TheHive

Use Integrations to
Speed Up Incident Response

Nabil Adouani - Jérôme Leonard

Co-founders of TheHive-Project
Co-founders of StrangeBee

StrangeBee

Who we are ?

- Jérôme Leonard | Nabil Adouani
- Co-founders of TheHive Project & StrangeBee
  - Leader and designer of TheHive and Cortex
  - Provide professional support and services

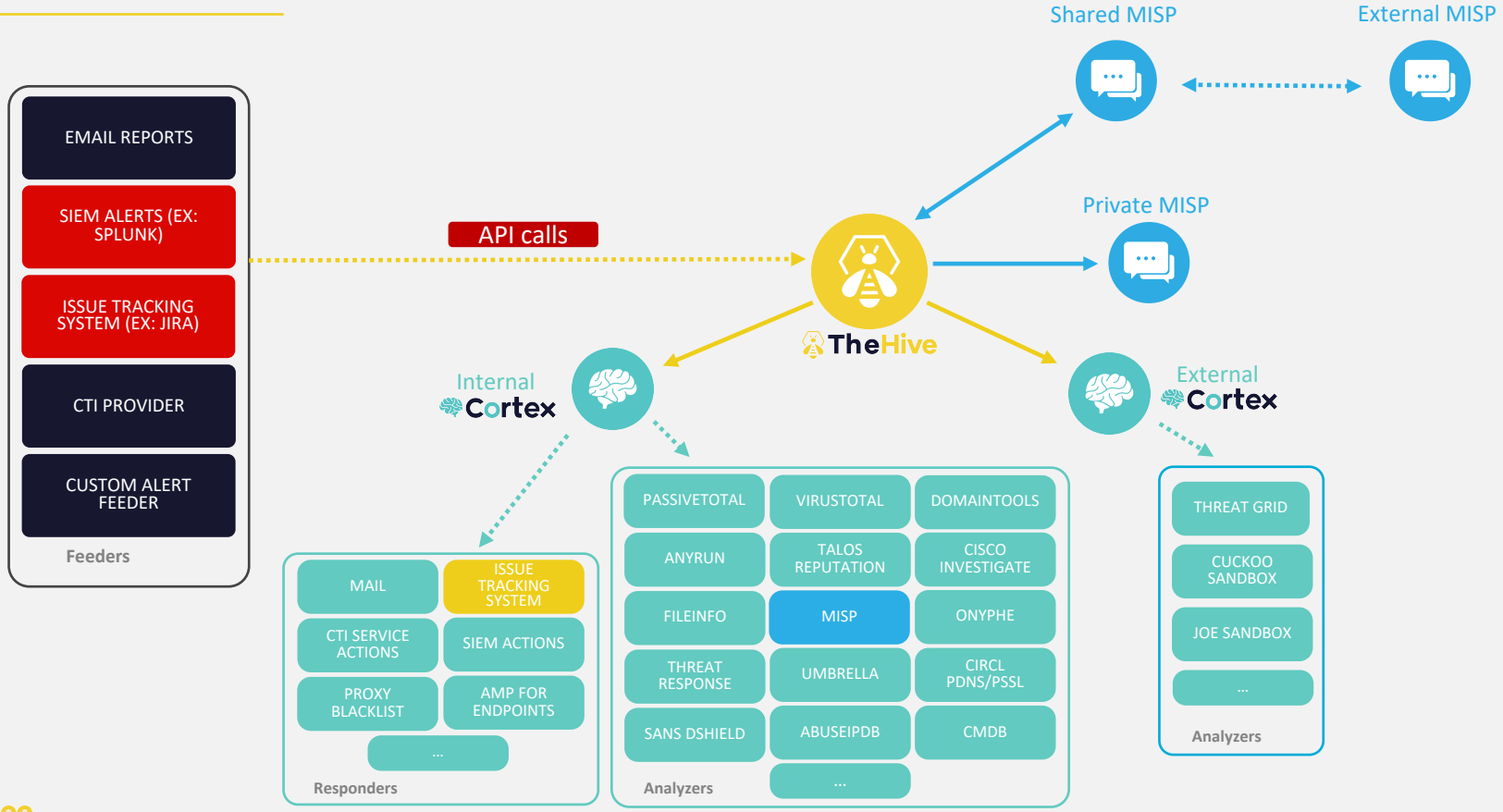
TheHive & Cortex

- (SIRP) Security Incident Response Platform
  - Handle incidents
  - Forensic Analysis
- Organise, structure, archive incidents
- Gather ***Alerts*** from many sources
- Implement IR process

- **Analysis** and **Response** engine
- Gather information and intelligence on observables with Analyzers
- Run active actions on your network or third-party services with Responders

