IDEA:
Classification of security events, their participants and detection probes

PAVEL KÁCHA
CESNET-CERTS Computer Security Incident Response Team
CESNET
Zikova 4, Prague
CZECH REPUBLIC
ph@cesnet.cz

Abstract: For IDEA (Intrusion Detection Extensible Alert) format to be really usable for security event data exchange, in addition to container and formats also taxonomies for description and classification has to be defined. We thus distil common classification by analysis and mutual mapping of number of existing taxonomies (creating translation between them on the way), and by identifying omissions, unsuitable semantics, unusual or too specific cases, and adding information conveyed in various types of real life security events, we also populate auxiliary dictionaries – classification of sources and destinations of attacks and description tags of detection probes. IDEA security event description may thus serve as simple to create and easy to understand form, onto which most of the existing automatically gained security information can be mapped.

Key-Words: alert, security event, incident response, taxonomy, classification, ids, honeypot, json

1 Introduction

IDEA is an attempt to address deficiencies in automated incident report exchange. We have already defined the container for security event data in [1]. Definition of container is just one part of the job, similarly important for allowing interoperability is also definition of dictionaries for classification of various types of data, and mapping of real world data onto the IDEA format in the sane, usable way.

None of the formats for incident report exchange tried to define or incorporate any kind of dictionary or taxonomy, fact partially stemming from the lack of widely accepted one. As we have tried to create modern incident container, we do not want to sidestep the issue and take the responsibility to define its basic classifications.

Creating any taxonomy, and security incident one in particular, is not a simple task. Users are driven by various needs and as expectations clash, CSIRT teams are ending up creating their own incident classifications for internal use. However, as need for more automated incident report exchange rises and tools for machine based security event dissemination emerge, usefulness of common ground at least for mapping other classifications to, becomes apparent.

Designing of security taxonomies is usually attempt to find following compromises.

1.1 Low level vs high level

Taxonomy may attempt to describe precise details of incident, as in venerable Howard/Longstaff [2] taxonomy. The set of incident aspects and impacts is then well defined, however higher level, widely understood modus operandi (for example that incident is phishing page) is not readily obvious.

On the other hand, too vague incident types might hide important details of impact (for example – does "phishing" mean phishing spam or phishing web page? Or both?).

1.2 Action vs modus operandi

Incidents range from purely technical actions (connection attempt, scan) to intricate scenarios (spear phishing, social engineering), thus taxonomies have to cope with wide nature of incident complexity.

1.3 Exhaustive vs transparent

On the one side, incident can be classified very precisely, as for example in CAPEC [3] enumeration. However this kind of detail is usually too much of a burden to use in common scenarios. On the other side, some taxonomies use very coarse distribution, based on simplicity and ease of use (for example).

For quick response security team cannot search extensive dictionary to find out meaning of very specific category. Examples of these are FICORA and CESNET taxonomies.

Incident taxonomy is usually used for classification during incident exchange and for statistical purposes. Most common statistic use case are reports and trend graphs of the most usual types of attacks, which do not need overly detailed division. Also, during incident exchange, basic incident description is usually accompanied with more detailed information if available – so there still remains possibility to use other more exhaustive specification or description of the event.

1.4 Rigid vs extensible

Taxonomies are usually rigid, rarely changed, causing their ageing and not being able to keep up with new types of incidents (as in Howard/Longstaff). Common ground taxonomy thus should not be static, but allow some form of extensions – be it by its authors, or by allowing side-stepping existing categories in case new incident type does not fit into predefined scenarios.

Also, sometimes one category is not enough, incidents may span more than one categories. For example security event, describing phishing email might get labelled as phishing and also as spam, because informed systems may choose to deal with incident as spam (add mail source to blacklist, learn Bayes database and so on) or specifically as phishing (add phishing URL to blacklist, inform human operator), whereas in case phishing web page gets discovered, another scenario may arise (dealing with defaced web page or poisoned DNS).

2 Existing taxonomies

There already exists a number of taxonomies, however, comparing to nowadays expectations, each of them is in various ways incomplete, outdated, or oriented to too narrow niche. Number of security teams created local taxonomies for addressing their specific needs, various security data management projects or security event detectors have their own classifications, based on their specific types of function, and of course, the real world can come up with not quite fitting security event types.

