SOAR-EDR HOME LAB

🛡️ SOAR–EDR Automated Incident Response Playbook

This project demonstrates a real-world **Security Operations Center (SOC)** workflow that integrates an **Endpoint Detection & Response (EDR)** platform with a **Security Orchestration, Automation, and Response (SOAR)** solution to detect threats and automatically isolate compromised machines with analyst approval.

The lab simulates an attack, detects it at the endpoint level, triggers automated alerts, requests human approval, and performs controlled containment — just like a modern SOC environment.

🚀 Project Objective

The objective of this project is to design and implement an automated incident response workflow that:

- Detects malicious activity on an endpoint
- Sends detection alerts to a SOAR platform
- Notifies security analysts via Slack and Email
- Allows an analyst to decide whether the machine should be isolated
- Automatically isolates the endpoint if approved
- Verifies and reports the isolation status

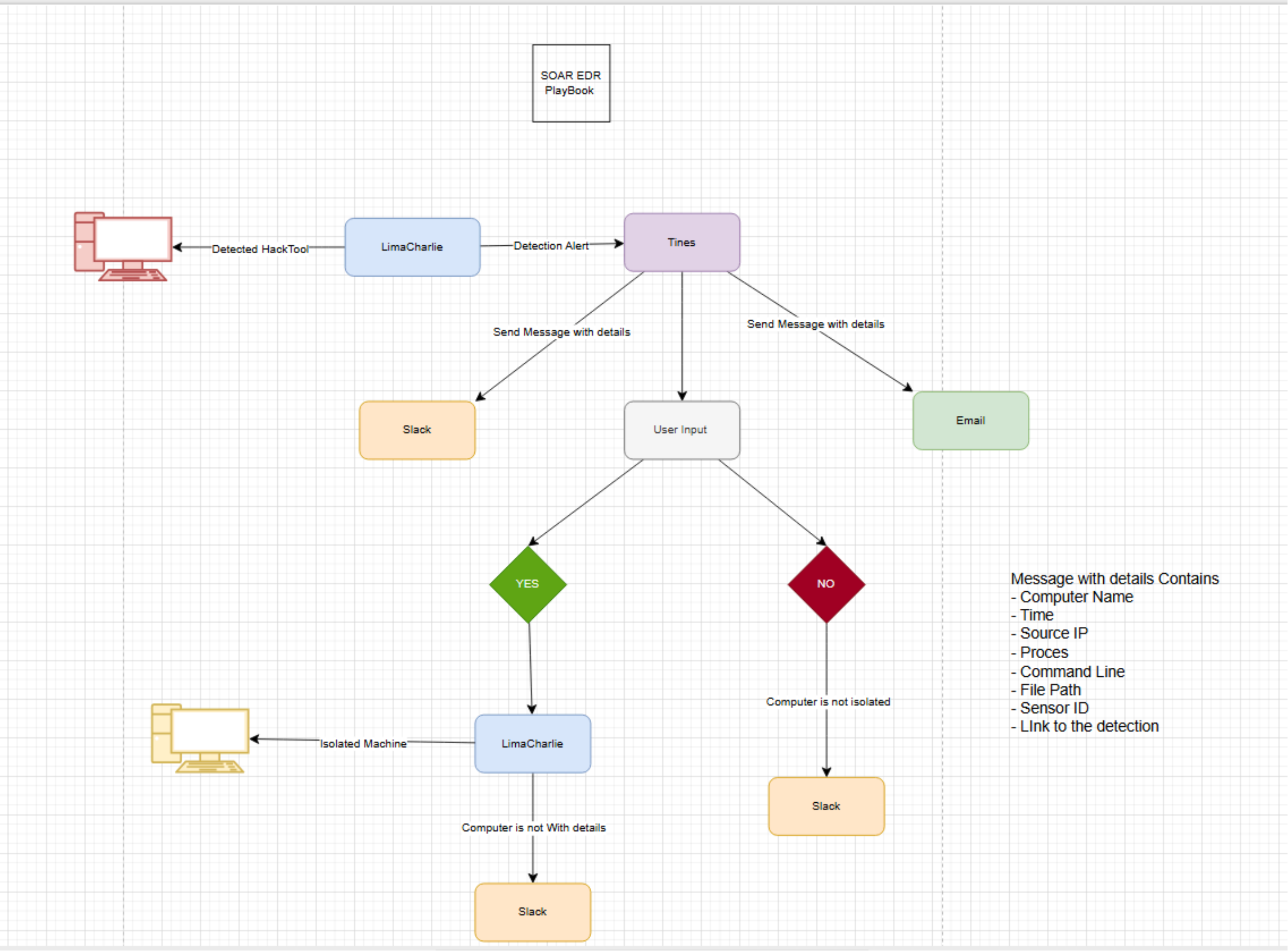
This project reflects how security teams combine automation with human decision-making to reduce response time while avoiding accidental disruptions.

🧰 Technologies Used

TOOL	ROLE IN THIS PROJECT
LimaCharlie (EDR)	Endpoint monitoring, detection, and response actions
Tines (SOAR)	Workflow automation and playbook orchestration
Slack	Real-time SOC alert notifications
Email	Secondary alerting channel
LaZagne	Simulated credential dumping attack tool
Windows 10 (Virtual Machine)	Target endpoint for attack simulation
VirtualBox	Virtual lab environment

🖥️ Lab Architecture Overview

The lab consists of a Windows 10 virtual endpoint monitored by LimaCharlie. When suspicious activity is detected, LimaCharlie forwards the event data to Tines through a webhook. Tines processes the event, sends alerts, requests analyst input, and performs automated containment if approved. The final containment status is then reported back to the SOC team.



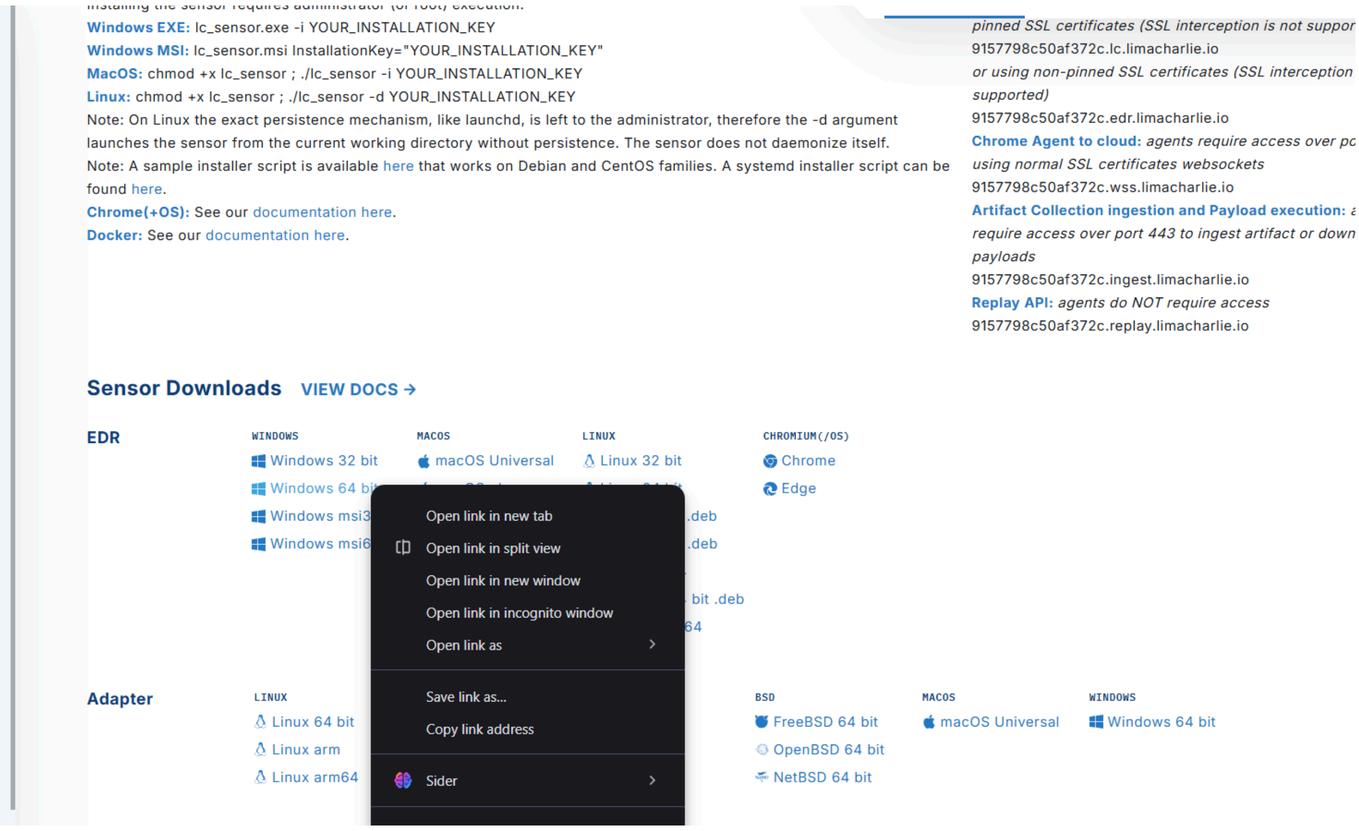
Message with details Contains
- Computer Name
- Time
- Source IP
- Proces
- Command Line
- File Path
- Sensor ID
- LInk to the detection

This architecture demonstrates the full lifecycle of **Detection** → **Alerting** → **Human Decision** → **Automated Response** → **Verification**.

⚙️ Step-by-Step Implementation

1 Endpoint & EDR Deployment

A Windows 10 virtual machine was deployed in VirtualBox to serve as the monitored endpoint. In LimaCharlie, an installation key was generated, and the sensor installation package was downloaded using the generated link. The sensor was installed on the Windows VM using this key.



Installing the sensor requires administrator (or root) execution.

Windows EXE: lc_sensor.exe -i YOUR_INSTALLATION_KEY

Windows MSI: lc_sensor.msi InstallationKey="YOUR_INSTALLATION_KEY"

MacOS: chmod +x lc_sensor ; ./lc_sensor -i YOUR_INSTALLATION_KEY

Linux: chmod +x lc_sensor ; ./lc_sensor -d YOUR_INSTALLATION_KEY

Note: On Linux the exact persistence mechanism, like launchd, is left to the administrator, therefore the -d argument launches the sensor from the current working directory without persistence. The sensor does not daemonize itself.

Note: A sample installer script is available here that works on Debian and CentOS families. A systemd installer script can be found here.

Chrome(+OS): See our documentation here.

Docker: See our documentation here.

pinned SSL certificates (SSL interception is not suppor

9157798c50af372c.lc.limacharlie.io

or using non-pinned SSL certificates (SSL interception supported)

9157798c50af372c.edr.limacharlie.io

Chrome Agent to cloud: *agents require access over pc using normal SSL certificates websockets*

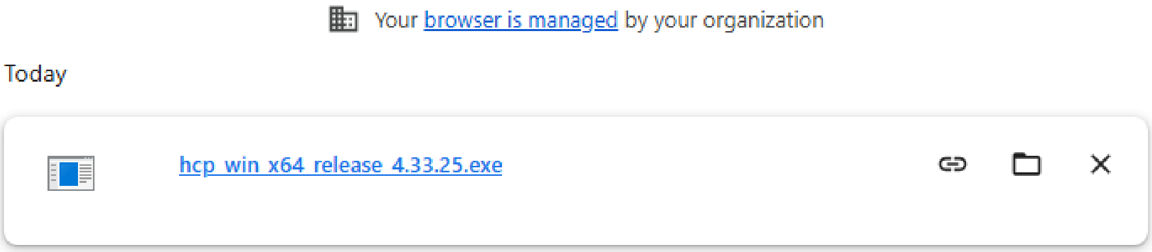
9157798c50af372c.wss.limacharlie.io

Artifact Collection ingestion and Payload execution: *a require access over port 443 to ingest artifact or down payloads*

9157798c50af372c.ingest.limacharlie.io

Replay API: *agents do NOT require access*

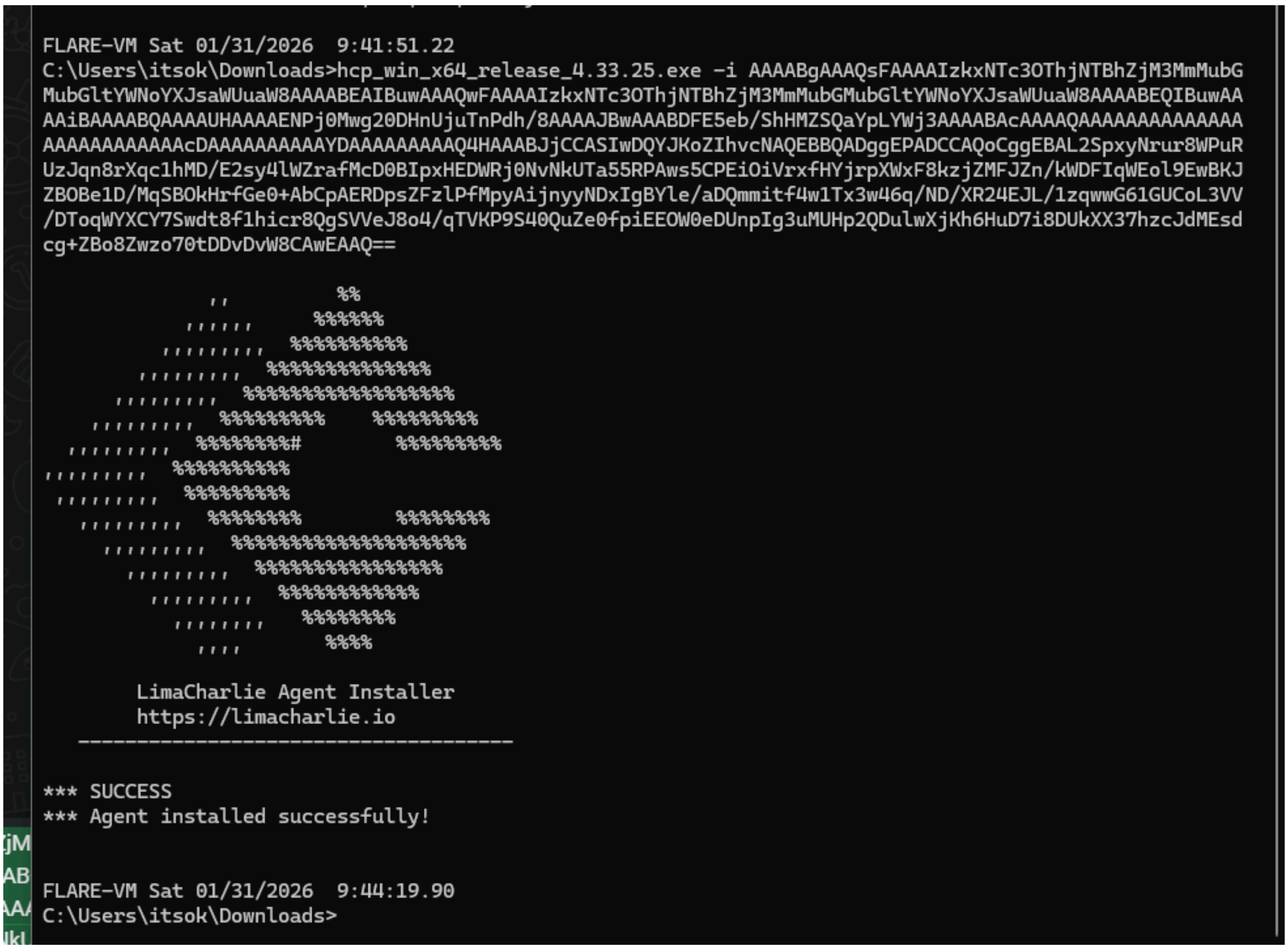
9157798c50af372c.replay.limacharlie.io



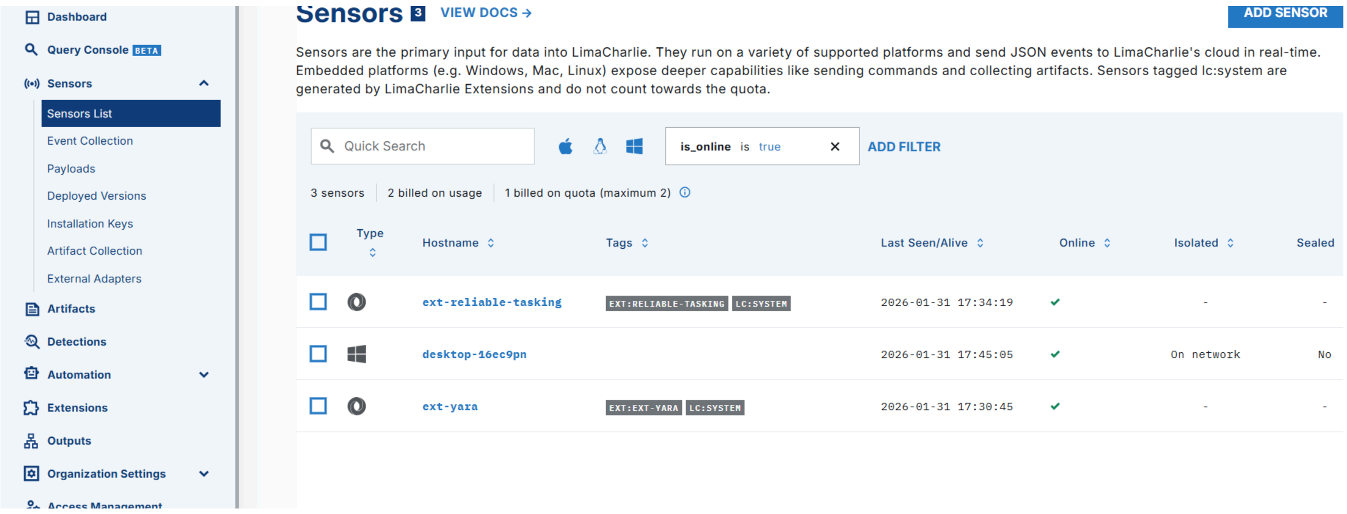
Your browser is managed by your organization