StrangeBee

Integration



StrangeBee

Typical IR timeline

StrangeBee

Typical IR timeline

1

Create Alerts including useful data with ***Feeders***

StrangeBee

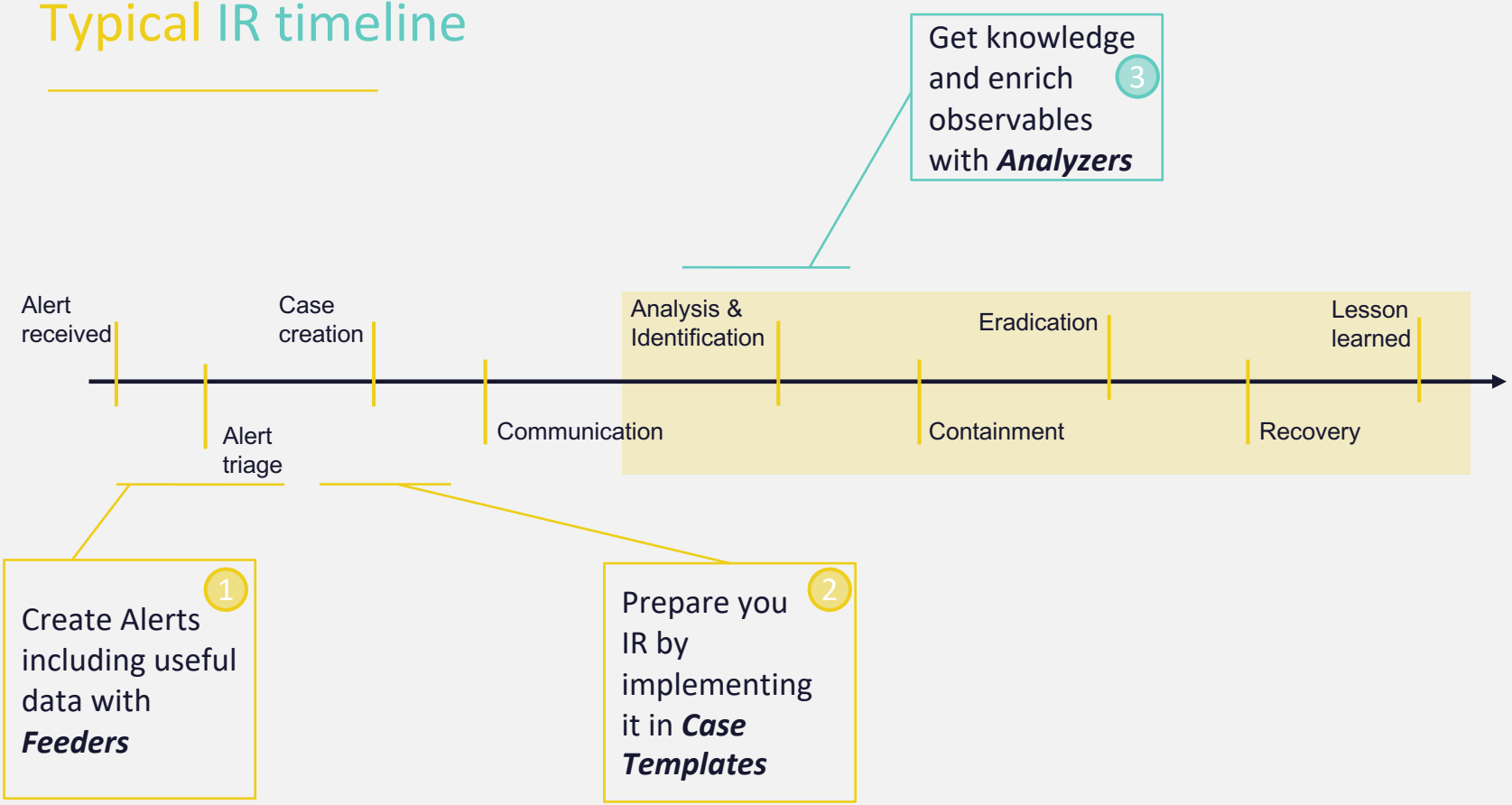
Typical IR timeline

1 Create Alerts including useful data with ***Feeders***

2 Prepare you IR by implementing it in ***Case Templates***

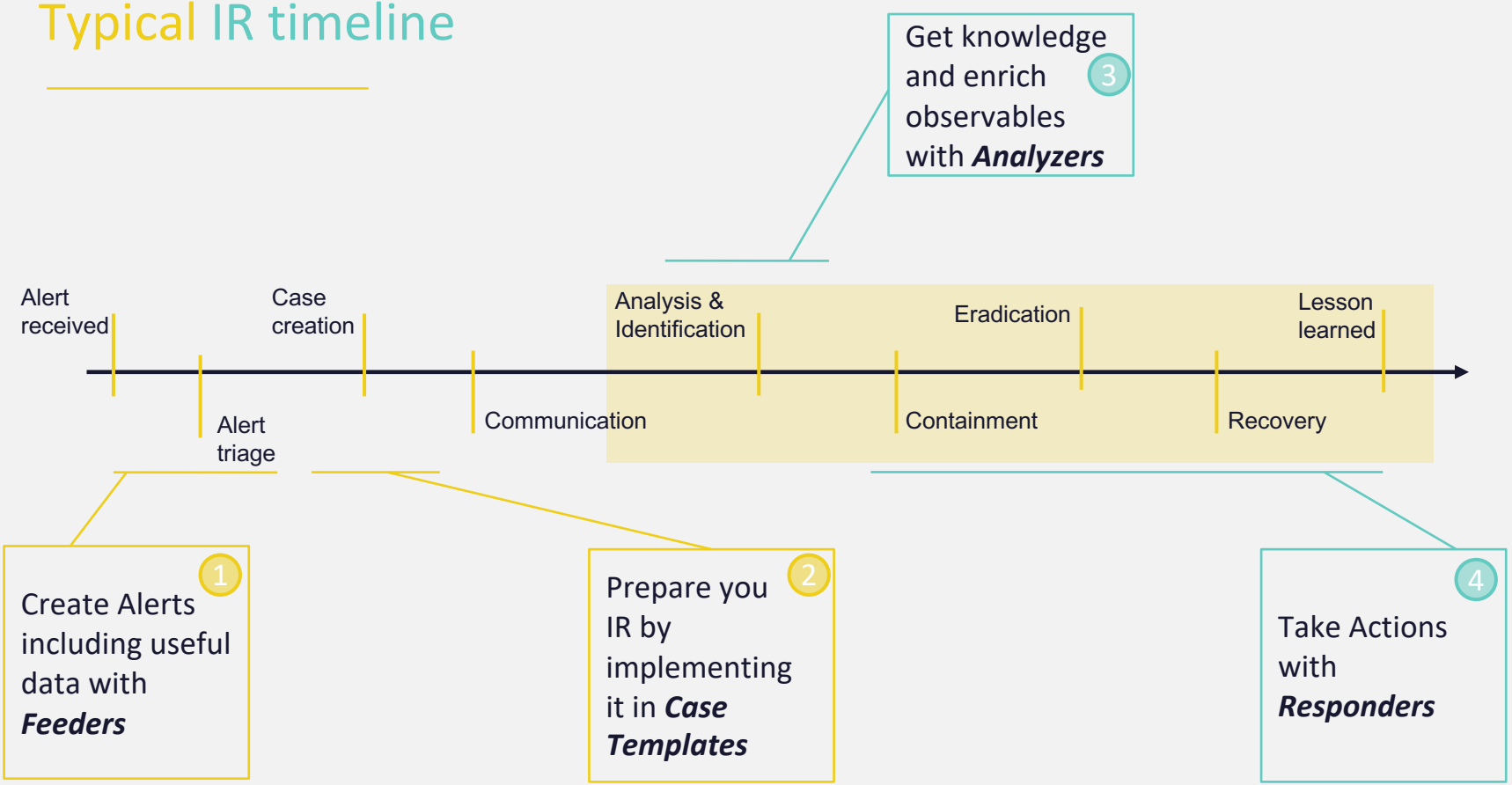
StrangeBee

Typical IR timeline



StrangeBee

Typical IR timeline



StrangeBee

Scenario: Investigate a malspam report

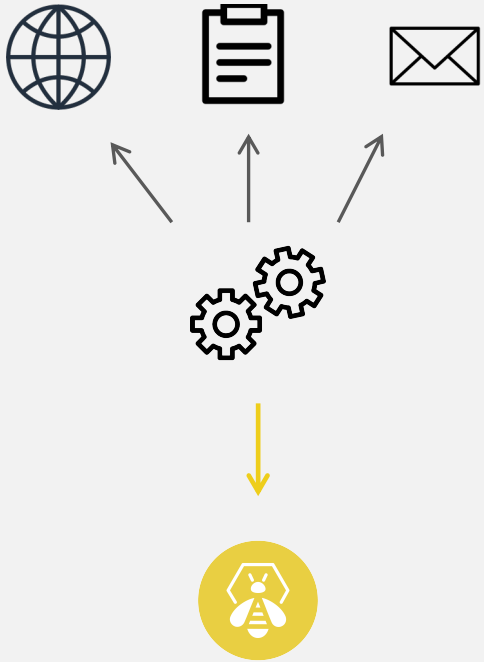
- A user received a suspicious message
- He used a button on his mail client to report it with comments to the SOC team
- An *Alert* is created automatically in TheHive with all interesting information

Receive ***Alerts*** in TheHive with *Feeders*

StrangeBee

Feeder

- Gather information from an external service
  - Mail server
  - CTI provider
  - SIEM …
- Process data and format for TheHive
  - TheHive uses Markdown text formating
- Import data as Case or Alert
- Python Client Library – TheHive4py



StrangeBee

Feeder : gather suspicious message forwarded by users

- Connect to csirt@ Mailbox and read mails with eml attachments
- Gather useful information from the ***raw*** message

```
            if '.eml' in att.filename:
                ## save suspicious mail content in file
                with open(os.path.join(TMP_PATH, att.filename), "w+b") as eml_file:
                    eml_file.write(att.payload)
                eml_file.close()

                ## gather useful information from suspicious mail content that can be directly added to
observables
                suspiciousSubject = MailMessage.from_bytes(att.payload).subject
                suspiciousSender = MailMessage.from_bytes(att.payload).from_
                suspiciousBody = MailMessage.from_bytes(att.payload).text
```

StrangeBee

Feeder : Process and format for TheHive

- Create ***Observables*** list by adding gathered data from suspicious message
- Create ***Observables*** with source email (*eml file*) and additional attachments

```
## Add this data as observables
artifacts.append(AlertArtifact(dataType='mail',
data=suspiciousSender,
tags=["suspicious_mail:src_addr"]
    )
)

artifacts.append(AlertArtifact(
    dataType='mail-subject',
    data=suspiciousSubject,
    tags=["suspicious_mail:mail-subject"]
    )
)
```

StrangeBee

