



Title	E.115-IP including encryption
Version	2.0
Date	20-09-2003
Status	Approved
Reference	E.115-IP-including-encryption-2.0
Classification	Unrestricted to EIDQ
Editor	Bas van Vliet, AffinIT B.V., b.vanvliet@affinit.nl
Number of pages	12

Document History

Version	Date	Brief Description
0.1	10-07-02	first draft
0.2		Second draft
0.3		Third draft
0.4		Fourth draft
1.0	19-09-02	Final version
2.0	20-09-03	Final version for ???
Changes Since Last Version		
None,		

1	Question.....	3
2	References	3
3	Background	3
3.1	Main Business reasons for migration towards IP-based networks	3
3.2	Requirements	3
3.2.1	Technical requirements	3
3.2.2	Operational requirements	4
3.3	Elected solution for TCP connections: principle	4
3.4	Comparison between E.115/X.25 and E.115/IP protocol	5
3.4.1	Implementation	5
3.4.2	Functionality	5
4	Detailed proposal.....	6
4.1	Overview.....	6
4.2	Service definition	6
4.2.1	Introduction.....	6
4.2.2	Connection establishment phase (socket connection)	7
4.2.3	Negotiation phase (application connection)	7
4.2.4	Data communication phase	7
4.2.5	Connection clearing phase	8
4.3	protocol description.....	9
4.3.1	Connection establishment phase	9
4.3.2	Negotiation phase.....	9
4.3.3	Data transfer phase.....	11
4.3.4	Connection clearing phase	12

1 QUESTION

How to implement E.115 protocol over TCP/IP stack instead of OSI stack and how to use XML and ASN.1 descriptions together in one protocol?

2 REFERENCES

- [1] E.115 access direct over TCP/IP (Version 1.0 - 08/2001)
- [2] E.115/IP using additional security (RC4) (Version 1.0 - 03/2002, Sorrento)
- [3] E.115 ITU-T Recommendation (02/1995)
- [4] RFC 793, Transmission Control Protocol
- [5] RFC 1321, MD5 Message-Digest Algorithm
- [6] RC4, Ron Rivest RSA Security

3 BACKGROUND

Strategically, two main aspects motivate the migration of the E.115 protocol on top of the TCP/IP stack:

- 1- Switch the directory traffic to IP-based networks (as X.25 technology becomes obsolete and very expensive)
- 2- It is expected that systems using UD will communicate via TCP/IP connections without use of the OSI layers. It is expected that the same network used for E.115 over TCP/IP will be used for UD, so adding TCP/IP to E.115 now can be thought of as an intermediate step evolving to UD.

This document does not go into the network types available, or if the Internet should be used compared to a virtual private network between two ROA's - that is left for whatever works out best between two ROA's

3.1 MAIN BUSINESS REASONS FOR MIGRATION TOWARDS IP-BASED NETWORKS

This proposal suggests a method of connecting E.115 systems together using TCP/IP networks, which is expected to give the following benefits as compared to X.25:-

- **Solution more open for new starters** as up-to-date and in phase with the current software market. X.25 OSI stacks, on which few competencies remain and no more valid maintenance exists, are not anymore required in favor of TCP/IP stacks today available on any systems. This situation limits the use of E.115 for directory exchanges, internationally and nationally.
- **Long life cycle:** indeed, X.25 becomes obsolete and major global carriers do not propose anymore X.25 service or increase their price to urge their customers to switch to IP-based networks, on which the whole market is being moved on.
- **Reduced network costs:** for IP connections, pricing policy is based on flat rate scheme whereas X.25 connections are famous for their high costs, including volume, duration and connections sensitive pricing scheme.
- **Reduced equipment costs:** X.25 cards, switches are much more expensive than Ethernet cards and routers.
- **XML solution more open for the market** in phase of description language for the E.115 protocol. With the current ASN.1 description language only a few competencies remain in the market. This situation limits the use of E.115 for directory exchanges, internationally and nationally. XML is currently the standard in the market for exchanging data. Today many tools are available on any systems. Also for new starters this is a big advantage to open the market.

