EISPP Common Advisory Format Description

Identifier: **EISPP-D3-001-TR**

Version 1.1
Date 2003/02/06

Table of Content

Glossary

BNF	Bacchus Naur Form
CAIF	Common Advisory Interchange Format
CERT	Computer Emergency Response Team
DoS	Denial Of Service
DTD	Document Type Description
EISPP	European Information Security Promotion Program
HTML	Hypertext Markup Language
IDMEF	Intrusion Detection Message Exchange Format
IODEF	Incident Object Description and Exchange Format
PCC	Project Coordination Committee
RUS CERT	Rechenzentrum Universitat Stuttgart CERT (University of Stuttgart)
SME	Small and Medium Enterprise
TBC	To Be Completed
TBD	To Be Defined
URL	Uniform resource location
WP	Workpackage
XML	eXtended Markup Language

Related documents

Applicable Documents

Ref.	*Title*	*Version*	*Date*
AD01	CONTRACT No IST-2001-35200 and Annexes		
AD02	Project Consortium Agreement		
AD03	Annex 1 - Description of Work	3.0	2002/04/11

Reference Documents

Ref.	*Title*	*Version*	*Date*
RD01	Common Advisory Interchange Format (CAIF) – Requirements http://cert.uni-stuttgart.de/projects/caif/		2002/06/05

1. EXECUTIVE SUMMARY

The European Information Security Promotion Programme (**EISPP**) strives to set up a network of expertise with the aim of providing European SMEs with those IT Security services that give them the necessary trust in e-commerce to develop their businesses in that direction. EISPP is a project fund by the EU through the fifth European Framework Program within the thematic program Information Society Technologies (IST). Further information about EISPP can be found at its website, http://www.eispp.org/.

Probably the most important security service SMEs have to be provided with, is an advisory service, i.e., the distribution of so-called security advisories that provides system administrators with precise and timely information about new vulnerabilities and what can be done against them. Such information is absolutely essential for IT security, because new vulnerabilities are discovered on a daily basis. IT systems can only be kept secure, if they are regularly upgraded or patched such that the latest security holes are closed again.

This document describes a corner stone of the EISPP approach towards supplying SMEs with security advisories: a common advisory format, which will enable an easy exchange of advisory data between the four CERTs participating in EISPP. The advisory format merges the best-practice information regarding security advisories of these four CERTs.

The format is defined using XML, so the various standards and standard tools of the XML-family can be used for advisory processing. The XML data-type description of this (and future versions) of the format, together with sample XSLT style sheets for displaying advisory data, are made publicly available on EISPP's website http://www.eispp.org.

2. INTRODUCTION

2.1. European Information Security Promotion Programme (EISPP)

Adequate IT security is probably the most important aspect of creating a European environment in which an information society can flourish: Deficits in IT security bring risks to an otherwise desirable expansion of Internet-use by businesses and governments, deter potential home users, and generally endanger what already has become the nerve system of our critical infrastructures. The European Commission therefore has increased the importance of IT security within its new action plan eEurope 2005.

Amongst other measures, the action plan envisions a European warning and information system, which should keep all users of IT infrastructure up-do-date with the latest security issues. The impact of newly discovered vulnerabilities would thus be reduced, massive attacks targeted at such vulnerabilities, e.g., through worm programs, could hopefully be contained before much damage is done.

The initial plans for establishing such a European warning and information system conform to the nature of the European Union : there are no plans for one organization in which all activities regarding IT security are to be centralized. Rather, the European Commission envisions an increased networking between national players such as CERT organizations and similar bodies.

The main objective of EISPP is to set-up a European framework aimed at providing European SMEs with the necessary IT Security services in order to give them the necessary trust in e commerce, which is important in developing their businesses. EISPP thus is a pioneer regarding the European Commission's vision of forming a European warning and information system on the basis of international networks and cooperations within the European Union. The results of EISPP will therefore be significant for all other attempts for creating networks of expertise in IT-security.

2.2. Workpackage 3

New vulnerabilities are discovered on a daily basis. IT systems can only be kept secure, if they are regularly upgraded or patched such that the latest security holes are closed again. System administrators therefore need precise and timely information about new vulnerabilities and what can be done against them. Such information is usually provided in form of "security advisories", issued by vendors for their own products and CERTs for the products that are of interest to each CERT's constituency.

The focus of WP3 (workpackage 3) is to create an infrastructure that enables CERTs to cooperate in the production of advisories. To make the cooperation worthwhile for the member CERTs, WP3 takes care of supplying processes and infrastructure for reuse of work (e.g., it should be possible to import an advisory of another CERT into one's production system to share the work on menial tasks such as collecting links to patches and references, etc.)

This document describes one key element of that infrastructure : the common format. This advisory format is needed to enable automated exchanges of CERTs' advisories within the EISPP community. This format is formally defined as an XML DTD (which describes the fields and sections that could exist in an advisory). This document also describes the format in plain language, and gives guidance there, on how fields must be completed.

It must be noted however that the common format does not include a description of how advisories must be presented (i.e. the final layout of an advisory as sent by a CERT to a user). The advisory format is XML and it differs from the format in which advisories are presented to the reader such as HTML or ASCII. WP3 has produced basic translation schemes from XML to HTML and ASCII.

This document also includes a survey of other recent projects involving structured data exchange.

3. EISPP ADVISORY FIELDS — OVERVIEW

As mentioned above, security advisories serve to provide users of IT infrastructure with timely information regarding newly discovered vulnerabilities and what can be done against them. The following table provides an overview of the contents that the EISPP consortium regards as essential for security advisories. They are described in detail in the following section.

Field	Description
Identification Data	
Issuer	Advisory Issuer
Reference Number	An advisory reference number
Language	Language of the advisory
Title	The advisory's title
Abstract	A short abstract that complements the information given already in the title.
History Data	
Version History	Information about the advisory's current version/revision, along with history information and change dates
Update Information	Information about the relation of the advisory to prior/later advisories of the same issuer
Vulnerability Classification	
Confidence Level	Information about the confidence the issuer puts into the presented information.
Vulnerability Category	A very short description of the vulnerability.
Attack Requirements	Technical requirements needed by an attacker to exploit the vulnerability.
Vulnerability Impact	Information about the impact for the targeted system if the vulnerability exploitation succeeds.
Attack Expertise	Level of knowledge for the attacker to exploit the vulnerability.
Risk	Overall assessment of the vulnerability.
System Information	
Affected Platform	Information about platforms affected by the described vulnerability.
Affected Software	Information about software affected by the described vulnerability.
Remarks	Additional remarks, e.g., with information about systems that may also be affected, are not affected, etc.
Description	
Publication Context	Information that puts the advisory into context.
Technical Context	Information that helps the user to understand the technical context of the advisory.
Description	Description of the vulnerability/vulnerabilities treated by the advisory.
Technical Information	Detailed technical information, targeted more at security experts than the average reader.
Diagnostic	Information to help the reader with diagnostics, i.e., to determine whether his system has the described vulnerability.
Solution	
Solution Introduction	General information about possible solutions.

Solution Sections	Each section describes a possible solution. Sections may be divided by solution type (patch, workaround, etc.), affected system, or both.
Vulnerability Identifiers and Additional Resources	
Vulnerability Identifiers	List of standard vulnerability identifiers such as CVE numbers, MS knowledgebase entries, etc., for the vulnerabilities described in this advisory.
Additional Resources	References to relevant material such as other advisories.

4. EISPP ADVISORY FIELDS — DETAILED DESCRIPTION

The EISPP advisory format is presented by describing all the fields that will be present within the format.

The field template

Each field is presented using the following template:

(x) Field name

Short field description.

