Mapping investigations and cases in MISP

E.205

CIRCL Computer Incident Response Center Luxembourg

MISP Project
https://www.misp-project.org/

April 8, 2022 - v0.7

Objectives of this module

- Recap on MISP data model and distribution levels
- Data from cases to be structured and encoded:
  - **Network indicators**: `ip`, `domain`, `url`, ...
  - **Files and binaries**: non-malicious / malicious `payload`
  - **Emails**: `content`, `header`, `attachment`, ...
  - **Web**: URL, `cookies`, `x509`
  - **Cryptographic materials**: public / private `key`, `certificate`
  - **Infrastructure and devices**
  - **Financial fraud**: `bank-account`, `phone-number`, `btc`
  - **Person**: `name`, `online accounts`, `passport`, `visa`
  - **Support tools**: `yara`, detection/remediation scripts
  - **Vulnerabilities**: `cve`
  - **External analysis**: Reports, blogpost, ransome notes
- Relationships and timeliness
- Enrichments via module and correlation
- Preparing data for sharing with other LE partners, CSIRT, SOC

MISP Data model and distribution levels

MISP Event

Event

Encapsulations for contextually linked information.

Purpose: Group datapoints and context together. Acting as an envelop, it allows setting distribution and sharing rules for itself and its children.

Usecase: Encode incidents/events/reports/...

- events can contain other elements such as attributes, objects and eventreports.
- The distribution level and any context added on an event (such as taxonomies) are propagated to its underlying data.

MISP Attribute

Attribute

Basic building block to share information.

Purpose: Individual data point. Can be an indicator or supporting data.
Usecase: Domain, IP, link, sha1, attachment, ...

- attributes cannot be duplicated inside the same event and can have sightings.
- The difference between an indicator or supporting data is usualy indicated by the state of the attribute's `to_ids` flag.

MISP Object

Advanced building block providing attribute compositions via templates.

Purpose: Groups attributes that are intrinsically linked together.
Usecase: File, person, credit-card, x509, device, ...

- objects have their attribute compositions described in their respective template. They are instanciated with attributes and can reference other attributes or objects.
- MISP is not required to know the template to save and display the object. However, *edits* will not be possible as the template to validate against is unknown.

MISP Relationships (aka object reference)

↗ Object Reference

Relationships between individual building blocks.

Purpose: Allows to create relationships between entities, thus creating a graph where they are the edges and entities are the nodes.

Usecase: Represent behaviours, similarities, affiliation, …

▶ references can have a textual relationship which can come from MISP or be set freely.

MISP Event report

Event Report

Advanced building block containing formated text.

Purpose: Supporting data point to describe events or processes.

Usecase: Encode reports, provide more information about the event, ...

▶ Event reports are markdown-aware and include a special syntax to reference data points or context.

General rule of thumb

Which structure should be used when encoding data?

- **Attribute** vs **Object**
  - If the value is contextually linked to another element or is a subpart of a higher concept, an **object** should be used
  - If the value is part of a large list of atomic data, an **attribute** should be used
- **`Annotation` Object** vs **Event Report**
  - If it is possible to encode the text (raw text or markdown), an **event report** is prefered
  - If the text is written in a specific format (e.g pdf, docx), an **`annotation` object** should be used

Case study 1: Scam call

Case study 1: Scam call

Case: A victim was asked to transfer money to a novice scammer

Chronology - 2022-03-24

11:42:43 UTC+0: Scammer called the victim pretending to be a microsoft employee

11:47:27 UTC+0: Scammer convinced the victim to be helped via remote desktop assistance

12:06:32 UTC+0: Scammer downloaded the binary on the victim's computer

12:08:18 UTC+0: Scammer installed the binary on the victim's computer

12:17:51 UTC+0: Scammer asked the victim to transfer money on a bank account for the help he provided

12:25:04 UTC+0: Victim executed the money transfer

2022-03-25 08:39:21 UTC+0: Victim contacted police

Case study 1: Scam call

Collected evidences

- RDP Log file
- Installed binary
- Victim's browser history
- Bank account statement
- Victim's phone call log

Data extracted from evidences

- Scammer's **ip address**
- Potentially **malicious binary**
- **URL** (and **domain**) from which the binary was downloaded
- Scammer's **bank account** and **phone number**
- Scammer's full name and nationality

Case study 1: Scam call

Extracted values

- `194.78.89.250`
  - ip-address from log file
- `bin.exe`
  - downloaded binary
- `https://zdgyot.ugicok.ru/assets/bin.exe`
  - download URL
- `GB 29 NWBK 601613 31926819`
  - IBAN number
  - Swift: `NWBK`, Account number: `31926819`, Currency: `GBP`
- `+12243359185`
  - phone number
- `Wallace Breen` is from `GB`
  - name and nationality

Case study 1: Scam call

Tasks

1. Create an new *event* to be shared with **all**
2. Encode `binary` to be shared with **CSIRT**
3. Encode `ip address` to be shared with both **ISP** and **CSIRT**
4. Encode `domain` and `url` to be shared with both **ISP** and **CSIRT**
5. Encode `bank account` to be shared with **Financial sector**
6. Encode `phone number` to be shared with **Telecomunication sector**
7. Encode `full name` and `nationality` to be shared with **LEA only**
8. Add relationships to recreate the events
9. Add time component to recreate the chronology
10. Perform enrichments on the binary, and other attribute
11. Add contextualization
12. Create a small write-up as an *event report*
13. Review the distribution level and publish

Case study 1: Scam call

▶ Creating the *event* in MISP

Date

2022-03-24

Distribution

All communities

Threat Level

Low

Analysis

Completed

Event Info

Successful Scam call involving money transfer

Extends Event

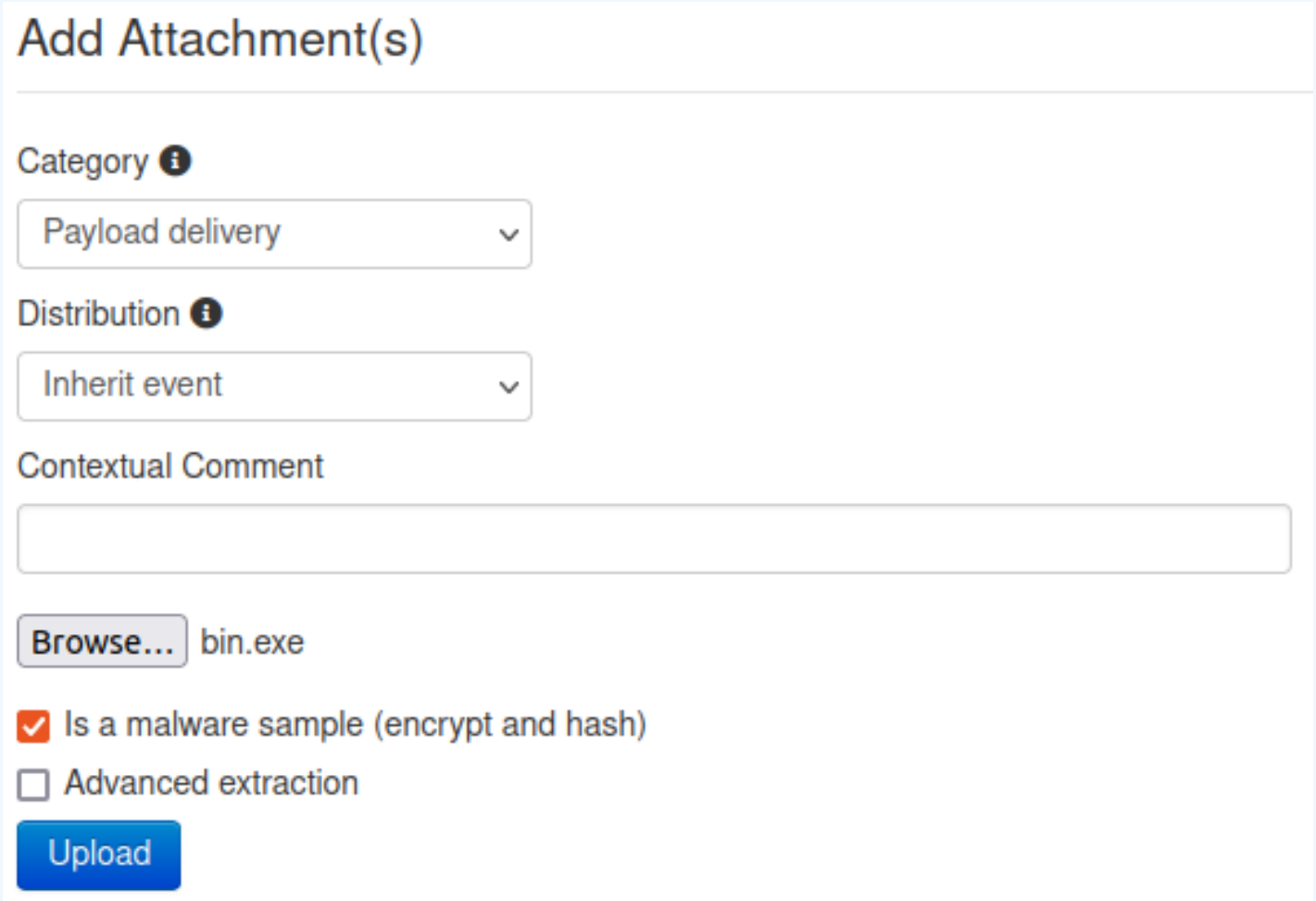
Event UUID or ID. Leave blank if not applicable.

Submit

Case study 1: Scam call

▶ Adding the binary as attachment

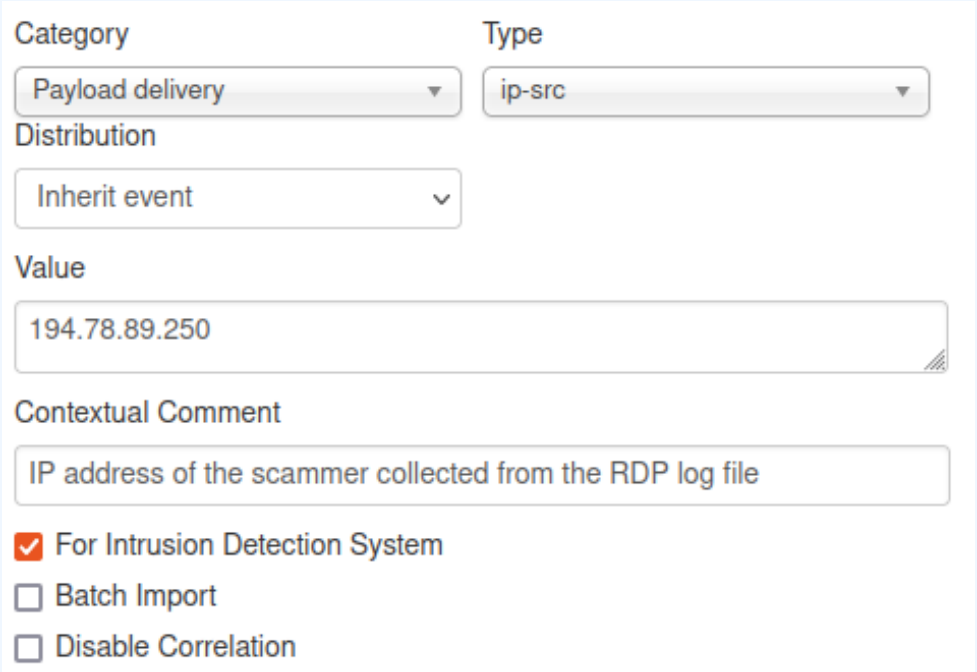
- Pick the `Payload Delivery` category
- Check *Is a malware sample*



Case study 1: Scam call

▶ Encode the IP address

- Encode the IP address of the scammer with an *attribute*
- Pick the `Payload Installation` category and `ip-src` type
- Check the `For Intrusion Detection System`
- Add a contextual comment such as
  - `IP address of the scammer collected from the RDP log file`



Case study 1: Scam call

- Encode the domain/URL used to download the binary

- As these two attributes are contextually linked between each others, we should use an URL *object*
- Add a contextual comment such as
  - `URL used by the scammer to download the binary`
- Include at least: `url`, `domain` and `ressource_path`

Case study 1: Scam call

Object pre-save review

Make sure that the below Object reflects your expectation before submitting it.

Name	url
Template version	9
Meta-category	network
Distribution	Inherit event
Comment	URL used by the scammer to download the binary
First seen	2022-03-24T12:06:32.000000+00:00
Last seen	

Attribute	Category	Type	Value	To IDS
url	Network activity	url	https://zdgyot.ugic0k.ru/assets/bin.exe	Yes
domain	Network activity	domain	zdgyot.ugic0k.ru	Yes
domain_without_tld	Other	text	zdgyot.ugic0k	No
resource_path	Other	text	/assets/bin.exe	No
scheme	Other	text	https	No
tld	Other	text	ru	No

Update object | Back to review | Cancel

Case study 1: Scam call

▶ Encode the bank account

- As these 4 attributes are contextually linked between each others, we should use an `bank-account` *object*
- Add a contextual comment such as
  - `Bank account that received the money. Supposed to belong to the scammer`
- Include at least: `iban`, `swift`, `account` and `currency_code`

Case study 1: Scam call

Object pre-save review

Make sure that the below Object reflects your expectation before submitting it.

