**“확장 가능한 이벤트 형식**(Extensible Event Format, 줄여서 **EVE**) 이벤트 로그는 **JSON** 형식으로 기록됩니다.”

—--------------------------------------------------------------------------------------------------------------------

## **Output types**

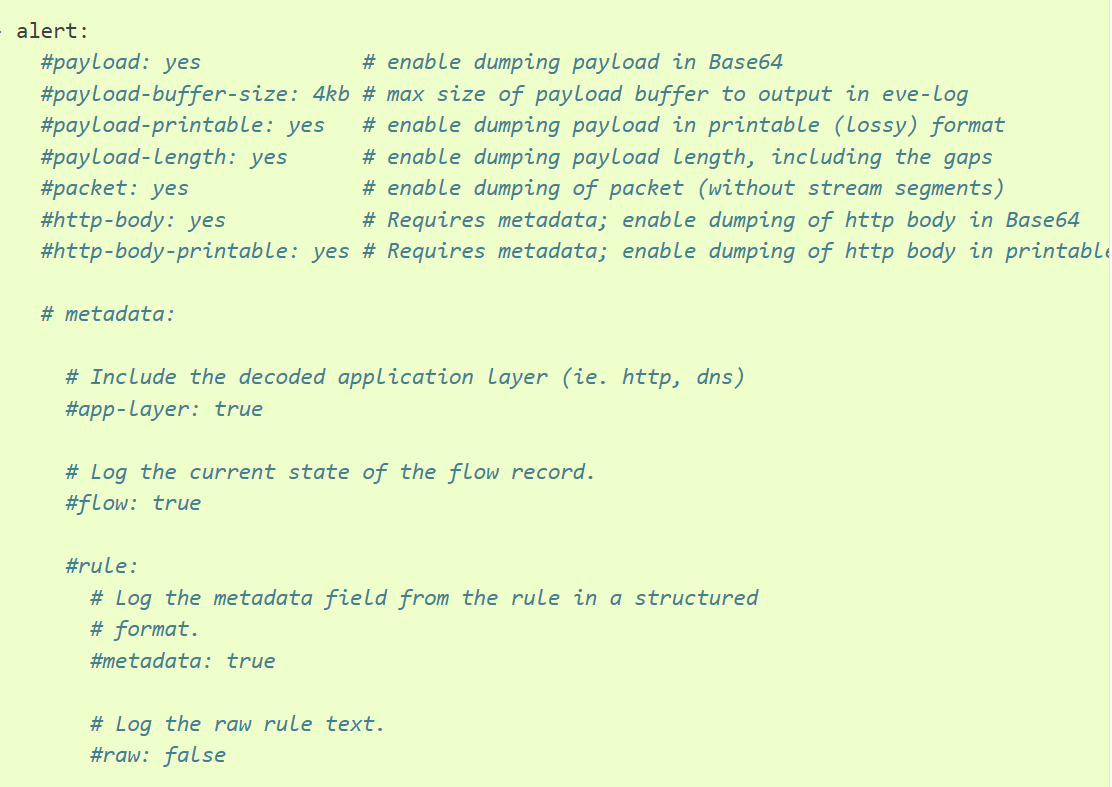
EVE는 여러 가지 출력 방법을 지원할 수 있습니다. regular는 일반 파일로 출력을 의미하며, 이외에도 다양한 출력 옵션이 있습니다. 예를 들어, syslog는 시스템 로그로 출력을 보내는 것이며, unix\_dgram과 unix\_stream은 각각 Unix 도메인 데이터그램 소켓과 스트림 소켓으로 출력을 보내는 방식입니다. redis는 Redis 데이터베이스로 로그를 전송하는 옵션입니다. 이러한 다양한 출력 방법을 사용하면 로그를 원하는 대로 관리하고 저장할 수 있습니다.

—--------------------------------------------------------------------------------------------------------------------

## **Alerts**

경고(Alerts)는 규칙에 일치한 이벤트 기록입니다. 이러한 경고는 메타데이터와 함께 추가될 수 있으며, 예를 들어 경고가 생성된 애플리케이션 계층 기록(HTTP, DNS 등) 및 규칙의 요소들이 포함될 수 있습니다. 이는 경고에 대한 추가적인 맥락 정보를 제공하여, 분석 시 어떤 상황에서 경고가 발생했는지를 더 잘 이해할 수 있게 합니다.