Content type

Information about the type of content within the field. Possible choices are

- Language-independent text, free text, or formatted text
- List-of-values
- Structured content (described with a semi-formal grammar).

Content Description

Information about how the field should be used and, if necessary, detailed description of the content type (in the case of structured content).

Further comments

Questions and thoughts about the field.

The semi-formal grammar

As was mentioned above, the content type may sometimes be described in form of a semi-formal grammar. Here is an example:

```
<telephone_list> ::= <person>*
<person>         ::= <name>.<telephone_nr>+ <email>* [birthday]
<name>           ::= [title] <first_name> <last_name>
<telephone_nr>    ::= language-independent text (a telephone number)
<email>          ::= language-independent text (an emai)l
<birthday>       ::= yyyy-mm-dd
<title>          ::= Mr | Mrs | Ms | Dr
```

The grammatic description uses the mechanisms of extended BNF. The most prominent features are:

- A (possibly empty) list is indicated with an asterix '*' in post-fix notation
- A non-empty list is indicated with a plus sign '+' in post-fix notation
- Options, i.e., zero or one occurrence, is indicated with square brackets '[…]'; if an option contains a single non-terminal, the pointed brackets that are used to enclose non-terminal names are not written.
- Choice between several options is indicated using the binary operator '|'.

Free text vs. formatted text

Fields whose content type is described as "free text" or "formatted text" have no formal restriction of their content. The difference between free text and formatted text is that formatted text may contain formatting tags that serves for pretty printing the text, providing for different font styles, paragraph styles, lists, etc. (See Section 5 for details.) Language-independent text never allows formatting tags.

The next section describes the fields of the EISPP advisory format.

Multi-language Feature

With the EISPP advisory format, several language versions of the same advisory can be stored within one file. As a result, all language-independent fields have to be maintained only once; it is only the language-dependent fields for which several versions are supplied. Basically, all entries that are described as containing either *free text* or *formatted text* can be supplied in multiple languages (see the XML description in Section 5.2 for further details.)

4.1. Identification Data

Both issuers and recipients of advisories have to manage an ever-growing set of advisories. It is therefore important that a given advisory can be uniquely identified and referenced within a set of (EISPP) advisories. To do so, there is a need for information about the issuing CERT and the reference number of the advisory. The advisory's title is also included into the identification data, because it is the most useful field for readers to recognize an advisory.

(a) Issuer

Issuer of the advisory.

Content type

List-of-values.

Content Description

An identifier for the (EISPP) CERT that issued the advisory. For FIRST members, the identifier should coincide with the short-name of the advisory as given in the FIRST member information (http://www.first.org/team-info/). Considering the EISPP-member CERTS, the issuer-field can be described as follows:

```
<issuer>       ::=   CERT-IST | esCERT-UPC | SBS | Siemens-CERT
```

(b) Reference Number

An advisory reference number.

Content type

Language-independent text.

Content Description

Each advisory must have a unique reference number that should not change during the life time of the advisory. The format of this field is defined by the policy of every issuer.

Further comments

Usually, the serial number is a combination of

- the year.
- a serial number.
- an identifier about the information type (e.g., Cert-IST uses the same reference number scheme for several document types, marking advisories with the tag AV, while Siemens CERT distinguishes four different lines of advisories (basically Windows, Unix, Network Equipment, and Miscellaneous) within the reference number.

Examples of reference numbers are

- *CERT-IST/AV-2002.217:* The 217th advisory by CERT IST in 2002
- *PC 42/02*: The 42nd advisory dealing with PCs (basically, machines running MS Windows) issued by Siemens CERT in 2002.

(c) Language

Information about the default language of the advisory.

Content type

Identifier conform to RFC 1766 ('en' for English, 'fr' for French, 'de' for German, 'it' for Italian', 'es' for Spanish, etc.)

Content Description

As mentioned above, all fields within the advisory that are described either as *free text* or *formatted* text can be given in several languages. The top-level language field determines which language is chosen as the default language of the advisory.

(d) Title

The advisory's title.

Content type

Free text (preferably less than 80 characters such that the title fits nicely on one line, e.g., the subject line of an email.)

Content Description

The title of an advisory should tell the reader in one sentence what the advisory is about. It should include information about (1) the affected platform and software, and (2) the vulnerability.

(e) Abstract (OPTIONAL FIELD)

A short abstract.

Content type

Free text (not more than one or two sentences.)

Content Description

The abstract should complement the information given in the title; the idea is that by reading the title and the abstract, the reader already has a pretty good idea about the contents of the advisory. This may be useful, for example, when displaying a list of advisories on a web page: giving only the title may be too little information, forcing the reader to click on each advisory.

4.2. History Data

Advisories are not issued into a void—usually there is a history of older advisories, some of which may be complemented or even superseded by the new advisory. Also, advisories may be revised, which means that there is a version history to be maintained.

The EISPP common advisory format provides two fields, one for the version history and one for keeping track of how an advisory relates to other advisories of the same issuer.

(f) Version History

Information about the advisory's current version/revision, along with history information and change dates.

Content type

Non-empty list with entries consisting of three (optionally four) fields:

```
<version_info>      ::= (<version_nr><date><change_descr>
                         [internal_comments])+
<version_nr>        ::= <version>.<revision>
<version>           ::= number
<revision>          ::= number
<date>              ::= yyyy/mm/dd
<change_descr.>     ::= free text
<internal_comments> ::= free text
```

Content Description

Version information is given as a non-empty list of a structured entry consisting of the version number (field `<version_nr>`), the date (field `<date>`), a short description of which changes have been carried out (field `<change_descr.>`), possibly some internal comments of the issuing CERT (e.g., author information).

Note the following important points:

- The list must be ordered: earliest changes are listed after latest changes, such that the latest change is always at the beginning of the list, and the information about the advisory creation at the end of the list.
- The EISPP CERTs have agreed on the following policy for version numbering:
  - Versions of form `0.x` are for draft advisories that have not yet been released
  - The first public release of an advisory to the readers always has version `1.0`.
  - Minor changes within an advisory that do not lead to a re-release of the advisory only increment the revision (e.g., from `1.0` to `1.1`).
  - Major changes within an advisory that lead to a re-release of the advisory lead to a new version, i.e., the version part is incremented by one and the revision part is set to zero (e.g., from `1.1` to `2.0`).

Further comments

The following information can be extracted from the version information:

- version number and date of the latest version (extractable from the head of the `version_info` list.)
- date of advisory creation (last element of the `version_info` list)
- date of first public release (extractable from the item in the `version_info` list of version `1.0`.)

(g) Update Information

Information about the relation of the advisory to prior/later advisories of the same issuer.

Content type

Non-empty list of reference numbers and associated tags describing the relationship:

```
<relation_info>    ::= (<relation_tag> <ref_num>)*
<relation_tag>     ::= complements | complemented_by
                     | supersedes | superseded_by
<ref_num>          ::= advisory reference number (see field 0)
```

Content Description

If updates to advisories are given in the form of new advisories rather than modifications to an existing advisory, a reference to the updated advisory is needed. At the same time, advisories that have been updated should be marked as such within the advisory database. Otherwise, when browsing the database, it may be difficult to see whether more recent information regarding a given advisory is available:

complements: An update of an advisory can mean that complementary information is published in a separate advisory—the old and new advisory should be read together. In this case, we say that the new advisory *complements* the older advisory (which itself is *complemented by* the newer one).

supersedes: An update of an advisory can mean that the newer advisory *supersedes* the older one—the older advisory can be completely discarded, as it is *superseded by* the newer one.

Information about complementing and superseding advisories is given as tagged lists of reference numbers: the `<relation_tag>` specifies the relation in which the present advisory stands to the advisories referenced by a reference number (`<ref_num>`) that follows the relation tag.