Today

hcp_win_x64_release_4.33.25.exe



```
FLARE-VM Sat 01/31/2026  9:41:51.22
C:\Users\itsok\Downloads>hcp_win_x64_release_4.33.25.exe -i AAAABgAAAQsFAAAAIzkxNTc3OThjNTBhZjM3MmMubG
MubGltYWNoYXJsaWUuaW8AAAABEAIBuwAAAQwFAAAAIzkxNTc3OThjNTBhZjM3MmMubGMubGltYWNoYXJsaWUuaW8AAAABEQIBuwAA
AAiBAAAABQAAAAUHAAAAENPj0Mwg20DHnUjuTnPdh/8AAAAJBwAAABDFE5eb/ShHMZSQaYpLYWj3AAAABAcAAAAQAAAAAAAAAAAAAA
AAAAAAAAAAAcDAAAAAAAAAAYDAAAAAAAAAQ4HAAABJjCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBAL2SpxyNrur8WPuR
UzJqn8rXqc1hMD/E2sy4lWZrafMcD0BIpxHEDWRj0NvNkUTa55RPAws5CPEiOiVrxfHYjrpXWxF8kzjZMFJZn/kWDFIqWEol9EwBKJ
ZBOBe1D/MqSBOkHrfGe0+AbCpAERDpsZFzlPfMpyAijnyyNDxIgBYle/aDQmmitf4w1Tx3w46q/ND/XR24EJL/1zqwwG61GUCoL3VV
/DToqWYXCY7Swdt8f1hicr8QgSVVeJ8o4/qTVKP9S40QuZe0fpiEEOW0eDUnpIg3uMUHp2QDulwXjKh6HuD7i8DUkXX37hzcJdMEsd
cg+ZBo8Zwzo70tDDvDvW8CAwEAAQ==

              ,,        %%
            ,,,,,,    %%%%%%
          ,,,,,,,,, %%%%%%%%%%
        ,,,,,,,,, %%%%%%%%%%%%%%
      ,,,,,,,,, %%%%%%%%%%%%%%%%%%
    ,,,,,,,,, %%%%%%%%%     %%%%%%%%%
  ,,,,,,,,, %%%%%%%%#         %%%%%%%%%
,,,,,,,,, %%%%%%%%%%
 ,,,,,,,,, %%%%%%%%%
   ,,,,,,,,, %%%%%%%%         %%%%%%%%
     ,,,,,,,,, %%%%%%%%%%%%%%%%%%%%
       ,,,,,,,,, %%%%%%%%%%%%%%%
         ,,,,,,,,, %%%%%%%%%%%%
           ,,,,,,,, %%%%%%%%
             ,,,,     %%%%

        LimaCharlie Agent Installer
        https://limacharlie.io
   -------------------------------------

*** SUCCESS
*** Agent installed successfully!

FLARE-VM Sat 01/31/2026  9:44:19.90
C:\Users\itsok\Downloads>
```



Sensors 3

Sensors are the primary input for data into LimaCharlie. They run on a variety of supported platforms and send JSON events to LimaCharlie's cloud in real-time. Embedded platforms (e.g. Windows, Mac, Linux) expose deeper capabilities like sending commands and collecting artifacts. Sensors tagged lc:system are generated by LimaCharlie Extensions and do not count towards the quota.

is_online is true

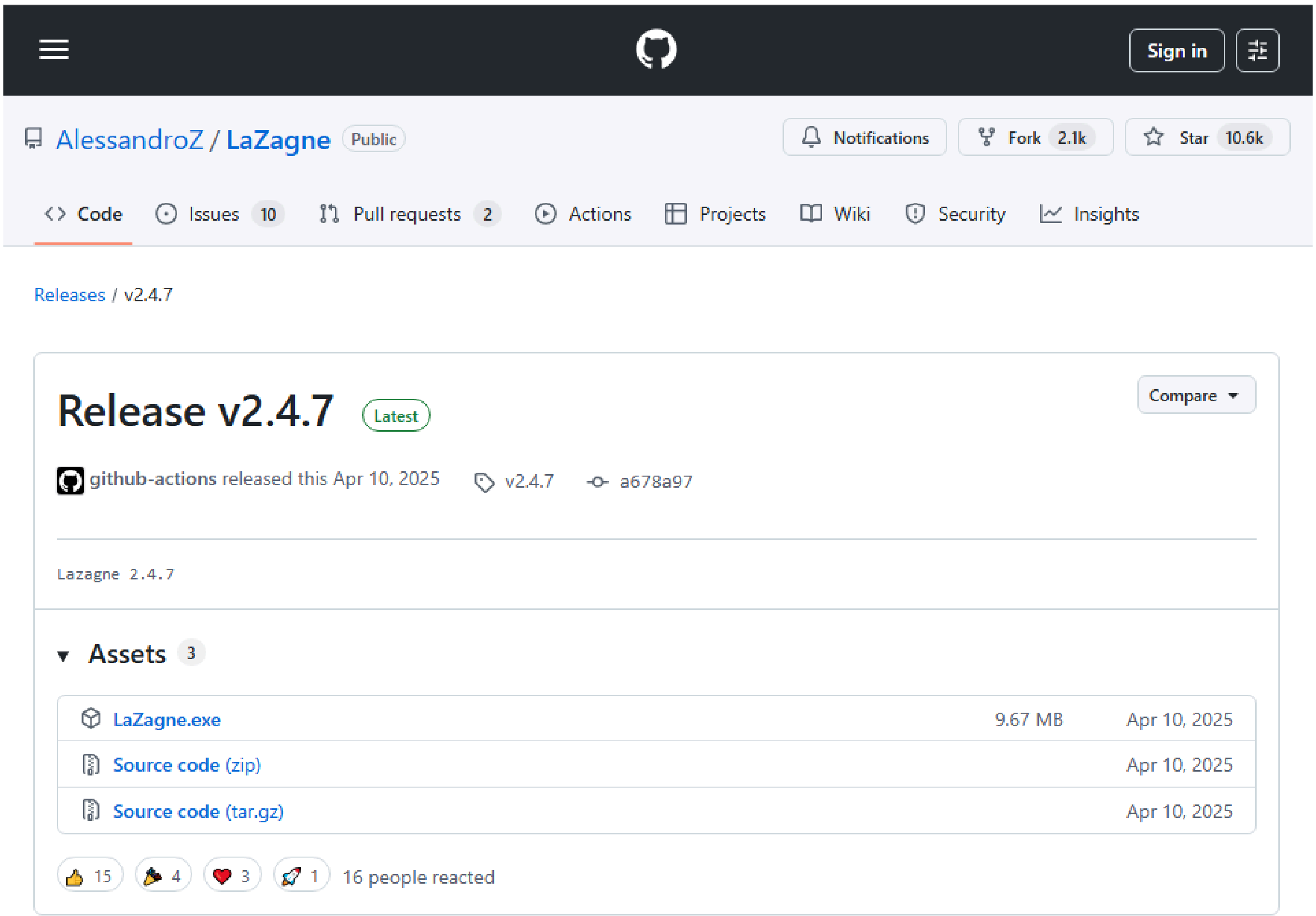
3 sensors | 2 billed on usage | 1 billed on quota (maximum 2)

□	Type	Hostname	Tags	Last Seen/Alive	Online	Isolated	Sealed
□		ext-reliable-tasking	EXT:RELIABLE-TASKING LC:SYSTEM	2026-01-31 17:34:19	✓	-	-
□		desktop-16ec9pn		2026-01-31 17:45:05	✓	On network	No
□		ext-yara	EXT:EXT-YARA LC:SYSTEM	2026-01-31 17:30:45	✓	-	-

After installation, the endpoint appeared in the LimaCharlie Sensors list, confirming that the EDR agent was successfully communicating with the platform and collecting telemetry.

2 Simulating Malicious Activity

To generate realistic security telemetry, the credential recovery tool **LaZagne** was executed on the Windows endpoint. LaZagne attempts to extract stored credentials, which makes it useful for simulating credential dumping behavior seen in real attacks.



Releases / v2.4.7

Release v2.4.7 Latest

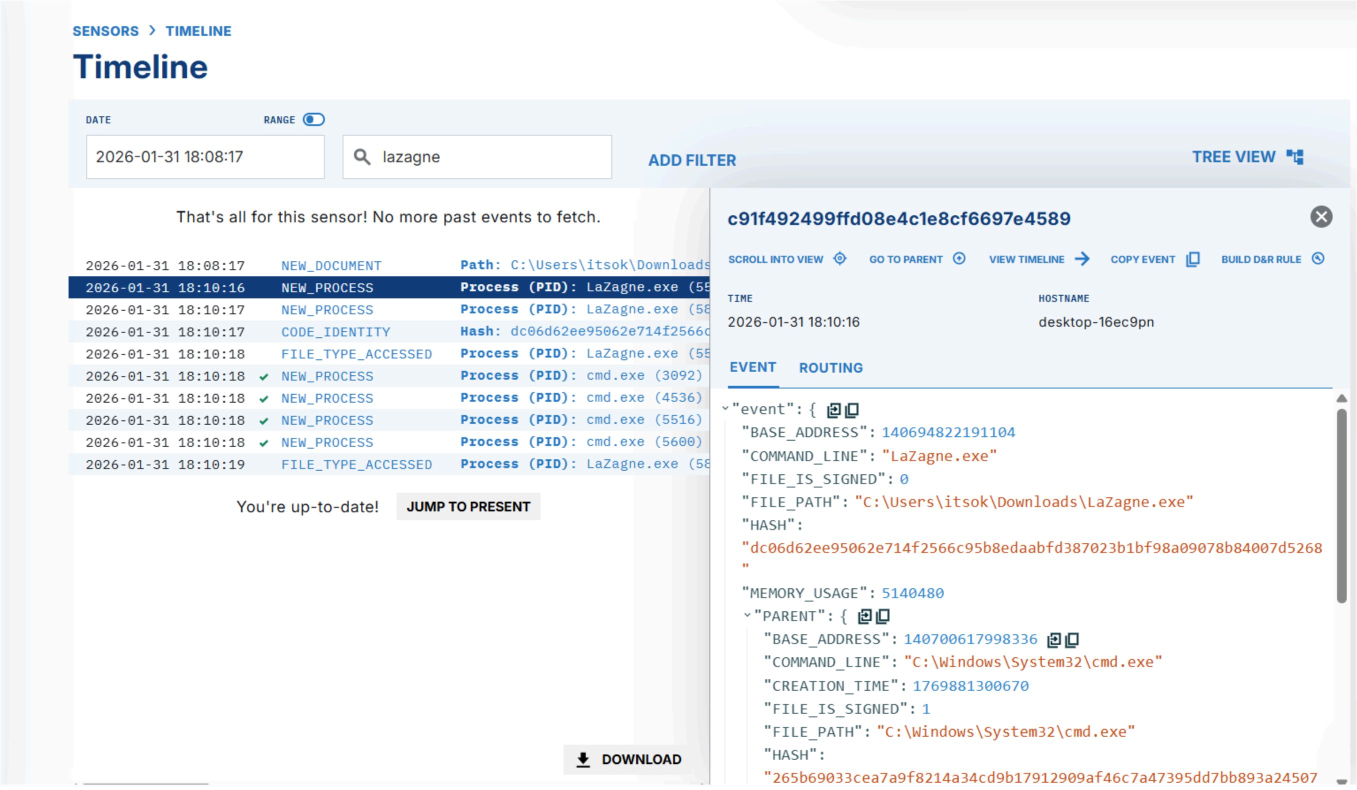
github-actions released this Apr 10, 2025 v2.4.7 a678a97

```
Lazagne 2.4.7
```

Assets 3

LaZagne.exe	9.67 MB	Apr 10, 2025
Source code (zip)		Apr 10, 2025
Source code (tar.gz)		Apr 10, 2025

👍 15 🎉 4 ❤️ 3 🚀 1 16 people reacted



DATE: 2026-01-31 18:08:17

lazagne

That's all for this sensor! No more past events to fetch.

```
2026-01-31 18:08:17   NEW_DOCUMENT         Path: C:\Users\itsok\Downloads
2026-01-31 18:10:16   NEW_PROCESS          Process (PID): LaZagne.exe (55
2026-01-31 18:10:17   NEW_PROCESS          Process (PID): LaZagne.exe (58
2026-01-31 18:10:17   CODE_IDENTITY        Hash: dc06d62ee95062e714f2566c
2026-01-31 18:10:18   FILE_TYPE_ACCESSED   Process (PID): LaZagne.exe (55
2026-01-31 18:10:18 ✓ NEW_PROCESS          Process (PID): cmd.exe (3092)
2026-01-31 18:10:18 ✓ NEW_PROCESS          Process (PID): cmd.exe (4536)
2026-01-31 18:10:18 ✓ NEW_PROCESS          Process (PID): cmd.exe (5516)
2026-01-31 18:10:18 ✓ NEW_PROCESS          Process (PID): cmd.exe (5600)
2026-01-31 18:10:19   FILE_TYPE_ACCESSED   Process (PID): LaZagne.exe (58
```

You're up-to-date!

c91f492499ffd08e4c1e8cf6697e4589

TIME: 2026-01-31 18:10:16

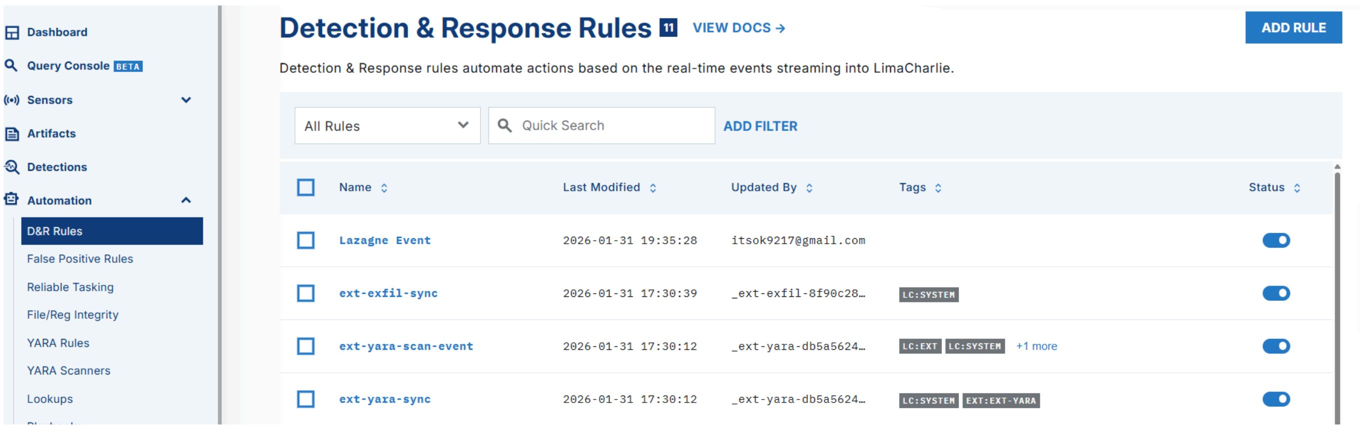
HOSTNAME: desktop-16ec9pn

```
"event": {
  "BASE_ADDRESS": 140694822191104
  "COMMAND_LINE": "LaZagne.exe"
  "FILE_IS_SIGNED": 0
  "FILE_PATH": "C:\Users\itsok\Downloads\LaZagne.exe"
  "HASH":
  "dc06d62ee95062e714f2566c95b8edaabfd387023b1bf98a09078b84007d5268
  "
  "MEMORY_USAGE": 5140480
  "PARENT": {
    "BASE_ADDRESS": 140700617998336
    "COMMAND_LINE": "C:\Windows\System32\cmd.exe"
    "CREATION_TIME": 1769881300670
    "FILE_IS_SIGNED": 1
    "FILE_PATH": "C:\Windows\System32\cmd.exe"
    "HASH":
    "265b69033cea7a9f8214a34cd9b17912909af46c7a47395dd7bb893a24507
```

Executing LaZagne produced process creation and command-line telemetry in LimaCharlie. This data was later used to build a detection rule.

3️⃣ Detection Engineering in LimaCharlie

A Detection & Response (D&R) rule was created in LimaCharlie to detect execution of `lazagne.exe`. Real telemetry from the endpoint timeline was used to test the detection logic to ensure accuracy and reduce false positives.

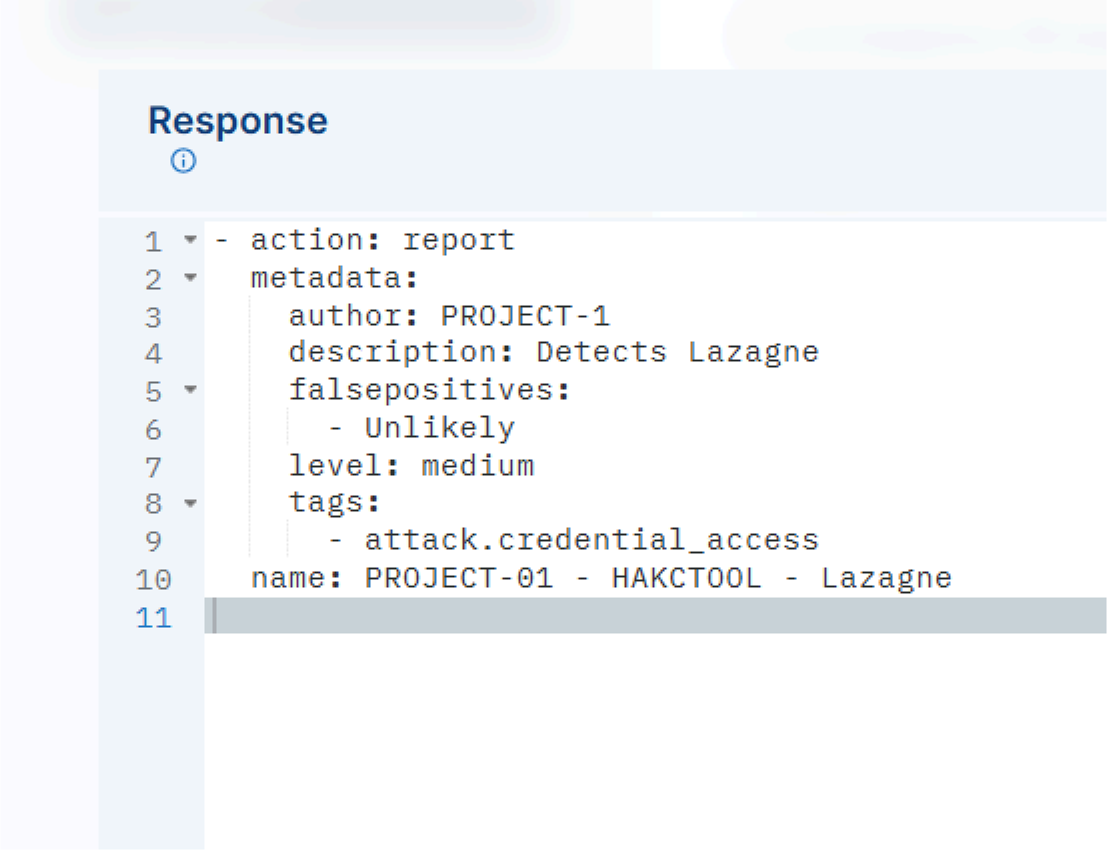


Detection & Response Rules 11

Detection & Response rules automate actions based on the real-time events streaming into LimaCharlie.