2.1 CSIRT origin

eCSIRT.net taxonomy [4] is one of the most practical takes is The European CSIRT Network Incident Classification, which is in turn based on Telia CERTCC work of Jimmi Arvidsson. Classification uses two levels, incident class and incident type. Classes are coarse grained groups, stemming from common usage, such as “Abusive Content”, “Intrusions” or “Fraud”, whereas types are more fine grained subclasses, such as “Spam”, “Worm”, or “DDoS”. The structure is very practical, however taxonomy shows its age in some missing, but nowadays common security event types.

Don Stikvoort from SURFnet have attempted to revive and modernize this taxonomy as *eCSIRT.net MkII* [5]. Several missing categories are added, like non malicious events, botnet related events, and vulnerability information. We will take it as a basis for extensions and mapping.

Venerable low level take on security event classification by *Howard/Longstaff* [2] is based on splitting number of event facets (“Attacker”, “Tool”, “Vulnerability”, “Action”, “Target”, “Unauthorized result”, “Objectives”, related to the timeline of the incident. While this attempt describes event in a great detail for recipient, it makes a great burden for sender/creator to deduce and correctly assign these facets. Also, classification shows its age and some nowadays common incident types are incorporated.

At NCSC-NL International Converence 2010 Tom Longstaff presented updated model [6], which takes into consideration monetary and social information incentives of today.

Several other taxonomies are in the wild, defined by various CSIRT teams for mostly internal or specific purposes.

Finnish Communications Regulatory Authority National Cyber Security Centre *FICORA* uses its own categorization at incident submission form [7].

SURF collaborative ICT organisation for Dutch higher education and research CERT team – *SURFcert* also uses its own [5].

We in *CESNET-CERTS* security incident response team also have generalized categorization [8], stemming mainly from need to have at least coarse overview of incident type distribution and trends.

Andrew Cormack from Terena have also tried to unify taxonomies already used [9].

2.2 Software origin

Various incident exchange and detection projects have also taken shot at defining their own dictionaries.

Collective Intelligence Framework is project for gathering event data from various sources for identification, detection and mitigation (usually blacklists). It uses specific categorization for its information feeds – *CIF API Feed Types v1* [10] and for its assessments – *CIF Taxonomy Assessment v1* [11].

Warden [12] is a system for sharing information about detected events, developed in CESNET. Given the types of information it supports, its classification is particularly terse [13].

We have included also *HP TippingPoint Event Taxonomy V2.2* – HP flagship intrusion detection and prevention system [14].

Working only with existing taxonomies would be great neglect of the real world. In the analysis and mapping we have taken into consideration types of real life events from the database of *CESNET Mentat* event gathering and correlation system.

2.3 Exhaustive repository

Common Attack Pattern Enumeration and Classification [3] is knowledge resource database of attack mechanisms and modes operandi. It is listed here for completeness, however it was not included into mapping – it makes a great encyclopedic resource, however its vast scope and detail makes it infeasible for operative security event classification.

3 Incident Classification Mapping

For IDEA event taxonomy we have created extensive cross reference mapping between previously mentioned incident classifications, available at IDEA web page for "Incident Classification Comparison" [15]. Apart from forming basis for analysis, as a side effect mapping can be readily used by security teams as a translation for communication with other parties. However, mapping is too large and extensive to be included in this paper, so we discuss and compare here only identified omissions from eCSIRT.net taxonomies, which turned out to be the most exhaustive (excluding CAPEC).

During creation we consulted also previous analyses on security incident categorization, namely [16] and [17].

In the mapping, the Corresponding incident type groups are clustered together where possible, and identified uncovered parts of taxonomies are left greyed out – or marked as catch-all category (other, unknown or similar), if particular taxonomy uses one.

If one category occupies more than one line, it means that it doesn't have counterpart in some other taxonomy.

Following chapters are discussion of this mapping and how we have come up with the result, suitable for IDEA.

4 Discussion and IDEA examples

Now we are going to find parts missing or clashing among taxonomies, and try to define way to map it onto IDEA, possibly modifying MkII, to get model, which is able to convey the meaning security event for both machine and human.

Along with reasoning, examples of IDEA messages, describing related event are inserted, to verify feasibility of the result. All the messages are anonymised and stripped to bare minimum, but still perfectly valid.

The *Table 1* summarises all the properties found in various taxonomies, which have not been found in eCSIRT.net/MkII (and also all the others, which is the reason, why these are not included in the table).

	Lists	Anomalies	Bounce	Scans	Vulnerabilities	Botnets	Phishing	Suspicious	Searches	Local
CIF API	x					x				
CIF Assesment	x					x			x	
CESNET			x				x			
Warden 2				x						
TippingPoint	x	x		x	x					x
Mentat				x	x	x		x		

Table 1: Differences between taxonomies

The detailed analysis follows.