Name	bank-account
Template version	3
Meta-category	financial
Distribution	Inherit event
Comment	Bank account that received the money. Supposed to belong to the scammer
First seen	
Last seen	

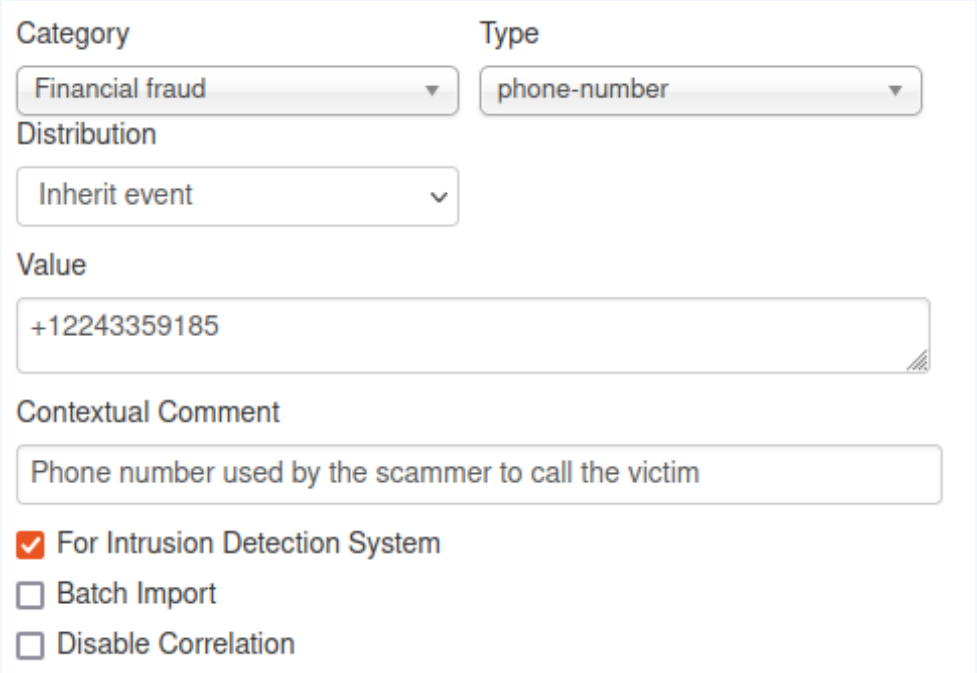
Attribute	Category	Type	Value	To IDS
iban	Financial fraud	iban	GB29NWBK60161331926819	Yes
swift	Financial fraud	bic	NWBK	Yes
account	Financial fraud	bank-account-nr	31926819	Yes
currency-code	Other	text	GBP	No

Update object | Back to review | Cancel

Case study 1: Scam call

▶ Encode the phone number

- Pick the `Financial Fraud` category and `phone-number` `type`
- Add a contextual comment such as
  - `Phone number used by the scammer to call the victim`
- Check *For Intrusion Detection System*



Case study 1: Scam call

- Encode the full name and nationality

- As these attributes are contextually linked between each others, we should use a `person` *object*
- Add a contextual comment such as
  - `Name of the scammer given to the victim`
- Include at least: `full-name`, `nationality` and `role`

Case study 1: Scam call

Object pre-save review

Make sure that the below Object reflects your expectation before submitting it.

Name	person
Template version	16
Meta-category	misc
Distribution	Inherit event
Comment	Name of the scammer given to the victim. Name confirmed to be the owner of the bank account and phone number
First seen	
Last seen	

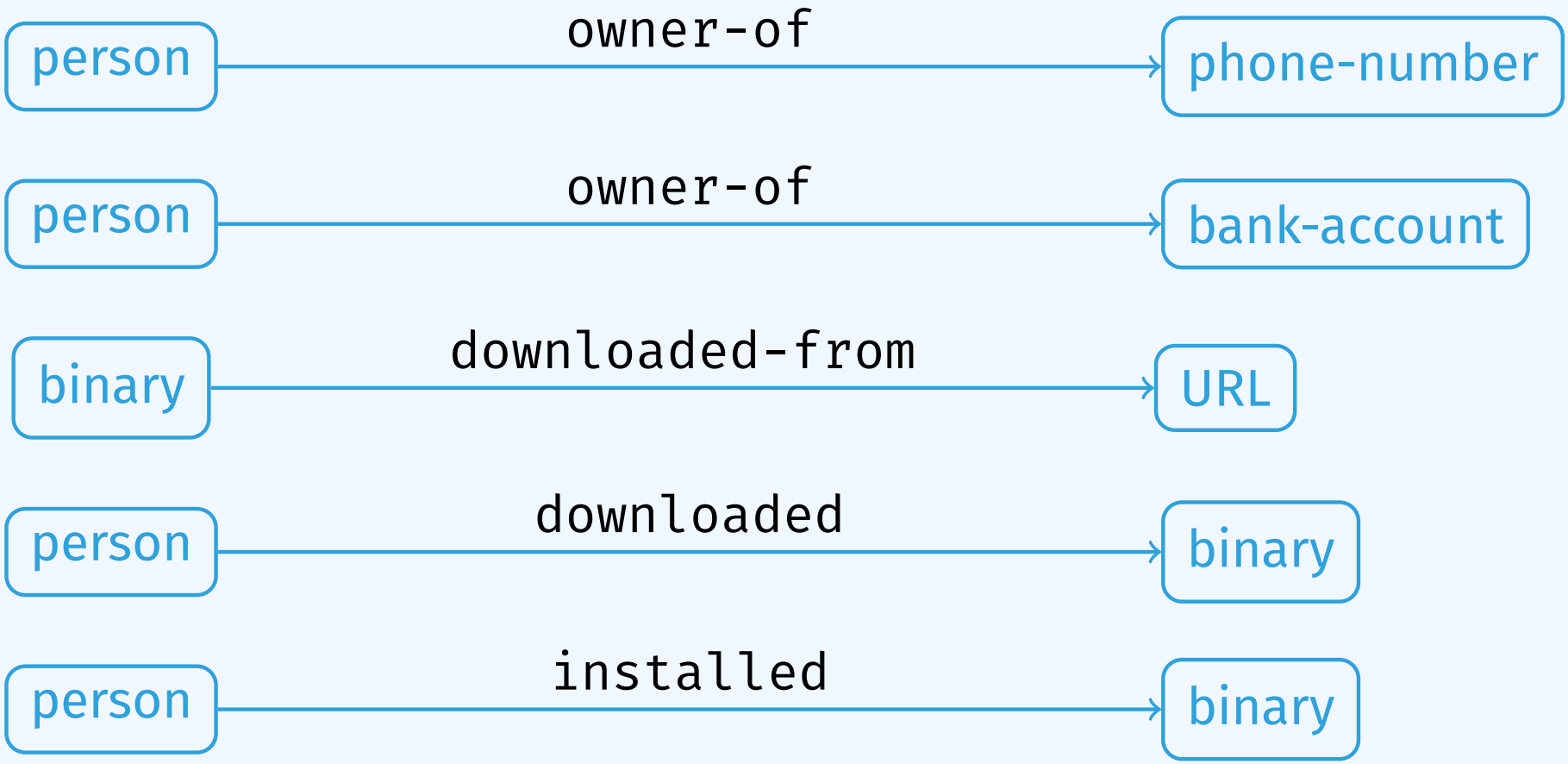
Attribute	Category	Type	Value	To IDS
last-name	Person	last-name	Breen	No
full-name	Person	full-name	Wallace Breen	No
first-name	Person	first-name	Wallace	No
role	Other	text	Accused	No
gender	Person	gender	Male	No
nationality	Person	nationality	British	No

Update object | Back to review | Cancel

Case study 1: Scam call

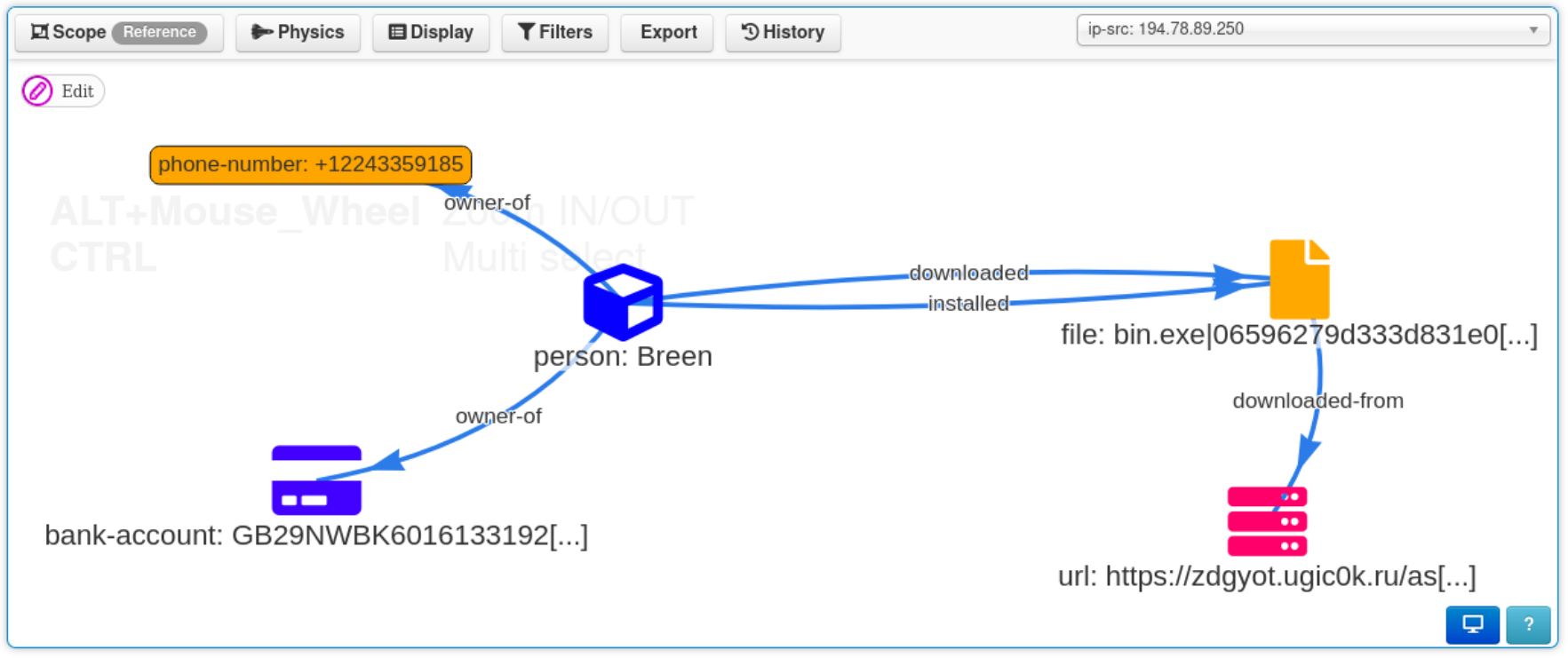
▶ Creating relationships

Add (at least) these relationships to recreate the story



Case study 1: Scam call

▶ Creating relationships

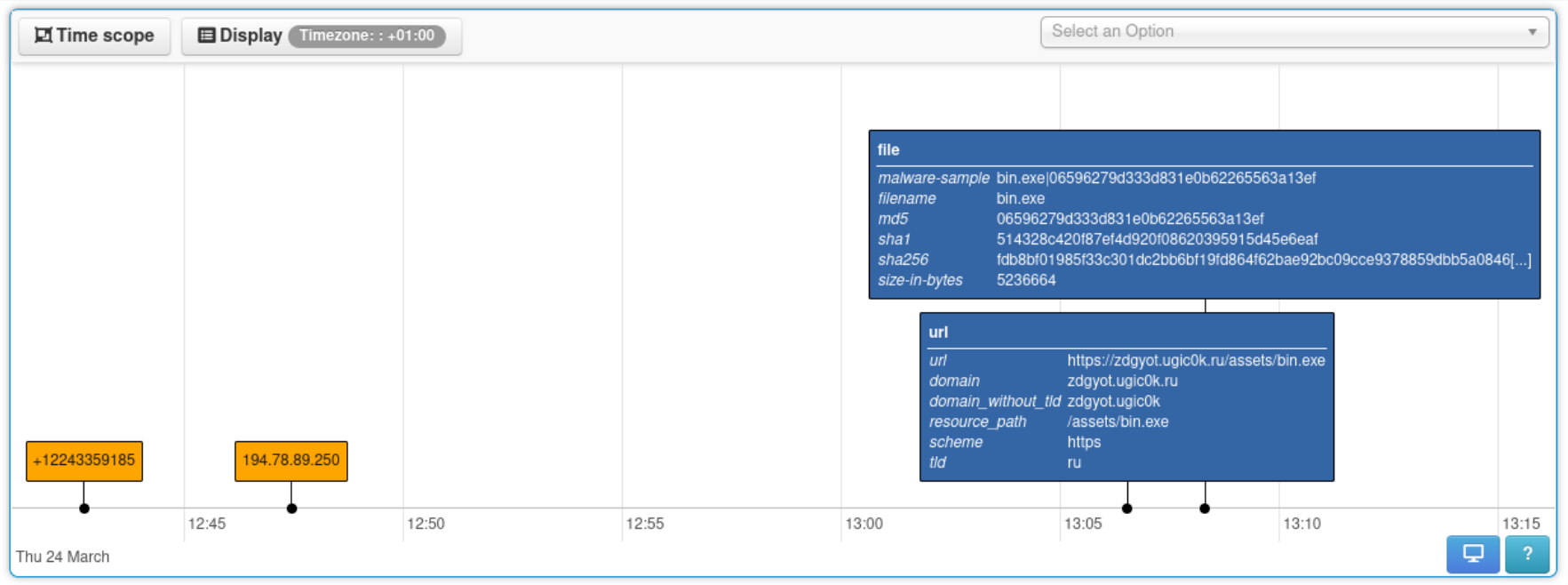


Case study 1: Scam call

▶ Adding time component

The time component is useful to recreate the chronology

- Main focus is the Cyber Threat Intelligence (CTI) aspect



Case study 1: Scam call

▶ Perform enrichments

- Scammer IP address to get its location
- Binary to check if it's an existing (and malicious) application

Mmdb Lookup:

Object: geolocation

country	Belgium
countrycode	BE
latitude	50.8333
longitude	4
text	db_source: GeoOpen-Country. build_db: 2022-02-05 10:37:33. Latitude and longitude are country average.