—--------------------------------------------------------------------------------------------------------------------



Suricata의 alert 섹션에서 사용 가능한 다양한 옵션들을 설명하고 있습니다. 각 항목은 주석으로 비활성화되어 있지만, 활성화할 경우 Suricata가 이벤트 로그에 포함할 내용을 제어합니다. 각 옵션의 의미는 다음과 같습니다:

1. **payload**: Base64 형식으로 페이로드(데이터의 본문)를 덤프할지를 결정합니다. 이 옵션을 활성화하면 경고가 발생할 때 패킷의 페이로드가 로그에 기록됩니다.
2. **payload-buffer-size**: 페이로드 버퍼의 최대 크기를 설정합니다. eve-log에 출력될 페이로드 버퍼의 최대 크기를 4KB로 설정합니다.
3. **payload-printable**: 페이로드를 출력 가능한 형식으로 덤프할지를 결정합니다. 이는 손실 가능성이 있는 포맷으로 페이로드를 출력합니다.
4. **payload-length**: 페이로드의 길이를 덤프할지를 결정합니다. 갭을 포함한 페이로드 길이가 로그에 기록됩니다.
5. **packet**: 패킷(스트림 세그먼트를 제외한)을 덤프할지를 결정합니다. 이 옵션을 활성화하면 패킷이 로그에 기록됩니다.
6. **http-body**: HTTP 본문을 Base64 형식으로 덤프할지를 결정합니다. 메타데이터가 필요합니다.
7. **http-body-printable**: HTTP 본문을 출력 가능한 형식으로 덤프할지를 결정합니다. 메타데이터가 필요합니다.
8. **metadata**: 이 섹션은 경고에 포함할 메타데이터 옵션들을 정의합니다.
   * **app-layer**: 디코딩된 애플리케이션 계층 데이터(예: HTTP, DNS)를 로그에 포함할지를 결정합니다.
   * **flow**: 현재 흐름(flow) 기록의 상태를 로그에 포함할지를 결정합니다.
   * **rule**: 규칙의 메타데이터 필드를 구조화된 형식으로 로그에 기록할지를 결정합니다.

이 설정을 통해 Suricata가 생성하는 경고 로그에 얼마나 많은 정보가 포함될지를 세밀하게 조정할 수 있습니다. 이 옵션들은 보안 분석가가 필요로 하는 정보의 양과 깊이를 조절하는 데 사용됩니다.

—--------------------------------------------------------------------------------------------------------------------

## **Anomaly**

Suricata의 비정상적인 이벤트 기록에 대한 설명입니다. 이 이벤트는 예상치 못한 값이 포함된 패킷이 처리될 때 생성됩니다. 이러한 이벤트는 다음과 같은 조건을 포함합니다:

* **잘못된 프로토콜 값**: 패킷의 프로토콜 값이 올바르지 않은 경우.
* **잘못된 프로토콜 길이 값**: 프로토콜 길이 값이 올바르지 않은 경우.
* **기타 의심스러운 조건**: 패킷을 의심스럽게 만드는 기타 조건들.

또한, 스트림의 정상적인 진행 중에도 발생할 수 있는 조건이 있는데, 이는 **스트림 이벤트**라고 하며, 올바르지 않은 값이 포함된 제어 시퀀스나 예상 순서에서 벗어난 제어 시퀀스를 포함합니다.

이러한 비정상적인 이벤트는 유형에 따라 보고되고 구성됩니다:

1. **Decode**: 패킷을 해석(디코드)하는 과정에서 발생하는 비정상적인 이벤트.
2. **Stream**: 스트림 진행 중 발생하는 비정상적인 이벤트.
3. **Application layer**: 애플리케이션 계층에서 발생하는 비정상적인 이벤트.

—--------------------------------------------------------------------------------------------------------------------

"anomaly" 설정은 네트워크에서 발생할 수 있는 예상치 못한 상황이나 비정상적인 동작을 기록하는 데 사용됩니다. 여기서 비정상적인 동작은 패킷이 손상되었거나 예상하지 못한 값이 포함된 경우 등입니다.

### **주요 내용:**

* **비정상 로그**는 손상된 패킷이나 잘못된 IP/UDP/TCP 길이 값을 가진 패킷 등 추가 처리에 적합하지 않은 패킷을 기록하며, 이미 설정된 스트림에서 예상하지 못한 동작을 설명합니다. 네트워크에서 비정상적인 상황이 자주 발생하면 패킷 처리 성능이 저하될 수 있습니다.
* **보고되는 비정상 사항**:
  + **Decode**: 개별 패킷을 디코딩하는 동안 감지된 잘못된 값과 조건. 여기에는 낮은 레벨 프로토콜 길이에 대한 잘못된 값이나 예상하지 못한 값도 포함됩니다.
  + **Stream**: 스트림과 관련된 이벤트(TCP 3-way 핸드셰이크 문제, 예상하지 못한 시퀀스 번호 등)를 포함합니다.
  + **Application layer**: 애플리케이션 계층에서 발생한 예상하지 못한 또는 잘못된 조건을 나타냅니다.
* 기본적으로, **비정상 로그 기록은 비활성화**되어 있으며, 비정상 로그 기록을 활성화하면 애플리케이션 계층의 비정상 보고도 활성화됩니다.
* **로그 기록 선택**:
  + types 설정에서 decode, stream, applayer 중 하나 또는 모두를 선택할 수 있습니다.
  + packethdr 설정을 통해 패킷 비정상에 대한 패킷 헤더의 로그 기록 여부를 결정할 수 있습니다.