4.1 Blacklists, whitelists

Information about being put into blacklist/whitelist is quite commonly communicated information – one is not able to process all and every blacklist/whitelist on the wild, moreover various lists and databases pop up and disappear frequently. People often rely on getting this information from third party sources, aggregators, etc.

Whitelists are either lists of addresses, knowingly clean in some particular aspect, or attempts to monetize on impression of legitimacy of certain company's email or internet assets (DNSWL), or site/vendor/organization specific exception lists, not relevant to security event dissemination.

Blacklists specifically important to security teams are those, which inform about vulnerabilities and specific security problems – lists of WWW pages, injected with phishing or malware, open relay mailservers, open recursive resolvers, etc.

In incident handling process, these are usually communicated in the same way as locally found vulnerabilities, with additional specifics accompanying the message.

These events can be in MkII represented as basic "Vulnerability", and if used at incident message, by additional labelling specific to transport protocol and/or format and/or concerned parties needs – I believe narrow categories akin to Phishing WWW, Malware WWW, Open Relay Mailserver are out of scope of such a general categorization.

Found in:

- CIF API Feed Types v1: infrastructure/whitelist, domain/whitelist, email/whitelist, url/whitelist
- CIF Taxonomy Assessment v 1: Whitelist
- HP Tipping Point: IP Filters/Deny, Accept

IDEA representation:

Note the marking of source node with "Open" tag.

```
{
  "Format": "IDEA0",
  "ID": "c34bf422-931c-4535-9c6b-257128185265",
  "DetectTime": "2014-11-03T10:33:12Z",
  "Category": ["Vulnerable.Open"],
  "Confidence": 0.5,
  "Description": "Open Recursive Resolver",
  "Source": [
    {
      "Type": ["Open"],
      "IP4": ["93.184.216.119"],
      "Proto": ["udp", "domain"]
    }
  ]
}
```

4.2 Anomalies

Anomalies, such as excessive traffic, might later be identified as security problem (for example DoS or DDoS), however they might end up as accidental peak or outage, or completely innocent. As anomalies can be important to security teams as indicator of possible attack, or as a correlation element in investigation, I think these should be taken into account in security events transfer. I see two possibilities to represent them:

- specific top level category, for example *Anomaly*, with suitable subcategories, I'd suggest *Traffic*, *Connection*, *Protocol*, *System*, *Application*, *Behaviour*
- when anomaly arises, we usually have suspicion, which types of incidents can it cause (excess traffic → DOS, overlaid TCP packets → exploit, too many connections → dictionary attack, etc.). So there is possibility to use these deduced categories, but for incident handling we might allow another dimension – certainty of detection (or self trust). However, that requires support from underlying transport format.

Both of these approaches have its use, first is usable, when we are not able to connect possible situation with any type of attack, whereas second describes situation, which could potentially evolve into real threat, or get recognized as such by closer analysis.

Found in:

- HP Tipping Point: Traffic Thresholds, Application or Protocol Anomaly

IDEA representation in Anomaly category:

"Anomaly" category used for distinction from real attack or DDOS.

```
{
  "Format": "IDEA0",
  "ID": "4390fc3f-c753-4a3e-bc83-1b44f24baf75",
  "DetectTime": "2014-02-01T18:32:03Z",
  "Category": ["Anomaly.Traffic"],
  "ConnCount": 3352,
  "Description": "Possible DoS",
  "Source": [
    {
      "IP4": ["93.184.216.119"],
      "Proto": ["tcp"]
    }
  ]
}
```

IDEA representation as suspicion:

"DoS" category used, however confidence indicates that we are not completely sure.


```
{
  "Format": "IDEA0",
  "ID": "0bd857b6-7c4d-4a17-ad1a-bcb1cc8eaa6b",
  "DetectTime": "2014-02-01T19:28:23Z",
  "Category": ["Availability.DoS"],
  "ConnCount": 3352,
  "Confidence": 0.5,
  "Description": "Possible DoS",
  "Source": [
    {
      "IP4": ["93.184.216.119"],
      "Proto": ["tcp"]
    }
  ]
}
```


4.3 Backscatter/Bounce

Bounce is distinct flavour of spam – DSN messages generated by servers in reaction to non deliverable spam messages with forged sender, thus sent to innocent forged recipients. That might validate another category. However mechanism of backscatter – forging sender data – is more general and abused also in DDOS attacks, like DNS amplification or various other types of UDP reflection attacks, which might indicate that this information should be represented or communicated differently/orthogonally, possibly as facet of the source.

