RE4DY

MANUFACTURING DATA NETWORKS

RE4DY TOOLKIT

Name of the Tool	Incident Response
Tool Owner	Industry Commons Foundation
Version	1.0
Date	Nov 2025
Version	V1.0

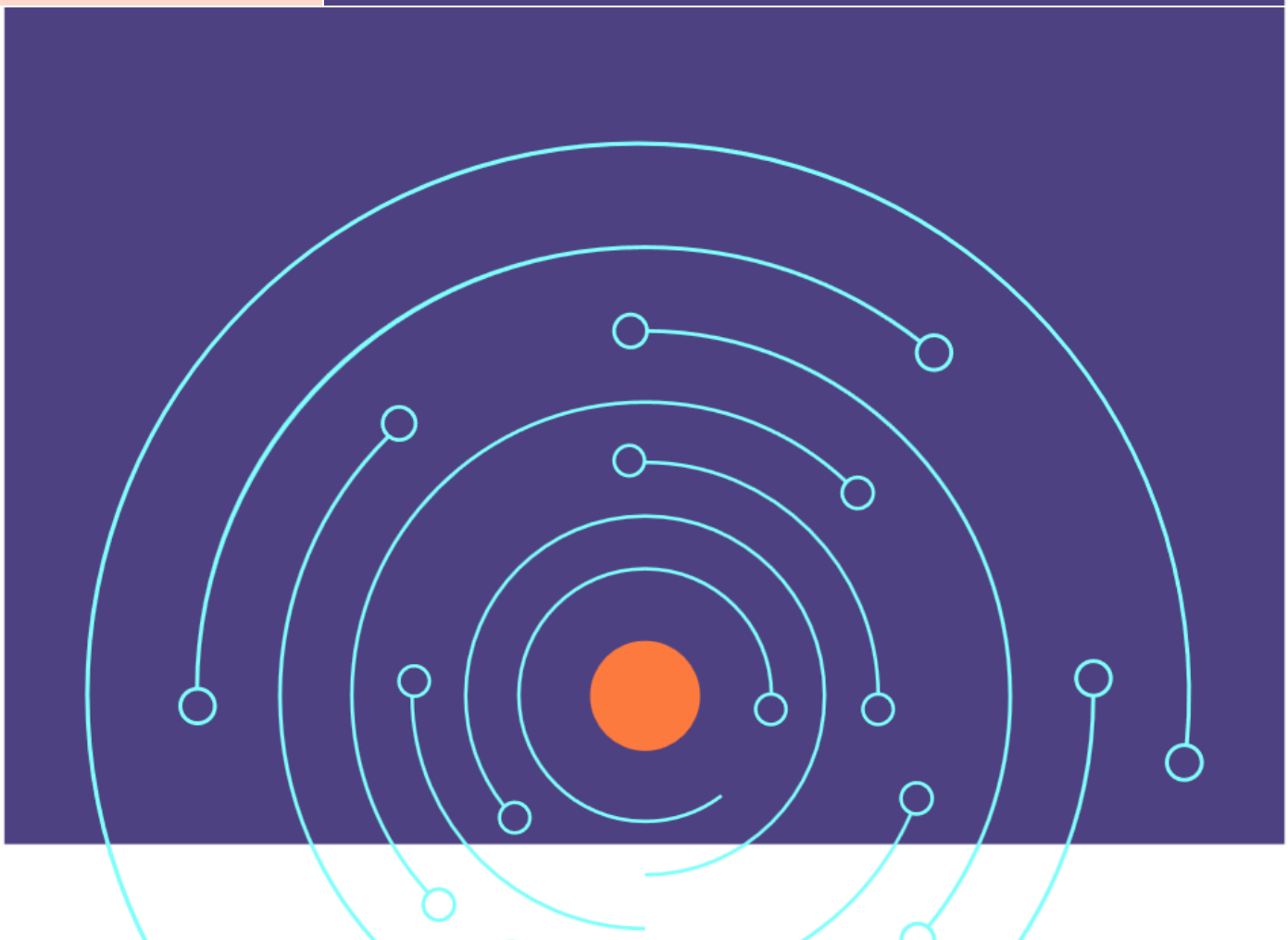

Table of contents

1. Component Description

The Incident response component receives all the alerts from the Incident detection component and creates a security case to perform a treatment and mitigation process. Information received can be enriched from other sources related to the original alert. Once the information is ready, with the incident response tool it is possible to orchestrate different kind of actions, such as automated responses like introducing a condition such as a firewall drop rule in the monitored endpoint. Other orchestrated actions are possible such us notifications to third party applications like ticketing systems, email or instant messaging applications.

