Install Docker and Docker Compose:

If you don't have them, open your Ubuntu terminal and run:

```
# Install Docker
sudo apt-get update
sudo apt-get install docker.io -y
sudo systemctl enable docker --now

# Install Docker Compose
sudo apt-get install docker-compose -y
```

Set Up the ELK Stack: Elastic provides a ready-to-use Docker Compose file.

- Create a directory for your ELK setup and navigate into it:

```
mkdir elk-stack && cd elk-stack
```

- Download the official docker-compose.yml file:

```
Git clone https://github.com/ayounes9/elk-on-docker
```

- Configure yml file:

Put the configuration that below to docker-compose.yml file instead of the current configuration:

```
version: "3.8"
volumes:
  esdata01:
    driver: local
  kibanadata:
    driver: local
networks:
  default:
    name: elastic
    external: false
services:
  es01:
    image: docker.elastic.co/elasticsearch/elasticsearch:8.9.2
    labels:
      co.elastic.logs/module: elasticsearch
    volumes:
      - esdata01:/usr/share/elasticsearch/data
    ports:
      - 9200:9200
    environment:
      - node.name=es01
      - cluster.name=my-elk-cluster
      - discovery.type=single-node
      - ELASTIC_PASSWORD=pass123!
      - bootstrap.memory_lock=true
      - xpack.security.enabled=true
      - xpack.security.http.ssl.enabled=false
      - xpack.security.transport.ssl.enabled=false
      - xpack.license.self_generated.type=basic
      - ES_JAVA_OPTS=-Xms1g -Xmx1g
    mem_limit: 2147483648
    ulimits:
      memlock:
        soft: -1
        hard: -1
    healthcheck:
      test:
        [
          "CMD-SHELL",
          "curl -s http://localhost:9200 | grep -q 'missing authentication credentials'",
        ]
      interval: 10s
      timeout: 10s
      retries: 120
  setup_passwords:
    image: docker.elastic.co/elasticsearch/elasticsearch:8.9.2
    command: >
      bash -c '
        if [ xpass123! == x ]; then
          echo "Set the ELASTIC_PASSWORD environment variable in the .env file";
          exit 1;
        elif [ xpass123! == x ]; then
          echo "Set the KIBANA_PASSWORD environment variable in the .env file";
          exit 1;
        fi;
        echo "Waiting for Elasticsearch availability";
        until curl -s http://es01:9200 | grep -q "missing authentication credentials"; do sleep 10; done;
        echo "Setting kibana_system password";
        until curl -s -X POST -u "elastic:pass123!" -H "Content-Type: application/json" http://es01:9200/_security/user/kibana_system/_password -d "{\"password\":\"pass123!\"}" | grep -q "^{}"; do sleep 10; done;
        echo "All done!";
      '
    depends_on:
      es01:
        condition: service_healthy
    restart: 'no'
  kibana:
    depends_on:
      es01:
        condition: service_healthy
      setup_passwords:
        condition: service_completed_successfully
    image: docker.elastic.co/kibana/kibana:8.9.2
    labels:
      co.elastic.logs/module: kibana
    volumes:
      - kibanadata:/usr/share/kibana/data
    ports:
      - 5601:5601
    environment:
      - SERVERNAME=kibana
      - ELASTICSEARCH_HOSTS=http://es01:9200
      - ELASTICSEARCH_USERNAME=kibana_system
      - ELASTICSEARCH_PASSWORD=pass123!
      -
XPACK_SECURITY_ENCRYPTIONKEY=an_super_secret_32_character_keyr_secret_32_chara
cter_key
      -
XPACK_ENCRYPTEDSAVEDOBJECTS_ENCRYPTIONKEY=an_super_secret_32_character_keyr_se
cret_32_character_key
      -
XPACK_REPORTING_ENCRYPTIONKEY=an_super_secret_32_character_keyr_secret_32_char
acter_key
      - xpack.license.self_generated.type=basic
      - XPACK_FLEET_ENABLED=true
    mem_limit: ${KB_MEM_LIMIT}
    healthcheck:
      test:
        [
          "CMD-SHELL",
          "curl -s -I http://localhost:5601 | grep -q 'HTTP/1.1 302 Found'",
        ]
      interval: 10s
      timeout: 10s
      retries: 120
```

Start the stack. This will pull the container images and start everything in the background (-d).

```
docker-compose up -d
```

Access Kibana (Your SIEM Interface):

