Types of incidents and taxonomies in the wild mapped to mkII

Pavel Kácha <ph@cesnet.cz>, Jan 10, 2014, CESNET, Czech Republic

1 Introduction

MkII[1] is attempt by Don Stikvoort (SURFcert) to revive eCSIRT.net[2] (and formerly Telia CERTCC's Jimmi Arvidsson's) taxonomy and adapt it to nowadays security team needs. This document accompanies table "Incident Classification Comparison (with eCSIRT.net mkII as main reference)" in attempt to identify potential omissions or disproportions against other taxonomies or real world examples of incidents. For short results scroll down to chapter 3 - Conclusion.

1.1 Taxonomies problems

Creating any taxonomy, and security incident taxonomy in particular, is in no way simple task. Various users are driven by various needs and as expectations usually 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 as tools for machine based security event dissemination continue to emerge, usefulness of common ground, which security teams could use at least for mapping other classifications to, becomes apparent.

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

1.1.1 Low level vs high level

Taxonomy may attempt to describe precise details of incident, as in venerable Howard/Longstaff[3] 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?).

1.1.2 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.

1.1.3 Exhaustive vs transparent

On the one side, incident can be classified very precisely, as for example in CAPEC[4] 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] http://www.terena.org/activities/tf-csirt/meeting39/20130523-DV1.pdf
[2] http://www.ecsirt.net/cec/service/documents/wp4-clearinghouse-policy-v12.html#HEAD6
[3] http://www.cert.org/research/taxonomy_988667.pdf
[4] http://capec.mitre.org/

1.1.4 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).

1.2 Description

1.2.1 Incident Classification Comparison Table

“Incident Classification Comparison (with eCSIRT.net mkII as main reference)” table is an attempt to map various taxonomies of security events and even some real world incident descriptions onto each other. Taxonomies are:

- eCSIRT.net by eCSIRT.net[5] (and formerly Telia CERTCC's Jimmi Arvidsson)
- eCSIRT.net mkII (Don Stikvoort + SURFnet, based on eCSIRT.net)
- Howard/Longstaff
- Longstaff NCSC 2010
- CIF API Feed Types v1
- CIF Taxonomy Assesment v1
- FICORA
- Andrew Cormack (proposal at Terena)
- SURFcert
- CESNET-CERTS
- Warden 2
- CESNET Mentat
- HP TippingPoint Event Taxonomy V 2.2

Both eCSIRT.net taxonomies are at the very left side as the main reference, mostly because they turned out to be the most exhaustive (not counting CAPEC, which has not been included because of magnitude of its scope and detail).

Corresponding incident type groups are clustered together where possible, and 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.

[5] http://www.ecsirt.net/cec/service/documents/wp4-clearinghouse-policy-v12.html#HEAD6

2 Discussion related to mkII

2.1 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 attempts to monetize on impression of legitimity of certain company's email (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 webpages, injected with phising or malware, open relay mailservers, open recursive resolvers, etc.

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

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.

Examples:

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

IP Filters/Accept

2.2 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 does not belong into general taxonomy.

Examples:

HP Tipping Point

Traffic Thresholds/Traffic Treshold

Traffic Thresholds/Application Treshold

Traffic Thresholds/Other

Application or Protocol Anomaly/Application Anomaly

Application or Protocol Anomaly/Evasion Technique

Application or Protocol Anomaly/Other Anomaly

Application or Protocol Anomaly/Protocol Anomaly

2.3 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. Moreover, it in fact describes the means, the technical facet of the attack, which I believe should again stay out of scope of general taxonomy.

Example:

Cesnet CERTS

Bounce

2.4 Scans

Number of existing taxonomies distinguishes between specific types of IP based reconnaisance, the basic types being host scan, port scan, service scan, application scan, port sweep, icmp probe. This again denotes techical facet of the attack, which can be communicate 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 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 reconnaisance or not, without additional information usually from other sources. Most important information, which this distinction conveys, is severity of the attack, and that's also orthogonal information, which might get communicated by other ways, but would overly complicate general taxonomy.