□	Name	Last Modified	Updated By	Tags	Status
□	Lazagne Event	2026-01-31 19:35:28	itsok9217@gmail.com		●
□	ext-exfil-sync	2026-01-31 17:30:39	_ext-exfil-8f90c28…	LC:SYSTEM	●
□	ext-yara-scan-event	2026-01-31 17:30:12	_ext-yara-db5a5624…	LC:EXT LC:SYSTEM +1 more	●
□	ext-yara-sync	2026-01-31 17:30:12	_ext-yara-db5a5624…	LC:SYSTEM EXT:EXT-YARA	●

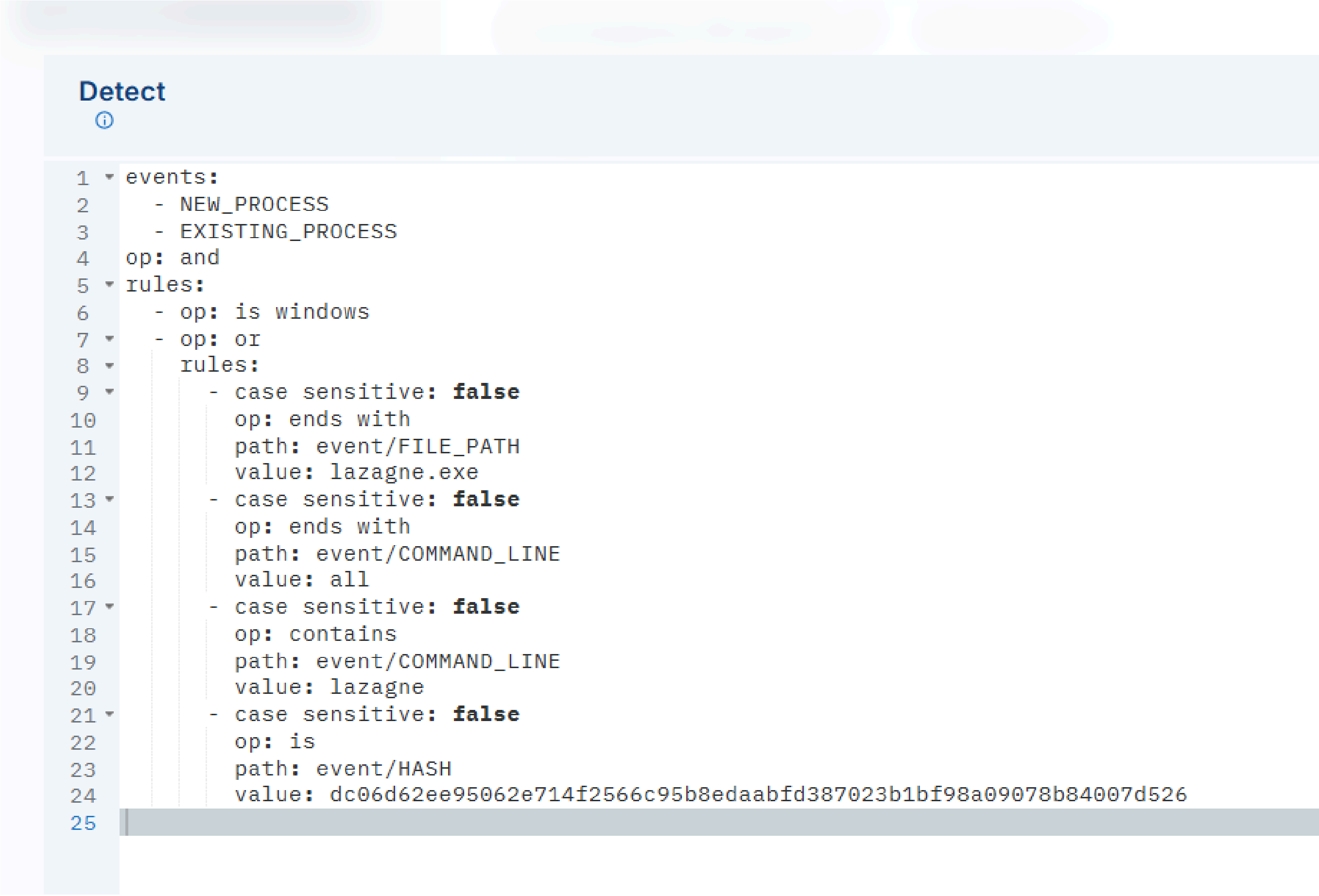
- Created the New rule in D&R rules section (Lazagne Event)



Response

```
- action: report
  metadata:
    author: PROJECT-1
    description: Detects Lazagne
    falsepositives:
      - Unlikely
    level: medium
    tags:
      - attack.credential_access
  name: PROJECT-01 - HAKCTOOL - Lazagne

```

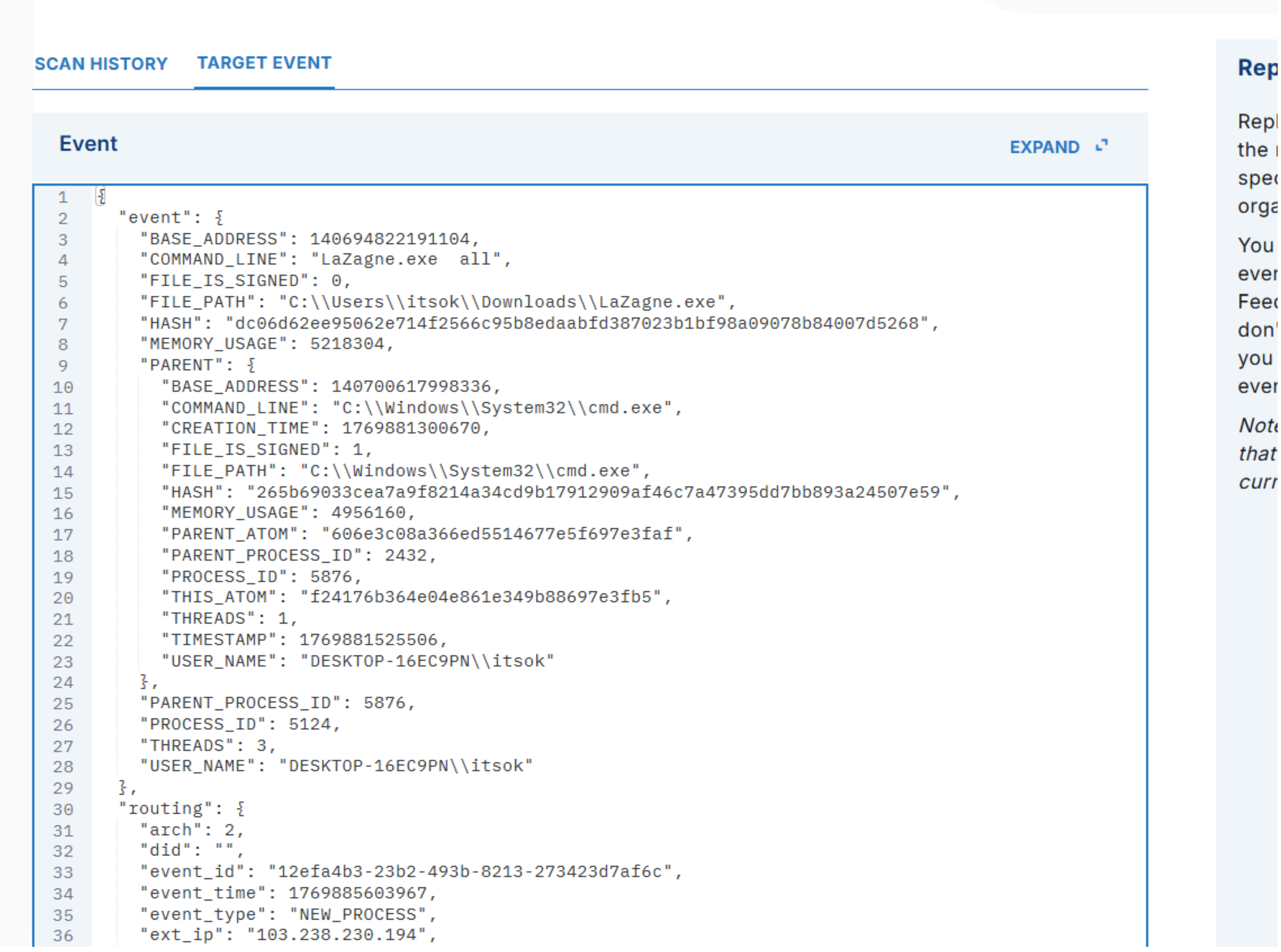
- This is the Response Rule for the below Detection



Detect

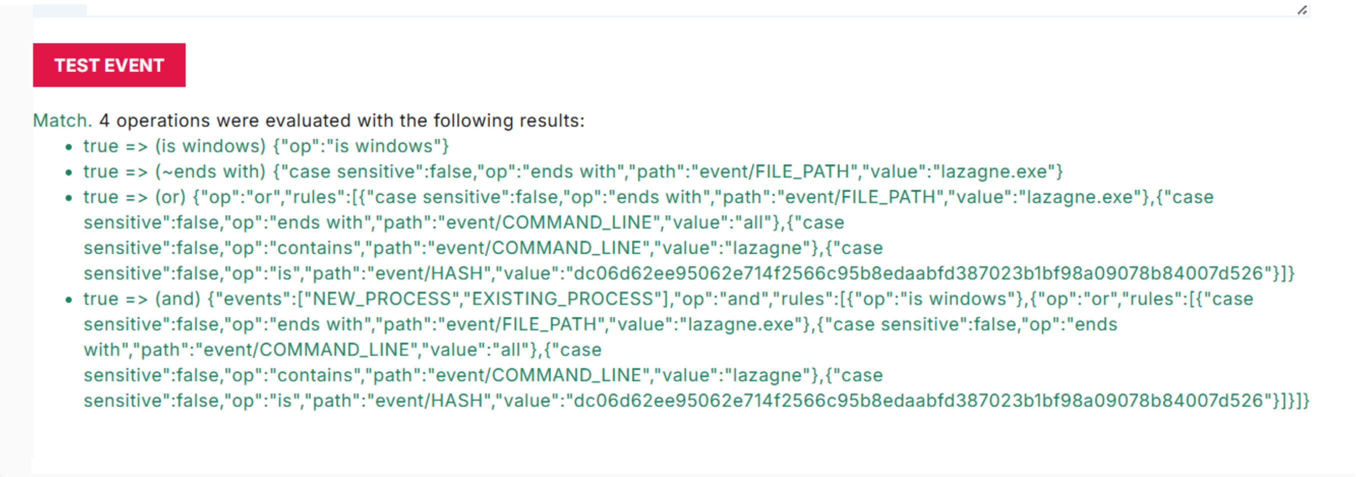
```
events:
  - NEW_PROCESS
  - EXISTING_PROCESS
op: and
rules:
  - op: is windows
  - op: or
    rules:
      - case sensitive: false
        op: ends with
        path: event/FILE_PATH
        value: lazagne.exe
      - case sensitive: false
        op: ends with
        path: event/COMMAND_LINE
        value: all
      - case sensitive: false
        op: contains
        path: event/COMMAND_LINE
        value: lazagne
      - case sensitive: false
        op: is
        path: event/HASH
        value: dc06d62ee95062e714f2566c95b8edaabfd387023b1bf98a09078b84007d526

```

- This is the Detection Rules written according to the Lazagne.exe Event by viewing the Timelines in the LimaCharlie



```
{
  "event": {
    "BASE_ADDRESS": 140694822191104,
    "COMMAND_LINE": "LaZagne.exe  all",
    "FILE_IS_SIGNED": 0,
    "FILE_PATH": "C:\\Users\\itsok\\Downloads\\LaZagne.exe",
    "HASH": "dc06d62ee95062e714f2566c95b8edaabfd387023b1bf98a09078b84007d5268",
    "MEMORY_USAGE": 5218304,
    "PARENT": {
      "BASE_ADDRESS": 140700617998336,
      "COMMAND_LINE": "C:\\Windows\\System32\\cmd.exe",
      "CREATION_TIME": 1769881300670,
      "FILE_IS_SIGNED": 1,
      "FILE_PATH": "C:\\Windows\\System32\\cmd.exe",
      "HASH": "265b69033cea7a9f8214a34cd9b17912909af46c7a47395dd7bb893a24507e59",
      "MEMORY_USAGE": 4956160,
      "PARENT_ATOM": "606e3c08a366ed5514677e5f697e3faf",
      "PARENT_PROCESS_ID": 2432,
      "PROCESS_ID": 5876,
      "THIS_ATOM": "f24176b364e04e861e349b88697e3fb5",
      "THREADS": 1,
      "TIMESTAMP": 1769881525506,
      "USER_NAME": "DESKTOP-16EC9PN\\itsok"
    },
    "PARENT_PROCESS_ID": 5876,
    "PROCESS_ID": 5124,
    "THREADS": 3,
    "USER_NAME": "DESKTOP-16EC9PN\\itsok"
  },
  "routing": {
    "arch": 2,
    "did": "",
    "event_id": "12efa4b3-23b2-493b-8213-273423d7af6c",
    "event_time": 1769885603967,
    "event_type": "NEW_PROCESS",
    "ext_ip": "103.238.230.194",
```

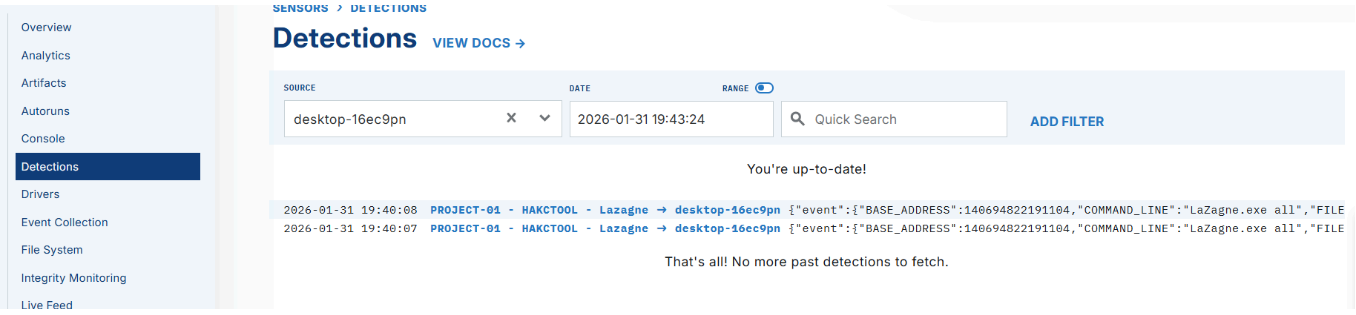
After successful testing, the rule was saved. When LaZagne was executed again, the detection triggered and appeared in the Detections section, confirming the detection rule was functioning correctly.



Match. 4 operations were evaluated with the following results:

- true => (is windows) {"op":"is windows"}
- true => (~ends with) {"case sensitive":false,"op":"ends with","path":"event/FILE_PATH","value":"lazagne.exe"}
- true => (or) {"op":"or","rules":[{"case sensitive":false,"op":"ends with","path":"event/FILE_PATH","value":"lazagne.exe"},{"case sensitive":false,"op":"ends with","path":"event/COMMAND_LINE","value":"all"},{"case sensitive":false,"op":"contains","path":"event/COMMAND_LINE","value":"lazagne"},{"case sensitive":false,"op":"is","path":"event/HASH","value":"dc06d62ee95062e714f2566c95b8edaabfd387023b1bf98a09078b84007d526"}]}
- true => (and) {"events":["NEW_PROCESS","EXISTING_PROCESS"],"op":"and","rules":[{"op":"is windows"},{"op":"or","rules":[{"case sensitive":false,"op":"ends with","path":"event/FILE_PATH","value":"lazagne.exe"},{"case sensitive":false,"op":"ends with","path":"event/COMMAND_LINE","value":"all"},{"case sensitive":false,"op":"contains","path":"event/COMMAND_LINE","value":"lazagne"},{"case sensitive":false,"op":"is","path":"event/HASH","value":"dc06d62ee95062e714f2566c95b8edaabfd387023b1bf98a09078b84007d526"}]}]}

- Here we can see the Test results of input Event (that is the Lazagne Event from Timeline) and we got the all true



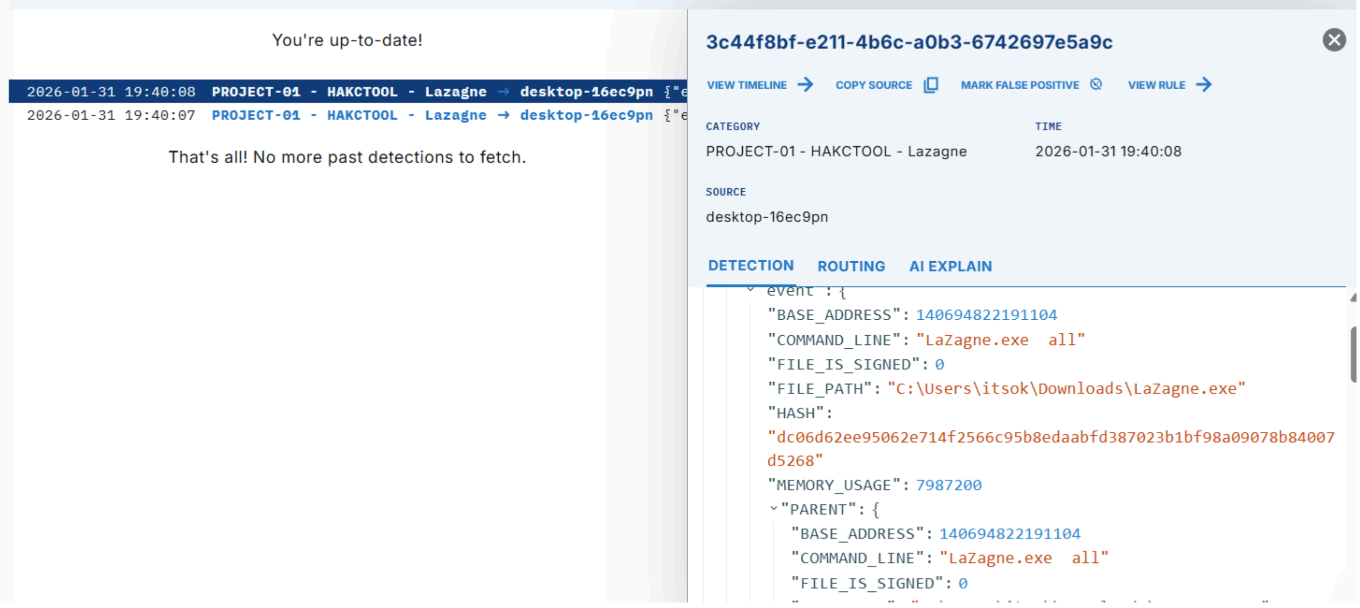
SOURCE: desktop-16ec9pn
DATE: 2026-01-31 19:43:24

You're up-to-date!

```
2026-01-31 19:40:08  PROJECT-01 - HAKCTOOL - Lazagne → desktop-16ec9pn {"event":{"BASE_ADDRESS":140694822191104,"COMMAND_LINE":"LaZagne.exe all","FILE
2026-01-31 19:40:07  PROJECT-01 - HAKCTOOL - Lazagne → desktop-16ec9pn {"event":{"BASE_ADDRESS":140694822191104,"COMMAND_LINE":"LaZagne.exe all","FILE
```

That's all! No more past detections to fetch.