3.2 REQUIREMENTS

3.2.1 Technical requirements

- **At all levels security:** Initiating phase MD5 and optional at data communication phase level RC4. With this type of security E.115 systems can be used in any environment (internationally, nationally, private and Internet networks).
- **Optional:** When both ROA's want to use data encryption between their environments, it must be bilaterally agreed and must be a configurable parameter to use encryption security functionality at data exchange level.

The following requirements should be fulfilled when migrate E.115 protocol on top of the TCP/IP stack.

- Network

The migration of E.115 over IP should enable the use of both network alternatives i.e. Internet network and private network (VPN's, Frame Relay, Leased Lines ...). The use of a data encryption algorithm must be completely independent from the type of network what has been used. Performance and security are enforced by using private networks.

- Security

When using IP-based networks, security becomes a critical item. The E.115/IP protocol should enable authentication and optional data encryption. Encryption of data at data exchange level can be needed for sensitive information in/or specific cases (use of E.115 for emergency services etc. or using E.115/IP over the public Internet), this mechanism has to be seen as a possible option and must be bilaterally agreed and which can also be used for International directory assistance services.

For authentication and encryption, standardized algorithms should be applied (MD5 and RC4).

Other additional security can be added by protecting the access to servers through firewalls giving possibility to restrict access for specific fixed calling IP addresses and port numbers.

- Connection orientation

The E.115/IP protocol should use a connection-oriented mechanism in order to optimize performance: all relevant information (requested service, inquiring ROA, authentication ...) are exchanged once at the connection establishment (as for E.115/X.25). This information is valid for the lifetime of the connection. At the opposite, in a connectionless mechanism, there is no end-to-end connection between ROA's: it requires that all relevant information should be sent with each data packet impacting heavily performance.

- Performance

Performance (processing time, response times) should be optimized and as close as possible to performance achieved with the E.115/X.25 protocol. The use of data encryption algorithm must NOT have any impact on the CPU of both system and the response time of an E.115 transaction.

3.2.2 Operational requirements

- Implementation

The implementation of the E.115/IP protocol and a data encryption algorithm should minimize the impacts in terms of software development and products integration.

- Interoperability

The interoperability should be achieved between products of different vendors.

- Deployment and migration

The use of the E.115./IP protocol will speed up the migration of directory traffic to IP-based. ROA connections may be based either on public networks (Internet) or private networks.

The use of E.115 service, well-known and mature, in a first step should ensure that potential problems in implementing this new protocol will not be related to the application level but to lower levels (network, transport ...). Once the IP migration will be achieved and the service stabilized in the frame of E.115, the migration of the application level itself (towards UD) will be achieved within a second step.

3.3 ELECTED SOLUTION FOR TCP CONNECTIONS: PRINCIPLE

This proposal concerns a direct mapping of E.115 over TCP/IP. The query-reply E.115 messages are unchanged, only the connection establishment phase is adapted to respond to TCP technology and OSI removal. This connection phase consists in:

1. a TCP connection phase
2. an application connection phase (negotiation) including: Inquiring ROA authentication, Type of requested service and negotiation phase protocol version (more flexibility for future evolutions)

This implementation implies modification of the existing X.25 based E.115 systems to remove the OSI stack completely, and use the TCP/IP stack instead.

This solution meets the requirements outlined above as it:

- allows E.115 systems to talk without the use of any parts of the OSI stack, over any standard TCP/IP networks, giving a much simpler connection between ROA's and less chance of configuration mistakes or incompatibility problems.
- optimizes the performance (and therefore is supposed to reach and improve the current quality of service provided by E.115/X.25 systems),
- uses only the basic TCP/IP stack available on all operating systems
- makes easy the implementation of a "light" client (especially for new starters)

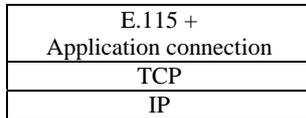


Fig1: Stack structure using direct mapping on TCP/IP

Note: Other alternatives, available to implement E.115 on an IP-based network, have been studied and discarded for the following reasons:

1. **XOT**: most of the major global carriers do not recommend and do not propose this option (heavy configuration, large overhead due to stacks accumulation)

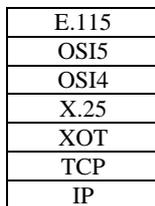


Fig2: Stack structure using XOT

2. **RFC1006**: still means use of OSI stack (limit spread of E.115 for new starters) and generates additional overhead as OSI layers are mapped onto TCP/IP