Feeder : Process and format for TheHive

- Create ***Observables*** list by adding gathered data from suspicious message
- Create ***Observables*** with source email (*eml file*) and additional attachments

```
## Add this data as observables
artifacts.append(AlertArtifact(dataType='mail',
data=suspiciousSender,
tags=["suspicious_mail:src_addr"]
    )
)

artifacts.append(AlertArtifact(
    dataType='mail-subject',
    data=suspiciousSubject,
    tags=["suspicious_mail:mail-subject"]
    )
)
```

```
## Create file observables
for f in os.listdir(TMP_PATH):
    if os.path.isfile(os.path.join(TMP_PATH, f)) and ".eml" in f:
        artifacts.append(AlertArtifact(
            dataType='file',
            data=os.path.join(TMP_PATH, f),
            tags=["suspicious_mail:email_source"]
            )
        )
for att in os.listdir(ATT_PATH):
    if os.path.isfile(os.path.join(ATT_PATH, att)):
        artifacts.append(AlertArtifact(
            dataType='file',
            data=os.path.join(ATT_PATH, att),
            tags=["suspicious_mail:email_attachment"]
            )
        )
```

StrangeBee

Feeder : Create the *Alert* in TheHive

- Prepare the ***Alert*** with thehive4py library *Alert* model
- Create the ***Alert*** in TheHive

```
## Prepare the alert
if suspiciousSubject:
    alertTitle = suspiciousSubject
else:
    alertTitle = reportedSubject
alert = Alert(title=alertTitle,
        tlp=2,
        tags=['suspicious email', 'submittedBy:{}'.format(reportedBy)],
        description=reportedText,
        type='email report',
        customFields= CFs,
        source='Internal mail Server',
        sourceRef=sourceRef,
        caseTemplate=caseTemplate,
        artifacts=artifacts)
```

StrangeBee

Feeder : Create the *Alert* in TheHive

- Prepare the ***Alert*** with thehive4py library *Alert* model
- Create the ***Alert*** in TheHive

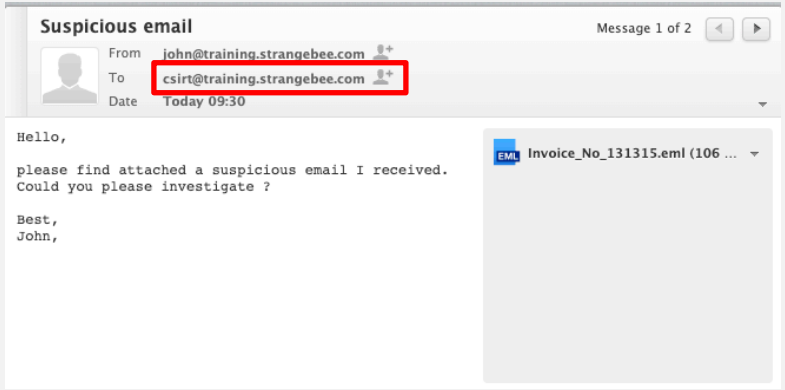
```
## Prepare the alert
if suspiciousSubject:
    alertTitle = suspiciousSubject
else:
    alertTitle = reportedSubject
alert = Alert(title=alertTitle,
        tlp=2,
        tags=['suspicious email', 'submittedBy:{}'.format(reportedBy)],
        description=reportedText,
        type='email report',
        customFields= CFs,
        source='Internal mail Server',
        sourceRef=sourceRef,
        caseTemplate=caseTemplate,
        artifacts=artifacts)
```