Found in:

- CESNET CERTS: Bounce

IDEA representation:

Note the *"Backscater"* tag of the source.


```
{
  "Format": "IDEA0",
  "ID": "bf8344d7-a0da-4724-92da-ccda382d7e72",
  "DetectTime": "2014-01-03T01:23:42Z",
  "Category": ["Abusive.Spam"],
  "Description": "Spam bounce",
  "Source": [
    {
      "Type": ["Spam", "Backscatter"],
      "IP4": ["93.184.216.119"],
      "Proto": ["tcp", "smtp"]
    }
  ]
}
```


4.4 Scans

Number of existing taxonomies distinguish between specific types of IP based reconnaissance, the basic observed types being host scan, port scan, service scan, application scan, port sweep, ICMP probe. This again denotes technical facet of the attack, which can be communicated by some other means – in security event description formats for example by type of network and application protocol used, and number of ports and machines scanned.

Some taxonomies also differentiate events based just on cardinality of attack – singular events might get marked akin to "connection attempt". In fact there is no way to be sure, whether singular events are part of greater reconnaissance or not, without additional information usually from other sources. Most important information, which this distinction conveys, is the severity of the attack, and that's also orthogonal information, which should get communicated by other ways.

Found in:

- HP Tipping Point: Reconnaissance or Suspicious Access
- Warden 2: Portscan, Probe
- Mentat: Probe, Portscan, Connection attempt, Ping probe, SYN/ACK scan or DOS attack

IDEA representation:


```
{
  "Format": "IDEA0",
  "ID": "3ad275e3-559a-45c0-8299-6807148ce157",
  "DetectTime": "2014-03-22T10:12:56Z",
  "Category": ["Recon.Scanning"],
  "ConnCount": 633,
  "Description": "Ping scan",
  "Source": [
    {
      "IP4": ["93.184.216.119"],
      "Proto": ["icmp"]
    }
  ]
}
```


4.5 Vulnerabilities

Various event detectors are also able to deduce attacked application or even name of the exploit used. That however also does not belong into general taxonomy, as this usually goes along as additional info – and there is number of well known databases of vulnerabilities, which can be used.

Found in:

- Mentat: EPMAPPER exploitation attempt, SMB exploitation attempt, SQL query attempt, URL attack attempt, Webattack, Open recursive resolver
- HP Tipping Point: Vulnerability

IDEA representation:

Note *"Ref"* link to public CVE identifier.


```
{
  "Format": "IDEA0",
  "ID": "3ad275e3-559a-45c0-8299-6807148ce157",
  "DetectTime": "2014-03-22T10:12:31Z",
  "Category": ["Recon.Scanning"],
  "ConnCount": 633,
  "Description": "EPMAPPER exploitation attempt",
  "Ref": ["cve:CVE-2003-0605"],
  "Source": [
    {
      "IP4": ["93.184.216.119"],
      "Proto": ["tcp", "epmap"],
      "Port": [24508]
    }
  ]
}
```


4.6 Botnets

Botnets are one of the most common threats today. Taxonomies sometimes differentiate at least between C&C servers and worker drones, because bringing down C&C is of higher benefit than cleaning up infected drons. Importance of this information might validate adding new category, however it's again more of a technical facet. When integrating taxonomy into security event format, this information should not be omitted, at least as severity of the incident, or as a property of attack source, also with indication of fastflux possibility.

Found in:

- CIF API Feed Types v1: infrastructure/botnet, url/botnet, domain/botnet, infrastructure/fastflux, domain/fastflux
- CIF Taxonomy Assessment v1: Botnet, Fastflux
- Mentat: Botnet Drone, Botnet Proxy, Botnet_c_c

IDEA representation:

Note the *"Type"*, denoting C&C server on fast-flux domain name.


```
{
  "Format": "IDEA0",
  "ID": "cca3325c-a989-4f8c-998f-5b0e971f6ef0",
  "DetectTime": "2014-03-05T15:52:22Z",
  "Category": ["Intrusion.Botnet"],
  "Description": "Botnet Command and Control",
  "Source": [
    {
      "Type": ["Botnet", "CC",  "FastFlux"],
      "Hostname": ["evil.example.com"],
      "Proto": ["tcp", "ircu"],
      "Port": [6667]
    }
  ]
}
```


4.7 Phishing/Pharming/Scam

At least one examined taxonomy distinguishes between phishing and pharming – that's also technicality, which should be identifiable from accompanying information (cache poisoning, DNS break-in, etc.).