4.3. Vulnerability Classification

The vulnerability classification helps the reader to quickly assess the nature and danger of the described vulnerability. This section presents the information defined by the EISPP common format to classify the advisories.

(h) Confidence level

A rating of the reliability of the vulnerability classification.

Content type

A list-of-values:

```
<confidence level> ::=   official+tested | official | tested
                       | probable | not_qualified
```

Content Description

The confidence level is set according to the following criteria:

- **Official and tested vulnerability:** The vulnerability has been released by an official authority (CERT, CIAC, etc.) or by a vendor. It was also successfully tested by the issuer or somebody trusted by the issuer (e.g., another EISPP CERT).
- **Official vulnerability:** The vulnerability has been released by an official authority (CERT, CIAC, etc.) or by a vendor.
- **Tested vulnerability:** The vulnerability has not been released by an official authority or a vendor, but it was successfully tested by the issuer or somebody trusted by the issuer (e.g., another EISPP CERT).
- **Probable vulnerability:** :The vulnerability has not been released by an official authority or vendor, but is highly probable (cross-checked between several information sources).
- **Not qualified vulnerability:** The vulnerability has not been released by an official authority or a vendor, and could neither be tested nor crosschecked, but its criticality justifies an advisory, which must be taken "with caution".

(i) Vulnerability Category

A very short description of the vulnerability.

Content type

Free Text.

Content Description

The vulnerability category is supposed to give the reader a first impression of what the vulnerability is about. Typical examples of entries are "Buffer overflow", "Cross site scripting", etc.

(j) Attack Requirements

Technical requirements needed by an attacker to exploit the vulnerability.

Content type

List-of-values:

```
<requirement>       ::=  remote_no_account_standard_service
                       | remote_no_account_exotic_service
                       | remote_with_account
                       | physical_access
                       | not_rated
```

Content Description

The attack requirements are set according to the following criteria:

- **`remote_no_account_standard_service`:** No particular resources are necessary; an attacker can work remotely without needing an account on the system; the vulnerable service aimed at (targeted) is "standard" (e.g., smtp, http, ftp, telnet, etc.)

- **`remote_no_account_exotic_service`**: No particular resources are necessary; an attacker can work remotely without needing an account on the system; the vulnerable service is "exotic".
- **`remote_with_account`**: An attacker can work remotely, but needs an account to exploit the vulnerability.
- **`physical_access`**: The hacker needs physical access to the system.
- **`not_rated`**: The issuer chose not to (or was not in a position to) rate the attack requirements in this advisory.

(k) Vulnerability Impact

Information about the impact for the targeted system if the vulnerability exploitation succeeds.

Content type

A comma-separated list of impact types, which follows the following grammar:

```
<impact> ::= take-control | gain-privilege | get-access | Dos
           | integrity | confidentiality | disrupt-service
           | leverage | hiding | not_rated
```

Content Description

The following table describes the various impact types and ranks their severity (from "1" for the highest to "4" for the lowest severity level). If a vulnerability has several possible impacts, its severity level corresponds to the highest level amongst all its impacts.

Impact	Description	Severity
Take control	An attacker (remote or local, depending on resources needed) can gain full administrative privileges on the vulnerable equipment.	1
Gain (limited) privileges	Gain more privileges that initially granted.	2
Get (limited) access	Get an unprivileged access to the system. Get access to a service he/she should not have access to.	2
DoS	An attacker is able to totally disrupt a service (DoS) or all the services.	3
Integrity	An attacker can make important damage to application data integrity.	3
Confidentiality	An attacker can make important damage to application data confidentiality.	3
Disrupt service	partial DoS (e.g. : a local user may reboot the system).	4
Leverage	An attacker obtains information on the targeted system or is now in a position to launch more efficient attacks.	4
Hiding	An attacker can make important damage to system data integrity (e.g. :erasing traces).	4
Not rated	The issuer has not rated the impact in this advisory.	—

(l) Attack Expertise

Level of knowledge for the attacker to exploit the vulnerability.

Content type

List-of-values:

```
<expertise>          ::= beginner | skilled | expert | not_rated
```

Content Description

The attack expertise is set according to the following criteria:

- **`beginner`**: It is sufficient for an attacker to use a script or tool available over the Internet, without any specific knowledge in the area.
- **`skilled`**: An attacker needs certain skills such as modifying an existing exploit.
- **`expert`**: An attacker needs expert skills.
- **`not_rated`**: the issuer chose not to (or was not in a position to) rate the attack expertise in this advisory.

(m) Risk

The "risk" criteria are an overall assessment of the vulnerability.

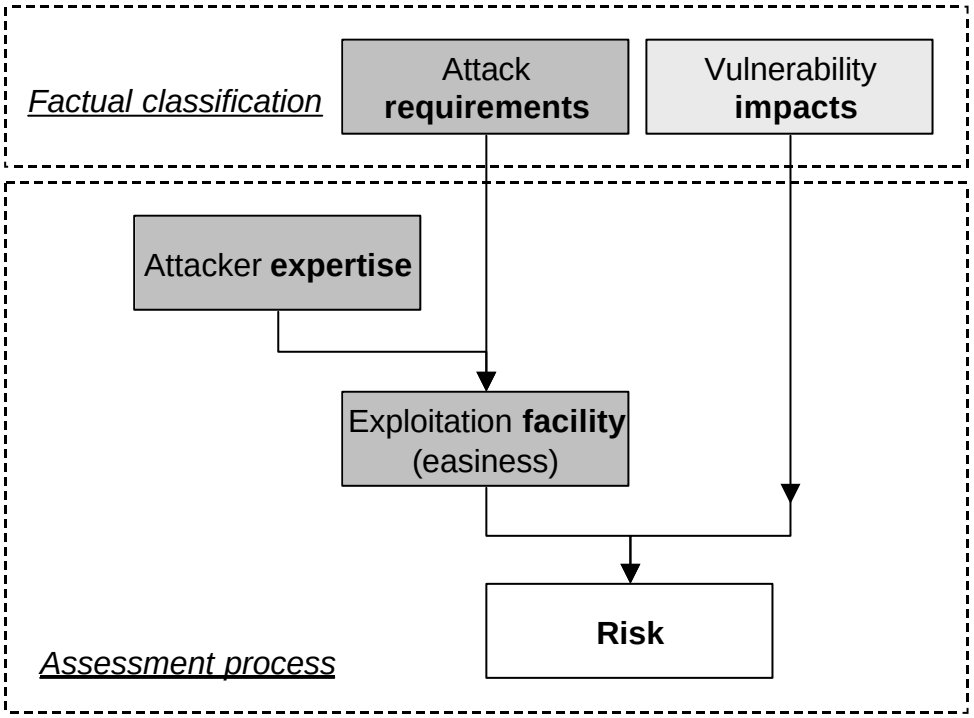
Content type

List-of-values:

```
<risk> ::= very-high | high | medium | low | not_rated
```

Content Description

The following diagram shows how the various criteria presented up to now are combined into a unique value named "risk".