2. Input

The incident response communicates with Incident detection and can communicate with other information sources to enrich the cases and offer a better understanding of the Response given to the system. The Alert or incdent received from Incident detection component is in json format:

{

"_index": "wazuh-alerts-4.x-2023.02.27",

"_type": "_doc",

"_id": "9q8DlIYBeKdNQkVPVHDn",

"_version": 1,

"_score": None,

"_source": {

"input": {

"type": "log"

},

"agent": {

"ip": "192.168.15.112",

"name": "RE4DY agent 1",

"id": "005"

},

"manager": {

```
    "name": "wazuh-manager"
  },
  "data": {
    "srcip": "192.168.15.177",
    "in_iface": "enp0s3",
    "src_ip": "192.168.15.186",
    "src_port": "6443",
    "event_type": "alert",
    "alert": {
      "severity": "3",
      "signature_id": "2210046",
      "rev": "2",
      "gid": "1",
      "signature": "SURICATA STREAM SHUTDOWN RST invalid ack",
      "action": "allowed",
      "category": "Generic Protocol Command Decode"
    },
    "tls":{
        "version":"TLS 1.3",
        "ja3":{
            "hash":"89be98bbd4f065fe510fca4893cf8d9b",
            "string":"771,49199-49200-49195-49196-52392-52393-49171-49161-49172-49162-
156-157-47-53-49170-10-4865-4867-4866,5-10-11-13-65281-18-43-51,29-23-24-25,0"
        },
        "ja3s":{
            "hash":"f4febc55ea12b31ae17cfb7e614afda8",
```


"string":"771,4865,43-51"
}
},
"app_proto":"tls",
"flow_id": "1331957630929130.000000",
"dest_ip": "192.168.15.177",
"proto": "TCP",
"dest_port": "48580",
"flow": {
"pkts_toserver": "18",
"start": "2023-02-27T01:54:58.624874+0000",
"bytes_toclient": "5318",
"bytes_toserver": "2212",
"pkts_toclient": "17"
},
"timestamp": "2023-02-27T01:59:53.111013+0000"
},
"rule": {
"firedtimes": 1,
"mail": True,
"level": 12,
"description": "DoS attack has been detected.",
"groups": [
"custom_active_response_rules"
],

"mitre": {
"id": [
"T1498"
],
"tactic": [
"Impact"
],
"technique": [
"Network Denial of Service"
]
},
"id": "100200"
},
"location": "/var/log/suricata/eve.json",
"decoder": {
"name": "json"
},
"id": "1677463193.114738",
"full_log": "{\"timestamp\":\"2023-02-27T01:59:53.111013+0000\",\"flow_id\":1.33195763092913e+15,\"in_iface\":\"enp0s3\",\"event_type\":\"alert\",\"src_ip\":\"192.168.15.186\",\"src_port\":6443,\"dest_ip\":\"192.168.15.177\",\"dest_port\":48580,\"proto\":\"TCP\",\"alert\":{\"action\":\"allowed\",\"gid\":1,\"signature_id\":2210046,\"rev\":2,\"signature\":\"SURICATA STREAM SHUTDOWN RST invalid ack\",\"category\":\"Generic Protocol Command Decode\",\"severity\":3},\"tls\":{\"version\":\"TLS 1.3\",\"ja3\":{\"hash\":\"89be98bbd4f065fe510fca4893cf8d9b\",\"string\":\"771,49199-49200-49195-49196-52392-52393-49171-49161-49172-49162-156-157-47-53-49170-10-4865-4867-4866,5-10-11-13-65281-18-43-51,29-23-24-25,0\"},\"ja3s\":{\"hash\":\"f4febc55ea12b31ae17cfb7e614afda8\",\"string\":\"771,4865,43-51\"}},\"app_proto\":\"tls\",

\"flow\":{\"pkts_toserver\":18,\"pkts_toclient\":17,\"bytes_toserver\":2212,\"bytes_toclient\":5318,\"start\":\"2023-02-27T01:54:58.624874+0000\"}}",

"timestamp": "2023-02-27T01:59:53.809+0000"

},

"fields": {

"data.timestamp": [

"2023-02-27T17:54:49.468Z"

],

"timestamp": [

"2023-02-27T17:54:49.909Z"

]

},

"highlight": {

"rule.id": [

"@kibana-highlighted-field@100201@/kibana-highlighted-field@"

]

},

"sort": [

1677520489908

]

}

3. Output

The Incident Response provides outputs to different system, such as a firewall (firewall drop rule in a monitored endpoint), ticketing systems, email or instant messaging applications. In the following figure it is shown an example of a published response message in MS Teams:

[30/10 11:45] RE4DY Response

New Case generated: ~245772392

RE4DY Nmap scripting engine detected.