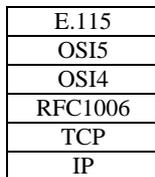


Fig3: Stack structure using RFC1006

3.4 COMPARISON BETWEEN E.115/X.25 AND E.115/IP PROTOCOL

3.4.1 Implementation

Layer	E.115/X.25 protocol	E.115/IP protocol
6/7 Presentation and application	E.115 Application	E.115 Application
5 Session	OSI session	Negotiation phase
4 Transport	OSI transport	
3 Network	X.25	TCP/IP
2 Link		
1 physical		

The Inquiry and Reply message formats used for E.115 are unchanged.

3.4.2 Functionality

Functionality	Parameter used within the old E.115/X.25 protocol	Parameter used within the new E.115/IP protocol

<i>Inquiring ROA identification</i>	Calling NUA (Network connection)	<i>Inquiring E.115/TCP identifier, Random Number, encrypted Random Number</i> fields, through an authentication mechanism Optionally: <ul style="list-style-type: none"> ▪ Application cross check with the calling TCP/IP address ▪ Firewall protection
<i>Requested service identification</i>	SSAP selectors (Session Connection)	<i>Requested service</i> field (Application connection)
Version Number	User Data (Session Connection)	Port Number (TCP connection)
Protocol Version	Not present	<i>Protocol version</i> field (Application connection)

4 DETAILED PROPOSAL

4.1 OVERVIEW

This part defines a profile of standardized TCP/IP protocols to realize the interconnection of International Directory Inquiry systems based on recommendation E.115. The starting point is based on the correspondence between a TCP Connection and an E.115 query request. The main features of the TCP/IP Profile defined for this Recommendation are outlined below and detailed in 4.2.

All inquiry messages from a ROA A to another ROA B and associated reply messages can be encrypted and transmitted over the same TCP Connection (permanent TCP Connection) established by the inquiring ROA. Indeed, for performance aspects, the TCP connection should be maintained and should not be released after each E.115 transaction. Reciprocally, all inquiry messages from the ROA B to the ROA A and associated reply messages can be encrypted and transmitted over a second TCP Connection, established by the inquiring ROA B.

To maintain high availability, each ROA may have duplicated its equipment for sending and receiving reply messages, and/or for receiving inquiry and sending reply messages (the equipment may have different network addresses). Additional TCP Connections to or from such equipment are allowed.

For parallel searches, multiple searches may be sent without the need to wait for replies on the connection. The replies may come back in a different order to the requests (due to different search times), so the E.115 field "Originating terminal code" should be used to direct replies back to the correct operator terminal.

The inquiring system is responsible for the TCP Connection release. The TCP Connection is closed after a certain amount of "inactivity time". This parameter is chosen by the TCP Service requestor. It should be maximum 15 minutes. After this amount of "inactivity time", if the connection is still pending, the replying E.115 application may close the TCP connection in order to free and optimize system resources.

In the case of using data encryption the inquiring and replying systems are both responsible for the TCP connection release. However the replying system may perform a TCP Connection release in the case of receiving a certain amount of E.115 transactions. It should be maximum 1000 transactions. After this amount of transactions the replying E.115 application may close the TCP connection in order to use a new MD5 key (every new session have a new and unpredictable key).

In the case of using XML description instead of ASN.1 description the inquiring and replying systems are both using XML as the specification language.

4.2 SERVICE DEFINITION

4.2.1 Introduction

When migrating E.115 from OSI to TCP/IP stacks, the main objective is to minimize the impact on existing E.115 applications in terms of implementation.

TCP is defined in RFC 793.

The use of TCP/IP services for E.115 interchanges can be described through four phases:

1. Connection establishment phase (standard TCP functionality)

2. Negotiation phase (specific functionality, requested service etc.)
3. Data transfer phase (standard TCP functionality, and optional data encryption (RC4))
4. Connection clearing phase (standard TCP functionality)

4.2.2 Connection establishment phase (socket connection)

This phase is standard TCP functionality. Before sending an E.115 request, a TCP connection should be established. The inquiring E.115/TCP application will initiate the establishment of a TCP Connection with this ROA by sending an active OPEN request specifying the local port and the foreign socket arguments (i.e. concatenation of an internet address identifying the TCP with a port identifier).

