Internet Engineering Task Force (IETF)
Request for Comments: 5941
Category: Informational
ISSN: 2070-1721

D. M’Raihi
VeriSign
S. Boeyen
Entrust
M. Grandcolas
Grandcolas Consulting, LLC
S. Bajaj
VeriSign

August 2010

Sharing Transaction Fraud Data

Abstract

This document describes a document format for exchanging transaction fraud (Thraud) information. It extends the Incident Handling Working Group (INCH WG) Incident Object Description Exchange Format (IODEF) incident reporting document format.

Status of This Memo

This document is not an Internet Standards Track specification; it is published for informational purposes.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Not all documents approved by the IESG are a candidate for any level of Internet Standard; see Section 2 of RFC 5741.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at http://www.rfc-editor.org/info/rfc5941.

Copyright Notice

Copyright (c) 2010 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction

Financial organizations and merchants that offer online access to their services frequently encounter fraud perpetrated against their customers’ accounts. In their attempts to combat these frauds, the organizations and their law enforcement agencies could benefit greatly by sharing intelligence about fraud incidents and patterns with similar organizations and agencies. This specification standardizes a document format by which they can share such information. It is intended to facilitate multi-vendor interoperability between conformant components of an open fraud reporting framework.

Information sharing can take place directly between financial organizations and merchants. However, the power of shared intelligence is multiplied many times if the information is gathered from multiple sources by a shared network, consolidated, and redistributed to participants.

In this arrangement, incident reports submitted to the network are called "inbound reports", and reports issued by the network are called "outbound reports".

Inbound reports will be submitted using a push-style protocol (such as email or the Simple Object Access Protocol (SOAP)). Outbound reports will be distributed using either a push-style protocol or a request/response protocol (such as HTTP).

Inbound reports identify the contributor of the report, as this information is essential in evaluating the quality of the information it contains and in contacting the source for the purpose of clarification. However, outbound reports commonly do not identify the original sources, as those sources may not wish to be identified to other subscribers. Such reports should, instead, identify the consolidator as the source.

A report may describe a particular transaction that is known to be, or believed to be, fraudulent, or it may describe a pattern of behavior that is believed to be indicative of fraud. The former type of report is called an "activity report" and the latter a "signature report".

The schema defined herein extends the IODEF XML incident reporting schema [RFC5070].

In Section 3, we introduce the actors in a typical transaction fraud. Fraud reporting by means of an IODEF-Document is described in Section 4. We define the elements of a Thraud Report in Section 5.

In Section 6, we describe the Activity Thraud Report profile of the IODEF specification. In Section 7, the profile for a Signature Thraud Report is described. In Section 8, we define new attribute values for the IODEF Incident class. Security considerations are described in Section 9. Section 10 contains IANA considerations regarding the registration of the associated media sub-type and XML namespace identifier. The Appendices contain the complete XML schema and a sample Thraud Report.

Data elements in this document are expressed in Unified Modeling Language (UML) syntax [UML].

XML namespace prefixes are used throughout this document to stand for their respective XML namespaces, as follows.

iodef:	urn:ietf:params:xml:ns:iodef-1.0
thraud:	urn:ietf:params:xml:ns:thraud-1.0
xs:	http://www.w3.org/2001/XMLSchema
xsi:	http://www.w3.org/2001/XMLSchema-instance

2. Requirements Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

3. Anatomy of a Transaction Fraud

The actors in a typical transaction fraud are shown in Figure 1.

```
    +--------------------------------------+
    |             Fraudsters               |
    | (collect & verify victim credentials |
    |   via phishing, malware, etc.)       |
    +--------------------------------------+
         |
         |recruit
         |
         |   ----------------disburse profits-----------------
         |   |                                               |
         v   v                                               |
    +-----------+                   +--------------+     +-------+
    |           |                   |              |     | Fraud |
    |           |--Open Dest Acct-->|  Financial   |---->| Dest. |
    |           |                   | Organization |     |Account|
    |   Fraud   |                   +--------------+     +-------+
    | Executors |                          ^ funds
    |           |                          | transfer
    |           |                   +--------------+     +-------+
    |           |                   |   Victim’s   |     |       |
    |           |---Init Transfer-->|  Financial   |<-o--|Victim |
    |           |                   | Organization |  |  |Account|
    +-----------+                   +--------------+  |  +-------+
                                                      v
                                                +-----------+
                                                |   Fraud   |
                                                | Detection |
                                                |  Sensors  |
                                                |(realtime/ |
                                                |  offline) |
                                                +-----------+
```

Figure 1. Transaction Fraud Elements

Transaction fraud activities normally involve the following actors:

1. Fraudsters: individuals or organizations that collect victims’ login credentials using a variety of means, including phishing and malware, and verify them (usually by attempting to log in to the victim’s account). Then, the Fraudsters may either recruit Fraud Executors themselves or wholesale the victims’ credentials to other Fraudsters, who will, in turn, recruit Fraud Executors.

