ICS 33 020
M 01

YD

中华人民共和国通信行业标准

YD/T 1827-2008

网络安全事件描述和交换格式

Network Incident Object Description Exchange Format

2008-07-28 发布　　2008-11-01 实施

中华人民共和国工业和信息化部　发布

目　次

前　言……II
引　言……III
1　范围……1
2　规范性引用文件……1
3　术语和缩略语……1
4　符号约定和格式……3
5　安全事件描述与交换格式的基础数据类型……3
6　安全事件描述与交换格式……5
7　安全事件描述交换格式的扩展和实现指南……32
附录 A（资料性附录）　事件描述实例……43
参考文献……48

前　言

本标准是参照国家标准和RFC有关文档，结合我国计算机网络安全和应急响应的特点制定的。

本标准的附录A为资料性附录。

本标准由中国通信标准化协会提出并归口。

本标准起草单位：国家计算机网络应急技术处理协调中心、清华大学

本标准主要起草人：袁春阳、段海新、周勇林、黄元飞、焦绪录、杨　臻、梁　晟、孙蔚敏、纪玉春、吴俊华、孙　彬

引　言

制定本标准的目的是为计算机安全应急响应组（Computer Security Incident Response Team，以下简称应急响应组或CSIRT）之间的安全事件交换提供统一的安全事件描述方法和交换格式。

随着互联网的发展，计算机安全事件突破了国家或地区的边界，跨越多个组织，各应急响应组织间的合作也突破了国界、语言和文化的约束。在我国，国家计算机网络与信息安全管理中心成立了国家计算机网络应急技术处理协调中心（简称CNCERT/CC），负责国内各部门、行业与机构的计算机安全应急响应组的协调工作。各商业网络运营商、大型公司、教育科研机构以及国家相关部门也逐步成立了计算机安全应急组。为了提高各CSIRT对计算机安全事件的响应能力和预防能力，规范我国各CSIRT之间计算机安全事件的描述和相关事件交换格式，特制定安全事件描述交换格式（Incident Object Description Exchange Format，IODEF）。

安全事件描述交换格式（IODEF）主要用于各应急响应组事件处理系统（Incident Handling System）之间的信息交换，是一种表示层的通信协议，它的应用环境如图1所示。

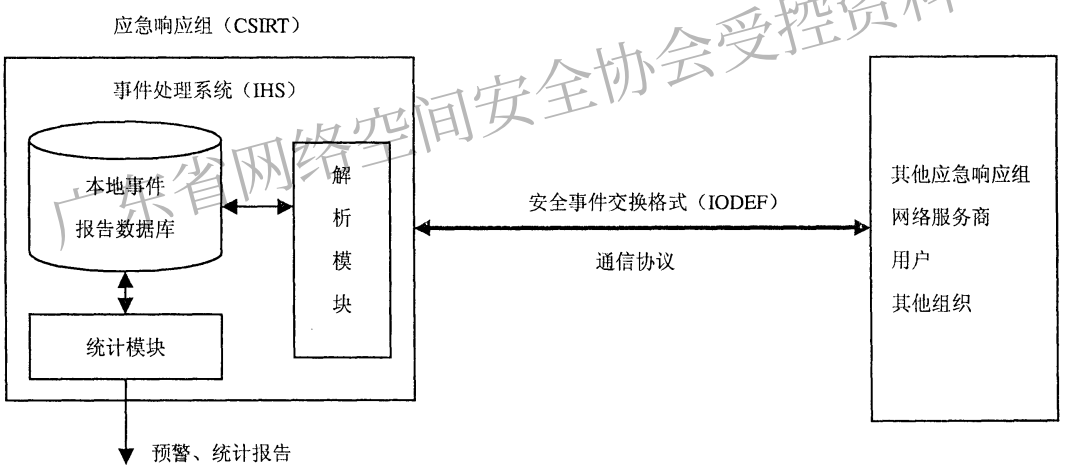


图 1　安全事件描述交换格式的应用环境

一般情况下，应急响应组需要某种软件工具把安全事件相关的信息生成IODEF的事件报告，然后通过任意的通信协议（比如HTTP、SMTP等）发送给其他相关的组织；当CSIRT收到其他CSIRT、网络服务商、用户或其他组织发送过来的IODEF文档时，一般需要经过事件处理系统中的IODEF解析模块或独立的IODEF解析程序生成符合CSIRT内部定义的数据格式，然后保存到本地事件报告数据库中，并进入事件处理的流程中。

网络安全事件描述和交换格式

1 范围

本标准规定了计算机安全事件描述和交换格式（IODEF）的术语、事件描述格式，并提供XML的参考实现。

本标准适用于在我国提供计算机安全事件应急响应服务的各种应急响应组，也可供其他建设、使用计算机安全事件处理系统或安全事件交换系统的组织参考。

2 规范性引用文件

下列文件中的条款通过本标准的引用而成为本标准的条款。凡是注日期的引用文件，其随后所有的修改单（不包括勘误的内容）或修订版均不适用于本标准。然而，鼓励根据本标准达成协议的各方研究是否可使用这些档的最新版本。凡是不注日期的引用文件，其最新版本适用于本标准。

GB/T 19716-2005	信息技术　信息安全管理实用规则（idt ISO/IEC 17799:2000）
GB/T 19715.1-2005	信息技术 信息技术安全管理指南 第1部分：信息技术安全概念和模型（idt ISO/IEC TR 13335.1:1996）
GB/T 19715.2-2005	信息技术 信息技术安全管理指南 第2部分：管理和规划信息技术安全（idt ISO/IEC TR 13335.2:1997）
ISO/IEC TR 13335.3:1998	信息技术 信息技术安全管理指南 第3部分：信息技术安全管理技术
ISO/IEC TR 13335.4:2000	信息技术 信息技术安全管理指南 第4部分：防护措施的选择
ISO/IEC TR 13335.5:2001	信息技术 信息技术安全管理指南 第5部分：网络安全管理指南
draft-ietf-inch-iodef-04	安全事件描述交换格式数据模型和XML实现
RFC 1305	网络时间协议规范和执行
RFC 2030	对于IPv4、IPv6和OSI的简单网络定时协议第4版
RFC 2256	对于使用LADPv3的X.509使用者计划的概述
RFC 2822	英特网信息格式
RFC 2396	统一资源标识符（URI）：一般句法

3 术语和缩略语

3.1 术语和定义

下列术语和定义适用于本标准。

3.1.1 攻击 Attack

指对系统安全的袭击，主要来源于人为的、技术上的威胁，例如，企图逃避安全服务和违背系统安全策略的一次技术上的攻击行为。

攻击可能是主动的，也可能是被动的；可能是内部人员，也可能是外部人员。

3.1.2 攻击者 Attacker

指为达到某种目的而尝试一次或多次攻击的个体。在IODEF中，攻击者由其网络标识，发起网络或计算机攻击的组织以及物理位置信息（可选）来描述。

3.1.3 计算机安全应急响应组 CSIRT

指处理计算机安全事件和创建安全事件报告的机构。CSIRT也可能涉及证据的收集和保管、安全事件请求等活动。CSIRT由其身份标识、机构名称、公开密钥等来描述。

3.1.4 损失 Damage

指攻击给目标系统产生的有意或者无意的后果。损害的描述可以包括对攻击的实际结果的自由形式的文本描述，如果可能，还可以包括有关被损害的系统、子系统或者服务的结构化信息。

3.1.5 活动 Event

指操纵目标的一种行为，其目的是引起目标的状态发生改变。从起源角度看，活动可以被定义为在引发报警的系统或网络中任何可观察到的现象。例如，在10秒钟内连续3次登录失败的活动，可能表示出现强行登录攻击事件。

3.1.6 证据 Evidence

指与活动（Event）相关的信息，该信息用来证明或支持活动相关的结论。对于安全事件（incident），可能包括但不局限于如下内容：由侵入检测系统（IDS）创建的数据转储（dump）文件、来自系统日志文件的数据、内核统计信息、高速缓存、内存、临时文件系统或者其他引起报警或在安全事件发生后收集的数据。

在存储、归档证据，特别是需要保持证据的完整性时，必须高度小心并采取特殊的规则，必要的时候，应当加密存储证据。按照证据收集和存档的原则，必须严格保护证据的安全。必须详细记录证据保管链，证据应当按照当地的法律进行收集、存档和保护是非常必要的。

3.1.7 安全事件 Incident

指涉及违反安全策略的安全性事件。安全事件可以定义为单次攻击或者一组攻击，可以根据攻击的方法、攻击者的身份、受害者、站点、目标和时间等特性将此单次攻击或此组攻击从其他的攻击中区分开来。

3.1.8 影响 Impact

用来描述根据用户或机构对攻击的结果的表述，例如资金上的损失或者时间花费等方面的代价。

3.1.9 目标 Target

指计算机或网络逻辑实体（如账号、进程或数据）、物理实体（组件、计算机、网络或国际互联网）。

3.1.10 受害者 Victim

指在安全事件报告中所描述的遭受到攻击的个人或组织。在IODEF中，受害者通常用其网络身份标识、组织或者物理位置等信息来描述。

3.1.11 漏洞 Vulnerability

指在系统的设计、实现或者运行和管理中的缺陷或弱点，这些缺陷或弱点可能会被利用，以突破系统的安全策略。

大多数系统都有某些类型的漏洞，但是这并不意味着系统不能使用。并不是每个漏洞都会导致攻击，也并不是每次攻击都会成功。攻击是否成功和漏洞的危险程度、攻击的力度以及采用应对措施的有效性有关。如果攻击需要利用的漏洞非常难实现，那么这样的漏洞是可以容忍的。如果攻击者从攻击中获得的收益非常小，此时即便是非常容易被利用的漏洞也是可以容忍的。然而，漏洞系统被大量的用户利用来实施攻击，此时某些攻击者可能从中获益不少。

3.1.12 安全事件处理系统 Incident Handle System

对计算机网络安全事件、资产、漏洞、威胁、风险、预警、安全策略、安全知识等安全要素进行收集、分析、管理，并提供计算机安全事件响应的流程管理软件系统。

3.1.13 XML 模式 Schema

一种基于XML的语法或规范，用来定义XML文档的标记方式，对XML文档的词汇表和语法进行约束和形式化。

3.2 缩略语

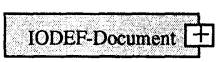
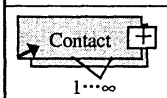
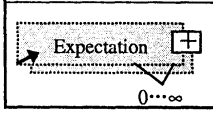
下列缩略语适用于本标准。

CSIRT	Computer Security Incident Response Team	计算机安全应急响应组
IODEF	Incident Object Description Exchange Format	安全事件描述与交换格式
IHS	Incident Handle System	事件处理系统
XML	Extensible Markup Language	可扩展的标记语言
DTD	Document Type Definition	文档类型定义
IDS	Intrusion Detection System	入侵检测系统
IDMEF	Intrusion Detection Message Exchange Format	入侵检测信息交换格式
CVE	Common Vulnerabilities and Exposures	通用漏洞披露，一种常见的漏洞描述字典
FQDN	Fully Qualified Domain Name	完整域名

4 符号约定和格式

本标准使用类图来描述数据模型。在类图中，各符号图例含义见表1。

表1 类图的图例说明

符号图例	含 义
	类IODEF-Document是聚合类，包含有子类
	类IncidentID是聚合父类必须包含的单个子类，只能存在一个实例
	类Contact是聚合父类必须包含的子类，可以存在多个实例，个数不限
	类DetectTime是聚合父类可能包含的子类，最多存在一个实例
	类AlternativeID是聚合父类可能包含的子类，最多存在一个实例，类AlternativeID本身是聚合类
	类Expectation是聚合父类可能包含的子类，可以存在多个实例，类Expectation本身是聚合类

5 安全事件描述与交换格式的基础数据类型

5.1 整数

由INTEGER数据类型表示整数属性，整数数据必须以10或者16为基底编码。

以10为基底的整数编码使用阿拉伯数字“0”到“9”，以及可选符号“+”或者“-”。例如，“123”、“-456”。

以16为基底的编码使用阿拉伯数字“0”到“9”，以及“a”到“f”（或者它们的大写形式），并且在前面加上字符“0x”。例如，“0x1a2b”。

5.2 实数

由REAL数据类型来描述实数（浮点）属性。实数数据必须以10为基底编码。

实数编码和POSIX函数例库中的“strtod”一样：一个可选符号后跟一个非空的小数位数串，可选地包含一个基数字符，然后是一个可选的指数部分。一个指数部分由一个“e”或者“E”，后跟一个可选的符号，接下来是一个或者多个小数位数。例如，“123.45e02”，“-567，89e-03”。

与本标准兼容的应用程序必须支持“.”和“，”基数字符。

5.3 字符和字符串

由CHARACTER数据类型来描述单字符属性，由STRING数据类型描述已知长度的多字符属性。

字符和字符串数据没有特殊的格式要求，除了偶尔需要使用转义字符来表示特殊的字符。

5.4 字节

字节数据类型BYTE用于描述二进制数据。

5.5 枚举类型

由ENUM数据类型描述枚举类型，枚举类型是由可接受的值构成的一个有序列表。每一个值代表一个关键字。在本标准中，枚举类型关键字被用作属性值。

5.6 日期－时间

由本标准的DATETIME数据类型描述日期－时间串。

5.7 NTP 时间戳

由NTPSTAMP数据类型描述NTP时间戳，在RFC 1305和RFC 2030 中有详细的描述。一个NTP时间戳是一个64比特的无符号定点数字。前32比特是整数部分，后32比特为小数（分数）部分。

IODEF文档必须将NTP时间戳编码为2个32比特的十六进制值，使用“.”分隔。例如，“0x12345678.0x87654321”。

5.8 端口列表

由PORTLIST数据类型描述网络端口列表，它由一个以逗号分隔的数字和范围（N-M表示端口号N至端口号M，包括M）的序列组成，可以在一个单独的序列中使用数字和范围的任意组合。例如“5-25，37，42，43，53，69-119，123-514”。