However, well known type of incidents are variation on Nigerian 419 scam. That might fit into "Abusive Content/Spam" category, but that does not tell the whole story – it's not *just* spam. It might also fit into "Fraud/Masquerade" category, but that depends on what designers of eCSIRT.net taxonomy exactly mean by "masquerade" – whether posturing as specific person (identity theft), or general con (variation of social engineering). I suggest adding "Fraud/Scam" category for clarity.

Found in:

- CESNET CERTS: Phishing, Pharming, Scam

IDEA representation:


```
{
  "Format": "IDEA0",
  "ID": "9729ea4a-a260-40c0-8e63-0cb0b2687177",
  "DetectTime": "2014-02-22T13:35:03Z",
  "Category": ["Fraud.Scam"],
  "Description": "419 mail scam",
  "Source": [
    {
      "Type": ["Spam"],
      "IP4": ["93.184.216.119"],
      "Proto": ["tcp", "smtp"]
    }
  ]
}
```


4.8 Suspicious

URLs found in spam messages or in sandboxed malware binaries may or may not be necessarily evil. They are definitely suspicious, but spammers and malware creators often incorporate innocent URLs to lure automated tools astray. I am not convinced of the necessity of new specific category, in security event messages this information will go under "Abusive Content/Spam" or "Malicious Code", and extracted URL should be marked as unclear by other means (specific type, reliability).

Found in:

- Mentat: Sandbox URL, Spam URL

IDEA representation:

Note the *"OriginSpam"* tag.

```
{
  "Format": "IDEA0",
  "ID": "4d52640a-5363-497a-a7d9-bcbde759cb7d",
  "DetectTime": "2014-02-21T16:01:32Z",
  "Category": ["Abusive.Spam"],
  "Description": "Spam URL reference",
  "Source": [
    {
      "Type": ["OriginSpam"],
      "URL": ["http://www.example.com/"],
      "Proto": ["tcp", "http", "www"]
    }
  ]
}
```

4.9 Searches

During reconnaissance, attackers often use Google searches ("Google Hacking"), or conduct various suspicious searches against company sites. This activity can be detected, either by Google aimed project (Google Hack Honeypot [18]) or by local IDS systems. This type of information gathering does not precisely fit into any MkII subcategory, I suggest adding "Information Gathering/Searching" category.

Found in:

- CIF Taxonomy Assessment v1: Searches

IDEA representation:

Note the *"Category"*.

```
{
  "Format": "IDEA0",
  "ID": "b7dd112c-9326-49e6-a743-b1dce8b69650",
  "DetectTime": "2014-02-13T02:21:15Z",
  "Category": ["Recon.Searching"],
  "Description": "Suspicious search",
  "Source": [
    {
      "IP4": ["93.184.216.119"],
      "Proto": ["tcp", "http", "www"]
    }
  ],
  "Target": [
    {
      "URL": ["http://www.example.com/search=%20union
%20select%20password%20from%20users%20%2D%2D"]
    }
  ]
}
```

4.10 Local

At least one taxonomy incorporates breaches into company policies. As these can be local specific, they don't belong into general taxonomy. In IDEA, these can be represented by locally defined nodes, as IDEA container is freely extensible.

Found in:

- HP Tipping Point: Security Policy

4.11 Unclassifiable

The situations may arise, where we are aware of wrongdoing, but are not able to classify it by means of existing taxonomy class. There are two possible scenarios:

1. We don't know what exact type of incident that is, and what particular class it belongs to, maybe because we need additional information to find out. We can then use educated guess (and possibly, if channel allows for that, add certainty of that guess), or it might again warrant "Anomaly" category.
2. We know the type of incident and it's completely new one, which does not fit into any of the existing categories. We can either use Other, or at least top level category (if it does fit into one). Or we can aim for extensibility and leave creating of new subcategories on users – and codify them later into standard based on what is experienced in the wild.

5 IDEA implementation

5.1 Security event taxonomy

eCSIRT.net MkII comes out as the most comprehensive, yet still practical solution. From mapping and comparison with other taxonomies and several real world incidents we have implemented following updates:

1. Adding "Anomaly" category, with following subcategories (incident examples): Traffic, Connection, Protocol, System, Application, Behaviour (see 4.11).
2. Add "Scam" incident example into "Fraud" (see 4.7).
3. Add "Searching" incident example into "Information Gathering" (see 4.9).
4. Don't stay rigid, allow side-stepping, make taxonomy extensible by users (see 1.4).
5. Allow multicategorization, where applicable (see 1.4).

