



Contents

How To Construct an IOC Web Form.....	2
What is Tines?	2
Requirements	2
Tines Technical Resources	2
JSON	3
Tines Terminology	3
Tines Story.....	3
Tines Pages.....	3
Tines Access Control Methods for Tines Pages	3
Tines Credential.....	4
Tines Templates.....	4
Workshop Goals	4
TIP	5
Task 1 – Create a webform using the Tines Page tool	6
Task 2 – Create a Tines Page to redirect back to the submission page.....	10
Task 3 – Create a trigger action to determine if a CSV was uploaded	12
Task 4 – Create a Parse CSV action	13
Task 5 – Create an Explode IOCs Action	14
Task 6 – Create an Extract IOCs Action.....	16
Task 7 – Create a Throttle Action.....	17
Task 8 – Search for the VirusTotal or URLScan.io template	18
Task 9 – Create a Trigger to determine if IOC is malicious.....	22
Task 10 – Dates and Time (DATE Function vs. AI Automatic mode).....	24
Task 11 – Create an IOC Object	25
Task 12 – Create a Tines Resource called IP Sunset List	26
Task 13 – Use the Tines Template Append Element to a Resource	28
Task 14 – Implode IPs Action	31
Task 15 – Objects to CSV Action	33
Task 16 – Send Email Action	34
Task 17 – IOC Sunsetting – Filter IPs older than 21 days – AI Automatic Mode	35
Task 18 – Explode IPs.....	37



Task 20 – Tines Resource Remove Element Template	38
Task 21 – IOC Sunsetting – Filter IPs older than 21 days- Filter Function	39
Task 22 – IOC Sunsetting – Extract IPs	39
Task 23 – Enabling Send to Story	40
Task 24 – Send To Story Action – Block	41
Task 25 – Block or Remove Trigger	43
Task 26 – Event Transformation – IOC Blocking Pipeline	44
Task 27 – Event Transformation – IOC Removal Pipeline	45
Task 28 – Event Transformation – Exit	45
Task 30 – Send To Story Action – Remove	47

How To Construct an IOC Web Form

What is Tines?

Tines is a no/low code automation or SOAR (Security Orchestration Automation & Response) platform. A person with limited coding ability can create automation workflows to accomplish repetitive tasks, such as running reports, ingesting IOCs and SIEM alerts into a blocking pipeline, or categorizing incidents in ServiceNow. Any system that uses the REST API framework can integrate with the Tines platform.

In this lesson, we will understand how to construct a Tines Web Form (Tines Page) to submit IOCs to VirusTotal and URLScan.io.

Requirements

There is no previous technical or automation experience needed for this workshop. The only requirements for this workshop are signing up:

1. For a free Tines Community Edition: <https://www.tines.com/pricing/>
2. For a free personal VirusTotal and URLScan.io API keys:
 - a. <https://www.virustotal.com/>
 - b. <https://urlscan.io/>

Once logged into the Tines Community Edition tenant, import the Tines story (.json file) by following the directions in link: <https://www.tines.com/docs/stories/importing-and-exporting/>

Tines Technical Resources

- [Tines Pages](#)
- [Tines Key Features Explained](#)
- [Tines Documentation](#)



- [How to Create a Tines Resource](#)

JSON

JSON – JavaScript Object Notation: an open standard file and data interchange format consisting of key-value pairs and arrays.

Object – **One or more key-value pairs that describe or relate to something.** In the below screenshot, there is the payload response from the VirusTotal IP Addresses Search API endpoint. In the Tines Event, the body key has a value that consists of an object. We can identify objects by the **open {and the close}**. The data key has an object which possesses four key-value pairs:

1. "id": "8.8.8.8"
2. "type": "ip_address"
3. "links.self": "https://www.virustotal.com/api/v3/ip_addresses/8.8.8.8"
4. "attributes": object

```
"search_for_ip_address": {  
  "body": {  
    "data": {  
      "id": "8.8.8.8",  
      "type": "ip_address",  
      "links": {  
        "self": "https://www.virustotal.com/api/v3/ip_addresses/8.8.8.8"  
      },  
      "attributes": { ... }  
    }  
  }  
}
```

Tines Terminology

Tines Story

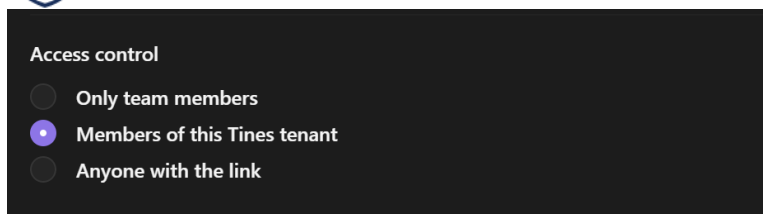
A collection of interconnected actions working to accomplish a certain task.

Tines Pages

Web pages, created in Tines, connect a workflow to user input. Tines pages can initiate a workflow by taking user input in the web form.

Tines Access Control Methods for Tines Pages

There are three methods to control access to Tines web pages URL:



The “Only team members” setting is the most restrictive one.

Tines Credential

A credential used in a workflow for authentication to an API endpoint or another system like AWS, Azure, CrowdStrike, etc. Tines has product-specific credential flows that will guide the user through the credential creation process, and for endpoints that use an OAuth2 token, it will automatically refresh it in the background once the credential is created. We will go through adding the VirusTotal credential.

Tines Templates

Tines provides pre-built templates for commonly used products to save you time while building your workflows.

Workshop Goals

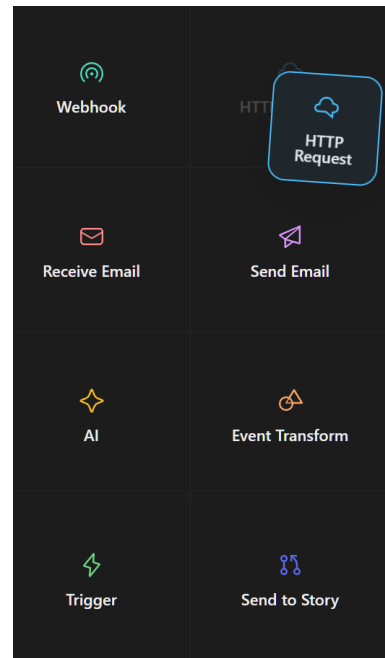
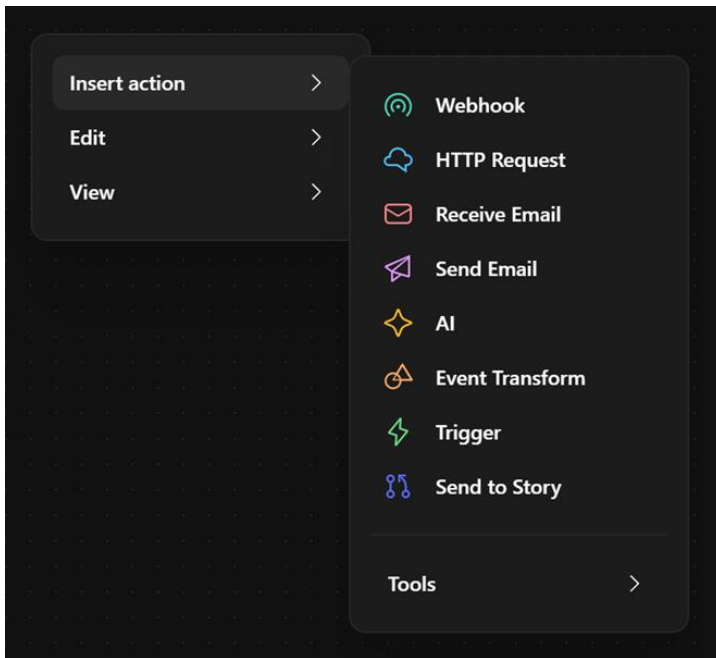
- Learn how to construct a Tines Page for submitting IOCs.
- Learn how to explode the elements of an array.
- Learn how to extract information submitted in a webform.
- Learn how to search the Tines Template Library to use preconfigured Tines Actions.
- Learn how to query VirusTotal and URLscan.io API endpoints.
- Learn about the Tines AI action.



TIP

engage • envision • enable

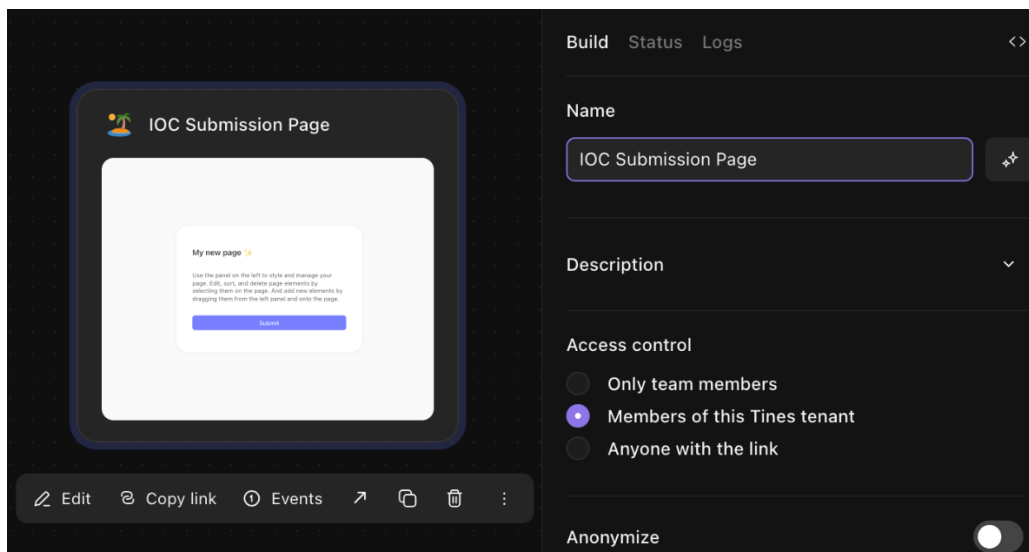
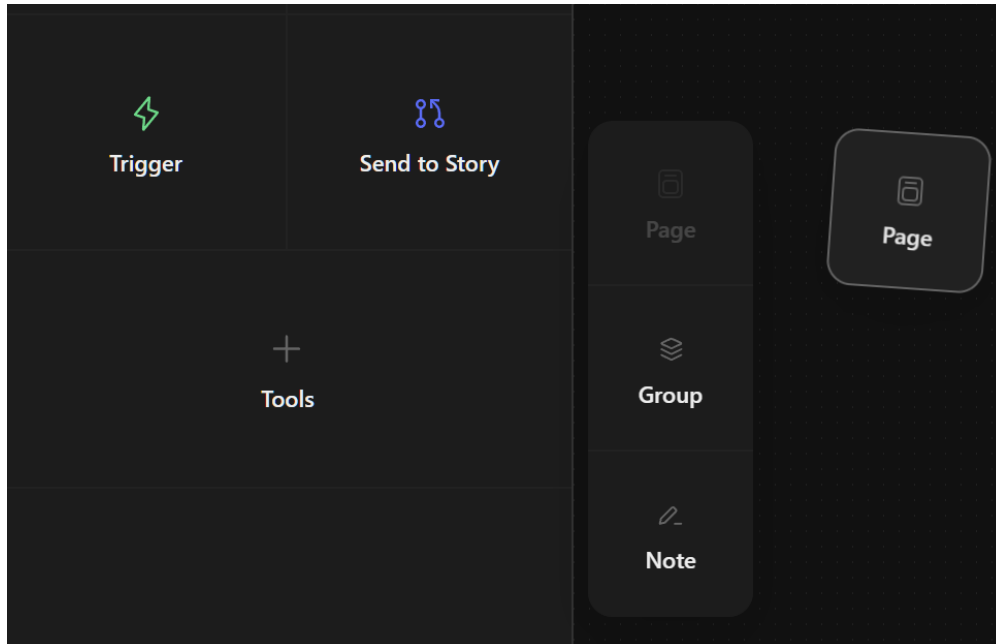
To create a Tines Action, right click the UI or drag and drop an Action from the left side of the UI:





Task 1 – Create a webform using the Tines Page tool


In the Tools menu, drag and drop a Tines page to the UI. Highlight the page. On the right side under the Build tab, rename it to IOC Submission Page.