5.9 邮政地址

由POSTAL数据类型描述邮政地址。POSTAL数据格式在RFC 2256 的5.17 - 5.19有详细说明。其格式如下：

建筑物，街道，邮政编码，城市，国家；或者邮政信箱，邮政编码，城市，国家。

5.10 个人或组织

由NAME数据类型描述个人或者组织的名称。格式如下：

名 姓，或者姓 名

NAME数据类型的格式参见RFC 2256的5.4。

5.11 电话和传真号码

由PHONE数据类型描述电话号码。电话和传真号码遵循ITU推荐的表达格式：

+（国际电码）（本地代码）（电话号码）

PHONE数据类型的格式参见RFC 2256的5.21。

5.12 电子邮件

由EMAIL数据类型描述电子邮件地址。EMAIL数据类型的格式参见RFC 2822的3.4.1。

5.13 统一资源标识

由URI数据类型描述统一资源标识符（URI）。URI数据类型的格式在RFC 2396中说明。

5.14 惟一标识

由UID数据类型描述IODEF文档的某个特定创建者（例如某个CSIRT）的惟一标识符。由GUID数据类型描述全局惟一的标识符。UID和GUID数据类型是由字母数字串构成。

6 安全事件描述与交换格式

6.1 安全事件的描述方法

本章详细描述安全事件描述交换格式所定义的类（Class）。对于每一个类，首先给出其语义，并用类图来表现和其他类之间的关系，然后用XML的文档类型定义（DTD）和模式（Schema）两种形式给出该类的具体描述格式。

对于每个类的描述包括6个部分:

- 类说明：简要描述类的具体含义；
- 类图：以图形的方式说明类的构成；
- 子类：描述该类所包含的子类，是否是必需的，存在实例个数及其简要说明；
- 属性：用于说明该类所具有的属性名及其含义；
- Schema定义：给出该类XML Schema实现片段；
- DTD定义：给出该类XML DTD实现片段。

6.2 文档（IODEF-Document）

类说明

IODEF-Document类在IODEF数据模型是顶层类，所有IODEF文档都是IODEF-Document的实例。

类图

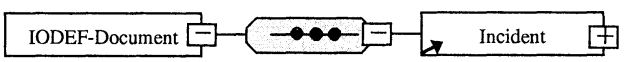


图2 IODEF-Document类

子类

- Incident：只能包括一个子类，包含所有与安全事件相关信息的安全事件类。

属性

version：必需，字符串。IODEF文档所遵循的本标准的版本号。本标准以下讨论的格式以IETF INCH工作组草案draft-ietf-inch-iodef-04.txt为参考，版本号为04。

Schema定义

```
<xs:element name="IODEF-Document">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="Incident"/>
        </xs:sequence>
        <xs:attribute name="version" type="xs:string" fixed="04"/>
```

```
        </xs:complexType>
    </xs:element>
```

DTD定义

```
    <!ELEMENT IODEF-Document （Incident）>
```

6.3 安全事件（Incident）

类说明

每一个报告给CSIRT或者由CSIRT处理的安全事件，由Incident类的一个实例来描述。Incident类为通常交换的安全事件数据提供一个标准的表示法，并且把所描述的活动和一个惟一的标识符联系起来。

Incident类概述安全事件活动以及某CSIRT信息处理的详细信息，也对构成incident的安全事件进行分类。

Incident的许多聚合类也会出现在EventData中，尽管出现的次数不同。然而，它们的语义是有区别的。Incident中的聚合类反映的是整个安全事件的相关信息，而EventData中的聚合类仅仅提供所描述的给定动作或者系统节点的相关信息。IncidentData类和EventData类的聚合类是互补关系。前者提供概要信息，而后者提供更加明确的细节。例如，在Incident中描述安全事件的总体影响可能是拒绝服务，但，在EventData描述中也可能会提及被彻底毁坏的机器。另一个例子，可以在IncidentData类中提供一个组织的联系信息，而在EventData类中提供更加明确的单个主机的联系信息。

类图

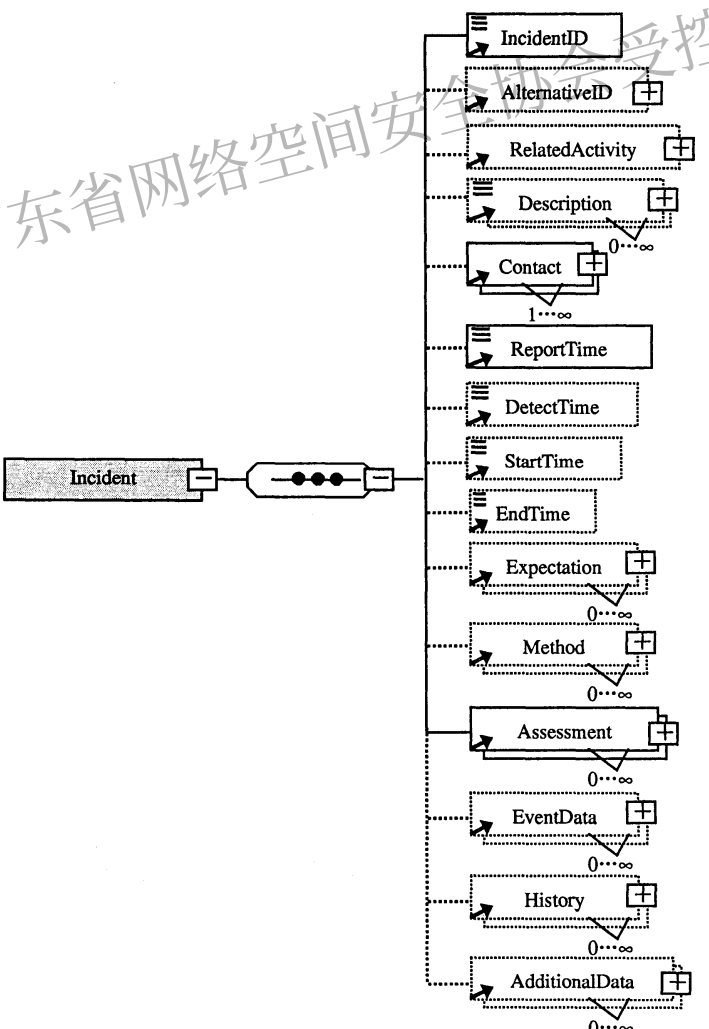


图3 Incident类

子类

— IncidentID：一个。文档的产生方指派给安全事件的事件跟踪号，或者惟一标识符。

— AlternativeID：零个或者一个。由其他CSIRT用来引用文档中所描述的同一活动的一列安全事件跟踪号。

— RelatedActivity：零个或者一个。引用相关安全事件的一列安全事件跟踪号。

— Description：零个或者多个，字符串类型。安全事件活动的自由形式的文本描述。

— Contact：一个或多个。安全事件有关的参与方的联系信息。

— ReportTime：一个。报告安全事件的时间。

— DetectTime：零个或者一个。安全事件活动最初被检测出的时间。

— StartTime：零个或者一个。安全事件活动开始的时间。

— EndTime：零个或者一个。安全事件活动结束的时间。

— Expectation：零个或多个。文档接收者将执行的预期动作。

— Method：零个或者多个。入侵者所使用的技术（譬如工具，漏洞）。

— Assessment：一个或者多个。评估安全事件活动影响的描述。

— EventData：零个或者多个。导致安全事件的活动数据的详细信息。

— History：零个或者一个。记录在处理安全事件的期间，发生的重要的事件或者采取的行动。

— AdditionalData：零个或者多个。使用不能在别的地方描述的信息来扩展数据模型的区域。

属性

purpose（目的）：必需，枚举类型。

说明：指出IODEF文档的目的。本属性被定义为一个枚举列表：

— handling：发送本IODEF-文档的目的是期望接收者处理安全事件；

— statistics：发送本IODEF-文档，只用于统计目的；

— warning：发送本IODEF-文档，只是作为一个警告；

— other：发送IODEF-文档目的将在AdditionalData元素中指明。

Restriction（限制）：可选，枚举类型。

说明：指出IODEF-Document的发送者期望接收者应该遵守的保密原则，当然文档的接收者自由决定是否遵守这个原则。逻辑上，子类可以继承父类的这个属性值。由于多数高层类都有restriction属性，这就有可能设置细粒度的保密策略。如果子类加紧或者放松保密规则，子类可以不考虑父类的保密规则。对一个没有指定restriction属性值的类，可以在其指定了restriction属性值的最邻近的祖先类中得出该类的restriction属性值。restriction属性被定义为一个枚举类型值，缺省值为“private”。

— public：对信息没有任何级别的限制；

— need-to-know：信息可以被和安全事件有关的其他方共享（举例来说，多个受害站点能够相互通告）；

— private：信息不能被共享；

— default：按照通信各方预先安排的信息保密规则，决定是否可共享信息。

Schema定义

<xs:element name="Incident">

```
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="IncidentID"/>
                <xs:element ref="AlternativeID" minOccurs="0"/>
                <xs:element ref="RelatedActivity" minOccurs="0"/>
                <xs:element ref="Description" minOccurs="0" maxOccurs="unbounded"/>
                <xs:element ref="Contact" maxOccurs="unbounded"/>
                <xs:element ref="ReportTime"/>
                <xs:element ref="DetectTime" minOccurs="0"/>
                <xs:element ref="StartTime" minOccurs="0"/>
                <xs:element ref="EndTime" minOccurs="0"/>
                <xs:element ref="Expectation" minOccurs="0" maxOccurs="unbounded"/>
                <xs:element ref="Method" minOccurs="0" maxOccurs="unbounded"/>
                <xs:element ref="Assessment" maxOccurs="unbounded"/>
                <xs:element ref="EventData" minOccurs="0" maxOccurs="unbounded"/>
                <xs:element ref="History" minOccurs="0"/>
                <xs:element ref="AdditionalData" minOccurs="0" maxOccurs="unbounded"/>
            </xs:sequence>
            <xs:attribute ref="restriction" default="default"/>
            <xs:attribute ref="purpose" use="required"/>
        </xs:complexType>
    </xs:element>
```

DTD定义

<!ELEMENT Incident （IncidentID，AlternativeID?，RelatedActivity?，Description*，Contact+，ReportTime，DetectTime?，StartTime?，EndTime?，Expectation*，Method*，Assessment+，EventData*，History?，AdditionalData*）>

6.4 事件标识（IncidentID）

类说明

事件标识（IncidentID）的内容代表一个安全事件跟踪号（UID），该UID在CSIRT的上下文中是惟一的。

类图

图4 IncidentID类

子类

无。

属性

restriction：可选，枚举类型，参见6.3中对这个属性的定义；

name：必需，GUID类型。产生IODEF-Document的CSIRT的标识符。

Schema定义

```
<xs:element name="IncidentID" type="IncidentIDType"/>
 <xs:complexType name="IncidentIDType" mixed="true">
    <xs:attribute name="name"/>
 <xs:attribute ref="restriction"/>
</xs:complexType>
```

DTD定义

```
<!ELEMENT IncidentID（#PCDATA）>
```

6.5 可选标识（AlternativeID）

类说明

可选标识AlternativeID类引用其他组织实体（例如其他CSIRT）的事件编号，用来在IODEF-Document中跟踪不同组织对同一安全事件的处理活动。因此，被列出作为AlternativeID的跟踪号的事件，是指由其他的CSIRT从不同的角度，检测到的同样的事件。

如果希望表示的不是同一个安全事件，而是相关的安全事件（譬如同样的的方法或者入侵者），则其安全事件跟踪号用在下面将要讨论的RelatedActivity类描述。

类图

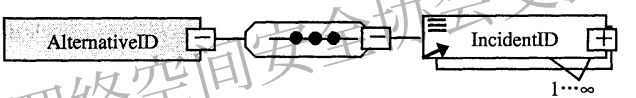


图5 AlternativeID类

子类

— IncidentID：一个或多个，表示由其他CSIRT分配给在IODEF-Document中描绘的同样的活动的惟一标识符。

属性

restriction：可选，枚举类型，请参考Incident类的restriction属性说明。

Schema定义

```
<xs:element name="AlternativeID">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="IncidentID" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute ref="restriction"/>
    </xs:complexType>
</xs:element>
```

DTD定义

```
<!ELEMENT AlternativeID（IncidentID+）>
```

6.6 相关活动（RelatedActivity）

类说明

相关活动RelatedActivity类引用在IODEF文档中所描述的与安全事件有关的其他安全事件跟踪号，或者安全事件的惟一标识符。这些引用可能是本地安全事件跟踪号，也可能是其他CSIRTs的安全事件跟踪号。

类图

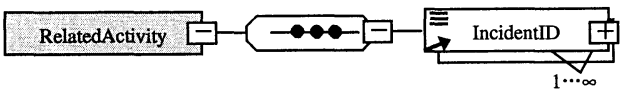


图6 RelatedActivity类

子类

— IncidentID：一个或者多个，表示CSIRT分配安全事件的惟一标识符。

属性

restriction：可选，枚举类型，参见6.3中对这个属性的定义。

Schema定义

```
<xs:element name="RelatedActivity">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="IncidentID" maxOccurs="unbounded"/>
            </xs:sequence>
            <xs:attribute ref="restriction"/>
        </xs:complexType>
    </xs:element>
```

DTD定义

```
<!ELEMENT RelatedActivity（IncidentID+）>
```

6.7 其他数据（AdditionalData）

其他数据（AdditionalData）类作为一个扩展机制，用于描述那些不能在数据模型中描述的信息。对于那些相对简单的信息，提供原子数据类型（整数、字符串等）和一种机制来对它们的含义做注解。通过封装整个符合另外DTD（例如IDMEF）的XML文档，AdditionalData类可以用于扩展数据模型、DTD或Schema以支持专门扩展。在第四节将详细讨论数据模型和DTD的扩展。

AdditionalData不像XML是自描述的。特别是，Additional数据必须能够给出数据的含义。在“meaning”属性中描述了这一信息。由于这些描述超出本规范的范围，需要一些额外的协调来保证使用AdditionalData类的文档接收者，能够弄清楚定制扩展的意思。