The following table suggests a method for rating the risk of a vulnerability based on attack requirements, attack impacts and attack expertise. The first table derives a measure for *exploitation facility* from attack requirements and attack expertise:

	<requirement>			
<expertise>	remote, no account, standard service	remote, no account, exotic service	remote, with account	physical access
beginner	Trivial	Easy	Medium	Difficult
Skilled	Easy	Medium	Difficult	Very-difficult
Expert	Difficult	Difficult	Very-difficult	Very-difficult

	Impact Severity			
Exploitation facility (easiness)	Take control	Get limited access Gain limited privilege	DoS Integrity impact Confidentiality impact	Disrupt service Leverage Hiding
Trivial	Very-high	High	High	Medium
Easy	Very-high	High	High	Medium
Medium	Very-high	High	Medium	Medium
Difficult	High	Medium	Medium	Low
Very-difficult	High	Medium	Low	Low

Further comments

This proposed way of rating the risk is compliant with the "Risk = Impact * likelihood" paradigm. The "risk" indicates to the reader how important the vulnerability is, and how urgently appropriate measures must be taken to counter the threat. Possible guidelines are the following:

Risk	**Recommendation**
Very-high	Act immediately on all systems
High	Act immediately on front-end systems and servers
Medium	Action can be delayed, but a security maintenance operation must be scheduled now.
Low	Action can be delayed until the next scheduled maintenance operation

4.4. System Information

The crucial question for assessing whether an advisory may be relevant for a given environment depends on the system affected by the vulnerability described in the advisory.

(n) System Information

Information about affected platforms and systems.

Content type

A structured field, which gives both informal (i.e., formatted text) and formal information about the affected systems.

```
<system_info>          ::= <affected_platform> <affected_software>
                           <remarks> <system_id_list>
<affected_platform>    ::= formatted text
<affected_software>    ::= formatted text
<remarks>              ::= formatted text
<system_id_list>       ::= (<model_tag><systems>)
<model_tag>            ::= tag for identifiying the categorization model used in the id
                           list
<systems>              ::= list of affected systems; the exact form of this list is defined
                           by a categorization model
```

Content description

More than for any of the other fields, system information is important for advisory distribution: if system information is kept in a machine-readable format, then filtering mechanisms can be used to distribute security advisories only to readers with systems that might actually be affected. On the other hand, machine-readable system information is probably unsuitable for displaying it directly within the advisory, which will be read by humans after all.

The EISPP advisory format tries to strike a balance by providing (1) formatted text fields to inform the reader about which systems are affected, and (2) a field for machine-readable information about affected systems.

We first describe the free-text fields:

`<affected_platform>`: The affected platform; a platform is either an operating system (e.g., Redhat Linux, or MS Windows XP), a list of operating systems, a family of operating systems (e.g., Unix), or hardware (e.g., CISCO).

`<affected_software>`: The affected software; a software may be a program (e.g., MS Excel, or Apache), but also an OS-service (e.g., telnet, or finger.)

`<remarks>`: Remarks, such as about systems that may be affected or are known not to be affected.

Now for the machine-readable field `<system_id_list>`: At present, EISPP has not yet defined a common categorization model. Therefore, a `<model_tag>` is used to identify the categorization model that has been used for describing the affected systems in the `<systems>` field.

4.5. Description

Vulnerability classification and system information enable the reader to quickly assess whether (1) the advisory might be relevant in his environment and (2) how quickly he should react. What follows is a description of the vulnerability in several fields (most of which are optional).

(o) Publication Context (OPTIONAL FIELD)

Information that puts the advisory into context.

Content Type

Formatted text.

Content Description

Examples for possible entries regarding the publication context are

- information as to what triggered the release of the present advisory (e.g., the release of a patch by a vendor)
- information about the relation of the present advisory to former advisories of the same issuer (e.g., for an advisory that complements an older one, what the new bit of information is).

(p) Technical Context (OPTIONAL FIELD)

Information that helps the user to understand the technical context of the advisory.

Content Type

Formatted text.

Content Description

Often, already the technical context of the vulnerability, e.g., if an exotic service is affected, needs explanation, which should be provided in this field. This field could be useful to explain technical concepts required to understand the vulnerability.

(q) Description

Description of the vulnerability/vulnerabilities treated by the advisory.

Content Type

Formatted text.

Content Description

This description must be understandable by any readers and does not require extended knowledge in IT and security. More technical description must be put in the "Technical information" field described below.

Except for that aspect, it is probably not possible to give very precise guidelines on how to fill out this field. The EISPP participants, however, all seem to favour short and concise descriptions.

(r) Technical Information (OPTIONAL FIELD)

Detailed technical information, targeted more at security experts than the average reader.

Content Type

Formatted text.

(s) Diagnostic (OPTIONAL FIELD)

Information to help the reader with diagnostics, i.e., to determine whether his system has the described vulnerability.

Content Type

Formatted text.

Content Description

Whether a given system is affected may not always be clear, despite the system information given earlier in the advisory. Two examples:

- The vulnerability concerns a Unix service that is unknown to all but the expert user.
- A vulnerability only applies, if the some obscure Windows file has a certain version (some readers might need assistance in locating the file and checking its version number.)

That field can also be used to indicate the kind of evidences an attack attempt may produce.

4.6. Solution

After the reader has understood the vulnerability and established which of his systems are affected, the question is how the vulnerability can be removed or at least alleviated.

(t) Problem Solution

Description of how the vulnerability can be removed/alleviated.

Content Type

List of structured solution fields:

```
<solutions>   ::= [sol_intro] <solution>*
<solution>    ::= (<sol_type> <sol_title> [sol_descr]
                   reference*)
<sol_type>    ::= patch | software_upgrade | workaround | other
<sol_intro>   ::= formatted text
<sol_title>   ::= free text
<sol_descr>   ::= formatted text
```

The <reference> field is used also in other sections; please refer to Section 4.8 for a detailed description.

Content Description

The contents of the solution section following the semi-formal grammar given above is explained

<solutions> An advisory may present several solutions, e.g., patch information and work arounds. If several solutions are presented, it may be desirable to start the solutions section with some introductory information (optional field <sol_intro>), followed by a list of solution entries.

<Solution> a solution has a short solution title (field <sol_title>), followed by a description of the solution (field <sol_descr>). After the description, a list of references can be given. The solution itself can be classified using the field <sol_type>—a distinction is made between solutions by patches, software upgrades, and workarounds.

<Reference> A reference provides a pointer to some resource "outside" the advisory such as other advisories, patches, etc. A reference can be given a (preferably short) title (field <ref_title>); the pointer itself is given as one or more URIs (field <uri>); all URIs should point to the same resource—this provides the possibility to point, for example, to local copies of the resource or different language versions of the resource (e.g., for advisories). Further information regarding size, checksum, etc. can also be given; this may be most useful when referencing patches. See Section 4.8 for a detailed description.

4.7. Vulnerability Identifiers and Additional Resources

The EISPP participants tend to issue short and concise advisories, hence additional references to additional sources of information will often be useful. Standardised identifiers for vulnerabilities such as CVE names do not provide additional information themselves (even though automatic links could be created), but help the reader to correlate vulnerability information from sources such as local or remote scanners with a collection of advisories.

(u) Vulnerability Identifiers

If the vulnerability or vulnerabilities treated in the advisory have CVE names or identifiers provided through some other de-facto standard such as Microsoft Q-numbers, these identifiers should be supplied in this field.

Content Type

The `<reference>` structure (described in Section 4.8) is used: standardized vulnerability identifiers are given as a list of references.

```
<vulnerability_id> ::= <reference>*
```

Content Description

Within each `<reference>` section, the following is done:

- the `<ref_type>` is set to value `vuln_id`.
- the `<issuer>` is set to the naming system that is used for the given vulnerability ID. Currently, the following identifiers for naming systems are used in a standardized way:

`BID`	Security Focus Bugtraq ID database entry
`CERT-VN`	CERT/CC Vulnerability Note
`CVE`	CVE Vulnerability Identifier
`MSKB`	Microsoft Knowledgebase Entry
`LOTUS-TN`	Lotus Notes Technical Note
`SUNBUG`	Sun Bug ID
`XF`	X-Force Vulnerability Database

- the `<ref_num>` is set to the given vulnerability identifier.
- the `<ref_title>` can be used to provide a short comment, e.g., identifying, which of several vulnerabilities described in the advisory the identifier refers to.
- with `<uri>`, a pointer to a resource related to the vulnerability identifier (e.g., the ICAT-database entry for a CVE number) can be given. It must be noted that the URI is not a mandatory field. Most of the time it will be omitted because it can be derived automatically from the issuer and reference-number fields.