Highlight the page and click the Edit button. On the right side, change the name of the Page. Toggle the logo button. Upload any image you like. For Appearance, choose light or dark mode and Large for Content Width.



Name



 IOC Submission Webform

Theme



Choose a theme 

Save as page theme

☒ Logo


 TinesLogoWhite-... Replace 


Appearance



Content width

Small Large Full

Background color 

Action color 

Click on Contents and type IPv4 IOC Submission Page.

Options **Conditions**  

Contents

IPv4 IOC Submission Page

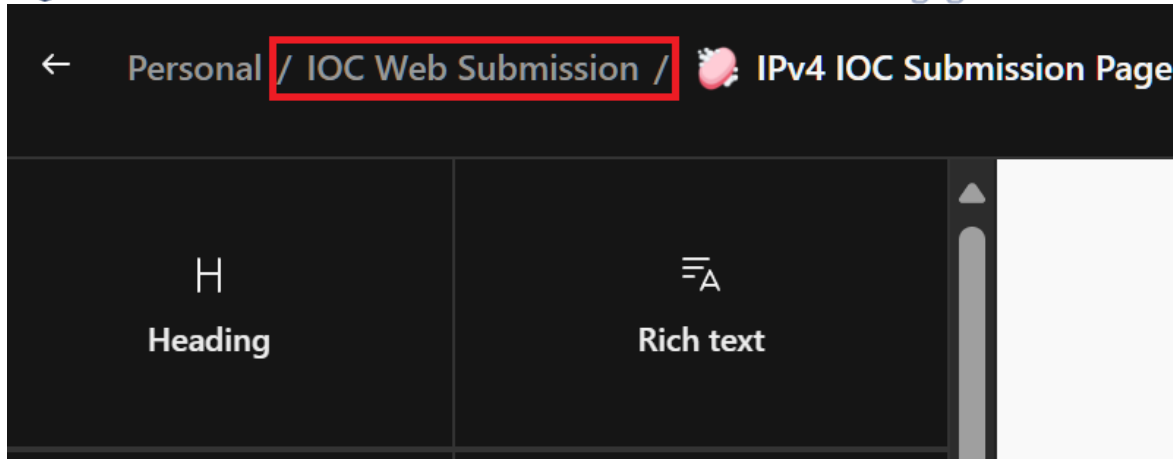


engage • envision • enable

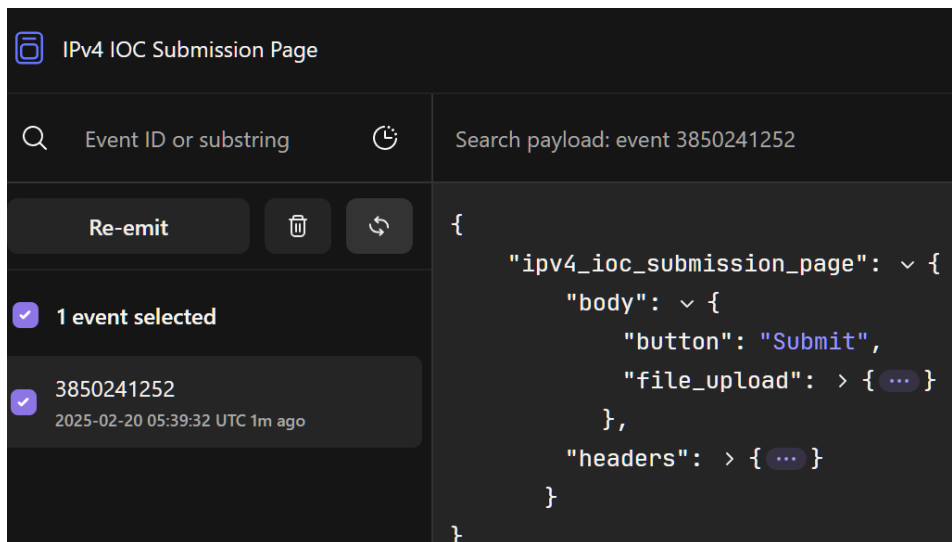
Under Input fields, drag and drop the File upload symbol onto the page. Then, drag and drop a Short text input field. Place them above the Submit button. For the Short text input field, type “IPv4 IOCs” into the Name field in the upper right-hand corner.

The Tines page should look similar to the page below.

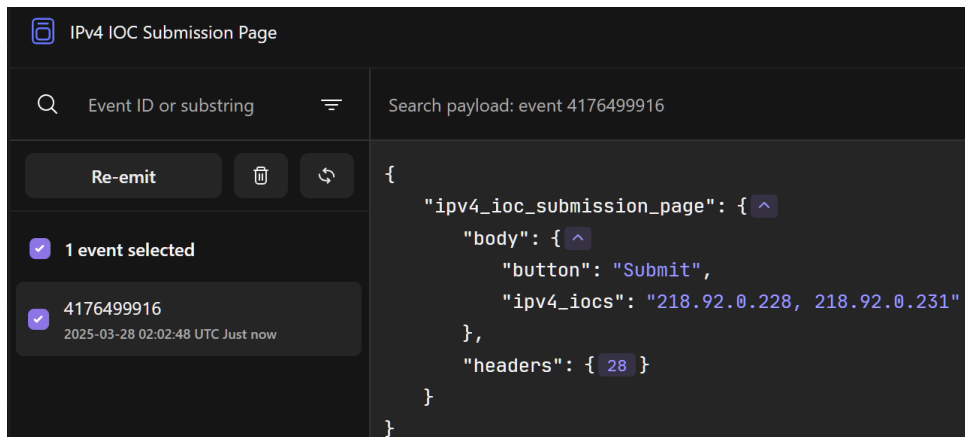
Once finished, exit the Page palette by clicking the IOC Web Submission link at the top of the page:



Highlight the page and click on the arrow next to Events to visit the web form. Upload the mal_ips.csv file to see if it works. Under Events, you should see something similar:



Enter some IP addresses into the IPv4 IOC fields and click the Submit button. See what event is generated.



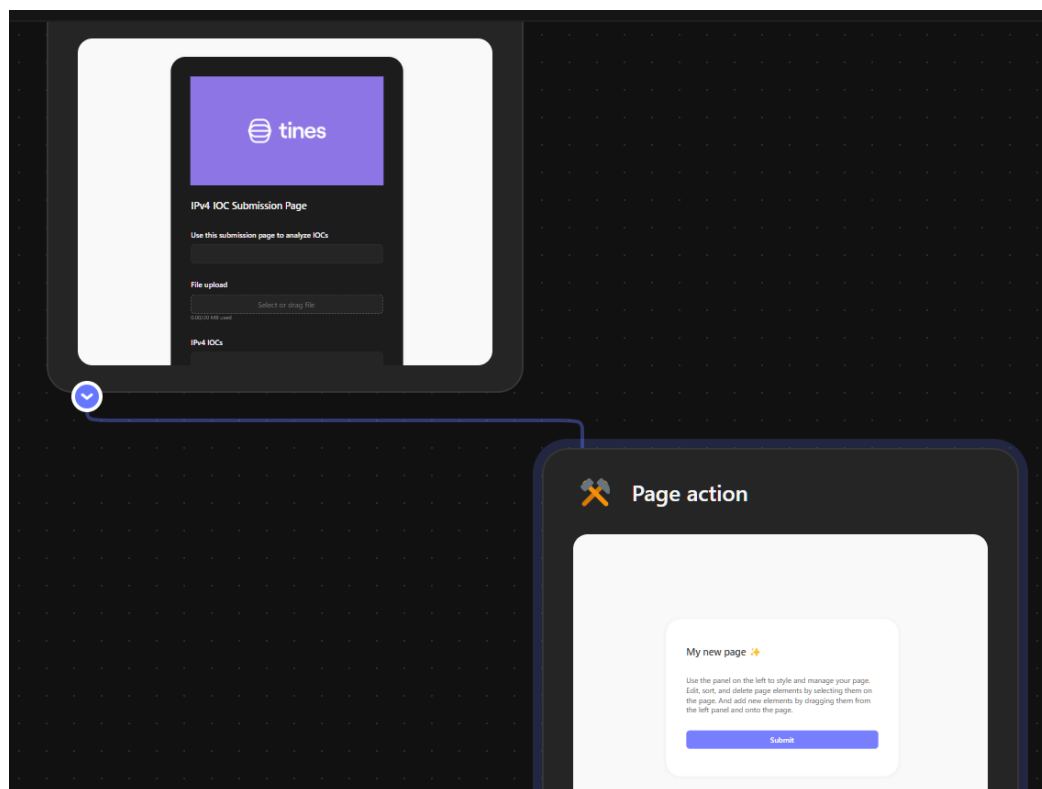


Task 2 – Create a Tines Page to redirect back to the submission page

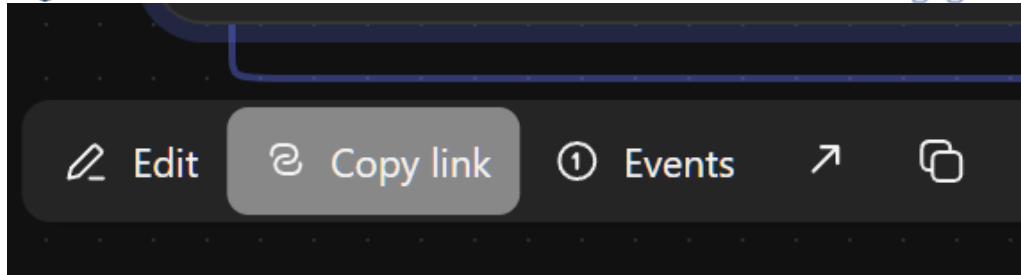
Click the page and change the Page behavior to “Move to next page”.

The screenshot shows the 'Page behavior' configuration panel in Tines. It has a dark theme. At the top, there's a dropdown menu set to 'Move to next page'. Below it is a toggle switch for 'Submit automatically', which is currently turned off. Further down is a text field for 'URL identifier' containing the value '760c8c07f0b06b659aa40ecb570261e3'. At the bottom is a text field for 'Loading message' containing the text 'Please wait...'.

Drag and drop another Page from Tools. Click on it. Under the Builder tab, go to Page behavior. Select “Redirect to next URL” from the drop-down list



Click the first page and click on “Copy link”. On the second page, past the link into the URL field:



Page behavior ⓘ

Redirect to URL ▼

Submit automatically 🔴

immediately ▼

URL identifier

460f48f21a4b8694a01347d46ea51e34

URL

<https://spring-water-5215.tines.com/pages/760c8c07f0b06b659aa40ecb570261e3/>

Visit the link for the first page, then type an ipv4 address, 8.8.8.8, into the IPv4 IOCs field. Then, click Submit. After submission, the original submission page will reload.

IPv4 IOCs

8.8.8.8

Submit



Task 3 – Create a trigger action to determine if a CSV was uploaded

Drag and drop a Trigger action. Under the Build tab, rename it to CSV Uploaded?

Build Status Logs

Name

CSV Uploaded?

Go back to the original event, when the csv was uploaded. Select it and click Re-emit.