类图

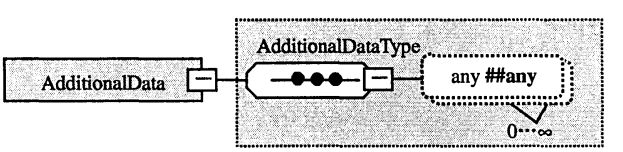


图7 AdditionalData类

属性

restriction：可选，枚举类型，参见6.3对该属性的定义。

type：必需，枚举类型。元素内容的数据类型，这个属性所允许的值如下所示，缺省值为“string”：

- boolean：元素包含一个布尔值，也就是串“true”或者“false”；
- byte：元素内容是一个8比特字节；
- character：元素内容是一个字符；
- date-time：元素内容是一个日期-时间串；
- integer：元素内容是一个整数；
- ntpstamp：元素内容是一个NTP时间戳；
- portlist：元素内容是一个端口列表；
- real：元素内容是一个实数；
- string：元素内容是一个字符串；
- xml：元素内容是XML-标记的（XML-tagged）数据。

meaning：可选，字符串类型。该类中用户自定义的数据的语义的描述。

Schema定义

```
<xs:element name="AdditionalData" type="AdditionalDataType"/>
    <xs:complexType name="AdditionalDataType">
        <xs:sequence>
            <xs:anynamespace="##any"processContents="lax"minOccurs="0"maxOccurs="unbounded" />
        </xs:sequence>
        <xs:attribute ref="dtype" use="required"/>
        <xs:attribute name="meaning" type="xs:string"/>
      </xs:complexType>
    </xs:element>
```

DTD定义

```
<!ELEMENT AdditionalData（*）>
```

6.8 联系（Contact）

类说明

联系（Contact）类描述安全事件有关的组织和个人的联系信息，Contact类封装了对有关方的命名，详细说明了能够通知到他们的联系信息，以及标识了它们在安全事件中的角色。

个人和组织都可以作为联系（Contact），也可以通过使用类的递归定义将个人和组织结合在一起，然后用“type”属性决定了所提供的联系信息的类型。

Contact类的递归定义，即Contact类聚合到Contact类，提供了一种不需要在类中显式地使用标识符来关联信息的方法。当将人以组织来分组，建议将人的实例嵌套到该类的组织的实例中。

类图

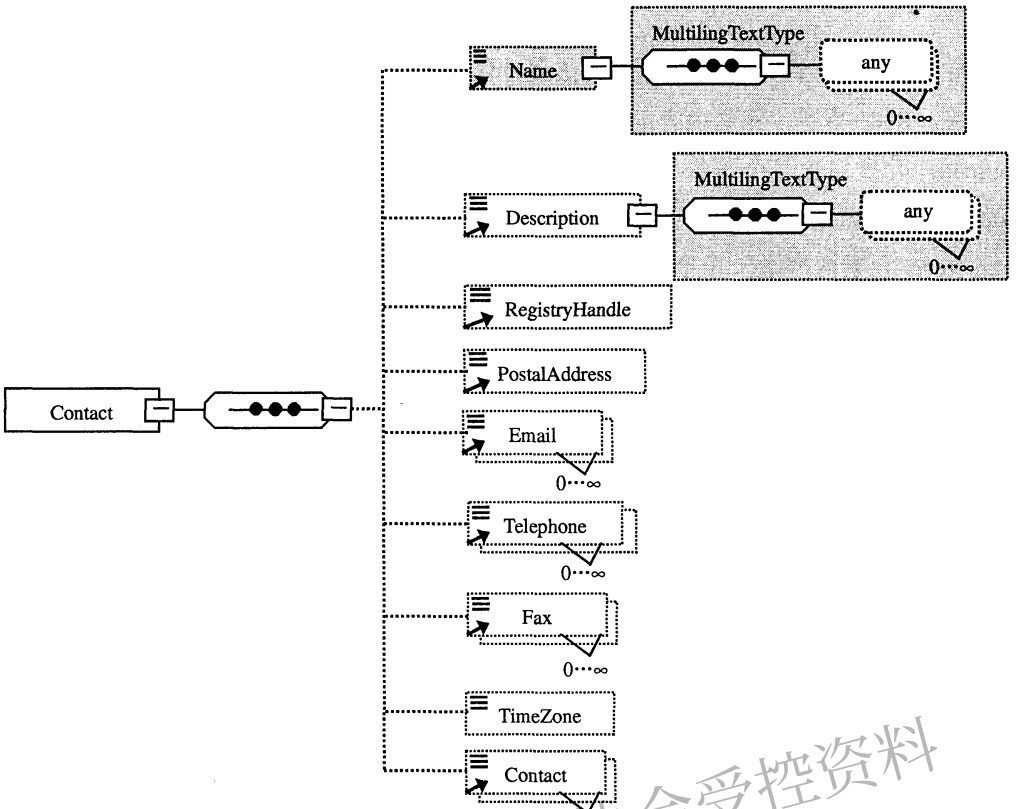


图8 Contact类

子类

— Name：零个或者一个，NAME类型。联系的名称，联系信息可以是一个组织，也可以是某个人，type属性规定联系信息的类型：组织或者个人。

— Description：零个或者一个，STRING类型。联系信息的自由形式的描述。当指个人的时候，通常是这个人的组织头衔。

— RegistryHandle：零个或者多个。在注册处理机构名称（比如运营商及管理员在APNIC注册的Handle），该子类必须对接收方有意义，组织内部的处理机构名称对于组织之外的通信没有什么意义。

— PostalAddress：零个或者一个。联系人或组织的邮政地址。

— Email：零个或者多个。联系人或组织的电子邮件地址。

— Telephone：零个或者多个。联系电话号码。

— Fax：零个或者一个。传真号码。

— Timezone：零个或者一个。联系人或组织所在的时区。

— Contact：零个或者多个。联系信息的递归定义，主要是考虑对数据进行分组。例如有多个联系人的组织。

属性

restriction：可选，枚举类型。参见6.3对该属性的定义。

contactrole：必需，枚举类型。

说明：指出联系信息的角色，这个属性被定义为一个枚举列表：

— creator：生成IODEF文档的实体；

— admin：主机或者网络的管理员；

— tech：主机或者网络的技术联系；

— irt：参与事件处理的CSIRT；

— cc：保持告知安全事件处理的实体。

type：必需，枚举类型。

说明：说明联系信息的类型，这一属性被定义为一个枚举列表：

— person：个人；

— organization：组织。

Schema定义

```
<xs:element name="Contact">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="Name" minOccurs="0"/>
      <xs:element ref="Description" minOccurs="0"/>
      <xs:element ref="RegistryHandle" minOccurs="0"/>
      <xs:element ref="PostalAddress" minOccurs="0"/>
      <xs:element ref="Email" minOccurs="0" maxOccurs= "unbounded"/>
      <xs:element ref="Telephone" minOccurs="0" maxOccurs= "unbounded" />
      <xs:element ref="Fax" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element name="TimeZone" type="xs:string" minOccurs="0"/>
      <xs:element ref="Contact" minOccurs="0" maxOccurs= "unbounded"/>
      </xs:sequence>
      <xs:attribute ref="contactrole" use="required"/>
      <xs:attribute ref="contacttype" use="required"/>
      <xs:attribute ref="restriction"/>
    </xs:complexType>
  </xs:element>
```

DTD定义

<!ELEMENT Contact（Name?，Description?，RegistryHandle?，PostalAddress?，Email*，Telephone*，Fax*，TimeZone?，Contact*）>

6.9 注册机构标识（RegistryHandle）

类说明

该类表示一个指向Internet注册机构或者特定团体的数据信息。具体信息由在元素内容中指定的名称和在registrytype属性中指定的所属数据库组成。

类图

图9 RegistryHandle类

子类

无。

属性

registrytype：必需，枚举类型。

说明：是指安全事件处理机构或个人所属的数据库。缺省值为“local”，可选值有：

- internic：互联网信息中心；
- apnic：亚太网络信息中心；
- arin：美国互联网号码登记处；
- lacnic：拉丁美洲和加勒比海地区IP地址登记处；
- ripe：法语“Réseaux IP Européens”即欧洲IP网络“European IP Networks”；
- ti：TERNEA可信介绍人；
- local：CSIRT本地数据库。

Schema定义

```
<xs:element name="RegistryHandle">
    <xs:complexType mixed="true">
        <xs:attribute ref="registrytype" use="optional" default="local"/>
    </xs:complexType>
</xs:element>
```

DTD定义

```
<!ELEMENT RegistryHandle（#PCDATA）>
```

6.10 时间（Time）

类说明

本数据模型使用不同的类来表示时间戳，它们的定义是相同的，但是为了表达语义上的差别，每一个命名不同。每个类的元素内容是依照DATETIME数据类型格式的时间戳。

- StartTime类表示活动开始的时间戳。
- EndTime类表示活动结束的时间戳。
- DetectTime类表示某活动第一次被检测出的时间戳。
- ReportTime类表示报告检测出的活动的时间戳。
- DateTime是时间戳的通用表示。

类图和子类：上述每个类元素都是DATETIME数据类型，无子类，因此省略类图。

属性

ntpstamp：可选，NTPTIMESTAMP类型。

NTP时间戳表示元素内容里的时间戳。由于这个属性是冗余的，ntpstamp属性的使用是可选的。不建议在元素内容和属性中都包含时间戳；如果元素内容和属性都使用了时间戳，它们的值必须相同。

6.11 期望（Expectation）

类说明

期望（Expectation）类表示文档的发送者期望接受者所采取的行动，比如阻止攻击行为、通知用户等等。

类图

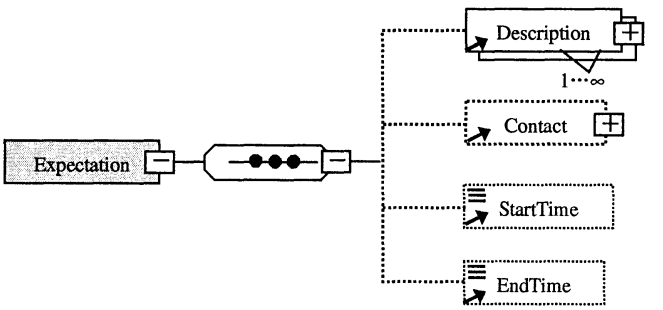


图10 Expection类

子类

— Description：一个或者多个，STRING类型。所期望的动作的自由形式的描述。

— StartTime：零个或者一个。开始处理事件动作的时间，如果这个时间比Incident类中指明的ReportTime还早的话，表示应当尽早完成期望所采取的行动。如果不存在这个元素，就表示由接收者决定何时执行。

— EndTime：零个或者一个。动作应当完成的时间。如果动作在此时还没有完成，动作将不再执行。

— Contact：零个或者一个。动作预期的参与方。

属性

restriction：可选，枚举类型。参见6.3中对该属性的定义。

priority：可选，枚举类型。

说明：指出动作的预设优先级，本属性是一个没有缺省值的枚举列表：

— low：低优先级；

— medium：中优先级；

— high：高优先级。

expect：可选，枚举类型。

说明：对所请求的动作类型分类，本属性是一个没有缺省值的枚举列表：

— nothing：不需要任何动作，对信息不采取任何动作；

— contact-site：联系在接收者的顾客名单上的站点；

— contact-me: 和文档的发起人联系；

— block：阻塞或者调查在文档接收者的顾客名单上的机器；

— investigate：调查与文档相关的安全事件；

— other：其他情况。

Schema定义

```
<xs:element name="Expectation">
```

```
            <xs:complexType>
                <xs:sequence>
                    <xs:element ref="Description" maxOccurs="unbounded"/>
                    <xs:element ref="Contact" minOccurs="0"/>
                    <xs:element ref="StartTime" minOccurs="0"/>
                    <xs:element ref="EndTime" minOccurs="0"/>
                </xs:sequence>
                <xs:attribute ref="restriction" default="default"/>
                <xs:attribute ref="priority"/>
                <xs:attribute ref="expect"/>
            </xs:complexType>
    </xs:element>
```

DTD定义

```
<!ELEMENT Expectation（Description+，Contact?，StartTime?，EndTime?）>
```

6.12 攻击方法（Method）

类说明

Method类提供攻击者所使用的方法。Method类可以引用著名的通用漏洞披露（CVE），列举出攻击者在攻击中所使用的工具，并提供有关入侵活动的自然语言形式的描述。

类图

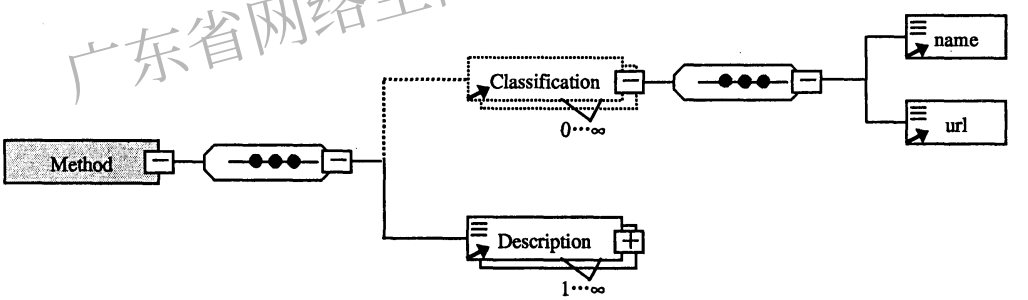


图11 Method类

子类

— Classification：零个或者多个。一般用攻击所用漏洞数据库中的漏洞名称或编号。

— Description：零个或者多个，STRING类型。在安全事件中所使用的攻击方法的自然语言描述。

— name：一个，STRING类型。是Classification类的元素，表示引用的数据库的名称，在origin属性中指定引用数据库的名称。

— url：一个，URI类型。是Classification类的元素，表示指向由name引用的漏洞其他相关信息的URL。

属性

Classification属性

origin：是必需，枚举类型。引用的数据库的名称，允许值如下所示：

— bugtraqid：Bugtraq；

— cve：通用漏洞披露；

— certcc：CERT/CC协调中心漏洞目录；

— vendor：制造商，其名字应当在name类中指定；

— local：本地数据库；

— other：其他。

Schema定义

```
<xs:element name="Method">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="Classification" minOccurs="0" maxOccurs= "unbounded" />
            <xs:element ref="Description" maxOccurs="unbounded"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="Classification">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="name"/>
            <xs:element ref="url"/>
        </xs:sequence>
        <xs:attribute ref="origin" default="other"/>
    </xs:complexType>
    </xs:element>
```