Object: geolocation

country	Belgium
countrycode	BE
latitude	50.8333
longitude	4
text	db_source: GeoOpen-Country-ASN. build_db: 2022-02-06 09:30:25. Latitude and longitude are country average.

Object: asn

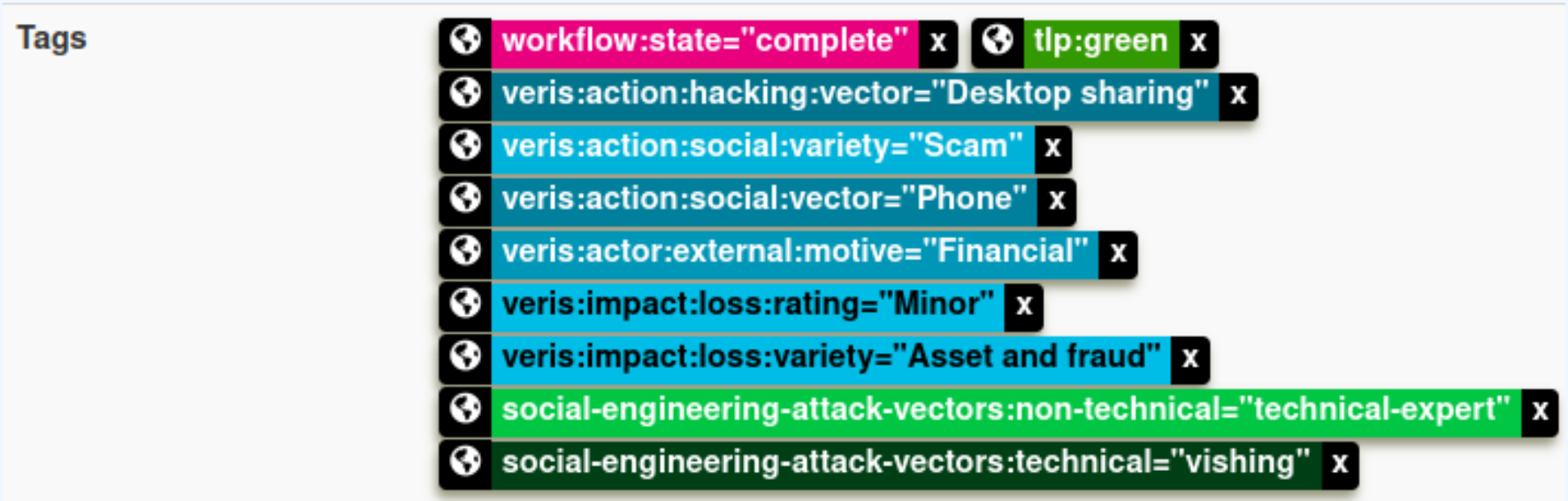
Case study 1: Scam call

▶ Contextualizing the data with *Taxonomies*

- Note: Different country / sectors might use different nomemclature
- Suggestions for tagging with taxonomies:
  - `circl:incident-classification="scam"`
  - `social-engineering-attack-vectors:non-technical="technical-expert"`
  - `social-engineering-attack-vectors:technical="vishing"`
  - `veris:action:hacking:vector="Desktop sharing"`
  - `veris:action:malware:vector="Direct install"`
  - `veris:action:social:variety="Scam"`
  - `veris:action:social:vector="Phone"`
  - `veris:actor:external:motive="Financial"`
  - `veris:impact:loss:rating="Minor"`
  - `veris:impact:loss:variety="Asset and fraud"`
  - `workflow:state="complete"`
  - `tlp:green`

Case study 1: Scam call

- Contextualizing the data with *Taxonomies*



Case study 1: Scam call

▶ Contextualizing the data with *Galaxy Clusters*

- Note: Different country / sectors might use different nomemclature
- Suggestions for tagging with Galaxies Clusters:
  - `MITRE Att&ck Pattern`

Case study 1: Scam call

▶ Mitigations and Detection

Thanks to the MITRE Att&ck contextualization, we can derive preventive measures from their catalogue

- Mitigations
  - Antivirus
  - Behavior Prevention on Endpoint
  - Execution Prevention
  - Network Intrusion Prevention
  - Restrict Web-Based Content
  - Software Configuration
  - User Training
- Detection
  - Application Log
  - Container
  - File
  - Network Traffic
  - Process

Case study 1: Scam call

▶ Write-up with an *event report*

- Create the *event report* with a concise name
- Example: Executive summary of the case
  - Leave its content empty as it can be edited with more ease in the editor afterward
- Write a summary with
  - Quick chronology
  - Written explanation of the steps tooks by the scammer
  - Reference to existing *attributes* or *objects* whenever possible
    - The special syntax is: `@[scope]{uuid}`

Case study 1: Scam call

- Write-up with an *event report*

Executive summary of the case

A victim was called by the suspected scammer person Wallace Breen using the following number: phone-number +12243359185 . The scammer prentended to be a a microsoft employee, managed to convince the victim that he could help by using remote desktop assistance.

Once he had access, the scammer downloaded a binary file bin.exe from the following url url https://zdgyot.ugic0k.ru/assets/bin.exe . He then proceed to install the binary, probably to use it a backdoor for future access.

After the installation, he asked the victim to transfer money to the scammer bank account: bank-account ↦ iban GB29NWBK60161331926819

The day after, the victim suspecting a scam contacted the police.

Technique used

Social vector	veris:action:social:vector="Phone"
Potential hacking vector	veris:action:hacking:vector="Desktop sharing"
Actor motive	veris:actor:external:motive="Financial"
Impacted loss	veris:impact:loss:variety="Asset and fraud"
Loss rating	veris:impact:loss:rating="Minor"

Information collected after analysis

- According to the phone number, IP address and bank account, the scammer person Wallace Breen is very likely based in geolocation ↦ country Belgium .

Timeline

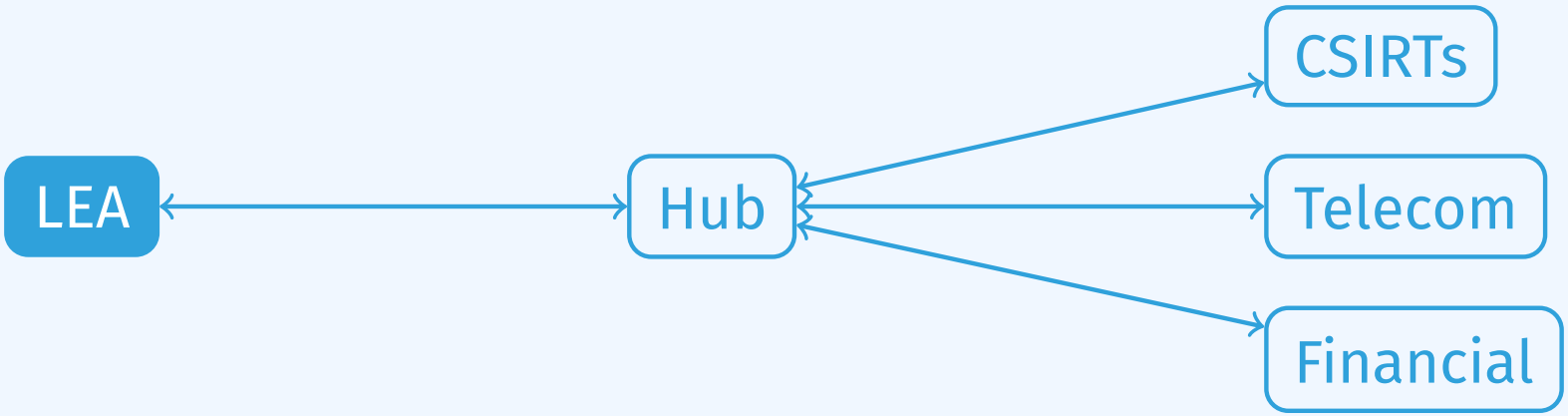
- **2022-03-25 11:42:43 UTC+0**: Scammer called the victim pretending to be a microsoft employee
- **2022-03-25 11:47:27 UTC+0**: Scammer convinced the victim to be helped via remote desktop assistance
- **2022-03-25 12:06:32 UTC+0**: Scammer downladed the binary on the victim's computer
- **2022-03-25 12:08:18 UTC+0**: Scammer installed the binary on the victim's computer
- **2022-03-25 12:17:51 UTC+0**: Scammer asked the victim to transfer money on a bank account for the help he provided
- **2022-03-25 12:25:04 UTC+0**: Victim executed the money transfer
- **2022-03-25 08:39:21 UTC+0**: Victim contacted police

Case study 1: Scam call

▶ Review the distribution level and publish

In our case, we consider the following MISP network topology

- The current instance is owned and managed by a LEA
- The current instance is connected to a central MISP instance acting as a "Hub"
- The "Hub" is connected to various other MISP instances such as other LEAs, CSIRTs, Financial and telecom institutions



Case study 1: Scam call

▶ Review the distribution level and publish

- `binary file`: **All communities**
- `person`: **LEA Sharing group**
- `geolocation`: **LEA Sharing group**
- `ip`: **LEA Sharing group**
  - The IP might be reassigned
- `phone`
  - If part of a telco sharing group **Telco Sharing group**
  - **Connected communities** otherwise
- `bank account`
  - If part of a financial sharing group **Financial Sharing group**
  - **Connected communities** otherwise

→ **Publish the event!**

Case study 2: Ransomware

Case study 2: Ransomware

Case: Ransomware infection via e-mail

Chronology - 2022-03-24

11:42:43 UTC+0: Email containing the ransomware from supposedly Andrew Ryan
11:47:27 UTC+0: Email was read and its attachment opened and executed
11:47:28 UTC+0: Malware add persistence
12:08:18 UTC+0: Malware successfully contacted the C2 to get the PK
12:08:19 UTC+0: Malware saved the PK in the registry
12:25:04 UTC+0: Malware began the encryption process
2022-03-25 08:39:21 UTC+0: Victim contacted the police

Case study 2: Ransomware

Splash message from the Ransomware

Case study 2: Ransomware

Collected evidences

- E-mail received by the victim
- E-mail attachment of the ransomware as an .exe payload
- Windows registry
- Ransomware's public key (PK)
- Captured network traffic
- Message displayed by the ransomware

Data extracted from evidences

- Original **e-mail**
- The actual ransomware **binary**
- **Registry Keys** for persistence and configuration
- **Public Key** used for encryption
- C&C server **ip address** used to generate the Private Key (SK)
- The **bitcoin address** on which the ransom should be paid
- The **person**, impersonated or fake that sent the email

Case study 2: Ransomware

```
Subject: 4829-2375
From: "Andrew_Ryan" <Andrew_Ryan@rindustries.rp>

Please see the attached Iolta report for 4829-2375.

We received a check request in the amount of $19,637.28 for the above referenced file.
      However, the attached report refects a $0 balance. At your earliest convenience,
      please advise how this request is to be funded.

Thanks.

Andrew_Ryan *
Accounts Payable

Ryan Industries
42, Central Control Hephaestus - Rapture
www.rindustries.rp

*Not licensed to practise law.

This communication contains information that is intended only for the recipient named and
       may be privileged, confidential, subject to the attorney-client privilege, and/or
      exempt from disclosure under applicable law. If you are not the intended recipient
      or agent responsible for delivering this communication to the intended recipient,
      you are hereby notified that you have received this communication in error, and
      that any review, disclosure, dissemination, distribution, use, or copying of this
      communication is STRICTLY PROHIBITED. If you have received this communication in
      error, please notify us immediately by telephone at 1-800-766-7751 or
      1-972-643-6600 and destroy the material in its entirety, whether in electronic or
      hard copy format.
```