IPv4 IOC Submission Page

Event ID or substring Search payload: event 3850241252

Re-emit [trash icon] [refresh icon]

1 event selected

Event ID	Timestamp
3858635709	2025-02-20 21:49:33 UTC 2m ago
3858510829	2025-02-20 21:39:30 UTC 12m ago
3858506799	2025-02-20 21:38:50 UTC 13m ago
3850241252	2025-02-20 05:39:32 UTC ~16h ago

```
{
  "ipv4_ioc_submission_page": {
    "body": {
      "button": "Submit",
      "file_upload": { ... }
    },
    "headers": { ... }
  }
}
```

In the Trigger action, navigate to the “ipv4_ioc_submission_page.body.file_upload.type”. Change the rule to “contains” and the value “text”.



Rules

Formula

`f ipv4_ioc_submission_page.body.file_upload.type`

contains

text

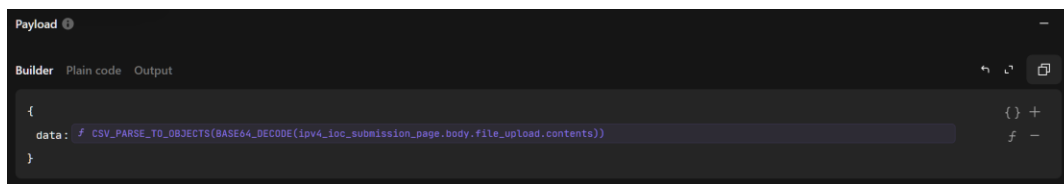
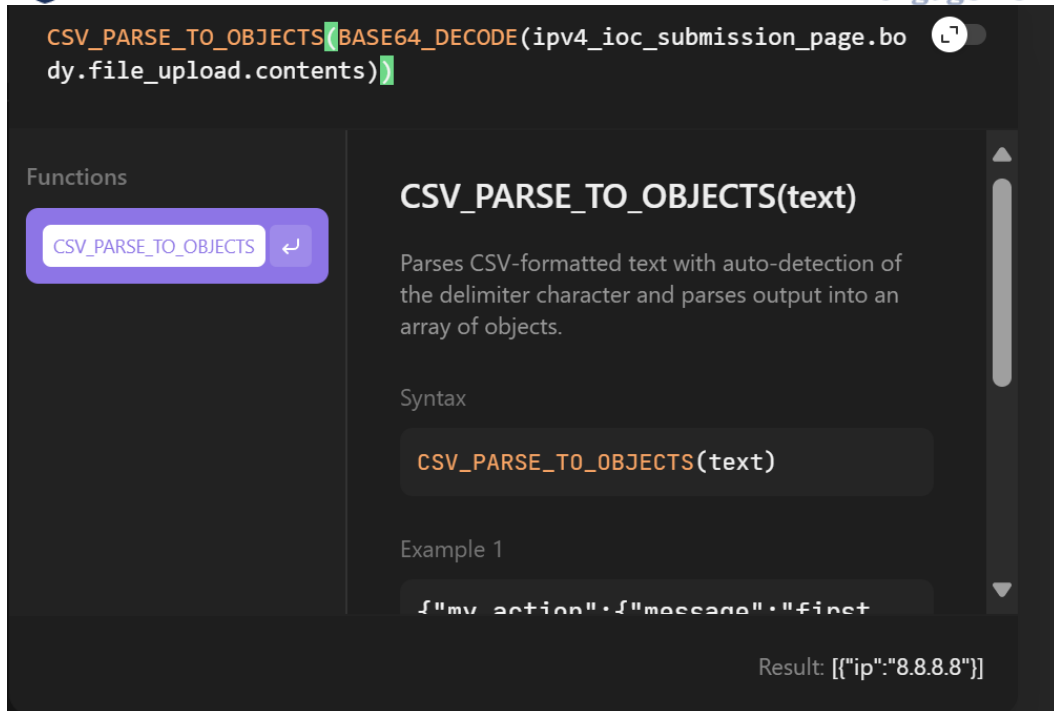
Under Events in the initial page action, re-emit the previous event. The Trigger should emit a new event since it is a match.

Task 4 – Create a Parse CSV action

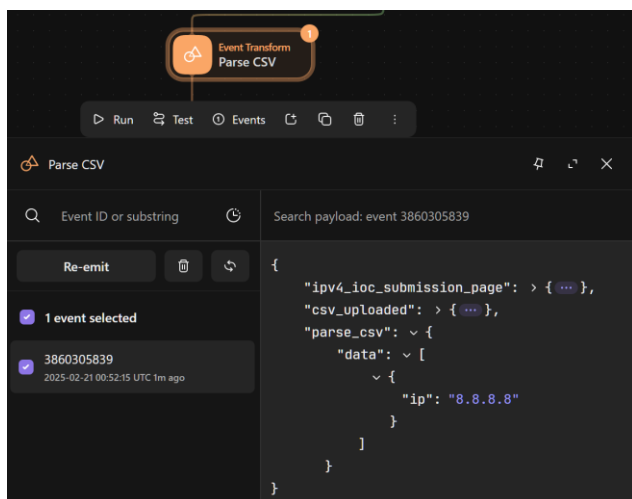
In the csv that was submitted, the header (column name) is “ip”.

	A	B
1	ip	
2	102.211.152.45	
3	152.53.119.28	

To parse the csv, the event transformation must use BASE64_DECODE function to decode the contents. Then, the CSV_PARSE_TO_OBJECTS function creates an ip object with ip as the key, and the submitted ipv4 address as the value. Use an Event Transformation action. Rename “message” to “data”. Then, configure the builder payload, using the formula type, as below:



Re-emit the event from the web submission form. The Parse CSV action should emit an array that contains the ip or ips submitted in the csv:



Task 5 – Create an Explode IOCs Action

Since the previous action created an array of ip objects, the next action needs to explode or emit each ip as a single event, to be processed by the subsequent VirusTotal action. Using the MAP function, we can



do that by referencing the parse_csv.data path using the “ip” property. Drag and drop an Event Transformation action on the UI. Set mode to explode and configure the builder payload as below. The path is the parse_csv.data as our array along with the MAP function and the “ip” as our property:

MAP(parse_csv.data, "ip")

Functions

MAP

MAP(array_of_objects, property)

Creates an array by extracting the values of a given key or path from an array of objects. If a property name has spaces, wrap it in square brackets and double quotes.

Syntax

MAP(array_of_objects, property)

Example 1

Result: ["8.8.8.8"]

Build Status Logs

Name

Explode IPs

Description

Mode

Explode

Path

f MAP(parse_csv.data, "ip")

To

ip

Limit

500

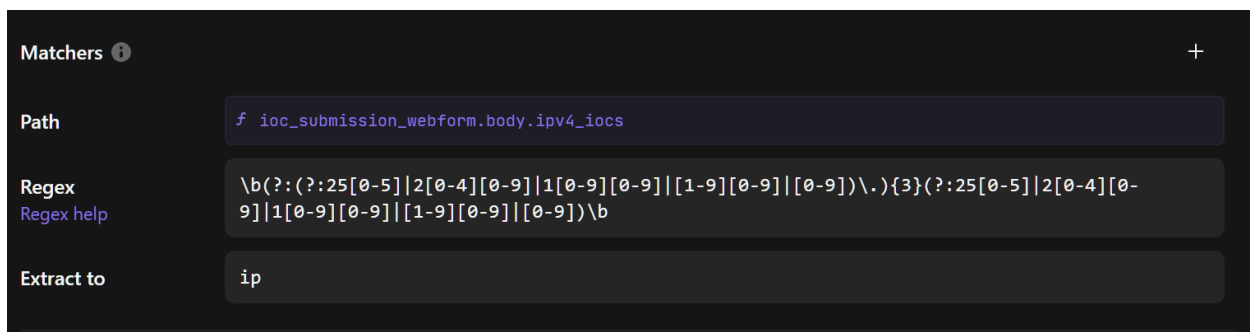


Task 6 – Create an Extract IOCs Action

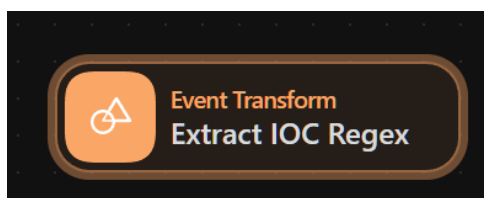
On the right side of the loop, we process the submitted IPv4 addresses by extracting them. Drag and drop another Event Transformation action, using Extract mode, and the body.ipv4_iocs payload as the path:



To check the validity of the IPv4 address, the matchers rule has the below regex:




The regex expressions can be found in this action:





Task 7 – Create a Throttle Action

We have to create a Throttle action since the personal VT API key, rate limits the number of requests per min, day, and month:

Access level	 Limited , standard free public API Upgrade to premium
Usage	Must not be used in business workflows, commercial products or services.
Request rate	4 lookups / min
Daily quota	500 lookups / day
Monthly quota	15.5 K lookups / month

Drag and drop an Event Transformation action and configure it as below:

Build **Status** **Logs**

Name
Throttle

Description
Describe your action..

Mode
Throttle

Interval
minute

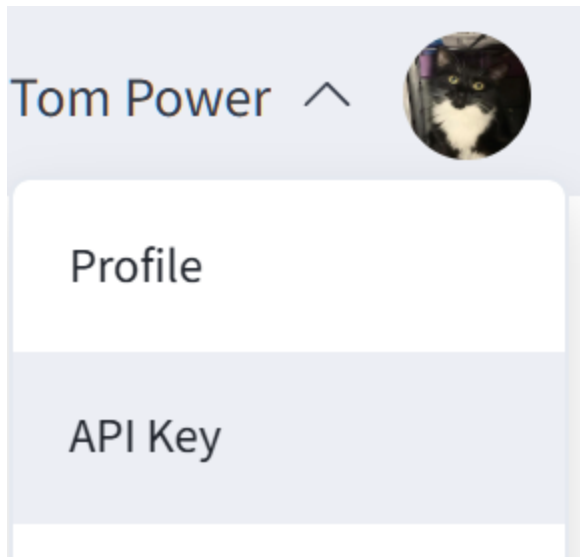
Runs per interval
4

Events per run
1

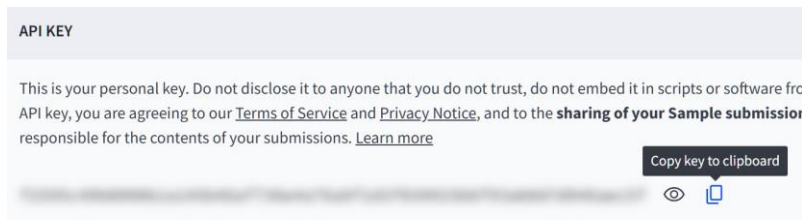


Task 8 – Search for the VirusTotal or URLScan.io template

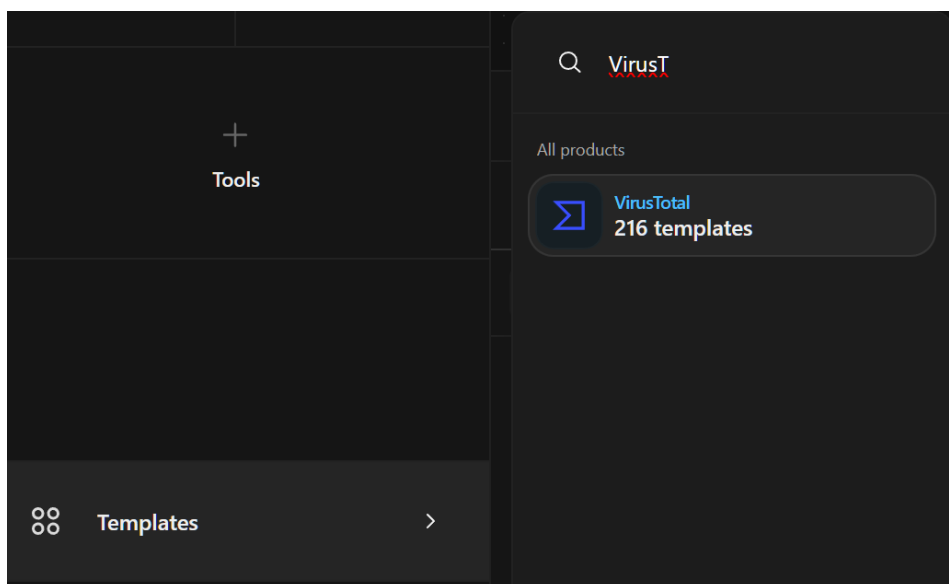
Log into your personal VT account. Navigate to your personal API key:



Copy it to the clipboard:

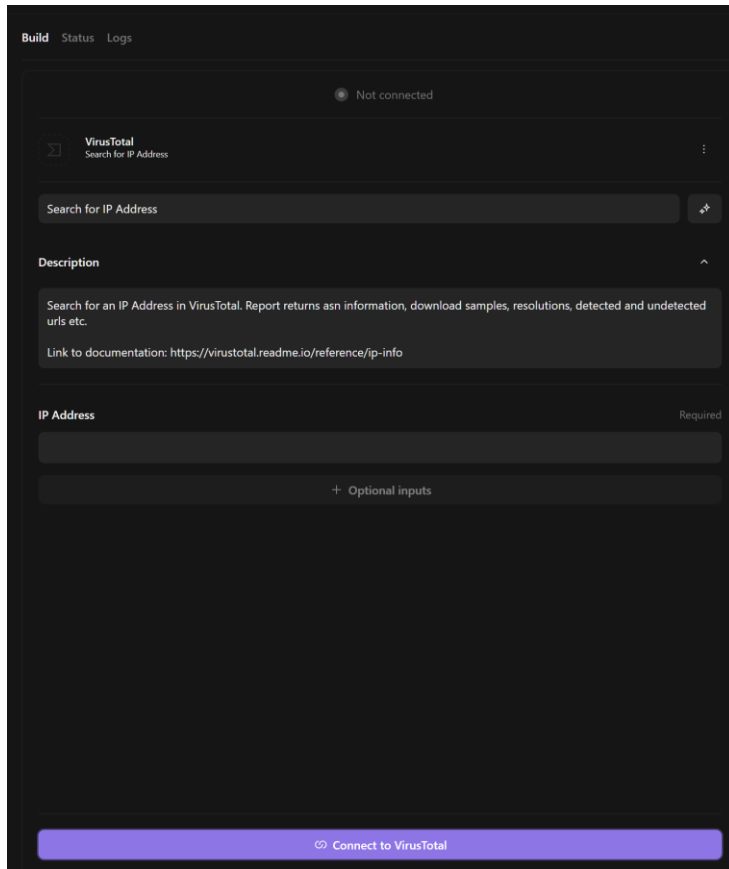


In the Tines Template section, search for VirusTotal:





Drag and drop the template on the storyboard. From the template choices, select Search for IP Address. Click on Connect To VirusTotal at the bottom of the Build tab:



Click on new connection. Paste your personal API key. Click Connect:



1. Get a VirusTotal API key

- Log in or sign up to [VirusTotal](#).
- Click on your profile on the top right-hand corner and select API key

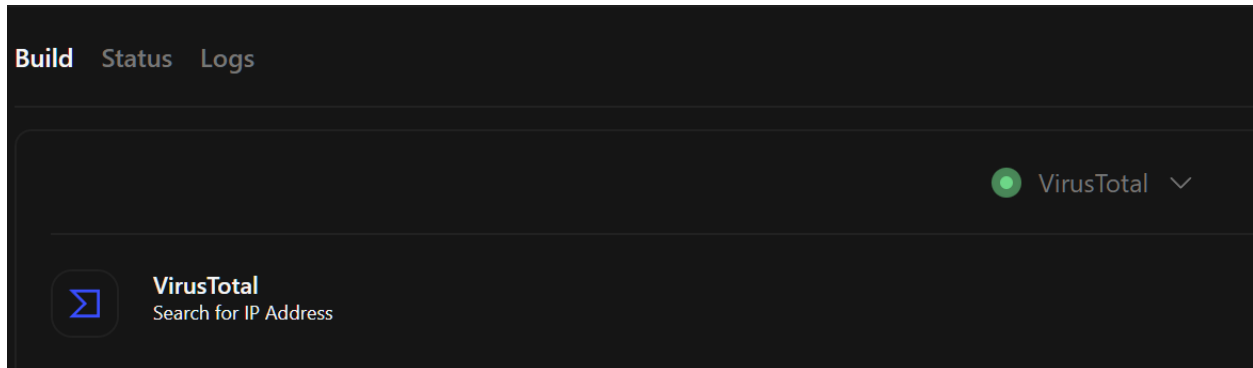
2. Copy and paste the API key into the field below

API key Required

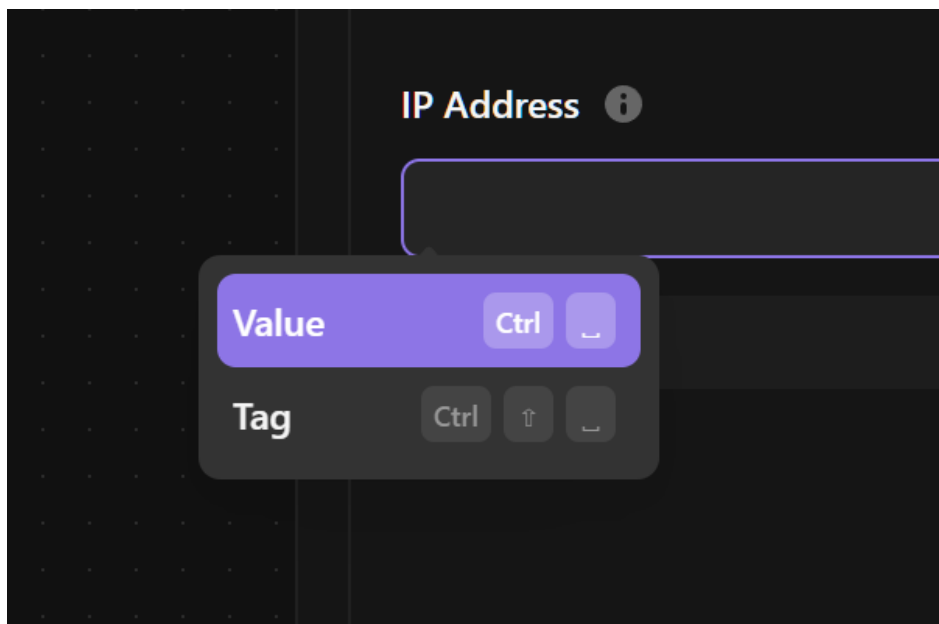
Connect



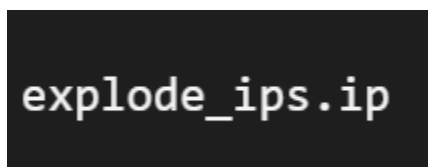
At the top of the action, you should see a green button next to VirusTotal:



To configure the action, click the plus in the IP Address field. Select Value:



Navigate to the explode_ips.ip path:



Visit the web submission form and enter the following IPs into the IPv4 IOCs field: 152.53.119.28, 2.57.122.32



IPv4 IOCs

152.53.119.28, 2.57.122.32

Submit

After clicking the submit button, we should see a status 200 for the search_for_ip_address action payload:

```
{
  "ipv4_ioc_submission_page": > { ... },
  "csv_uploaded": > { ... },
  "extract_ips": > { ... },
  "explode_ips": > { ... },
  "throttle": > { ... },
  "search_for_ip_address": ∨ {
    "body": > { ... },
    "headers": > { ... },
    "status": 200,
    "meta": > { ... }
  }
}
```

Check the Logs tab. We can see the API call under the VT Template for IP search:

Sending request to https://www.virustotal.com/api/v3/ip_addresses/2.57.122.32 with options:

```
{
  "url": "https://www.virustotal.com/api/v3/ip_addresses/2.57.122.32",
  "params": {},
  "query_params": {},
  "body": null,
  "headers": {
    "x-apikey": "*****",
  }
}
```



```
"User-Agent": "Tines (Advanced Security Automation; spring-water-5215.tines.com)",  
"Content-Type": null  
},  
"method": "get"  
}
```

Task 9 – Create a Trigger to determine if IOC is malicious

Drag and drop a Trigger action to the storyboard. In the Build tab, navigate to the `search_for_ip_address.body.data.attributes.last_analysis_stats.malicious`. Other options can be chosen, but I thought this was the easier way to decide whether or not an IOC is malicious if more than three VT vendors are reporting it as malicious. If a cyber team can tolerate more false positives, lower the number.

Build Status Logs

Name

Malicious?

Description

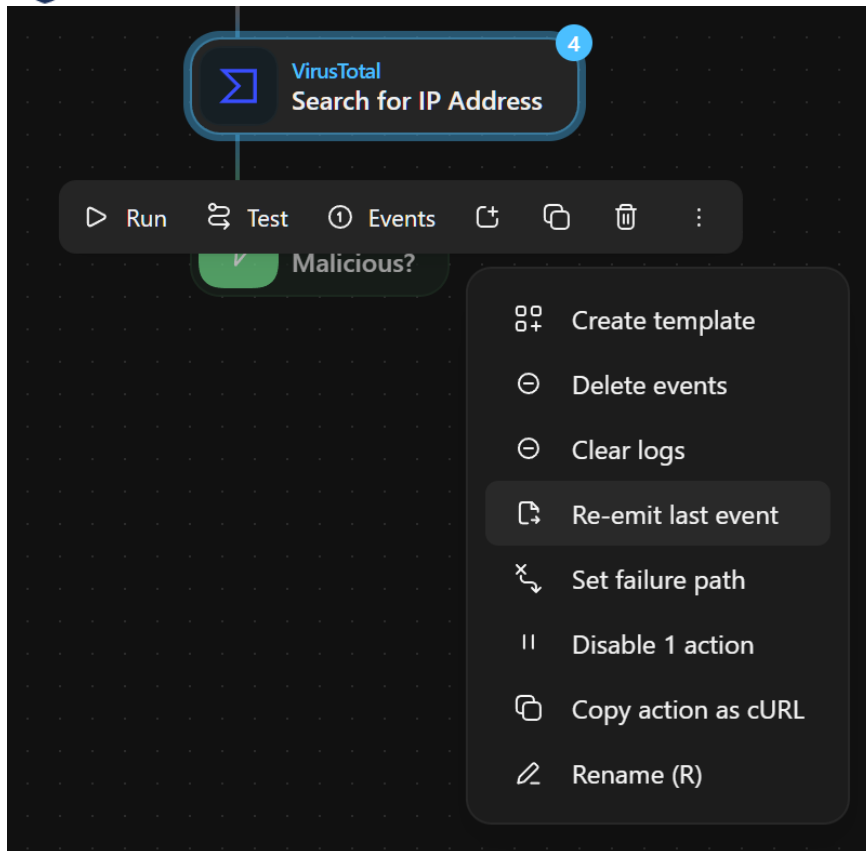
Rules ⓘ

f search_for_ip_address.body.data.attributes.last_analysis_stats.malicious

is greater than

3

Go to the previous event in the VT IP Search action. Remit the event to check the Trigger action.



