NPL Report CISE 4/95

December 1995

Issues in Secure Electronic Data Interchange (EDI)

Bronia Szczygiel
Centre for Information Systems Engineering
National Physical Laboratory
Teddington
Middlesex
United Kingdom
TW11 0LW

ABSTRACT

There is a growing requirement for secure electronic communication of information across national and market boundaries. The solution lies in the development of a secure EDI system based on international standards such as EDIFACT and X.435. This report details the issues uncovered whilst developing a Strict Conformance Test Suite for a secure EDI system. The test suite is based on the secure EDIFACT and X.435 standards and is published as an accompanying document. The findings show that secure EDIFACT is likely to provide a good solution whereas X.435 has significant weaknesses.

ISSN 1361-407X

National Physical Laboratory
Teddington, Middlesex, UK, TW11 0LW

Approved on behalf of the Managing Director of NPL,
by Mr A J Marks, Director, Centre for Information Systems Engineering

CONTENTS

INTRODUCTION

With the growing reliance on information technology in all aspects of business, there comes a growing requirement for the ability to exchange information relating to trade electronically. Since much of the information is commercially sensitive in some respect, it requires adequate protection. Hence the need for a secure electronic means of information exchange.

This electronic exchange of information is termed Electronic Data Interchange (EDI) and where the means of exchange provides protection against particular threats it is termed secure EDI.

As part of the current work programme on Strict Conformance Testing (SCT) the Data Security Group (DSG) has developed a SCT test suite for secure EDI [1].

The development of the test suite and the liaison with EDI experts which accompanied it have led to the identification of a number of issues associated with the progress of stable standards for secure EDI. These issues may lead to a delay in the uptake of the relevant standards by implementors.

This report highlights the findings of the research programme and recommends a way forward in some areas.

Section 2 introduces EDI and the main players for various aspects associated with EDI. Section 3 identifies the issues uncovered and section 4 contains recommendations for progress.

SECURE EDI

This section gives a brief introduction to EDI, secure EDI and the associated standards. The organisations involved in the development of standards and test suites for EDI are identified. In addition, information on implementations of EDI is given.

EDI

Organisations have been exchanging information electronically for many years. This has led to the establishment of several different proprietary EDI solutions across the world, for example, SWIFT in the financial market, TRADACOMS in retail and manufacturing in the UK. However, in recent years attempts have been made to establish common standards for EDI in order to facilitate the exchange of information across company, market sector and national boundaries.

A common solution relies heavily on the definition of a common syntax and data structure for information exchange. There are currently two major contenders in the standards arena namely, ANSI X.12 and UN EDIFACT, also published as an ISO

standard, [3]. X.12 is mostly used in the US and EDIFACT in Europe. The intention is for the two to converge in the long term. For the purposes of this report we will refer only to EDIFACT.

A common means of communicating the information is also required. Where communication is electronic, the preferred solution should be an Open Systems Interconnection (OSI) solution to maximise interoperability. Currently, the most promising underlying OSI communications protocol for carrying EDIFACT messages is X.400 [2]. However, Interactive EDI will require a more sophisticated protocol which could be OSI Transaction Processing (OSI TP) [9]. This report considers only batch EDI.

The X.400 series of standards includes a standard, X.435 [8], specifically designed to provide an interface between EDIFACT and X.400. The X.435 standard includes some additional security functionality and assumes no security in EDIFACT as it pre-dates the secure EDIFACT standard.

The SCT test suite developed by the DSG is based on the secure EDIFACT standards with additional tests for X.435.

EDIFACT

EDIFACT defines a syntax and structure for basic data exchange. There is no security functionality contained in the base standard. This is being addressed with the development of separate documents [4,5,6,7] detailing requirements, recommendations, additional syntax and structure and implementation guidelines for secure EDIFACT. There are two ways of supplying the security information required - either with a security header and trailer attached to each message or with a separate security message containing information relating to one or more interchanges. The latter may also be used for acknowledgement and is known as the AUTACK message. The AUTACK message obeys the construction rules for an EDIFACT message but has a specific security related function.

