Information Framework (SID)

Security Entity Definitions

Addendum GB922-7 Security

Version 1.5



October, 2012

Notice

No recipient of this document and code shall in any way interpret this material as representing a position or agreement of TM Forum or its members. This material is draft working material of TM Forum and is provided solely for comments and evaluation. It is not “Forum Approved” and is solely circulated for the purposes of assisting TM Forum in the preparation of final material in furtherance of the aims and mission of TM Forum.

Although it is copyrighted material of TM Forum:

* Members of TM Forum are only granted the limited copyright waiver to distribute this material within their companies and may not make paper or electronic copies for distribution outside of their companies.
* Non-members of the TM Forum are not permitted to make copies (paper or electronic) of this draft material other than for their internal use for the sole purpose of making comments thereon directly to TM Forum.
* If this material forms part of a supply of information in support of an Industry Group Liaison relationship, the document may only be used as part of the work identified in the Liaison and may not be used or further distributed for any other purposes

Any use of this material by the recipient, other than as set forth specifically herein, is at its own risk, and under no circumstances will TM Forum be liable for direct or indirect damages or any costs or losses resulting from the use of this material by the recipient.

This material is governed, and all recipients shall be bound, by all of the terms and conditions of the Intellectual Property Rights Policy of the TM Forum (<http://www.tmforum.org/Bylaws/1094/home.html>) and may involve a claim of patent rights by one or more TM Forum members or by non-members of TM Forum.

Direct inquiries to the TM Forum office:

240 Headquarters Plaza,

East Tower – 10th Floor,

Morristown, NJ  07960 USA

Tel No.  +1 973 944 5100

Fax No.  +1 973 944 5110

TM Forum Web Page: [www.tmforum.org](http://www.tmforum.org/)

Table of Contents

[Notice 2](#_Toc339010740)

[Table of Contents 3](#_Toc339010741)

[Table of Figures 4](#_Toc339010742)

[List of Tables 5](#_Toc339010743)

[Executive Summary 6](#_Toc339010744)

[1. Business Entities 9](#_Toc339010745)

[1.1. Security Domain 9](#_Toc339010746)

[1.1.1. Introduction 9](#_Toc339010747)

[1.1.2. Overview of ‘Security’ 9](#_Toc339010748)

[1.2. Security Entity 10](#_Toc339010749)

[1.2.1. Security Entity CPE Relationships 11](#_Toc339010750)

[1.3. Security Event 12](#_Toc339010751)

[1.3.1. Security Event – Basic Entities 13](#_Toc339010752)

[1.3.2. Security Event Specification 13](#_Toc339010753)

[1.3.3. Security Event Specification Characteristics 13](#_Toc339010754)

[1.3.4. Security Event Characteristic Value 14](#_Toc339010755)

[1.3.5. Security Event Types 15](#_Toc339010756)

[1.4. Security Threat 16](#_Toc339010757)

[1.4.1. Security Threat Basic Entities 16](#_Toc339010758)

[1.4.2. Security Threat Technique 18](#_Toc339010759)

[1.4.3. Security Threat Technique Specification 18](#_Toc339010760)

[1.4.4. Security Threat Tool 19](#_Toc339010761)

[1.4.5. Security Threat Tool Specification 20](#_Toc339010762)

[1.4.6. Security Threat Actor 21](#_Toc339010763)

[1.4.7. Security Threat Indicator 21](#_Toc339010764)

[1.4.8. Security Threat Indicator Specification 22](#_Toc339010765)

[1.5. Security Vulnerability 22](#_Toc339010766)

[1.5.1. Security Vulnerability Association with Other Business Entities 24](#_Toc339010767)

[1.5.2. Vulnerability Scoring 25](#_Toc339010768)

[1.6. Security Incident 31](#_Toc339010769)

[1.6.1. Security Incident Assessment 31](#_Toc339010770)

[1.7. Business Entity Definitions – Security 33](#_Toc339010771)

[2. Administrative Appendix 34](#_Toc339010772)

[2.1. About this document 34](#_Toc339010773)

[2.2. Document History 34](#_Toc339010774)

[2.2.1. Version History 34](#_Toc339010775)

[2.2.2. Release History 34](#_Toc339010776)

[2.3. Company Contact Details 35](#_Toc339010777)

[2.4. Acknowledgments 35](#_Toc339010778)

Table of Figures

Figure A – Security Management Processes and States (Level 0) 6

Figure B – Illustration from NSA “Network Defense Data Model, An Overview and Implementation Guide” 7

Figure SE.00 - Security Overview 10

Figure SE.01 - Security Entity 11

Figure SE.02 - Security Entity CPE Relationships 12

Figure SE.03 - Security Event – Basic Entities 13

Figure SE.04 – Security Event Specification 13

Figure SE.05 – Security Event Specification Characteristic 14

Figure SE.06 – Security Event Specification Characteristic Value 14

Figure SE.07– Security Event Types 16

Figure SE.08 – Security Threat Basic Entities 17

Figure SE.09 – Security Threat Technique 18

Figure SE.10 – Security Threat Technique Specification 19

Figure SE.11 – Security Threat Tool 20

Figure SE.12– Security Threat Tool Specification 20

Figure SE.13 – Security Threat Actor 21

Figure SE.14 – Security Threat Indicator 22

Figure SE.15 – Security Threat Indicator Specification 22

Figure SE.16 – Security Vulnerability 24

Figure SE.17 – Security Vulnerability Association with Other Business Entities 25

Figure SE.17a – Security Vulnerability Category Types 26

Figure SE.17b – Security Vulnerability Scoring Definitions 27

Figure SE.17c – Security Vulnerability Metrics 28

Figure SE.17d – Security Vulnerability Score 29

Figure SE.17e – Security Vulnerability Score Algorithm 30

Figure SE.18 – Security Incident 31

Figure SE.19 – Security Incident Assessment 32

List of Tables

Insert if applicable

Executive Summary

Security is an essential part of network operations and management. Today’s threats vary from system-wide fraud to service disruption. While this has always been true, the risk is greater today, because dependencies, criticality, threats, probabilities and consequences of security incidents are all greater than they have been previously. For example:

* Networks are considered “critical infrastructure”, so much so that governments are taking responsibility for ensuring they stay up and running
* Cyber threats have no geographical boundaries, and their signatures are constantly evolving
* The use of hybrid and public clouds requires new security and management controls
* Personal data is being compromised at record rates

Simply stated, Security Management is responsible for providing telecommunications networks and systems protection from unauthorized access.

It has been said that the most significant network and cyber security challenges include: reducing the attack surface, providing diagnostic capability and engaging with counter measures, and reducing anonymity on the network, so it is known who is there. These challenges demonstrate the proactive and reactive postures of Security Management.

As the world-class industry standards/best practices body for communications service providers and the surrounding business ecosystem, the TM Forum is committed to addressing all essential aspects of network operations and management in its frameworks. This includes Security Management.

Ultimately,our goal is to produce TM Forum Frameworx enriched with Security Management guidance, at a level of detail suitable for vendors to certify products against. The “big picture” view of Security Management in the TM Forum Frameworx content is depicted in the following diagram and consists both of high-level process components and operational states, many of which may be active at any point in time.

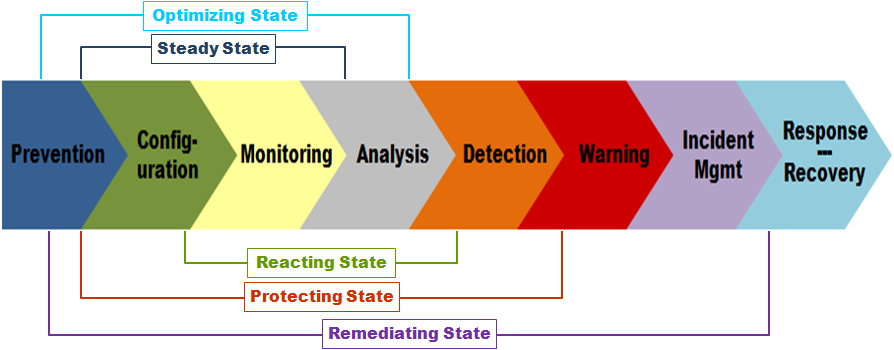


Figure A – Security Management Processes and States (Level 0)

In January 2010, the TM Forum received a generous contribution of intellectual property from the National Security Agency (NSA). This contribution, a Net-Defense (NetD) data model, Implementation Guide, and accompanying data dictionary, has provided the TM Forum valuable insight on how to model information related to threats, incidents, events, assets, and vulnerabilities. Several standards, such as the National Institute of Standards and Technology (NIST) Security Content Automation Protocol (SCAP) suite of standards, were embraced during the creation of the NetD model.