2. Fraud Executors: individuals who attempt the fraudulent funds transfer or payment. In the case of fraudulent funds transfers, an account at either the same financial organization as that of the victim or a different one is opened as the destination account for the fraudulent transfer. Alternatively, a fraudulent payment is made using a check or electronic transfer.

3. Victims of both credential theft and transaction fraud.

4. Financial organizations that hold the victim’s and the Fraud Executor’s accounts.

5. Sensors at the financial organization that detect fraudulent transaction attempts, either in real-time or after the fact.

The intention of Thraud reporting is to enable any organization that has detected fraud to share this information, either internally or with other potential victim organizations. The receiving organization can use this information, for example, to institute manual review of transactions initiated from suspicious IP addresses.

4. IODEF-Document Incident Class

A Thraud Report SHALL be an instance of the IODEF-Document class, as defined in [RFC5070]. The report SHALL contain at least one Incident object, as defined in [RFC5070]. Each Incident object SHOULD contain information about a single fraud strategy. One Incident object MAY contain information about multiple fraudulent transactions that are consistent with the same fraud strategy. Each fraudulent transaction SHALL be described in a separate EventData object. The data model for the Incident class is defined in [RFC5070] and is repeated here, as Figure 2, for the reader’s convenience.

```
     +-------------+
     | Incident    |
     +-------------+
     |ENUM         |<>----------[ IncidentID ]
     | purpose     |<>--{0..1}--[ AlternativeID ]
     |STRING       |<>--{0..1}--[ RelatedActivity ]
     | ext-purpose |<>--{0..1}--[ DetectTime ]
     |ENUM         |<>--{0..1}--[ StartTime ]
     | lang        |<>--{0..1}--[ EndTime ]
     |ENUM         |<>----------[ ReportTime ]
     | restriction |<>--{0..*}--[ Description ]
     |             |<>--{1..*}--[ Assessment ]
     |             |<>--{0..*}--[ Method ]
     |             |<>--{1..*}--[ Contact ]
     |             |<>--{1..*}--[ EventData ]<>--[ AdditionalData ]
     |             |<>--{0..1}--[ History ]
     |             |<>--{1..*}--[ AdditionalData ]
     +-------------+
```

Figure 2. Data Model of the Incident Class

The AdditionalData abstract class is an extension point in the schema of the EventData class. Implementers SHALL include exactly one of the following objects in AdditionalData: FraudEventPayment, FraudEventTransfer, FraudEventIdentity, or FraudEventOther. Collectively, these are known as Thraud Records. The corresponding classes are defined by this specification in Section 5, below.

The Thraud profile of the Incident class is defined in Sections 6 and 7, below.

5. Thraud Record Class Definitions

Thraud Records are expressed in XML. Therefore, the dtype attribute of the AdditionalData element SHALL be assigned the value "xml".

A payment Thraud Record SHALL be structured as shown in Figure 3. See also Section 5.1.

```
          +------------------+
          | AdditionalData   |
          +------------------+
          | ENUM dtype (xml) |<>-----[ FraudEventPayment ]
          +------------------+
```

Figure 3. The FraudEventPayment Extension

A funds-transfer Thraud Record SHALL be structured as shown in Figure 4. See also Section 5.2.

```
          +------------------+
          | AdditionalData   |
          +------------------+
          | ENUM dtype (xml) |<>-----[ FraudEventTransfer ]
          +------------------+
```

Figure 4. The FraudEventTransfer Extension

An identity Thraud Record SHALL be structured as shown in Figure 5. See also Section 5.3.

```
          +------------------+
          | AdditionalData   |
          +------------------+
          | ENUM dtype (xml) |<>-----[ FraudEventIdentity ]
          +------------------+
```