Final taxonomy, based on eCSIRT.net, incident classification mapping and its discussion in this paper, comes out as follows:

Abusive
Spam, Harassment, Child, Sexual, Violence
Malware
Virus, Worm, Trojan, Spyware, Dialer, Rootkit
Recon
Scanning, Sniffing, SocialEngineering, Searching
Attempt
Exploit, Login, NewSignature
Intrusion
AdminCompromise, UserCompromise, AppCompromise, Botnet
Availability
DoS, DDoS, Sabotage, Outage
Information
UnauthorizedAccess, UnauthorizedModification, UnauthorizedUsage
Fraud
Copyright, Masquerade, Phishing, Scam
Vulnerable
Open
Anomaly
Traffic, Connection, Protocol, System, Application, Behaviour
Other
Test

(For full description of categories, see IDEA Classifications and Enumerations page [15].)

Basic taxonomy is short, with clear granularity, supporting simple and straightforward assignments and workflow, with clear and widely understood indicators of event nature.

5.2 Source/target specifics taxonomy

Along with security event taxonomy, we have added attack source/target classification, which stems partially from the need to complement MkII with more specific information about attack origin or destination, not suitable into global (security event wide) taxonomy. Source/target tags are designed base on various omissions, identified in chapter *4*: *Discussion and IDEA examples*. Some of the source/destination types have been already shown in examples, however here comes full list. These classification names are meant to be used as (possible multiple) tags in "*Source.Type*" or "*Target.Type*" field of IDEA messages.

Proxy
Describes service providing indirect access to other services. May denote HTTP proxies, SOCKS proxies and others. Not necesarilly malicious - but since discovered during or as means of security event, worth inspecting.

OriginMalware
Information (usually hostname or URL) was discovered by static analysis of malware binary. Not necesarilly malicious, may have been inserted as a decoy - but worth inspecting.
OriginSandbox
Information (usually hostname or URL) was discovered by sandbox or live-mode analysis of malware binary. Not necesarilly malicious, may have been inserted as a decoy - but worth inspecting.
OriginSpam
Information (usually hostname or URL) was extracted from spam message/data. Not necesarilly malicious, may have been inserted as a decoy - but worth inspecting.
Phishing
Host of the phishing text. Usually a web page over HTTP, however not necessarily.
Malware
Host of the malicious code. Usually a web page, however not necessarily – another example is FTP, or even raw TCP socket.
MITM
Host, conducting man-in-the-middle attack.
Spam
Origin of the spam (be it common spam, phishing or fraud message). May apply to SMTP MTAs, but also to web sites (for example comment spam), instant messaging gateways and others.
Backscatter
Reflector of the attack/event. May be used for SMTP protocol in case of spam bounce, or for DNS/SNMP/NTP and others in case of reflection or amplification attacks.
Open
Host's service access is unlimited. May apply to SMTP MTAs ("open relays"), web proxies, open resolvers and others.
Poisoned
Host's service provides data, manipulated by attacker. Usually applies to services, which provide name translation or redirection data, namely DNS.
FastFlux
Host's service provides rapidly changing data (to evade investigators). Usually applies to services, which provide name translation or redirection data, namely DNS.
Botnet
Machine/service is part of the botnet, i. e. runs cooperating and/or remotely controlled malware.
CC
This part of the botnet is the command-and-control server.

5.2.1 Examples

Tags are designed to be grouped where applicable, one or more keyword may be used to describe particular source or target of the event..

- Open proxy: *Open*, *Proxy*
- Botnet command-and-control server: *Botnet*, *CC*
- Botnet drone: *Botnet*
- Botnet drone, acting as proxy: *Botnet*, *Proxy*
- MTA sending phishing (or other) spams: *Spam* (part of phishing is communicated by event type *Fraud.Phishing*)
- MTA returning misdirected bounces: *Spam*, *Backscatter*
- URL extracted from (even phishing) spam: *OriginSpam* (not *Spam*, URL itself is not spammer)
- URL, extracted from phishing spam and verified pointing to phishing page: *OriginSpam*, *Phishing*

5.3 Detector specifics taxonomy

Moreover, as various sources and types of security detectors and probes spring up every now and then, no human is able to know the types, names and functionality of majority of them. Also, many detectors are created in-house and their name, moreover internals, are not available. We have thus added detector classification, allowing recipients to make assumptions about detection methodologies. For example, security breach detected by honeypot may bear high significance and trustworthiness for some recipients, because to get detected on the honeypot, the real attack must have succeeded. On the other hand, when deduced by flow statistical analysis, probability of false positive may be significantly higher – and trustworthiness lower.