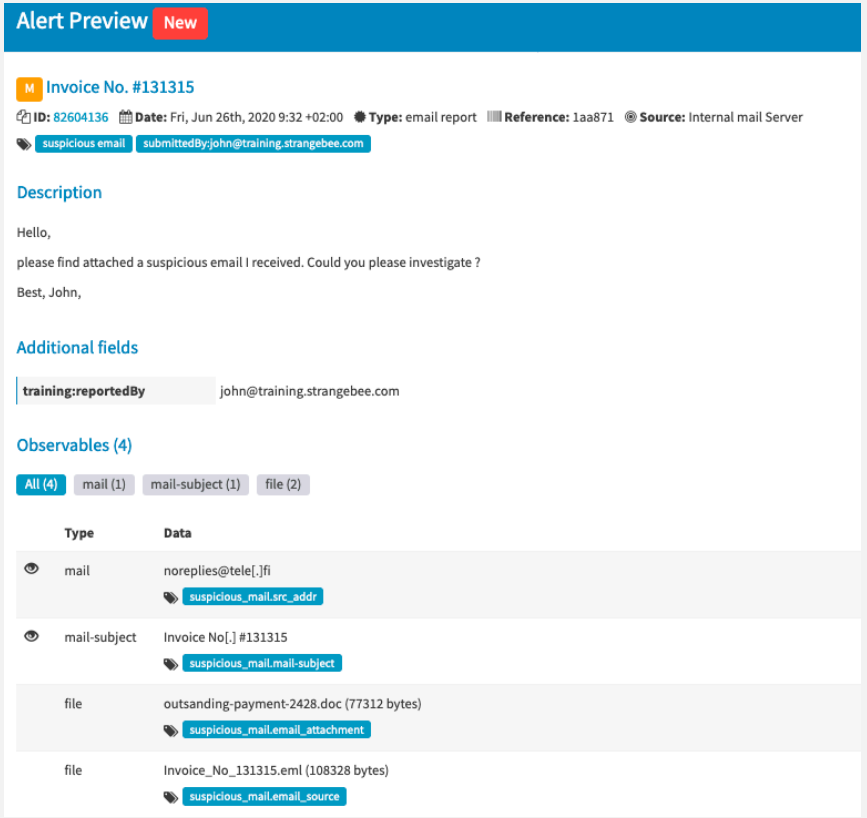
```
# Create the Alert
api = TheHiveApi( THEHIVE_URL ,APIKEY, organisation=ORG)
response = api.create_alert(alert)
```

StrangeBee

Hand's On

- Gather reports of suspicous messages receives by internal users, and generate Alerts





