also be noted that multiple IP addresses can be used within a single session.

5.4.3 Error Number: Although this message is not an “acknowledgement”, it is convenient to include an error field that uses the same conventions as acknowledgement errors (see Section 3.2.9).

5.4.4 Server Address: the address to which the server would prefer the mobile to send data. The intention of this is to allow the server to perform load balancing across the multiple servers. If the Server Address is omitted or is not supported by the PA then the original configured base address to which the log on requires was sent should continue to be used by the PA.

5.4.5 Server Port: the port on [Server Address] to which data is to be sent from the PA.

5.4.6 The SVID is allocated by the server on the first logon on request received. This should be

reused for the same PA (as identified by PA id , operator id) on each subsequent logon request received before a logoff message from the PA is received or the server "times-out" the session id. The message counter should not be reset if the same PA requests a log on when an SVID is already allocated to it. The optional Serial Number can be used to validate the uniqueness of the PA requesting a connection and allows the server to reject duplicated PA IDs.

5.4.7 If this message is received by the PA and the SVID in the telegram is different to the SVID the PA currently has then the PA should reset the message counter and use the new SVID in all subsequent communications.

4.10. ERROR RESPONSE (CENTRE TO PA) – MESSAGE ID #101

This Message Type does require an acknowledgement

Parameter	Mandatory/ Optional	Null Value Allowed?	Format	Length (bytes)
Message ID	Mandatory	No	101	1
Error ID	Mandatory	No	X	1

5.11.1 Message ID: This parameter uniquely defines the message type. The Message ID requires a single byte capable of describing 255 (1-255) message types with 0 (Zero) being reserved.
5.11.2 Error ID:

5. ANNEX I: SECURING COMMUNICATIONS

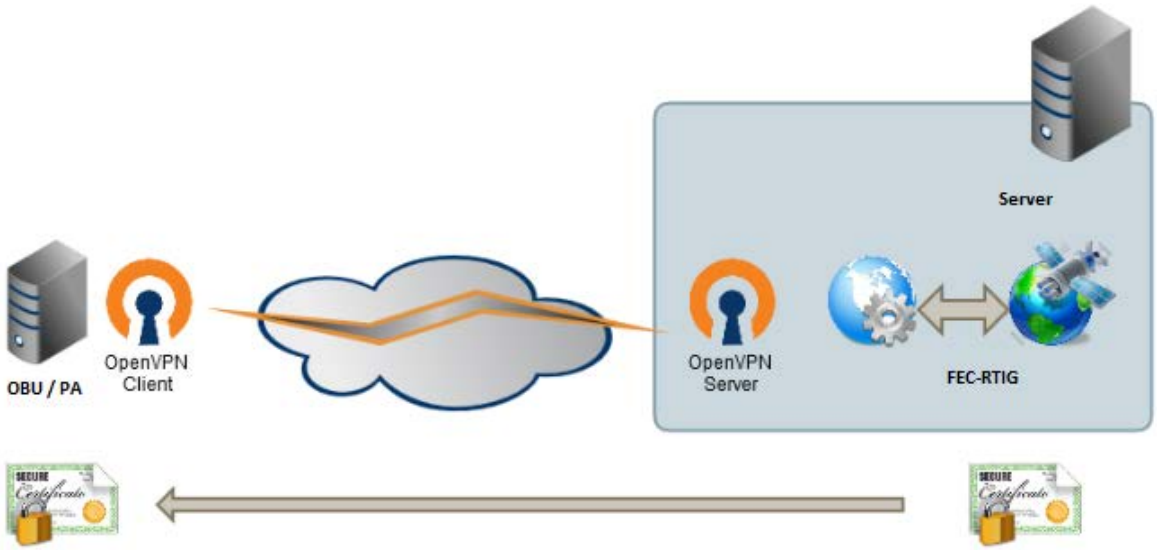
Since this protocol is an open format, a higher standard for secure communications is defined.

The proposal to establish a safe environment for open communication protocol would be based on the use of tunnels. For this terminator is installed in the tunnels based on OpenVPN server.

OpenVPN is a connectivity solution based on open source SSL VPN offering point-to-point connectivity. Both client and server software that is available for most current platforms, providing a high level of compatibility.

With respect to encryption and authentication, to increase security levels will use asymmetrical encryption SSL / TLS. These SSL / TLS libraries are part of the OpenSSL software and implements encryption and authentication mechanisms based on certificates. Thus, the server should always provide certificates to devices that want to use the open communication protocol.

The possible solution would have the following architecture:



Where on the server a new range of private IPs can be set up to establish such communications.

END OF DOCUMENT