- That detections logs falls under the Detections Section in LimaCharlie



You're up-to-date!

```
2026-01-31 19:40:08  PROJECT-01 - HAKCTOOL - Lazagne → desktop-16ec9pn {"e
2026-01-31 19:40:07  PROJECT-01 - HAKCTOOL - Lazagne → desktop-16ec9pn {"e
```

That's all! No more past detections to fetch.

3c44f8bf-e211-4b6c-a0b3-6742697e5a9c

CATEGORY
PROJECT-01 - HAKCTOOL - Lazagne

TIME
2026-01-31 19:40:08

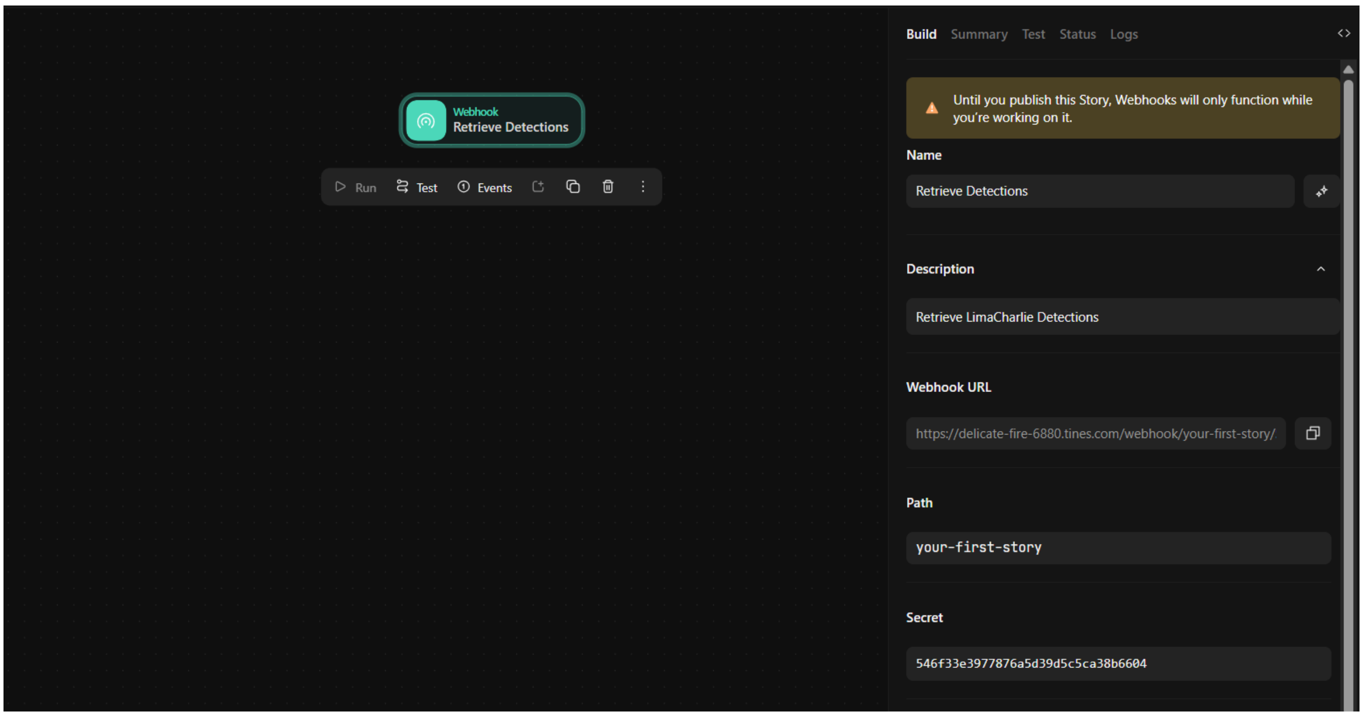
SOURCE
desktop-16ec9pn

```
  event : {
  "BASE_ADDRESS": 140694822191104
  "COMMAND_LINE": "LaZagne.exe  all"
  "FILE_IS_SIGNED": 0
  "FILE_PATH": "C:\Users\itsok\Downloads\LaZagne.exe"
  "HASH":
  "dc06d62ee95062e714f2566c95b8edaabfd387023b1bf98a09078b84007
  d5268"
  "MEMORY_USAGE": 7987200
  "PARENT": {
    "BASE_ADDRESS": 140694822191104
    "COMMAND_LINE": "LaZagne.exe  all"
    "FILE_IS_SIGNED": 0
```

This step demonstrates practical detection engineering using real endpoint telemetry.

4️⃣ Sending Detections to SOAR (Tines)

To automate incident response, LimaCharlie detections were forwarded to Tines using a webhook integration. A webhook was created in Tines, and its URL was configured as an output destination in LimaCharlie.



Until you publish this Story, Webhooks will only function while you're working on it.

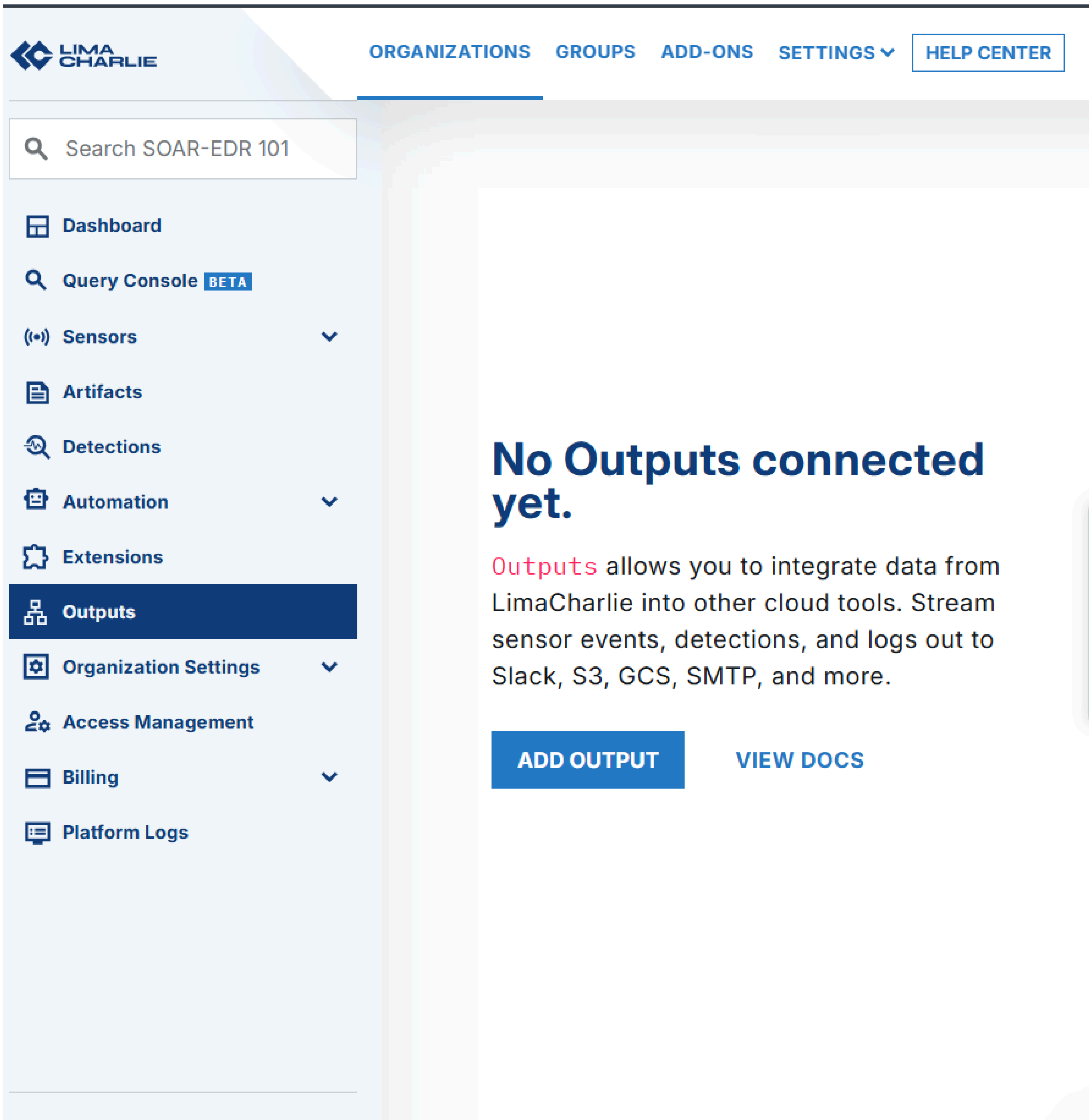
Name
Retrieve Detections

Description
Retrieve LimaCharlie Detections

Webhook URL
https://delicate-fire-6880.tines.com/webhook/your-first-story/

Path
your-first-story

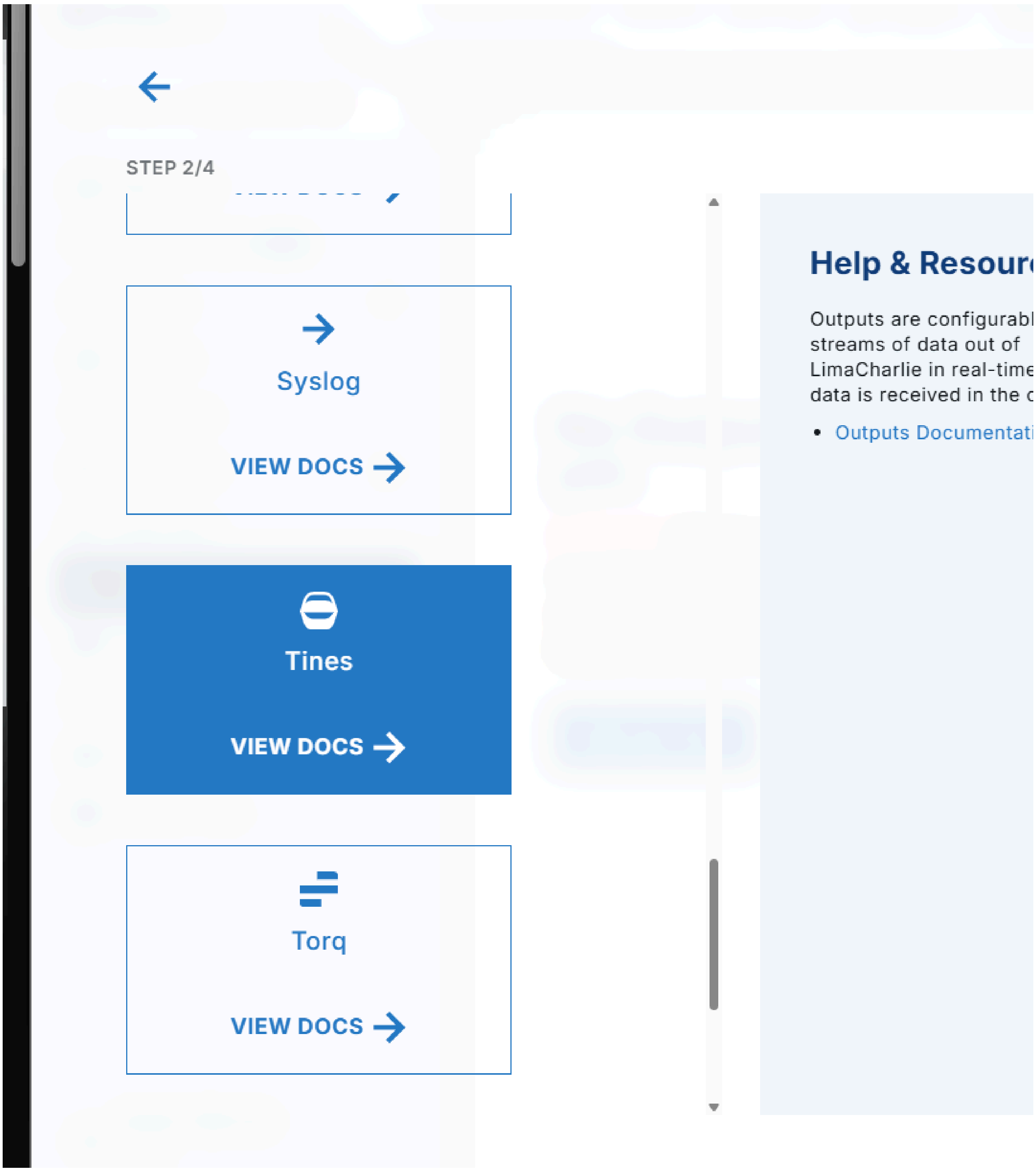
Secret
546f33e3977876a5d39d5c5ca38b6604



No Outputs connected yet.

Outputs allows you to integrate data from LimaCharlie into other cloud tools. Stream sensor events, detections, and logs out to Slack, S3, GCS, SMTP, and more.

- In OUPUT Section , we need to select th application we choosen if not we can go with Webhook
  option to insert the URL to retrieve the Event.



STEP 2/4

Syslog

VIEW DOCS

Tines

VIEW DOCS

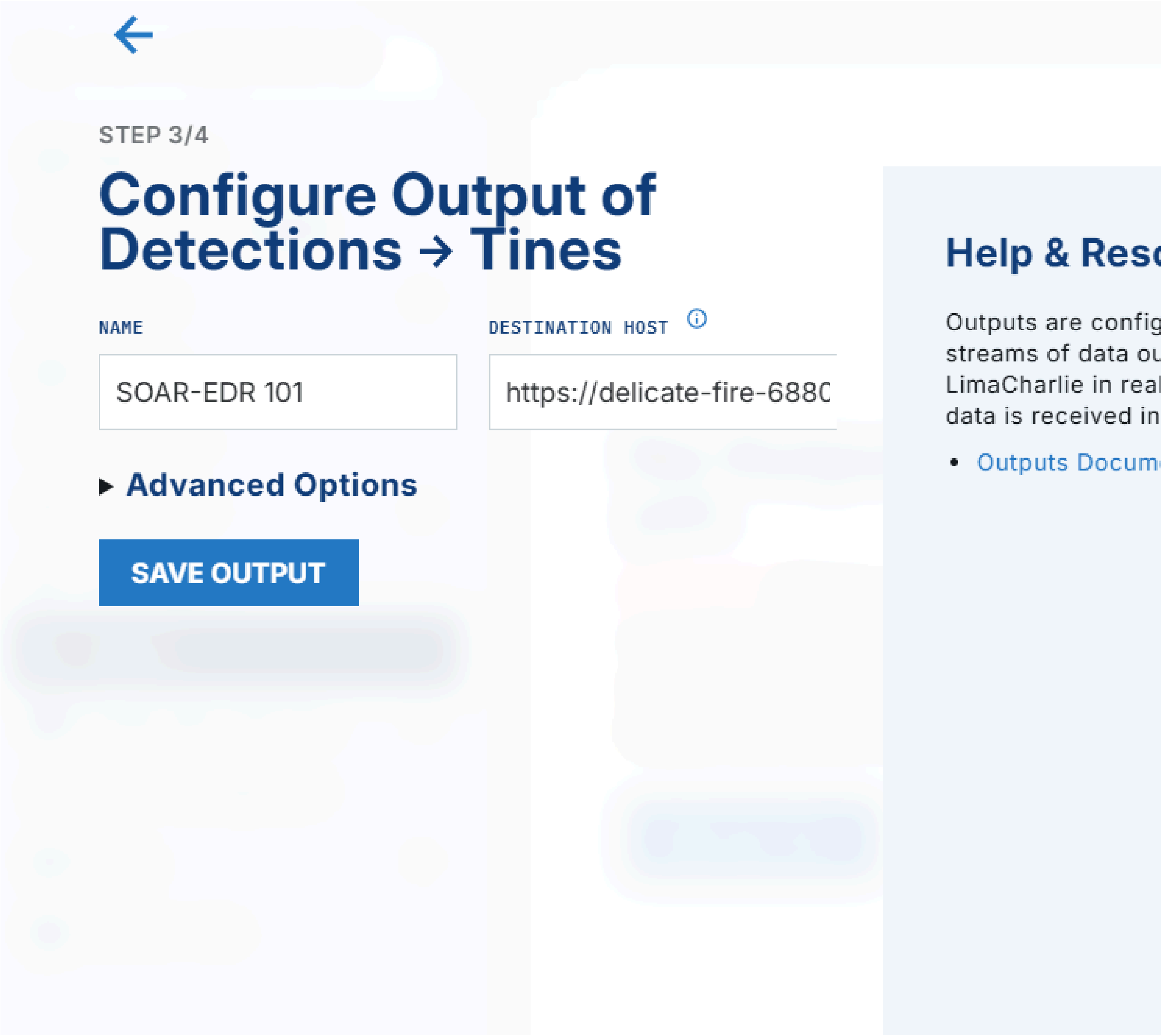
Torq

VIEW DOCS

Help & Resour

Outputs are configurabl
streams of data out of
LimaCharlie in real-time
data is received in the c