Further comments

Giving standard identifiers only makes sense, if the identifiers are updated also *once* the advisory has been issued.

(v) Additional Resources

References to relevant material such as advisories.

Content Type

We use the `<reference>` structure (described in Section 4.8) and provide pointers to additional resources as a list of reference:

```
<additional_resources > ::= <reference>*
```

Content Description

The main use of the *additional references* field is to provide references to relevant material such as advisories published by other bodies. See Field (t) for further details about the `<reference>` structure.

4.8. Description of the reference-structure

A reference provides a pointer to some resource "outside" the advisory such as other advisories, patches, etc. A reference can be given a (preferably short) title (field `<ref_title>`); the pointer itself is given as one or more URIs (field `<uri>`); all URIs should point to the same resource—this provides the possibility to point, for example, to local copies of the resource or different language versions of the resource (e.g., for advisories). Further information regarding size, checksum, etc. can also be given; this may be most useful when referencing patches. The reference structure described is used in Fields (t),(u), and (v).

```
<reference>    ::= <ref_type> [issuer] [ref_num]
                   [ref_title] <uri>*
<ref_type>     ::= patch | software_upgrade | workaround
                 | technical_information
                 | vendor_advisory | eispp_advisory | other_advisory
                 | vuln_id
                 | other
<issuer>       ::= identifier of the issuer of the resource pointed to by the reference
<ref_num>      ::= reference number associated with the resource pointed to by the reference
<ref_title>    ::= free text
<uri>          ::= [size] [checksum] [checksum_alg] [language] standard URL
<size>         ::= size of the resource pointed to in bytes
<checksum>     ::= checksum of the resource pointed to
<checksum_alg> ::= identifier for the checksum algorithm used to calculate the checksum
<language>     ::= identifier for the language of the resource (e..g., an advisory) pointed to.
```

`<ref_type>`: The type of the resource pointed to by the reference. Most types are self-explanatory; the `vuln_id` value is described in field "vulnerability_id".

`<issuer>`: An identifier of the organization that issued the resource that is pointed to.The following identifiers are currently used in a standardized way:

AUSCERT	AUSCERT advisory
BID	Security Focus Bugtraq ID database entry
BUGTRAQ	Posting to Bugtraq mailing list
CALDERA	Caldera security advisory
CERT	CERT/CC Advisories
CERT	VN
CIAC	DOE CIAC (Computer Incident Advisory Center) bulletins
CISCO	Cisco security advisory
COMPAQ	COMPAQ Service Security Patch
CVE	CVE number
DEBIAN	Debian Linux Security Information
EEYE	eEye security advisory
FREEBSD	FreeBSD security advisory
HP	HP security advisories
ISS	ISS Security Advisory
LOTUS-TN	Lotus Notes Technical Note
MANDRAKE	Linux Mandrake advisory
MS	Microsoft Security Bulletin
MSKB	Microsoft Knowledge Base article
NETBSD	NetBSD Security Advisory
NTBUGTRAQ	Posting to NTBugtraq mailing list
OPENBSD	OpenBSD Security Advisory

REDHAT	Security advisories
SGI	SGI Security Advisory
SUN	Sun security bulletin
SUNBUG	Sun bug ID
SUSE	SuSE Linux: Security Announcements
XF	X-Force Vulnerability Database

<ref_num>: The reference number (if any) given to the resource by its issuer (e.g., for advisories of other references, the reference number of that advisory, or, for patches from Microsoft, the patch number.)

<size>: Information about the size of the resource that is pointed to (in bytes).

<checksum>: The checksum of the resource that is pointed to (as output by the checksum algorithm that was used).

<checksum_alg>: An identifier for the checksum algorithm that was used to calculate the checksum of the resource that is pointed to.

<language>: An identifier for the language of the resource that is pointed to (e.g., for advisories); language identifiers are given as described in RFC 1766.

5. XML FORMAT

5.1. Introduction

The advisory format, as presented in the previous section, has been translated into an XML DTD. In most cases, the translation from the semi-formal grammar given above into the XML DTD is straightforward; points that required special consideration were

- where to use XML elements and where to use XML attributes
- how to implement the multi-language feature
- how to treat lists-of-values (for XML-attributes, lists-of-values can be constrained via the DTD)
- which HTML-tags can be used within the *FormattedText* elements

The DTD is extensively commented; we hope that together with the description given in Section 4, it can be easily understood.