StrangeBee

Implement IR process with *Case Templates*

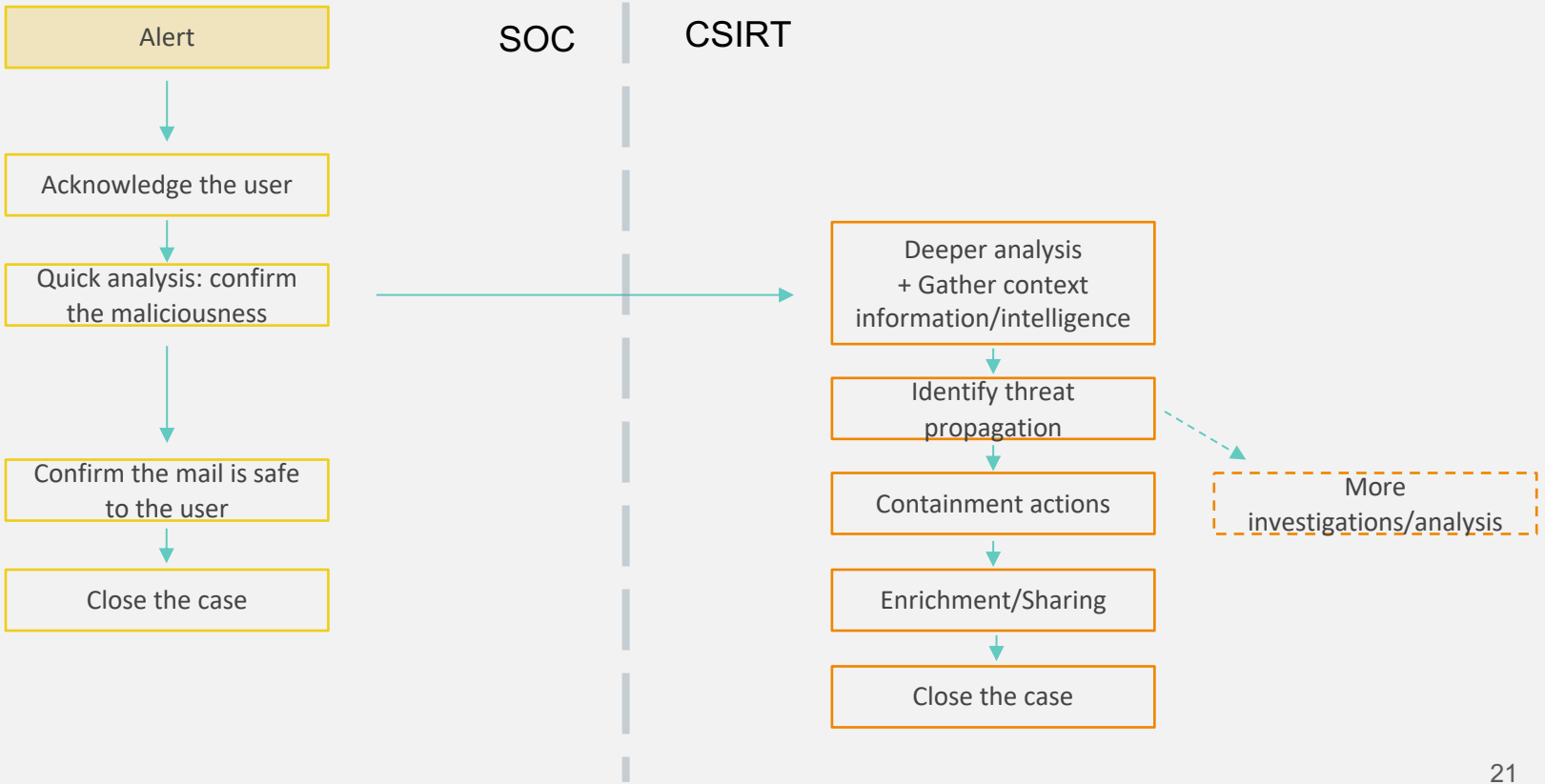
StrangeBee

Organisation

- SOC Team
  - In charge of reviewing and confirm that something malicious or fraudulent is happening
  - Then, escalate to CSIRT Team *or* Close the case
- CSIRT Team
  - Incident Handler
  - In charge of deeper analysis when *Cases* need more expertise
  - In charge of *Forensics analysis and Incident Response*