Case study 2: Ransomware

Extracted values

- `e-mail` from previous slide
- `cryptolocker.exe`
  - Ransomware attached to the mail
- `81.177.170.166`
  - ip-address of a C2 server used to generate the SK
- `HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run "CryptoLocker"`
  - The registry key used for persistence
- `HKCU\SOFTWARE\CryptoLocker VersionInfo`
  - The registry key containing configuration data
- `HKCU\SOFTWARE\CryptoLocker PublicKey`
  - The registry key containing the RSA public key received from the C2 server
- `0x819C33AE`
  - XOR key used to encode the configuration data

Case study 2: Ransomware

```
-----BEGIN PUBLIC KEY-----
MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQDaogllvHPytDAdUWZPk9aWXJ5G
Lk9F+HzDaj5qGXou8XmISwChbia/NC84QmBHTiyg4B1tqVjqk5X6yh6pcZuVw+GX
0CrH5O5o2Q0XVYzYYsEZQB36VHxwm7xTx21yOy2rSOQyOupQ6e7HMGtu7p7+RlWO
D5UfPkv337plrEiUuwIDAQAB
-----END PUBLIC KEY-----
```

- The public key received from the C2 used to encrypt files

- `1KP72fBmh3XBRfuJDMn53APaqM6iMRspCh`
  - Bitcoin address on which to transfer the ransom
- `Andrew Ryan, Andrew_Ryan@rindustries.rp`
  - `Accountant, Suspect & Victim & Originator`
  - Person, e-mail, occupation and role

Tasks

1. Create an new *event* to be shared with **all**
2. Encode data to be shared
3. Add relationships to recreate the events
4. Add time component to recreate the chronology
5. Perform enrichments on the binary, and other attributes
6. Add contextualization
7. Create a small write-up as an *event report*
8. Review the distribution level and publish

Case study 2: Ransomware

▶ Creating the *event* in MISP

Date

2022-03-24

Distribution

All communities

Threat Level

Medium

Analysis

Completed

Event Info

CryptoLocker ransomware infection via e-mail

Extends Event

Event UUID or ID. Leave blank if not applicable.

Submit

Case study 2: Ransomware

▶ Add the original e-mail

- As the email contains multiple contextually linked data points, we should use an `Email` *object*
- Add contextual comment such as:
  - `Email received by the victim containing the ransomware`
- Include at least: `from`, `subject` and body

Case study 2: Ransomware

▶ Add the original e-mail

Object pre-save review

Make sure that the below Object reflects your expectation before submitting it.

Name	email
Template version	18
Meta-category	network
Distribution	Inherit event
Comment	
First seen	2022-03-24T11:42:43
Last seen	

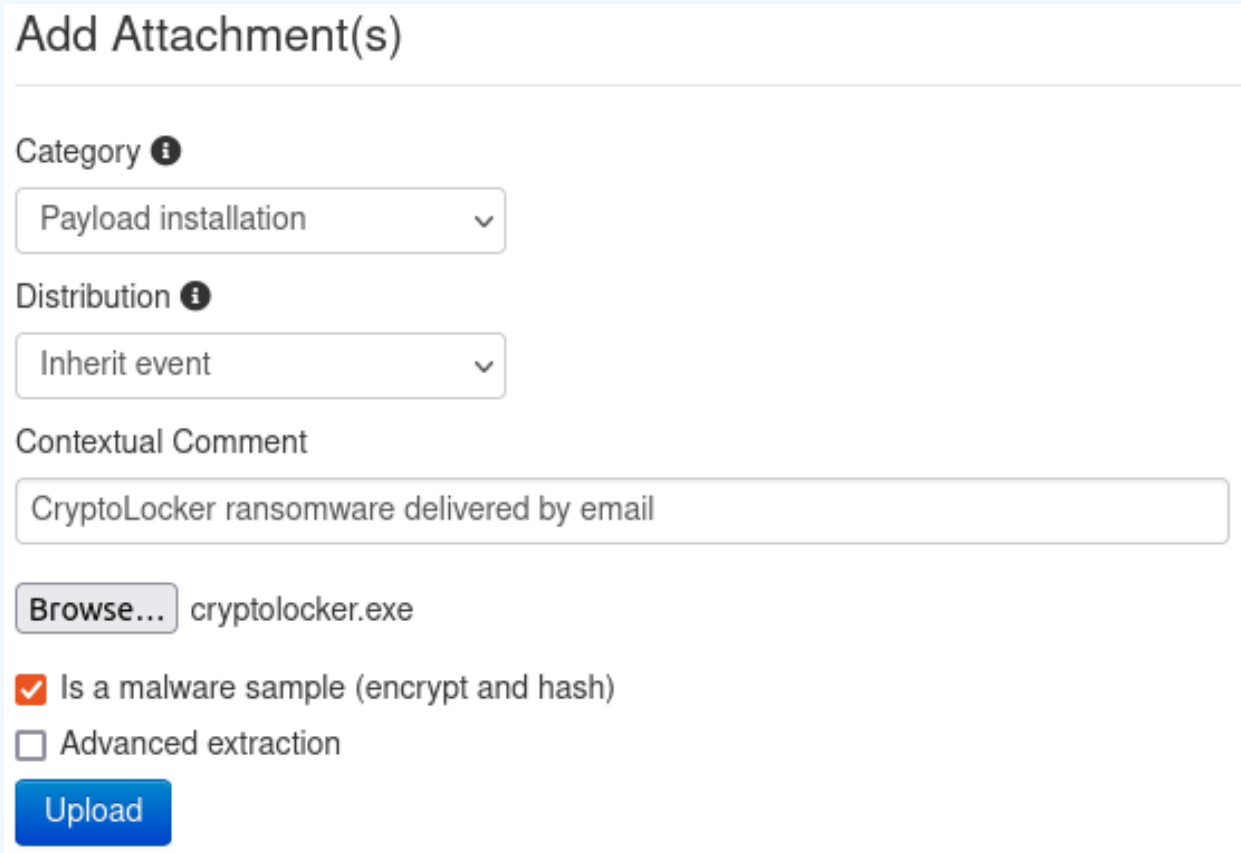
Attribute	Category	Type	Value	To IDS
subject	Payload delivery	email-subject	4829–2375	No
from	Payload delivery	email-src	Andrew_Ryan@rindustries.rp	Yes
email-body	Payload delivery	email-body	Please see the attached Iolta report for 4829-2375. We received a check request in the amount of $19,637.28 for the above referenced file. However, the attached report refects a $0 balance. At your earliest convenience, please advise how this request is to be funded. Thanks. Andrew_Ryan * Accounts Payable Ryan Industries 42, Central Control Hephaestus - Rapture www.rindustries.rp *Not licensed to practise law. This communication contains information that is intended only for the recipient named and may be privileged, confidential, subject to the attorney-client privilege, and/or exempt from disclosure under applicable law. If you are not the intended recipient or agent responsible for delivering this communication to the intended recipient, you are hereby notified that you have received this communication in error, and that any review, disclosure, dissemination, distribution, use, or copying of this communication is STRICTLY PROHIBITED. If you have received this communication in error, please notify us immediately by telephone at 1-800-766-7751 or 1-972-643-6600 and destroy the material in its entirety, whether in electronic or hard copy format.	No

Create new object | Back to review | Cancel

Case study 2: Ransomware

▶ Add the ransomware binary as attachment

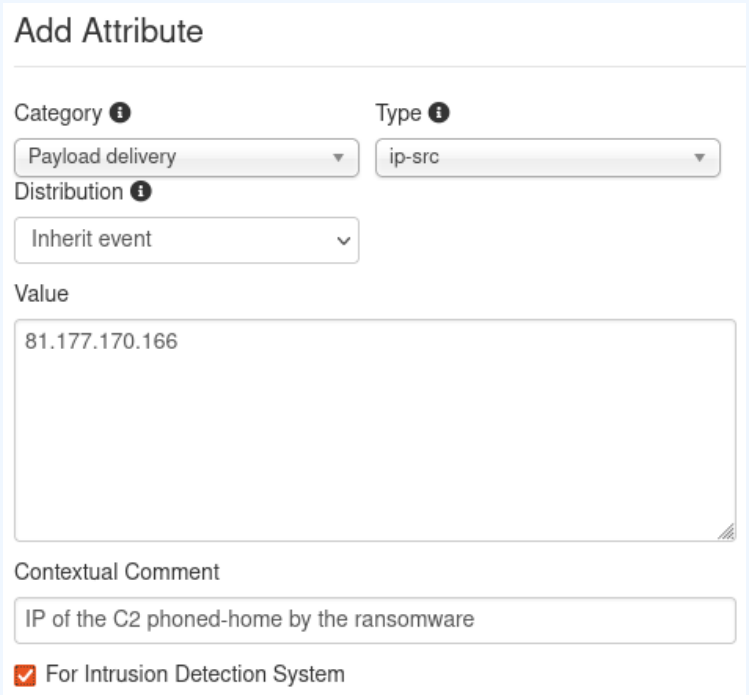
- Pick the `Payload Delivery` category
- Add contextual comment such as:
  - `CryptoLocker ransomware delivered by email`
- Check *Is a malware sample*



Case study 2: Ransomware

▶ Encode the C2's IP address

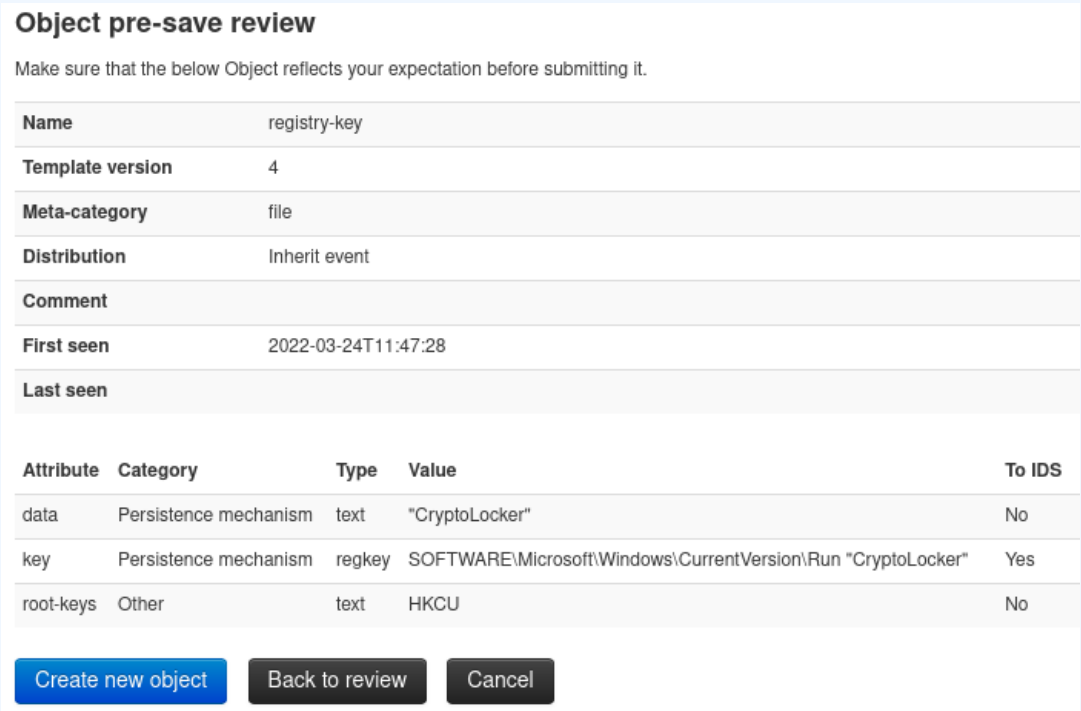
- Create an *attribute* and pick the `Payload Installation` category and `ip-src` type
- Check the `For Intrusion Detection System`
- Add a contextual comment such as
  - `IP address of the scammer collected from the RDP log file`



Case study 2: Ransomware

▶ Encode the registry keys used for persistence

- As the registry keys contains multiple contextually linked data points, we should use an `registry-key` *object*
- Add a contextual comment such as
  - `The registry key used for persistence, making sure it gets run again after an OS reboot`



Object pre-save review

Make sure that the below Object reflects your expectation before submitting it.