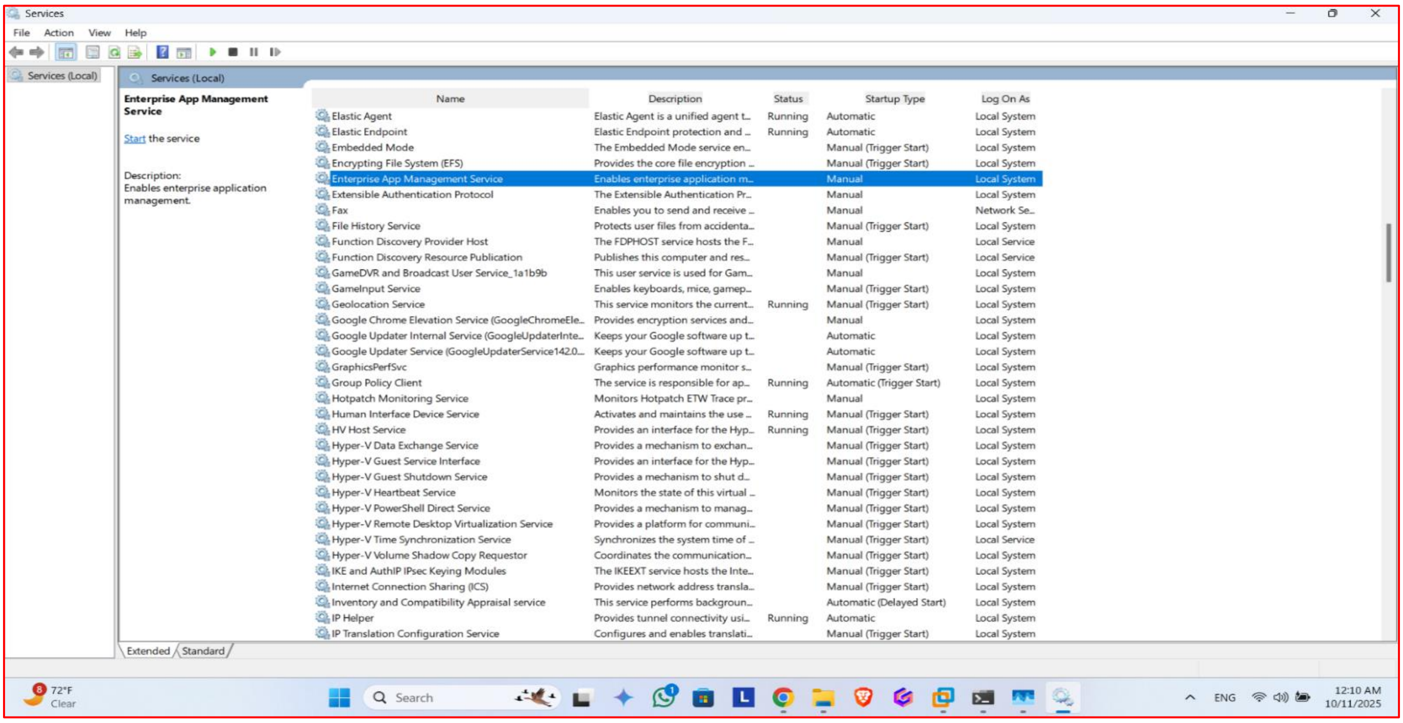
- The services will take a few minutes to start up. You can check the status with `docker-compose ps.`
- Once running, access Kibana in your web browser at: `http://elk-server-ip:5601`
- Log in with the username `elastic` and the password from .env file

You now have a running ELK stack!

Configure Fleet Server & Fleet Agent:

- Run elasticsearch & kibana
- Navigate to kibana UI and then -> Management -> fleet
- Click on add fleet server
- Navigate to the server that hosts ELK
- Follow the commands
- Then add an agent but remember to add –insecure flag after the token in the last command when adding an agent to the server (to ignore certificates)for local env
- Add the elasticsearch host ip from the fleet setting to the elk server ip instead of localhost:9200
- For the policy that is associated to the machines (not the server) add an integration called *system* for this policy and it will collect the needed logs for those machines
- For windows need also a type of integration called windows integration to collect logs
- If needed we can install sysmon for the windows, auditd for linux for more log details