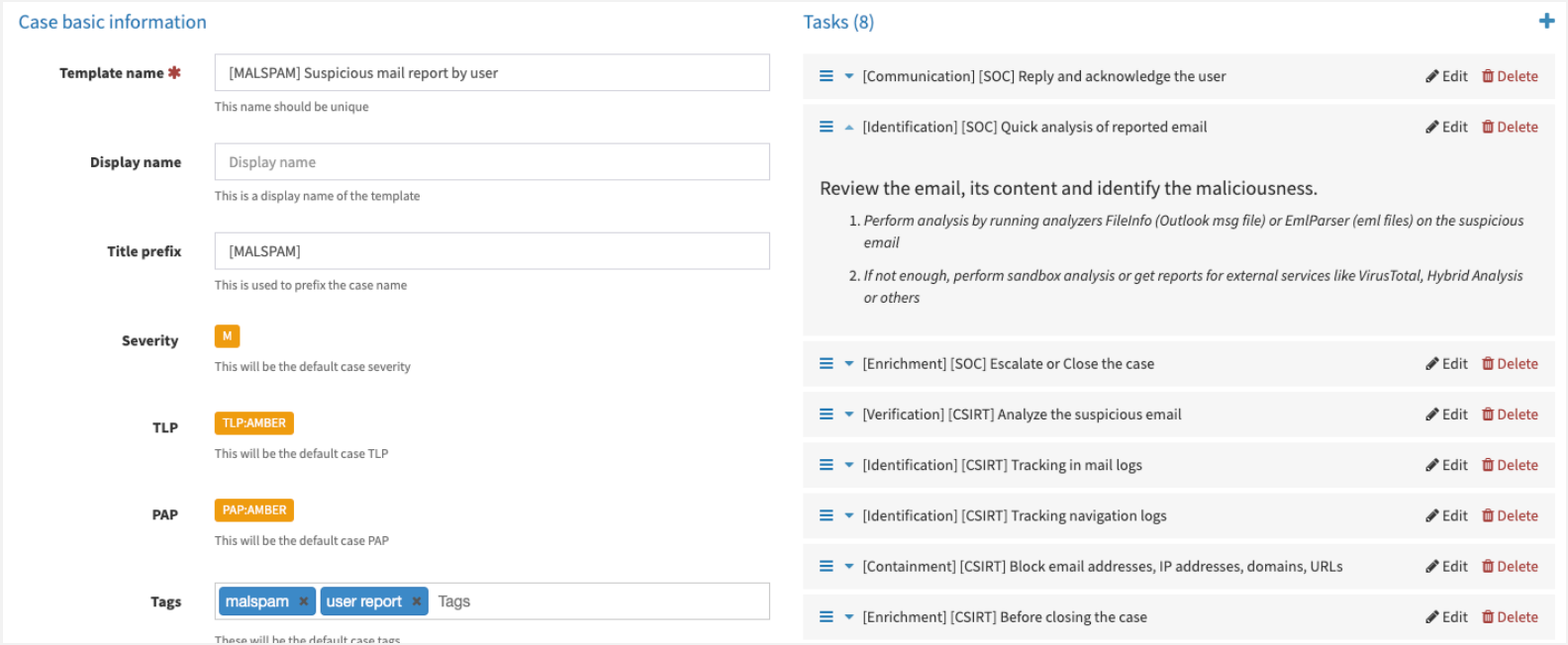
StrangeBee

Incident Response process : Investigate a malspam report



Hand’s On

- Create a ***Case template*** *for « Suspicious mail reports »*



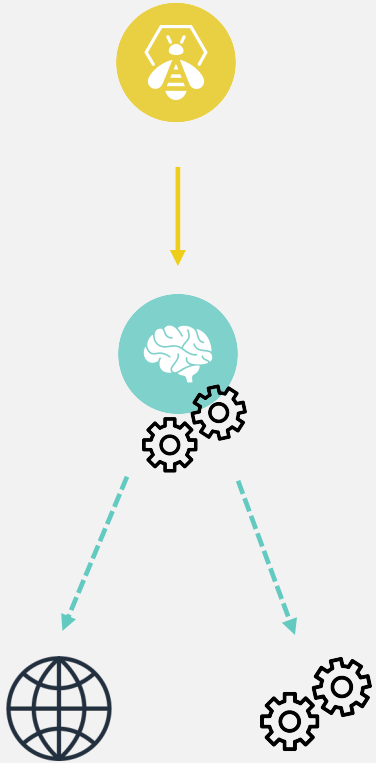
StrangeBee

Enrich your Case by running ***Analyzers*** on observables

StrangeBee

Analyzers

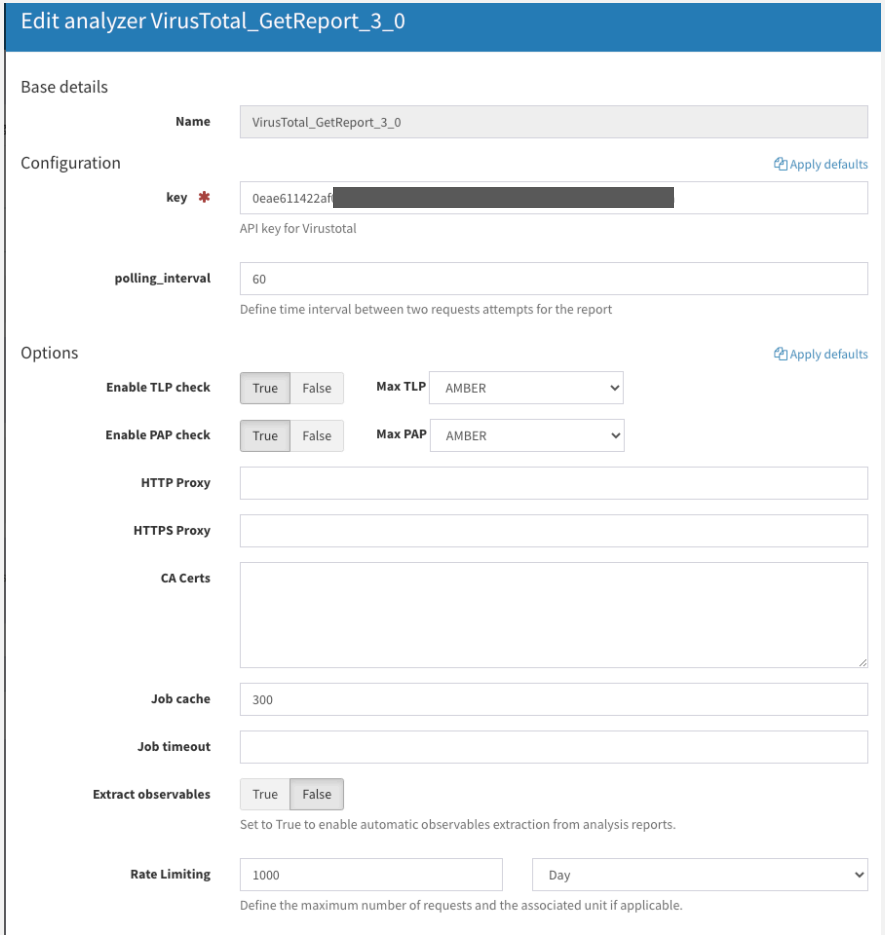
- Program processing 1 ***Observable*** and delivering reports
  - Query external services or run other programs
- *Input*: Observable metadata and configuration
- *Output*:
  - Summary report
  - Long report
  - Observables *(optional)*



StrangeBee

Analyzers: Configuration in Cortex

- Enable desired Analyzers in *Cortex*
- Customise configuration if needed
  - Takes care of TLP/PAP
  - Define Rate limiting



StrangeBee

Hand’s on: Run Analyzers from TheHive

- Different analyzers for different dataTypes

[URL]: *hxxps://www[.]sbsandco[.]com/files/SMSCount&Invoice[.]rar*

VT:GetReport="2/66" | PhishTank:In_Database="False" | TG:Analysis="56" | TG:Analysis="56" | TR:Talos Intelligence="Unknown"

Analysis

Run all

Analyzer	Last analysis	Actions
Abuse_Finder_3_0	*None*	
PhishTank_CheckURL_2_1	✔ Tue, May 26th, 2020 15:44 +02:00 (Cortex Prod)	
ThreatGrid_1_0	*None*	
ThreatResponse_1_0	✔ Thu, Jul 2nd, 2020 12:14 +02:00 (Cortex Prod)	
VirusTotal_GetReport_3_0	✔ Tue, May 26th, 2020 15:44 +02:00 (Cortex Prod)	

StrangeBee

Hand’s on: Review reports and extract new observables

- EmlParser analyzer on ***raw*** mail message

Email message details

From	ravibusinessclub@gmail.com ()
To	john.smith@strangebee.com (john.smith@strangebee.com)
Subject	[Avis Business Club] Booking Confirmation Email
Topic	-
Bcc	-
Attachments	This message file includes 1 attachment

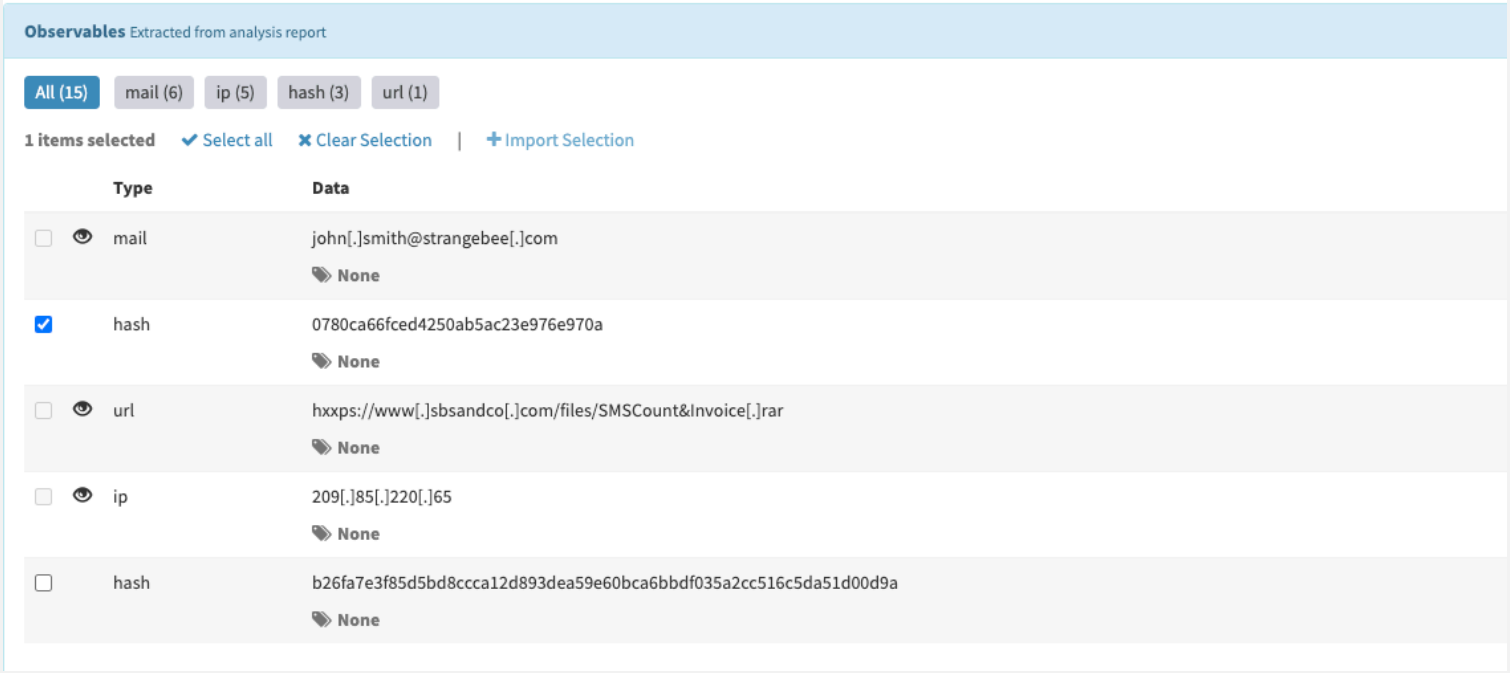
Filename	File information
AvisBusinessClub-Invoice-LKM55I.doc	[MD5]: 0780ca66fced4250ab5ac23e976e970a
	[SHA1]: 4f56abfb593661d757b094fb44d9600a37d4554c
	[SHA256]: b26fa7e3f85d5bd8ccca12d893dea59e60bca6bbdf035a2cc516c5da51d00d9a
	Composite Document File V2 Document, Little Endian, Os: Windows, Version 6.1, Code page: 1252, Title: Lalypyshowukyhu, Subject: Laly, Author: Lalypy, Template: Normal.dotm, Revision Number: 1, Name of Creating Application: Microsoft Office Word, Create Time/Date: Thu May 3 07:14:00 2018, Last Saved Time/Date: Thu May 3 07:14:00 2018, Number of Pages: 1, Number of Words: 0, Number of Characters: 1, Security: 0