Name	registry-key
Template version	4
Meta-category	file
Distribution	Inherit event
Comment	
First seen	2022-03-24T11:47:28
Last seen	

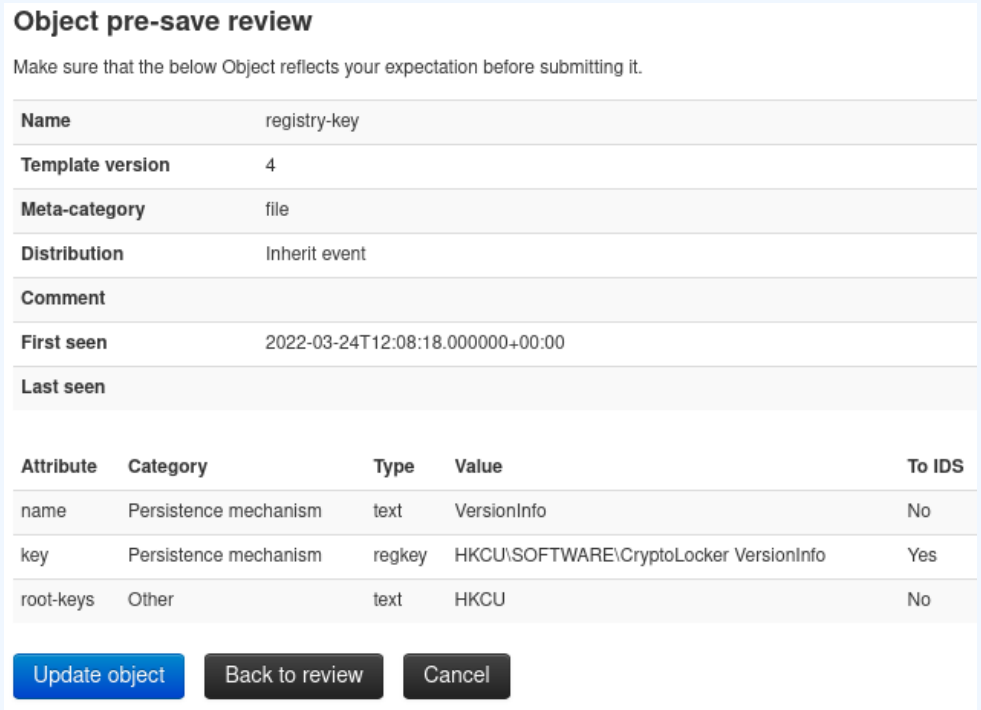
Attribute	Category	Type	Value	To IDS
data	Persistence mechanism	text	"CryptoLocker"	No
key	Persistence mechanism	regkey	SOFTWARE\Microsoft\Windows\CurrentVersion\Run "CryptoLocker"	Yes
root-keys	Other	text	HKCU	No

Create new object | Back to review | Cancel

Case study 2: Ransomware

▶ Encode the registry keys used for storing the configuration

- As the registry keys contains multiple contextually linked data points, we should use an `registry-key` *object*
- Add a contextual comment such as
  - `Containing configuration data (C2 address, malware version and installation timestamp)`



Object pre-save review

Make sure that the below Object reflects your expectation before submitting it.

Name	registry-key
Template version	4
Meta-category	file
Distribution	Inherit event
Comment	
First seen	2022-03-24T12:08:18.000000+00:00
Last seen	

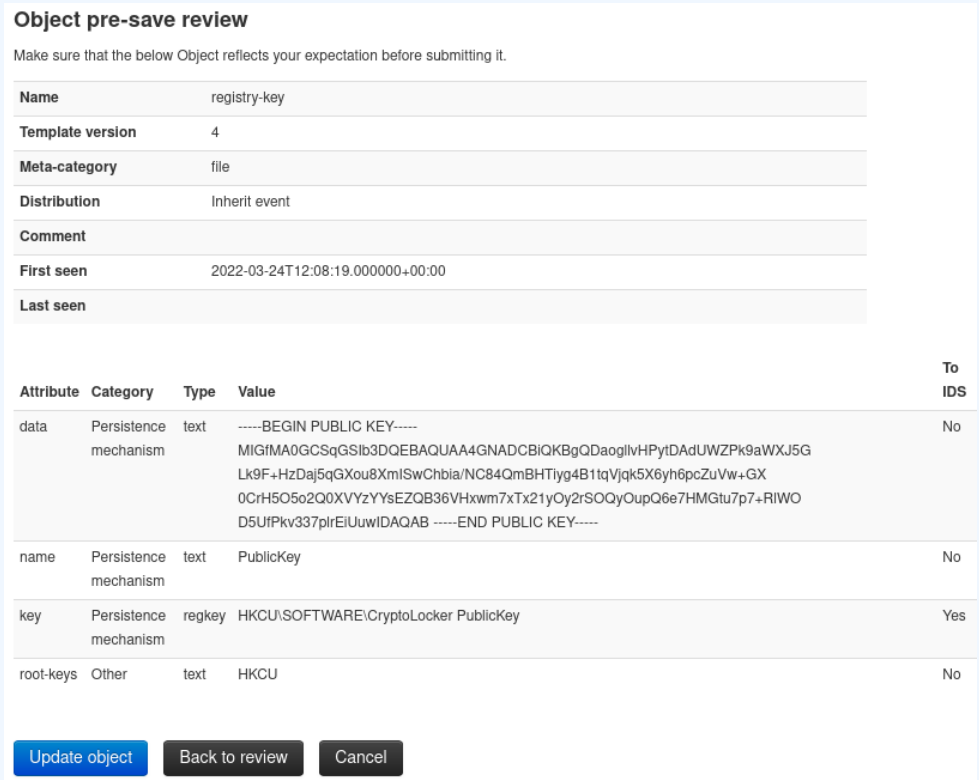
Attribute	Category	Type	Value	To IDS
name	Persistence mechanism	text	VersionInfo	No
key	Persistence mechanism	regkey	HKCU\SOFTWARE\CryptoLocker VersionInfo	Yes
root-keys	Other	text	HKCU	No

Update object | Back to review | Cancel

Case study 2: Ransomware

▶ Encode the registry keys used for storing the PK

- As the registry keys contains multiple contextually linked data points, we should use an `registry-key` *object*
- Add a contextual comment such as
  - `Contains the RSA public key received from the C2 used for encryption`



Object pre-save review

Make sure that the below Object reflects your expectation before submitting it.

Name	registry-key
Template version	4
Meta-category	file
Distribution	Inherit event
Comment	
First seen	2022-03-24T12:08:19.000000+00:00
Last seen	

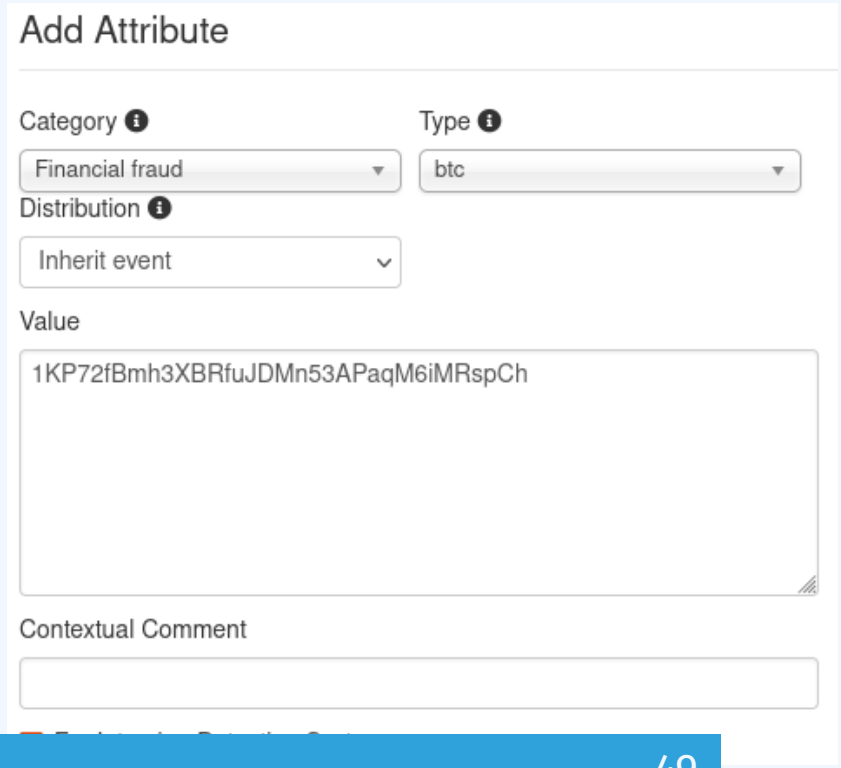
Attribute	Category	Type	Value	To IDS
data	Persistence mechanism	text	-----BEGIN PUBLIC KEY----- MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQDaoglIvHPytDAdUWZPk9aWXJ5G Lk9F+HzDaj5qGXou8XmISwChbia/NC84QmBHTiyg4B1tqVjqk5X6yh6pcZuVw+GX 0CrH5O5o2Q0XVYzYYsEZQB36VHxwm7xTx21yOy2rSOQyOupQ6e7HMGtu7p7+RIWO D5UfPkv337plrEiUuwIDAQAB -----END PUBLIC KEY-----	No
name	Persistence mechanism	text	PublicKey	No
key	Persistence mechanism	regkey	HKCU\SOFTWARE\CryptoLocker PublicKey	Yes
root-keys	Other	text	HKCU	No

Update object | Back to review | Cancel

Case study 2: Ransomware

▶ Encode the bitcoin address used to reveive the ransom

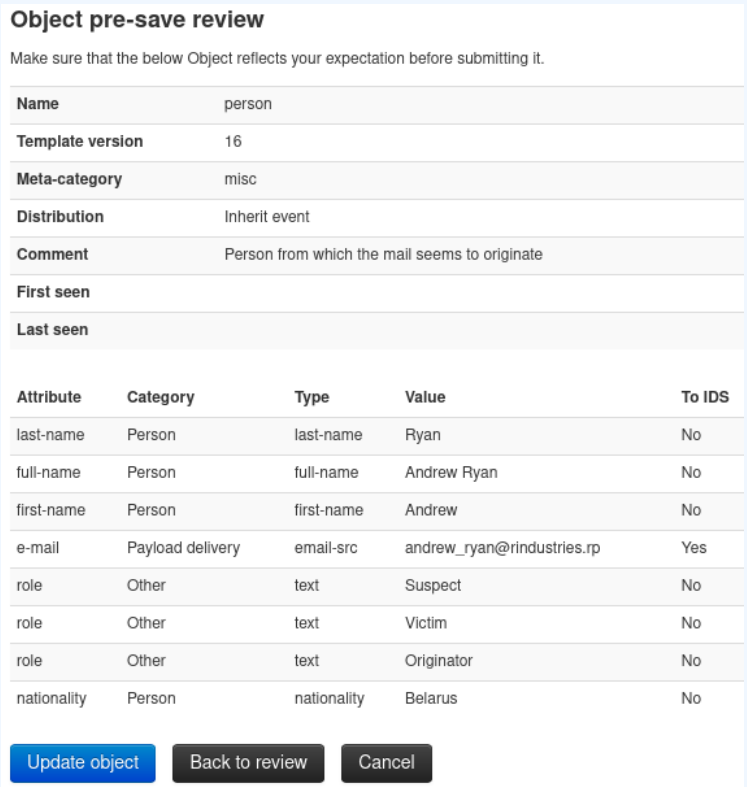
- Create an *attribute* and pick the `Financial Fraud` category and `btc` `type`
- Check the `For Intrusion Detection System`
- Add a contextual comment such as
  - `Hardcoded address on which the ransom is asked to be transfered`



Case study 2: Ransomware

▶ Encode the name and roles of the person

- As these attributes are contextually linked between each others, we should use a `person` *object*
- Add a contextual comment such as
  - `Person from which the mail seems to originate`
- Include at least: `full-name`, `e-mail` and `roles`



Object pre-save review

Make sure that the below Object reflects your expectation before submitting it.

Name	person
Template version	16
Meta-category	misc
Distribution	Inherit event
Comment	Person from which the mail seems to originate
First seen	
Last seen	

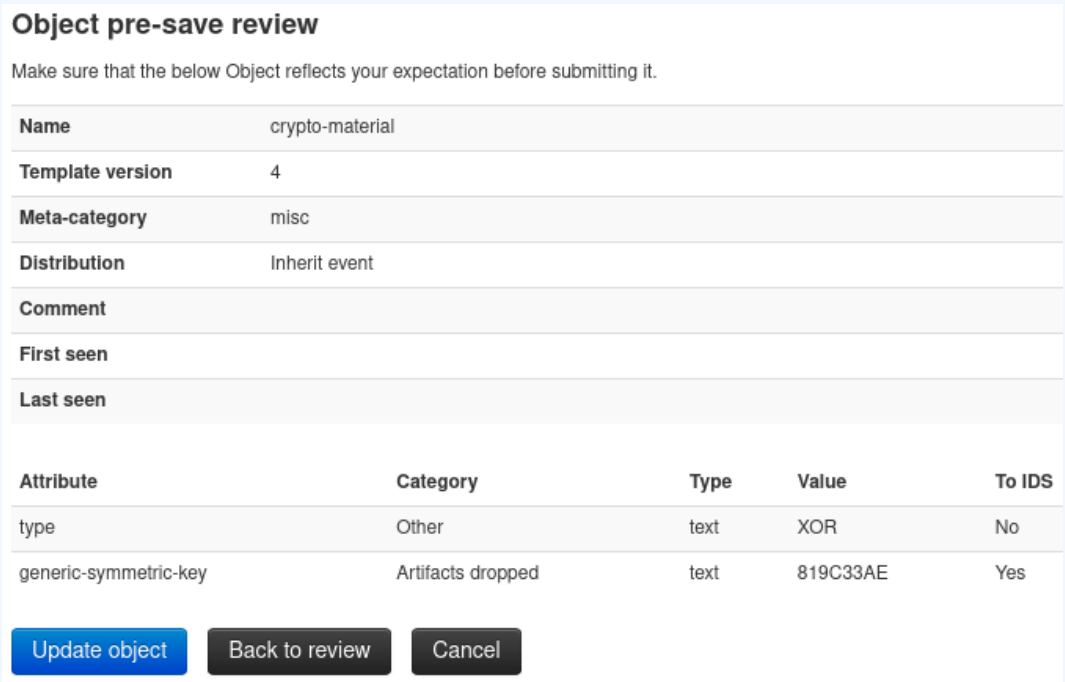
Attribute	Category	Type	Value	To IDS
last-name	Person	last-name	Ryan	No
full-name	Person	full-name	Andrew Ryan	No
first-name	Person	first-name	Andrew	No
e-mail	Payload delivery	email-src	andrew_ryan@rindustries.rp	Yes
role	Other	text	Suspect	No
role	Other	text	Victim	No
role	Other	text	Originator	No
nationality	Person	nationality	Belarus	No

Update object | Back to review | Cancel

Case study 2: Ransomware

▶ Encode the XOR key

- As these attributes are contextually linked between each others, we should use a `crypto-material` *object*
- Add a contextual comment such as
  - `XOR key used to encode the malware's configuration in the registry`
- Include at least: `type` and `generic-symmetric-key`



Object pre-save review

Make sure that the below Object reflects your expectation before submitting it.