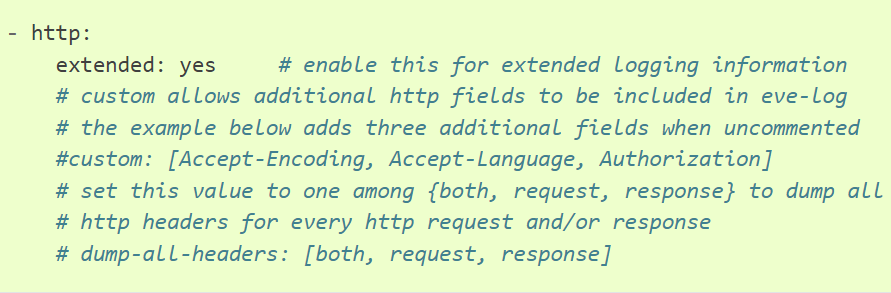
### **설정 예시:**

* types에서 decode, stream, applayer 설정을 사용하여 특정 유형의 비정상 로그를 활성화하거나 비활성화할 수 있습니다.
* packethdr 옵션을 사용하여 패킷 비정상에 대한 패킷 헤더 로그 기록을 활성화할 수 있습니다.

이 설정은 네트워크 보안을 강화하기 위해 비정상적인 네트워크 동작을 모니터링하고 로그로 기록할 수 있도록 도와줍니다.

—--------------------------------------------------------------------------------------------------------------------

## **HTTP**

Suricata에서 HTTP 트랜잭션을 로그로 기록하는 방법을 구성하는 설정입니다. 이를 통해 HTTP 요청 및 응답과 관련된 다양한 정보를 기록할 수 있습니다.

### **주요 설정 항목:**

* **extended: yes**
  + 이 옵션을 yes로 설정하면, HTTP 트랜잭션에 대한 확장된(더 자세한) 로그 정보를 활성화할 수 있습니다. 기본 로그 정보 외에도 더 많은 HTTP 관련 데이터가 기록됩니다.
* **custom**
  + 이 옵션을 사용하면 추가적으로 포함하고 싶은 HTTP 필드를 정의할 수 있습니다. 예를 들어, Accept-Encoding, Accept-Language, Authorization 같은 필드를 추가적으로 로그에 포함시킬 수 있습니다.
  + 주석이 제거되면(앞의 # 기호를 제거하면), 이 필드들이 로그에 기록됩니다.

예시:  
yaml  
코드 복사  
custom: [Accept-Encoding, Accept-Language, Authorization]

* **dump-all-headers**
  + 이 옵션은 HTTP 요청 및/또는 응답에 대한 모든 HTTP 헤더를 로그로 기록할지 여부를 설정합니다.
  + 가능한 값:
    - both: 요청과 응답의 모든 헤더를 로그로 기록합니다.
    - request: 요청의 모든 헤더만 로그로 기록합니다.
    - response: 응답의 모든 헤더만 로그로 기록합니다.

예시:  
yaml  
코드 복사  
dump-all-headers: both

### **설정 예시:**

yaml

코드 복사

http:

extended: yes

custom: [Accept-Encoding, Accept-Language, Authorization]

dump-all-headers: both

이 설정은 HTTP 트랜잭션과 관련된 다양한 정보를 수집하고 분석할 수 있도록 도와줍니다. 특히, 웹 애플리케이션의 동작이나 보안 관련 사항을 모니터링하고자 할 때 유용합니다.

—--------------------------------------------------------------------------------------------------------------------

## **DNS**

—--------------------------------------------------------------------------------------------------------------------

—--------------------------------------------------------------------------------------------------------------------

—--------------------------------------------------------------------------------------------------------------------

—--------------------------------------------------------------------------------------------------------------------

—--------------------------------------------------------------------------------------------------------------------

—--------------------------------------------------------------------------------------------------------------------

—--------------------------------------------------------------------------------------------------------------------

—--------------------------------------------------------------------------------------------------------------------

—--------------------------------------------------------------------------------------------------------------------

—--------------------------------------------------------------------------------------------------------------------

—--------------------------------------------------------------------------------------------------------------------

—--------------------------------------------------------------------------------------------------------------------

—--------------------------------------------------------------------------------------------------------------------

—--------------------------------------------------------------------------------------------------------------------

—--------------------------------------------------------------------------------------------------------------------

—--------------------------------------------------------------------------------------------------------------------

—--------------------------------------------------------------------------------------------------------------------

—--------------------------------------------------------------------------------------------------------------------

—--------------------------------------------------------------------------------------------------------------------

—--------------------------------------------------------------------------------------------------------------------

—--------------------------------------------------------------------------------------------------------------------

—--------------------------------------------------------------------------------------------------------------------

—--------------------------------------------------------------------------------------------------------------------

—--------------------------------------------------------------------------------------------------------------------

—--------------------------------------------------------------------------------------------------------------------