In order to provide security for EDIFACT messages the standard defines headers and trailers which contain fields for security parameters. The means used to generate values of these parameters is out of scope of the standard.

The secure EDIFACT series has a document [6] which details the requirements, the threats and the mechanisms which counter the threats. There are no details of potential vulnerabilities and we believe the list of threats is incomplete. There are some errors in the current version of the secure EDIFACT standard but a new version is due to be published shortly.

X.435

The X.400 series of standards provides an Open Systems messaging service which is modelled as a communicating set of entities providing different messaging functions. At the user interface there is a User Agent (UA) which communicates with the underlying Message Transfer System (MTS) to provide end-to-end message handling. Messages may also be stored in the Message Store (MS) or accessed via the Access Unit (AU). The security aspects required for running EDIFACT over

X.400 are met by X.435, which defines additional requirements and protocol for the UA, MS and AU.

The X.400 series of standards contain details of the requirements, the threats and the mechanisms which counter the threats but the information is dispersed throughout several documents and is not coherent. There are no details of potential vulnerabilities and we believe the list of threats is incomplete. There are some errors in the standards.

ORGANISATIONS INVOLVED IN EDI

There are many international organisations involved in the development of standards and guidance documents for EDI. The UN is developing the EDIFACT standards and CCITT publishes the X.400 series of standards, all of which are also available as ISO/IEC standards.

Within the UN, a committee has been established to develop the EDIFACT set of standards. The decision to work outside of ISO was deliberately taken in an attempt to achieve a common solution as quickly as possible. The UN organisation works through a network of Regional Boards who are developing standard messages using the EDIFACT syntax. Each regional board has a sub-structure with groups representing different market sectors. Common messages are developed within the Trade group and sector specific ones within the relevant sector group.

Whilst the work of the above organisations is well known in the EDI world, there are other significant contributions coming from less well known organisations within Europe. One of these, the European Workshop on Open Systems (EWOS), has an Expert Group in EDI (EG-EDI). The group studies issues concerning EDI and liaises with other organisations in the field. They have recently completed the specification of a formal notation for describing EDI messages and are currently collecting information on the use of Interactive EDI (I-EDI). In addition, they have produced guidelines for other EWOS EGs on the use of EDI and are currently looking at issues associated with secure EDI in the light of the findings of the EC TEDIS project.

A representative of the DSG has attended meetings of EG-EDI. These meetings have been a valuable source of information and have led, directly or indirectly, to much of the material in this report.

Within the UK, the EDI Association (EDIA) is working to promote EDI and to educate potential users. The Association publishes EDI material, runs training sessions and provides a forum for users to share experiences and benefit from the knowledge of others. It is also actively involved in EDI standards making and related activities (for example, within EWOS EG-EDI). There are similar organisations in other countries around the world.

IMPLEMENTATIONS OF SECURE EDIFACT

It is clear from the EDI report published by UNICOM [1] that there are many EDIFACT translators on the market. The report contains details of various companies around the world who manufacture translators.

The DSG made contact with approximately forty manufacturers to discuss the possibility of arranging a test bed for the test suite under development. The replies received to date indicate that no-one is currently implementing secure EDI but some intend to. The most likely configuration will probably have a clear separation between the translator and the module providing the security algorithms. There are already several implementations on the market which claim to meet the requirements for the security algorithms used by secure EDIFACT.

Most manufacturers implement the EDIFACT translator as a stand alone module which communicates with applications through a separate API. The API provides the sorting and flattening of data between the application and the translator. The translator therefore appears to take in a flat ASCII file of data for building the EDIFACT message. The use of mapping tables within the translator could imply some translation of text to the identifiers required by EDIFACT.