- Severity: 3
- Case Id: 1043
- Tags: ['SIEM:Wazuh', 'alert', 'RE4DY']
- mitre-id: T1595
- agent-name: RE4DY agent 1
- rule-description: Nmap scripting engine detected.
- event-type: alert
- automated: False
- mitre-technique:
- mitre-tactic:
- asset_name:
- asset_id:
- agent-ip: 192.168.15.112
- alert-category: Generic Protocol Command Decode
- alert-signature: SURICATA ICMPv4 unknown code
- alert-action: allowed
- source-api: SIEM:Wazuh
- source-rule-level: 12
- source-alert-severity: 3
- source-agent-id: 005
- source-rule-id: 100201
- source-alert-id: 1677520489.9846755

More Info: http://167.71.53.46:9000/index.html#!/case/~245772392/details

Figure 1Incident response: Teams publication response example

4. Information Flow

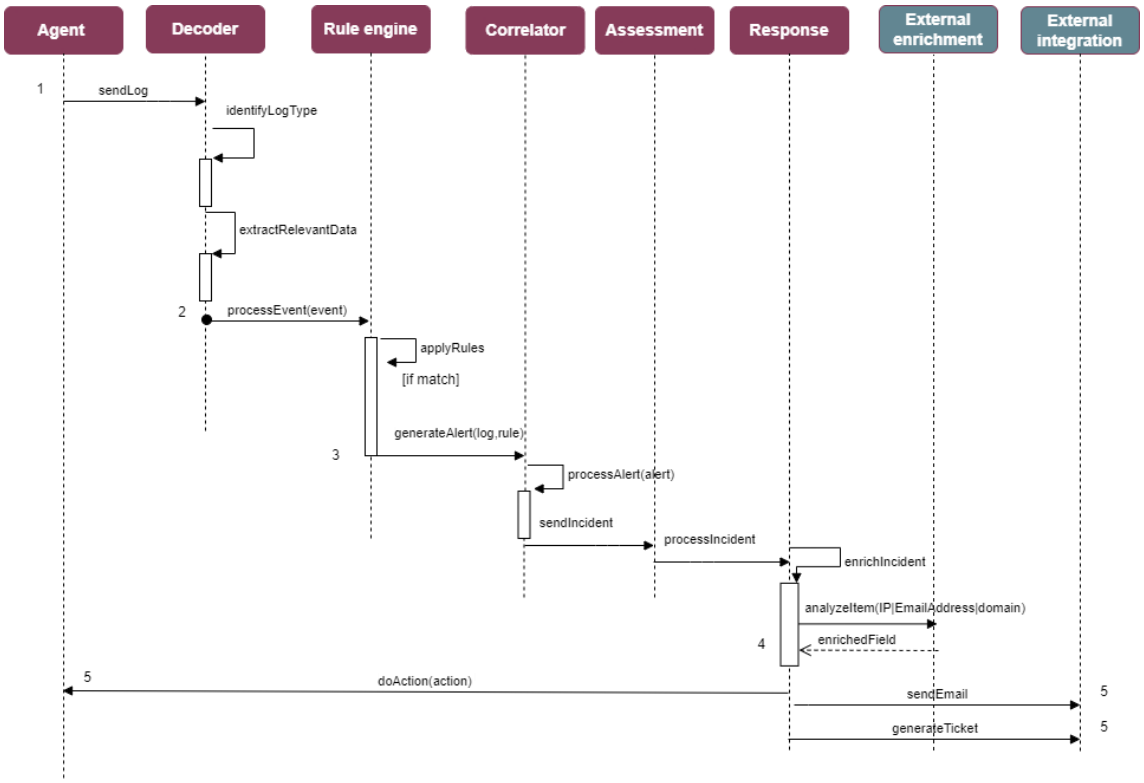

Figure 2 Incident detection and response information flow diagram

- Incident Detection will receive logs and information from the agents deployed.
- Incident Detection will decode the log, identify the type of log and extract some useful fields.
- Incident Detection will have a ruleset to be applied to the received logs.
- Incident Detection will apply the active rules to the received log, and if there is a match, it will generate an alert.
- Incident Response will normalize the alert event and correlate until determine if it is only a simple alert or a real incident.
- Incident Response will enrich the incident with useful information, to facilitate the assignment of the risk level of the incident and the response actions to be done.
- Incident Response can do predefined actions for incident mitigation depending on the incident, such as communicate with the agent so that it performs an action, send an email or send the incident to a ticketing system.