- Outputs Documentati



STEP 3/4

Configure Output of Detections → Tines

NAME

SOAR-EDR 101

DESTINATION HOST

https://delicate-fire-688C

▶ Advanced Options

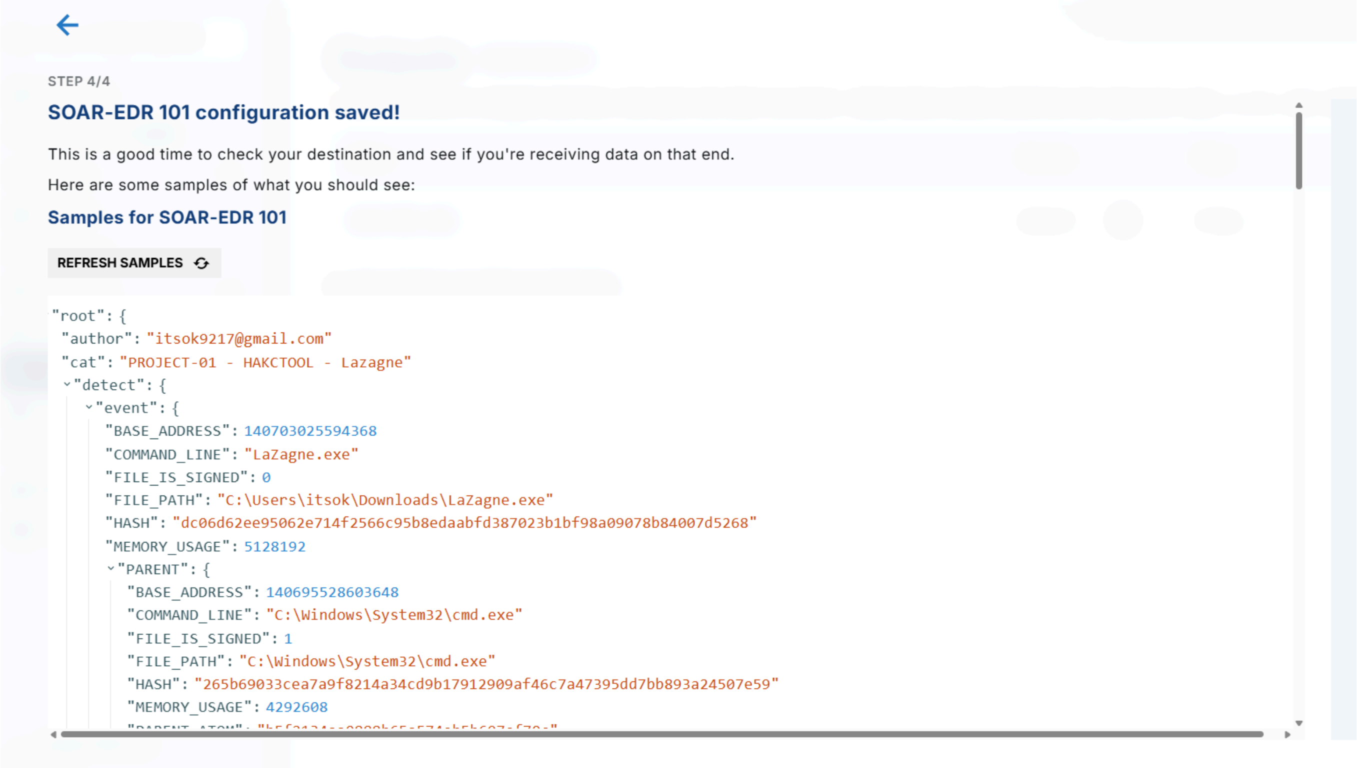
SAVE OUTPUT

Help & Reso

Outputs are config
streams of data ou
LimaCharlie in real
data is received in

- Outputs Docum

When the detection rule triggered again, the event data was successfully received by Tines. This established a working pipeline between the EDR and SOAR platforms.



STEP 4/4

SOAR-EDR 101 configuration saved!

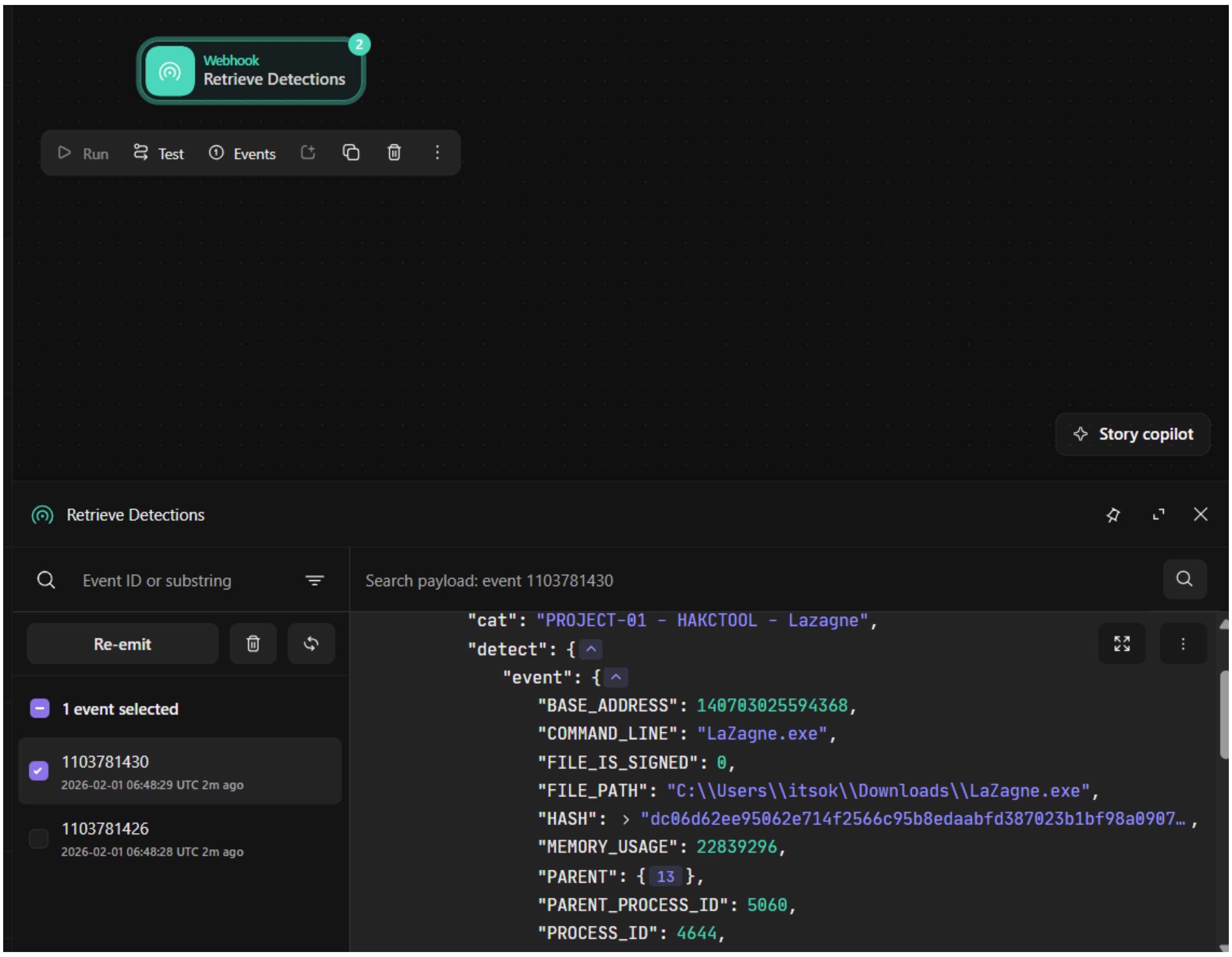
This is a good time to check your destination and see if you're receiving data on that end.

Here are some samples of what you should see:

Samples for SOAR-EDR 101

REFRESH SAMPLES

```
"root": {
 "author": "itsok9217@gmail.com"
 "cat": "PROJECT-01 - HAKCTOOL - Lazagne"
 ˅"detect": {
   ˅"event": {
     "BASE_ADDRESS": 140703025594368
     "COMMAND_LINE": "LaZagne.exe"
     "FILE_IS_SIGNED": 0
     "FILE_PATH": "C:\Users\itsok\Downloads\LaZagne.exe"
     "HASH": "dc06d62ee95062e714f2566c95b8edaabfd387023b1bf98a09078b84007d5268"
     "MEMORY_USAGE": 5128192
    ˅"PARENT": {
       "BASE_ADDRESS": 140695528603648
       "COMMAND_LINE": "C:\Windows\System32\cmd.exe"
       "FILE_IS_SIGNED": 1
       "FILE_PATH": "C:\Windows\System32\cmd.exe"
       "HASH": "265b69033cea7a9f8214a34cd9b17912909af46c7a47395dd7bb893a24507e59"
       "MEMORY_USAGE": 4292608
```



1 event selected

1103781430
2026-02-01 06:48:29 UTC 2m ago

1103781426
2026-02-01 06:48:28 UTC 2m ago

Search payload: event 1103781430

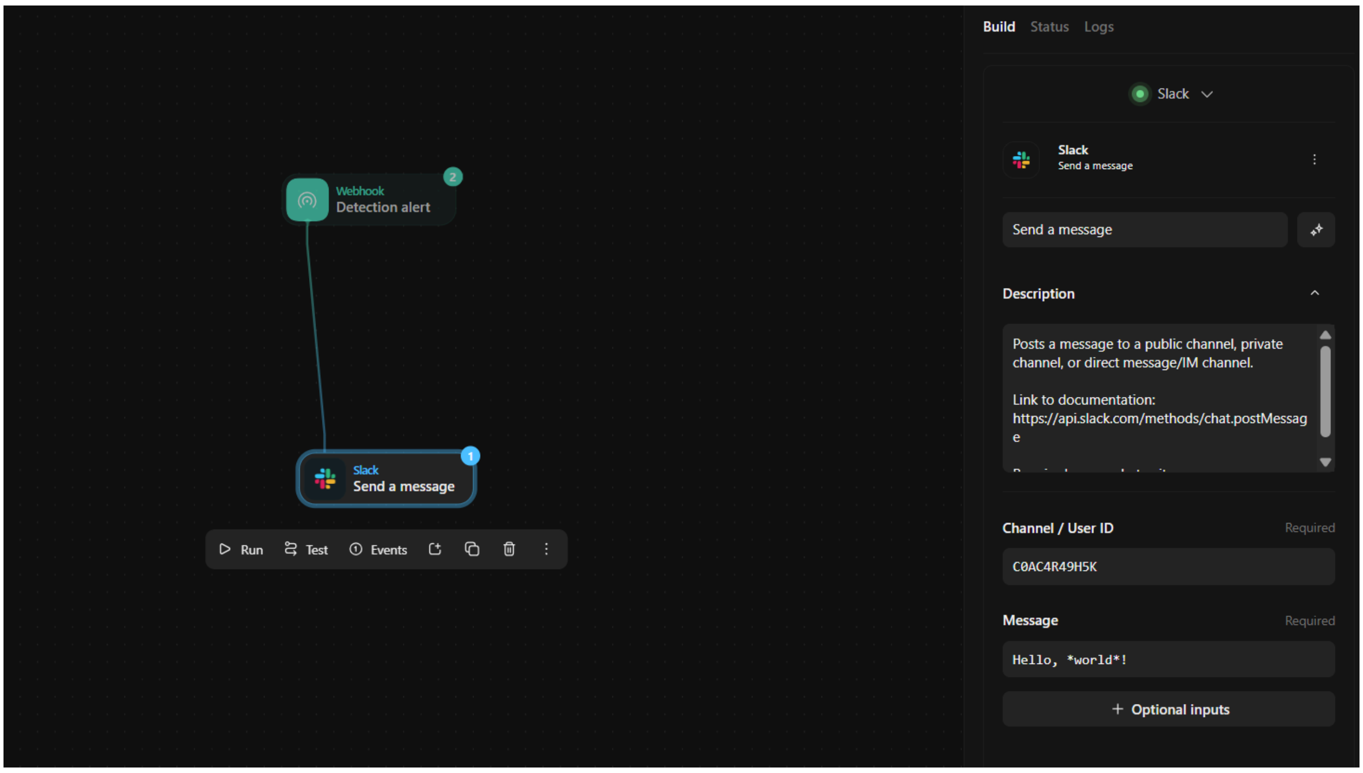
```
    "cat": "PROJECT-01 - HAKCTOOL - Lazagne",
    "detect": {
      "event": {
        "BASE_ADDRESS": 140703025594368,
        "COMMAND_LINE": "LaZagne.exe",
        "FILE_IS_SIGNED": 0,
        "FILE_PATH": "C:\\Users\\itsok\\Downloads\\LaZagne.exe",
        "HASH": > "dc06d62ee95062e714f2566c95b8edaabfd387023b1bf98a0907… ,
        "MEMORY_USAGE": 22839296,
        "PARENT": { 13 },
        "PARENT_PROCESS_ID": 5060,
        "PROCESS_ID": 4644,
```

- And we see the Event retrieved in the webhook we created.

5 Alerting the SOC Team

Within Tines, automated alert actions were configured to notify analysts:

- A Slack message containing detection details
- An email notification with the same information



Send a message

Description

Posts a message to a public channel, private channel, or direct message/IM channel.

Link to documentation:
https://api.slack.com/methods/chat.postMessage

Channel / User ID Required

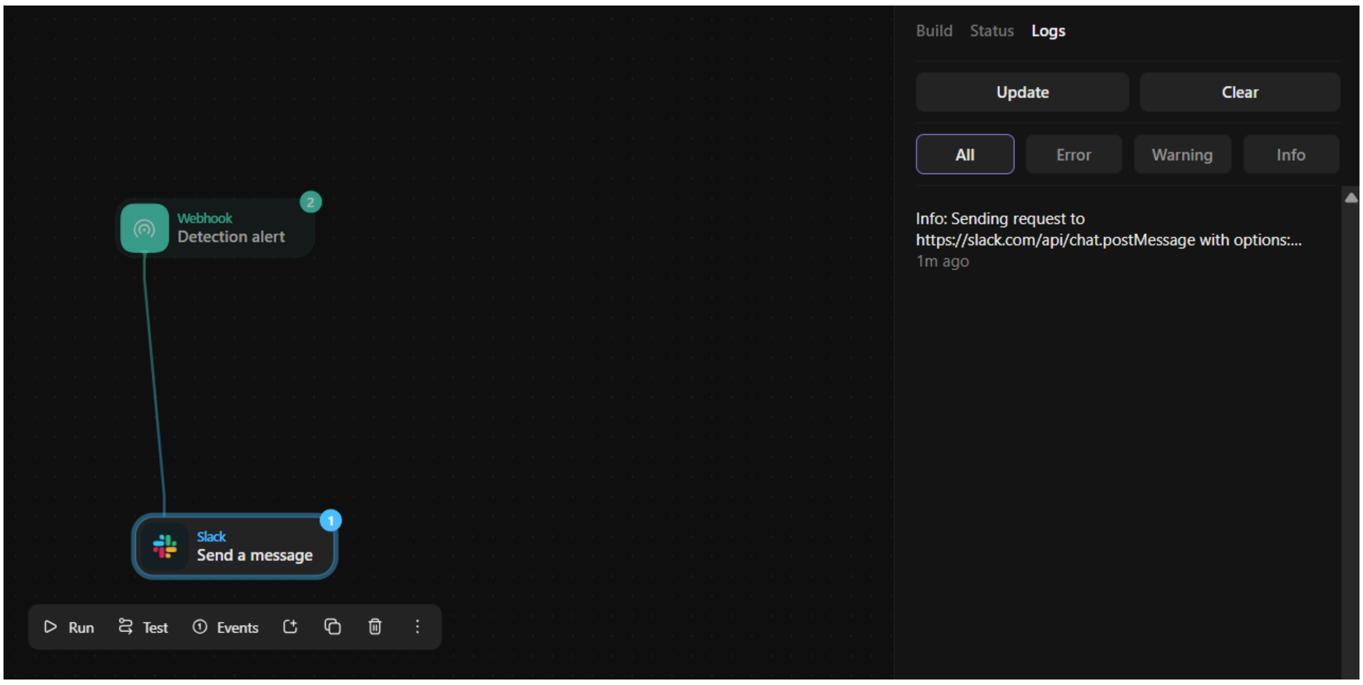
C0AC4R49H5K

Message Required

Hello, *world*!