TESTING OF EDI

There are currently no known conformance testing services for EDI in operation in Europe. However, two conformance test suites were developed under the European Conformance Testing Services (CTS) project, one for EDIFACT and one for X.435. These have been considered by several organisations (including British Telecom) for use in a conformance testing service. As yet no service has been launched.

The CTS EDIFACT test suite contains syntactic tests for EDIFACT without security. There are no tests at all for security headers and trailers and none for AUTACK. The test suite requires the use of a newly defined format, ALF, for specifying the input files to the translator. This effectively adds another layer of translation and conformance requirements to the implementation which may make it difficult to use.

The existing X.435 test suite contains syntactic tests for security. There are no semantic tests for security and very few invalid behaviour tests. The datatyping part of the test suite can be re-used in drawing up a SCT test suite but there are many errors which need correction.

In designing the SCT test suite under the current programme, the DSG has assumed a separation between the module containing the security algorithms and the translator as described in section 2.3. This makes testing easier and reflects the most common approach envisaged by developers. If the security algorithms are built into the translator, the DSG SCT test suite will require significant alterations.

Traditional conformance testing relies upon the definition and use of an Implementation Conformance Statement (ICS) which is completed by the developer and contains implementation details which are required for testing. The various options from the standard which have been implemented are detailed and the ICS forms the basis of the static conformance review. In developing the SCT test suite for

secure EDI, it has become clear that an extended form of ICS is required which gives details of the security functionality implemented. This can then be used to parameterise the test suite. The relationship between the ICS and the security policy has not yet been fully determined but it is clear that there are strong links.

ISSUES

As a result of the DSG work in developing a test suite for secure EDI and liaising with organisations involved in standards and guidelines for EDI, a number of issues have surfaced which could hold back the development of secure EDI implementations.

DEFINITIONS

There are a number of problems associated with missing or overlapping definitions in the EDI field.

API definition

A typical EDI architecture (see figure 1) will require a user application, an EDIFACT translator and some underlying communications network. The application will produce some message which must be translated into a UN Standard Message (UNSM) and into EDIFACT syntax and structure. It is likely that the application will not produce an output file in the form required by the translator. An Application Programming Interface (API) is therefore required.

There is no standard definition of the form of the API or its functionality. Therefore all implementations are likely to be different. This leads to difficulties in testing of those elements covered by standards as discussed under section 3.3. The multiplicity of formats allowed for the output of the API and the input of an EDIFACT translator leads to unnecessary multiplicity of solutions. If one standard format were laid down for the input to a translator it would be possible to design a single API per application package which would interface to any translator. The EDIA are considering the requirements for a standard API for EDI but more effort is required.

X.435 Interface definition

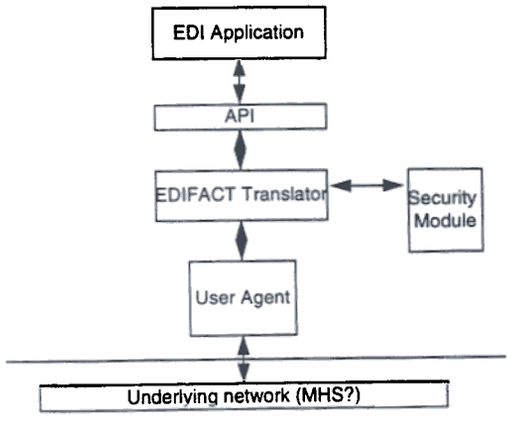
Similarly, there is no clear definition of the interface between the EDIFACT translator and the underlying network. For example, when interfacing to X.435 there are many parameters required but the only definition is in the Abstract Service of the User Agent, as defined in F.435, for which there are no conformance requirements.

Parameter definitions

Looking more closely at the definition of parameters within the various standards, it is clear that overlap exists. For example, with sender and recipient identifiers.

EDIFACT defines identifiers within the standard header. There is another definition within the security header which fulfils a security function and is therefore a tighter definition. Careful definition of the original identifiers could have avoided this redundancy. The solution would be to persuade the standards makers to adopt a more rigorous definition in the base standard. There are likely to be other instances of redundancy between the different parts of the standard. These need to be identified and solutions proposed in order to achieve maximum clarity and efficiency.