We see that the rule matched, so the IOC is considered malicious. The number 3 was the cutoff. There were 12 VT vendors that considered the IOC malicious.

```
"malicious": {
  "rule_matched": true,
  "rule_results": [
    {
      "type": "field>value",
      "path": ">=search_for_ip_address.body.data.attributes.last_analysis_s...",
      "matched": true,
      "field": 12,
      "value": "3"
    }
  ]
}
```



Task 10 – Dates and Time (DATE Function vs. AI Automatic mode)

These actions convert date and time actions are not necessary part of the workflow, but I want to emphasize how important time is and introduce the DATE function since it will be used in many workflows. In addition, we will see how easy it is to use the AI automatic mode in event transformation.

Drag and drop an event transformation on the UI. Use Message mode for the first event transformation. Configure the action as seen below:

```
{
  mod_date: f DATE(search_for_ip_address.body.data.attributes.last_modification_date, "%Y-%m-%d", "America/Vancouver")
  analysis_date: f DATE(search_for_ip_address.body.data.attributes.last_analysis_date, "%Y-%m-%d", "America/Vancouver")
}
```

Re-emit the last VT event that was malicious and check what the dates are. Drag and drop another Event Transformation action. Select Automatic mode. For the input, navigate to the `search_for_ip_address.body.data.attributes.last_modification_date` path. Click the Generate button after typing in the prompt to see what the output is. Compare the date to the Tines DATE Function from the other Event Transformation action.

Automatic mode

`f search_for_ip_address.body.data.attributes.last_modification_`

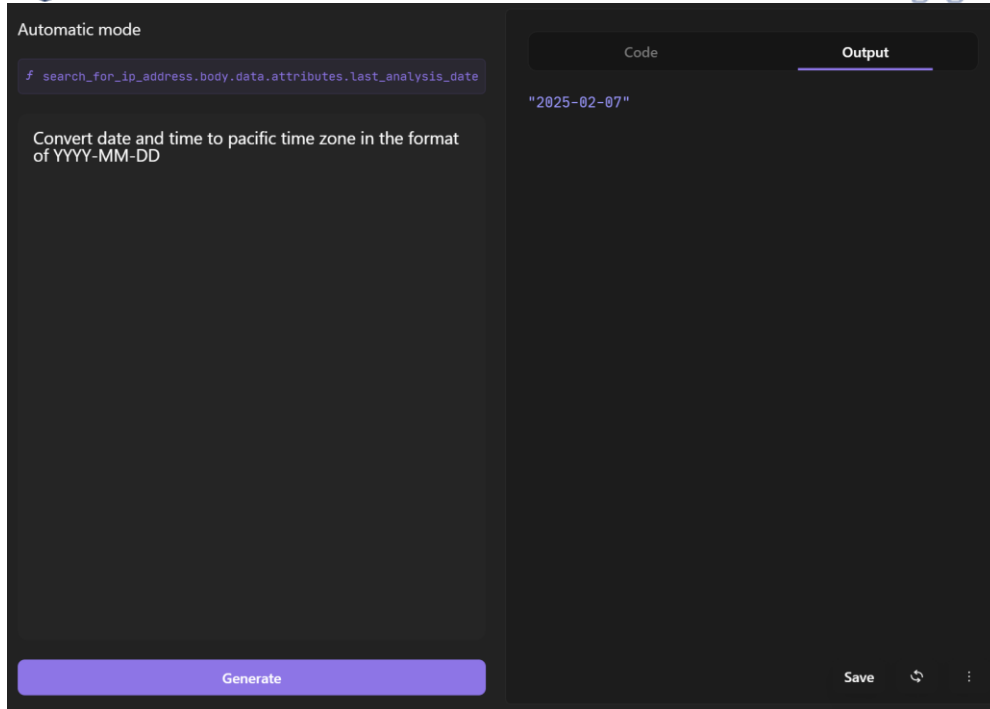
Convert date and time from Unix time to time in format of YYYY-MM-DD for the Pacific time zone.

Code Output

"2025-02-21"

Generate Save ↺ ⋮

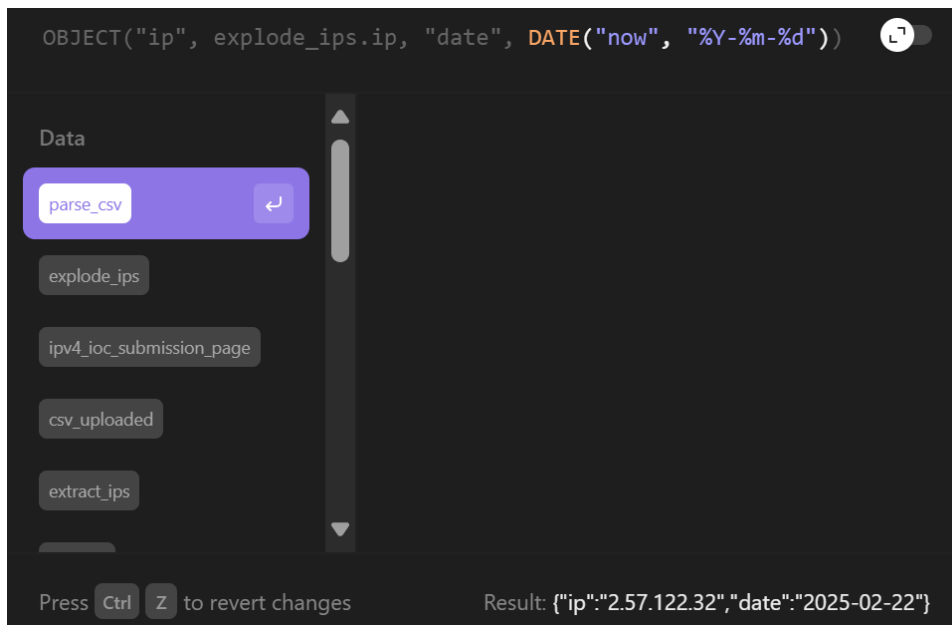
Use another Event transformation action in Automatic mode and navigate to `search_for_ip_address.body.data.attributes.last_analysis_date`. Click the Generate button after typing in the prompt:



Task 11 – Create an IOC Object

With the OBJECT function, we can create an object from any data emitted from a Tines action. The IOC object created contains two key-value pairs: ip and date. The action provides the data needed to write to a Tines resource in order to lifecycle IP addresses.

Drag and drop an Event Transformation to the UI. Using the formula, type OBJECT and select the OBJECT function. Within the parentheses, type “ip”, explode_ips.ip, “date”, DATE(“now”, “%Y-%m-%d”). We can see the results of the object with two key-value pairs:





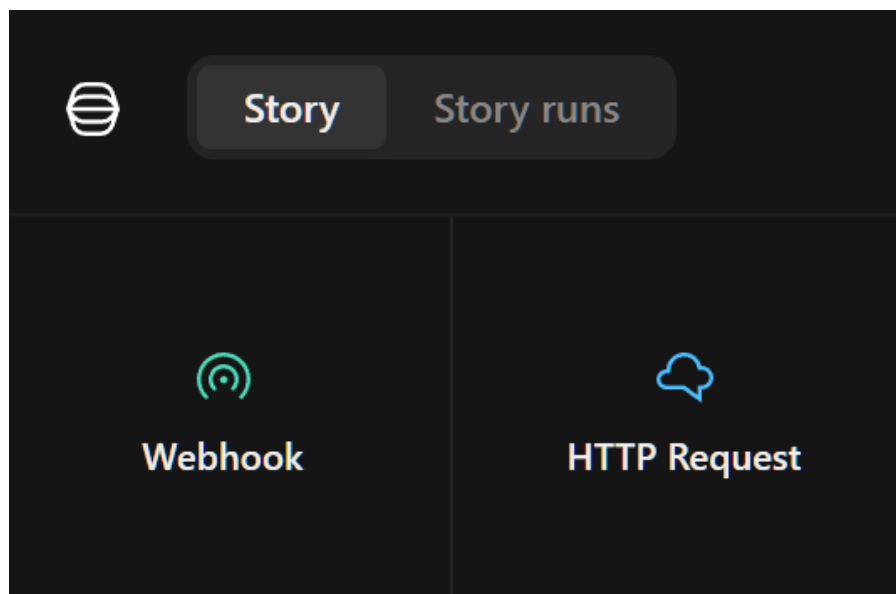
Rename message to ioc_object:

```
Builder Plain code Output
{
  ioc_object: f OBJECT("ip", explode_ips.ip, "date", DATE("now", "%Y-%m-%d"))
}
```

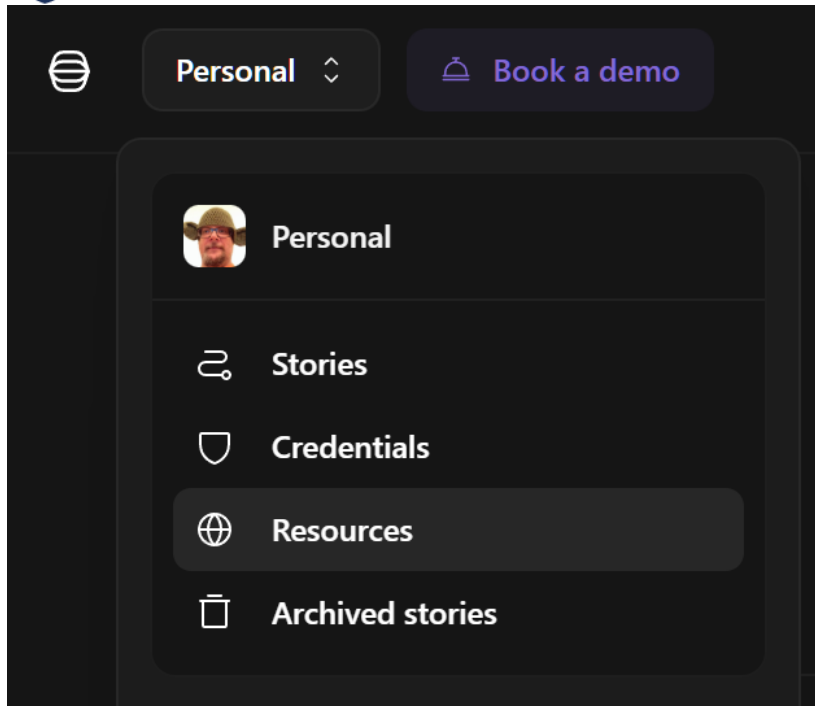
Task 12 – Create a Tines Resource called IP Sunset List

Tines Resources allow the storage and reuse of information, that is located outside the workflow. Information, located in a Resource, can be used or referenced in any Tines action.

To create a Tines Resource, click the Tines Dashboard symbol next to Story.



The Resource can be created under Personal or the Tines Team. Select Resources.



Click the New button. Name the Resource the ip_sunset_list. In the Plain code section, create a blank array by typing []. Choose to where the resource can be used under (Personal) or with all teams. Click Save Resource.

New resource

Name
ip_sunset_list

Description
Resource for IOC Lifecycle Management - Malicious IPs

Builder Plain code File

1 []

Access
Control where this resource can be used.