These classification names are meant to be used as (possible multiple) tags in "*Node.Type*" field of IDEA messages.

5.3.1 Medium tags

Describe the origin of the data.

Connection
: Analysis of connections to particular host (LaBrea, iptables logs, ...)

Datagram
: Packet header analysis (iptables, ...)

Content
: Stateful datagram content and/or application protocol based analysis (Snort, Suricata, ...)

Data
: Analysis of local application data (SpamAssassin, antivirus under MTA, ...)

File
: File or host filesystem based analysis (Aide, Tripwire, antivirus, antimalware, ...)

Flow
: Netflow based analysis (FTAS, FlowMon, ...)

Log
: System log based (Logcheck, SSHGuard, Prelude with LML, also other analyzers of application protocols...)

Protocol
: Analysis of application protocol (Dionaea, Hihat, Policyd, Asterisk, greylisting, nolisting...)

Host
: Watching/analysis of machine state (Nagios, SNMP watchguards, ...)

Network
: Watching/analysis of general network state (Nagios, SNMP watchguards, HP OpenView, ...)

Correlation
: Engines, correlating various data, or data from various sources (Prelude, ACARM-ng, ...), additional tags describing the correlated sources should be also used.

External
: External source, additional tags describing character of the source should be also used.

Reporting
: Incident reporting, ticket systems, human detected events, additional tags describing the source should be also used.

5.3.2 Method tags

Describes the technique used to discover security events from the medium.

Blackhole
: Detectors based on redirection, triggered by known aspect of malicious traffic (for example sinkhole DNS servers, diverting traffic based on knowledge of botnet name generation).

Signature
: Signature based ids' (SpamAssassin, Vipul's Razor, Snort, antivirus, ...)

Statistical
: Statistical anomaly analysis (SpamAssassin, SSHGuard, usually netflow based detectors).

Heuristic
: Heuristical, approximative methods or combination of various methods (described by additional tags).

Integrity

File or system integrity checker (Samhain, Tripwire, Aide, ...)

Policy

Detection of protocol/data policy violations (Ossec, greylisting, nolisting, Postfix SMTP rules itself, ...)

Honeypot

Detection traps (Kippo, Dionaea, Hihat, Asterisk based honeypots, ...)

Tarpit

Services or honeypots intentionally holding and delaying incoming connections (LaBrea, greylisting, Stockade, ...)

Recon

Reconnaissance and vulnerability scanning (Nmap, OpenVAS...)

Monitor

Monitoring of production machines, services, applications (Nagios, SNMP monitors, HP OpenView, ...)

5.3.3 Examples

Tags are designed to be grouped where applicable, one or more keyword may be used to describe particular detector of the event.

- SSH bruteforce detector: *Log*, *Statistical*
- Events from external LaBrea: *External*, *Connection*, *Tarpit*
- Events from advanced Postfix installation with Policyd and SpamAssassin: *Protocol*, *Policy*, *Data*, *Signature*, *Statistical*, *Tarpit*

6 Conclusion

In this report we have created mapping of various incident taxonomies to each other, identified some practical deficiencies and omissions in most recent of them – MkII, and recommended and implemented modifications. We have also created auxiliary classifications of attack sources/destination and detection nodes–probes, and shown real world examples to verify feasibility. The whole specification is available at [19].

Taxonomy mapping is also readily usable for translation between classification in various security team, thus simplifying teams work [15].

We have decided to go for more of the practical approach, higher level, than exhaustive CAPEC approach. Too detailed taxonomy would enable more precise description of incident type for machines, but would also mean much higher burden on both generating and understanding of events by human operators. However, to enable more information where feasible, we have established facility for additional tags, which can explain various specific facets of the event, and have defined dictionaries for them.

With this updates IDEA is able to reasonably encompass majority of information from other taxonomies in simple to use and comprehensible manner, and describe all security events we have encountered so far, along with useful detail about concerned parties, which usually plays significant role in further event assessment.

6.1 Future research

We will now work hardly on finalizing of real world implementations, hunt for specifics and outliers and incorporate them into formats and taxonomies accordingly.

Should the world show us that our taxonomy is not sufficient, there are still possibilities to extend it or make it more detailed; or, as our IDEA format is extensible, retract and pursue completely different approach.

However, common language and unambiguous representation is just a starting point. All this work creates basis for gathering of the data, their deep analysis, for searching for methods to correlate, identify patterns, and (in ideal case) prevent more severe security breaches.