It is recommended to use the port number 3611 for providing the E.115 service. However, for more flexibility and to avoid potential port assignment problems, the E.115 port number should be configurable to allow bilateral changes between ROA's. Future directory usage will also be managed through different port numbers (to be defined or bilaterally agreed).

4.2.3 Negotiation phase (application connection)

4.2.3.1 Principle

This phase is specific functionality based on the TCP connection. Once the TCP connection becomes "established", additional information are interchanged on this connection between both inquiring and receiving E.115/TCP applications. This negotiation phase, based on a "four-way-handshake" mechanism, will be used to confirm the connection establishment phase and is necessary for:

1. determining the type of service requested
2. identifying and authenticating the inquiring E.115/TCP application

If the negotiation procedure failed for any reason, the connection is reset by the service provider.

4.2.3.2 Determining the type of service requested

Within the E.115/X.25 recommendation, the type of service requested is determined within the session establishment phase (through the use of specific SSAP). As it is no longer possible to transfer such information when establishing a TCP connection, the type of service requested will be determined through the negotiation mechanism.

4.2.3.3 Identifying and authenticating the inquiring E.115/TCP application

Within the E.115/X.25 recommendation, the identification and authentication of the inquiring E.115/TCP application is achieved when establishing the network connection (through the control of the calling NSAP). As the calling IP address may not be systematically known or secured when establishing a TCP connection, the identification and authentication of the inquiring E.115/TCP application will be achieved through the negotiation mechanism. In case the calling socket is known (bilaterally exchanged between inquiring and receiving ROA's), supplementary control (based on IP calling address and port number) may be achieved by the service provider.

As security is to be considered when IP networks are used, a mechanism able to authenticate communications is put in place (critical for billing aspects, etc ...). On the other hand, data privacy is not required in the frame of international directory assistance services but can be required in special circumstances, i.e. E.115 transactions themselves should not be encrypted but can be when necessary. Only the connection establishment should be secured and data security is optional.

The authentication mechanism is based on the MD5 algorithm and the data encryption mechanism is based on the RC4 algorithm. These procedures require both inquiring and receiving E.115/TCP applications to share a key (password): a secret piece of information that is used to encrypt a message. The major security problem is then to privately choosing and exchanging a key before communicating and to keep it in confidence.

4.2.4 Data communication phase

This phase is standard TCP functionality. The data format is based on specific fixed fields used to identify the E.115 messages followed by E.115 inquiry and reply fields as defined in E.115 recommendation [3]. The Inquiring E.115/TCP application indicates in each SEND call whether the data in that call should be immediately pushed through to the receiving user by setting the PUSH flag. If the PUSH flag is not set, the data may be combined with data from subsequent SENDs for transmission efficiency. It is recommended to use the push function after each E.115 request so that it could be transmitted promptly to the receiving E.115/TCP application.

If the connection has not been opened, the SEND is considered an error.

In the case that the optional data encryption is used and the connection and negotiation phases occur correctly, then both E.115 systems must create the RC4 key. For creating the key the MD5 algorithm is used. As input is used the password and random number ("password:random number"). This creates another unique 128-bit fingerprint as key for the RC4 encryption algorithm (in the negotiation phase "random number:password" was used as input). When using the same random-number and password (only in different order) then there is no additional key exchange necessary between ROA's. Both sides have to implement in the data communication phase the RC4 encrypting algorithm functionality.

4.2.5 Connection clearing phase

This phase is standard TCP functionality. The clearing phase is achieved either using a FIN or a RST flag within the TCP segment. The mechanism used may depend on each TCP stacks implementation.

4.2.5.1 Orderly release

Either side may request a disconnect at any time. It is normal for the requesting E.115 application to time-out the TCP connection if not used for a period of time (max of 15 minutes). If this is not done, the receiving E.115 application can close the connection.

4.2.5.2 Inquiring E.115/TCP application initiated abort

On detection of a serious problem and generally whenever a segment arrives which apparently is not intended for the current connection, the inquiring E.115/TCP application may issue a reset and connection goes to the CLOSED state.