☒ This team (Personal)

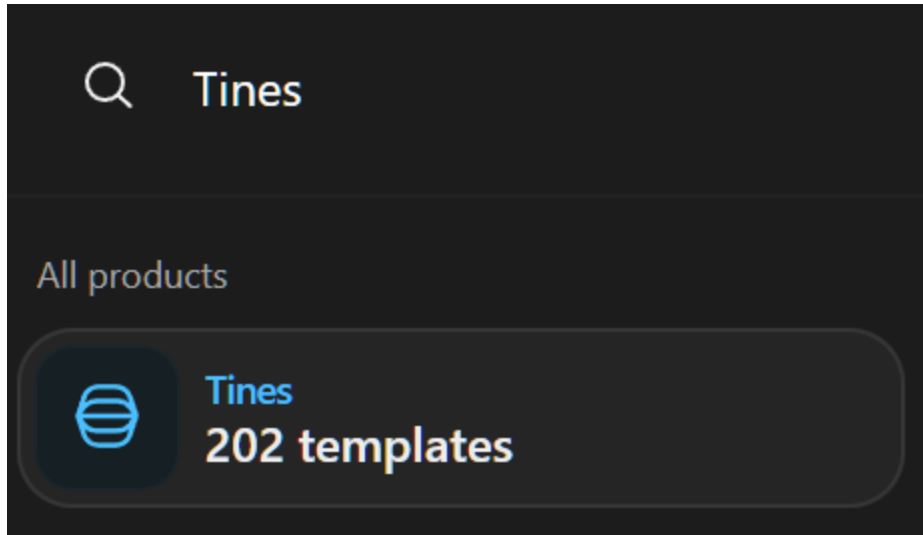
☐ All teams

Save resource

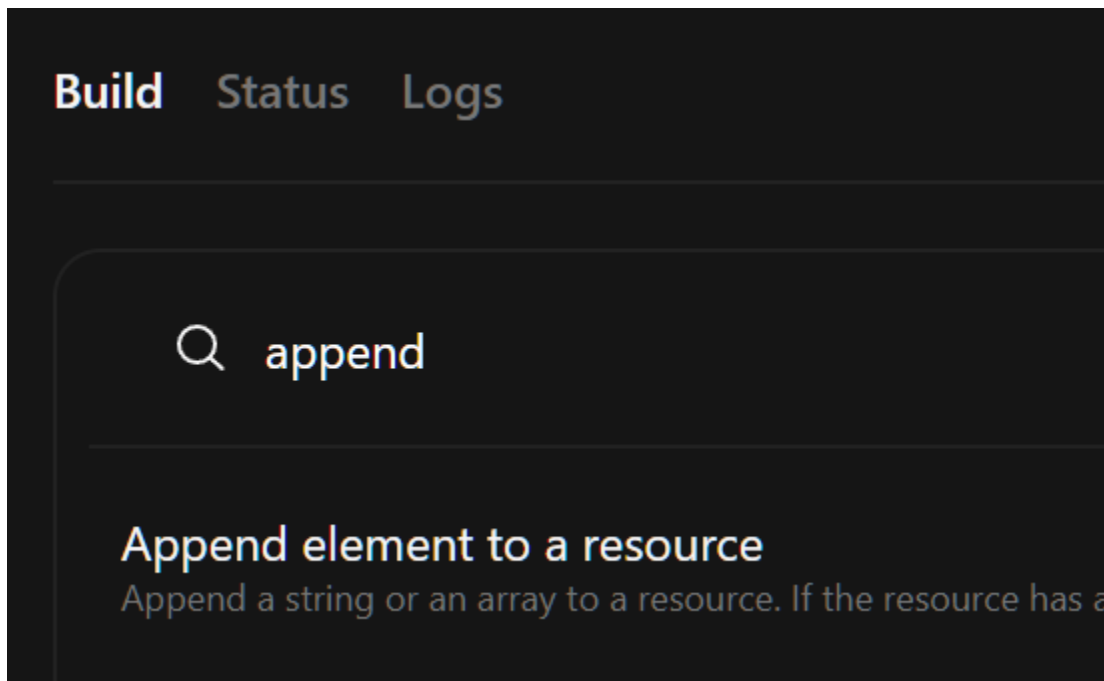


Task 13 – Use the Tines Template Append Element to a Resource

In the lower left-hand corner of the UI, click Templates and type Tines in the search bar. Drag and drop the template to the UI:



Search for append and select the Append element to a resource:



Connect this action to the previous action. Remove the Text value field by clicking the minus symbol. Under Resource ID, click the path. Type resource.ip_sunset_list.id:



INFO.resource.ip_sunset_list.id

Data

id

Functions

DIVIDED_BY

IN_CIDR

IS_VALID_JSON_SCHEMA

JSON_SCHEMA_VALIDATE

Press **Ctrl** **Z** to revert changes

Result: 41451

In the Builder under Array value, click the array symbol and choose formula. In the formula, choose the create_ioc_object.ioc_object:

Tenant domain Required

META.tenant.domain

Resource ID

f INFO.resource.ip_sunset_list.id

Array value


Builder Plain code Output


f create_ioc_object.ioc_object

+ Optional inputs


At the bottom, click Connect and select New connection. Select the New API Key option. I chose Tenant owner which can access and write to any resource located in the tenant:



 Use formula

 New connection

Provide a name and description. Click Save.

 **New tenant owner API key**

Name

Description

In the IOC web form, submit IP, 2.57.122.32. Under Events, we should see the following object created:

"ip": "2.57.122.32"

"date": "today'sdate"



```
"append_element_to_a_resource": {  
  "body": [  
    {  
      "ip": "2.57.122.32",  
      "date": "2025-02-22"  
    }  
  ],  
  "headers": { ... },  
  "status": 200,  
  "meta": { ... }
```

We can go back to Resources and see what it looks like. We have an array (a list) with an IOC object containing two key-value pairs:

Details Actions Lock resource

Name
ip_sunset_list

Description
Resource for IOC Lifecycle Management - Malicious IPs

Builder Plain code File

```
[  
  {  
    ip: 2.57.122.32  
    date: 2025-02-22  
  }  
]
```

Update resource

Task 14 – Implode IPs Action

The Implode mode of an Event Transformation action collects exploded events. After collecting all the events or waiting for a certain amount of time, the Implode action will emit one event. For example, if 20



IPs are exploded into individual events to be processed by the VirusTotal search for IP action, the Implode action will count the number of events collected. After 20 events collected, the action will emit an event to initiate the next action in the workflow. If an event gets “lost”, the Implode action will not fire.

Configure the implode action as seen below:

Build Status Logs

Name

Implode IPs

Description

Mode

Implode

Item path

f explode_ips.ip

Identifier path

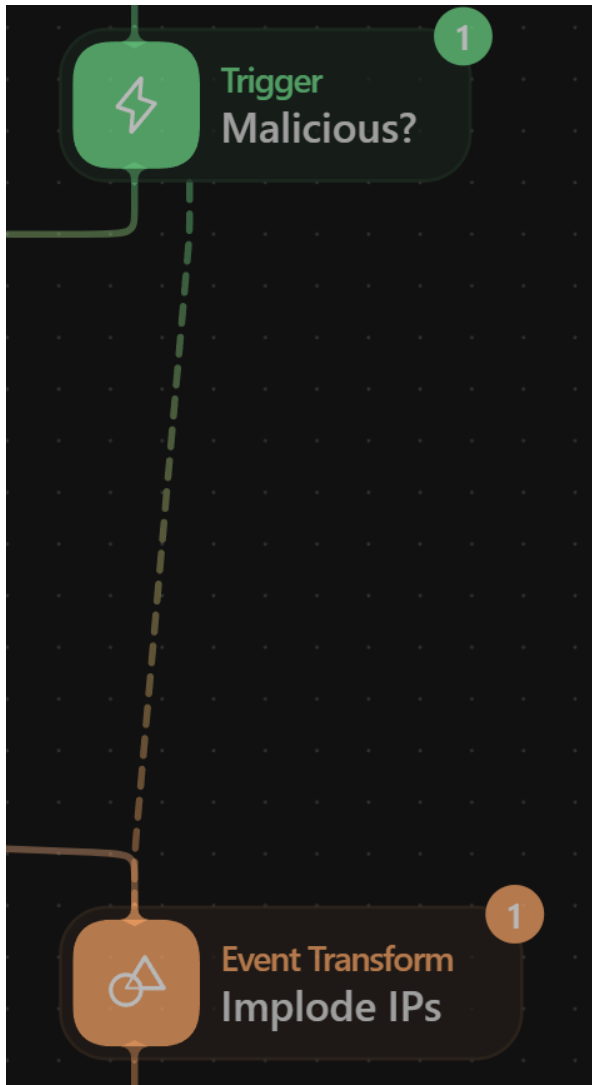
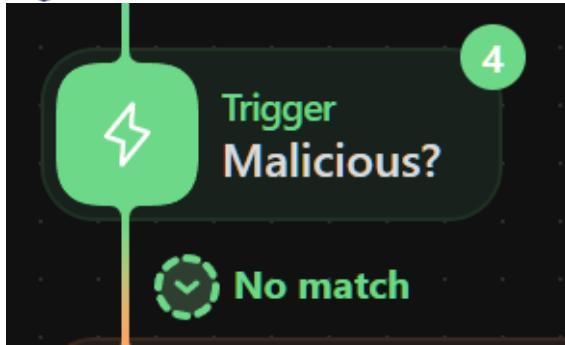
f explode_ips.guid

Size

f explode_ips.size

We will submit a safe IOC, such as 8.8.8.8, and see what happens with the implode action, since the IP will not be deemed malicious by the Trigger action.

Then, connect the No match of the Malicious? Trigger action by attaching it to the implode action.



Task 15 – Objects to CSV Action

To create the csv to email, drag and drop another Event Transformation action in message mode. Change the builder type to formula and click on it. Type and select Resource. Choose the ip_sunset_list. Move the cursor to the beginning of the word resource and type OBJECTS. Pick the OBJECTS_TO_CSV Function:



Build Status Logs

Name

Objects to CSV

Description

Mode

Message only

Loop

Payload

Builder Plain code Output

```
{
  message: f OBJECTS_TO_CSV(RESOURCE.ip_sunset_list)
}
```

Task 16 – Send Email Action

Drag and drop a Send email action to the UI. Configure the Sender name and subject line. Click Option and select ATTACHMENTS:

Search options...

ATTACHMENTS

Attachments to send with the email. For each file, specify "filename", its contents encoded in Base64 "base64encodedcontents" and "content_type", in an object. The "content_type" is "auto" by default and automatically infers the content-type of the file.

BCC

Specify the email address or list of email addresses that should be BCC'ed on the email.

CC

Specify the email address or list of email addresses that should be CC'ed on the email.

CONTENT TYPE

Provide a content type for the email by

[View docs](#)

Sender name

Tines IOC Management Workflow

Subject

Malicious IOCs

Body

A real email body could go here

Advanced HTML mode

☒

+ Option



In the File Name section, click on `download_file.body.filename`. Remove the words and type TODAY. Select the TODAY function. Next to `TODAY()`, type “`_Malicious_IOC.csv`”. Click on `download_file.body.base64encodecontents`. Navigate to the `objects_to_csv.message` path. Type `BASE64_ENCODE` and select that function. Check the Result to see if the encoding worked.

BASE64_ENCODE(objects_to_csv.message)

Functions

BASE64_ENCODE

BASE64_ENCODE(text)

Encode text using the Base64 encoding algorithm.

Syntax

BASE64_ENCODE(text)

Example 1

```
{"my_action":{"message":"hello world"}}
```

Press **Ctrl Z** to revert changes Result: "aXAsZGF0ZQoyLjU3LjEyMi4zMiw yMDI1LT AyL..."

Attachments

File name: { TODAY() } _Malicious_IOC.csv

Base64 contents: { BASE64_ENCODE(objects_to_csv.message) }

Content type: auto

Content ID: Optional

Task 17 – IOC Sunsetting – Filter IPs older than 21 days – AI Automatic Mode

After uploading the malicious IPs csv file to the submission form and letting the workflow process each IP address, check the IP Sunset List resource. It is populated with the malicious IOCs. At UBC, after submitting the IOC, there is a workflow that submits it to an IOC blocking pipeline where the malicious IOC is added to back-end security solution such as EDR or a NGFW.