Other Configuration Screens

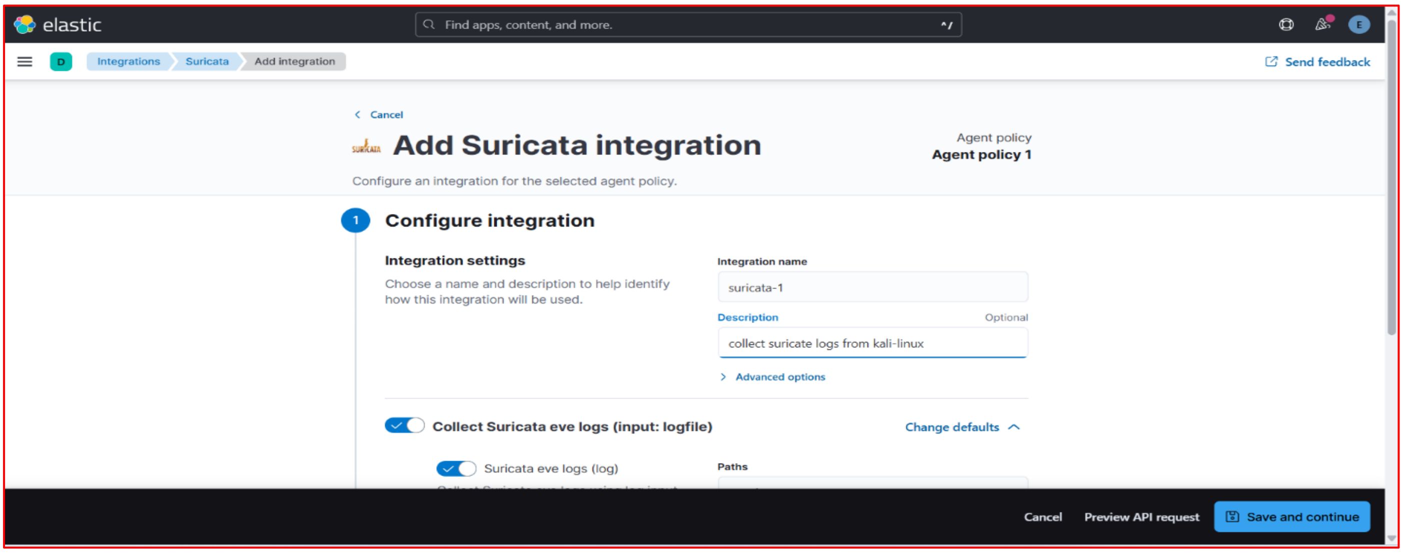


Enterprise App Management Service

Start the service

Description:
Enables enterprise application management.

Name	Description	Status	Startup Type	Log On As
Elastic Agent	Elastic Agent is a unified agent t...	Running	Automatic	Local System
Elastic Endpoint	Elastic Endpoint protection and ...	Running	Automatic	Local System
Embedded Mode	The Embedded Mode service en...		Manual (Trigger Start)	Local System
Encrypting File System (EFS)	Provides the core file encryption ...		Manual (Trigger Start)	Local System
Enterprise App Management Service	Enables enterprise application m...		Manual	Local System
Extensible Authentication Protocol	The Extensible Authentication Pr...		Manual	Local System
Fax	Enables you to send and receive ...		Manual	Network Se...
File History Service	Protects user files from accidenta...		Manual (Trigger Start)	Local System
Function Discovery Provider Host	The FDPHOST service hosts the F...		Manual	Local Service
Function Discovery Resource Publication	Publishes this computer and res...		Manual (Trigger Start)	Local Service
GameDVR and Broadcast User Service_1a1b9b	This user service is used for Gam...		Manual	Local System
GameInput Service	Enables keyboards, mice, gamep...		Manual (Trigger Start)	Local System
Geolocation Service	This service monitors the current...	Running	Manual (Trigger Start)	Local System
Google Chrome Elevation Service (GoogleChromeEle...	Provides encryption services and...		Manual	Local System
Google Updater Internal Service (GoogleUpdaterInte...	Keeps your Google software up t...		Automatic	Local System
Google Updater Service (GoogleUpdaterService142.0...	Keeps your Google software up t...		Automatic	Local System
GraphicsPerfSvc	Graphics performance monitor s...		Manual (Trigger Start)	Local System
Group Policy Client	The service is responsible for ap...	Running	Automatic (Trigger Start)	Local System
Hotpatch Monitoring Service	Monitors Hotpatch ETW Trace pr...		Manual	Local System
Human Interface Device Service	Activates and maintains the use ...	Running	Manual (Trigger Start)	Local System
HV Host Service	Provides an interface for the Hyp...	Running	Manual (Trigger Start)	Local System
Hyper-V Data Exchange Service	Provides a mechanism to exchan...		Manual (Trigger Start)	Local System
Hyper-V Guest Service Interface	Provides an interface for the Hyp...		Manual (Trigger Start)	Local System
Hyper-V Guest Shutdown Service	Provides a mechanism to shut d...		Manual (Trigger Start)	Local System
Hyper-V Heartbeat Service	Monitors the state of this virtual ...		Manual (Trigger Start)	Local System
Hyper-V PowerShell Direct Service	Provides a mechanism to manag...		Manual (Trigger Start)	Local System
Hyper-V Remote Desktop Virtualization Service	Provides a platform for communi...		Manual (Trigger Start)	Local System
Hyper-V Time Synchronization Service	Synchronizes the system time of ...		Manual (Trigger Start)	Local Service
Hyper-V Volume Shadow Copy Requestor	Coordinates the communication...		Manual (Trigger Start)	Local System
IKE and AuthIP IPsec Keying Modules	The IKEEXT service hosts the Inte...		Manual (Trigger Start)	Local System
Internet Connection Sharing (ICS)	Provides network address transla...		Manual (Trigger Start)	Local System
Inventory and Compatibility Appraisal service	This service performs backgroun...		Automatic (Delayed Start)	Local System
IP Helper	Provides tunnel connectivity usi...	Running	Automatic	Local System
IP Translation Configuration Service	Configures and enables translati...		Manual (Trigger Start)	Local System