Figure 1 An EDI Architecture



Profile definitions

There is also overlap in the security functionality provided by the different standards which support secure EDI. Many of the threats which secure EDIFACT claims to address are also addressed by X.435. Unless additional information is to be protected there will obviously be some redundancy if both solutions are implemented. Clear definitions of secure EDI profiles are required in order to avoid inefficient solutions.

In particular, profiles which are of interest to commerce are required. This implies a profile which addresses security across the whole business application not just within the IT implementation. This approach stretches the boundaries of the current approach to specification and testing of IT security. The feasibility of such an approach is currently being studied within the UK using a bought ledger application as a working example.

The definition of a secure EDI profile within the IT domain is discussed in more detail in section 4. A technique for describing EDIFACT message profiles is discussed in the following section

Description of EDIFACT message profiles

UN Standard Messages are generally designed to meet the requirements of a broad market and, as such, are too general to be implemented directly. Subsets are therefore developed to meet more specific business requirements and these are recorded in Message Implementation Guideline documents (MIGs). However, the specification of these subsets using natural language leads to errors and ambiguities and to difficulties in designing translators. For this reason, the EWOS EG-EDI group has supported the development of a formal description technique to be used for the definition of EDI message profiles. The language is intended to be processable thereby facilitating the handling of messages by translators and the testing of implementations for conformance to the specification.

The technique is newly developed and therefore has not yet been widely used for real examples. Existing MIGs should be specified using the technique in order to gain the full benefit. Examples of useful messages could be identified in any number of areas (for example, invoicing) and the results would be useful across many market sectors. Once message specifications have been drawn up, the implications for testing can be investigated using real examples.

LACK OF FUNCTIONALITY

Although the EDIFACT standards contain a carefully defined syntax and structure for secure EDI messages, there is no indication of the action to be taken on any of the fields. In some cases this will be implementation dependent but in others there is a need for some definition of functionality if the parameters used in the standard message are to be of any use. This is particularly true of the security parameters. It is difficult to see the point of including a security value if an implementation never checks that value and if there is no standard definition of the action to be taken should some security check fail. If the required functionality is seen as belonging within the application then so should the parameters. The problem can be resolved by defining standard functionality for an EDIFACT translator as discussed in section 3.3.

CONFORMANCE REQUIREMENTS

In general, standards in the Open Systems field have a clear definition of conformance requirements against which an implementation can be tested. This is true of X.435 but not of EDIFACT. It is also generally a problem with the majority of security standards. In the absence of clear conformance requirements it is difficult to assess whether or not an implementation conforms to the claimed standard. A standard without conformance requirements is no more than a guideline document.

The EDIFACT standard lays down a structure and syntax for message specification. This then clearly defines the required output of any EDI translator. However, there is no definition of the expected format of the input or of the expected behaviour of the translator in any particular event of interest (for example, when an error occurs). This is outside the scope of the EDIFACT standards but presents a problem when attempting to assess compliance of translators. There is currently a sector specific approach to the design and implementation of translators reflecting the different application packages used. This clearly leads to an uneconomical solution to the problem.

There is therefore a requirement for a set of generic conformance requirements for EDIFACT translators. The absence of such requirements has led to some problems in defining a conformance test suite.

The CTS project chose to specify the ALF format for input files (see section 2.3) to which an implementation would have to conform. The DSG has used a different approach which leaves the specification of the input file in an abstract form which can be fully determined as part of the production of an executable test suite.

In addition, the DSG test suite requires the implementation to indicate error conditions in some way, which is again left at an abstract level.

A related problem arose with the definition of error behaviour for X.435. Although there is some error behaviour defined for submission of messages, the standard clearly states that no error checking is done on receipt. In the usual manner of standards no explanation is given for this rather peculiar arrangement. It does not appear to make sense and therefore the test suite developed by the DSG assumes that some error checking is required of an implementation in order that security requirements can be met.