DTD定义

```
<!ELEMENT Method（Classification*，Description+）>
<!ELEMENT Classification（name，url）>
```

6.13 评估（Assessment）类

类说明

评估（Assessment）类描述安全事件活动的技术与非技术方面的影响。

类图

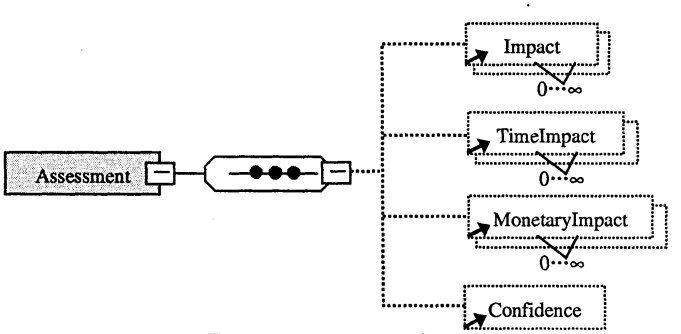


图12 Assessment类

子类

— Impact：零个或者多个。安全事件活动对计算机和网络的技术影响的子类；其元素内容可以为空，或者包含技术影响的自由形式的描述。

— TimeImpact：零个或者多个。安全事件活动用时间度量的影响的子类；其元素内容是一个具体说明影响的数值（实数REAL），它是关于时间的函数，本属性描述明确的单位和度量标准。

— MonetaryImpact：零个或者多个。安全事件活动用货币度量的影响的子类；元素内容是一个具体说明影响的数值（实数REAL），它是关于金钱的函数，这个属性描述明确的货币和货币度量标准。

— Confidence：零个或者一个。在评估中的信心估计的子类；这个元素应当仅在CSIRT能够产生有意义信息的时候才使用。如果必须给出粗略的评估时，应当使用“low”，“medium”或“high”作为等级值。

注意：以上4个子类都是用于安全时间的影响评估，通过各自的属性进行描述，具体见属性部分。

属性

Assessment属性：

restriction：可选，枚举类型。参见6.3中对该属性的定义。

Impact类属性：

severity：可选，枚举类型。对活动的相对严重性的估计，可供选取的值如下所示，该属性没有缺省值：

— low：低严重性；

— medium：中等程度严重性；

— high：高严重性。

completion：可选，枚举类型。IODEF文档的创建者是否相信活动成功的一个信号，可选值如下所示，该属性没有缺省值：

— failed：攻击企图没有成功；

— succeeded：攻击企图成功了。

impacttype：必需，枚举类型。可以给出一个大致的影响类型，可供选取的值如下所示，缺省值为“unknown”：

— admin：企图得到或者已经得到的管理特权；

— dos：企图或者成功完成拒绝服务攻击；

— file：企图或者成功地对文件进行未授权操作；

— recon：企图或者成功进行网络探测；

— user：企图或者成功得到的用户权限；

— none：活动没有任何（技术）影响；

— unknown：影响未知；

— other：不属于以上范畴的任何情况。

TimeImpact类属性：

severity：可选，枚举类型。对事件影响的严重性估计，可供选取的值如下所示，该属性没有缺省值：

— low：低严重性；

— medium：中等程度严重性；

— high：高严重性。

metric：必需，枚举类型。描述事件影响的尺度，可供选择的值如下，该属性没有缺省值：

— labor：恢复活动的总共的人员时间（例如，2个雇员每人工作4小时，就是8小时）；

— elapsed：从开始恢复到完成总共经历的时间；

— downtime：某些提供的服务中断（不能得到）持续的时间。

units：必需，枚举类型。定义时间度量单位。可供选择的值如下，缺省值为“hours”：

— seconds：秒；

— minutes：分；

— hours：小时；

— days：天。

MonetaryImpact类属性：

severity：可选，枚举类型。对事件影响的严重性估计，可供选取的值如下所示，该属性没有缺省值：

— low：低严重性；

— medium：中等程度严重性；

— high：高严重性。

currency：必需，枚举类型。事件造成的经济损失，在ISO 4217:2001中定义了可供选取的许可值，该属性没有缺省值。

Confidence类属性：

rating：必需，枚举类型。 指示CSIRT对安全时间的评估信心，可选值如下，缺省值为“numeric”：

— low：低；

— medium：中；

— high：高；

— numeric：CSIRT提供的表明其对评估的信心的概率值；

— unknown：未知。

注意：如果rating属性没有被设置为“numeric”，则元素内容可以为空；否则，必须提供一个信心值。

Schema定义

```
<xs:element name="Assessment">
  <xs:complexType>
      <xs:sequence>
          <xs:element ref="Impact" minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="TimeImpact" minOccurs="0" maxOccurs= "unbounded" />
          <xs:element ref="MonetaryImpact" minOccurs="0" maxOccurs= "unbounded" />
          <xs:element ref="Confidence" minOccurs="0"/>
      </xs:sequence>
      <xs:attribute ref="restriction"/>
  </xs:complexType>
</xs:element>
```

DTD定义

```
<!ELEMENT Assessment（Impact*，TimeImpact*，MonetaryImpact*，Confidence?）>
```

6.14 历史（History）

类说明

历史（History）类是发生的重要事件，或者事件参与方（例如，最初报告人，调查中的CSIRT，或有关的系统管理员）在处理安全事件期间所采取行动的日记或日志。在日志中维护的细节的程度交由那些处理安全事件的参与方自行决定。

HistoryItem类是History日志里的一个条目，在History日志中记录了在处理当前安全事件期间所发生的事件，或者特别重要的动作。在日志中条目的细节用自由语言描述，但是也可以分类。

类图

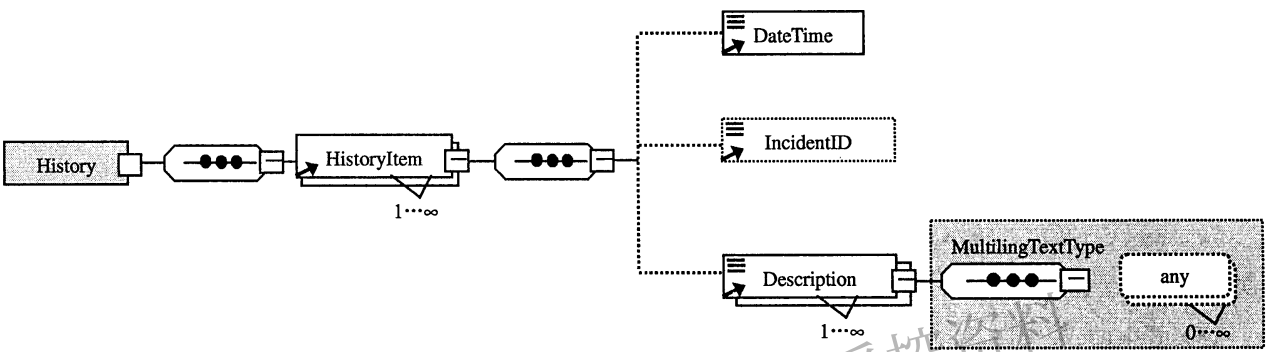


图13 History类

子类

— HistoryItem：一个或者多个。在重要的事件或者有关方所采取的行动的历史记录条目。

— IncidentID：零个或者一个。在由多个参与方产生的历史日志中，IncidentID提供一种方法，指明哪个CSIRT产生的特定条目，以及引用该组织对该活动的本地安全事件跟踪号。当单个组织维护历史日志时，可以忽略这个类。

— DateTime：一个。条目在历史日志中的时间戳（例如，在Description中所描述的动作发生的时间）。

— Description：一个或者多个，STRING类型。将在历史日志中记录的动作或者事件的自由形式的文本描述。

属性

History类属性：

restriction：可选，枚举类型，参见6.3中对该属性的定义。

HistoryItem类属性：

restriction：可选，枚举类型，参见6.3中对该属性的定义；

historycat：可选，枚举类型。对在历史日志条目中纪录的活动或事件的类型分类，条目的细节是在Description类中记录的自由形式的描述，可能的值是一个枚举列表，缺省值为“other”：

— triaged：安全事件数据由IHS接收和处理；

— notification：在安全事件中，被发送给有关方的通知，例如，一个CSIRT发送一个消息给正受攻击的站点管理员；

— shared-info：与未直接卷入安全事件的人员共享的与事件有关的信息；

— received-info：有关接收到的安全事件的额外信息；

— remediation：安全事件已经解决，可以包含一个简短的描述；

— other：其他。

Schema定义

```
<xs:element name="History">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="HistoryItem" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute ref="restriction" default="default"/>
    </xs:complexType>
</xs:element>
<xs:element name="HistoryItem">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="DateTime"/>
            <xs:element ref="IncidentID" minOccurs="0"/>
            <xs:element ref="Description" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute ref="restriction"/>
        <xs:attribute name="historycat"/>
    </xs:complexType>
</xs:element>
```

DTD定义

```
<!ELEMENT History（HistoryItem+）>
<!ELEMENT HistoryItem（DateTime，IncidentID?，Description+）>
```

6.15 活动数据（EventData）

类说明

活动数据（EventData）类描述发生在某个特定主机集合或者网络安全事件涉及的活动。这一描述包括那些引起活动的系统以及作为目标的系统，攻击者所使用技术的评估，活动对组织的影响，执行的安全事件处理任务的列表，以及任何发现的取证证据。

在Incident和EventData的聚合类中，存在有重复出现的类。然而，这些类的语义大不相同。Incident的聚合类提供整个安全事件的概要信息，而EventData的聚合类则提供有关安全事件子集的信息。举例来说，注意到Assessment类被聚合在这两个类中。考虑这样的情况，将数值x赋给Incident:Assessment:MonetaryImpact，并考虑赋值y（其中$y<x$），聚合在EventData类中给定的MonetaryImpact，这两个值的语义都是金融损失。在Incident类中出现的损失是安全事件范围的，而在EventData类中出现的损失是整个损失的一个子集，这就允许人们描述构成安全事件的事件的某个子集一

个特定的损失。通过这种方法可以有效地提供先前在Incident类中指明的整个损失的一个细目（或者更加明确的描述）。

EventData类的递归定义，即EventData类被集合到EventData类中，给相关联的信息提供了一种不需要在类中显式地使用惟一的属性标识符的方法。

EventData类的子类（及其所有的兄弟）逻辑上“继承”EventData父类的聚合类。然而，EventData兄弟类的存在（在EventData类中仅有一个EventData子类，兄弟类决不会没有意义）意味着事件存在一些不相交的性质。EventData父类的子类描绘这些区别，同时依然保留一种描述共同性质的方法（也就是说，父-子关系）。例如，一个EventData类可能被用来描述卷入到安全事件中的两台机器。可以使用System类的多个实例来描述这一情况。这两台机器的技术联系（也就是说，contact类）碰巧是相同的，而事件的影响（也就是说，Assessment类）却是不相同。

类图

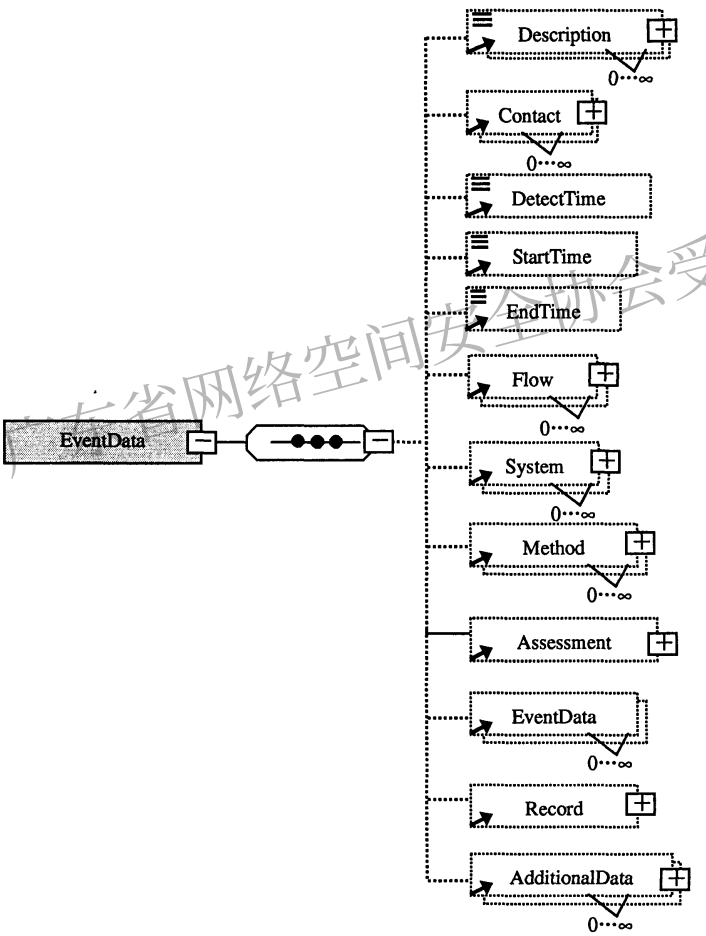


图14　EventData类

子类

- Description：零个或者多个，字符串类型。安全事件活动的自由形式的文本描述。
- Contact：一个或多个。安全事件有关的参与方的联系信息。
- DetectTime：零个或者一个。安全事件活动最初被检测出的时间。
- StartTime：零个或者一个。安全事件活动开始的时间。
- EndTime：零个或者一个。安全事件活动结束的时间。