4.2.5.3 Receiving E.115/TCP application (Service Provider) initiated abort

The Service Provider may issue a reset for any of a variety of reasons, for instance:

1. on detection of a serious local problem
2. whenever a segment arrives which apparently is not intended for the current connection
3. whenever the negotiation procedure has failed (see 4.2.3)
4. if the connection does not exist (CLOSED state), a reset is sent in response to any incoming segment except another reset

In these cases, the receiving E.115/TCP application issues a reset and connection remains or goes to the CLOSED state.

4.3 PROTOCOL DESCRIPTION

This paragraph describes the mechanisms used to provide the service defined above.

4.3.1 Connection establishment phase

The connection establishment is made possible if a passive OPEN command has been requested by the receiving E.115/TCP application. This passive command specifies that the connection establishment is to be passively waited for and leads to LISTEN for an incoming connection.

The connection becomes then "established" when sequence numbers have been synchronized in both directions (utilizing the synchronize -SYN- and acknowledge -ACK- control flags).

When receiving the "SYN" segment, the replying E.115/TCP application may send a positive answer or a negative answer. For example, the replying E.115/TCP application may reset an incoming connection initiation, in case the passive connection does not exist (CLOSED) on the receiving TCP side.

If there exists duplicated equipment, the inquiring application may set up an active TCP Connection to this equipment, even if it already has an active TCP Connection to this ROA.

Note: A passive open may have either a fully specified foreign socket to wait for a particular connection or an unspecified foreign socket to wait for any call. It is recommended that both mechanisms should be implemented. The fully specified foreign socket will be preferably used as it will be then possible to manage the incoming connections (restriction in terms of number of simultaneous connections, security aspects i.e. control of the calling TCP/IP address ...). However, as the foreign socket is not always known (for example if the connection is established through the Internet network), the receiving E.115/TCP application should be able to issue a passive OPEN request with an unspecified foreign socket in order to establish connection for unknown calling TCP/IP addresses.