5.2. XML DTD

```
<?xml version="1.0" encoding="UTF-8"?>
<!--============================================================================-->
<!--=====EISPP Common Advisory Format, version 1.1        ===============-->
<!--============================================================================-->

<!--================== Type Definitions ==================================-->
<!-- The following type definitions only serve to make the DTD itself more
     readable; instead of writing "CDATA" all over the place, we can give
     an indication of exactly what kind of data should be entered. -->

<!-- A language code, as per [RFC1766] -->
<!ENTITY % LanguageCode "NMTOKEN">

<!-- Date information, format yyyy/mm/dd -->
<!ENTITY % Date "CDATA">

<!-- LOV. These are lists-of-values. Lists-of-values as entries rather
  than attributes cannot be constrained by the DTD. Still, such
  lists-of-values are standardized by the EISPP advisory format; see
  the end of this file for a complete list of such LOVs. -->
<!ENTITY % LOV "#PCDATA">

<!--============================================================================-->
<!--================== List-of-values Definitions (attributes) ==========-->
<!--============================================================================-->

<!-- LOVs that are implemented as attributes can be constrained via the
     DTD. We do this for those LOVs that can be expected to be rather
     static.-->


<!--Values for the relate attribute within the relation information    -->
<!ENTITY % attvals.relation_attr "
   ( complements | complemented_by | supersedes | superseded_by)
 ">

<!--Values for the sol_type attribute within the sol_sec    -->
<!ENTITY % attvals.sol_type_attr "
   (patch | software_upgrade | workaround | other)
 ">


<!--Values for the issuer attribute within the reference information    -->
<!ENTITY % attvals.ref_type_attr "
```

```
   (  patch | software_upgrade | workaround | technical_information |
      vendor_advisory | eispp_advisory | other_issuer_advisory |
      vuln_id |
      other)
 ">



<!--================== Meta data (attribute lists) =====================-->
<!--
XML tags can be associated with attributes. Here we define attribute lists
that

 - will apply to several tags, or
 - are important enough to show them at the beginning of this document
   rather than somewhere in the middle of it.

This way, we can make changes locally rather than hunting through the
whole document. Also, the DTD gets more readable.
-->

<!--Attributes of the EISPP-advisory element.-->
<!--
In general, the fixed values of these
attributes will change each time a new version of the DTD is released.
-->

<!ENTITY % attlist.eispp "
   version             CDATA                   #FIXED    '1.1'
">


<!--====================================================================-->
<!--==================Free text and formatted text======================-->
<!--====================================================================-->
<!--
We define two kinds of elements for storing language-dependent content:
"FreeText" for text without markup, and "FormattedText" for text with markup.

Language-dependent fields such as "Title" can have several FreeText or
FormattedText-entities as content, one for each language.

-->

<!ELEMENT FreeText (#PCDATA)>
<!ATTLIST FreeText
        xml:lang            %LanguageCode;          #IMPLIED
>


<!--
Before defining what is formatted text, we need to make some
definitions: We want to be able to use a few very basic html-elements
in formatted text, namely:

 - font changes (emphasis, strong emphasis, code font)

 - linebreaks

 - paragraphs

 - lists (unnumbered and numbered)

 - tables (very simple ones)

-->
```

```
<!--
 First we define some categories of markup:
 -->


<!-- We only allow headings h3 - h6, because h1 and h2 would
     produce too large headers and thus cause trouble with
     the overall presentation of the advisory.
 -->

<!ENTITY % heading "h3 | h4 | h5 | h6">

<!ENTITY % phrase "em | strong | code"> <!-- Font changes -->

<!ENTITY % inline "%phrase; | br"> <!-- Stuff within text: font change and line break
-->

<!ENTITY % Inline "(#PCDATA | %inline;)*"> <!-- normal text -->

<!ENTITY % lists "ul | ol "> <!-- unnumbered and numbered lists -->

<!ENTITY % block "p | %heading; | %lists; | table | pre"> <!-- text blocks:
paragraphs,
                                        preformatted text, lists, and tables -->



<!-- We are very restrictive: In a list, we only allow other lists, font changes,
     and linebreaks.
  -->

<!ENTITY % ListBody "(#PCDATA | %lists; | %phrase; | br)*">

<!-- Now we are in a position to define formatted text:

     We are very restrictive: we only allow block elements such as
     tables and preformatted text on the very toplevel of
     formatted text; they may not be contained within a list, for example
-->

<!ELEMENT FormattedText (#PCDATA | %inline; | %block;)*>
<!ATTLIST FormattedText
        xml:lang            %LanguageCode;          #IMPLIED
>

<!-- What follows are the definitions for lists, tables, etc. -->

<!ELEMENT table (tbody)>
<!ELEMENT tbody (tr)+>
<!ELEMENT tr (th|td)+>
<!ELEMENT th %Inline;>
<!ELEMENT td %Inline;>



<!ELEMENT em %Inline;> <!-- emphasis -->

<!ELEMENT strong %Inline;> <!-- strong emphasis -->

<!ELEMENT code %Inline;> <!-- program code -->

<!ELEMENT br EMPTY>
```

```
<!-- Preformatted text is always understood as code, ie., it
     should be typeset with a fixed-width font. We allow no
     markup whatsoever within "pre" -->

<!ELEMENT pre (#PCDATA)>

<!-- In a heading we allow only font changes -->

<!ELEMENT h3  (#PCDATA | %phrase;)*>

<!ELEMENT h4  (#PCDATA | %phrase;)*>

<!ELEMENT h5  (#PCDATA | %phrase;)*>

<!ELEMENT h6  (#PCDATA | %phrase;)*>


<!-- In a paragraph we allow normal text, no lists, tables, etc. -->

<!ELEMENT p %Inline;>

<!-- Unordered list -->

<!ELEMENT ul (li)+>

<!-- Ordered (numbered) list -->

<!ELEMENT ol (li)+>

<!-- list item -->

<!ELEMENT li %ListBody;>




<!--============================================================================-->
<!--================== EISPP advisory format ===================================-->
<!--============================================================================-->
<!--
Here we define the EISPP advisory format.

In its attributes we find some fields that the informal description
lists as identification data:
 - language :
   Making the language-information an attribute is standard for XML;
   putting it into the top element makes most sense. Because the
   format supports multiple-language content, the top-level language
   attribute defines the default language.

 - (EISPP-)issuer:
   Will not be displayed by most presentation-engines (which are for
   readers) and applies to the whole advisory
    ===> attribute in top-element.

-->

<!ELEMENT EISPP-Advisory (Id_Data,
                          History_Data,
                          Vulnerability_Class,
                          System_Information,
                          Description,
                          Solution,
                          Vulnerability_ID?,
                          Additional_Resources?)>
<!ATTLIST EISPP-Advisory
```

```
        %attlist.eispp;
        issuer CDATA #REQUIRED
        xml:lang            %LanguageCode;          #REQUIRED
>


<!-- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -->
<!-- - - - - - - - - -  Identification Data - - - - - - - - - - - - - - - - -->
<!-- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -->


<!ELEMENT Id_Data (ref_num, title, abstract?)>


<!-- - - - - - - - - - Ref_num- - - - - - - - - - - - - - - - - - - - - - - - - -->
<!ELEMENT ref_num (#PCDATA)>

<!-- - - - - - - - - - Title- - - - - - - - - - - - - - - - - - - - - - - - - -->
<!-- Here is the first occurrence of using FreeText to implement a multi-language
     feature.  -->

<!ELEMENT title (FreeText+)>

<!ELEMENT abstract (FreeText+)>

<!-- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -->
<!-- - - - - - - - - -  History Data - - - - - - - - - - - - - - - - - - - - -->
<!-- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -->

<!ELEMENT History_Data (version_history, update_information)>

<!-- - - - - - - - - - version_history- - - - - - - - - - - - - - - - - - - - -->
<!-- Several fields identified in the informal description make excellent
     attributes, namely:
      - version
      - date

  If we want to have multi-language features, one question is whether
  those should be used for the change_descr. Probably yes, because I might
  want to display this info also to my readers, in which case I have
  to give it in all languages that I support in my CERT. The internal
  comments, on the other hand, will be in the working language of the
  issuing CERT.
  -->

<!ELEMENT version_history (change_descr+)>

<!ELEMENT change_descr (FreeText+, internal_comment?)>
<!ATTLIST change_descr
        version  CDATA  #REQUIRED
        date            %Date;        #REQUIRED
>

<!ELEMENT internal_comment (#PCDATA)>

<!-- - - - - - - - - - update_information- - - - - - - - - - - - - - - - - - - -->
<!-- Here is the first occurrence, where an element occurs that is not
     explicit in the informal description. There we find

    <update information> :== (<relation_tag> <ref_num>)*

     What "update_pointer" does is to give a name to the group
             (<relation_tag> <ref_num>)

     We have turned the relation_tag into an attribute; it fits its
     role, and this way we can contrain it with the DTD.
```

```
 -->

<!ELEMENT update_information (update_pointer*)>

<!ELEMENT update_pointer (#PCDATA)>
<!ATTLIST update_pointer
        relation %attvals.relation_attr; "complements"
>



<!-- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -->
<!-- - - - - - - - -  Vulnerability Classification- - - - - - - - - - - -->
<!-- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -->

<!-- The vulnerability classification has lots of LOVs. We could make
     them attributes, and constrain them with the DTD. But they are
     by nature more like real entries - - after all, they will be printed
     in almost any presentation. Furthermore, the impact can be a list
     of impacts, which a sum-type attribute cannot cater for.
     Conclusion: We make the classification data entries rather than
     attributes.

     We take the following strategy with respect to fields that not all
     CERTs might want to use:
       - all such fields are optional: if a CERT does not use a field
         on a regular basis, it just leaves it away.
       - the LOVs of fields have, where appropriate, a value that
         says "not_rated".

     For the allowed LOVs, see at the end of this file.
  -->



<!ELEMENT Vulnerability_Class (confidence_level?,vuln_cat?,impact*,
                               attack_expertise?,attack_requirements?,risk?)>

<!-- - - - - - - - - - confidence_level- - - - - - - - - - - - - - - - - - -->
<!ELEMENT confidence_level (%LOV;)>


<!-- - - - - - - - - - vuln_cat- - - - - - - - - - - - - - - - - - - - - - -->
<!ELEMENT vuln_cat (FreeText+)>


<!-- - - - - - - - - - impact- - - - - - - - - - - - - - - - - - - - - - - -->

<!ELEMENT impact (%LOV;)>

<!-- - - - - - - - - - attack_expertise- - - - - - - - - - - - - - - - - - -->
<!ELEMENT attack_expertise (%LOV;)>

<!-- - - - - - - - - - attack_requirements- - - - - - - - - - - - - - - - -->
<!ELEMENT attack_requirements (%LOV;)>

<!-- - - - - - - - - - risk- - - - - - - - - - - - - - - - - - - -->
<!ELEMENT risk (%LOV;)>

<!-- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -->
<!-- - - - - - - - -  System Info - - - - - - - - - - - - - - - - - - - - -->
<!-- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -->

<!ELEMENT System_Information (affected_platform, affected_software,
                              remarks?,system_id_list?)>
```

```
<!-- - - - - - - - - - -affected_platform- - - - - - - - - - -->
<!ELEMENT affected_platform (FormattedText+)>

<!-- - - - - - - - - - -affected_software - - - - - - - - - -->
<!ELEMENT affected_software (FormattedText+)>

<!-- - - - - - - - - - -remarks- - - - - - - - - - - - - - - -->
<!ELEMENT remarks (FormattedText+)>

<!-- - - - - - - - - - -system_id_list- - - - - - - - - - -->
<!-- For  machine-readable system information, a so-called
     categorization model is necessary. Such a model must
     describe with which machine-readable system identifiers
     the systems supported by a CERT are denoted.

     Since we have no standard categorization model (yet),
     in this version of the advisor format we add an attribute "cat_model"
     that says which categorization model has been used.

     We can, however be reasonably sure that whatever
     categorization model is used, we will have to deal
     with a list of things ...
  -->

<!ELEMENT  system_id_list (system_id)*>
<!ATTLIST system_id_list
        cat_model   CDATA #IMPLIED
>

<!ELEMENT system_id (#PCDATA)>

<!-- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -->
<!-- - - - - - - - - - -  Description - - - - - - - - - - - - - - - - - - - -->
<!-- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -->

<!ELEMENT Description (publication_context?,technical_context?,
                       description,technical_information?,diagnostic?)>


<!-- - - - - - - - - - -publication_context- - - - - - - - - - - - - - - -->
<!ELEMENT publication_context (FormattedText+)>

<!-- - - - - - - - - - -technical_context - - - - - - - - - -->
<!ELEMENT technical_context (FormattedText+)>

<!-- - - - - - - - - - -description - - - - - - - - - -->
<!ELEMENT description (FormattedText+)>

<!-- - - - - - - - - - -diagnostic - - - - - - - - - -->
<!ELEMENT diagnostic (FormattedText+)>

<!-- - - - - - - - - - -technical_information - - - - - - - - - -->
<!ELEMENT technical_information (FormattedText+)>

<!-- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -->
<!-- - - - - - - - - - -  Solution - - - - - - - - - - - - - - - - - - - - -->
<!-- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -->

<!-- sol_section gives a name to the grouping
  ( <sol_type> <sol_title> [ <sol_descr> ] <reference>* )
-->

<!ELEMENT Solution (sol_intro?, sol_section*)>

<!-- - - - - - - - - - -sol_intro- - - - - - - - - - - - - - - - -->
<!ELEMENT sol_intro (FormattedText+)>
```

```
<!-- - - - - - - - - - -sol_section  - - - - - - - - - -->
<!-- sol_type makes a good attribute -->

<!ELEMENT sol_section (sol_title, sol_descr?, reference*)>
<!ATTLIST sol_section
        sol_type  %attvals.sol_type_attr; #IMPLIED
>

<!-- - - - - - - - - - -sol_title - - - - - - - - - -->
<!ELEMENT sol_title (FreeText+)>

<!-- - - - - - - - - - -sol_descr - - - - - - - - - -->
<!ELEMENT sol_descr (FormattedText+)>




<!-- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -->
<!-- - - - - - - - - -  Vulnerability_ID- - - - - - - - - - - - - - - - - -->
<!-- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -->

<!ELEMENT Vulnerability_ID (reference+)>


<!-- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -->
<!-- - - - - - - - - -  Additional resources- - - - - - - - - - - - - - - -->
<!-- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -->

<!ELEMENT Additional_Resources (reference+)>


<!-- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -->
<!-- - - - - - - - - -  reference- - - - - - - - - - - - - - - - - - - - -->
<!-- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -->

<!-- Many of the fields for "reference" identified in the informal
     description make good attributes -->

<!ELEMENT reference (ref_title?, uri*)>

<!ATTLIST reference
        ref_type %attvals.ref_type_attr; #REQUIRED
        issuer   CDATA #IMPLIED
        ref_num  CDATA #IMPLIED
>

<!-- - - - - - - - - - -ref_title - - - - - - - - - - - - - - -->
<!ELEMENT ref_title  (FreeText+)>

<!-- - - - - - - - - - -uri - - - - - - - - - - - - - - -->
<!ELEMENT uri  (#PCDATA)>


<!ATTLIST uri
        xml:lang      %LanguageCode; #IMPLIED
        size          CDATA #IMPLIED
        checksum      CDATA #IMPLIED
        checksum_alg  CDATA #IMPLIED
>



<!-- - - - - - - - - - LOVs - - - - - - - - - - - - - - - - - - - - - - - - -
```

```
* LOV describing the issuer within the EISPP-advisory element.

  The attribute that denotes a CERT corresponds to the CERT's name as
  given in the FIRST member information (www.first.org/team-info/).

  (Since we use an attribute, we could constrain this LOV via the DTD.
   However, this LOV will be rather dynamic, and thus probably should
   not be part of the DTD.

  At the moment, the LOV is

  <issuer> ::= CERT-IST | esCERT-UPC | SBS | Siemens-CERT | other

  <confidence level> :== official+tested | official | tested | probable | not_rated

* <impact_type> :==  take_control | gain_privilege | get_access | DoS | integrity |
                       confidenciality | disrupt_service | leverage | hiding |
not_rated

* <attack expertise> :== beginner | skilled | expert | not_rated

* <attack requirements> :== remote_no_account_standard_service |
                             remote_no_account_exotic_service |
                             remote_with_account | physical_access | not_rated

* <risk> :== very-high | high | medium | low | not_rated



- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -->
```