Figure B – Illustration from NSA “Network Defense Data Model, An Overview and Implementation Guide”

This contribution combined with strong member commitments to this topic have expedited projects aimed at integrating Security Management into the TM Forum Frameworx.

At the highest level, key information revolves around understanding:

* What assets defenders are charged with protecting
* What vulnerabilities the assets have
* What threats are targeting the vulnerabilities on those assets
* What events are occurring on the network
* What incidents result from assessments of the events
* What systems are, or may be impacted by successful exploits of vulnerabilities
* Where assets, and their associated vulnerabilities, exist within the defense-in-depth infrastructure represented by surrounding networks
* What policies are in effect that direct how devices and networks should be configured, how vulnerabilities are remediated, and how incidents and events are to be recorded and reported?

Note: While the TM Forum has incorporated the NSA’s “Network Defense Data Model” into the Information Framework (SID), classes and attributes associated with “missions”, “operations”, “policies”, and “systems” were intentionally excluded.

In order to align with the NIST SCAP Release 1.2, the TM Forum extended the Information Framework Enterprise Security ABE with Security Vulnerability Scoring for the Frameworx 12.5 release. The NIST SCAP standard supports three types of scoring to calculate the severity of a particular vulnerability: the Common Configuration Scoring System (CCSS)[[1]](#footnote-1), the Common Vulnerability Scoring System (CVSS)[[2]](#footnote-2), and the Common Misuse Scoring System (CMSS)[[3]](#footnote-3). While each of the scoring systems applies a different algorithm to calculate a score, the approach to scoring (use of metrics, standard values, and interim calculations) between the three systems is quite similar. So the TM Forum was able to model the Security Vulnerability Scoring classes, attributes and relationships in a generalized way that in the end supports all three scoring systems.

# Business Entities

## Security Domain

### Introduction

When incorporating key Security Management concepts into the SID model, the modelers focused on the five primary concepts: assets, threats, vulnerabilities, events, and incidents, related information, and their relationships.

Security was added to the Enterprise Risk area of the SID because of their very nature. Each entity was qualified by the name “Security” to distinguish these types of entities as specific to Security. For example, there are other types of entities that deal with Incidents and Events.

Also, the Device (Asset) entity was generalized to accommodate Vulnerabilities, Threats, Events, and Incidents associated with any type of SID entity, such as Services, Products, and other types of Resources, such as connections.

The key entities actually represent groups of entities. Therefore, they were added as ABEs.

### Overview of ‘Security’

When incorporating key Security Management concepts into the SID model, the modelers focused on the five primary concepts: assets, threats, vulnerabilities,

The following entities are core to the security model:

SecurityEntity: Used to describe any asset that exposes security related vulnerabilities, and has the potential for security related events.

SecurityVulnerability: Potential weakness that may be exploited by an adversary or threat.

SecurityThreatExploit and SecurityThreatTechnique: Threats represent entities and characteristics of entities that may maliciously or non-maliciously exploit vulnerabilities.

SecurityThreatActor: Use “techniques” such as social engineering, or phishing to maneuver assets into positions where they can use “exploits” to take advantage of weaknesses to gain access or information that is either inherently valuable, or can be used to gain further access.

SecurityEvents and SecurityIncidents: Incidents and events are recorded and reported. The primary difference is that incidents result from assessment(s) of event(s).

SecurityIncidentImpact: Supports information such as, how the incident was allowed to happen, what threat actor caused it, and how “bad” the damage is.

The model shows that a Security Entity, such as a network element, exhibits Vulnerabilities, which may be taken advantage of by a Security Threat Technique, specifically targeted by a Security Threat Exploit. A Security Event exposes the e Vulnerability and may result in a Security Incident being created to resolve and fix the Vulnerability.

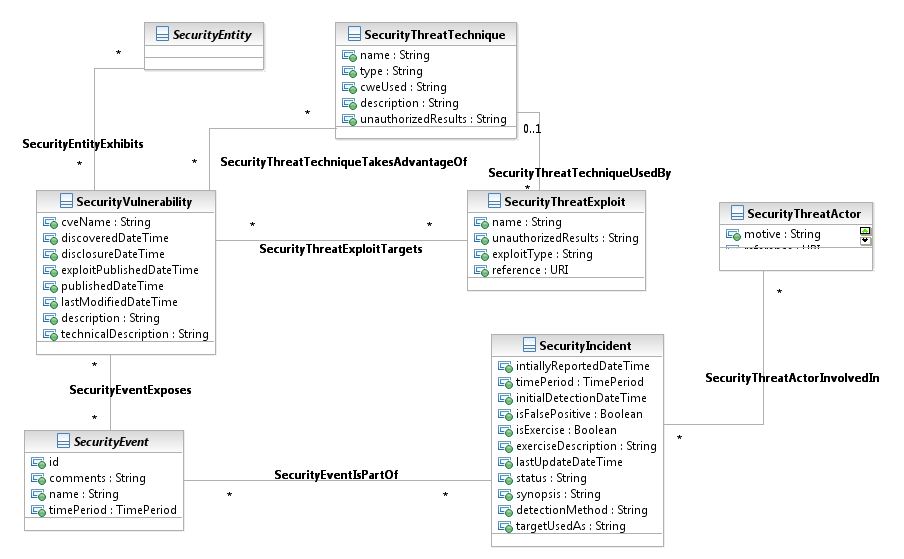


Figure SE.00 - Security Overview

## Security Entity

Examples of a Security Entity include devices, facilities, people, policies, circuits, networks, systems, and organizations.

Some aspect of the asset model is envisioned to be used by all parties engaged in network defense.

SecurityEntity has been sub-typed into three entities:

* ResourceSecurityEntity, which recognizes physical assets like facilities, equipment, systems, etc.,
* PartySecurityEntity, which recognizes organizations, people, etc., and
* PartyRoleSecurityEntity, which recognizes that a Party can have one or more roles.

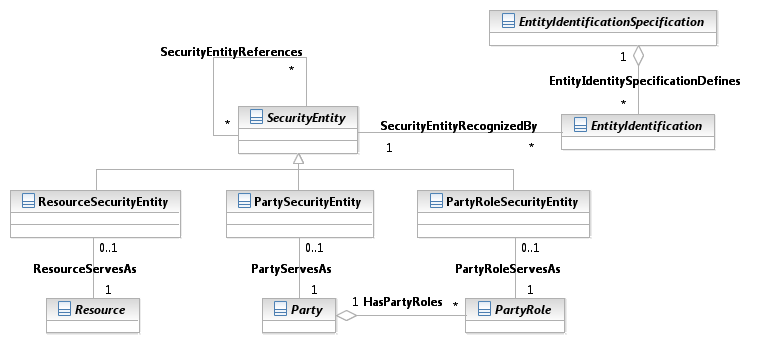


Figure SE.01 - Security Entity

Following SID convention, a SecurityEntity, which is derived from Entity, has one or more identification (EntityIdentification).

### Security Entity CPE Relationships

Common Platform Enumeration (CPE) is part of the NIST SCAP set of standards. It is a structured naming scheme for information technology platforms (hardware, operating systems, and applications), based upon the generic syntax for Uniform Resource Identifiers (URI). The CPE Specification includes the naming syntax and conventions for constructing CPE Names from product information, a dictionary (and associated XML Schema) that holds a collection of all known CPE Names as well as a binding of descriptive and diagnostic information, a language for creating complex platform descriptions, and a matching algorithm. For the up-to-date CPE Dictionary, and for the complete list of abbreviations and formatted names to be used, please visit the CPE web site at:

<http://cpe.mitre.org/>

In the SID, a CPE is associated to the three entities it can identify: hardware, software, and operating systems. Inclusion of CPE into the SID enables correlation between resources that are running a specific version of an operating system or software and the security vulnerabilities associated with the operating system or software. Outside of Security Management, CPE also provides a naming convention that facilitates inventory management use cases by providing a common name for software, hardware, and operating systems deployed in an IT environment.



Figure SE.02 - Security Entity CPE Relationships

The CPE Name conforms to the following structure:

cpe:/ *{part}* : *{vendor}* : *{product}* : *{version}* : *{update}* : *{edition}* : *{language}*

## Security Event

The event model is designed to carry information about Security Events – activities that are sensed on networks and resources.