Headers

```
Delivered-To: john.smith@strangebee.com
Received: by 2002:a0c:ab48:0:0:0:0:0 with SMTP id i8csp2090114qvb;
        Tue, 16 Oct 2018 09:52:22 -0700 (PDT)
X-Received: by 2002:a17:906:5249:: with SMTP id y9-v6mr24642939ejm.139.1539708741979;
        Tue, 16 Oct 2018 09:52:21 -0700 (PDT)
ARC-Seal: i=1; a=rsa-sha256; t=1539708741; cv=none;
        d=google.com; s=arc-20160816;
        b=FX/Fl6WP9Vw66muOi0Op6Et09QZA/nx5jnqYQmOKKe2BBq0V9jfJxn/wd0Tvazj+gF
         4DmNIJswptPQquOftg1TTvrh2SP4jPtBLIKAz/9vuJkRzVFXr7zx44AVR6ObkKgLuW6K
```

Hand’s on: Review reports and extract new observables

- EmlParser analyzer on ***raw*** mail message – extracted observables

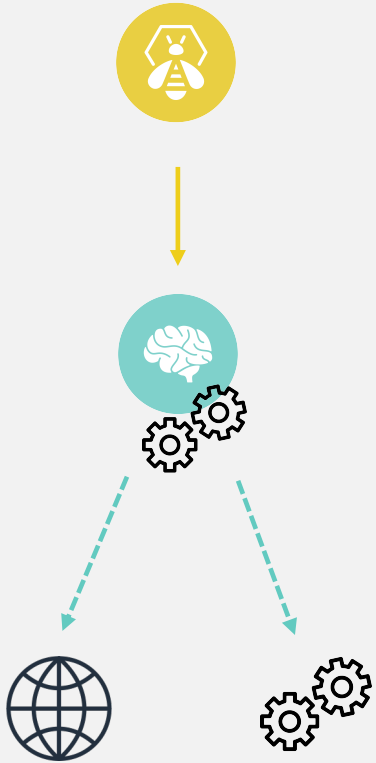


Take actions by using ***Responders***

StrangeBee

Responders

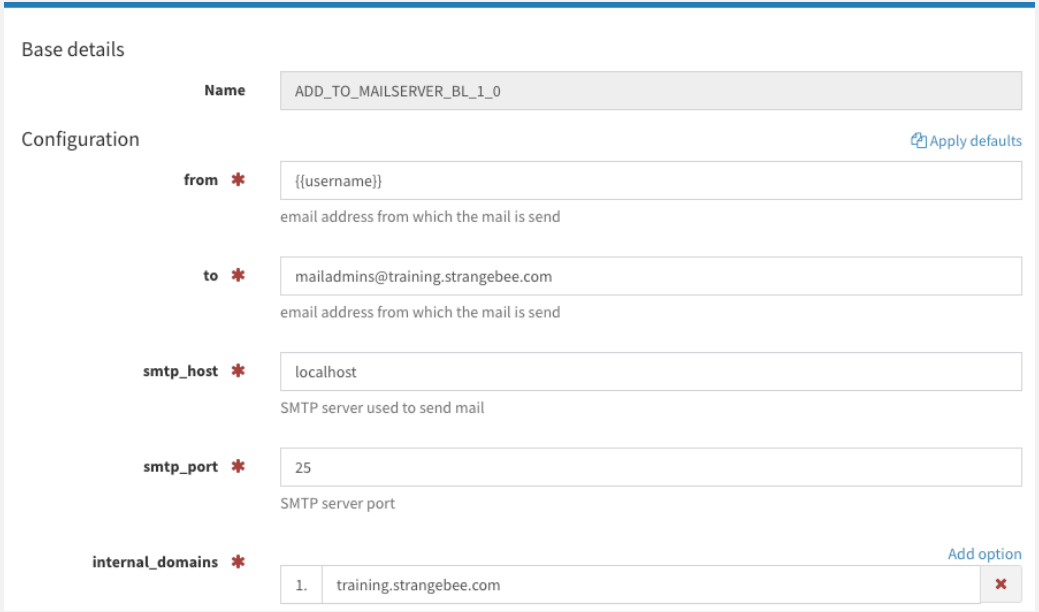
- Program processing *Alert, Case, Task, Log, Observable*
  - « *Create a ticket in ticketing system application from a Case or task* »
  - « *Add observable to Blacklist* »
- *Input: Data and metadata*
- *Output 1:* actions result
  - Success or Failure
- *Output 2*: run ***operations*** back in TheHive
  - « Add tags to Case/Observable
  - « Add/Close a Task or a log »