— Method：零个或者多个。入侵者所使用的技术和方法（譬如工具，漏洞）。

— Assessment：一个或者多个。安全事件活动影响的描述。

— Expectation：零个或多个。文档接收者将执行的预期动作。

— EventData：零个或者多个。导致安全事件的事件数据的详细信息。

— Record：零个或者一个。提供事件有关信息的支撑数据（例如，日志文件）。

— AdditionalData：零个或者多个。使用不能在别的地方描述的信息来扩展数据模型的区域。

属性

Restriction：可选，枚举类型。

说明：这个属性指出IODEF-Document的发送者期望接收者应该遵守的保密原则，当然文档的接收者自由决定是否遵守这个原则。逻辑上，子类可以继承父类的这个属性值。由于多数高层类都有restriction属性，这就有可能设置细粒度的保密策略。如果子类收紧或者放松保密规则，子类可以不考虑父类的保密规则。对一个没有指定restriction属性值的类，可以在其指定了restriction属性值的最邻近的祖先类中得出该类的restriction属性值。restriction属性被定义为一个枚举类型值，缺省值为“private”。

— public：对信息没有任何级别的限制；

— need-to-know：信息可以被和安全事件有关的其他方共享（举例来说，多个受害站点能够相互通告）；

— private：信息不能被共享；

— default：按照通信各方预先安排的信息披露规则，决定是否可共享信息。

Schema定义

```
<xs:element name="EventData">
 <xs:complexType>
     <xs:sequence>
         <xs:element ref="Description" minOccurs="0" maxOccurs= "unbounded"/>
         <xs:element ref="Contact" minOccurs="0" maxOccurs= "unbounded"/>
         <xs:element ref="DetectTime" minOccurs="0"/>
         <xs:element ref="StartTime" minOccurs="0"/>
         <xs:element ref="EndTime" minOccurs="0"/>
         <xs:element ref="Flow" minOccurs="0" maxOccurs="unbounded"/>
         <xs:element ref="System" minOccurs="0" maxOccurs="unbounded"/>
         <xs:element ref="Method" minOccurs="0" maxOccurs= "unbounded"/>
         <xs:element ref="Assessment" minOccurs="0"/>
         <xs:element ref="EventData" minOccurs="0" maxOccurs= "unbounded"/>
         <xs:element ref="Record" minOccurs="0"/>
         <xs:element ref="AdditionalData" minOccurs="0" maxOccurs= "unbounded" />
     </xs:sequence>
     <xs:attribute ref="restriction" default="default"/>
 </xs:complexType>
```

</xs:element>

DTD定义

<!ELEMENT EventData（Description*，Contact*，DetectTime?，StartTime?，EndTime?，Flow*，System*，Method*，Assessment?，EventData*，Record?，AdditionalData*）>

6.16 流（Flow）和系统（System）

类说明

流（Flow）类描述一组安全事件，它们可能来源于相同网络的不同系统或应用程序。

系统（System）类描述被卷入安全事件中给定的计算机，或者网络技术方面的信息。由这个类描述的系统，经由systemcat属性按照它们在安全事件中充当的角色加以分类。

Node、Service类的含义，与System类中的systemcat属性值有关。如果在System类的systemcat属性是“source”，则所描述的聚合类表示引发活动的机器，用户，进程或者服务。如果systemcat属性是“target”或者“intermediary”，则所描述的机器、用户、进程或者服务就是在活动中，目标或者中介的机器、用户、进程或服务。

类图

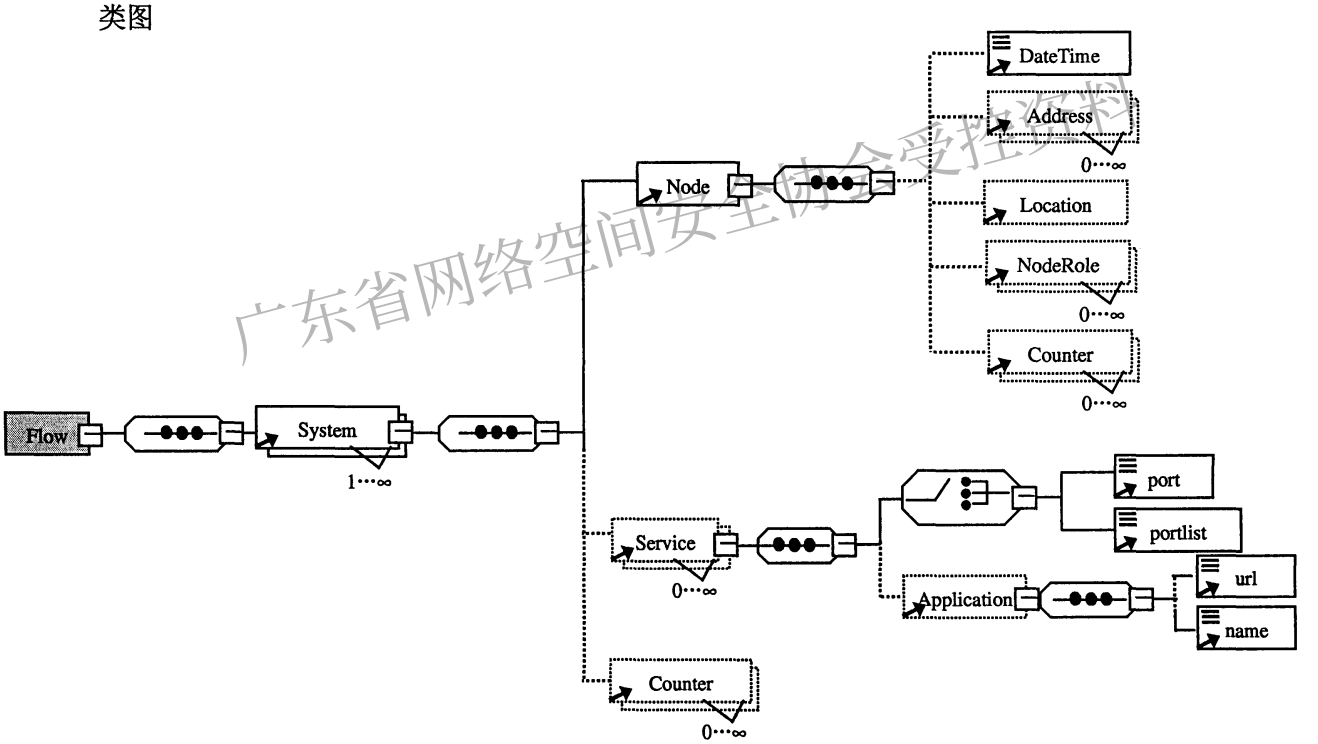


图15 Flow类和System类

子类

System类：产生安全事件的源、目的或中间系统（节点或网络）。

— Node：一个。描述与安全事件活动有关的主机或者网络的聚合类。

— DateTime：零个或者一个。执行名称和地址之间解析的时间戳，如果同时给出了Address和Name，就应当提供这一信息。

— Address：零个或者多个。是Node的子类，表示设备的网络地址或者硬件地址。除非提供了名称，至少要指定一个地址。

— Location：零个或者一个，STRING。是Node的子类，表示设备的物理位置。

— Name：零个或者一个，STRING。设备的完整域名（FQDN）名称。如果没有给出Address信息，必须提供Name信息。

— NodeRole：零个或者多个，设备的预期目的，具有一个属性。

— Service：零个或者一个，在Node中指定的主机上的目标网络服务。

— Counter：零个或多个。

属性

System类属性：

restriction：可选，枚举类型，参见6.3中对该属性的定义。

systecat：必需，枚举类型。对System类中指明的系统在安全事件活动中的角色进行分类，可能的值是：

— source：System是攻击的源。

— target：System是受攻击的目标。

— intermediate：System是在攻击中被利用的中介机器。

Interface：可选，STRING。指明在原始系统上的事件的接口。

Spoofed：可选，枚举类型。关于System是否是真正的目标或者只是虚晃的攻击目标，可供选择的值如下所示，缺省值为“unknown”：

— unknown：category信息的正确性未知。

— yes：将主机或者网络归类为源或者目标的category值，很可能不正确。被归类为源的System 很可能是一个圈套，被归类为目标的System很可能不是预想的受害系统。

— no：将主机或网络归类为源或者目标的category值被认为是正确。

Schema定义

```
<xs:element name="Flow">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="System" maxOccurs="unbounded"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="System">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="Node"/>
            <xs:element ref="Service" minOccurs="0" maxOccurs="unbounded"/>
            <xs:element ref="Counter" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute ref="restriction"/>
        <xs:attribute name="interface" type="xs:string"/>
```

```
            <xs:attribute ref="systemcat"/>
            <xs:attribute ref="spoofed" default="unknown"/>
        </xs:complexType>
    </xs:element>
```

DTD定义

```
<!ELEMENT System（Node，Service*，Counter*）>
```

6.17 节点（Node）

类说明

Node类用来惟一标识主机或者网络设备（例如路由器，交换机）。

NodeRole类是Node类的子类，描述某特定主机执行的功能（基于一个预先定义的功能列表）。

类图

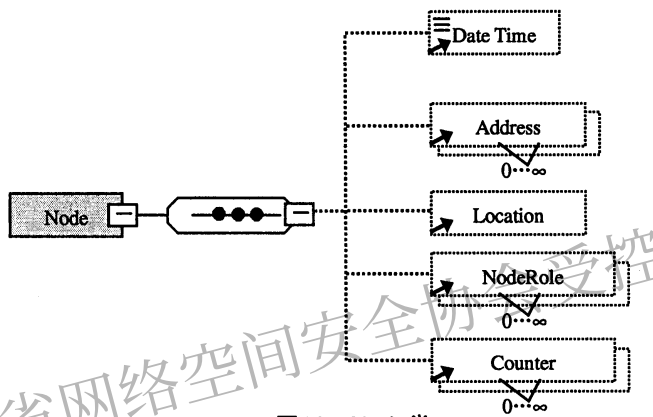


图16 Node类

子类

— DateTime：零个或者一个。执行名称和地址之间解析的时间戳，如果同时给出了Address和Name，就应当提供这一信息。

— Address：零个或者多个。是Node的子类，表示设备的网络地址或者硬件地址。除非提供了名称，至少要指定一个地址。

— Location：零个或者一个，STRING。是Node的子类，表示主机或设备的物理位置。

— NodeRole：零个或者多个。该节点的角色，具有一个属性。

— Counter：一个或多个，用于概括本主机或网络被事件影响的次数。

属性

Node属性：

Nodecat：可选，枚举类型。可能取值如下所示，缺省值为“unknown”：

— unknown：域未知或者不相关。

— ads：Windows 2000高级目录服务。

— afs：Andrew文件系统。

— coda：Coda分布式文件系统。

— dfs：分布式文件系统（IBM）。

— dns：域名系统。

— hosts：本地主机文件。
— kerberos：Kerberos域。
— nds：Novell目录服务。
— nis：网络信息服务（Sun）。
— nisplus：网络信息服务+（Sun）。
— nt：Windows NT域。
— wfw：Windows工作组。

NodeRole属性：

noderolecat：必需。由节点提供的功能，如果指定值为“other”，应当在元素的内容里提供描述，缺省值为“other”：

— client：客户计算机。
— server-internal：带有内部服务的服务器。
— server-public：带有公开服务的服务器。
— www：WWW服务器。
— mail：邮件服务器。
— messaging：消息服务器（例如NNTP、IRC、IM）。
— streaming：流媒体服务器。
— voice：Voice服务器（例如SIP、H.323）。
— file：文件服务器（例如SMB、CVS、AFS）。
— ftp：FTP服务器。
— p2p：Peer-to-peer节点。
— name：Name服务器（例如DNS、WINS）。
— directory：目录服务器（例如LDAP、finger、whois）。
— credential：机要（Credential）服务器（例如domain控制器、Kerberos）。
— print：打印服务器。
— application：应用服务器。
— database：数据库服务器。
— infra：基础设施服务器（例如路由器、防火墙、DHCP）。
— log：日志服务器。
— other：没有在本列表中出现的其他角色。

Schema定义

```
<xs:element name="Node">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="DateTime" minOccurs="0"/>
            <xs:element ref="Address" minOccurs="0" maxOccurs="unbounded"/>
            <xs:element ref="Location" minOccurs="0"/>
```

```
                <xs:element ref="NodeRole" minOccurs="0" maxOccurs="unbounded"/>
                <xs:element ref="Counter" minOccurs="0" maxOccurs="unbounded"/>
            </xs:sequence>
            <xs:attribute ref="nodecat" default="unknown"/>
        </xs:complexType>
    </xs:element>
```

DTD定义

```
<!ELEMENT Node（DateTime?，Address*，Location?，NodeRole*，Counter*）>
```

6.18 服务（Service）

类说明

Service类描述主机或网络所提供的服务，例如WWW服务、FTP服务、Email服务等。服务通过端口或端口列表，以及监听这些端口的服务程序来确定。

类图

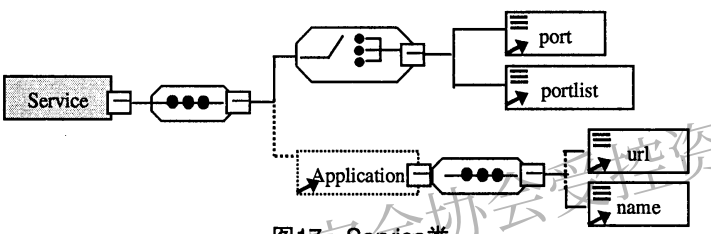


图17 Service类

子类

— Application：零个或一个，描述端口或端口表上绑定的应用程序；

— Port或Portlist：一个，描述网络服务程序所使用的端口或端口表。

属性

无。

Schema定义

```
<xs:element name="Service">
    <xs:complexType>
        <xs:sequence>
            <xs:choice>
                <xs:element ref="port"/>
                <xs:element ref="portlist"/>
            </xs:choice>
            <xs:element ref="Application" minOccurs="0"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
```

DTD定义

```
<!ELEMENT Service（(port | portlist)，Application?）>
```

6.19 记录（Record）和记录项（RecordItem）

类说明

Record类对提供安全事件活动纪录的日志和审计数据分组。典型地，数据源是监控工具的输出（例如，由IDS产生的IDMEF消息，Web服务器的连接日志），这些监控工具被用来揭露恶意的活动。这些日志应当提供向CSIRT报告的人为什么相信安全事件已经发生的有关证据。