Add Suricata integration

Agent policy
Agent policy 1

Configure an integration for the selected agent policy.

1 **Configure integration**

Integration settings

Choose a name and description to help identify how this integration will be used.

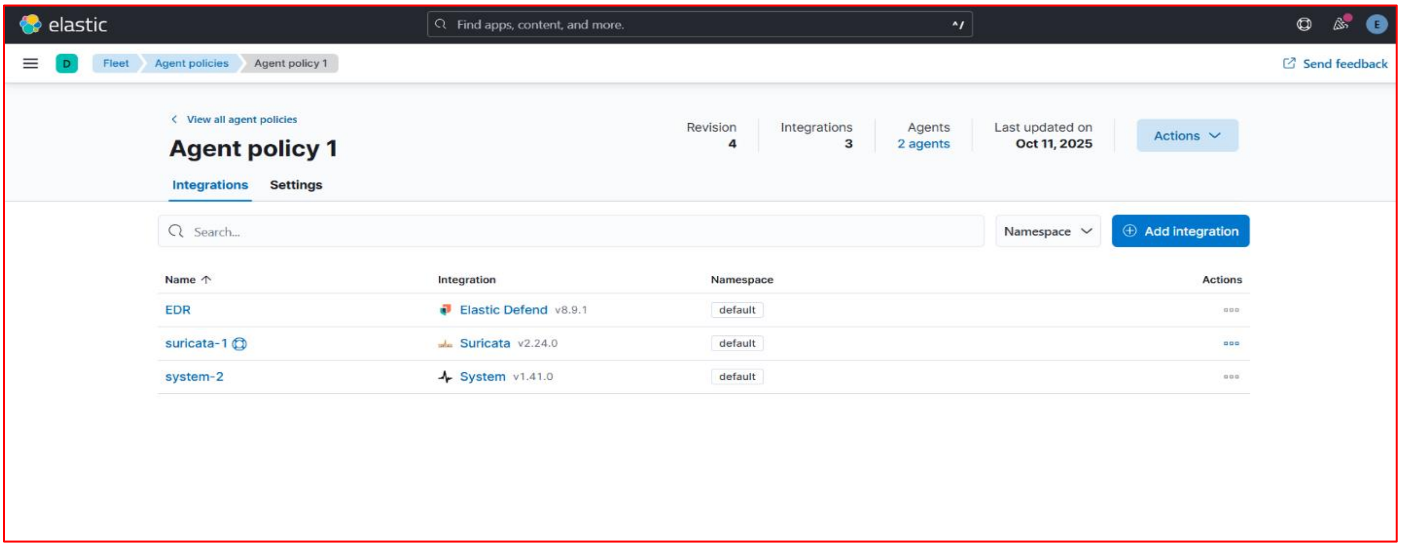
Integration name: suricata-1

Description (Optional): collect suricate logs from kali-linux

Advanced options

☑ **Collect Suricata eve logs (input: logfile)** — Change defaults

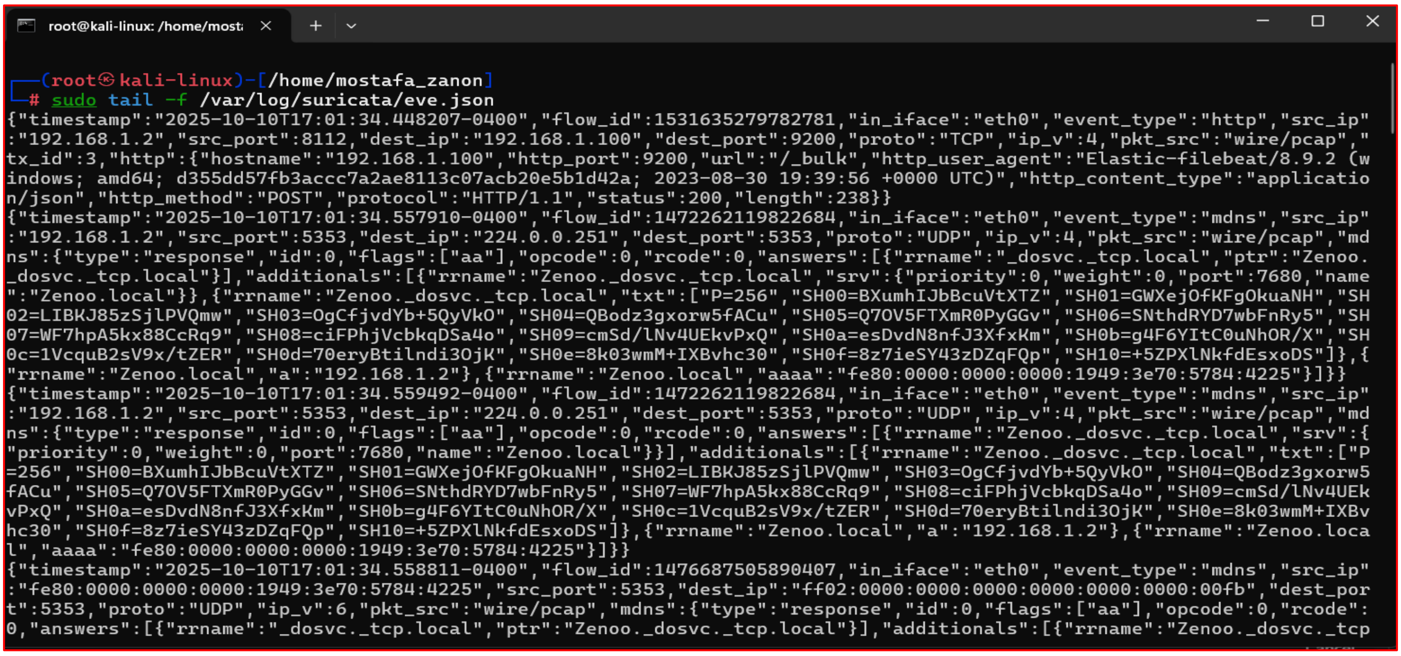
☑ Suricata eve logs (log) — Paths



Agent policy 1

Revision	Integrations	Agents	Last updated on
4	3	2 agents	Oct 11, 2025

Name	Integration	Namespace	Actions
EDR	Elastic Defend v8.9.1	default	•••
suricata-1	Suricata v2.24.0	default	•••
system-2	System v1.41.0	default	•••



```
┌──(root㉿kali-linux)-[/home/mostafa_zanon]
└─# sudo tail -f /var/log/suricata/eve.json
{"timestamp":"2025-10-10T17:01:34.448207-0400","flow_id":1531635279782781,"in_iface":"eth0","event_type":"http","src_ip"
:"192.168.1.2","src_port":8112,"dest_ip":"192.168.1.100","dest_port":9200,"proto":"TCP","ip_v":4,"pkt_src":"wire/pcap","
tx_id":3,"http":{"hostname":"192.168.1.100","http_port":9200,"url":"/_bulk","http_user_agent":"Elastic-filebeat/8.9.2 (w
indows; amd64; d355dd57fb3accc7a2ae8113c07acb20e5b1d42a; 2023-08-30 19:39:56 +0000 UTC)","http_content_type":"applicatio