Incident Detection will update information on a GUI so that the admin user can see the status of the agents and the alert/incident information.

(if possible) Provide a UML Sequence diagram of the main information flow(s) of the component

5. Internal Architecture

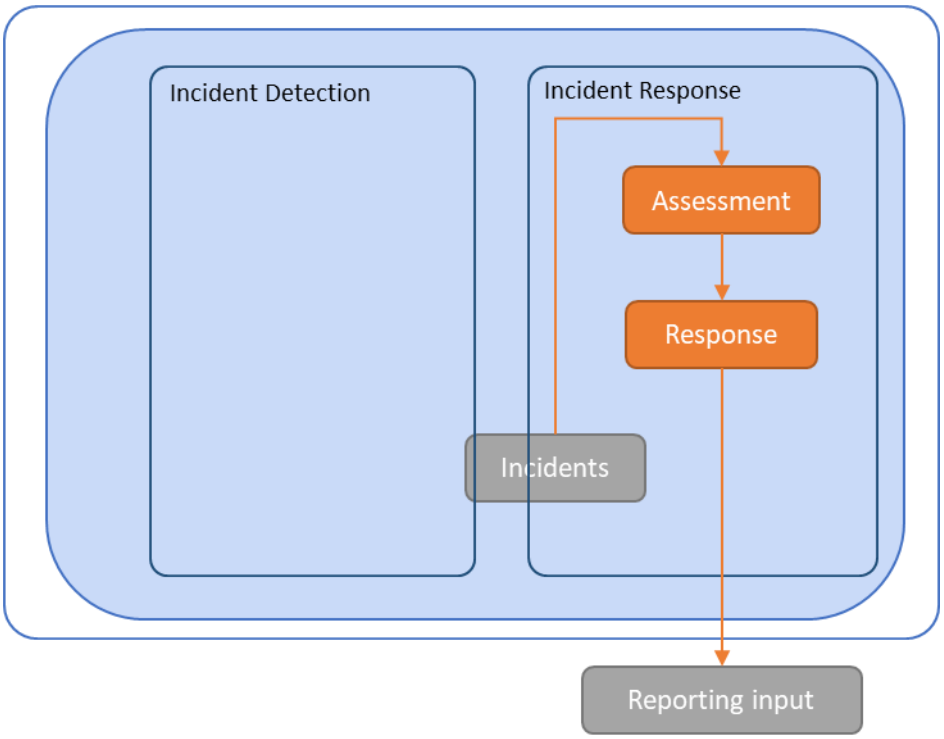

Figure 3 Incident response component architecture

In the Figure 3 it is shown how the different blocks are connected in the component. In the left side it is shown the incident detection component with the incidents generated in the component that are sent to the incident response component.

In the right side, it is shown the incident response component, that collets the incidents generated in the previous step and how these incidents pass through all the steps to finally generate a response to the incident.

6. API

METHOD	DESCRIPTION	ENDPOINT
POST	Authenticate a user and get session cookie	{SOAR}/api/v1/login
POST	Create an organisation	{SOAR}/api/v1/organisation
GET	List all user profiles	{SOAR}/api/v0/profile
POST	Create a new profile	{SOAR}/api/v0/profile
GET	Get information of the given profile	{SOAR}/api/v0/profile/{profile}
PATCH	Update profile	{SOAR}/api/v0/profile/{profile}
DELETE	Remove the profile	{SOAR}/api/v0/profile/{profile}
POST	Create a new user	{SOAR}/api/v1/user
GET	Show information of the current user	{SOAR}/api/v1/user/current
GET	Show information of the given user	{SOAR}/api/v1/user/{user}

PATCH	Update information of the given user	{SOAR}/api/v1/user/{user}
DELETE	Remove an user	{SOAR}/api/v1/user/{user}/force
POST	Set the user password	{SOAR}/api/v1/user/{user}/password/set
POST	Change the user password	{SOAR}/api/v1/user/{user}/password/change
GET	Get the user API key	{SOAR}/api/v1/user/{user}/key
DELETE	Remove the user API key	{SOAR}/api/v1/user/{user}/key
POST	Renew the user API key	{SOAR}/api/v1/user/{user}/key/renew

7. Implementation Technology

TheHive is open-source software which can be installed and executed on bare metal, on a virtual machine, as well as in containerized manner using Docker or Kubernetes. We have chosen the container-based approach for all services related to incident response of RE4DY platform. Therefore, Doker is installed on a virtual machine running Ubuntu 22.04 Linux OS and the compose plugin is used to orchestrate the deployment of the necessary services. We use the official TheHive Docker image as base for our Incident Response instance.

8. Comments

The component will be integrated with incident response detection.