Name	crypto-material
Template version	4
Meta-category	misc
Distribution	Inherit event
Comment	
First seen	
Last seen	

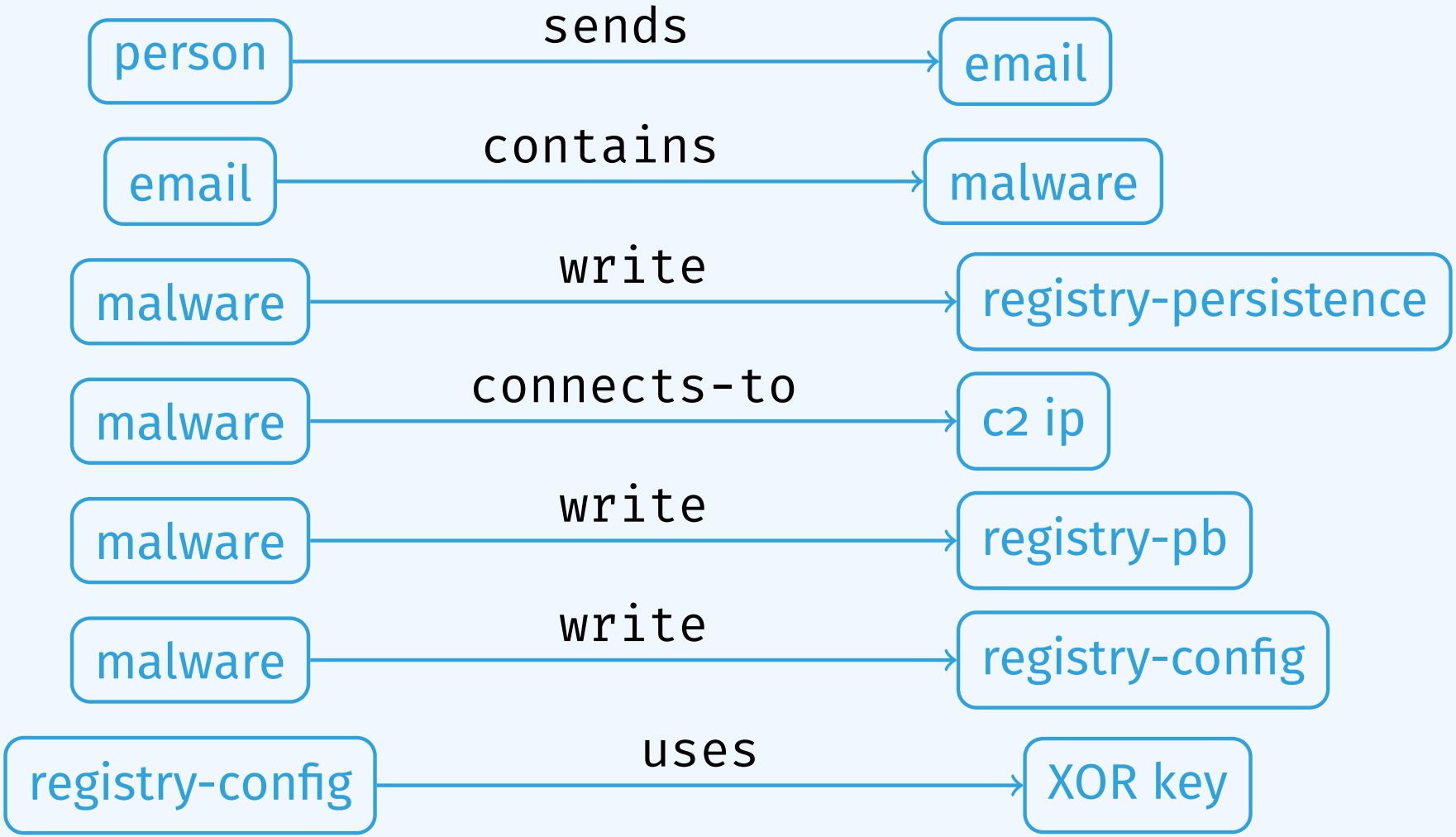
Attribute	Category	Type	Value	To IDS
type	Other	text	XOR	No
generic-symmetric-key	Artifacts dropped	text	819C33AE	Yes

Update object | Back to review | Cancel

Case study 2: Ransomware

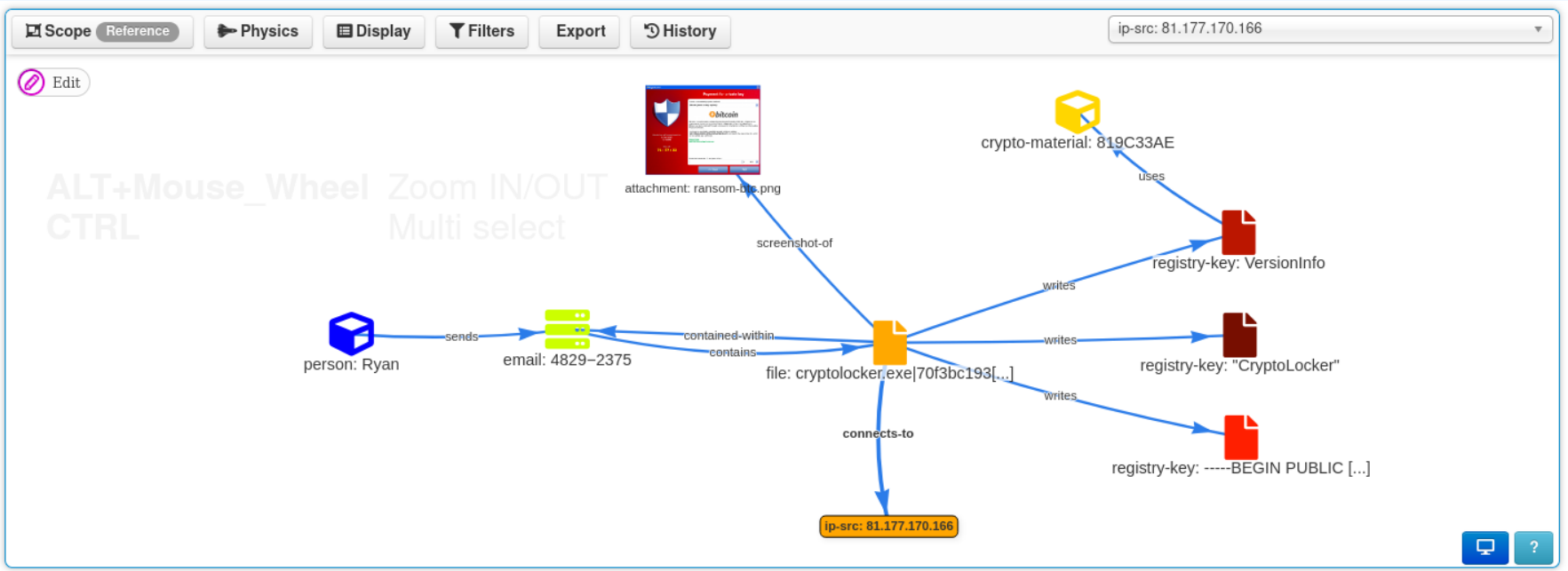
▶ Creating relationships

Add (at least) these relationships to recreate the story



Case study 2: Ransomware

▶ Creating relationships

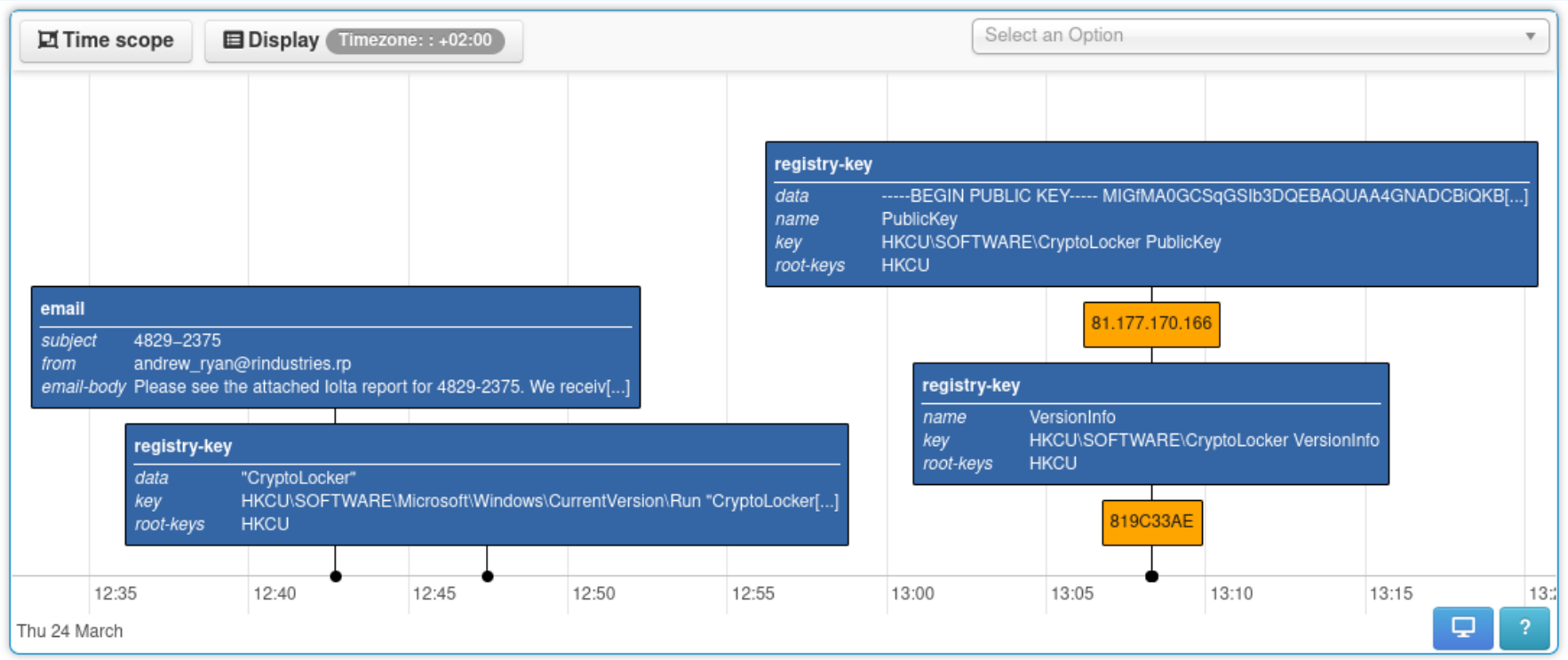


Case study 2: Ransomware

▶ Adding time component

The time component is useful to recreate the chronology

- Main focus is the Cyber Threat Intelligence (CTI) aspect



Case study 2: Ransomware

▶ Perform enrichments

- IP address to get its location

Mmdb Lookup:

Object: geolocation

country	Russia
countrycode	RU
latitude	60
longitude	100
text	db_source: GeoOpen-Country. build_db: 2022-02-05 10:37:33. Latitude and longitude are country average.

Object: geolocation

country	Russia
countrycode	RU
latitude	60
longitude	100
text	db_source: GeoOpen-Country-ASN. build_db: 2022-02-06 09:30:25. Latitude and longitude are country average.

Object: asn

asn	8342

db: 2022-02-06 09:30:25.