Inquiring E.115/TCP

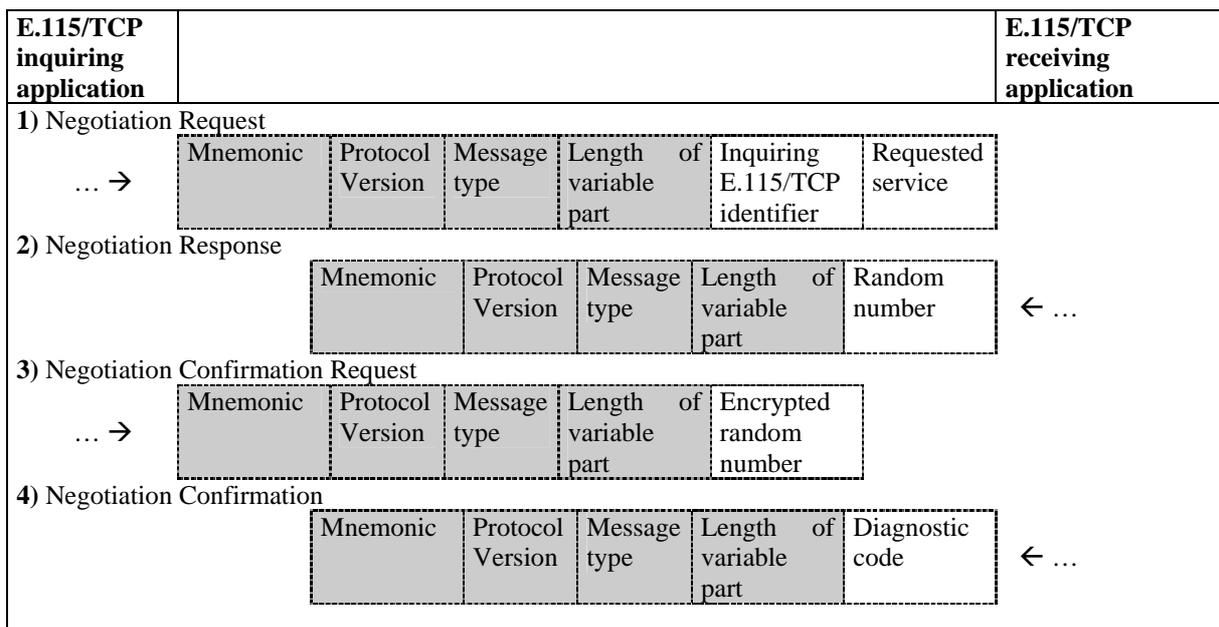
Receiving E.115/TCP

- | | | | | | | | |
|----------------|-----|-----------------------------------|--|--|--|--|--------------|
| 1. CLOSED | | | | | | | LISTEN |
| 2. SYN-SENT | --> | <SEQ=100><CTL=SYN> | | | | | SYN-RECEIVED |
| 3. ESTABLISHED | <-- | <SEQ=300><ACK=101><CTL=SYN,ACK> | | | | | SYN-RECEIVED |
| 4. ESTABLISHED | --> | <SEQ=101><ACK=301><CTL=ACK> | | | | | ESTABLISHED |
| 5. ESTABLISHED | --> | <SEQ=101><ACK=301><CTL=ACK><DATA> | | | | | ESTABLISHED |

Basic 3-Way Handshake for Connection Synchronisation

4.3.2 Negotiation phase

The negotiation phase is based on the following "4-way-handshake" mechanism:



If all stages of the above additional information interchange occurs correctly, the TCP connection is validated and E.115 transactions (inquiry/reply) can be transmitted through the connection (see 4.3.3). Else the connection is aborted after the last message exchange (stage 4), with reason for reset provided in the *Diagnostic code* field.

The fields format is as follows:

a) Fixed part:

The fixed-length header contains four fields and is handled by the E.115/TCP applications to determine Directory messages, protocol version, type of message and size of the variable part to be processed.

- *Mnemonic* –indicator identifying a message to the international inquiry service; 4 characters: Form: EIDQ
- *Protocol Version* –identifies the protocol version used; 4 characters: Form: 0100 (this version)
- *Message Type* – identifies the type of message; 2 characters:
From Diagram 4.3.2 “4 way handshake”
 - 1) ”Negotiation Request”; Form: NI
 - 2) ”Negotiation Response”; Form: NR
 - 3) ”Negotiation confirmation request” ; Form: CI
 - 4) ”Negotiation confirmation” ; Form: CR
- *Length of variable part* – identifies the length in bytes of the directory message; encoded within 2 bytes, the left most bit of the field being the most significant bit. For instance 2612 will be encoded 0x0A 0x34

b) Variable part:

- *Inquiring E.115/TCP identifier* – 8 characters: aligned from the left (if necessary, supplemented by spaces)
The *Inquiring E.115/TCP identifier* field is used by the receiving E.115/TCP application to associate a connection to an accessing ROA, manage the service specification associated, the billing aspects etc ... The value of this field is bilaterally agreed upon Inquiring and receiving ROA’s.

As an option, a cross check of this identifier against the calling TCP/IP address (when known) is sensible as extra check that this identifier is consistent with the network address of the inquiring ROA (see 4.3.1).

- *Requested service* – 8 characters: aligned from the left (if necessary, supplemented by spaces)
The type of requested service is used to manage a more precise service specification for a given inquiring E.115/TCP application.

The following codes have been defined:

- PUBLIC : Electronic directory service
- OPERATOR : Directory assistance service
- SECUREP : Secure Electronic directory service
- SECUREO : Secure Directory assistance service
- ASNNONPU : Electronic directory service (using ASN.1)
- ASNNONOP : Directory assistance service (using ASN.1)
- ASNSECPU : Secure Electronic directory service (using ASN.1)
- ASNSECOP : Secure Directory assistance service (using ASN.1)
- XMLNONPU : Electronic directory service (using XML)
- XMLNONOP : Directory assistance service (using XML)
- XMLSECPU : Secure Electronic directory service (using XML)
- XMLSECOP : Secure Directory assistance service (using XML)
- ...(*Others to be defined*)
- *Random number and Encrypted random number*
The structure and coding of this information format use the following ASN.1 notation:
 - *Random number*
 - 1010 0000
 - LENGTH
 - Random Number
 - *Encrypted random number*
 - 1010 0001
 - LENGTH
 - Encrypted Random Number