Change the date, older than 21 days, on some of the IOCs. Make a note on which submission dates were changed. Then, click Update resource.

```
1 v [
2 v   {
3     "ip": "102.211.152.45",
4     "date": "2025-02-24"
5   },
6 v   {
7     "ip": "152.53.119.28",
8     "date": "2025-02-24"
9   },
10 v  {
11    "ip": "2.57.122.32",
12    "date": "2025-01-24"
13  },
14 v  {
15    "ip": "2.57.122.189",
16    "date": "2025-02-24"
17  },
18 v  {
19    "ip": "2.57.122.186",
20    "date": "2025-02-24"
  }
```

Update resource

Drag and drop an event transformation and select Automatic mode. Click on Input and navigate to the ip sunset list resource. Type “Extract IPs from objects older than 21 days”. Click on the Generate button. Check the output to see if it matches the IPs that you changed the date in the previous step. Click Save.

Automatic mode

RESOURCE.ip_sunset_list

Extract IPs from objects older than 21 days.

Generate

Code Output

```
[
  "2.57.122.32",
  "2.57.122.190",
  "218.92.0.198",
  "218.92.0.228"
]
```

Save ↺ ⋮



Run the Action. Under Events, there should be an array with a list of IPs.

Task 18 – Explode IPs

Configure an Event transformation action as below:

The screenshot shows a configuration form for an 'Explode IPs' action. The form has a dark theme. The fields are: Name (Explode IPs), Description (empty), Mode (Explode), Path (f filter_ip_sunset_list_21_days_old_automatic_mode.output), To (ip), and Limit (500).

Name	Explode IPs
Description	
Mode	Explode
Path	f filter_ip_sunset_list_21_days_old_automatic_mode.output
To	ip
Limit	500

Run the action. The action should emit each IP address as a single event.

Task 19 – Throttle Event Transformation

Drag and drop an event transformation action and configure it as seen below:

The screenshot shows a configuration form for a 'Throttle' action. The form has a dark theme. The fields are: Mode (Throttle), Interval (minute), Runs per interval (10), and Events per run (1).

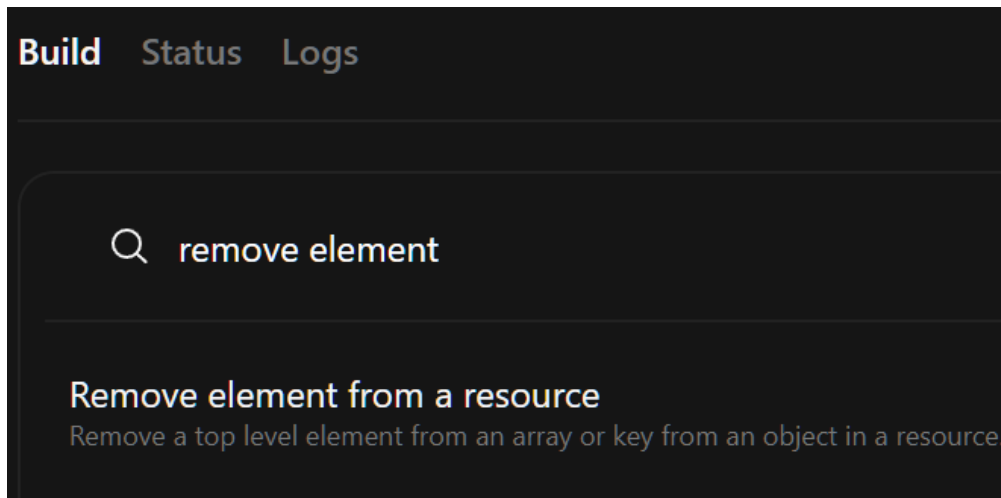
Mode	Throttle
Interval	minute
Runs per interval	10
Events per run	1



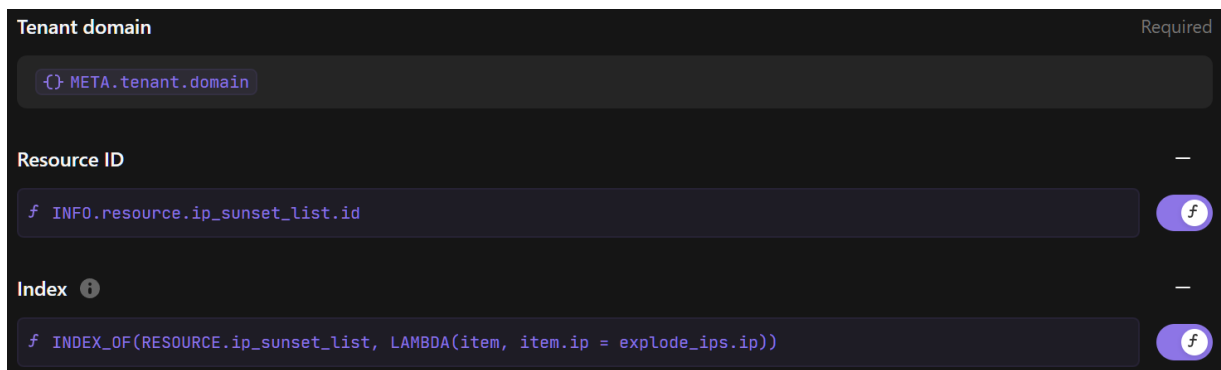
We need to throttle the emitted IPs to the next action, where the action will check what index, the IP exists in the resource. Once the index is obtained, it will remove it. If the events are not throttle, there is a race condition with the index of the IP. It will change before the next the event is process, so the throttle gives the resource time to update the indexes of the IP addresses.

Task 20 – Tines Resource Remove Element Template

Drag and drop a Tines Resource template. Search and select remove element from a resource:



Remove the Key field by clicking the minus sign. For Resource ID, navigate to the IP sunset List resource and select the ID. To remove the IP object (element) from the IP sunset list, use the INDEX_OF function to find the element in the resource. To find the index of the IP, use the LAMBDA function to create a custom function to check the resource (item.ip) for the IP that is exploded. So, this action checks the resource to know what the index (the location of the IP object within the resource) for the matching exploded IP. Once found, the action removes the element from the ip_sunset_list.



Check the IP Sunset List Resource to see if the IP was removed.



Task 21 – IOC Sunsetting – Filter IPs older than 21 days- Filter Function

Drag and drop an Event Transformation action to the UI. Rename message to data. Make sure formula (f) is the type. Click the field and type RESOURCE. Select the ip_sunset_list. Type FILTER and select the FILTER function. Type a comma after list, then type LAMBDA to select the LAMBDA function. Type item, DATE_DIFF to select the date differential function. Type item.date, DATE("Today", "%Y-%m-%d")) and to pass the value by typing |> and compare it to the number of days greater than 21. % represents the value from the previous functions, so it is the number of value compared to 21.

The screenshot shows the 'Builder' interface with tabs for 'Builder', 'Plain code', and 'Output'. The 'Builder' tab is active, displaying a configuration for a Filter function. The 'data' field is set to 'f FILTER(RESOURCE.ip_sunset_list, LAMBDA(item, DATE_DIFF(item.date, DATE("Today", "%Y-%m-%d")) |> %.days > 21))'. The 'Plain code' tab shows the corresponding JSON code:

```
{
  data: f FILTER(RESOURCE.ip_sunset_list, LAMBDA(item, DATE_DIFF(item.date, DATE("Today", "%Y-%m-%d")) |> %.days > 21))
}
```

Task 22 – IOC Sunsetting – Extract IPs

Drag and drop another Event Transformation to the UI. Select Automatic mode and for the input, navigate to the previous output. Type "Extract the value from the ip key" and click Generate button. Check the output and click Save.

The screenshot shows the 'Automatic mode' interface. On the left, the 'Automatic mode' tab is active, displaying the input 'f filter_ip_sunset_list_21_days_old_filter_function.data' and the description 'Extract the value from the ip key'. On the right, the 'Code' tab is active, showing the Python code:

```
1 def main(input):
2     ips = [item.get('ip') for item in input]
3     return ips
```

 At the bottom, there are buttons for 'Generate', 'Save', and a refresh icon.



Check the event to see if the IPs were extracted into a single array:

Extract IPs

Event ID or substring

Search payload: event 3979869969

Re-emit

1 event selected

3979869969
2025-03-04 05:39:36 UTC 5m ago

```
{
  "filter_ip_sunset_list_21_days_old_filter_f... : > { ... },
  "extract_ips": {
    "output": [
      "152.53.119.28",
      "134.122.0.63"
    ]
  }
}
```

Task 23 – Enabling Send to Story

Go to the Webhook-SendToStory story. Drag and drop a webhook action and Event Transformation event. Name the ET action Exit. Connect the two actions.

Click on a blank spot on UI. On the right-side of the page, there is a Send to Story button:

Enable webhook API responses

Send to Story

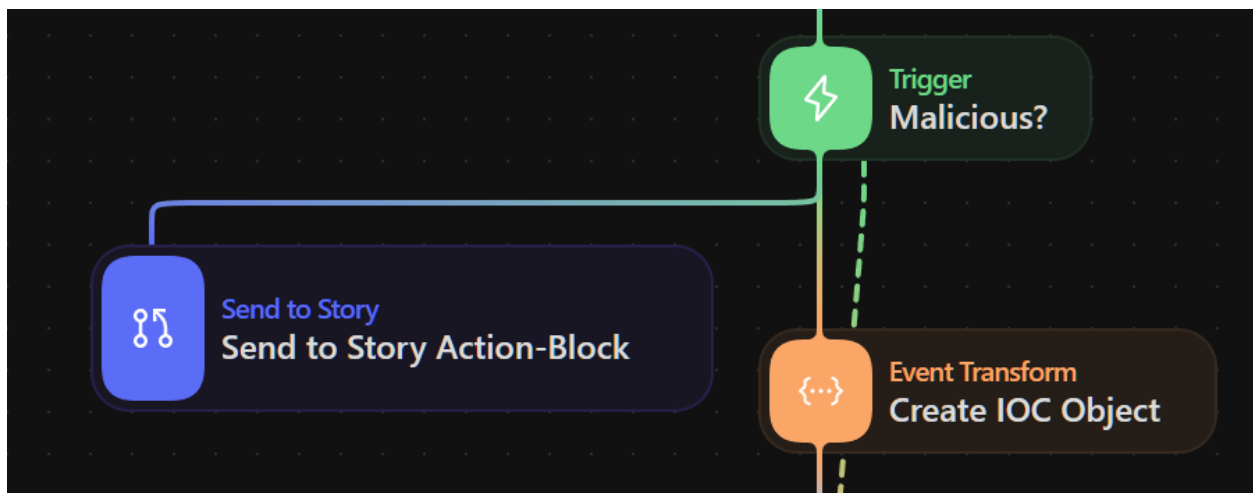
Keep events and logs 1 day

Toggle it and configure the input webhook action and the Exit Event Transformation action as the output:

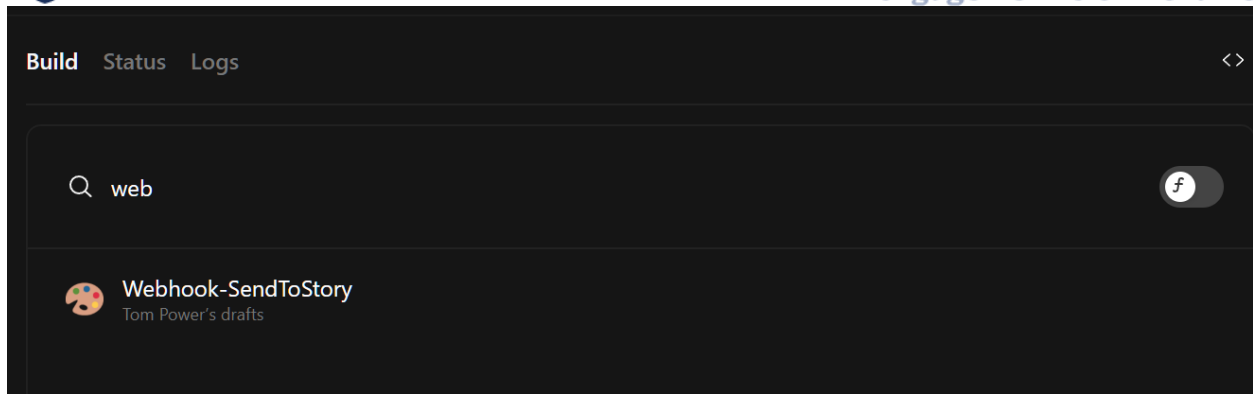


Task 24 – Send To Story Action – Block

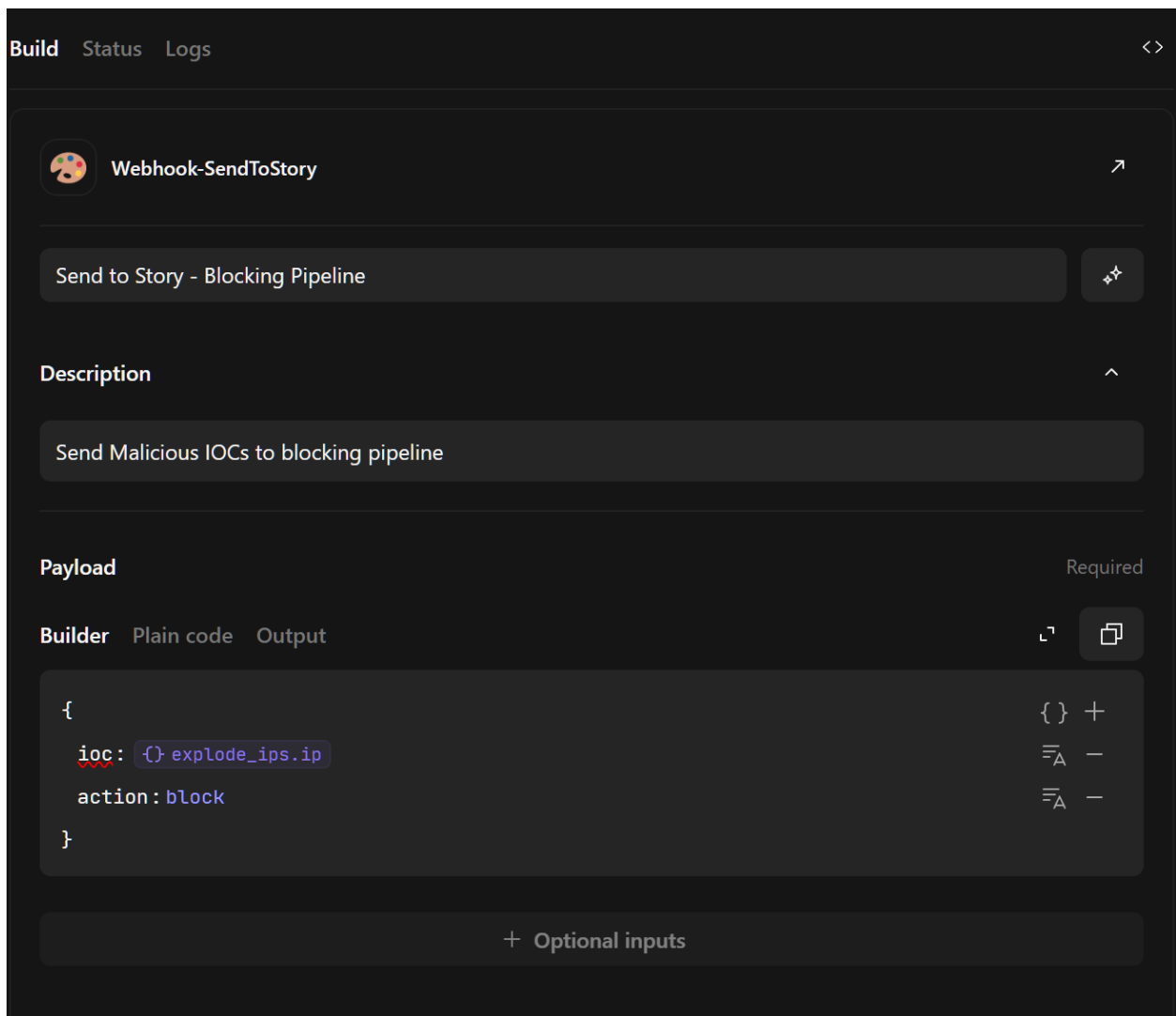
Go back to the IOC Web Submission story and drag a Send to Story action on the UI. Connect it to the Malicious? Trigger action:



In the Build tab, search for the Webhook-SendToStory story where we enabled the Send to Story feature.



Configure the builder payload as below, where there is an object with two keys, ioc and action. The value of the IOC is the exploded_ips.ip and “block” for action.



In the IP web submission form, submit a malicious IP or two.



IPv4 IOCs

218.92.0.228, 218.92.0.231

Submit

Check the events for the webhook action in the Webhook-SendToStory:

Webhook Action

Event ID or substring Search payload: event 4160604046

Re-emit

1 event selected

4160604046
2025-03-26 04:16:03 UTC Just now

```
{
  "webhook_action": {
    "#event_id": 4160604023,
    "#agent_id": 3146967,
    "body": {
      "ioc": "218.92.0.228",
      "action": "block"
    }
  }
}
```

Task 25 – Block or Remove Trigger

In the webhook SendToStory Story, disconnect the webhook from the Exit ET action. Under the webhook action, create a Trigger action configured as below:

Build Status Logs

Name

Block or Remove?

Description

Rules

Formula

`f webhook_action.body.action`

contains

block



Task 26 – Event Transformation – IOC Blocking Pipeline

Drag and drop an Event Transformation configured as below:

Build Status Logs

Name
IOC Blocking Pipeline

Description

Mode
Message only

Loop ☒

Payload ⓘ

Builder Plain code Output

```
{  
  message: This IOC, { webhook_action.body.ioc }, has been blocked.  
}
```

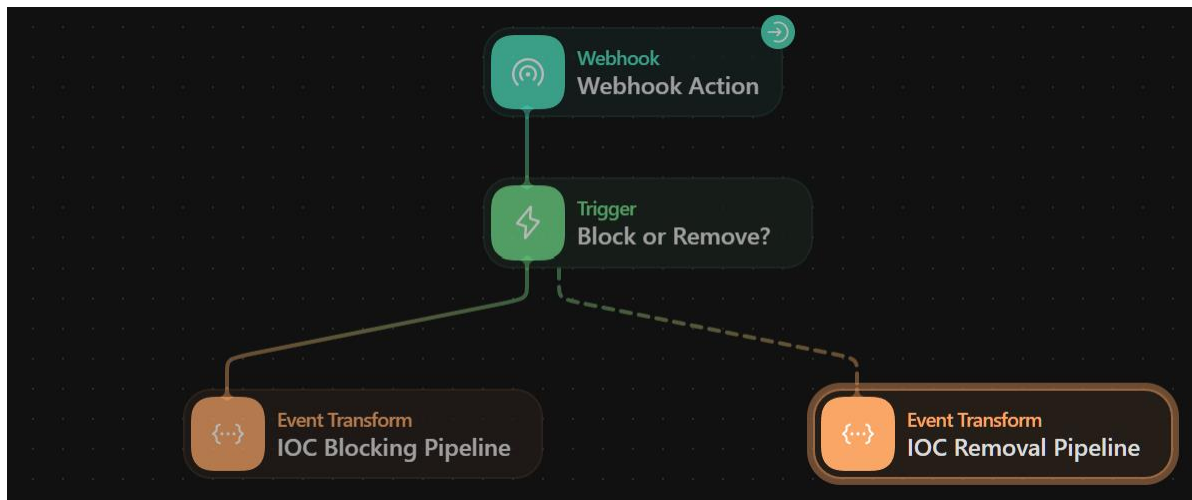


Task 27 – Event Transformation – IOC Removal Pipeline

Drag and drop an ET action and configure it as below:

The screenshot shows the configuration interface for an Event Transformation (ET) action. The top section has tabs for 'Build', 'Status', and 'Logs'. Below this, the 'Name' field is set to 'IOC Removal Pipeline'. The 'Description' field is empty. The 'Mode' is set to 'Message only'. The 'Loop' toggle is turned off. The 'Payload' section shows a JSON message: `{ message: This IOC, {} webhook_action.body.ioc , has been sunsetted! }`. The bottom section has tabs for 'Builder', 'Plain code', and 'Output', with the 'Builder' tab selected.

Connect it to the unmatched trigger pathway.



Task 28 – Event Transformation – Exit

Configure the Exit ET action like below:



Build Status Logs

Name

Exit

Description

Mode

Message only

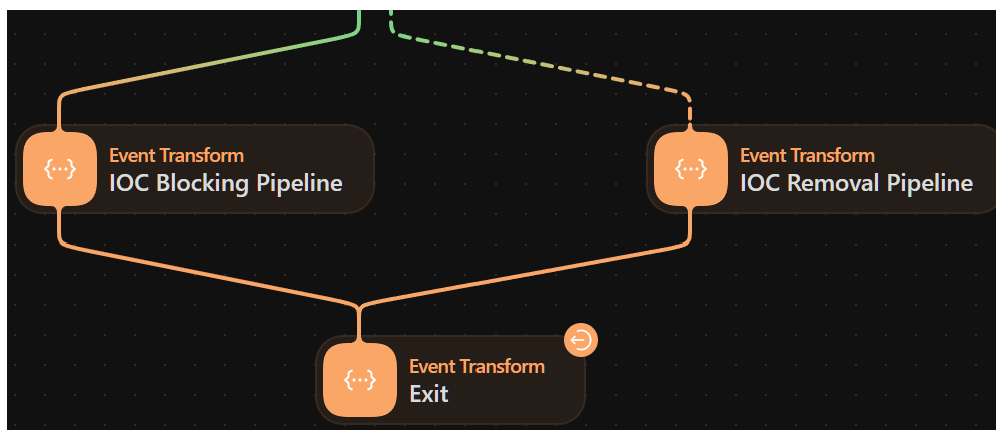
Loop ? ☐

Payload

Builder Plain code

```
{
  message: IOC received!
}
```

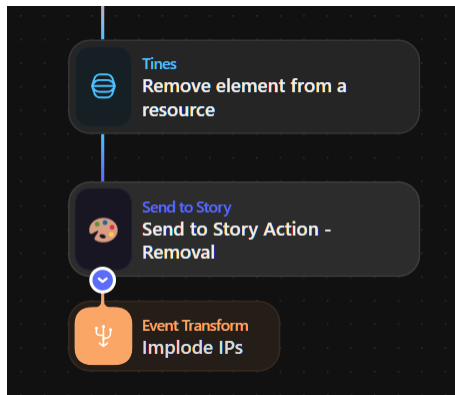
Connect it the other two ET actions to it.





Task 30 – Send To Story Action – Remove

Create a Send to Story action and place it after Remove element action in the IOC sunsetting workflow.



Configure it as below:

Webhook-SendToStory

Send to Story Action - Removal

Description

Send stale IOCs to be removed from IOC Blocking pipeline

Payload Required

Builder Plain code Output

```
{
  ioc: {} explode_ips.ip
  action: Remove
}
```

Change the submission dates for some of the IOCs in ip sunset list, so the date is older than 21 days. Initiate the sunsetting workflow. Check for events in the webhook.



Webhook Action

Event ID or substring

Search payload: event 4168738778

Re-emit

1 event selected

☒ 4168738778
2025-03-27 03:24:48 UTC 16m ago

☐ 4168738714
2025-03-27 03:24:46 UTC 16m ago

```
{
  "webhook_action": {
    "#event_id": 4168738777,
    "#agent_id": 3146972,
    "body": {
      "ioc": "152.53.119.28",
      "action": "Remove"
    }
  }
}
```