Figure 5. The FraudEventIdentity Extension

Other Thraud Records SHALL be structured as shown in Figure 6. See also Section 5.4. The FraudEventOther class has an open definition to act as a placeholder for event types that emerge in the future.

```
          +------------------+
          | AdditionalData   |
          +------------------+
          | ENUM dtype (xml) |<>----[ FraudEventOther ]
          +------------------+
```

Figure 6. The FraudEventOther Extension

5.1. FraudEventPaymentType Class

The FraudEventPaymentType class is used to report payee instructions for a fraudulent payment or fraudulent payment attempt. Fraudsters sometimes use the same payee instructions (including the amount) for multiple fraudulent payment attempts. By reporting the payment instructions used in the fraud, other organizations may be able to detect similar fraudulent payment attempts to the same payee.

The structure of the FraudEventPaymentType class SHALL be as shown in Figure 7.

```
          +-------------+
          | FraudEvent- |
          | PaymentType |
          +-------------+
          |             |<>--{0..1}--[ PayeeName ]
          |             |<>--{0..1}--[ PostalAddress ]
          |             |<>--{0..1}--[ PayeeAmount ]
          +-------------+
```

Figure 7. The FraudEventPaymentType Class

The contents of the FraudEventPaymentType class are described below. At least one component MUST be present.

5.1.1. PayeeName

Zero or one value of type iodef:MLString. The name of the payee.

5.1.2. PostalAddress

Zero or one value of type iodef:MLString. The format SHALL be as documented in Section 2.23 of [RFC4519], which defines a postal address as a free-form multi-line string separated by the "$" character.

5.1.3. PayeeAmount

Zero or one value of type thraud:AmountType. See Section 5.5.

5.2. FraudEventTransferType Class

The FraudEventTransferType class is used to report the payee instructions for a fraudulent funds transfer or fraudulent funds transfer attempt. Fraudsters sometimes use the same payee instructions (including the amount) for multiple fraudulent funds transfer attempts. By reporting the funds transfer instructions used in the fraud, other organizations may be able to detect similar fraudulent funds transfer attempts to the same payee.

The structure of the FraudEventTransferType class SHALL be as shown in Figure 8.

```
          +--------------+
          | FraudEvent-  |
          | TransferType |
          +--------------+
          |              |<>--{0..1}--[ BankID ]
          |              |<>--{0..1}--[ AccountID ]
          |              |<>--{0..1}--[ AccountType ]
          |              |<>--{0..1}--[ TransferAmount ]
          +--------------+
```

Figure 8. The FraudEventTransferType Class

The contents of the FraudEventTransferType class are described below. At least one component MUST be present.

5.2.1. BankID

Zero or one value of type thraud:BankIDType. The structure of the BankIDType class SHALL be as shown in Figure 9. The contents SHALL be of type xs:string. The namespace attribute SHALL be of type xs:anyURI and SHALL identify the numbering system used to identify the bank or account.

```
          +-------------------+
          | BankIDType        |
          +-------------------+
          | STRING            |
          |                   |
          |  STRING namespace |
          +-------------------+
```

Figure 9. The BankIDType Class

A list of registered namespace identifiers is maintained at:

http://www.openauthentication.org/thraud/resources/bank-id-namespace.htm

The following namespace attribute values and their semantics are registered.

One of the nine-digit Routing Numbers registered to the financial organization that holds the account, as administered by The American Bankers Association.

http://www.openauthentication.org/thraud/resources/bank-id-namespace.htm#american_bankers_association

The three-digit Institution Number registered to the financial organization that holds the account, as administered by The Canadian Payments Association.

http://www.openauthentication.org/thraud/resources/bank-id-namespace.htm#canadian_payments_association

The corresponding AccountID represents the ISO 13616 International Bank Account Number [ISO13616-1:2007] in the "electronic form" (i.e., containing no spaces) that is assigned to the account, as administered by the Society for Worldwide Interbank Financial Telecommunication (SWIFT). The corresponding BankID xs:string value SHOULD be set to the null string. Receiving organizations SHOULD ignore the corresponding BankID value.

http://www.openauthentication.org/thraud/resources/bank-id-namespace.htm#iso13616_1_2007

The eight-character Bank Identifier Code [ISO9362:1994] registered to the financial organization that holds the account, as administered by SWIFT.

http://www.openauthentication.org/thraud/resources/bank-id-namespace.htm#iso9362_1994

Other namespace values MUST be agreed upon among participants. Requests to register new values SHOULD be made at:

http://www.openauthentication.org/thraud/form/bank-id-namespace

Note that a single organization may be identified by more than one value for any one or more of these namespaces. Therefore, receiving organizations SHOULD take this into account in their matching procedure.