A small set of core information about any given event is considered to be common across all event reporting. That data consists of information such as start time, stop time, unique identifier, a name, and data about the sensor that was used to collect the information. However, other information collected about events varies significantly based on the type of activity that is reported. In order to deal with the requirement to track multiple event types while still enhancing the probability of successful correlation, the event models described in this section all build on the core event data set, which allows for correlation based on time, sensor, location, unique identifier, and other functions of the collecting sensor. Each event type then adds attributes to describe the particular type of activity it monitored.

### Security Event – Basic Entities

In the NetD Data Model contribution, events were assumed to be sensed, or collected, by a network or host-based sensor, such as an Intrusion Detection System (IDS). In the SID, the collection method was generalized to accommodate three types of security event collections methods: Resource, Party, and Party Role.

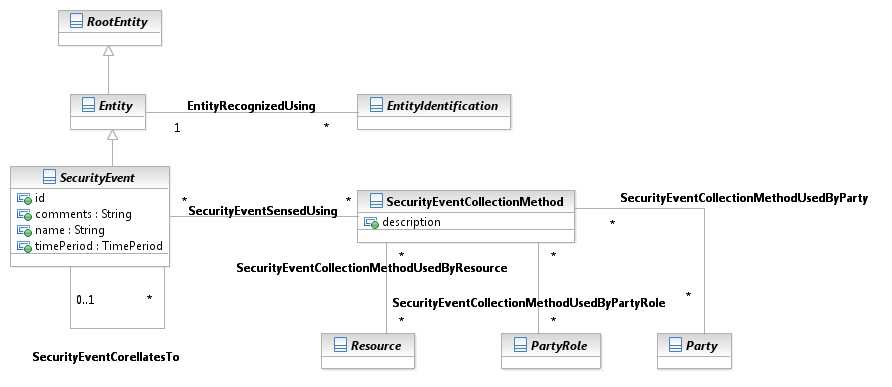


Figure SE.03 - Security Event – Basic Entities

### Security Event Specification

Security Event Specification describes the Security Event.

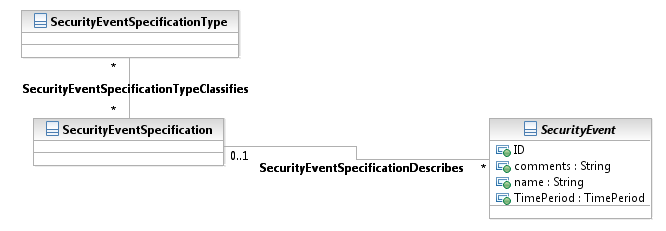


Figure SE.04 – Security Event Specification

### Security Event Specification Characteristics

The entity specification and characteristic pattern were added to the Security Event ABE to enable the dynamic creation of new types of security events with new attributes.  The predefined event types were all derived from the NetD Data Model, which used multiple existing standards, as well as DoD and industry Security information and event managers, but it current delineation of event types is not expected to be comprehensive. GB922 Addendum 1R Common Business Entity Definitions – Root Business Entities provides more detailed information on how to define new attributes through the entity and characteristic specifications.

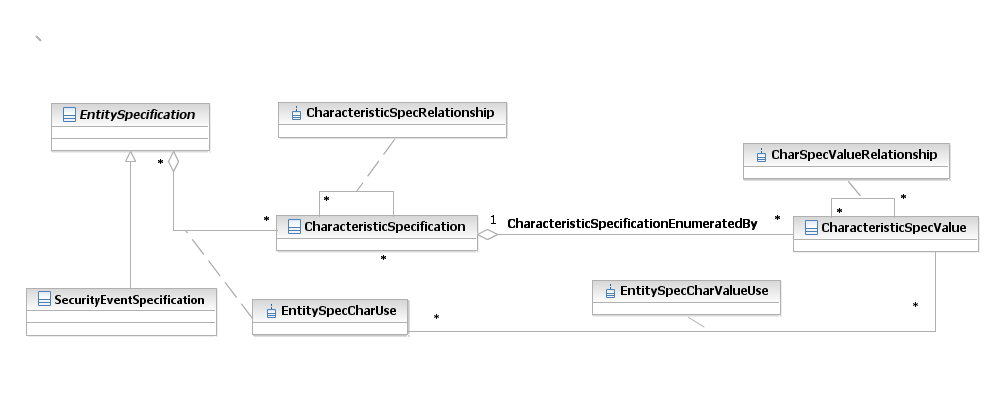


Figure SE.05 – Security Event Specification Characteristic

### Security Event Characteristic Value

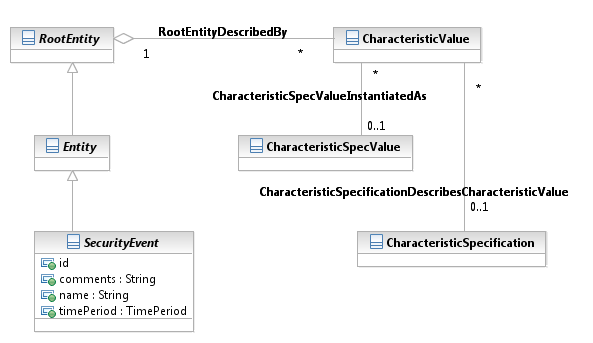


Figure SE.06 – Security Event Specification Characteristic Value

### Security Event Types

Several event types are defined, primarily as placeholders, pending development of better standards and guidance on how the data elements they may contain are defined.

#### Anomaly Event

The Anomaly Event type is one of these types. The elements that are used to extend the base event type for anomalies consist of a text field to describe the anomaly and a text field to contain log data.

#### Network Access Control (NAC)

Network Access Control (NAC) events are generated whenever new devices and/or users attempt to gain access to a network. The NAC event record will record the name of the network where access was attempted, the access decision (permit/deny), IP and MAC address of the device/user, and the name provided on the credential that was used to gain access – X.509 subject name, MAC address, or other name. Network access may be granted on the basis of either device or user credentials depending on the local security policy.

#### Network Address Translation (NAT)

Network Address Translation (NAT) events are recorded so that network defenders can discover the true originating IP address and port number of a device, which has sent information through a firewall proxy or another type of proxying device, which changes the internal source IP address. The NAT event will record the internal source IP address and port number, the IP address and port number it was translated to, as well as the external destination IP address and port number.

#### Signature

Signature events are the most mature of the event model. The signature event model is composed to meet the reporting requirement of signature-based applications, to include IDSs, IPSs, antivirus applications, anti-spyware applications, and any other sensing technology that is signature based. Core signature event information includes data the sensor collected, including the total amount of packets collected and the total bytes of data collected.

#### Network Flow

The Network Flow event type directly mirrors a network flow as transmitted by the Internet Protocol Flow Information eXport (IPFIX) standard. A net flow is a summary set of metadata about a one-way portion of a network session. A net flow will not provide any layer 7 information about the data that was transmitted, but will instead summarize information about where the transaction originated, where it terminated, how long it lasted, how many bytes were passed, and what ports and protocols were used. These flows, while not exclusively security events, do aid attack analysis by highlighting traffic trends that is abnormal to normal business and could indicate a variety of security incidents.

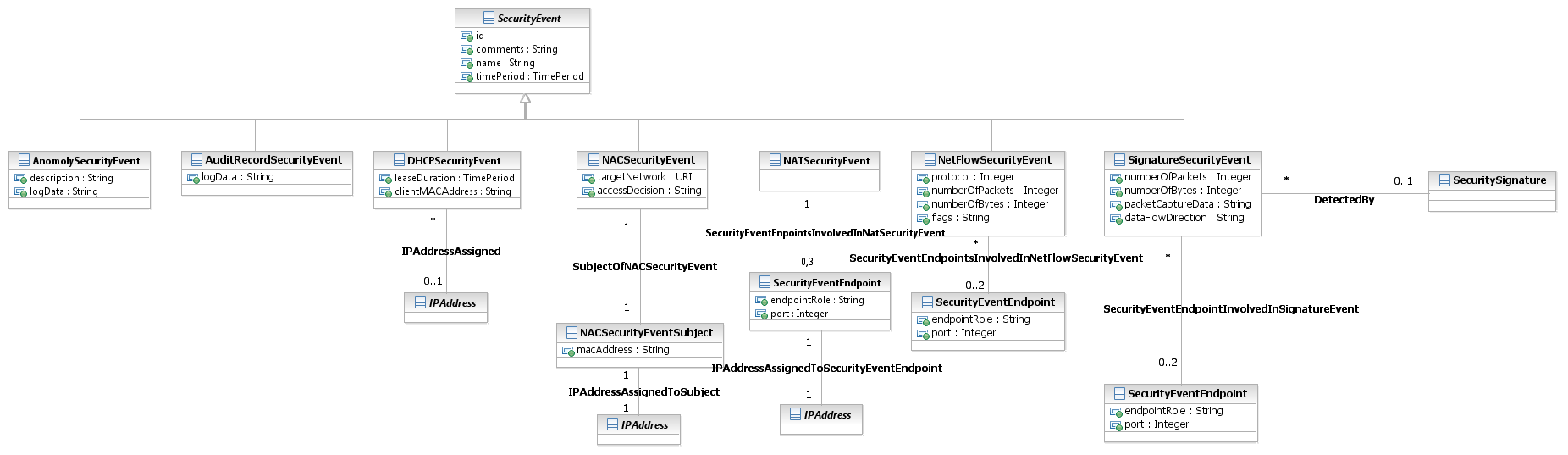


Figure SE.07– Security Event Types

## Security Threat

The vulnerability portion of the model is designed to carry information about potential weaknesses in hardware, applications, and operating systems that may be exploited by an adversary or threat. (The original contribution was designed around the NIST National Vulnerability Database (NVD) and tools.)