A Random number is generated by the receiving E.115/TCP application and is used to authenticate the *Inquiring E.115/TCP application as follows:*

1. After reception of the first message from the Inquiring E.115/TCP application, the E.115/TCP receiving application generates and sends a non-encrypted random number (arbitrary length and value).
2. The Inquiring E.115/TCP application encrypts the random number together with its secret key (*random number: password*), using MD5 algorithm, and sends the output encrypted number (a 128-bit "fingerprint").
3. The E.115/TCP receiving application encrypts the random number together with the secret key (*random number: password*) corresponding to the inquiring E.115/TCP application specified in the *Inquiring E.115/TCP identifier* field. The result is compared to the encrypted random number received within the third negotiation exchange, if it corresponds, authentication is done and E.115 transactions can be exchanged on the TCP connection, else the connection is aborted.

- *Diagnostic code* - 2 characters

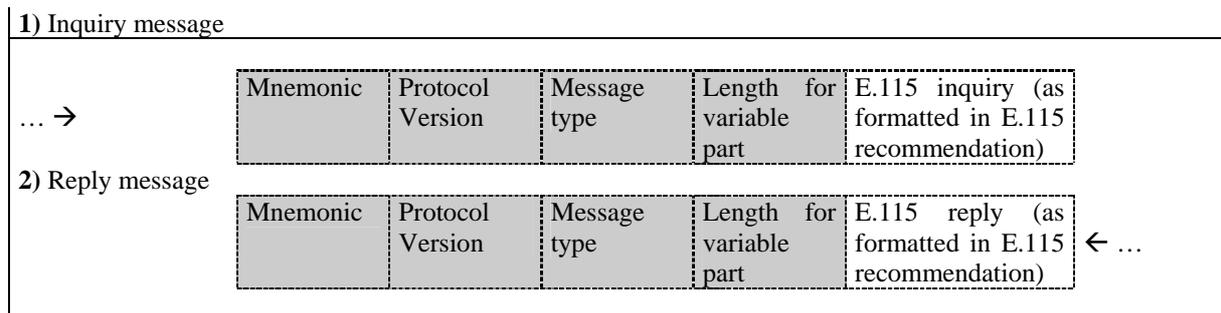
The diagnostic code is used by the receiving E.115/TCP application to qualify the service negotiation (during the fourth negotiation exchange)

The following codes have been defined:

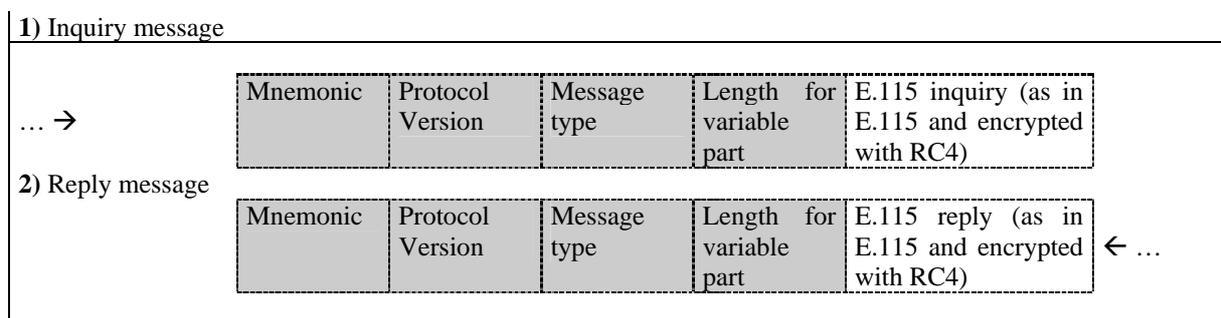
- 00: negotiation accepted
- 10: invalid authentication or unknown ROA (access not permitted)
- 20: protocol error or protocol version not supported
- 30: lack of resources (too much connections established)
- 40: requested service not supported or not bilaterally agreed
- ...(*Others to be defined*)

4.3.3 Data transfer phase

When connection and negotiation phases occurs correctly, E.115 transaction messages can be interchanged between both inquiring and receiving ROA's with the following formats:



Or



The field's format is as follows:

a) Fixed part:

The fixed-length header contains two fields and is used by the E.115/TCP applications to determine the beginning of the message and the size of the variable part to be processed.