7 Acknowledgement

The work has been supported by the CESNET association and the operator of the Czech National Research and Education Network referred to as CESNET2 within its “Large Infrastructure” (LM2010005) research programme, running within 2010–2015 timeframe.

References:

[1] Kácha, P., "IDEA: Designing the Data Model for Security Event Exchange", *17th International Conference on Computers: Recent Advances in Computer Science*, WSEAS, Rhodos, 16 July 2013, ISBN: 978-960-474-311-7, ISSN: 1790-5109.

[2] J. D. Howard, T. A. Longstaff, *A Common Language for Computer Security Incidents*. Sandia National Laboratories, October 1998. SAND98-8667. Available: http://www.cert.org/research/taxonomy_988667.pdf

[3] *Common Attack Pattern Enumeration and Classification* [online]. MITRE. Cited 13 February 2014. Available: http://capec.mitre.org

[4] *WP4 Clearinghouse Policy* [online]. eCSIRT.net. © 2002-2003 by PRESECURE Consulting GmbH, Germany. Available: http://www.ecsirt.net/cec/service/documents/wp4-clearinghouse-policy-v12.html#HEAD6

[5] D. Stikvoort, *Incident Classification* [online]. 23 May 2013. Available: http://www.terena.org/activities/tf-csirt/meeting39/20130523-DV1.pdf

[6] T. Longstaff, *Where the Wild Things Are* [online]. NCSC-NL International Conference 2011. Available: https://www.ncsc.nl/binaries/en/conference/conference-2011/speakers/tom-longstaff/1/TomLongstaffPresentation.pdf

[7] E. Koivunen, *Effective Information Sharing for Incident Response Coordination*. Aalto University, 30 May 2010. Available: http://personal.inet.fi/koti/erka/Studies/DI/DI_Erka_Koivunen.pdf

[8] Kácha, P., *OTRS: streamlining CSIRT incident management workflow*. 13th WSEAS International Conference: Recent Advances in Computer Engineering. WSEAS, Rhodos, 2009, ISBN: 978-960-474-099-4, ISSN: 1790-5109.

[9] A. Cormack, *Proposed top level classification of incidents* [online]. TERENA. Cited 13 February 2014. http://www.terena.org/activities/tf-csirt/pre-meeting3/TLversion0_2.html

[10] *CIF API Feed Types v1* [online]. Cited 13 February 2014. Available: https://code.google.com/p/collective-intelligence-framework/wiki/API_FeedTypes_v1

[11] *\CIF Taxonomy Assesment v1* [online]. Cited 13 February 2014. Available: https://code.google.com/p/collective-intelligence-framework/wiki/TaxonomyAssessment_v1

[12] *Warden* [online]. CESNET. Copyright 2010-1013. Last updated 17 April 2013. Available: http://warden.cesnet.cz

[13] *Warden archive* [online]. CESNET. Cited 13. March 2014. Available: ftp://homeproj.cesnet.cz/tar/warden/warden-client-2.1.tar.gz

[14] *TippingPoint Event Taxonomy, Version 2.2* [online]. Cited 13 February 2014. Available: http://hitec.com.do/carlos/CarlosMeza/__Curso_IPS/ProductDocumentation/TECHD94-EventTaxonomy.pdf

[15] Kácha, P., *Incident Classification Comparison (with eCSIRT.net mkII as main reference)* [online], CESNET, 10 January 2014. Available: https://csirt.cesnet.cz/IDEA/Classifications?action=AttachFile&do=get&target=Incident+classification+comparison.ods

[16] Debar, H., Dacier, M., & Wespi, A. *Towards a taxonomy of intrusion-detection systems*. Computer Networks, Vol. 31, Issue 8, pages 805-822. Elsevier, 1999. ISSN: 1389-1286.

[17] Abbas, A., El Saddik, A., & Miri, A. *A comprehensive approach to designing internet security taxonomy*. Canadian Conference on Electrical and Computer Engineering (CCECE'06), pages 1316-1319. IEEE, May 2006. ISBN: 1-4244-0038-4

[18] *Google Hack Honeypot* [online]. Honeynet Alliance. Cited 13 February 2014. Available: http://ghh.sourceforge.net/

[19] P. Kácha, *IDEA/Classifications and Enumerations [online],* CESNET. Cited 13 February 2014. Available: https://csirt.cesnet.cz/IDEA/Classifications

[20] Vachek, P., "CESNET Audit System", *Proceedings of the 13th WSEAS International Conference on COMPUTERS*, Rodos Island, July 23-25, 2009, ISBN: 978-960-474-099-4.