The threat area of the model is constructed to represent entities and characteristics of entities that may maliciously or non-maliciously exploit vulnerabilities on network-connected devices.

Much of the information in the threat model describes elements that commonly trigger signatures or algorithms for detecting malicious code or behaviors. In order to better understand the threat environment, it’s necessary to be aware of common attacks that are being attempted. Summaries of events, which may represent successful and unsuccessful attacks against systems and incident reports that report successful compromises, are required.

### Security Threat Basic Entities

#### Threat Actor

At its highest level, a Security Threat Actor can be a person, an organization, or a tool that has been created and released onto networks, but no longer has any relationship with its original creator. As can be inferred, organizations can be made up of subordinate organizations and/or people. Both organizations and people can use tools, so any given threat can be composed of organizations, people, and tools.

#### Threat Indicators

At its highest level, the ability to detect and describe threats is key to the vast majority of use cases for the threat model. To support the ability to detect threats, a large part of the threat model deals with describing “indicators” of threat activity or presence. Indicators of specific threat actors or general threat activity range from commonly used threat IP connection endpoints, to dynamic and static DNS names, to indicators showing commonly used techniques such as suspicious email attachments, SQL injections, known malicious PKI certificates, and others. Other threat indicators may be related to suspicious traffic that is generally prohibited on networks.

#### Threat Techniques

At its highest level The Threat Actors can use “techniques” such as social engineering, or phishing to maneuver devices and/or network users into positions where they can use “exploits” to take advantage of weaknesses the victim has to gain access or information that is either inherently valuable, or that can be used to gain further access.

#### Threat Tools

Organized and distributed software that can be used to attack an IT resource or service. Threat tools include viruses, botnets, and other software that enables or automates a threat actor’s attack.

#### Threat Exploit

A threat exploit takes advantage of a vulnerability to gain access or cause an adverse effect.

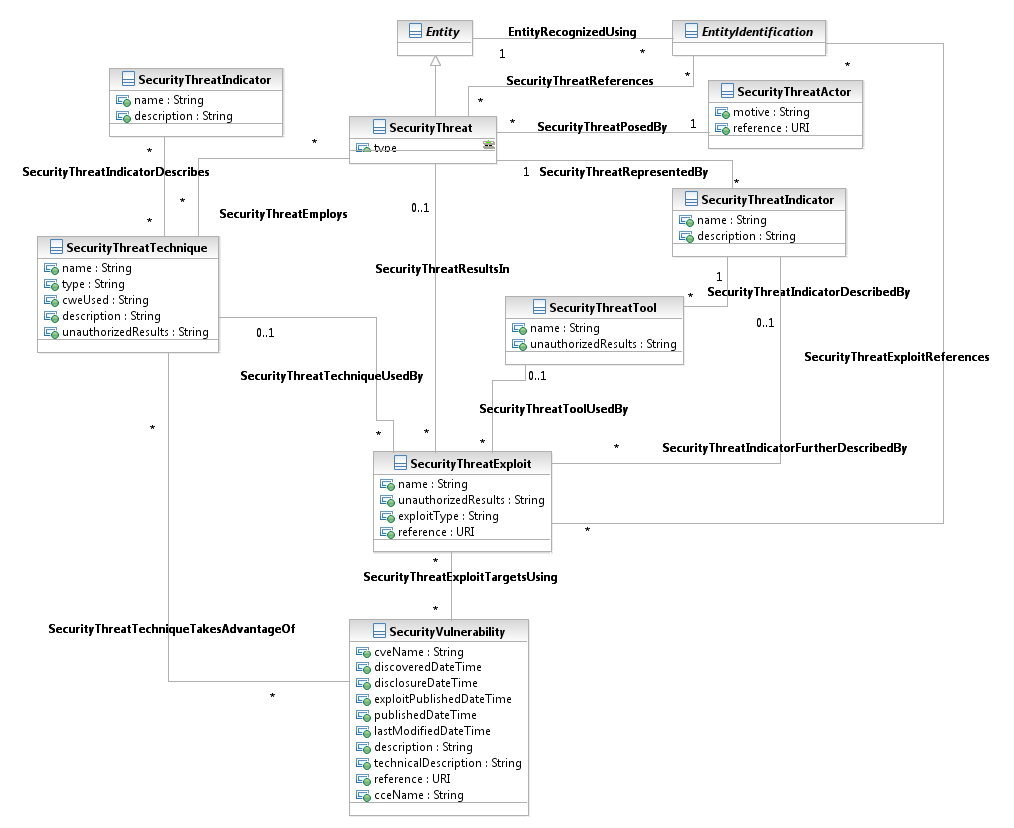


Figure SE.08 – Security Threat Basic Entities

### Security Threat Technique

A threat actor may use multiple techniques to exploit a vulnerability on an IT resource. The existing Security Threat ABE pre-defines a few threat techniques, namely SQL, such as a SQL injection attack, PKI, such as spoofing a PKI certificate, and email, including phishing, social engineering, and malicious code attachments.

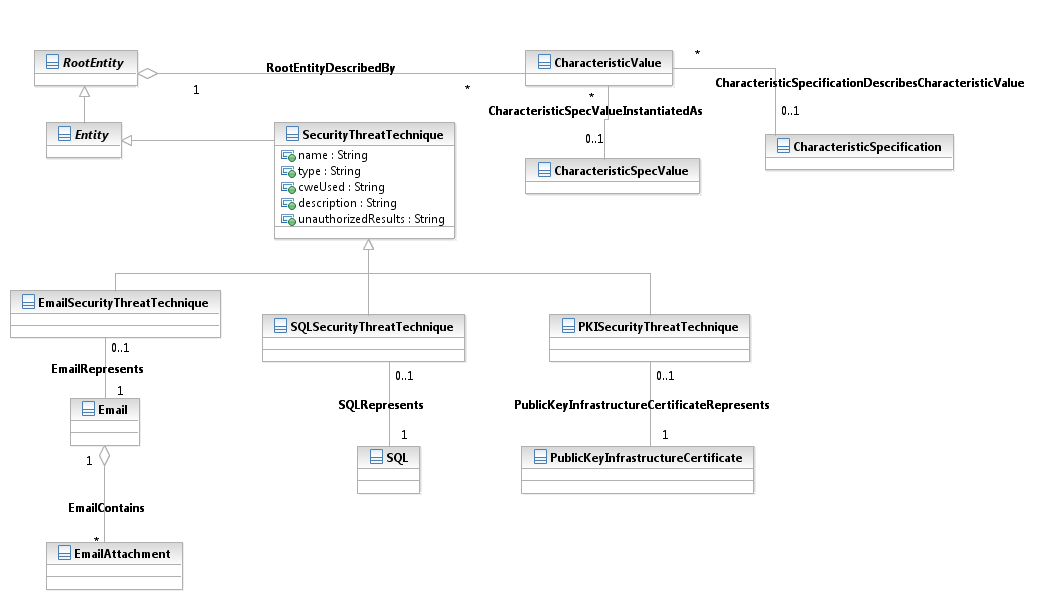


Figure SE.09 – Security Threat Technique

### Security Threat Technique Specification

The previous section delineated a few threat techniques that are far from an exhaustive list. Additional threat techniques can be defined by extending the model with new classes that inherit from SecurityThreatTechnique, or new techniques can be dynamically created using the entity and characteristic specification classes.