RecordData类对由给定的传感器获得的日志或者审计数据（例如IDS，防火墙日志）进行分组，并提供一种方法对输出做注释。

RecordItem类提供一个将有关日志，审计追踪或者取证数据合并在一起的方法，来支持在事件分析期间所得出的结论。可以直接将该数据封装为文档的一部分，或者被引用，借此使用该类仅仅作为一个指向有关信息的指针。

类图

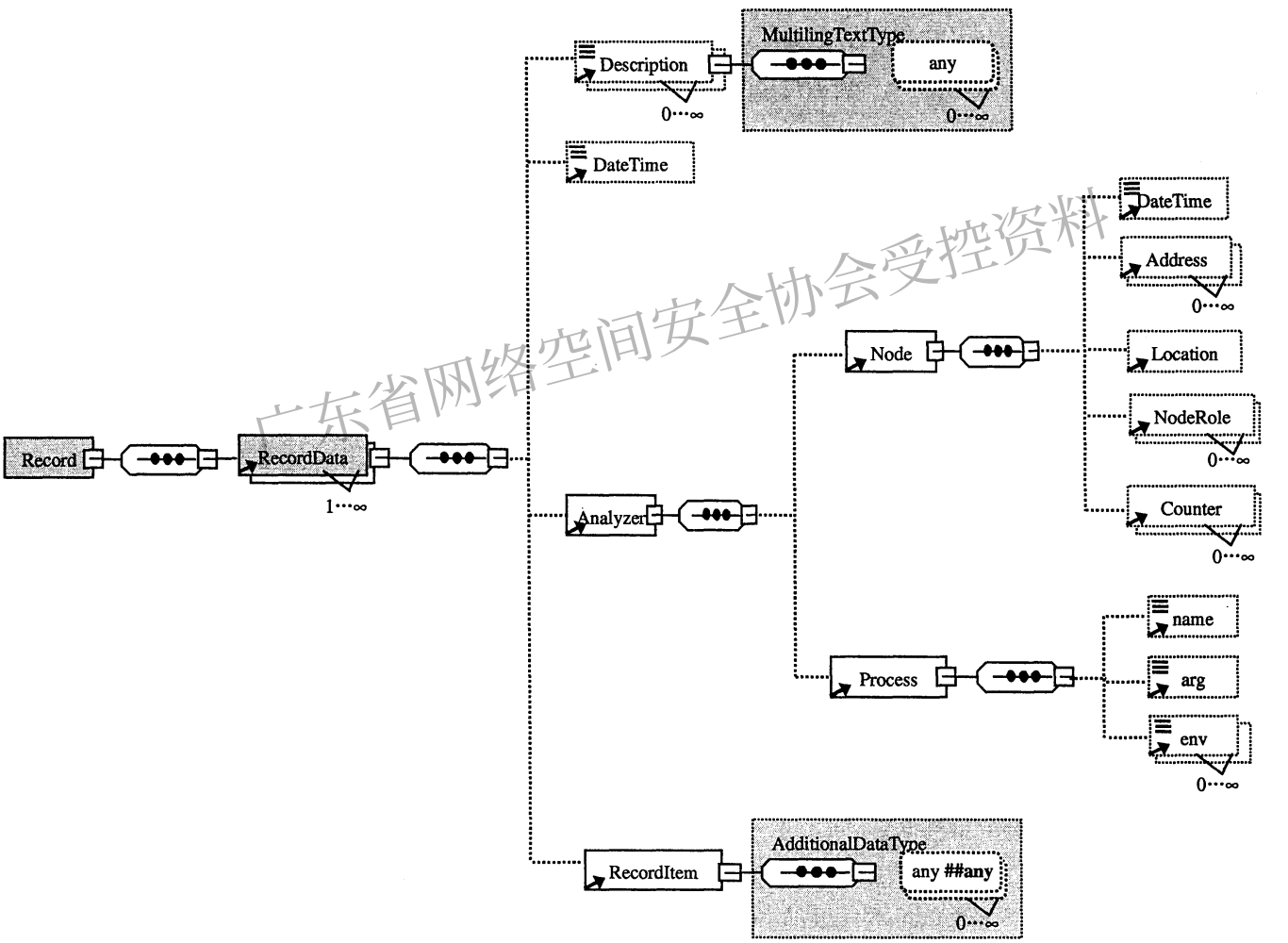


图18 Record类和RecordItem类

子类

— RecordData：一个或者多个。由一个特定类型的传感器产生的日志或者审计数据。由给定的传感器获得的日志或者审计数据（例如IDS，防火墙日志）进行分组，并提供一种方法对输出做注释。

— Description：零个或者多个，STRING类型。所提供的RecordItem 数据的自由形式的文本描述，这个描述至少应当传达所提供的RecordItem 数据的重要性。

— DateTime：零个或者一个。RecordItem数据的时间戳信息。

— Analyzer：零个或者多个。用来产生RecordItem数据的传感器的有关信息；标识用来生成特殊日志或审计数据的传感器（例如入侵检测系统IDS，防火墙，Web服务器）。

— RecordItem：一个或者多个。日志，审计，或者取证数据。提供一个将有关日志，审计追踪或者取证数据合并在一起的方法，来支持在事件分析期间所得出的结论。可以直接将该数据封装为文档的一部分，或者被引用，借此使用该类仅仅作为一个指向有关信息的指针。

属性

Record类属性：

restriction：可选，枚举类型，参见6.3中对该属性的定义。

RecordData类属性：

restriction：可选，枚举类型，参见6.3中对该属性的定义。

RecordItem类属性：

dtype：必需，规定在这个类中出现的日志数据的类型，本质上，RecordItem类是一个能够支持安全事件数据的专门表示法的扩展类，在XML中，并不是所有的属性都是必需的。包含在元素内容中的数据的类型，这个属性的许可值如下，缺省值为“string”：

— boolean：元素包含一布尔值，也就是说，串“true”或者“false”。

— byte：元素内容是一个8比特字节。

— character：元素内容是一个字符。

— date-time：元素内容是一个日期-时间串。

— integer：元素内容是一个整数。

— ntpstamp：元素内容是一个NTP时间戳。

— portlist：元素内容是一个端口列表。

— real：元素内容是一个实数。

— string：元素内容是一个字符串。

— file：元素内容是一个base64编码的二进制文件。

— path：元素内容是一个文件系统路径。

— url：元素内容是一个URL。

— xml：元素内容是带XML-标记的数据。

Meaning：可选，RecordItem的元素的具体含义。

Schema定义

```
<xs:element name="Record">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="RecordData" maxOccurs="unbounded"/>
        </xs:sequence>
```

```
            <xs:attribute ref="restriction"/>
        </xs:complexType>
    </xs:element>
    <xs:element name="RecordData">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="Description" minOccurs="0" maxOccurs="unbounded"/>
                <xs:element ref="DateTime" minOccurs="0"/>
                <xs:element ref="Analyzer" minOccurs="0"/>
                <xs:element ref="RecordItem" minOccurs="0"/>
            </xs:sequence>
            <xs:attribute ref="restriction"/>
        </xs:complexType>
    </xs:element>
```

DTD定义

```
<!ELEMENT Record （RecordData+）>
<!ELEMENT RecordData（Description*，DateTime?，Analyzer?，RecordItem?）>
```

6.20 分析器（Analyzer）

类说明

Analyzer类标识用来生成特殊日志或审计数据的分析器（例如入侵检测系统IDS、防火墙、Web服务器）。

类图

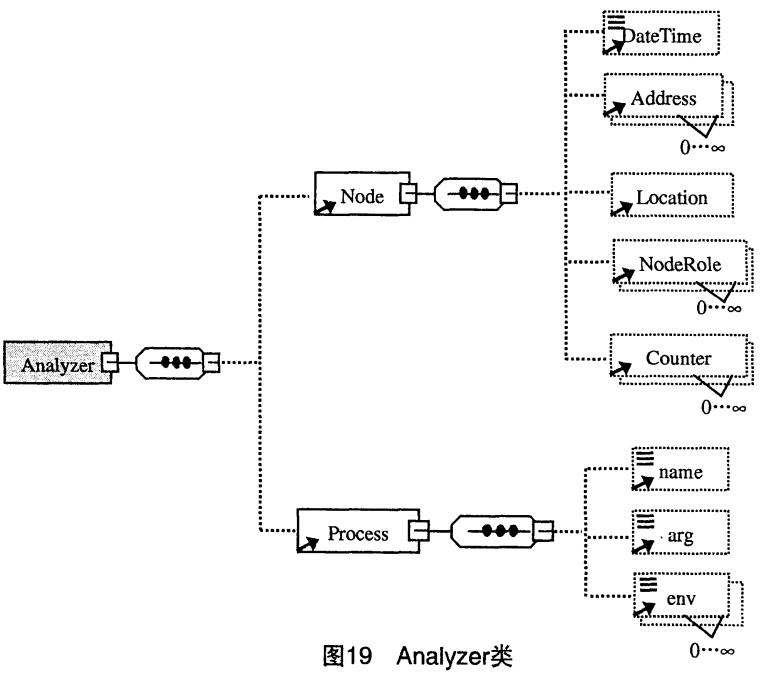


图19 Analyzer类

子类

— Node：零个或一个。描述与安全事件活动有关的主机或者网络的聚合类。

— Process：零个或一个。描述与分析器相关的进程类。

属性

analyzerid：可选，STRING。

manufacturer：可选，STRING，表示分析器的制造商。

model：可选，STRING，表示分析器的型号。

version：可选，STRING，表示分析器的版本。

class：可选，STRING，表示分析器的类型。

ostype：可选，STRING，表示运行分析主机的操作系统类型，如Windows、Linux、Unix和Mac OX等。

osversion：可选，STRING，表示运行分析器主机的操作系统版本。

Schema定义

```
<xs:element name="Analyzer">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="Node" minOccurs="0"/>
            <!-- Node presence is agreed for IODEF-04 -->
            <xs:element ref="Process" minOccurs="0"/>
        </xs:sequence>
        <xs:attribute name="analyzerid" type="xs:string" default="0"/>
        <xs:attribute name="manufacturer" type="xs:string"/>
        <xs:attribute name="model" type="xs:string"/>
        <xs:attribute name="version" type="xs:string"/>
        <xs:attribute name="class" type="xs:string"/>
        <xs:attribute name="ostype" type="xs:string"/>
        <xs:attribute name="osversion" type="xs:string"/>
    </xs:complexType>
</xs:element>
```

DTD定义

```
<!ELEMENT Analyzer （Node?，Process?）>
```

7 安全事件描述交换格式的扩展和实现指南

安全事件描述交换格式应该是可扩展的，主要是基于两方面的原因。一方面，随着互联网及其相关技术不断发展，今后会出现全新的、各种类型的安全事件，仅仅使用基础数据模型是不能完全地、充分地描述它们独特的特征。另一方面，各CSIRTs行为也是不断变化的，数据模型需要进行更新，能够及时反映这些变化，同时及时更新数据模型的具体实现，可以快速适应CSIRTs对安全事件处理流程的变化。因此，数据模型应能够合理、有效和方便地进行扩展。需要对IODEF数据模型进行扩展，以便于IODEF可以精确、全面地描述新出现的安全事件，以此增加新的特征。

本章首先描述数据模型的扩展机制，然后以DTD为具体实例说明IODEF的扩展实现。

7.1 扩展机制

基础数据模型可以通过继承和聚合这两种机制进行扩展。

（1）继承：通过父类派生出新的子类，在子类中定义父类中没有的额外属性或者操作。

（2）聚合：通过定义新的子类，并与父类相结合，生成全新的、自包含的类。

对于上述两种扩充机制，继承应作为首选方案，其优点是它保留了已有的数据模型，以及在该模型的类上所执行的操作。

7.2 扩展原则

7.2.1 基本原则

对IODEF数据模型进行扩展，应该遵循下面两个基本原则。

（1）对于任意“原子”数据（例如，整数，字符串等），应采用直接而简单的方法，就是将数据包含在AdditionalData类和RecordItem类中。

（2）对于任意复杂的数据类型和事件类，应采用创建数据类型或事件类的外部说明文件，如DTD或Schema，并通过数据模型引用该说明文件，实现数据模型的扩展。这些数据类型和事件类的实例应作为AdditionalData类和RecordItem类的子类。

7.2.2 实现原则

通过使用外部DTD或Schema扩展数据模型，必须遵循如下原则。

（1）在IODEF基础数据模型定义文件中必须定义包含扩展的DTD或Schema的位置信息。

（2）扩展的DTD或Schema必须在一个单独名称空间中申明其所有元素和属性，不应该申明包含在“IODEF”或者缺省的名称空间里的任何元素或者属性。

（3）新定义的数据类型和事件类必须仅被包含在“dtype”属性为“xml”的AdditonalData类或RecordItem类中。

7.3 IODEF 的扩充实例

为了适应国内业界对安全事件的分类需求，各CSIRT之间进行安全事件交换的实际需求，同时也为了使国内的安全事件信息交换与国际接轨，实现信息互通，按照上述的IODEF扩充原则，对IODEF数据模型进行了扩展。在下面的扩展类说明中，我们只给出Schema的类型定义。

7.3.1 网页篡改事件类

类说明

该类用户描述网页篡改事件，包括服务器软件信息（如操作系统的版本、已安装系统补丁、WEB服务器软件等）、网站类型、被篡改网页的URL、篡改后网页的内容描述和篡改网页的目的等信息。