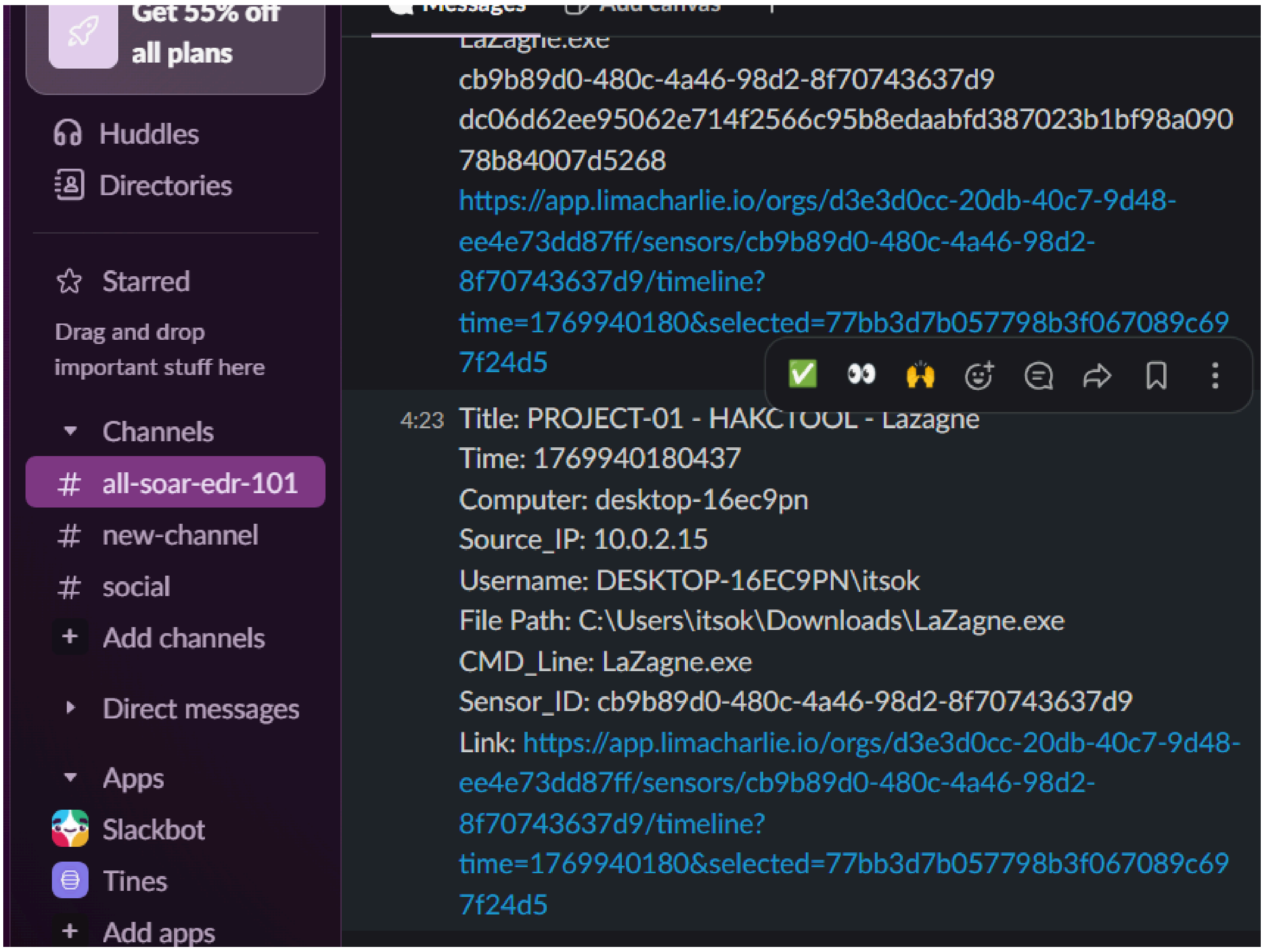
+ Optional inputs

- before using the slack template in tines , we need to add the credentail of the slack in tines.



```
Info: Sending request to
https://slack.com/api/chat.postMessage with options:...
1m ago
```

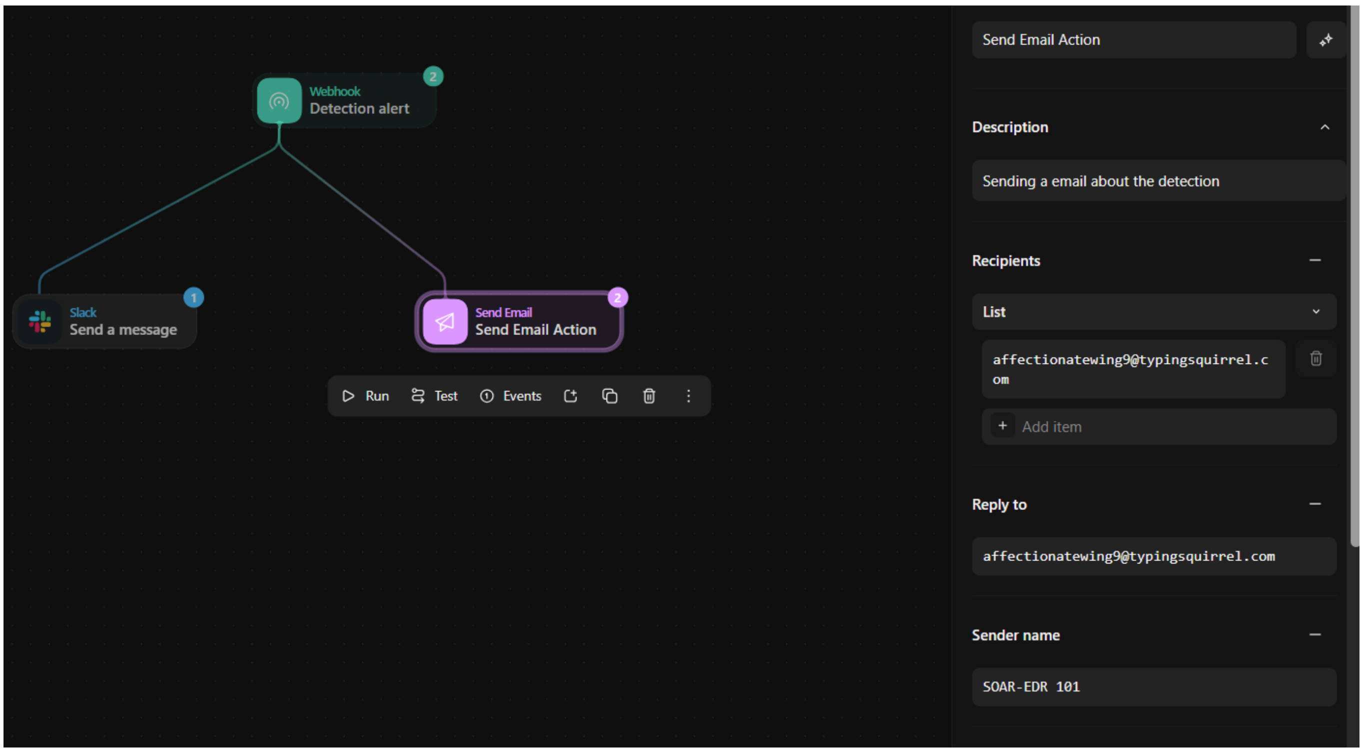
- Here we can see the Logs that , a msg has been send to that slack.



LaZagne.exe
cb9b89d0-480c-4a46-98d2-8f70743637d9
dc06d62ee95062e714f2566c95b8edaabfd387023b1bf98a09078b84007d5268
https://app.limacharlie.io/orgs/d3e3d0cc-20db-40c7-9d48-ee4e73dd87ff/sensors/cb9b89d0-480c-4a46-98d2-8f70743637d9/timeline?time=1769940180&selected=77bb3d7b057798b3f067089c697f24d5

4:23 Title: PROJECT-01 - HAKCTOOL - Lazagne
Time: 1769940180437
Computer: desktop-16ec9pn
Source_IP: 10.0.2.15
Username: DESKTOP-16EC9PN\itsok
File Path: C:\Users\itsok\Downloads\LaZagne.exe
CMD_Line: LaZagne.exe
Sensor_ID: cb9b89d0-480c-4a46-98d2-8f70743637d9
Link: https://app.limacharlie.io/orgs/d3e3d0cc-20db-40c7-9d48-ee4e73dd87ff/sensors/cb9b89d0-480c-4a46-98d2-8f70743637d9/timeline?time=1769940180&selected=77bb3d7b057798b3f067089c697f24d5

- Here is the message that came from the Tines about the Detection Retrieved. (WITH A DETAILS)



Send Email Action

Description

Sending a email about the detection

Recipients

List

affectionatewing9@typingsquirrel.com

+ Add item

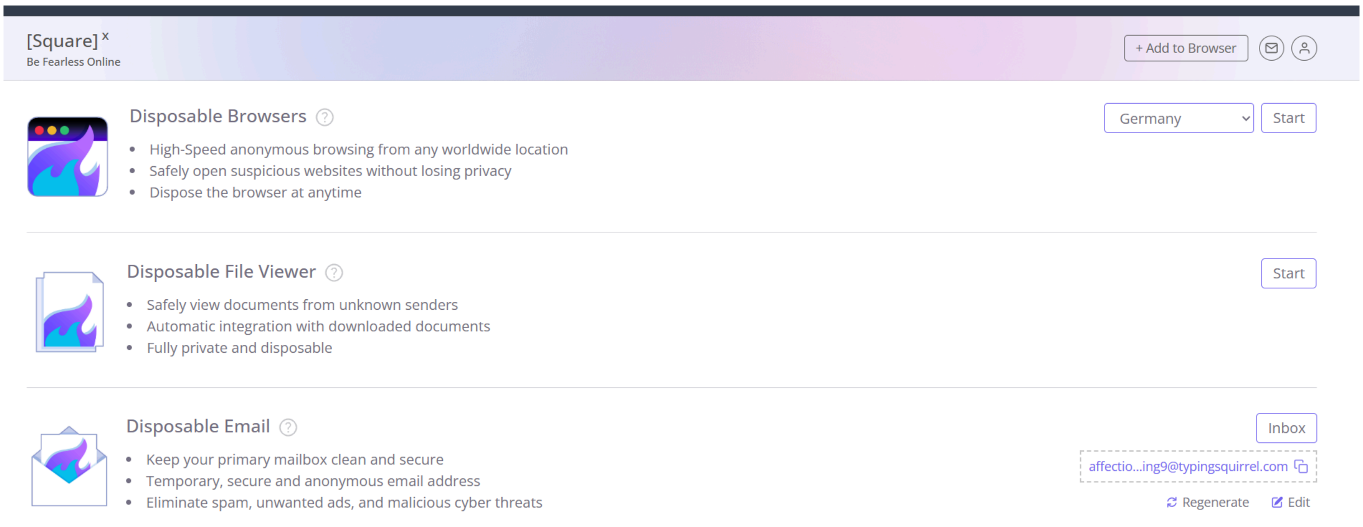
Reply to

affectionatewing9@typingsquirrel.com

Sender name

SOAR-EDR 101

- Using the mail template , and using a temporary mail from SquareX.



Disposable Browsers

- High-Speed anonymous browsing from any worldwide location
- Safely open suspicious websites without losing privacy
- Dispose the browser at anytime

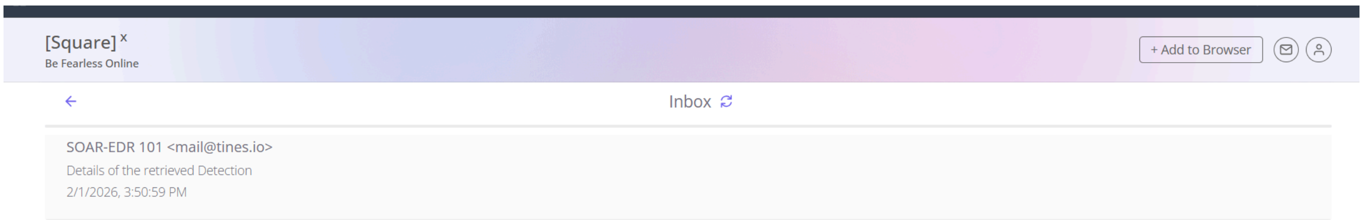
Disposable File Viewer

- Safely view documents from unknown senders
- Automatic integration with downloaded documents
- Fully private and disposable

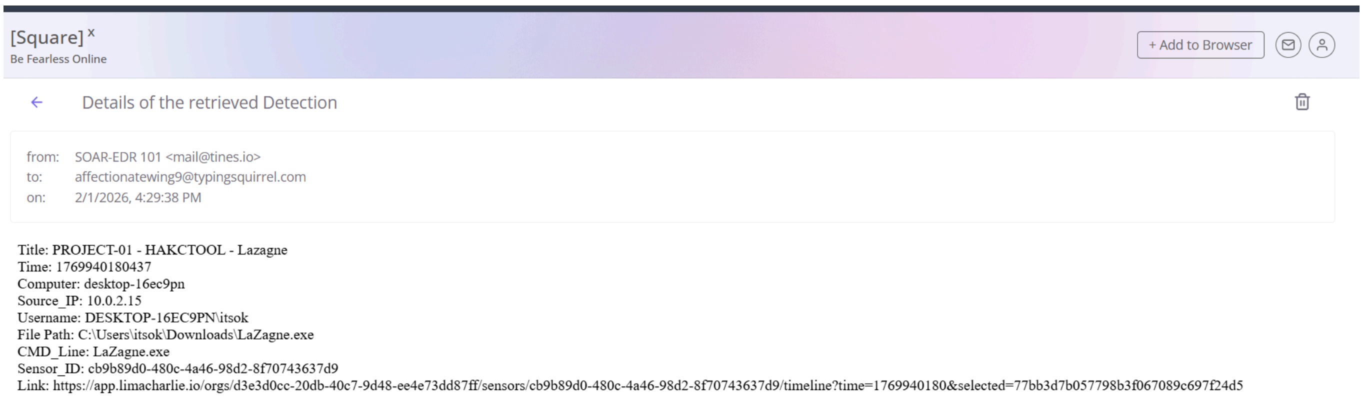
Disposable Email

- Keep your primary mailbox clean and secure
- Temporary, secure and anonymous email address
- Eliminate spam, unwanted ads, and malicious cyber threats

affectio...ing9@typingsquirrel.com



SOAR-EDR 101 <mail@tines.io>
Details of the retrieved Detection
2/1/2026, 3:50:59 PM



Details of the retrieved Detection

from: SOAR-EDR 101 <mail@tines.io>
to: affectionatewing9@typingsquirrel.com
on: 2/1/2026, 4:29:38 PM

```
Title: PROJECT-01 - HAKCTOOL - Lazagne
Time: 1769940180437
Computer: desktop-16ec9pn
Source_IP: 10.0.2.15
Username: DESKTOP-16EC9PN\itsok
File Path: C:\Users\itsok\Downloads\LaZagne.exe
CMD_Line: LaZagne.exe
Sensor_ID: cb9b89d0-480c-4a46-98d2-8f70743637d9
Link: https://app.limacharlie.io/orgs/d3e3d0cc-20db-40c7-9d48-ee4e73dd87ff/sensors/cb9b89d0-480c-4a46-98d2-8f70743637d9/timeline?time=1769940180&selected=77bb3d7b057798b3f067089c697f24d5
```

- And we can see the Detailed mail about the Detection

The alert messages included key forensic information such as:

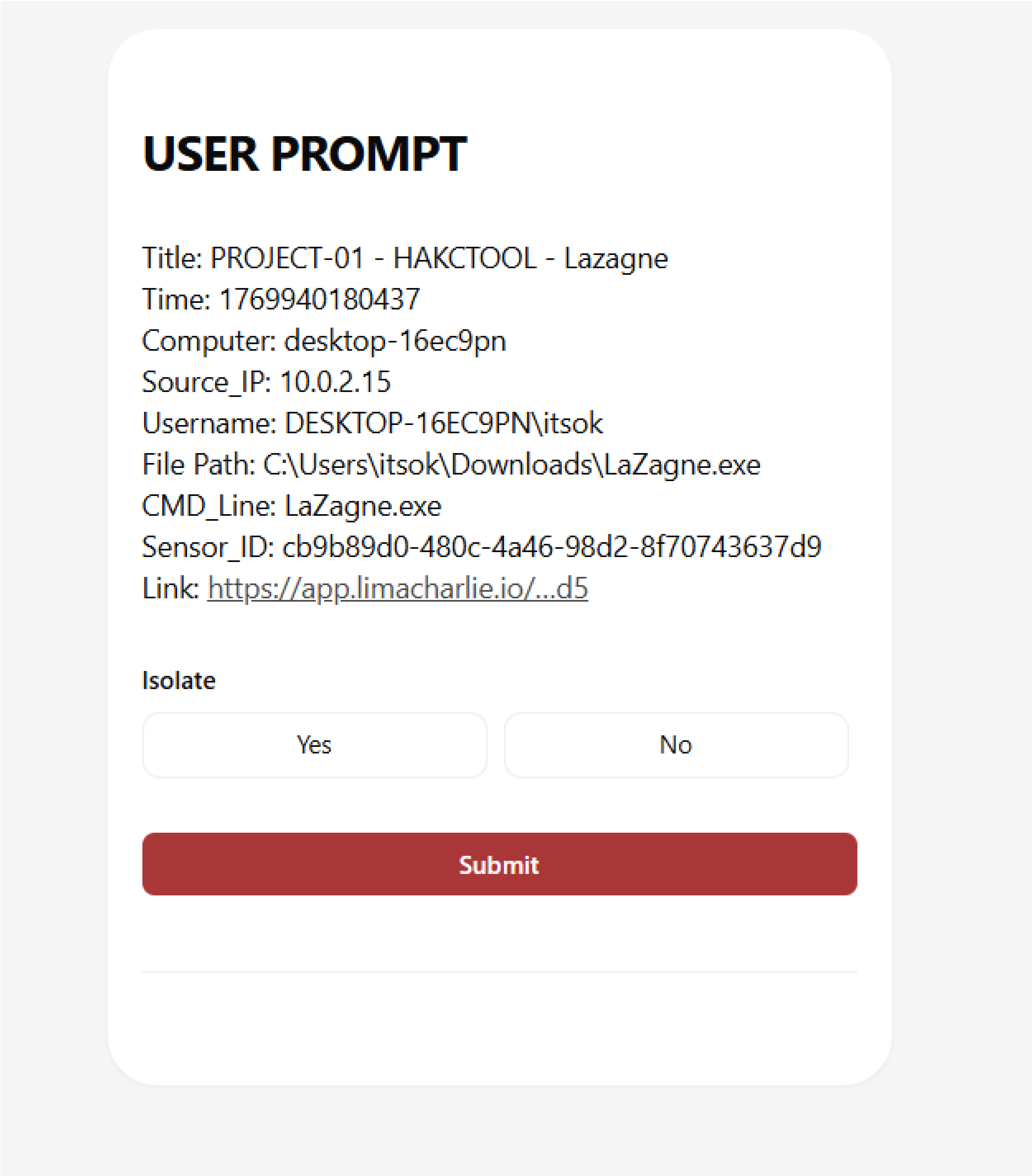
- Computer name
- Username
- Process name
- Command line arguments
- Timestamp of detection

This ensures analysts have sufficient context to evaluate the threat before taking action.