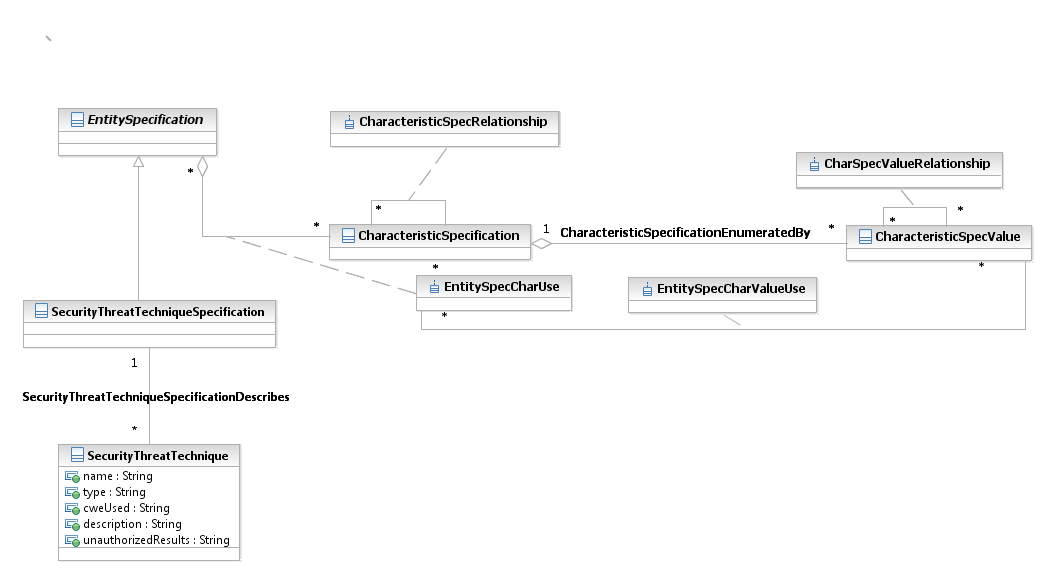


Figure SE.10 – Security Threat Technique Specification

### Security Threat Tool

A threat actor may use organized and distributed software to attack an IT resource or service. The current model only defines three types of threat tools as examples. Threat tools include viruses, botnets, and other software that enables or automates a threat actor’s attack. Currently in the model, a tor is a network of virtual tunnels that is used to anonymously perform a cyber-attack, a peer-to-peer file sharing application is used to distribute malicious code as part of a cyber-attack, and a botnet is used as an attacker.



Figure SE.11 – Security Threat Tool

### Security Threat Tool Specification

Similar to the Security Threat Techniques, the three tools defined in the model are only illustrative. The Security Threat Tool Specification allows dynamic creation of new threat tools.

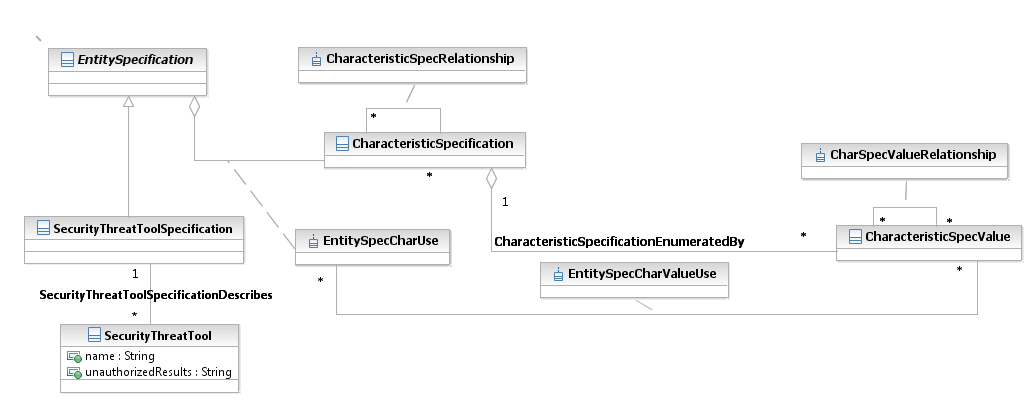


Figure SE.12– Security Threat Tool Specification

### Security Threat Actor

A Security Threat Actor is an individual or organization that has motive and means to perform a cyber attack. The Security Threat Actor will use techniques and tools to exploit vulnerabilities on IT resources and services.

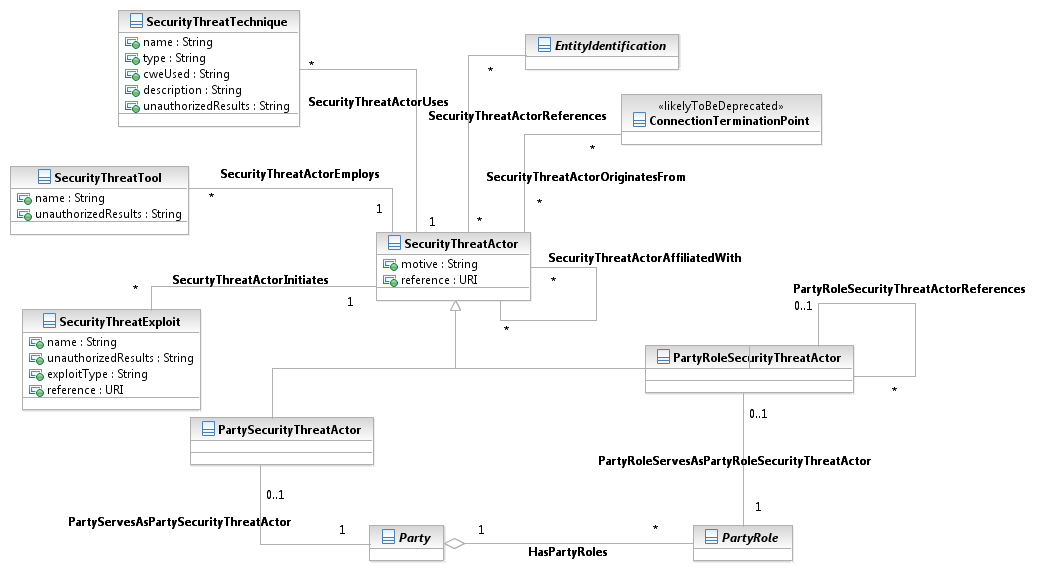


Figure SE.13 – Security Threat Actor

### Security Threat Indicator

A Security Threat Indicator is activity that by itself isn't harmful, but indicates malicious activity (e.g., a DNS call to resolve a known name known to be bad). Threat indicators are activity a security analyst monitors for to detect attacks against its IT infrastructure.

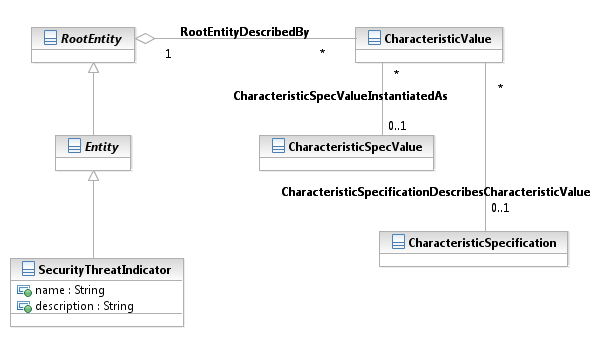


Figure SE.14 – Security Threat Indicator

### Security Threat Indicator Specification

The Security Threat Indicator Specification class enables dynamic definition of characteristics of indicators.

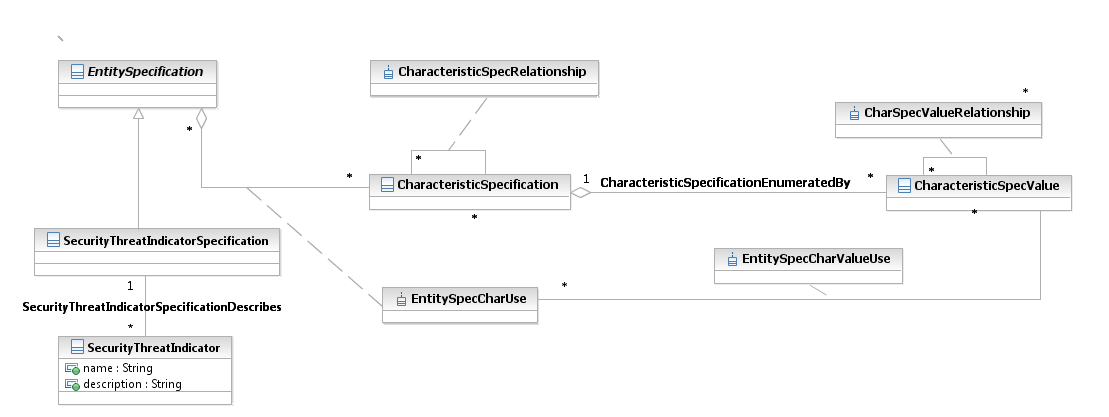


Figure SE.15 – Security Threat Indicator Specification

## Security Vulnerability

The vulnerability portion of the model is designed to carry information about potential weaknesses in hardware, applications, and operating systems that may be exploited by an adversary or threat. The original contribution was designed around the NIST National Vulnerability Database (NVD) and tools.