类图

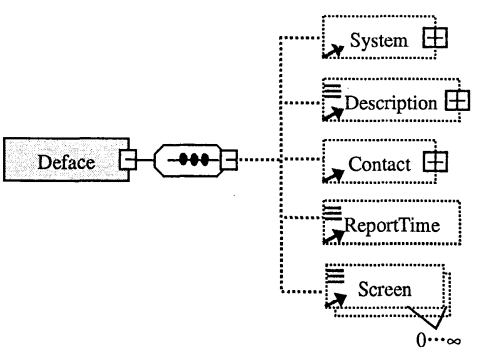


图20 网页篡改事件类

子类

— System：可选，零个或多个，表示WEB服务器的相关信息，包括如操作系统的版本、已安装系统补丁、WEB服务器软件等；

— Description：可选，零个或多个，表示被篡改后网页的内容描述；

— Contact：可选，表示网站联系人信息；

— ReportTime：可选，事件报告事件；

— Screen：可选，被篡改网页事件发生前后的页面截图。

属性

purpose：可选，枚举类型，用于说明网页篡改的目的：

— 政治相关；

— 欺骗证明；

— 攻击技术；

— 其他。

sitetype：可选，枚举类型，用于说明网站类型：

— 政府网站；

— 企业网站；

— 商业网站；

— 教育科研机构网站；

— 个人网站；

— 其他非盈利机构网站；

— 其他网站。

Schema定义

```
<xs:element name="Deface">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="System" minOccurs="0"/>
            <xs:element ref="Description" minOccurs="0"/>
            <xs:element ref="Contact" minOccurs="0"/>
            <xs:element ref="ReportTime" minOccurs="0"/>
            <xs:element name="Screen" minOccurs="0" maxOccurs="unbounded">
                <xs:complexType mixed="true">
                    <xs:attribute name="type">
                        <xs:simpleType>
                            <xs:restriction base="xs:NMTOKEN">
                                <xs:enumeration value="normal"/>
                                <xs:enumeration value="current"/>
                            </xs:restriction>
```

```
                    </xs:simpleType>
                </xs:attribute>
            </xs:complexType>
        </xs:element>
    </xs:sequence>
    <xs:attribute name="purpose" type="xs:string"/>
    <xs:attribute name="source" type="xs:string"/>
    <xs:attribute name="sitetype" type="xs:string"/>
    <xs:attribute name="country" type="xs:string"/>
    <xs:attribute name="province" type="xs:string"/>
    <xs:attribute name="isp" type="xs:string"/>
</xs:complexType>
</xs:element>
```

7.3.2 拒绝服务攻击事件类

类说明

该类用于描述拒绝服务攻击事件，包含拒绝服务攻击类型、受攻击的服务或系统、攻击的延续时间、攻击流量等信息。

类图

图21 拒绝服务攻击事件类

子类

无。

属性

dostype：枚举类型，说明拒绝服务的攻击类型：

- DOS引起的带宽占用；
- DDOS引起的带宽占用；
- 使服务或服务器崩溃或性能降低的行为；
- 通过有意触发自动的安全保护机制而使得服务不可访问。

service：枚举类型，说明受攻击的服务或系统：

- Web服务器；
- 电子商务Web服务；
- E-mail服务器；
- 内网的连通性/带宽；
- 互联网的连通性/带宽；
- 其他服务。

lastime：枚举类型，说明攻击的延续时间：

- 少于12小时；

— 12~24小时；

— 1~2天；

— 2~3天；

— 3天以上。

flux：枚举类型，说明攻击流量等信息：

— 56~64kbit/s；

— 128 kbit/s；

— 1~2Mbit/s；

— 10 Mbit/s；

— 35~45 Mbit/s；

— 100Mbit/s；

— 155 Mbit/s；

— 622 Mbit/s 或更高；

— 不知道。

Schema定义

```
<xs:element name="Dos">
    <xs:complexType>
        <xs:attribute name="dostype" type="xs:string"/>
        <xs:attribute name="lastingtime" type="xs:string"/>
        <xs:attribute name="flux" type="xs:string"/>
        <xs:attribute name="service" type="xs:string"/>
    </xs:complexType>
</xs:element>
```


7.3.3 恶意代码网站事件类

类说明

该类描述恶意代码网站时间，包含具有恶意代码网页的URL、是如何发现该网页的以及恶意代码的表现。

类图

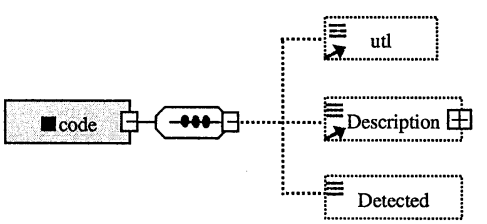


图22 恶意代码网站事件类

子类

— url：零个或一个，描述包含恶意代码网页的URL；

— Description：零个或多个，描述恶意代码的表现或症状；

— Detected：零个或一个，描述报告者如何发现恶意代码网站。

属性

无。

Schema定义

```
<xs:element name="Mcode">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="url" minOccurs="0"/>
            <xs:element ref="Description" minOccurs="0"/>
        </xs:sequence>
        <xs:attribute name="howcatch" type="xs:string"/>
    </xs:complexType>
</xs:element>
```

7.3.4 网络仿冒事件类

类说明

该类描述网络仿冒事件，包含仿冒单位、仿冒网站的URL、运行仿冒网站的主机以及仿冒目的等信息。

类图

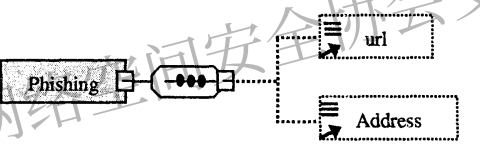


图23 网络仿冒事件类

子类

- url：零个或一个，描述仿冒网站的URL；
- Address：零个或一个，描述运行仿冒网站的主机地址。

属性

purpose：STRING，说明仿冒的目的。

name：STRING，说明被仿冒单位的名称。

Schema定义

```
<xs:element name="Phishing">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="url" minOccurs="0"/>
            <xs:element ref="Address" minOccurs="0"/>
        </xs:sequence>
        <xs:attribute name="purpose" type="xs:string"/>
        <xs:attribute name="name" type="xs:string"/>
    </xs:complexType>
```

```
</xs:element>
```

7.3.5 病毒或蠕虫事件类

类说明

该类描述与病毒、蠕虫或木马相关的事件，包含病毒名称、感染途径、特征和症状等信息。

类图

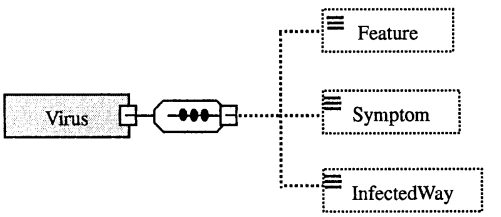


图24 病毒或蠕虫事件类

子类

— Feature：可选项，零个或一个，描述病毒、蠕虫或木马的特征，比如邮件标题、邮件正文、邮件附件等，是特征的自由格式描述。

— Symptom：可选，零个或一个。描述被感染机器或网络的症状，如系统资源耗尽、重启、某些功能失效、占用网络带宽等。

— InfectedWay：可选，零个或一个。描述病毒、蠕虫或木马可能的感染途径，比如本机存在漏洞、共享目录、弱口令、P2P、邮件、可移动介质、其他。

属性

name：STRING，表示病毒、蠕虫或木马的名字。

Schema定义

```
<xs:element name="Virus">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="Feature" type="xs:string" minOccurs="0"/>
            <xs:element name="Symptom" type="xs:string" minOccurs="0"/>
            <xs:element name="InfectedWay" type="xs:string" minOccurs="0"/>
        </xs:sequence>
        <xs:attribute name="name" type="xs:string"/>
    </xs:complexType>
</xs:element>
```

7.3.6 恶意探测扫描事件类

类说明

该类用于描述恶意探测或扫描类事件，包含报告者对扫描或探测行为的推测、扫描性质以及其他相关信息。

类图

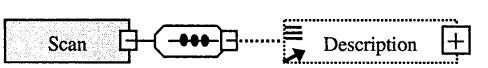


图25 恶意探测扫描事件类

子类

— Description：零个或多个，说明扫描行为的性质，如蠕虫引起的、自动扫描工具或不知道等。

属性

exception：可选，枚举类型，说明对报告人的网络而言，扫描的行为是否为异常：

— 正常；

— 端口不常被扫描；

— 端口的扫描有所增加；

— 端口的扫描增加较多；

— 不确定。

tool：可选，STRING，说明产生扫描行为的工具或蠕虫的名字。

Schema定义

```
<xs:element name="Scan">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="Description" minOccurs="0"/>
            </xs:sequence>
            <xs:attribute name="exception" type="xs:string"/>
            <xs:attribute name="tool" type="xs:string"/>
        </xs:complexType>
    </xs:element>
```


7.3.7 垃圾邮件事件类

类说明

该类用于描述垃圾邮件事件，包含邮件原始信息、邮件地址、SMTP服务器IP地址以及垃圾邮件的性质等。

类图

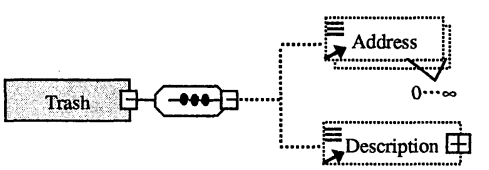


图26 垃圾邮件事件类

子类

— Description：零个或一个，描述垃圾邮件的原始信息；

— Address：零个或多个，描述邮件地址和SMTP服务器IP地址。

属性

property：枚举类型，说明垃圾邮件的性质：

— 广告；

— 政治相关；

— 散布谣言；

— 其他。

Schema定义

```
<xs:element name="Trash">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="Address" minOccurs="0" maxOccurs="unbounded"/>
            <xs:element ref="Description" minOccurs="0"/>
        </xs:sequence>
        <xs:attribute name="property" type="xs:string"/>
    </xs:complexType>
</xs:element>
```

7.3.8 非授权访问或修改数据事件类

类说明

该类用于描述非授权访问或修改数据、盗取数据事件，包含攻击者采取了何种动作、被安装的工具或文件名、被修改的系统文件名等信息。

类图

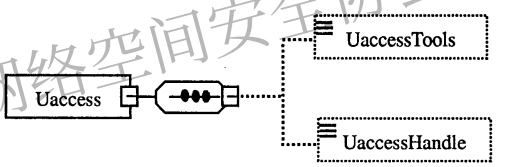


图27 非授权访问或修改数据事件类

子类

— UaccessTools：零个或一个，描述攻击者在受害机上安装的工具或文件名，以及修改的系统文件名；

— UaccessHandle：零个或一个，描述攻击者采取了何种动作，例如读/拷贝数据文件、修改/删除操作系统文件、修改/删除数据文件、安装攻击工具（如rootkit，DDoS工具）、安装方便进一步控制系统的软件（如IRC bot）、安装不知道的其他文件或不知道等。

属性

无。

Schema定义

```
<xs:element name="Uaccess">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="UaccessTools" type="xs:string" minOccurs="0"/>
            <xs:element name="UaccessHandle" type="xs:string" minOccurs="0"/>
        </xs:sequence>
```

```
        </xs:complexType>
    </xs:element>
```

7.4 实现中的注意事项

7.4.1 惟一标示符

各CSIRT通过为事件的上下文中分配一个标识符来对事件进行跟踪。在IODEF数据模型中，这个标识符是通过IncidentID类来表示的。通过这个标识符可以实现IODEF文档与IHS的联系。这就意味着相同的活动可能会由其他CSIRT使用它们自己的独特的标识符。每个CSIRT在它们的IHS中使用自己的标识符与特定的事件进行关联。事实上，IODEF为CSIRT提供一个框架以引用其他的数据，IODEF提供可选的AlternateID类来实现引用其他CSIRT相同事件的跟踪标识符。

将CSIRT名字和事件标识符组合起来就可以为特定的事件形成全局惟一的标识符。在具体实现中，建议IncidentID对象由组织名称、组织编号和时间处理流水号构成。例如国家安全中心的名称为CNCERT、国际上的组织编号为3101，2005年4月1日某IODEF文档的处理流水号000002，那么该文档的 IncidentID为<IncidentID> CNCERT#3101-20050401000002 <IncidentID>。

（1）规范定义

IODEF草案定义了数据模型，事实上，进行事件交换的双方，必须对实际应用中数据的具体用法和确切语义进行协商。这意味着IODEF文档的生成者对数据的含义和语义进一步明确和细化。这一策略要求CSIRT需要定义一份规范文件，用于明确CSIRT所生成文档内容的含义。规范文件包括以下内容。

（2）必需数据的规定

在规范文件中，必须明确规定交换数据的确切内容。IODEF基础数据模型中明确规定一部分域是必须实现的数据，而其他却是可选择的。规范文件必须是在基础数据模型的基础上，进一步定义哪些可选的数据对CSIRT而言是必须实现的。发送对方不感兴趣的信息是没有意义的。同样，不充分的信息将需要额外地沟通，并将结果作为规范文件的一部分。

不同种类的事故报告是由不同的数据类型组成的。例如，描述管理缺失的数据和描述违反规则的数据是不相同的。因此，规范文件能清晰区分事件的不同类型，并指定与事件相关的必须存在的域。

（3）语义的定义

对于给定IODEF数据模型的实现（DTD或Schema）和规范文件，IODEF文档的接收者应该能理解相关的内容。规范文件必须消除在交换过程中所有主观数据的语义歧义。同时应记录CSIRT的命名习惯。

（4）格式化的处理

关于格式化的约定应尽可能实现标准化，以便于计算机处理IODEF文件。当处理自由文本时，该标准化过程就显得特别重要。

除了内容的格式化外，IODEF文件的整体结构也进行约定。因为IODEF的数据模型是非判定性的，规范文件应该指定表达信息所需要的方式。

7.4.2 国际化和本地化

国际化和本地化是IODEF应特别关注的问题。因为，众多安全事件必须通过多CSIRT合作才能得到解决，这种合作通常是需要跨越语言的屏障。

XML已经支持不同的字符编码。这一灵活性使得可以用大部分书写语言对IODEF中的信息进行编码。此外，通过XML提供的xml:lang属性，可具体指明某一给定元素所使用的语言类型。通过xml:lang属性，IODEF的使用者能够在相同的文档中使用不同的语言。

支持不同的语言允许CSIRT本地化IODEF。然而，如果某文档的接收者不懂所使用的语言，这也不能帮助数据交换。为了确保文档的接收者至少能够粗略地了解文档的内容，数据模型必须依赖于已经标准化的枚举属性来传达含义。

7.4.3 文档的数字签名

由于在IODEF中描述的某些数据的敏感的本性，在传输中应当确保这些文档的完整性、机密性和不可否认性。尽管可以经由传输机制来提供这类保护，但还是建议对IODEF实例本身应用安全保护。然而，应用于IODEF文档特殊的保护措施（通过XML或者潜在的传输协议）应当根据合作者的需求来决定。

应用的保护措施必须使用密码技术。XML数字签名应当被用来确保信息的完整性和不可否认性，XML加密应当被用来确保IODEF文档的机密性。在使用密码技术的时候，必须处理密钥管理方面的问题（是使用对称密码还是使用公钥密码）。为了保证IODEF-Document处理环境的安全，必须应用全面的安全措施。XML数字签名可以应用于任何的数字内容（数据对象），包括XML。一个XML数字签名可以应用到一个或多个资源的内容。

附录A给出了一个带有XML签名的IODEF文档例子。

7.4.4 文档的加密

待加密的数据可以是任意的数据（包括XML文档），XML元素或者XML元素的内容。对数据加密的结果是一个XML加密的EncryptedData元素，该元素包含（或者经由其一个子元素内容）或者标识（经由URI引用）密文数据。

当对某个XML元素或者元素内容进行加密时，EncryptedData元素在XML文档加密的版本中分别替换到元素或者元素内容。

附 录 A
（资料性附录）
事件描述实例

A.1 红色代码检测通告

下面的消息是一个典型的安全事件案例，其中一个主机被感染病毒。最初的报告是通过电子邮件发送的，随后显示的IODEF-Document详细说明CSIRT和它的委托人之间的通信。对于CSIRT来说，委托人是一个联系人，它负责协调其站点内一些必要的活动。

A.1.1 红码检测通知（最初的报告）

From e-citizen@hisdomain.de

Date: 13 Sep 2001 23:19:24 -0000

From: e-citizen@hisdomain.de

To: cert-for-ourdomain.pl@ourdomain.pl

Subject: 10.1.1.2 - Code Red Virus detected

Automated message，you don't have to reply to this email.