6. COMPARISON WITH OTHER INTERCHANGE FORMATS

6.1. Other Candidates

The project identified three other "projects" dealing with related issues :

- To the Project's knowledge, there is only one other formal definition for the exchange of advisory information that is being actively maintained. The Common Advisory Interchange Format (CAIF) is a proposal from Rechenzentrum Universitat Stuttgart CERT (RUS CERT).

  See http://cert.uni-stuttgart.de/projects/caif/

  Oliver Goebel presented this project to European CERT teams at the 6th TF-CSIRT Meeting 2002 in Copenhagen.

- A project of interest is the Incident Object Description and Exchange Format (IODEF) produced by J. Arvidsson, A. Cormack, Y. Demchenko, J. Meijer produced with assistance from TERENA.

  See http://www.terena.nl/tech/task-forces/tf-csirt/iodef/index.html

  The object of this work is to define a common data format for the description, archiving, and exchange of information about incidents between CERTs. It is currently being developed within the Internet Engineering Task Force's Request For Comments (RFC) standards development process.

- Another project is the Intrusion Detection Message Exchange Format (IDMEF) that is designed for Intrusion Detection Systems to exchange information.

  See http://www.ietf.org/html.charters/idwg-charter.html

  It is currently being developed within the Internet Engineering Task Force's Request For Comments (RFC) standards development process.

6.2. Lessons learned from Other Candidates

All the projects above use XML Document Type Definitions to formally describe their data formats. This is the approach adopted by WP3.