At the highest level, vulnerability can be described as a software flaw, which would be identified using a Common Vulnerability Enumeration (CVE) identifier or a weak configuration, which would be identified with a Common Configuration Enumeration (CCE) identifier. Information directly included in the Vulnerability core class includes a summary, the date and time it was discovered, the date and time it was disclosed, the data an exploit was published against it, the date and time the vulnerability was published, and the level of access (security protection - root, user, or other) that can be obtained by exploiting the vulnerability.

#### Fix Action

One or more “fix” actions can be applied to vulnerability to remedy the exposure. The fix action techniques or tools must be identified.

Some number of fix actions exists for each vulnerability. Fix actions may consist of updating software, applying patches, implementing administrative policies, other external mitigation, or changing the configuration of the vulnerable asset. Fix actions are typically described as remediation or mitigations. The model allows for these titles to be used; however, where there is any ambiguity, the choice is also provided to label the fix action as a complete or partial fix. Fix actions may contain multiple steps, so recursive fix action steps can provided. Fix actions will also reference the configurations they are target to, the configuration of the tool that is required to apply a fix action, patches they apply, and the specific check to verify successful completion of each fix action.

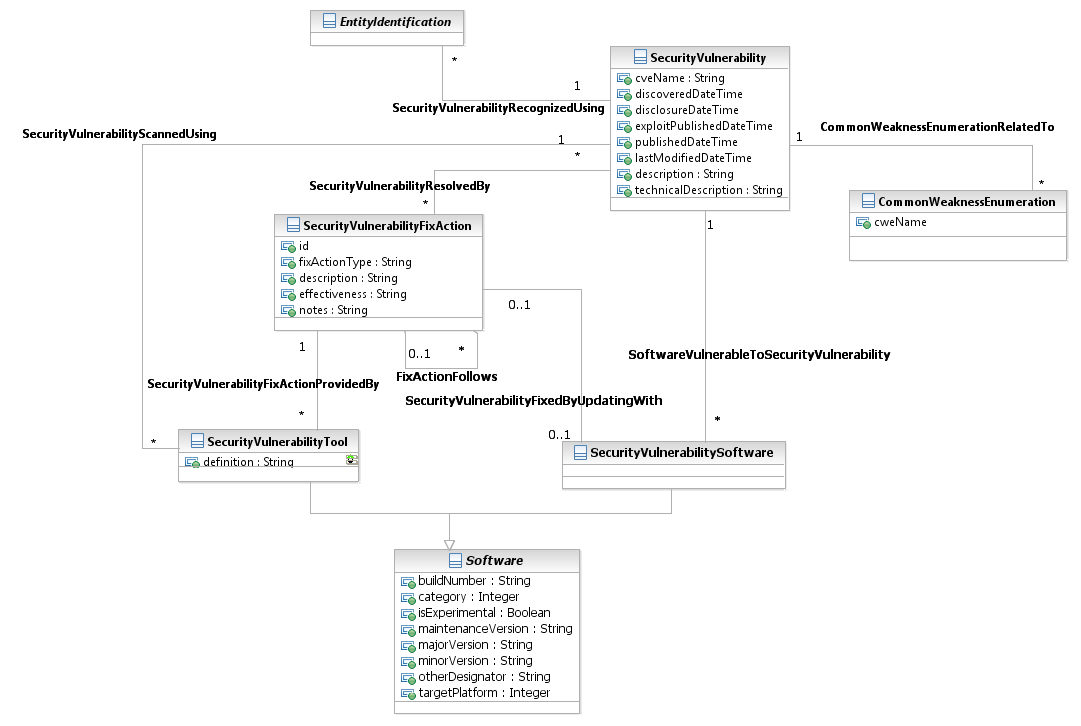


Figure SE.16 – Security Vulnerability

### Security Vulnerability Association with Other Business Entities

Security Vulnerability Association with Other Business Entities show associations between vulnerability entities and other entities in the security model.  The associations depict the associations of vulnerability entities with the SecurityEntity.  A SecurityEntity exhibits one or more SecurityVulnerability that may be taken advantage of by SecurityThreats.  A SecurityVulnerabilityThreatAction applies not only to the SecurityVulnerabilityFixAction (association not shown in the figure) but also to the SecurityEnitty that exhibits the SecurityVulnerability.

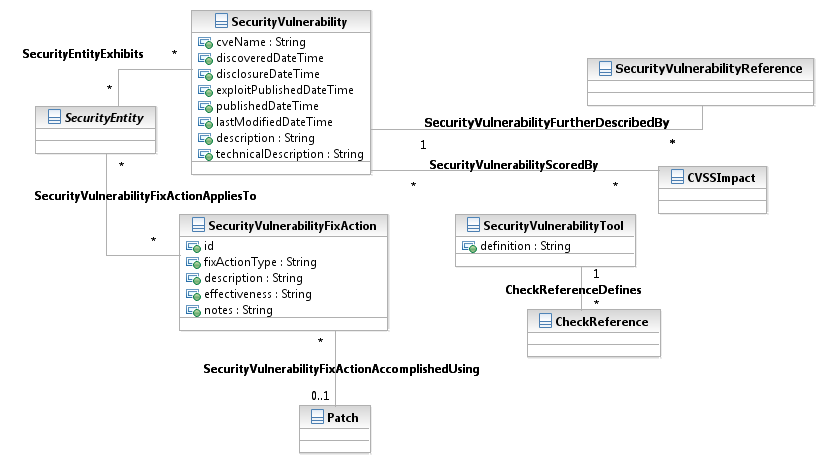


Figure SE.17 – Security Vulnerability Association with Other Business Entities

### Vulnerability Scoring

It is important to be able to assign risk to vulnerabilities, so that decisions regarding remediation can be made. One way to do this is to “score” a vulnerability using a pre-defined algorithm with some number of inputs that are either fixed or calculated. With the Frameworx 12.5 release, the Information Model (SID) has been updated to support scoring. At a minimum, the model supports NIST’s Common Configuration Scoring System (CCSS), the Common Vulnerability Scoring System (CVSS), and the Common Misuse Scoring System (CMSS), but it is expected that the generalized nature of this model makes it a contender for other types of scoring.

The next five figures illustrate the scoring model. Data samples have been included to make the model more understandable. If you would like more information on the specifics of configuration, vulnerability, or misuse scoring, consult the respective NIST documents.

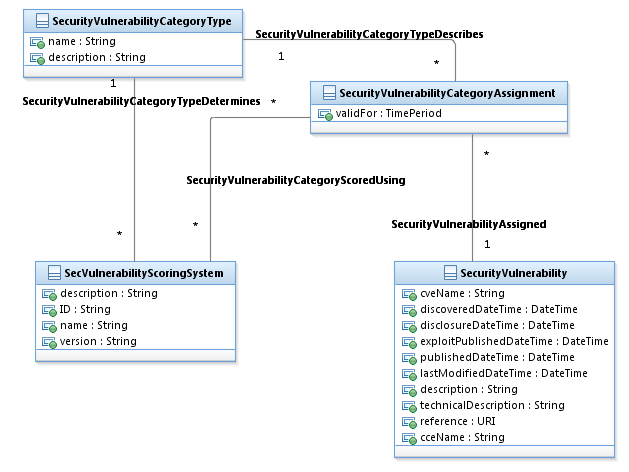


Figure SE.17a – Security Vulnerability Category Types

The Security Vulnerability Category Types figure (SE.17a) shows the relationships between a Security Vulnerabilities and the Scoring System based on the classification of the Vulnerability (software flaw, software feature misuse, or software configuration issue).