n/json","http_method":"POST","protocol":"HTTP/1.1","status":200,"length":238}}
{"timestamp":"2025-10-10T17:01:34.557910-0400","flow_id":1472262119822684,"in_iface":"eth0","event_type":"mdns","src_ip"
:"192.168.1.2","src_port":5353,"dest_ip":"224.0.0.251","dest_port":5353,"proto":"UDP","ip_v":4,"pkt_src":"wire/pcap","md
ns":{"type":"response","id":0,"flags":["aa"],"opcode":0,"rcode":0,"answers":[{"rrname":"_dosvc._tcp.local","ptr":"Zenoo.
_dosvc._tcp.local"}],"additionals":[{"rrname":"Zenoo._dosvc._tcp.local","srv":{"priority":0,"weight":0,"port":7680,"name
":"Zenoo.local"}},{"rrname":"Zenoo._dosvc._tcp.local","txt":["P=256","SH00=BXumhIJbBcuVtXTZ","SH01=GWXejOfKFgOkuaNH","SH
02=LIBKJ85zSjlPVQmw","SH03=OgCfjvdYb+5QyVkO","SH04=QBodz3gxorw5fACu","SH05=Q7OV5FTXmR0PyGGv","SH06=SNthdRYD7wbFnRy5","SH
07=WF7hpA5kx88CcRq9","SH08=ciFPhjVcbkqDSa4o","SH09=cmSd/lNv4UEkvPxQ","SH0a=esDvdN8nfJ3XfxKm","SH0b=g4F6YItC0uNhOR/X","SH
0c=1VcquB2sV9x/tZER","SH0d=70eryBtilndi3OjK","SH0e=8k03wmM+IXBvhc30","SH0f=8z7ieSY43zDZqFQp","SH10=+5ZPXlNkfdEsxoDS"]},{
"rrname":"Zenoo.local","a":"192.168.1.2"},{"rrname":"Zenoo.local","aaaa":"fe80:0000:0000:0000:1949:3e70:5784:4225"}]}}
{"timestamp":"2025-10-10T17:01:34.559492-0400","flow_id":1472262119822684,"in_iface":"eth0","event_type":"mdns","src_ip"