6️⃣ Human-in-the-Loop Decision Page

A Tines Page was created to allow an analyst to make a containment decision. The page presented a simple question:

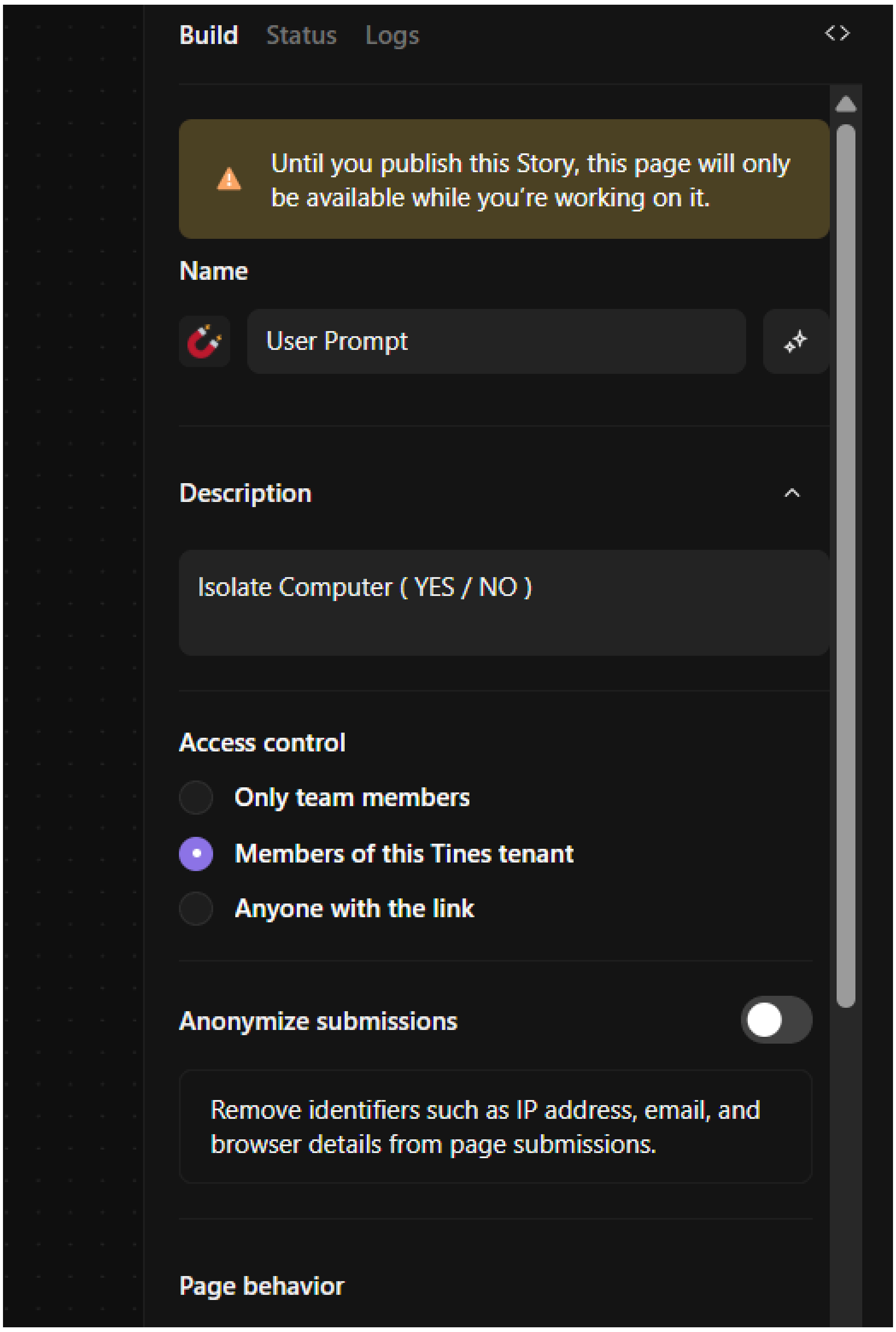
“Do you want to isolate this computer? (YES / NO)”



USER PROMPT

Title: PROJECT-01 - HAKCTOOL - Lazagne
Time: 1769940180437
Computer: desktop-16ec9pn
Source_IP: 10.0.2.15
Username: DESKTOP-16EC9PN\itsok
File Path: C:\Users\itsok\Downloads\LaZagne.exe
CMD_Line: LaZagne.exe
Sensor_ID: cb9b89d0-480c-4a46-98d2-8f70743637d9
Link: https://app.limacharlie.io/...d5

- here are included the info about the detection happen in the computer with main details to do further investigation.



Until you publish this Story, this page will only be available while you're working on it.

Name

User Prompt

Description

Isolate Computer (YES / NO)

Access control

○ Only team members

● Members of this Tines tenant

○ Anyone with the link

Anonymize submissions

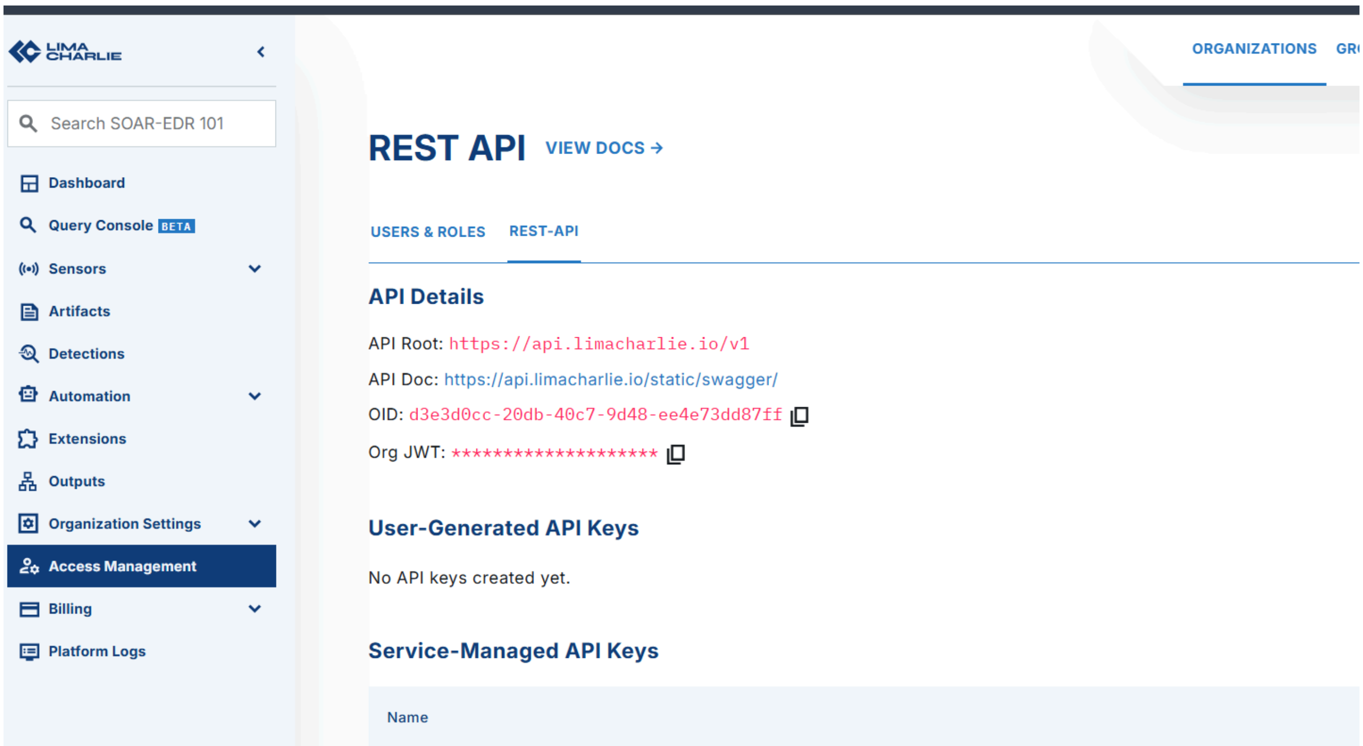
Remove identifiers such as IP address, email, and browser details from page submissions.

Page behavior

This step introduces a human approval checkpoint before performing disruptive response actions like network isolation. This mirrors real SOC procedures where containment must often be approved by an analyst.

7️⃣ Automated Endpoint Isolation (If YES)

If the analyst selected **YES**, Tines triggered an API request to LimaCharlie to isolate the endpoint. This placed the Windows machine into network isolation mode, preventing it from communicating with other systems.



REST API

API Details

API Root: https://api.limacharlie.io/v1

API Doc: https://api.limacharlie.io/static/swagger/

OID: d3e3d0cc-20db-40c7-9d48-ee4e73dd87ff

Org JWT: ********************

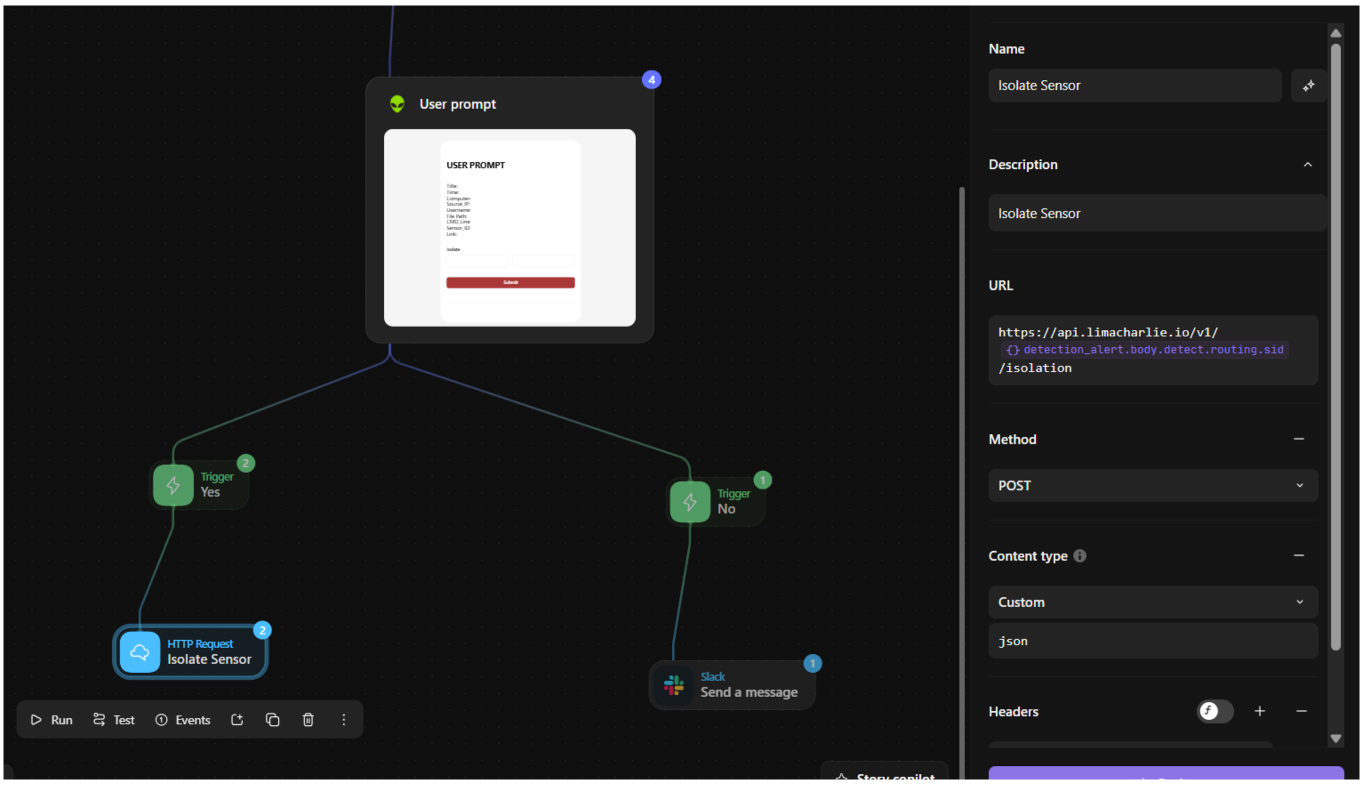
User-Generated API Keys

No API keys created yet.

Service-Managed API Keys

Name

- here we need to use the API key to use the LIMACHARLIE template in tines to isolate the network access of the windows machine , so this API will be insert at the build page of the template



Name

Isolate Sensor

Description

Isolate Sensor

URL

```
https://api.limacharlie.io/v1/
{} detection_alert.body.detect.routing.sid
/isolation
```

Method

POST

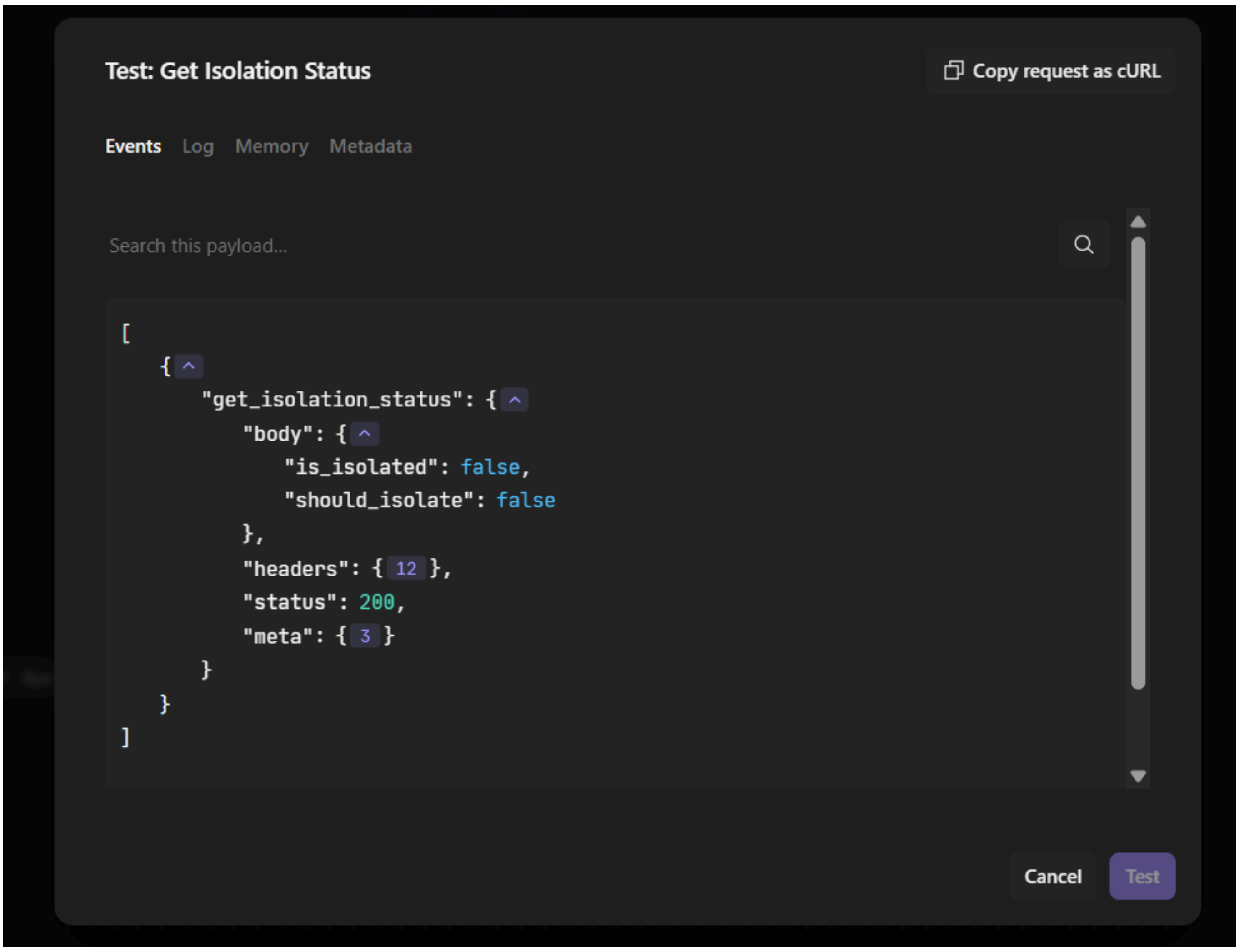
Content type

Custom

json

Headers

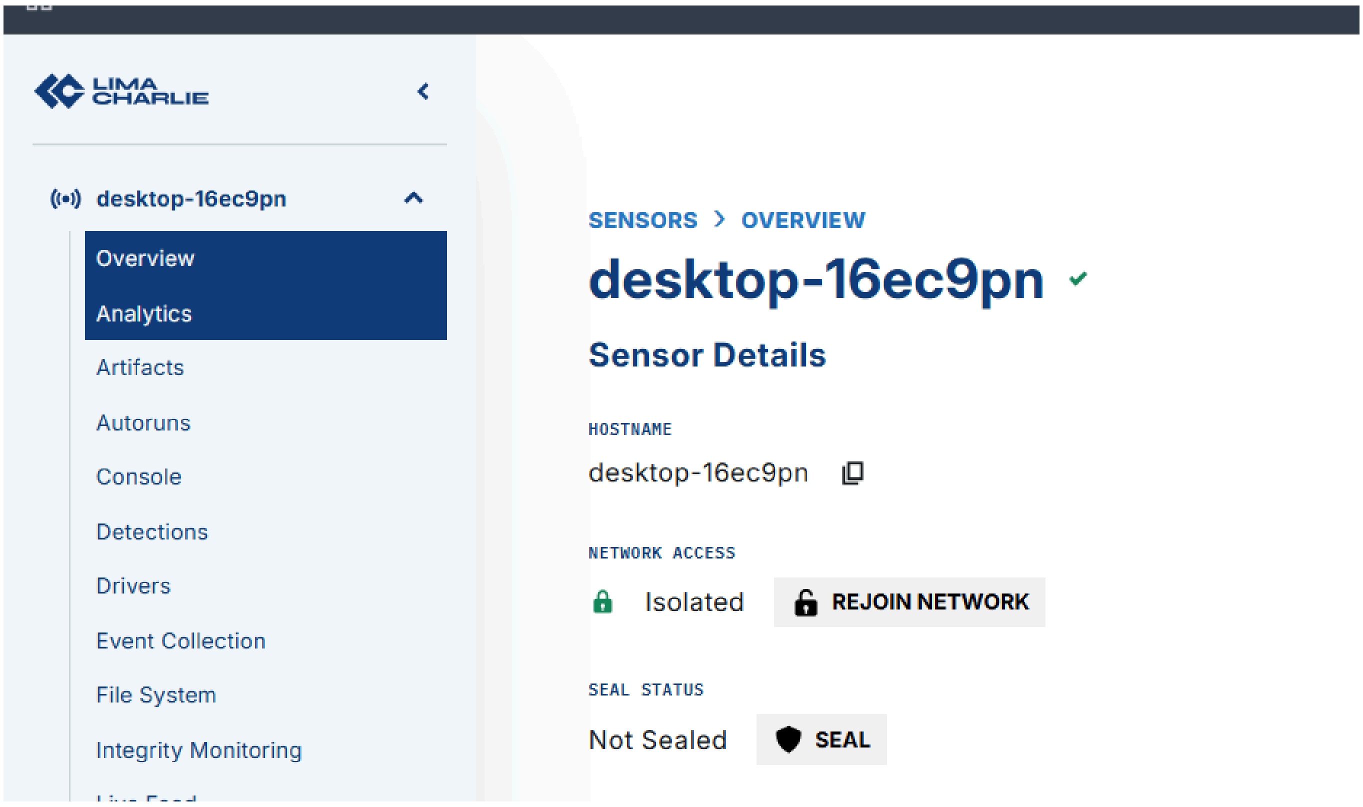
- Using the Isolate Sensor of Limacharlie to isolate the mahcine , Here we are isolating using the (SID) that is Sensor ID to identify the machine.