SOLUTIONS WITHIN UK GOVERNMENT

Secure EDI is of interest to a number of UK government departments, including the NHS, DSS, Customs and Excise, Home Office and MoD. There have already been several attempts at demonstrating a secure EDI solution. Each department is uncovering the problems identified within this report and trying to find a solution that meets their needs. No-one has yet succeeded. The result will eventually be a diversity of solutions which could lead to problems later.

A single well directed attempt at resolving the identified problems is required. This should lead to a Commercial Off The Shelf (COTS) solution which will appeal to a wide market. The existing standards offer a good basis from which to work but more effort is required.

A SECURE EDI ARCHITECTURE

The following threats to EDIFACT interchanges are assumed to be present in the environment:

Loss, deletion, addition, duplication, replay or re-ordering of messages;
Modification of message contents;
Masquerade by interchange sender;
Repudiation of message origination;
Repudiation of message receipt;
Sender wrongly claims authorship of a message;
Two or more individuals claim to be legitimate (temporary) owners of a message;
One party refuses to complete a transaction;
Disclosure of the identity of the sender;
Disclosure of message contents.

A secure implementation of an EDI messaging service must therefore provide the following security services to counter these threats:

message content integrity (MCI)	non-repudiation of origin (NRO)
message sequence integrity (MSI)	non-repudiation of receipt (NRR)
message origin authentication (MOA)	access control (AC)
peer entity authentication (PEA)	confidentiality (C)

Although audit services may not be provided by the EDI application, they are considered almost essential for two reasons; firstly, in case countermeasures fail and, secondly, to provide evidence of the operations that have taken place.

The TEDIS project [10] defined a number of new security services, which it was claimed were needed for secure EDI business transactions. These were:

- Claim of origin: The ability to prove that an organisation was the author of a particular document.
- Claim of ownership: The ability to demonstrate ownership of a particular electronic document (e.g. bill of lading) at a particular point in time.
- Fair exchange of values: The ability to ensure that no party can pull out of a transaction when the others have committed themselves.

Anonymity: The ability to carry out a transaction anonymously (c.f. digital cash).

Examination of these services, and the possible mechanisms to implement them, leads us to believe that the first three are not EDI issues as such, but must be provided by the supporting infrastructure (e.g. the network protocol carrying the messages). The last security service, anonymity, can be provided at the EDI level. Tests are included in the SCT test suite to ascertain whether this service is provided.

Figure 1 illustrates a possible architecture for a simple EDI solution. However, hidden within the simplicity is a complex problem, namely where to put the various security services. The solution depends to a large extent upon the environment in which the system is to work and the threats within that environment. The distribution of processing capabilities across a network leads to an increase in vulnerabilities and therefore an increased need for security functionality at each node of the network. Ensuring adequate security without overlapping functionality is the primary aim. It should be possible to identify a finite, manageable number of security profiles which meet this aim. It should also be possible then to parameterise the SCT test suite developed under this work programme to match the profiles.

Table 1 shows the security services provided within the EDI stack and shows how the services provided by secure EDIFACT and X.435 relate to each other. The findings are discussed in more detail in the remainder of this section.

Table 1. Security services within the EDI system

	MCI	*MSI*	*C*	*AC*	*MOA*	*PEA*	*NRO*	*NRR*
Application	1	1	1	1	1	1	1	1
Translator	2	4	-	9	5	5	6	6
X.435 UA	2	4	7	9	5	5	6	6
Network	3		8					

In this table, the term Application refers to the proprietary EDI application and the associated API. The term Translator refers to the secure EDIFACT translation software.

The fields within the table are interpreted as follows:

1- If the application is remote from the translator, in any sense, the data from the application may need full protection at the application level. Some form of providing a trusted path between the application and the translator must be considered.