Some example data for this figure includes:

|  |  |
| --- | --- |
| Security Vulnerability | An instance of a Security Vulnerability such as an “Office 10” bug. |
| Security Vulnerability Category Assignment | Using our Office 10 bug example, the category assigned to this vulnerability would be “software flaw”. |
| Security Vulnerability Category Type | This is the standard set of vulnerability categories. NIST defines them as software flaw, software feature misuse, and software configuration issue. |
| Security Vulnerability Scoring System | NIST defines three types of scoring systems: Common Configuration Scoring System (CCSS), Common Vulnerability Scoring System (CVSS), and Common Misuse Scoring System (CMSS). In our example, the software flaw would be “scored” using the CVSS. |

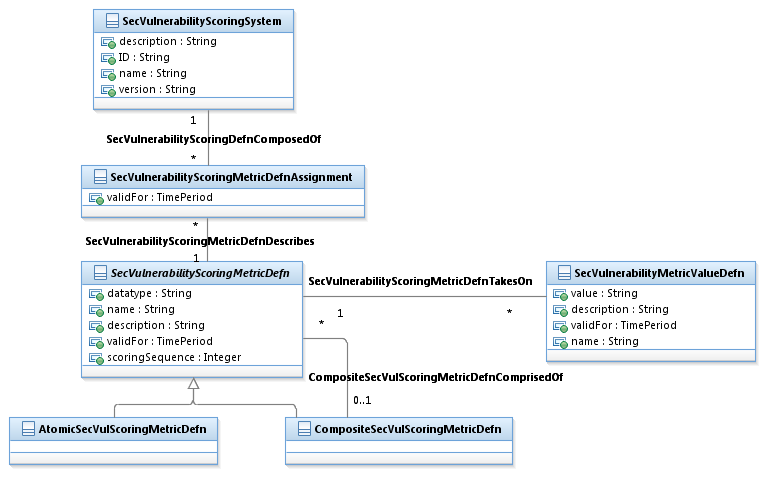


Figure SE.17b – Security Vulnerability Scoring Definitions

Figure SE.17b - Security Vulnerability Scoring Definition) shows the classes and relationships that define the metrics associated with a particular scoring system.

Some example data for this figure includes:

|  |  |
| --- | --- |
| Security Vulnerability Scoring System | NIST defines three types of scoring systems: Common Configuration Scoring System (CCSS), Common Vulnerability Scoring System (CVSS), and Common Misuse Scoring System (CMSS). In our example, the software flaw would be “scored” using the CVSS. |
| Security Vulnerability Scoring Metric Definition Assignment | Each of the three scoring systems, CCSS, CVSS, and CMSS have three types of metrics: Base, Temporal, and Environmental. |
| Composite Security Vulnerability Scoring Metric Definition | Often the Scoring Metrics are “composed” of other metrics. As an example, the Base Metrics are made up of Base Impact and Base Exploitability Metrics which are further decomposed into Confidentiality Impact, Integrity Impact, Availability Impact, Access Vector, Authentication, and Access Complexity (respectively). Using the atomic/composite pattern facilitates the breaking down metric definitions to the smallest unit, so that values can be assigned. |
| Security Vulnerability Metric Value Definition | As an example, standard values for Access Vector (CCSS) are A – Adjacent Network, L – Local, and N – Network. |

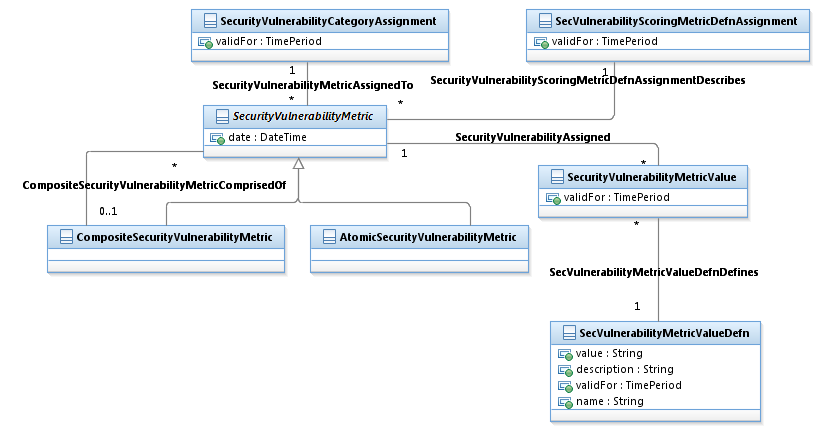


Figure SE.17c – Security Vulnerability Metrics

Figure SE.17c - Security Vulnerability Metrics identifies, or assigns, metrics to a particular vulnerability.

If we continue our Office 10 bug example, data for this figure might include:

|  |  |
| --- | --- |
| Security Vulnerability Category Assignment | Office 10 – Software Flaw |
| Security Vulnerability Metric Atomic | For a Software Flaw, Access Vector and Authentication are examples of two atomic metrics. |
| Security Vulnerability Metric Value | Assigns the value to a Vulnerability Metric. In our example, Access Vector could have a value of L – local and Authentication could have a value S (single). |



Figure SE.17d – Security Vulnerability Score

Figure SE.17d - Security Vulnerability Score shows the entities used to calculate the score for a particular vulnerability. It uses the algorithm assigned to the particular vulnerability along with the values assigned to each of the metrics, and calculates the score. (See SE. 17e to better understand how the classes and relationships associated with building the algorithm.)

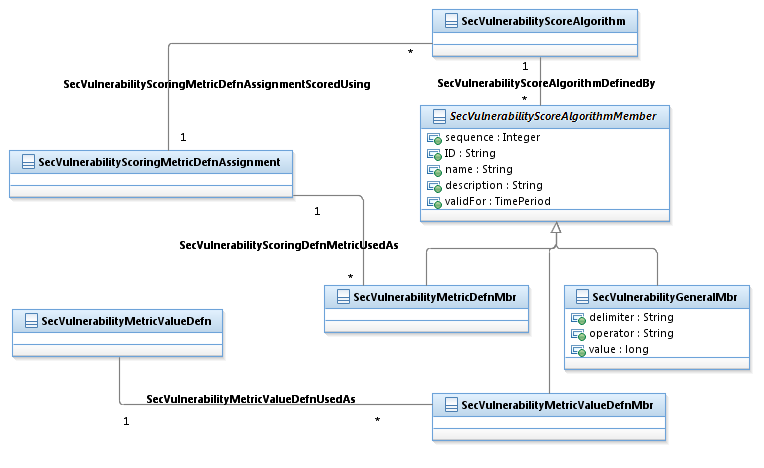


Figure SE.17e – Security Vulnerability Score Algorithm

Figure SE.17c - Security Vulnerability Scoring Algorithm identifies, or assigns, values, delimiters, and operators to metrics for the purposes of calculating a score for a particular vulnerability.

If we continue our Office 10 bug example, data for this figure might include:

|  |  |
| --- | --- |
| Security Vulnerability Scoring Algorithm | There is a Base Equation for the CCSS which is made up of several calculations. One example is: Exploitability = 20 \* AccessVector \* Authentication \* AccessComplexity. |
| Security Vulnerability Scoring Algorithm Member | Provides the ability to break down the individual elements of the algorithm into values, delimiters, and operators. It also recognizes the need to “order” these elements within the equation. So, for the equation above:   1. 20 \* 2. Access Vector (AV) for S (.646) 3. \* 4. Authentication (AU) for S (.45) 5. \* 6. Access Complexity (AC) for S (.35) |

## Security Incident

Security Incidents often have an assessment performed to determine their root cause. A lifecycle is associated with an incident, and provides both the reported and detected time and the associated interval (time period). Other important information is who is taking action, what actions are and have been taken, who is affected, and whether the incident is an exercise or real. Trouble tickets can be attached to Security Incidents.

In order to perform incident analysis and trending, it is important to capture how the incident happened, what threat actor caused it, and how “bad” the damage is.

The underlying event data, often captured by sensors/observers, are provided for additional analysis.