Your system with the IP number 10.1.1.2 seems to be infected with the Code Red virus.

For more information see http://www.incidents.org/react/code_redII.php

Please fix the problem or inform a person who is responsible for that machine to do so.

>From our web server logs （Port 80）:

10.1.1.2 - - [13/Sep/2001:18:11:21 +0200] "GET

/default.ida?XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

A.1.2 红码检测通知（CSIRT响应）

以下是XML的描述：

```
<!DOCTYPE IODEF-Document SYSTEM "IODEF-Document.dtd" [
    <!ENTITY % x-extension SYSTEM "CCERT.dtd">
    %x-extension;
]>
    <IODEF-Document version="1.0">
        <Incident restriction="need-to-know" purpose="handling">
```

```
            <IncidentID
name="CERT-FOR-OUR-DOMAIN.PL">CERT-FOR-OUR-DOMAIN.PL#189</IncidentID>
            <IncidentData>
                <Description>Host sending out Code Red probes</Description>
                <ReportTime>2001-09-13T23:19:24+00:00</ReportTime>
                <Expectation category="other">
                    <Description>Track and clean host</Description>
                </Expectation>
                <Assessment>
                    <Impact severity="low" completion="failed" type="none"></Impact>
                </Assessment>
                <Contact role="creator" role="irt" type="organization">
                    <name>CERT-FOR-OUR-DOMAIN.PL</name>
                    <Email>cert-for-our-domain.pl@ourdomain.pl</Email>
                </Contact>
                <Contact role="tech" type="organization">
                    <name>Constituency-contact for 10.1.1.2</name>
                    <Email>Constituency-contact@10.1.1.2.pl</Email>
                </Contact>
                <History>
                    <HistoryItem type="notification">
                        <IncidentID
name="CERT-FOR-OUR-DOMAIN.PL">CERT-FOR-OUR-DOMAIN.PL#189
    </IncidentID>
                        <Description>Notification sent to Constituency-contact@10.1.1.2.pl
    </Description>
                        <DateTime>2001-09-14T08:19:01+00:00</DateTime>
                    </HistoryItem>
                </History>
                <EventData>
                    <System category="source">
                        <Node>
                            <Address category="ipv4-addr">10.1.1.2</Address>
                        </Node>
                    </System>
                    <System category="target">
                        <Service>
```

```
                    <port>80</port>
                </Service>
            </System>
            <Record>
                <RecordData>
                    <DateTime>2001-09-13T18:11:21+02:00</DateTime>
                    <Description>Web-server logs</Description>
                    <RecordItem>  10.1.1.2  -  -  [13/Sep/2001:18:11:21  +0200]  "GET
/default.ida?XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
                    </RecordItem>
                </RecordData>
            </Record>
        </EventData>
    </IncidentData>
</Incident>
</IODEF-Document>
```


A.2 带有XML签名的IODEF文档

下面给出对http://www.ccert.edu.cn/IODEF/Example5做XML数字签名的例子。

```
<?xml version="1.0" encoding="UTF-8"?>
<Signature xmlns="http://www.w3.org/2000/09/xmldsig#">
  <SignedInfo>
    <CanonicalizationMethod
Algorithm="http://www.w3.org/TR/2000/WD-xml-c14n-20001011" />
    <SignatureMethod
Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1" />
    <Reference URI="http://www.ccert.edu.cn/IODEF/Example5">
      <DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
      <DigestValue>60NvZvtdTB+7UnlLp/H24p7h4bs=</DigestValue>
    </Reference>
  </SignedInfo>
  <SignatureValue>
THQJyd3C6ww/OJz07P4bMOgjqBdznSUOsCh6P+0MpF69w2tln/PFLdx/EP4/VKX2uW0gxKb8QgDf46e
XCsulAzx0Yy2bvmRZ+kJjm3B1cVP+1Hpd3cxC7TUDdgptEFbJSTRLsLMFQ8v/l1xgAFmwEa3daF6fSLHiTy
N8vXxR8g=</SignatureValue>
```

<KeyInfo>
<KeyValue>
<RSAKeyValue>
<Modulus>
CiukpgOaOmrq1fPUTH3CAXxuFmPjsmS4jnTKxrv0w1JKcXtJ2M3akaV1d/karvJlmeao20jNy9r+/vKwibj
M77F+3bIkeMEGmAIUnFciJkR+ihO7b4cTuYnEi8xHtu4iMn6GODBoEzqFQYdd8p4vrZBsvs44nTrS8qyyhba
648=
</Modulus>
<Exponent>
AQAB
</Exponent>
</RSAKeyValue>
</KeyValue>
<X509Data>
<X509SubjectName>
CN=WANG Chang-Ji，OU = Network Research Center，O= Tsinghua University，C=CN，E=
wangcj@cernet.edu.cn
</X509SubjectName>
<X509IssuerSerial>
<X509IssuerName>
CN=Student CA，OU= Tsinghua Certificate Center，O = Tsinghua University，C = CN，E
= wangcj@ccert.edu.cn
</X509IssuerName>
<X509SerialNumber>
00B9 12B4 AE99 5C91 4D83 6EAC 8F2A 2B74 27
</X509SerialNumber>
</X509IssuerSerial>
<X509Certificate>
MIIDbzCCAlegAwIBAgIRALkStK6ZXJFNg26sjyordCcwDQYJKoZIhvcNAQEFBQAwgYoxIjAgBgkqhkiG9
w0BCQEWE3dhbmdjakBjY2VydC5lZHUuY24xCzAJBgNVBAYTAkNOMRwwGgYDVQQKExNUc2luZ2h1
YSBVbml2ZXJzaXR5MSQwIgYDVQQLExtUc2luZ2h1YSBDZXJ0aWZpY2F0ZSBDZW50ZXIxEzARBgNV
BAMTClN0dWRlbnQgQ0EwHhcNMDMwOTI0MDYxMDAwWhcNMTMwOTE1MDEzODQ4WjCBijEjMC
EGCSqGSIb3DQEJARYUd2FuZ2NqQGNlcm5ldC5lZHUuY24xCzAJBgNVBAYTAkNOMRwwGgYDVQQK
ExNUc2luZ2h1YSBVbml2ZXJzaXR5MSAwHgYDVQQLExdOZXR3b3JrIFJlc2VhcmNoIENlbnRlcjEWMBQ
GA1UEAxMNV0FORyBDaGFuZy1KaTCBnzANBgkqhkiG9w0BAQEFAAOBjQAwgYkCgYEA06JgAQwQi
D2b5kT3T2mN3OINGhLlPmpPSxCSXJxDmM/y6ZrSDfGI6McKAXNB8s0vecggFpuNsFhqKyySCyYC50Mrt
fdiHeWiUkFkAhOUjA2ztw5XUPN8TGK1t8PcfaUFzpd0ow6tnljvCQ6Pdx+cMoet3R5qcslLYRM2x7mKm6cC

AwEAAaNSMFAwHwYDVR0jBBgwFoAUMfT/jGiE6j4o2Fvm/RbaChufdzIwDgYDVR0PAQH/BAQDAgP4
MB0GA1UdDgQWBBQqsl6kAh+70/3BHxhhKWB+3nJMRjANBgkqhkiG9w0BAQUFAAOCAQEAUygJ2r3B
tw1FIbj+K6OXlryeX6gt/yzvTsRnujab3C4Hc3e9buQNv1Bx0R1EE3MGsvZ9e7BRz0FCxCk1E7lNJu6k5Q8KN
gPJsYPQDq8jzbPz1XEzrm8X96edBi7sulOkV8+otWPUeE8Y3q3JUyxp9i8ykoFtorLMiM583xwGimp2meZyT
H40IKHXznVxXNeGMpXyYoaMIkXuhSa7xImq7PMw0bxF19tdOTvD2JFeIy7Nped2h8RIhZlBeG/nVrJN8apj
wTrpPL1yl4tioepwCtqIGRChg3+bjW7LAFTLtPqJzCWahqOq0elvlZBLloqybv+TwazWFjbXdicwItd5jg==

```
    </X509Certificate>
        </X509Data>
      </KeyInfo>
    </Signature>
```

A.3 使用XML加密的IODEF文档的例子

下面给出对11.1节中IncidentData元素的加密，

```
    <IODEF-Document version="1.0">
        <Incident restriction="need-to-know" purpose="handling">
            <IncidentID name="CERT-FOR-OUR-DOMAIN.PL">
CERT-FOR-OUR-DOMAIN.PL#189</IncidentID>
        <EncryptedData Type='http://www.w3.org/2001/04/xmlenc#Element'
       xmlns='http://www.w3.org/2001/04/xmlenc#'>
        <CipherData>
<CipherValue>A23B45C56HUSEDLIHUEFDJDF03LJ9DSJIKJODSHOIJ</CipherValue>
        </CipherData>
      </EncryptedData>
        </Incident>
    </IODEF-Document>
```

参 考 文 献

[1] Demchenko,Y.,Hiroyuki,H. and G. Keeni,"Requirements for Format for Incident Report Exchange",RFC XXX,September 2003.

[2] World Wide Web Consortium,"Extensible Markup Language（XML）1.0 （Second Edition）",October 2000,<http://www.w3.org/TR/2000/REC-xml-20001006>.

[3] World Wide Web Consortium,"Namespaces in XML",,January 1999,http://www.w3.org/TR/REC-xml-names/>.

[4] World Wide Web Consortium,"Extensible Stylesheet Language （XSL）Version 1.0",October 2001,<http://www.w3.org/TR/xsl/>.

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

[6] Alvestrand,H.,"Tags for the Identification of Languages",RFC 3066,January 2001.

[7] Curry,D. and H. Debar,"Intrusion Detection Message Exchange Format",RFC XXX,January 2003.

[8] Berners-Lee,T.,Fielding,R. and L. Masinter,"Uniform Resource Identifiers （URI）: Generic Syntax",RFC 2396,August 1998.

[9] Freed,N.,"IANA Charset Registration Procedures",BCP 2278,January 1998.

[10] Mills,D.,"Network Time Protocol （Version 3）Specification,Implementation,and Analysis",BCP 2278,March 1992.

[11] Mills,D.,"Simple Network Time Protocol （SNTP）Version 4 for IPv4,IPv6 and OSI",RFC 2030,October 1996.

[12] Wahl,M.,"A Summary of the X.500（96）User Schema for use with LDAPv3",RFC 2256,December 1997.

[13] Resnick,P.,"Internet Message Format",RFC 2822,April 2001.

[14] Klyne,G. and C. Newman,"Date and Time on the Internet: Timestamps",RFC 3339,July 2002.

[15] International Organization for Standardization,"International Standard: Data elements and interchange formats - Information interchange - Representation of dates and times",ISO 8601,Second Edition,December 2000.

[16] Eastlake 3rd,D.,Reagle,J. and D. Solo,"（Extensible Markup Language）XML-Signature Syntax and Processing",RFC 3275,March 2002.

[17] Imamura,T.,Dillaway,B. and E. Simon,"XML Encryption Syntax and Processing,W3C Recommendation",December 2002,<http://www.w3.org/TR/2002/REC-xmlenc-core-20021210/>.

[18] International Organization for Standardization,"International Standard: Codes for the representation of currencies and funds,ISO 4217:2001",ISO 4217:2001,August 2001.

[19] Rumbaugh,J.,Jacobson,I. and G. Booch,"The Unified Modeling Language Reference Model,ISBN 020130998X,Addison-Wesley",1998.

[20] Helme,A. and R. Danyliw,"The IODEF Implementation Guide,document to be created by the INCH WG",2003.

中华人民共和国
通信行业标准
网络安全事件描述和交换格式
YD/T 1827-2008
*
人民邮电出版社出版发行
北京市崇文区夕照寺街14号A座
邮政编码：100061
北京新瑞铭印刷有限公司印刷

*
开本：880×1230　1/16　　2008年10月第1版
印张：3.5　　2008年10月北京第1次印刷
字数：90千字
ISBN 978-7-115-1716/08-160
定价：35元
本书如有印装质量问题，请与本社联系　电话：(010)67114922