5.2.2. AccountID

Zero or one value of type xs:string. The destination primary account number, as administered by the financial organization identified in the BankID element. In the case where the BankID namespace attribute value is "iso13616_1_2007", this element SHALL contain the International Bank Account Number in the "electronic form" (i.e., containing no spaces) that is assigned to the account. In all other cases, the element SHALL contain only the account number, as administered by the financial organization that holds the account. The reporting organization SHALL remove all prefixes that identify the country, bank, or branch.

5.2.3. AccountType

Zero or one value of type thraud:AccountTypeType. See Section 5.6.

5.2.4. TransferAmount

Zero or one value of type thraud:AmountType. See Section 5.5.

5.3. FraudEventIdentityType Class

The FraudEventIdentityType class is used to report a fraudulent impersonation or fraudulent impersonation attempt. By reporting the impersonation event, other potential victims may be able to detect similar fraudulent impersonation attempts.

The structure of the FraudEventIdentityType class SHALL be as shown in Figure 10.

```
+--------------+
| FraudEvent-  |
| IdentityType |
+--------------+
|              |<>--{1..*}--[ IdentityComponent ]
+--------------+
```

Figure 10. The FraudEventIdentityType Class

The contents of the FraudEventIdentityType class are described below.

5.3.1. IdentityComponent

One or more values of type iodef:ExtensionType. This specification defines two extensions: EmailAddress and UserID.

5.3.1.1. EmailAddress

In reporting an identity fraud event, the reporting institution MAY include the victim’s email address. This SHALL be achieved by placing an object of type iodef:Email in the IdentityComponent object. It SHALL contain the email address of the intended fraud victim.

The IdentityComponent.dtype attribute SHALL be set to the value "string".

The IdentityComponent.meaning attribute SHALL be set to the value "victim email address".

5.3.1.2. UserID

In reporting an identity fraud event, the reporting institution MAY include the victim’s user identifier. This SHALL be achieved by placing an object of type iodef:ExtensionType in the IdentityComponent object. The data type of the extension contents SHALL be xs:string. It SHALL contain the user identifier of the intended fraud victim.

The IdentityComponent.type attribute SHALL be set to the value "string".

The IdentityComponent.meaning attribute SHALL be set to the value "victim user id".

5.4. FraudEventOtherType Class

The FraudEventOtherType class SHALL be used to report fraudulent events other than those detailed above, such as new event types that may emerge at some time in the future. This class enables such events to be reported, using this specification, even though the specific characteristics of such events have not yet been formally identified. By reporting the details of these unspecified event types, other institutions may be able to detect similar fraudulent activity.

The structure of the FraudEventOtherType class SHALL be as shown in Figure 11.

```
+-------------+
| FraudEvent- |
| OtherType   |
+-------------+
|             |<>----------[ OtherEventType ]
|             |<>--{0..1}--[ PayeeName ]
|             |<>--{0..1}--[ PostalAddress ]
|             |<>--{0..1}--[ BankID ]
|             |<>--{0..1}--[ AccountID ]
|             |<>--{0..1}--[ AccountType ]
|             |<>--{0..1}--[ PayeeAmount ]
|             |<>--{0..1}--[ OtherEventDescription ]
+-------------+
```

Figure 11. The FraudEventOtherType Class

Many of the components of the FraudEventOtherType class are also components of the FraudEventPaymentType or FraudEventTransferType classes. Their use in the FraudEventOtherType class is identical to their use in those classes. Therefore, their descriptions are not duplicated here. Only components that are unique to the FraudEventOtherType class are described below.

5.4.1. OtherEventType

One value of type xs:anyURI. A name that classifies the event.

A list of registered "other event type" identifiers is maintained at:

http://www.openauthentication.org/thraud/resources/other-event-type.htm

Requests to register new values SHOULD be made at:

http://www.openauthentication.org/thraud/form/other-event-type

5.4.2. OtherEventDescription

Zero or one value of type iodef:MLString. A free-form textual description of the event.