2- Secure EDIFACT protects the content but not the standard headers and trailers of an EDI message. X.435 on the other hand offers protection of the full interchange and therefore is a more complete solution, however if there is danger of attack between the translator and the X.435 implementation additional security functionality would have to be added to the translator.

3- The underlying network may offer additional protection in the form of parity checking or the like.

4- Message sequence integrity within secure EDIFACT is at the individual message level with no guarantee of interchange protection, although it can be achieved. Within X.435 protection is at the interchange level with no individual message protection.

5- Authentication at the secure EDIFACT level is user to user, whereas at the X.435 level it is UA to UA. Therefore, the secure EDIFACT solution alone is adequate whereas the X.435 solution would need additional functionality at a higher level.

6- The services provided by secure EDIFACT and X.435 are the same and therefore either can be used but both are not necessary. Secure EDIFACT is the preferred option as it is higher in the stack.

7- Secure EDIFACT does not currently offer confidentiality. For the moment then, the confidentiality service would have to be provided by encryption at the X.435 level.

8- The underlying network may add to the confidentiality service through the provision of traffic padding or routing control.

9- Secure EDIFACT provides a certificate detailing user authorisation levels, privileges etc. There is no indication of how this can be used. X.435 offers additional access management through the exchange of message tokens. The use of one or the other will depend upon the kind of access control required.

In the following sections we expand on these comments on a layer by layer basis.

APPLICATION SOFTWARE

The data that leaves the application is the data which should arrive at the intended destination. Clearly, integrity of that data is a prime concern. The underlying network must not corrupt the data in any way. Although the underlying layers may provide a message integrity service, that service is directed at providing end-to-end integrity between peer layers, for example, between the local and remote translators. Once the protection is in place the underlying layers cannot corrupt the data however this does not protect against corruption within the higher layers. For example, if content integrity is provided by the translator this would not prevent the translator itself or an attacker on the connection between the two corrupting the data. There is therefore a requirement for some security functionality at the boundary between the application and the translator if an insecure network connects the two. This may be within the API if it is co-located with the application.

Similarly, for the other security services, there is some requirement for security at the application level and on any insecure network connection. For example, secure EDIFACT provides user to user peer entity authentication but this will depend upon some form of access control at the application level so that an impostor cannot access the system at that level.

The general case appears to be that if the translator is not co-located with the application software and the network connection between the two is insecure, additional security measures have to be taken in order that the security services provided by the translator are not compromised. It may be sufficient to use an encryption/decryption system on the network connection (decrypting before input to the local implementation).

In addition, access control measures and audit trails should be used on the application system.

EDIFACT TRANSLATOR

A translator which implements secure EDIFACT is probably the most efficient way to provide the security services required. It is however not complete.

Secure EDIFACT protects the integrity of the message content and, optionally, the secure header (USH) and trailer (UST). It does not protect the standard header (UNH) or trailer (UNT). This can lead to a vulnerability if sufficient care is not taken within the translator to ensure consistency between the normal set and the secure set. For example, information on the sender and recipient identity is contained within both the security header (USH) and the standard header (UNH). An attacker cannot modify the USH content if it is protected but could modify the UNH content. If this information is used for any purpose and a comparison is not made with the USH copy of the information there would be a vulnerability. The simplest solution would be to ensure that implementations do not use information from the UNH or UNT elements that are copied and protected in USH or UST. Alternatively, modifications would have to be made to the secure EDIFACT standard to allow for protection of the UNH and UNT or X.435 would have to be used to give full protection.

Secure EDIFACT provides message sequence integrity at the individual message level. The use of timestamps will ensure correct ordering of messages and interchanges but it may be difficult to detect when a message in a sequence has been deleted or inserted. If sequence numbers are used deletion or insertion of messages within the sequence can be detected, but the sequencing must span interchanges in order to achieve interchange sequence integrity. Even then, depending on the numbering system, it may be difficult to detect when the last message of an interchange or indeed the last interchange has been deleted.