Test: Get Isolation Status

```
[
    {
        "get_isolation_status": {
            "body": {
                "is_isolated": false,
                "should_isolate": false
            },
            "headers": { 12 },
            "status": 200,
            "meta": { 3 }
        }
    }
]
```

The isolation status was verified in LimaCharlie, where the sensor showed that network isolation was active. A final Slack message was sent to inform the SOC team that the endpoint had been successfully isolated.



desktop-16ec9pn ✓

Sensor Details

HOSTNAME

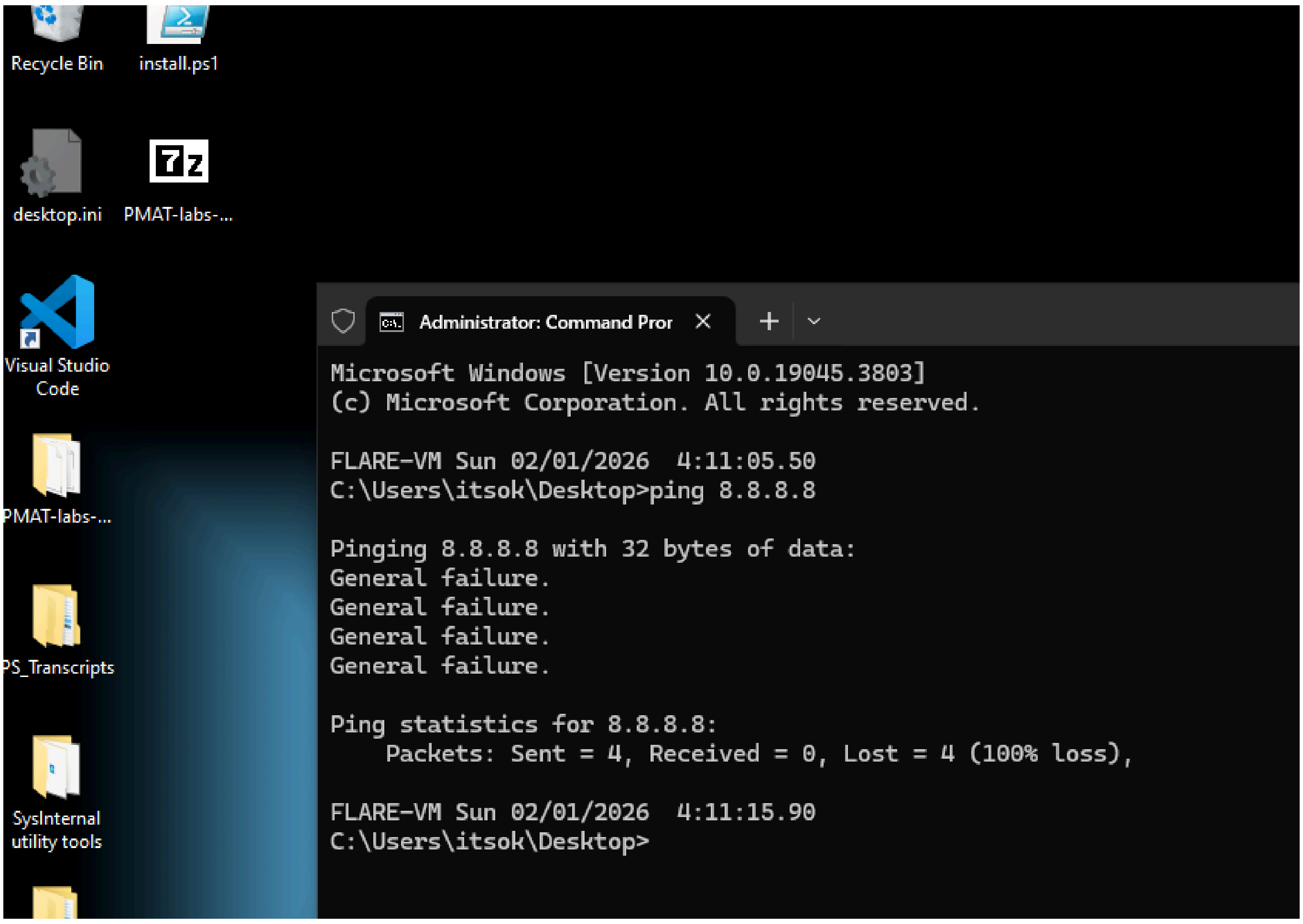
desktop-16ec9pn

NETWORK ACCESS

Isolated

SEAL STATUS

Not Sealed

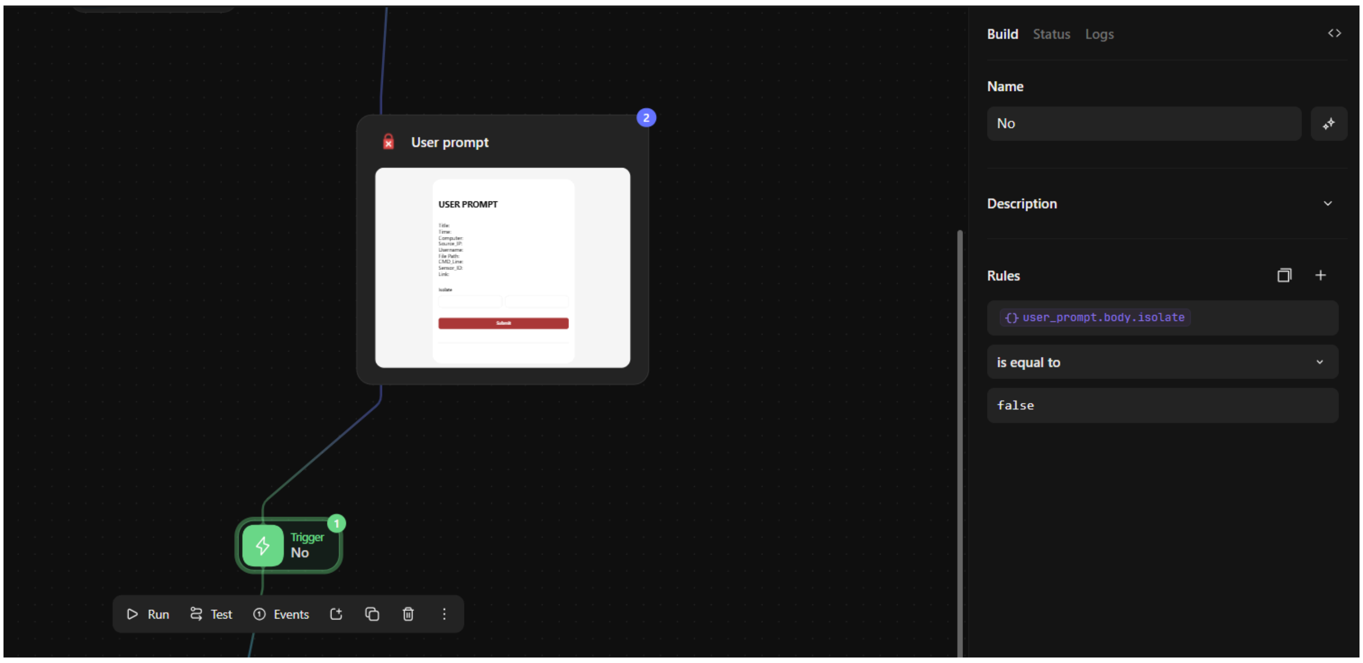


```
Microsoft Windows [Version 10.0.19045.3803]
(c) Microsoft Corporation. All rights reserved.

FLARE-VM Sun 02/01/2026  4:11:05.50
C:\Users\itsok\Desktop>ping 8.8.8.8

Pinging 8.8.8.8 with 32 bytes of data:
General failure.
General failure.
General failure.
General failure.

Ping statistics for 8.8.8.8:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

FLARE-VM Sun 02/01/2026  4:11:15.90
C:\Users\itsok\Desktop>
```

- after the isolation , running a ping operation to check the network connection and we can see that we arenot getting any response back (getting GENERAL FAILURE), Cuz we are isolated the Machine .
  This step demonstrates automated containment with verification.

8 No Isolation Path (If NO)

If the analyst selected **NO**, the system did not isolate the endpoint. Instead, Tines sent a Slack message stating that the machine was not isolated and required further investigation. This maintains visibility while allowing analysts to continue manual investigation.



Name: No

Description

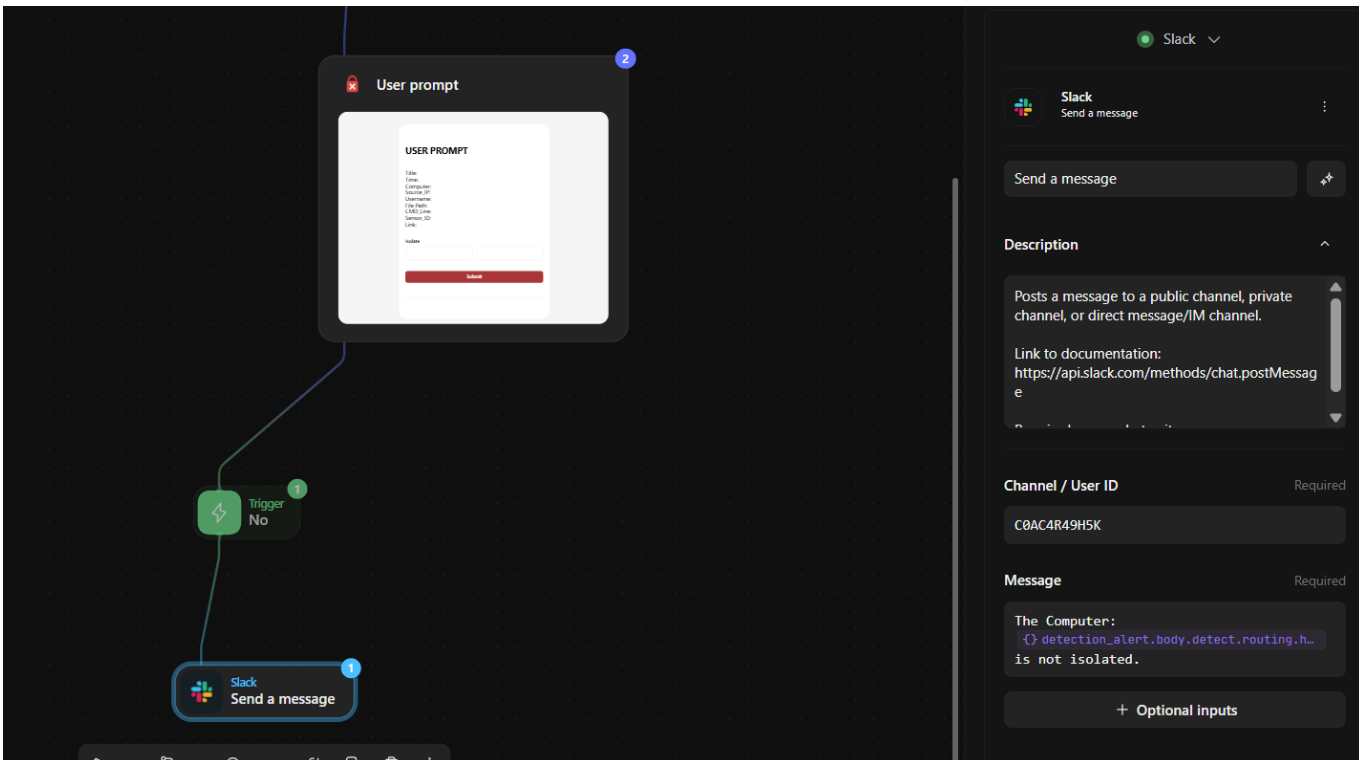
Rules

user_prompt.body.isolate

is equal to

false

Trigger No



Send a message

Description

Posts a message to a public channel, private channel, or direct message/IM channel.

Link to documentation: https://api.slack.com/methods/chat.postMessage

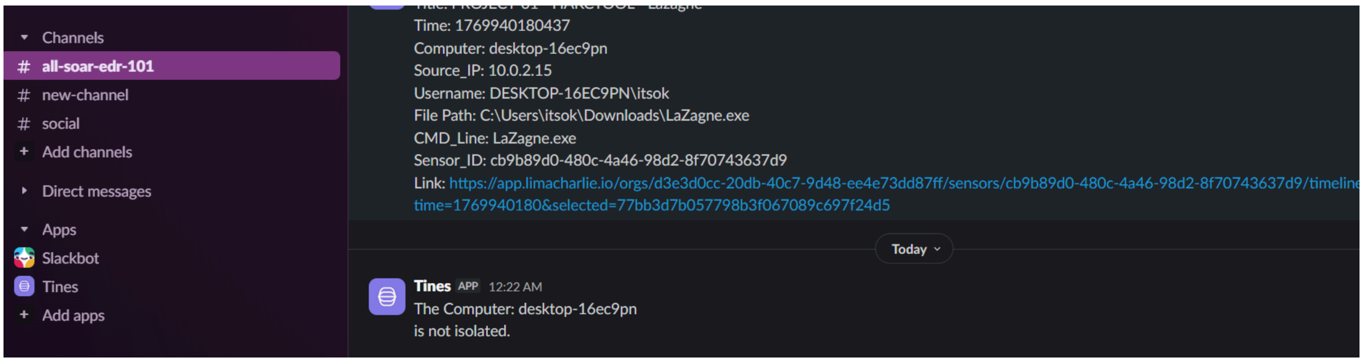
Channel / User ID Required

C0AC4R49H5K

Message Required

```
The Computer:
{} detection_alert.body.detect.routing.h…
is not isolated.
```

- here if the analyst (user prompt) selected "NO" then details containing about the detection will be sent to slack that "the computer is not isolated ", do furthur investigation.

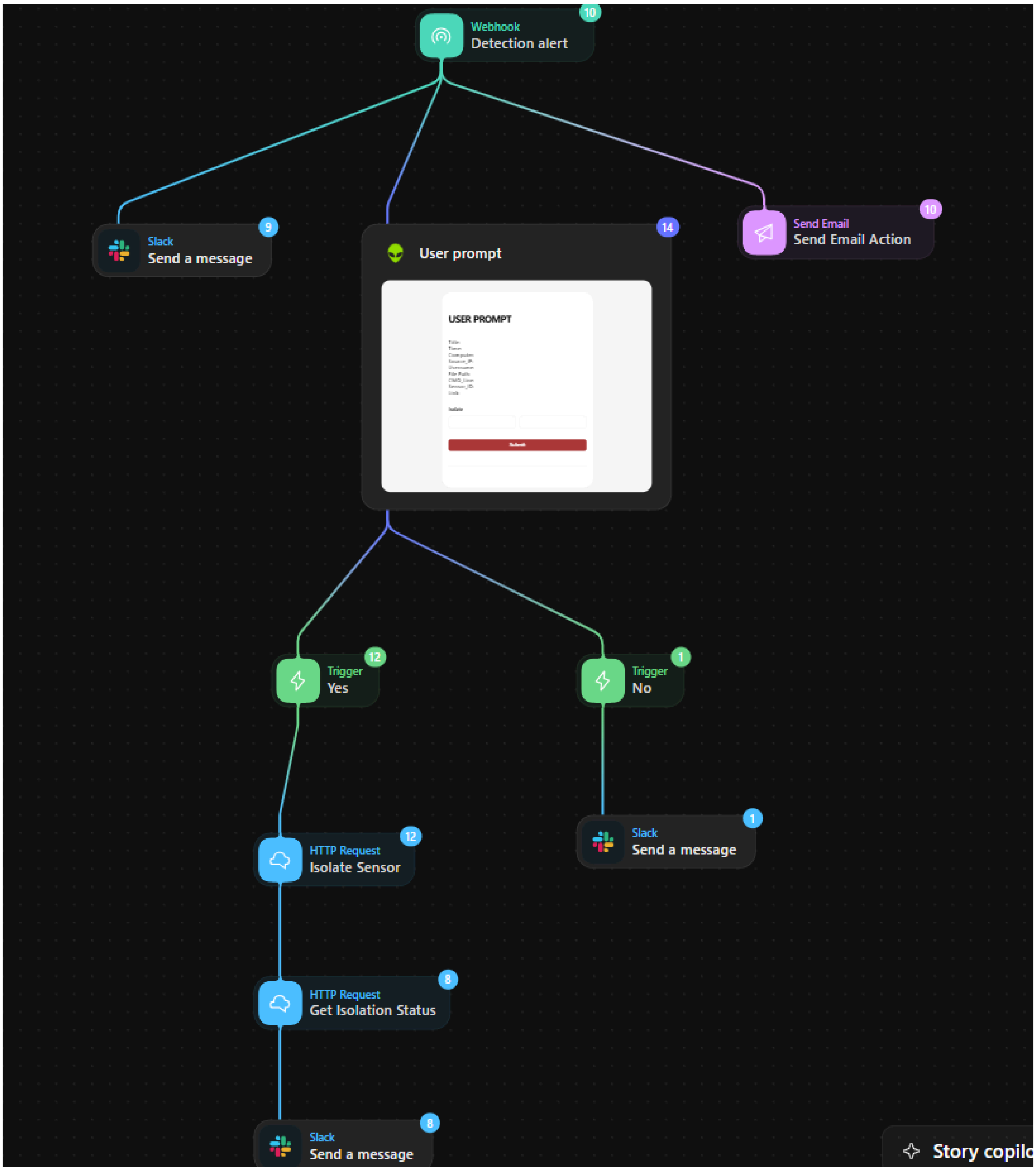


Time: 1769940180437
Computer: desktop-16ec9pn
Source_IP: 10.0.2.15
Username: DESKTOP-16EC9PN\itsok
File Path: C:\Users\itsok\Downloads\LaZagne.exe
CMD_Line: LaZagne.exe
Sensor_ID: cb9b89d0-480c-4a46-98d2-8f70743637d9
Link: https://app.limacharlie.io/orgs/d3e3d0cc-20db-40c7-9d48-ee4e73dd87ff/sensors/cb9b89d0-480c-4a46-98d2-8f70743637d9/timeline
time=1769940180&selected=77bb3d7b057798b3f067089c697f24d5

Tines APP 12:22 AM
The Computer: desktop-16ec9pn
is not isolated.

🔄 Final Playbook Workflow

1. Suspicious process executes on the endpoint
2. LimaCharlie detects the activity
3. Detection event sent to Tines via webhook
4. Slack and Email alerts sent to analysts
5. Analyst decision requested through Tines page
6. If YES → Endpoint isolated automatically
7. Isolation status verified and reported



This workflow reflects a real-world SOC playbook combining automation and analyst oversight.

🎯 Skills Demonstrated

- Endpoint Detection & Response deployment
- Detection engineering using real telemetry
- SOAR playbook design and automation
- Webhook-based platform integration
- API-driven security response actions
- Human-in-the-loop incident response
- Real-time SOC alerting and communication