5.5. AmountType Class

The AmountType class SHALL be as shown in Figure 12. It SHALL be used to report the amount of a payment or transfer fraud.

```
+------------------+
| AmountType       |
+------------------+
| DECIMAL          |
|                  |
|  STRING currency |
+------------------+
```

Figure 12. The AmountType Class

The contents of the AmountType class are described below.

5.5.1. Class Contents

REQUIRED DECIMAL. The amount of the payment or transfer.

5.5.2. Currency

REQUIRED STRING. The three-letter currency code [ISO4217:2008].

5.6. AccountTypeType Class

The AccountTypeType class SHALL be as shown in Figure 13. It SHALL be used to report the type of the destination account.

```
+-----------------+
| AccountTypeType |
+-----------------+
| STRING          |
|                 |
|  STRING lang    |
+-----------------+
```

Figure 13. The AccountTypeType Class

Receiving organizations MUST be capable of processing contents containing spelling variations.

6. IODEF Profile for an Activity Thraud Report

This section describes the profile of the IODEF Incident class for a compliant Activity Thraud Report.

6.1. Mandatory Components

A Thraud Report SHALL conform to the data model specified for an IODEF-Document in [RFC5070]. The following components of that data model, while optional in IODEF, are REQUIRED in a conformant Thraud Report.

Receiving organizations MAY reject documents that do not contain all of these components. Therefore, reporting organizations MUST populate them all.

Except where noted, these components SHALL be interpreted as described in [RFC5070].

Incident.Contact.ContactName - The name of the reporting organization. In case the reporting organization acts as a consolidator of reports from other organizations, elements of this class SHALL contain the name of the consolidator.
Incident.Contact.Email - An email address at which the reporting organization may be contacted.
Incident.Contact.Telephone
Incident.EventData
Incident.EventData.AdditionalData - SHALL contain exactly one Thraud Record.

6.2. Recommended Components

Receiving organizations SHOULD be capable of processing the following components. However, they MUST NOT reject documents because they are either present or absent.

If available, reporting organizations SHOULD include these components in Thraud Reports. Except where noted, these components SHALL be interpreted as described in [RFC5070].

```
Incident.Contact.Contact
Incident.Contact.Contact.ContactName - The name of the reporting
   fraud analyst.
Incident.Contact.Contact.Email - The email address of the reporting
   fraud analyst.
Incident.Contact.Contact.Telephone - The telephone number of the
   reporting fraud analyst.
Incident.EventData.Method
Incident.EventData.Method.Description
Incident.Assessment.Confidence
Incident.Assessment.Impact
Incident.Assessment.MonetaryImpact
Incident.EventData.DetectTime
Incident.EventData.StartTime
Incident.EventData.EndTime
Incident.EventData.Flow
Incident.EventData.Flow.System
Incident.EventData.Flow.System.Service
Incident.EventData.Flow.System.Node.NodeName
Incident.EventData.Flow.System.Node.Address
```

6.3. Deprecated Components

This profile provides no guidance to receiving organizations on the proper processing of the following components. Therefore, the reporting organization has no assurance that the receiving organization will handle them in an appropriate manner and SHOULD NOT include them in a Thraud Report. However, receiving organizations MUST NOT reject reports that do contain these components.

```
Incident.DetectTime
Incident.AlternativeID
Incident.RelatedActivity
Incident.StartTime
Incident.EndTime
Incident.ReportTime
Incident.Description
Incident.Method
```

```
   Incident.History
   Incident.AdditionalData
   Incident.ext-purpose
   Incident.IncidentID.instance
   Incident.Contact.Description
   Incident.Contact.RegistryHandle
   Incident.Contact.PostalAddress
   Incident.Contact.Fax
   Incident.Contact.TimeZone
   Incident.Contact.AdditionalData
   Incident.Contact.Contact.Description
   Incident.Contact.Contact.RegistryHandle
   Incident.Contact.Contact.PostalAddress
   Incident.Contact.Contact.Fax
   Incident.Contact.Contact.TimeZone
   Incident.Contact.Contact.AdditionalData
   Incident.Contact.ext-role
   Incident.Contact.ext-type
   Incident.Contact.Contact.ext-role
   Incident.Contact.Contact.ext-type
   Incident.EventData.Method.Reference
   Incident.EventData.Method.Reference.Description
   Incident.EventData.Method.AdditionalData
   Incident.EventData.Method.Reference.URL
   Incident.Assessment.TimeImpact
   Incident.Assessment.AdditionalData
   Incident.Assessment.Impact.type
   Incident.EventData.Description
   Incident.EventData.Contact
   Incident.EventData.Assessment
   Incident.EventData.Expectation
   Incident.EventData.Record
   Incident.EventData.EventData
   Incident.EventData.Flow.System.OperatingSystem
   Incident.EventData.Flow.System.Counter
   Incident.EventData.Flow.System.Description
   Incident.EventData.Flow.System.AdditionalData
   Incident.EventData.Flow.System.ext-category
   Incident.EventData.Flow.System.Node.Location
   Incident.EventData.Flow.System.Node.DateTime
   Incident.EventData.Flow.System.Node.NodeRole
   Incident.EventData.Flow.System.Node.Counter
   Incident.EventData.Flow.System.Node.Address.ext-category
   Incident.EventData.Flow.System.Service.ProtoType
   Incident.EventData.Flow.System.Service.ProtoCode
   Incident.EventData.Flow.System.Service.ProtoField
   Incident.EventData.Flow.System.Service.Application
```

