IODEF Design principles and IODEF Data Model Overview

IODEF Data Model and XML DTD
pre-draft Version 0.03

TERENA IODEF WG
Yuri Demchenko <demch@terena.nl>
Outlines

• Current IODEF documents
• IODEF Design Principles
• Relation between IDMEF and IODEF
• IODEF Data Model Overview
Current IODEF documents

- Incident Object Description and Exchange Format Requirements
  - Published as RFC 3067
    http://www.ietf.org/rfc/rfc3067.txt
- IODEF Data Model and XML DTD pre-draft Version 0.03

Other documents
http://www.terena.nl/tech/task-forces/tf-csirt/iodef/

- Taxonomy of the Computer Security Incident related terminology
  http://www.terena.nl/tech/task-forces/tf-csirt/iodef/docs/i-taxonomy_terms.html
- Relations between the IODEF Incident handling systems and IDMEF developed by IETF IDWG - Request for comments to IDWG and ITD/IODEF WG – IETF50, March 2001
  http://www.terena.nl/tech/task-forces/tf-csirt/iodef/docs/iodef-idmef-xmldtd-00-rfc.html
IODEF Design: Problems addressed

Problems addressed by IODEF:

- Incident data are inherently heterogeneous
  - May change during lifetime/investigation
- Incident information can originate from different sources
  - Incident Object may be created by CSIRT, reported by community or initially based on IDS Alert
- Incident description may contain sensitive information
  - Sensitive information should be protected
  - Evidence integrity and in some cases confidentiality should be secured
IODEF Design Principles

• Content-driven
  ♦ Object oriented approach allows simple introduction of new objects to extend new content description
  ♦ Possibility to apply different attributes to different elements

• Unambiguous representation
  ♦ The same Incident descriptions created by different CSIRTs should be identified as one Incident

• Support correlation of related incidents
  ♦ Provide basis for necessary cooperation between CSIRTs
  ♦ **Principle of Incident Object ownership – key concept for Incidents correlation and unambiguous presentation**

• XML implementation

• Seamless IDMEF integration
XML Implementation

- Human readable, but machine parsable
  - XML DT vs XML Schema
- Easily extensible

- Tools are widely (and free) available
- Internationalisation
  - Support of CSIRT’s local languages

- Significant classes re-use from IDMEF
  - Open source IDMEF idmeflibxml extension
Relation between IDMEF and IODEF

1. IODEF should be compatible with IDMEF and be capable to use/include IDMEF message into IO
   - Current IODEF implementation provides two options:
     - Use IncidentAlert class container to wrap up Alert/IDMEF
     - Decompose Alert/IDMEF message into Incident/IODEF classes

2. IODEF follows IDMEF development
   - IDMEF last call draft version 0.6 will be completely incorporated changes into IODEF pre-draft version 0.04
     - On request by IODEF WG IDWG made considerable changes to IDMEF elements definition
Differences between IDMEF and IODEF

- Main IODEF actors are CSIRTs – not IDS
  - CSIRT as owner of the IO
- IODEF is human (interface/interaction) oriented
  - Human readable, but machine parsable
- Incident Object has longer lifetime compare to one time use of IDMEF message
  - Incident handling (reporting, investigation, etc.)
  - Incident storage
  - Statistics and trend analysis
## IODEF vs IDMEF: Top level classes

### IODEF top level classes:
- Incident
  - Attack <- Target
  - Attacker <- Source
  - Victim <- Target
  - Method <- Classification
  - Evidence <- Analyzer
  - Assessment
  - Authority
  - CorrelationIncident
  - History
  - AdditionalData
- IncidentAlert
  - Alert

### IDMEF top level classes:
- Alert
  - Source
  - Target
  - Classification
  - CreatTime
  - DetectTime
  - AnalyzerTime
  - Analyzer
  - Assessment
  - CorrelationAlert
  - ToolAlert
  - OverflowAlert
  - AdditionalData
IODEF Top-Level Classes

- **IODEF-Description** is the root container class
- Different types of incident reports derive subclass
  - Incident: incident report
  - IncidentAlert: IDMEF Alert
  - Vulnerability: (proposed) vulnerability report
The Incident Class

- **Incident**
  - **WHAT/WHERE**: Attack
  - **WHO**: Attacker/Victim: information on source and destination of the incident
  - **HOW**: Method of attack, analysis of incident, Assessment
  - **PROOF**: Evidence: support for incident analysis
  - **OWNER**: Authority: incident creator
  - **Log of events/actions**: History
  - **Extension mechanism**: Additional Data

- **Attack**
- **Attacker**
- **Victim**
- **Method**
- **Assessment**
- **Evidence**
- **Authority**
- **History**
- **AdditionalData**
IODEF-003: Full Data Model

http://www.terena.nl/tech/task-forces/tf-csirt/iodef/docs/iodef-datamodel-draft-003.html
IODEF Data Model – remaining issues

To be solved in IODEF-004 or moved to IETF INCH WG
1. XML DTD YET to BE UPDATED with recent changes 15 and 16
   Modify the CorrelateIncident Class (Section 5.2.2.2)
2. Modify Section 7.2 Unrecognized XML Tags to target better IDMEF and IODEF integration
3. How to provide for packet and flow representations in <Evidence>
4. Representing vulnerability data (vul-description)
5. Further discussion and presentation in the IODEF document the issue about reusing IDMEF Classes in the IODEF documents and in guidance for implementers.
6. Granularity of setting restrictions on various elements?
7. Certain identifier attributes may not be necessary