Case study 2: Ransomware

▶ Perform enrichments

- Bitcoin wallet to view the transactions

Btc Steroids:

Address: 1KP72fBmh3XBRfuJDMn53APaqM6iMRspCh
Balance: 0.0000000000 BTC (+54.9083000000 BTC / -54.9083000000 BTC)
Transactions: 40

#40	19 Nov 2013 12:03:48 UTC	-0.00020000 BTC	0.13 USD	0.10 EUR
#39	15 Oct 2013 15:16:44 UTC	-2.00000000 BTC	316.18 USD	227.78 EUR
#39	15 Oct 2013 15:16:44 UTC	-1.99950000 BTC	316.10 USD	227.72 EUR
#39	Sum:	-3.99950000 BTC	632.28 USD	455.50 EUR
#38	15 Oct 2013 02:12:02 UTC	-2.00000000 BTC	316.18 USD	227.78 EUR
#37	13 Oct 2013 21:03:42 UTC	-2.00000000 BTC	295.06 USD	211.26 EUR
#36	11 Oct 2013 21:23:33 UTC	-2.00000000 BTC	280.20 USD	204.02 EUR
#36	11 Oct 2013 21:23:33 UTC	-2.00000000 BTC	280.20 USD	204.02 EUR
#36	Sum:	-4.00000000 BTC	560.40 USD	408.04 EUR
#35	08 Oct 2013 23:24:22 UTC	-2.00000000 BTC	272.98 USD	199.28 EUR
#35	08 Oct 2013 23:24:22 UTC	-2.00000000 BTC	272.98 USD	199.28 EUR
#35	Sum:	-4.00000000 BTC	545.96 USD	398.56 EUR
#34	07 Oct 2013 08:26:25 UTC	-2.00000000 BTC	271.60 USD	198.90 EUR
#34	07 Oct 2013 08:26:25 UTC	-2.00000000 BTC	271.60 USD	198.90 EUR
#34	07 Oct 2013 08:26:25 UTC	-2.00000000 BTC	271.60 USD	198.90 EUR
#34	07 Oct 2013 08:26:25 UTC	-2.00000000 BTC	271.60 USD	198.90 EUR
#34	Sum:	-8.00000000 BTC	1086.40 USD	795.60 EUR

Case study 2: Ransomware

▶ Contextualizing the data with *Taxonomies*

- Different country / sectors might use different nomemclature
- Suggestions of taxonomies for tagging:
  - `adversary`: adversary infrastructure
  - `circl`: Classification in Incident Response
  - `enisa`: ENISA structuring aid for information and threats
  - `europol-*`: Describe the type of events or incidents
  - `maec-*`: Malware Attribute Enumeration and Characterization
  - `malware_classification`: Based on SANS malware 101
  - `ms-caro-malware`: Microsoft's Malware Type and Platform
  - `ransomware`: ransomware types and the elements
  - `veris`: Vocabulary for Event Recording and Incident Sharing
  - `collaborative-intelligence`: Support analysts
  - `workflow`: Support analysts
  - `tlp`: Traffic Light Protocol

Case study 2: Ransomware

▶ Contextualizing the data with *Taxonomies*

- Incident type
  - `circl:incident-classification="ransomware"`
  - `enisa:nefarious-activity-abuse="ransomware"`
  - `europol-incident:malware="infection"`
  - `europol-incident:malware="c&c"`
  - `ms-caro-malware:malware-type="Ransom"`
- Malware type
  - `malware_classification:malware-category="Ransomware"`
  - `ransomware:type="crypto-ransomware"`
- Collaration and Sharing
  - `collaborative-intelligence:request="extracted-malware-config"`
  - `workflow:state="complete"`
  - `tlp:green`

Case study 2: Ransomware

▶ Contextualizing the data with *Taxonomies*

- Infection vector
  - `europol-event:dissemination-malware-email`
  - `maec-delivery-vectors:maec-delivery-vector="email-attachment"`
  - `ransomware:infection="phishing-e=mails"`
- Adversary infrastructure
  - `adversary:infrastructure-type="c2"`
  - `veris:action:malware:variety="C2"`

Malware-specific information

- maec-malware-capabilities:maec-malware-capability="fraud"
- maec-malware-capabilities:maec-malware-capability="persistence"
- maec-malware-capabilities:maec-malware-capability="communicate-with-c2-server"
- maec-malware-capabilities:maec-malware-capability="compromise-data-availability"
- ransomware:element="ransomnote"
- ransomware:element="dropper"
- ransomware:complexity-level="file-restoration-possible-using-shadow-volume-copies"
- ransomware:complexity-level="file-restoration-possible-using-backups"
- ransomware:complexity-level="decryption-key-recovered-from-a-C&C-server-or-network-communications"
- ransomware:complexity-level="encryption-model-is-seemingly-flawless"
- ransomware:purpose="deployed-as-ransomware-extortion"
- ransomware:target="pc-workstation"
- ransomware:communication="dga-based"
- ransomware:malicious-action="asymmetric-key-encryption"

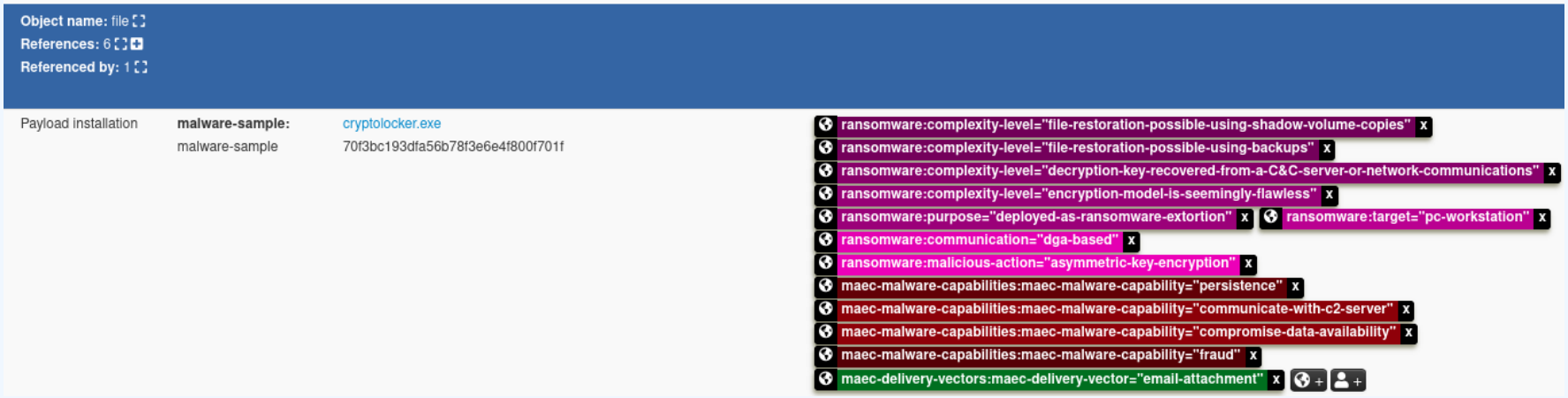
Case study 2: Ransomware

▶ Contextualizing the data with *Taxonomies*

- Danger of over-classification
  - Make things cluttered and unreadable
  - Mixing classification scheme
  - Introduce a non-negligible overhead when using *LIKE* filters (e.g. `tlp:%`)

Case study 2: Ransomware

▶ Contextualizing the data with *Taxonomies*



- Depending on the community, being complete on the contextualization can be useful for metrics and trends

Case study 2: Ransomware

▶ Contextualizing the data with *Taxonomies*

- Adding tags on attribute level make the role of the data clearer
- Make searches and exports easier

☐	2022-03-29	Payload delivery	ip-src	81.177.170.166	adversary:infrastructure-type="c2" x veris:action:malware:variety="C2" x
☐	2022-03-29	Artifacts dropped	attachment		ransomware:element="ransomnote" x

	2022-03-29	**Object name:** email **References:** 1 **Referenced by:** 2			
☐	2022-03-28	Payload delivery	**subject:** email-subject	4829–2375	
☐	2022-03-28	Payload delivery	**from:** email-src	andrew_ryan@rindustries.rp	
☐	2022-03-29	Payload delivery	**email-body:** email-body	Please see the attached Iolta report for 4829-2375. We received a check request in the amount of $19,637.28 for the above referenced file. However, the attached report refects a $0 balance. At your earliest convenience, please advise how this request is to be funded. Thanks. Andrew Ryan *	ransomware:element="dropper" x

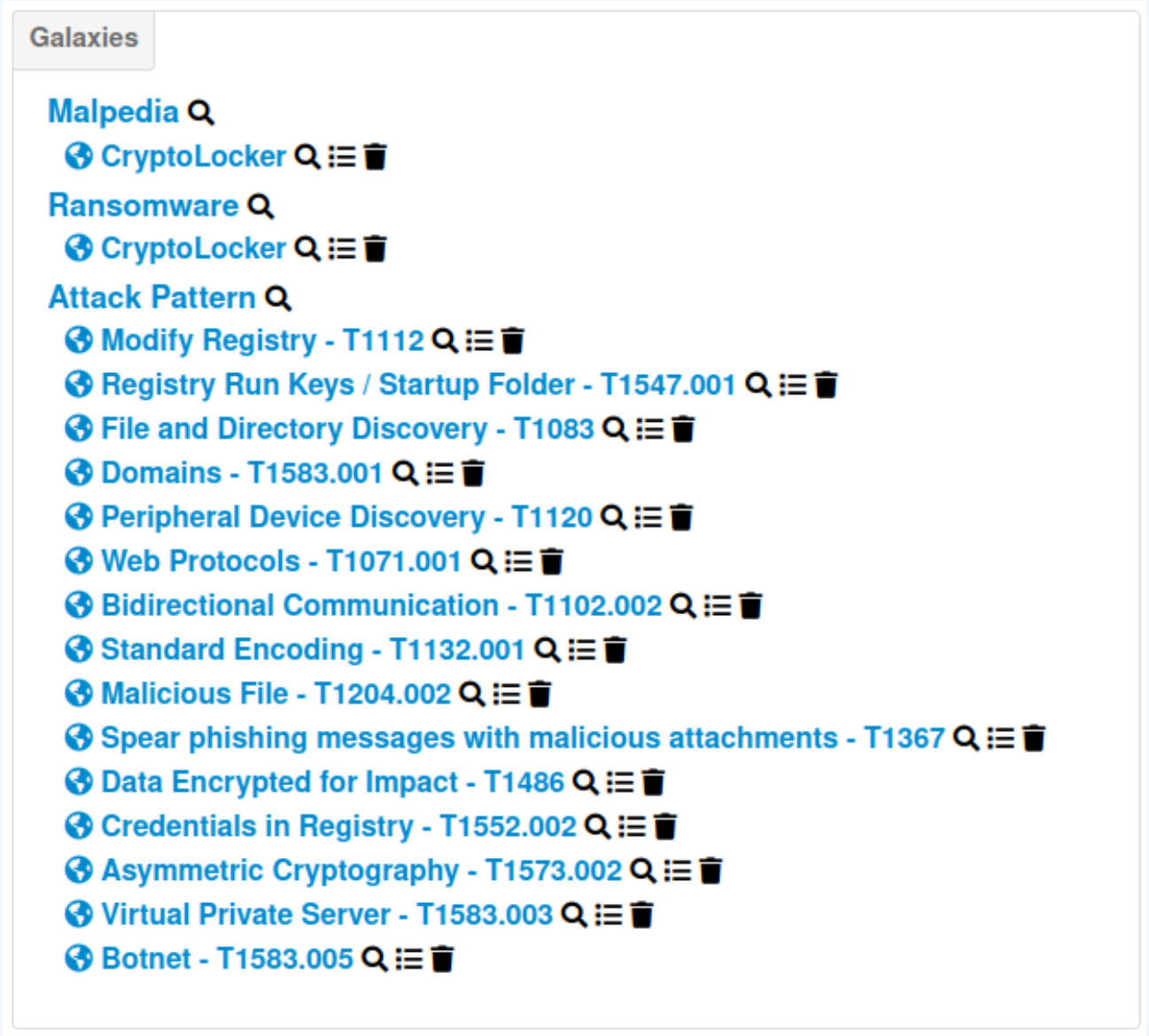
Case study 2: Ransomware

▶ Contextualizing the data with *Galaxy Clusters*

- Note: Different country / sectors might use different nomemclature
- Suggestions for tagging with Galaxies:
  - Malpedia
  - Ransomware
  - MITRE Att&ck Pattern
  - Preventive Measure