7. IODEF Profile for a Signature Thraud Report

A Signature Thraud Report SHALL convey information about the behavior associated with fraudulent events, rather than reporting the details of the specific events themselves.

Sharing Signature Thraud Reports helps receiving organizations to detect suspicious behavior in their own systems.

A Signature Thraud Report SHALL conform to the profile described in Section 6.

8. IODEF Additional Attribute Values

Additional IODEF attribute standard values are defined here.

8.1. Purpose Attribute

The following additional values are defined for the Incident.purpose attribute.

Add - The enclosed Thraud Record values SHOULD be added to the corpus by the receiving organization.

Delete - The enclosed Thraud Record types SHOULD be deleted from the corpus by the receiving organization.

Modify - The enclosed Thraud Record values SHOULD replace the corresponding values in the corpus. Where no corresponding types currently exist in the corpus, the enclosed values SHOULD be added to the corpus by the receiving organization.

9. Security Considerations

This document describes a document format for exchanging information about successful or attempted transaction and authentication fraud incidents. The information is intended to be used to improve the effectiveness of participants’ fraud detection and prevention programs. The effectiveness of such programs depends critically on the accuracy, reliability, confidentiality, and timeliness of both the information and the participants in its exchange. Threats to accuracy, reliability, and confidentiality include (but are not limited to) those described here.

Fraudsters may attempt to introduce reports that delete or modify incident information in the corpus. Therefore, origin authentication MUST be employed. Human review SHOULD be performed prior to implementing modifications to the corpus.

Fraudsters may attempt to interrupt or redirect submissions, thereby preventing the sharing of intelligence concerning their fraud strategies. Therefore, authenticated receipts SHOULD be employed.

Fraudsters may attempt to impersonate legitimate submitters, thereby poisoning their reputations and rendering ineffective their future submissions. Origin authentication MUST be used to ensure that the sources of reports are properly identified.

Fraudsters that can view incident reports may adapt their fraud strategies to avoid detection. Therefore, reports MUST be protected by confidentiality services including transport encryption and access control.

In order to prevent inadvertent disclosure of incident data, incident reports SHOULD be encrypted while in storage.

The submitter of an incident report may incorrectly identify legitimate activity as a fraud incident. This may lead to denial of service by a receiving organization that relies on the report or information derived from the report. Receiving organizations SHOULD operate a reputation service, in which the reliability of the information from particular sources is assessed and tracked and subsequent reports are weighted accordingly. The source of reports MUST be authenticated. Receiving organizations SHOULD use reports to step up authentication assurance, rather than simply denying service.

A receiving organization may misuse a Thraud Report to deny service, resulting in a loss for a legitimate user. If such a user were to learn the identity of the source of the information that led to the denial of service, then that source may become implicated in any resulting claim for compensation. This, in turn, may discourage reporting organizations from participating in intelligence sharing. Therefore, original sources SHOULD NOT be identified in consolidated reports.

Any origin authentication and data integrity mechanism that is acceptable to both parties MAY be used.

Any transport confidentiality mechanism that is acceptable to both parties MAY be used.

This specification does not include a data compression technique. Therefore, it does not introduce any denial of service vulnerabilities related to decompression.

10. IANA Considerations

This specification registers two identifiers:

- The media sub-type name "thraud+xml" in the standard registration tree.

- The xml namespace identifier - urn:ietf:params:xml:ns:thraud-1.0.