- *Mnemonic* –indicator identifying a message to the international inquiry service; 4 characters: Form: EIDQ
- *Protocol Version* –identifies the protocol version used; 4 characters: Form: 0100 (this version)

- *Message Type* – identifies the type of message; 2 characters:
From Diagram 4.3.3 Data Communication Phase
 - 1) "Inquiry message"; Form: IM
 - 2) "Reply message"; Form: RM
 - *Length of variable part* – identify the length in bytes of the directory message; encoded within 2 bytes, the left most bit of the field being the most significant bit. For instance 2612 will be encoded 0x0A 0x34
 - *b) Variable part:*
 - *E.115 inquiry* – contains all E.115 inquiry fields as formatted and described in E.115 recommendation
 - *E.115 reply* – contains all E.115 reply fields as formatted and described in E.115 recommendation
- Or
- *E.115 inquiry* – contains all E.115 inquiry fields as formatted and described in E.115 recommendation and encrypted with the RC4 algorithm
 - *E.115 reply* – contains all E.115 reply fields as formatted and described in E.115 recommendation and encrypted with the RC4 algorithm

4.3.4 Connection clearing phase

The connection clearing phase consists in either:

- close the connection which leads to terminate gracefully the message flow
- reset the connection which leads to delete the ongoing messages

4.3.4.1 Closing Connection

When the decision to close the TCP Connection is taken, the inquiring E.115/TCP application initiates to CLOSE the connection

In this case, a FIN segment can be constructed and placed on the outgoing segment queue. No further SENDs from the user will be accepted by the inquiring TCP, and it enters the FIN-WAIT-1 state. RECEIVES are allowed in this state. All segments preceding and including FIN will be retransmitted until acknowledged. When the receiving TCP has both acknowledged the FIN and sent a FIN of its own, the inquiring TCP can ACK this FIN. Note that a TCP receiving a FIN will ACK but not send its own FIN until its user has CLOSED the connection also.

<i>Inquiring E.115/TCP</i>		<i>Receiving E.115/TCP</i>
1. ESTABLISHED		ESTABLISHED
2. (Close)		
FIN-WAIT-1 --> <SEQ=100><ACK=300><CTL=FIN,ACK>	-->	CLOSE-WAIT
3. FIN-WAIT-2 <-- <SEQ=300><ACK=101><CTL=ACK>	<--	CLOSE-WAIT
4.		(Close)
TIME-WAIT <-- <SEQ=300><ACK=101><CTL=FIN,ACK>	<--	LAST-ACK
5. TIME-WAIT --> <SEQ=101><ACK=301><CTL=ACK>	-->	CLOSED
6. (2 MSL)		
CLOSED		

Normal Close Sequence

The inquiring E.115/TCP application which CLOSEs may continue to RECEIVE until it is told that the other side has CLOSED also. Thus, a program could initiate several SENDs followed by a CLOSE, and then continue to RECEIVE until being signaled that the other side has closed, and then the inquiring E.115/TCP application can terminate its side gracefully.

After 15 minutes of inactivity, the receiving TCP can initiate to CLOSE the connection by sending a FIN control signal. The inquiring TCP will ACK it and tell the user that the connection is closing. The user will respond with a CLOSE, upon which the inquiring TCP can send a FIN to the receiving TCP after sending any remaining data. The inquiring TCP then waits until its own FIN is acknowledged whereupon it deletes the connection. If an ACK is not forthcoming, after the user timeout the connection is aborted and the user is told.

Note: both E.115/TCP applications (user) may close simultaneously. A simultaneous CLOSE by users at both ends of a connection causes FIN segments to be exchanged. When all segments preceding the FINs have been processed and acknowledged, each TCP can ACK the FIN it has received. Both will, upon receiving these ACKs, delete the connection.

4.3.4.2 Reset Connection

In that case the inquiring E.115/TCP application issues a reset and connection goes to the CLOSED state.

In the different cases enumerated in 4.2.5.3, the receiving E.115/TCP application issues a reset and connection remains or goes to the CLOSED state.