Case study 2: Ransomware

▶ Contextualizing the data with *Galaxy Clusters*



Case study 2: Ransomware

▶ Contextualizing the data with *Galaxy Clusters*

MITRE ATT&CK Matrix

mitre-mobile-attack | **mitre-attack** | mitre-pre-attack

0 1 Show all

Initial access (19 items)	Execution (39 items)	Persistence (114 items)	Privilege escalation (101 items)	Defense evasion (169 items)	Credential access (56 items)	Discovery (42 items)	Lateral movement (23 items)	Collection (38 items)	Command and control (40 items)	Exfiltration (17 items)	Impact (26 items)
Cloud Accounts	Malicious File	Registry Run Keys / Startup Folder	Registry Run Keys / Startup Folder	Modify Registry	Credentials in Registry	File and Directory Discovery	Application Access Token	ARP Cache Poisoning	Asymmetric Cryptography	Automated Exfiltration	Data Encrypted for Impact
Compromise Hardware Supply Chain	AppleScript	.bash_profile and .bashrc	.bash_profile and .bashrc	Abuse Elevation Control Mechanism	/etc/passwd and /etc/shadow	Peripheral Device Discovery	Component Object Model and Distributed COM	Adversary-in-the-Middle	Bidirectional Communication	Data Transfer Size Limits	Account Access Removal
Compromise Software Dependencies and Development Tools	At (Linux)	Accessibility Features	Abuse Elevation Control Mechanism	Access Token Manipulation	ARP Cache Poisoning	Account Discovery	Distributed Component Object Model	Archive Collected Data	Standard Encoding	Exfiltration Over Alternative Protocol	Application Exhaustion Flood
Compromise Software Supply Chain	At (Windows)	Account Manipulation	Access Token Manipulation	Application Access Token	AS-REP Roasting	Application Window Discovery	Exploitation of Remote Services	Archive via Custom Method	Web Protocols	Exfiltration Over Asymmetric Encrypted Non-C2 Protocol	Application or System Exploitation
Default Accounts	Command and Scripting Interpreter	Active Setup	Accessibility Features	Asynchronous Procedure Call	Adversary-in-the-Middle	Browser Bookmark Discovery	Internal Spearphishing	Archive via Library	Application Layer Protocol	Exfiltration Over Bluetooth	Data Destruction
Domain Accounts	Component Object Model	Add Office 365 Global Administrator Role	Active Setup	BITS Jobs	Bash History	Cloud Account	Lateral Tool Transfer	Archive via Utility	Commonly Used Port	Exfiltration Over C2 Channel	Data Manipulation
Drive-by Compromise	Component Object Model and Distributed COM	Add-ins	AppCert DLLs	Binary Padding	Brute Force	Cloud Groups	Pass the Hash	Audio Capture	Communication Through Removable Media	Exfiltration Over Other Network Medium	Defacement
Exploit Public-Facing Application	Container Administration Command	Additional Cloud Credentials	AppInit DLLs	Bootkit	Cached Domain Credentials	Cloud Infrastructure Discovery	Pass the Ticket	Automated Collection	DNS	Exfiltration Over Physical Medium	Direct Network Flood
External Remote Services	Container Orchestration Job	AppCert DLLs	Application Shimming	Build Image on Host	Cloud Instance Metadata API	Cloud Service Dashboard	RDP Hijacking	Browser Session Hijacking	DNS Calculation	Exfiltration Over Symmetric Encrypted Non-C2 Protocol	Disk Content Wipe
Hardware Additions	Cron	AppInit DLLs	Asynchronous Procedure Call	Bypass User Account Control	Container API	Cloud Service Discovery	Remote Desktop Protocol	Clipboard Data	Data Encoding	Exfiltration Over Unencrypted/Obfuscated Non-C2 Protocol	Disk Structure Wipe

Case study 2: Ransomware

▶ Mitigations and Detection

Thanks to the MITRE Att&ck contextualization, we can derive preventive measures from their catalogue.
Just to name a few

- Mitigations
  - Restrict Registry Permissions
  - Antivirus/Antimalware
  - Network Intrusion Prevention
  - Restrict Web-Based Content
  - Software Configuration
- Detection
  - Application Log
  - Command
  - Network Traffic
  - Process
  - Windows Registry

Case study 2: Ransomware

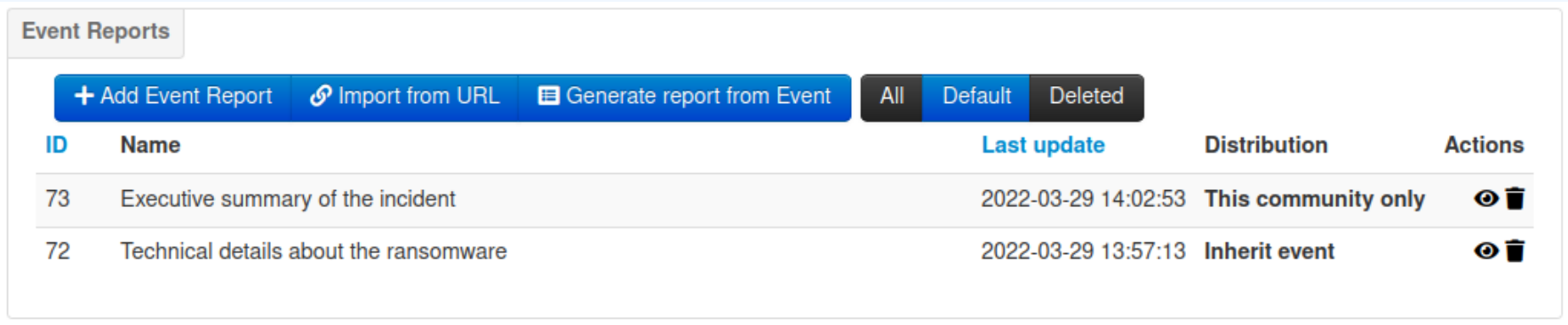
▶ Write-up with an *event report*

- Create the *event report* with a concise name
- Example: Executive summary of the case
  - Leave its content empty as it can be edited with more ease in the editor afterward
- Write a summary with
  - Quick chronology
  - Written explanation of the steps tooks by the ransomware
  - Reference to existing *attributes* or *objects* whenever possible
    - The special syntax is: `@[scope]{uuid}`

Case study 2: Ransomware

▶ Write-up with an *event report*

- We could have one technical report and another report for the incident



Case study 2: Ransomware

▶ Write-up with an *event report* (technical)

Technical details about the ransomware

The ransomware in question seems to an early version of the `ransomware ↦ CryptoLocker` ransomware, or at least an extremely close version.

Infection vector

Distributed through spam or spearphishing emails. In this case, the mail `email` `Please see the attached Iolta report for 4829-2375....` was sent to lure the victim to read it and get infected. The ransomware payload `file` `cryptolocker.exe` was attached to the mail with a PDF icon and relied on the fact that Windows hides the extensions of known file to get the user to execute it once it's opened.

Execution and persistence

Cryptolocker hides its presence from the victims until it has successfully contacted the command and control (C2) server `ip-src` `81.177.170.166` . Prior to this action, the malware ensures its persistence by copying itself and adding an autorun registry key `registry-key` `"CryptoLocker"` . It also store additional configuration data such as the C2 address `ip-src` `81.177.170.166` , the malware version and installation timestamp in another registry key `registry-key` `VersionInfo` . This registry key is encoded with the key `crypto-material` `819C33AE` .

Network

The malware try to contacts the C2 server and once sucsessul recover the RSA public-key (generated by the C2) used to encrypt the files on the victim's computer.

Encryption

Once the malware has its public-key, it begins the encryption process by enumerating files and encrypting it. A small amout of metadata and the encrypted file contents are then written back to disk, replacing the original files. Encrypted files can only be recovered by obtaining the RSA private key held exclusively by the threat actors. After finishing the file encryption process, Cryptolocker displays a window containing instructions on how to decrypt the file by paying the ransom as seen in the picture below

Case study 2: Ransomware

► Write-up with an *event report* (technical)

Payment

The ransom amount is set to 2 BTC to be transfered on the bitcoin address **btc** 1KP72fBmh3XBRfuJDMn53APaqM6iMRspCh before the countdown timer expires. According to the ransomware, the private key associated to the public key is destroyed if the payement is not done, rendering the decryption of the files impossible.

Ransomware details

Delivery	maec-delivery-vectors:maec-delivery-vector="email-attachment"
Complexity Level	ransomware:complexity-level="file-restoration-possible-using-shadow-volume-copies"
	ransomware:complexity-level="file-restoration-possible-using-backups"
	ransomware:complexity-level="decryption-key-recovered-from-a-C&C-server-or-network-communications"
	ransomware:complexity-level="encryption-model-is-seemingly-flawless"
Purpose	ransomware:purpose="deployed-as-ransomware-extortion"
Malicious Action	ransomware:malicious-action="asymmetric-key-encryption"
Capability	maec-malware-capabilities:maec-malware-capability="persistence"
	maec-malware-capabilities:maec-malware-capability="fraud"
	maec-malware-capabilities:maec-malware-capability="communicate-with-c2-server"
	maec-malware-capabilities:maec-malware-capability="compromise-data-availability"

Mitigation

Techniques and MITRE ATT&CK

- mitre-attack-pattern ↦ Botnet - T1583.005
- mitre-attack-pattern ↦ Domains - T1583.001
- mitre-attack-pattern ↦ Virtual Private Server - T1583.003
- mitre-attack-pattern ↦ Spear phishing messages with malicious attachments - T1367
- mitre-attack-pattern ↦ Malicious File - T1204.002
- mitre-attack-pattern ↦ Registry Run Keys / Startup Folder - T1547.001
- mitre-attack-pattern ↦ Data Encrypted for Impact - T1486
- mitre-attack-pattern ↦ File and Directory Discovery - T1083
- mitre-attack-pattern ↦ Asymmetric Cryptography - T1573.002
- mitre-attack-pattern ↦ Bidirectional Communication - T1102.002
- mitre-attack-pattern ↦ Standard Encoding - T1132.001
- mitre-attack-pattern ↦ Web Protocols - T1071.001
- mitre-attack-pattern ↦ Credentials in Registry - T1552.002
- mitre-attack-pattern ↦ Modify Registry - T1112
- mitre-attack-pattern ↦ Peripheral Device Discovery - T1120

mitre-mobile-attack | **mitre-attack** | mitre-pre-attack

0 — 1 — Show all

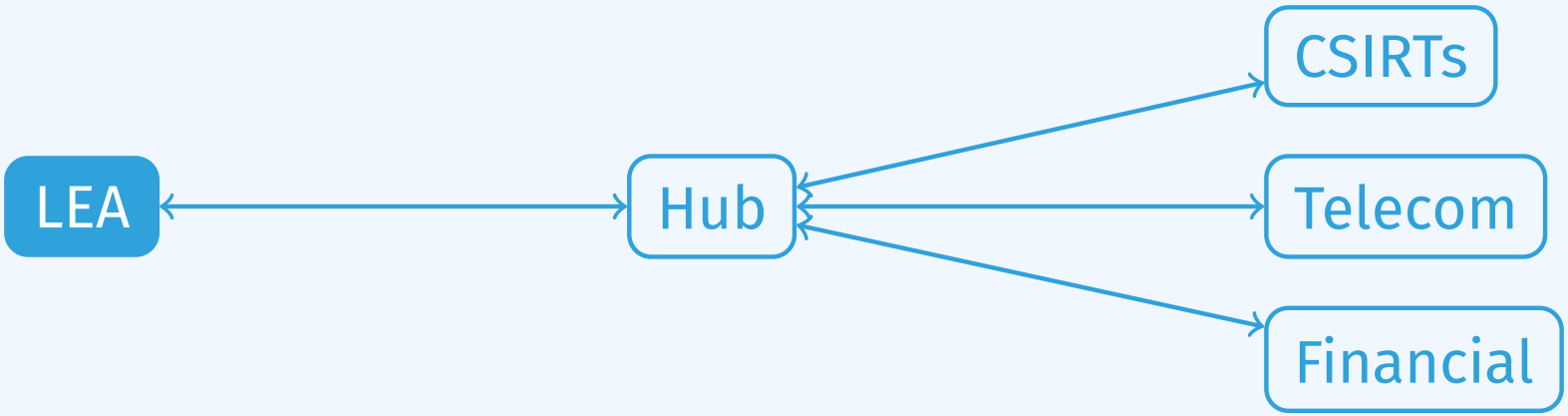
Initial access (19 items)	Execution (39 items)	Persistence (114 items)	Privilege escalation (101 items)	Defense evasion (169 items)	Credential access (56 items)	Discovery (42 items)	Lateral movement (23 items)	Collection (38 items)	Command and control (40 items)	Exfiltration (17 items)	Impact (26 items)
Cloud Accounts	Malicious File	Registry Run Keys / Startup Folder	Registry Run Keys / Startup Folder	Modify Registry	Credentials in Registry	File and Directory Discovery	Application Access Token	ARP Cache Poisoning	Asymmetric Cryptography	Automated Exfiltration	Data Encrypted for Impact
Compromise	AppleScript	.bash_profile	.bash_profile	Abuse Elevation	/etc/passwd and	Peripheral	Component	Adversary-in-	Bidirectional	Data Transfer	Account Access

Case study 2: Ransomware

▶ Review the distribution level and publish

In our case, we consider the following MISP network topology

- The current instance is owned and managed by a LEA
- The current instance is connected to a central MISP instance acting as a "Hub"
- The "Hub" is connected to various other MISP instances such as other LEAs, CSIRTs, Financial and telecom institutions



Case study 2: Ransomware

▶ Review the distribution level and publish

- `binary file`: **All communities**
- `C2 ip & geolocation`: **All communities**
- `crypto-material & registry-keys`: **All communities**
- `person`: **All communities**
  - Even though `Andrew Ryan` could be a victim due to impersonation, it's very likely that it's a fake name
  - The email address `andrew_ryan@rindustries.rp` should be considered as an IoC

→ **Publish the event!**