10.1. Media Sub-Type

Type name: application

Subtype name: thraud+xml

Required parameters: none

Optional parameters: "charset": same as the charset parameter of application/xml, as specified in [RFC3023].

Encoding considerations: same as encoding considerations of application/xml, as specified in [RFC3023].

Security considerations: in addition to the security considerations described in Section 9, this registration has all of the security considerations described in [RFC3023].

Interoperability considerations: None beyond the interoperability considerations described in [RFC3023].

Published specification: the media type data format is defined in RFC 5941.

Applications that use this media type: transaction and authentication fraud analysis and reporting applications, and risk-based transaction and authentication evaluation applications.

Additional information
Magic number(s): none
File extension: .tfi
Macintosh file type codes: none

Person and email address to contact for further information:
"D M’Raihi <davidietf@gmail.com>"

Intended usage: LIMITED USE

Restrictions on usage: thraud media are intended for no usage other than the exchange of fraud intelligence data.

Author: D M’Raihi

Change controller: the IESG

10.2. XML Namespace

IANA has registered the xml namespace identifier:

URI: urn:ietf:params:xml:ns:thraud-1.0

Registrant Contact:

Siddharth Bajaj
VeriSign, Inc.
487 E. Middlefield Road
Mountain View, CA 94043
USA
Email: sbajaj@verisign.com

XML: None. Namespace URIs do not represent an XML specification.

11. Conclusion

This specification introduces a transaction fraud (Thraud) reporting document structure that enables the sharing of fraud data. Based on the IODEF-Document format, the proposed extension facilitates interoperability to increase the security of online applications.

12. References

12.1. Normative References

[ISO13616-1:2007] Financial services - International bank account number (IBAN) -- Part 1: Structure of the IBAN, ISO 13616-1:2007.

[ISO4217:2008] Financial services - Codes for the representation of currencies and funds, ISO 4217:2008.

[ISO9362:1994] Banking -- Banking telecommunication messages -- Bank identifier codes, ISO 9362:1994.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC3023] Murata, M., St. Laurent, S., and D. Kohn, "XML Media Types", RFC 3023, January 2001.

[RFC4519] Sciberras, A., Ed., "Lightweight Directory Access Protocol (LDAP): Schema for User Applications", RFC 4519, June 2006.

[RFC5070] Danyliw, R., Meijer, J., and Y. Demchenko, "The Incident Object Description Exchange Format", RFC 5070, December 2007.

12.2. Informative References

[UML] Information technology -- Open Distributed Processing -- Unified Modeling Language (UML) Version 1.4.2, ISO/IEC 19501:2005.

Appendix A. Thraud Record XML Schema

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema targetNamespace="urn:ietf:params:xml:ns:thraud-1.0"
xmlns:thraud="urn:ietf:params:xml:ns:thraud-1.0"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:iodef="urn:ietf:params:xml:ns:iodef-1.0"
elementFormDefault="qualified"
attributeFormDefault="unqualified">
 <xs:import namespace="urn:ietf:params:xml:ns:iodef-1.0"
schemaLocation="
http://www.cert.org/ietf/inch/schema/rfc5070.xsd"/>
 <xs:element name="FraudEventPayment"
type="thraud:FraudEventPaymentType"/>
 <xs:element name="FraudEventTransfer"
type="thraud:FraudEventTransferType"/>
 <xs:element name="FraudEventIdentity"
type="thraud:FraudEventIdentityType"/>
 <xs:element name="FraudEventOther"
type="thraud:FraudEventOtherType"/>
 <xs:complexType name="FraudEventPaymentType">
  <xs:sequence>
   <xs:element name="PayeeName" type="iodef:MLStringType"
minOccurs="0"/>
   <xs:element name="PostalAddress" type="iodef:MLStringType"
minOccurs="0"/>
   <xs:element name="PayeeAmount" type="thraud:AmountType"
minOccurs="0"/>
  </xs:sequence>
 </xs:complexType>
 <xs:complexType name="FraudEventTransferType">
 <xs:sequence>
   <xs:element name="BankID" type="thraud:BankIDType"
minOccurs="0"/>
   <xs:element name="AccountID" type="xs:string" minOccurs="0"/>
   <xs:element name="AccountType" type="iodef:MLStringType"
minOccurs="0"/>
   <xs:element name="TransferAmount" type="thraud:AmountType"
minOccurs="0"/>
  </xs:sequence>
 </xs:complexType>
 <xs:complexType name="FraudEventIdentityType">
  <xs:sequence maxOccurs="unbounded">
   <xs:element name="IdentityComponent"
type="iodef:ExtensionType"/>
  </xs:sequence>
 </xs:complexType>
 <xs:complexType name="FraudEventOtherType">
```