StrangeBee

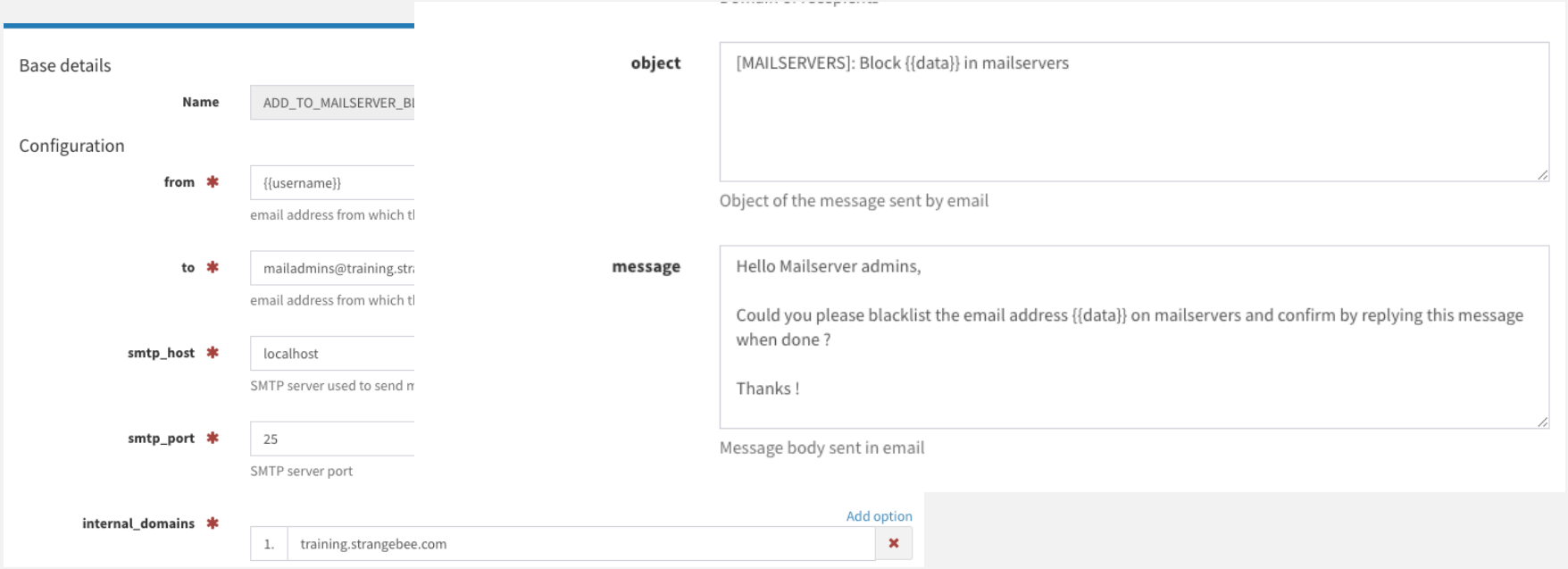
Responder: Configuration in Cortex

- Enable desired Responders in *Cortex*
- Customise configuration if needed



Responder: Configuration in Cortex

- Enable desired Responders in *Cortex*
- Customise configuration if needed



StrangeBee

Hand’s on: Ask for containment actions

- Contact external team and ask for containment actions

StrangeBee

Orchestration using TheHive

StrangeBee

TheHive vs Orchestration/Automation platform

- TheHive can export all events to a webhook endpoint
- TheHive offers the options needed to orchestrate and automate workflows
  - **REST APIs** to execute any action on TheHive/Cortex
  - Trigger a **Webhook call** on any change made on TheHive

StrangeBee

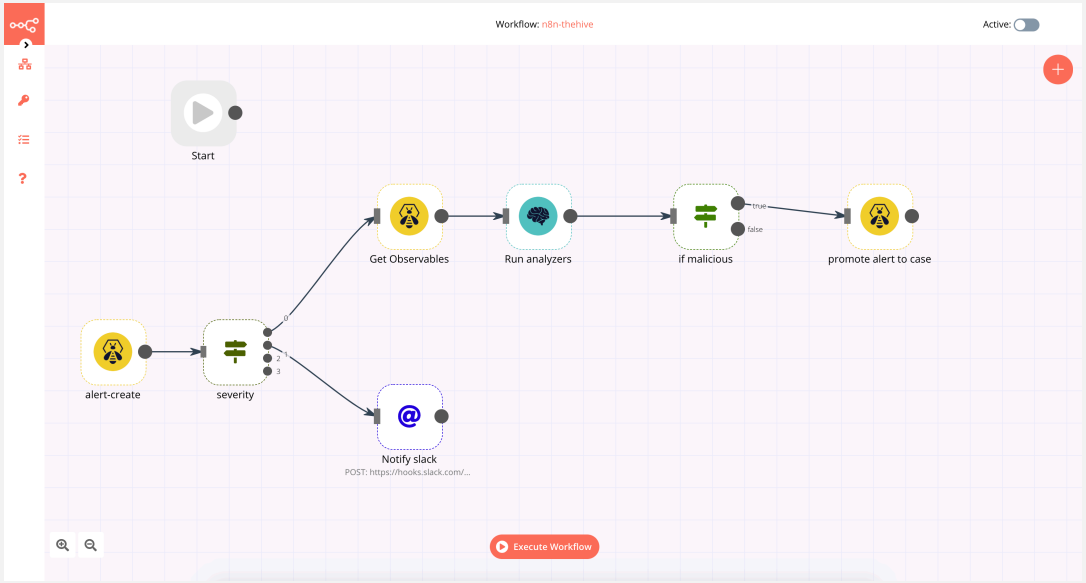
TheHive vs Orchestration/Automation platform

- TheHive and Cortex can be used by existing automation platforms and workflow editors:
  - Open Source tools

    Node Red

    Shuffle

    n8n
  - Commercial tools



Thank You

Community

- https://blog.thehive-project.org
- https://gitter.im/TheHive-Project/TheHive
- https://twitter.com/TheHive_Project
- users@thehive-project.org

- https://www.strangebee.com
- contact@strangebee.com

StrangeBee