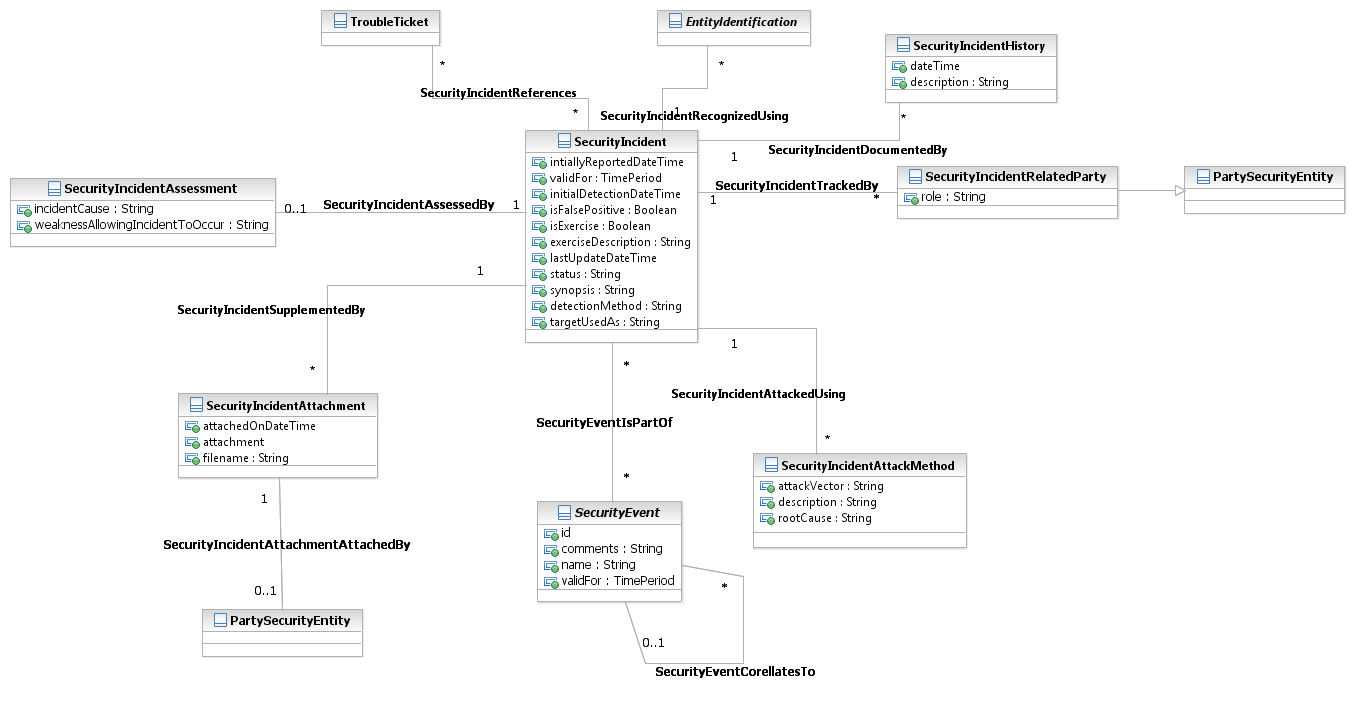


Figure SE.18 – Security Incident

### Security Incident Assessment

Assessment identifies the extent of the incident’s impact. (How “bad” is the damage?) What is the cost (financial impact) in terms of time and/or money? Was there a technical impact or an operational impact associated with the incident?

By understanding the full impact of the security incident, an appropriate response can be determined, and recovery initiated.

Also, assessment identifies the actor “intruder” – who initiated the security incident.

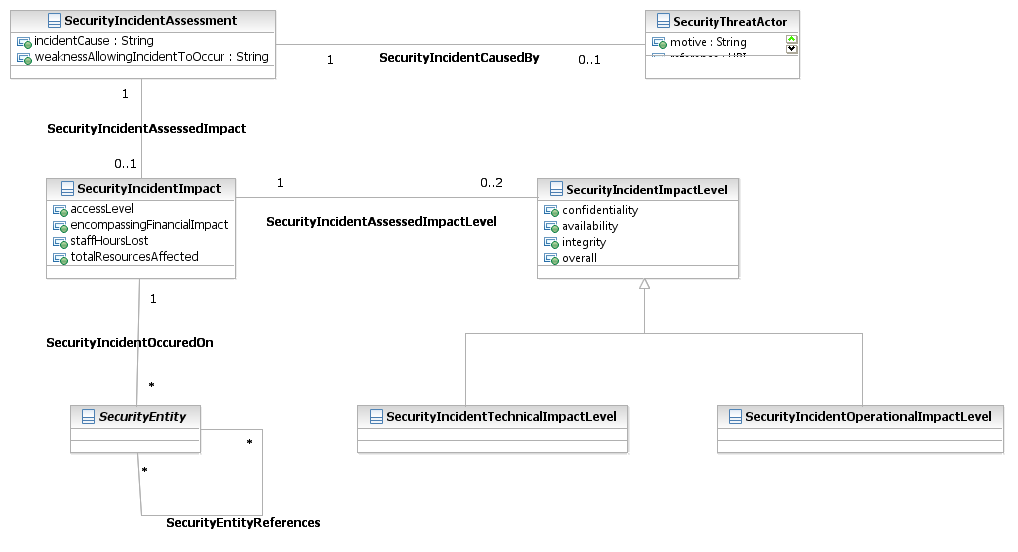


Figure SE.19 – Security Incident Assessment

## Business Entity Definitions – Security

Entity and attribute definitions can be found in all available Information Framework (SID) model formats. The next release of the SID will move all entity and attribute definitions from domain/ABE addenda to a single addendum containing all definitions. The definitions will be generated from the SID UML model. All model formats will continue to contain definitions.

# Administrative Appendix

This Appendix provides additional background material about the TM Forum and this document. In general, sections may be included or omitted as desired, however a Document History must always be included.

## About this document

This is a TM Forum Guidebook. The guidebook format is used when:

* The document lays out a ‘core’ part of TM Forum’s approach to automating business processes. Such guidebooks would include the Telecom Operations Map and the Technology Integration Map, but not the detailed specifications that are developed in support of the approach.
* Information about TM Forum policy, or goals or programs is provided, such as the Strategic Plan or Operating Plan.
* Information about the marketplace is provided, as in the report on the size of the OSS market.

## Document History

### Version History

|  |  |  |  |
| --- | --- | --- | --- |
| **Version Number** | **Date Modified** | **Modified by:** | **Description of changes** |
| 1.0 | 25/OCT/10 | C. Coffey | Initial draft. |
| 1.1 | 2/Dec/10 | 1. Morgan | Added improved diagrams. |
| 1.2 | 24/Mar/11 | Alicja Kawecki | Minor formatting corrections prior to web posting and ME |
| 1.3 | 6/Sep/11 | John Reilly | Corrected Figure SE.19. |
| 1.4 | 12/Sep/12 | J. Reilly, C. Coffey, S. Schreiner | Added Vulnerability Scoring |
| 1.5 | 26/Oct/12 | Alicja Kawecki | Minor style/cosmetic corrections prior to web posting and Member Evaluation |

### Release History

|  |  |  |  |
| --- | --- | --- | --- |
| **Release Number** | **Date Modified** | **Modified by:** | **Description of changes** |
| 1.0 | 27/Dec/10 | John Reilly | First issue of document |
| 1.0 | 6/Sep/11 | John Reilly | Corrected Figure SE.19 |
| 1.0 | 21/Sep/12 | J. Reilly, C. Coffey, S. Schreiner | Added Vulnerability Scoring |

## Company Contact Details

The following individuals led the project or actively participated.

|  |  |
| --- | --- |
| **Company** | **Team Member Representative** |
| MITRE | *Adam S. Morgan*  *Ryan Megredy*  *Susan Schreiner (Frameworx 12.5)* |
| AT&T | *Mark Arvidson* |
| TM Forum | *John Reilly*  [*jreilly@tmforum.org*](mailto:jreilly@tmforum.org)  *Christy Coffey*  [*ccoffey@tmforum.org*](mailto:ccoffey@tmforum.org) |
| Booz Allen Hamilton | *Larry Frank (Frameworx 12.5)* |

## Acknowledgments

This document was prepared by the members of the TM Forum Security Management / Network Defense project team.

We would like to recognize the many companies who participated in the development of the NSA’s “Network Defense Data Model, An Overview and Implementation Guide”, which was contributed to the TM Forum.

We would like to recognize the companies and individuals who participated in our Security Management workshop on June 14th, 2010:

* + - *DISA:* Dave Nameroff, Danny Phillips, Bilal Alam, Ed Hanley, Guy Seebaluck, Daniel Sullivan (Raytheon), Steve Brumbaugh (Technica)
    - *MITRE:* Adam Morgan, Bob Natale, Susan Schreiner
    - *BAH:* Larry Frank, James Nixon, Tieu Luu
    - *AT&T:* Mark Arvidson
    - *Layer 7:* Adam Vincent
    - *Telcordia:* Ron Roman, Andy Mayer
    - *TM Forum:* John Reilly, Christy Coffey

Their subject matter expertise helped finalize the initial set of entities and relationships for the SID.

Lastly, we would like to recognize the SID team leads for serving as mentors to the Security Management project team.

1. <http://csrc.nist.gov/publications/nistir/ir7502/nistir-7502_CCSS.pdf> [↑](#footnote-ref-1)
2. <http://csrc.nist.gov/publications/nistir/ir7435/NISTIR-7435.pdf> [↑](#footnote-ref-2)
3. <http://csrc.nist.gov/publications/drafts/nistir-7517/Draft-NISTIR-7517.pdf> [↑](#footnote-ref-3)