The IODEF and IDMEF projects are both linked to the Internet Engineering Task Force's Request For Comments (RFC) standards development process. The CAIF project intends to become part of this standards process. This process ensures widespread publicity and peer review of a project. The EISPP exchange format is currently not a part of the RFC process.

The EISPP project deadlines are very tight and the RFC process would add dependencies to external groups that could interfere with project delivery plans. The EISPP project is peer reviewed by project partners from five EU countries. Once the format is established within EISPP and has undergone further internal review, an RFC process for a later version may be one possibility.

6.3. Current status of CAIF

Currently the CAIF project has produced a requirements document available in various formats from http://cert.uni-stuttgart.de/projects/caif. RUS-CERT is still working on a format specification so a direct comparison with the EISPP interchange format is not possible. They are also producing author's guidelines and reader's guidelines. This comparison of the CAIF and EISPP exchange formats is based on the requirements documentation and presentation materials.

Neither CAIF nor the EISPP format specify a standard naming system for affected systems, a so-called "categorization model", yet. Within CAIF, work seems to be under way towards defining such a model. It is very likely that a CAIF proposal for a categorization model would be adopted by EISPP if the proposal is also workable within the EISPP context.

6.4. Detailed Comparison of data in CAIF and EISPP formats

The following table summarizes the contents of the CAIF and EISPP exchange formats. More detailed notes below explain the similarities and differences.

CAIF	EISPP	Comment
[Platform/Product/Protocol] Subject	Title	
Source	Additional Resources	
Issue Date		Stored in Version History
Advisory ID including an Issuer ID	Reference Number	See note 1 below
	Issuer	
Version	Version History	See Note 2 below
	Update Information	
Abstract	Abstract	
Affected System	System Information	See Note 3 below
UnAffected System (Optional)		See Note 3 below
Attack Vector		See Note 4 Below
	Attack requirements	
Impact	Vulnerability Impact	
Vulnerability Class	Vulnerability Category	
	Attack Expertise	See Note 5 below
Severity	Risk	
Context (Optional)	Technical Context (Optional)	See Note 6 below
	Publication Context (Optional)	
Description	Description	
Vendor Status (Optional)		See Note 7 below
	Confidence Level	See Note 7 below
Determination Of Vulnerability	Diagnostic (Optional)	
Solution	Problem Solution	See Note 8 below
Workaround	Problem Solution	See Note 8 below
Vulnerability ID	Vulnerability Identifiers	
More Information on this Issue	Additional Resources	
Other Documents		See Note 9 below

Note 1: It is not clear if those are two fields in the CAIF format or one. The EISPP approach is to split these two fields to allow easier identification of the issuer without having to parse the Advisory ID.

Note 2: The CAIF format includes a version number. The EISPP format has a version history that lists the changes to an advisory. There is a suggestion that the CAIF format will include a "supersedes" field to indicate the advisory that is replaced by the current one. EISPP's version history allows this relationship to be modelled as well as other relationships that EISPP considers important. Typical common and useful relationships that can be modelled are "complements", "supersedes" and the reverse relationships "complemented by", "superseded by".

Note 3: The CAIF format has optional fields to inform about "maybe affected systems" and "not affected systems". Such data can be included into the Remarks field of the EISPP System Information field.

Note 4: The CAIF format specifies "Attack Vector" that describes the pre-requisites for an attack. The EISPP format does not explicitly include this field but the data can be included in technical information. EISPP does include a structured data field called "Attack Requirements" that provides an indication of the level of access required to use a vulnerability.

Note 5: The CAIF format does not include a structured data field indicating the required attack expertise. This is a very useful indicator of the threat posed by a vulnerability and it is included in the EISPP format.

Note 6: The EISPP format includes two fields that further classify the information relating to an advisory. The CAIF format does not have separate fields for publication_context and technical_context. These fields allow finer control over the sort of text displayed to recipients of advisories. The publication context field gives the Editor a chance to explain why the advisory is being published. It can be used to emphasise the high profile of a problem in the media or the recent development of an automated attack tool that uses the vulnerability. The Technical Context field allows end users to be shielded from excess technical detail.

If less knowledgeable end users are going to understand the advisory risk and impact ,the presentation must be well controlled and tailored to the audience. Finer grained categories of advisory text are necessary for this to be done effectively and automatically.

Note 7: The CAIF advisory includes an optional Vendor Status field. The EISPP format has incorporated some of this information in the values allowed for the confidence level. If there is an official vendor alert then the confidence level value can reflect this. Any other free text information relating to the Vendor Status can be included in the Problem Solution.

Note 8: The CAIF format has two fields for "Workaround" and "Solution" information that contain free-text information. The EISPP format sums up all information regarding possible solutions within the "Solution" field. Solutions can be tagged as "patch", "upgrade", or "workaround".

Note 9: The CAIF format description implies that CAIF format documents may additionally serve as "containers" for other documents that are related to the advisory. This facility is not provided for in the EISPP format, however extensive linking of documents is possible. This may be a facility to add to future releases of the EISPP format.

6.5. Comparison of text formatting in CAIF and EISPP format

The CAIF format supports text formatting that is not currently provided for in the EISPP format. Specific examples include Log file extracts and Terminal interaction. These two possibilities should be considered for future versions of the EISPP format.

7. ISSUES AND CONCLUSION

The exchange format for security advisories described in this document will be put to the test within EISPP: it will serve as a common basis for exchanging information, and collaborating on security advisories. Further revisions of the format will incorporate lessons learned through this cooperation.

The XML data-type description of this and future versions of the format, together with sample XSLT stylesheets for displaying advisory data are made publicly available on EISPP's website www.eispp.org.

The EISPP consortium invites everybody interested in the common format to download and use the materials. Comments and questions can be addressed via email to info@eispp.org

Document management information

Title	EISPP Common Advisory Format Description
Identifier	EISPP-D3-001-TR
Confidentiality[1]	PU
Status	A (Approved)
Creation Date	2002/07/30
Version	1
Revision	1
Revision Date	2003/02/06
Deliverable Reference	D3.01
Authors	Philippe Bourgeois, Oscar Conesa, Bernd Grobauer, Gareth Price
Keywords	EISPP, XML, Advisory, Format, Exchange

Approval Section

Company	*Alcatel CIT*	*BT Ignite*	*InetSecur*	*SIEMENS*	*UPC*
Date	2002/10/21	2002/10/14	2002-10-23	2002/10/15	2002-10-23
Comments	PGP	PGP	PGP	PGP	PGP
Approving Person	Michel Miqueu	Richard Jones	Oscar Conesa	Udo Schweigert	Manel Medina

[1] Confidentiality indicator according to the following table

PU	Public
PP	Restricted to other programme participants (including the Commission Services)
RE	Restricted to a group specified by the consortium (including the Commission Services)
CO	Confidential, only for members of the consortium (including the Commission Services)

Document History

Version	*Date*	*Reason for modification*	*No. of pages*		
			added	*modified*	*deleted*
0.1	2002-07-30	Document creation (bg)	all		
0.2	2002-08-12	Rewrite of vuln. class. section by pb; added appendix documenting relationship to CAIF.	§ 5	all	
0.3	2002-08-16	Changes according to telephone conference (bg)		all	
0.4	2002-08-26	Modifications: fields have been removed/changed. (oc)		all	
0.47	2002-09-22	Integrating in the document the successive changes discussed for the common format			
0.5	2002-09-27	Adding introductionnary chapters, the DTD, and the CAIF comparative	§ 1,2,5 and 6	all	
0.6	2002-10-09	Taking into account peer-reviewers comments		all	
1.0	2002-10-21	Approbation by EISPP partners		34, 35	
1.1	2002-02-05	Updated format description after discussions within WP3	all		