:"192.168.1.2","src_port":5353,"dest_ip":"224.0.0.251","dest_port":5353,"proto":"UDP","ip_v":4,"pkt_src":"wire/pcap","md
ns":{"type":"response","id":0,"flags":["aa"],"opcode":0,"rcode":0,"answers":[{"rrname":"Zenoo._dosvc._tcp.local","srv":{
"priority":0,"weight":0,"port":7680,"name":"Zenoo.local"}}],"additionals":[{"rrname":"Zenoo._dosvc._tcp.local","txt":["P
=256","SH00=BXumhIJbBcuVtXTZ","SH01=GWXejOfKFgOkuaNH","SH02=LIBKJ85zSjlPVQmw","SH03=OgCfjvdYb+5QyVkO","SH04=QBodz3gxorw5
fACu","SH05=Q7OV5FTXmR0PyGGv","SH06=SNthdRYD7wbFnRy5","SH07=WF7hpA5kx88CcRq9","SH08=ciFPhjVcbkqDSa4o","SH09=cmSd/lNv4UEk
vPxQ","SH0a=esDvdN8nfJ3XfxKm","SH0b=g4F6YItC0uNhOR/X","SH0c=1VcquB2sV9x/tZER","SH0d=70eryBtilndi3OjK","SH0e=8k03wmM+IXBv
hc30","SH0f=8z7ieSY43zDZqFQp","SH10=+5ZPXlNkfdEsxoDS"]},{"rrname":"Zenoo.local","a":"192.168.1.2"},{"rrname":"Zenoo.loca
l","aaaa":"fe80:0000:0000:0000:1949:3e70:5784:4225"}]}}
{"timestamp":"2025-10-10T17:01:34.558811-0400","flow_id":1476687505890407,"in_iface":"eth0","event_type":"mdns","src_ip"
:"fe80:0000:0000:0000:1949:3e70:5784:4225","src_port":5353,"dest_ip":"ff02:0000:0000:0000:0000:0000:0000:00fb","dest_por
t":5353,"proto":"UDP","ip_v":6,"pkt_src":"wire/pcap","mdns":{"type":"response","id":0,"flags":["aa"],"opcode":0,"rcode":
0,"answers":[{"rrname":"_dosvc._tcp.local","ptr":"Zenoo._dosvc._tcp.local"}],"additionals":[{"rrname":"Zenoo._dosvc._tcp
```