Examples:

HP Tipping Point

Reconaissance or Suspicious Access/Host scan

Reconaissance or Suspicious Access/Port scan

Reconaissance or Suspicious Access/Suspicious Service Request

Reconaissance or Suspicious Access/Suspicious Application Access

Reconaissance or Suspicious Access/Other

Reconaissance or Suspicious Access/Host scan

Warden 2

Portscan

Probe

Mentat

- Probe
- Portscan
- Connection attempt
- Other
- Ping probe
- SYN/ACK scan or DOS attack

2.5 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.

Examples:

Mentat

- EPMAPPER exploitation attempt
- SMB exploitation attempt
- SQL query attempt
- URL attack attempt
- Webattack
- Open recursive resolver

HP Tipping Point

- Vulnerability/Access Validation
- Vulnerability/Buffer-Heap Overflow
- Vulnerability/Configuration Error
- Vulnerability/Denial of Service (Crash/Reboot)
- Vulnerability/Invalid Input (Command Injection, XSS, SQLi, etc.)
- Vulnerability/Other
- Vulnerability/Race Condition

2.6 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 servers is of more benefit, then cleaning up workstation infected by drone. 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.

Examples:

CIF API Feed Types v1

- infrastructure/botnet, url/botnet, domain/botnet
- infrastructure/fastflux, domain/fastflux

CIF Taxonomy Assesment v1

Botnet

Fastflux

Mentat

Botnet Drone

Botnet Proxy

Botnet_c_c

2.7 (D)DOS

At least one examined taxonomy incorporates specific identification of (D)DOS. This technical level info again does not belong into high level taxonomy, however, should get considered in incident report communication.

Examples:

HP Tipping Point

Distributed Denial of Service Syn Flood Attack/Other Flood Attack

Distributed Denial of Service Syn Flood Attack/Iterative Application Attack

Distributed Denial of Service Syn Flood Attack/Other

2.8 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 concrete specific person (identity theft), or general con (variation of social engineering). I'd suggest adding "Fraud/Scam" category for clarity.

Examples:

CESNET CERTS

Phishing

Pharming

Scam (example: Nigerian 419 scam)

additional Identity theft

2.9 Suspicious

URLs found in spam messages or in sandboxed malware binaries may or may not be necessarilly evil. They are definitely suspicious, but spammers and malware creators often incorporate innocent URLs to lure automated tools astray. I'm 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).

Examples:
Mentat

Sandbox URL

Spam URL

2.10 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[6]) or by local IDS' systems. This type of information gathering does not precisely fit into any mkII subcategory, I'd suggest adding “Information Gathering/Searching” category.

Examples:

CIF Taxonomy Assesment v1:

Searches

2.11 Local

At least one taxonomy incorporates breaches into company policies. As these can be local specific, they don't belong into general taxonomy.

Examples:

HP Tipping Point

Security Policy/Autentication Failure (login failed, bruteforce, etc.)

Security Policy/Chat and Instant Messaging

Security Policy/Email Attachments

Security Policy/Forbidden Application Access or Service Request (Telnet, SMB Null Session, etc.)

Security Policy/Other

Security Policy/P2P

Security Policy/Spyware

Security Policy/Streaming Media

2.12 Unknown

I was not able to deduce what email/registrant category in CIF stands for.

Example:

CIF Api Feed Types v1

email/registrant

2.13 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

[6] http://ghh.sourceforge.net/

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.

3 Conclusion

MkII comes out as the most comprehensive of still usable solutions. From comparison with other taxonomies and several real world incidents I have following suggestions:

1. Add “Anomaly” category, with following subcategories (incident examples) for start: *Traffic*, *Connection*, *Protocol*, *System*, *Application*, *Behaviour* (see 2.2 and 2.13).
2. Add “Scam” incident example into “Fraud” (see 2.8).
3. Add “Searching” incident example into “Information Gathering” (2.10).
4. Don't stay rigid, allow side-stepping, make taxonomy extensible by users (1.1.4).
5. Allow multicategorization, where applicable (1.1.4).