The CIPHER facility planned for secure EDIFACT will provide confidentiality. It is not however available at the moment for scrutiny and therefore cannot be considered further in this report.

Message origin authentication is achieved through the use of a message authentication code and digital signature. This is sufficient for secure EDI.

Peer entity authentication is not one of the services which secure EDIFACT claims to be provide. It can however be achieved through the exchange of AUTACK messages containing a timestamp signed using a secret key. This exchange can be performed prior to the transmission of any messages. The use of timestamps and digital signatures within individual secure EDIFACT messages will serve to continually authenticate the peers.

Digital signature techniques are also used to provide non-repudiation of origin and receipt. The latter through the use of the AUTACK message.

In conclusion then, secure EDIFACT currently provides sufficient security functionality to achieve all the required security services except confidentiality and complete data integrity.

USER AGENT

An X.435 conformant User Agent should be able to supply all the required security services to the EDI application without an implementation of secure EDIFACT. However, there are gaps and there is also a problem associated with the probable location of the X.435 implementation.

The User Agent may not be located in the same system as the EDIFACT translator and the application. Unless the higher level network connection is guaranteed to be secure there is a problem with vulnerabilities above the X.435 level. For example, although adequate functionality exists within X.435 to address non-repudiation of origin and receipt it may be possible for an attacker with access to the network above the UA to falsify the evidence.

The message sequence integrity offered is actually interchange sequence integrity which means that an attacker could alter the order of messages within an interchange without detection.

Authentication is at the User Agent level rather than the user which means that there is no distinction between different users at the same UA. One user can therefore masquerade as another at the same UA. There is therefore a requirement for security functionality higher in the stack if X.435 is used.

In conclusion then, X.435 can offer most of the required security services but the authentication and message sequence integrity services may be inadequate. The potential for an attack on the local network above the UA may also be a problem.

UNDERLYING NETWORK

The underlying network may offer additional protection in the way of traffic padding, routing control, error correction and parity checking. It may however, also be a source of problems in terms of delaying messages which are time critical. The network performance would need careful consideration in order to avoid such problems.

CONCLUSION

Performing secure electronic interchange of information is of prime concern to a growing commercial community. There are a number of stable international standards which aim to achieve secure EDI. However, there are still a number of issues to be resolved.

The standards contain errors, ambiguities and omissions and there is a significant amount of overlap between different standards. There is a lack of clearly defined standardised functionality for various elements in the secure EDI system.

This report has, however, demonstrated that secure EDIFACT has the potential to deliver a secure EDI solution if the issues identified are resolved.

ACKNOWLEDGEMENTS

This work forms part of the NPL programme on Strict Conformance Testing and is jointly sponsored by the Telecommunications Division of the Department of Trade and Industry and the Ministry of Defence through the Defence Reasearch Agency.

REFERENCES

[1] Lampard RP, Hehir S, Szczygiel BM. *SCT test suite for Secure EDI.* NPL report to be published.

[2] CCITT. *Data Communications Networks - Message Handling Systems.* X.400 1988.

[3] International Organisation for Standardisation. *Electronic Data Interchange for Administration, Commerce and Transport (EDIFACT).* ISO 9735

[4] UN/EDIFACT Security JWG. *EDIFACT Security Implementation Guidelines* December 1993.

[5] UN/EDIFACT Security JWG. *Secure authentication and acknowledgement message.* Version D, March 1994.

[6] UN/EDIFACT Security JWG. *Recommendations for UN EDIFACT message level security.* December 1993.

[7] UN/EDIFACT Security JWG. *MIG Handbook - UN/EDIFACT Message AUTACK.* Version 4, February 1994.

[8] CCITT. *Message Handling Systems: Electronic Data Interchange Messaging System.* Recommendation X.435.

[9] International Organisation for Standardisation. *Information Technology - Open Systems Interconnection - Distributed Transaction Processing.* ISO/IEC 10026.

[10] Cryptomathic & MBLE. *Security in Open Environments.* Deliverable B7,TEDIS II Programme.