```
  <xs:sequence>
   <xs:element name="OtherEventType" type="xs:anyURI"/>
   <xs:element name="PayeeName" type="iodef:MLStringType"
minOccurs="0"/>
   <xs:element name="PostalAddress" type="iodef:MLStringType"
minOccurs="0"/>
   <xs:element name="BankID" type="thraud:BankIDType"
minOccurs="0"/>
   <xs:element name="AccountID" type="xs:string" minOccurs="0"/>
   <xs:element name="AccountType" type="iodef:MLStringType"
minOccurs="0"/>
   <xs:element name="PayeeAmount" type="thraud:AmountType"
minOccurs="0"/>
   <xs:element name="OtherEventDescription"
type="iodef:MLStringType" minOccurs="0"/>
  </xs:sequence>
 </xs:complexType>
 <xs:complexType name="AmountType">
  <xs:simpleContent>
   <xs:extension base="xs:decimal">
    <xs:attribute name="currency" type="xs:string"/>
   </xs:extension>
  </xs:simpleContent>
 </xs:complexType>
 <xs:complexType name="BankIDType">
  <xs:simpleContent>
   <xs:extension base="xs:string">
    <xs:attribute name="namespace" type="xs:anyURI"
use="required"/>
   </xs:extension>
  </xs:simpleContent>
 </xs:complexType>
 <xs:element name="UserID" type="xs:string"/>
</xs:schema>
```

Appendix B. Example of a Thraud Report

```
<?xml version="1.0" encoding="UTF-8"?>
<IODEF-Document xmlns="urn:ietf:params:xml:ns:iodef-1.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:ietf:params:xml:ns:iodef-1.0"
lang="en">
 <Incident purpose="reporting">
  <IncidentID name="fraud.openauthentication.org">908711
       </IncidentID>
  <ReportTime>2006-10-12T00:00:00-07:00</ReportTime>
  <Assessment>
   <Impact severity="high" completion="failed"/>
   <Confidence rating="high"/>
  </Assessment>
    <Contact type="organization" role="creator">
         <ContactName>Example Corp.</ContactName>
         <Email>contact@example.com</Email>
         <Telephone>+1.972.555.0150</Telephone>
    </Contact>
  <EventData>
   <DetectTime>2006-10-12T07:42:21-08:00</DetectTime>
   <Flow>
    <System category="source">
     <Node>
      <Address category="ipv4-addr">192.0.2.53</Address>
     </Node>
     <Description>Source of numerous attacks</Description>
    </System>
   </Flow>
   <AdditionalData dtype="xml">
    <FraudEventTransfer xmlns="urn:ietf:params:xml:ns:thraud-
1.0" xmlns:iodef="urn:ietf:params:xml:ns:iodef-1.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:ietf:params:xml:ns:thraud-1.0">
     <BankID
namespace="http://www.openauthentication.org/thraud/resources/
bank-id-namespace.htm#american_bankers_association">123456789</BankID>
     <AccountID>3456789</AccountID>
     <AccountType lang="en">saving</AccountType>
     <TransferAmount currency="USD">10000</TransferAmount>
    </FraudEventTransfer>
   </AdditionalData>
  </EventData>
 </Incident>
</IODEF-Document>
```

Authors' Addresses

David M'Raihi
VeriSign, Inc.
685 E. Middlefield Road
Mountain View, CA 94043
USA
Phone: 1-650-426-3832
EMail: davidietf@gmail.com

Sharon Boeyen
Entrust, Inc.
1000 Innovation Drive
Ottawa, ON, K2K 3E7
Canada
Phone: 1-613-270-3181
EMail: sharon.boeyen@entrust.com

Michael Grandcolas
Grandcolas Consulting, LLC
247 Ocean Park Blvd.
Santa Monica, CA 90405
USA
Phone: 1-310-399-1747
EMail: michael.grandcolas@hotmail.com

Siddharth Bajaj
VeriSign, Inc.
487 E. Middlefield Road
Mountain View, CA 94043
USA
Phone: 1-650-426-3458
EMail: sbajaj@verisign.com