# 了解Snort 3：状态签名评估Byte_Jump

## 目录



## 简介

本文档介绍Snort 3从7.4开始添加的新技术。

## 背景信息

- Snort 3检测模块以块模式工作。虽然这种方法具有性能优势和实现相对简单（相对），但在检测跨多个数据块的签名方面存在一些限制。
- 为了简化用户体验，Snort中已经进行了一些改进，即：
  1. Flowbits允许规则编写器使用用户定义的属性标记网络流；该属性可以在来自流的任何数据包上设置、清除和测试（它提供了一种推断数据包上某些较大签名的方法）。
- 流模块将有线数据包累积为重建数据包，该数据包比原始数据包更大、更有意义；对照重建数据包评估IPS规则可以更有机会看到整体情况并匹配更大的模式（签名）。
- 在某些情况下，重建的数据包不仅提供新数据，而且包括已由检测处理过的先前数据的某些部分；同样，该累积数据块允许检测向后跨越流（在某种程度上）的签名。
- 数据流拆分器将数据流分成多个块，但是剪切点可能是攻击者可用来避免模式检测的薄弱点；因此Snort实施了抖动机制，使拆分更加不可预测。这使得攻击者的分析更加复杂。

## 新特性

状态签名评估是一种可以添加到列表中的新技术。它通过在多个块上启用IPS规则评估来扩展检测功能。因此，如果当前块缺少数据，规则不会立即不匹配，而是等待更多数据到达。

# 支持的平台

## 最低软件和硬件平台

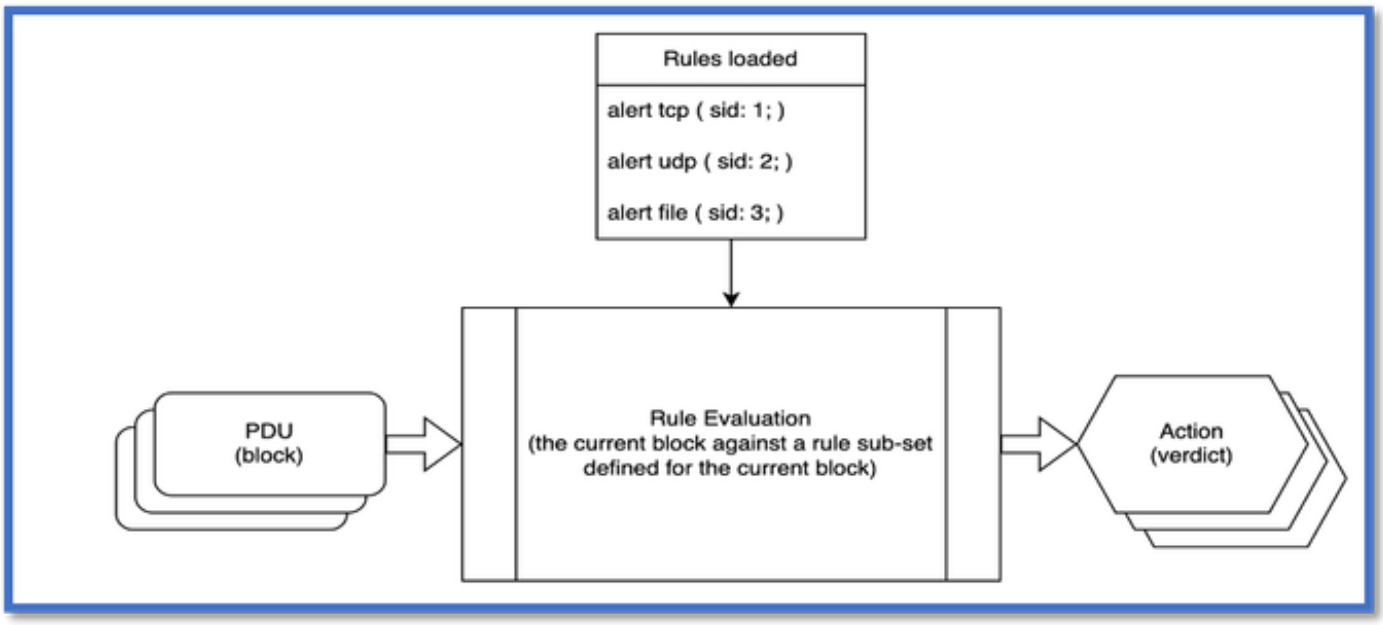
| 支持的最低管理器版本 | 受管设备 | 需要支持的最低受管设备版本 | 备注 |
|---|---|---|---|
| 管理中心7.4.0 | FTD | 7.4.0 | 仅Snort 3 |
| 设备管理器7.4.0 | 支持FDM管理的任何FTD | 7.4.0 | 仅Snort 3 |

# 功能详细信息

## 功能功能说明

它如何运转?

检测模块工作流程如图所示。在流量处理阶段，模块已加载所有规则，并且它以逐个接受数据块、评估规则以及定义要对处理状态签名评估块执行的操作。




有关计划的附注：

1. 一旦为当前数据块定义了规则子集，则系统会独立于其他规则评估其中的每个规则。

2. 每个数据块都独立于其他块进行评估。
3. 数据块是为当前数据包评估的一组IPS缓冲区的抽象。
4. Action是为当前数据包评估的操作的列表；最终判定将在稍后确定。

要了解状态签名评估的工作原理，请查看如何评估通用IPS规则，以及数据块如何形成数据流。

通用规则评估

IPS规则可以以下列形式显示：

```
action protocol source → destination ( option_1: parameters; option_2: parameters;
option_3: parameters; gid: 1; sid: 1; meta_option_1; meta_option_2; meta_option_3; )
```

其中：

action —如果规则触发，则对数据包执行IPS操作

protocol -要匹配的协议

源、目标- IP地址和端口

option_1、option_2、option_3 -作为规则评估一部分的IPS选项

gid、sid -标识规则的唯一对（类似于元数据选项）

meta_option_1、meta_option_2、meta_option3 -规则元数据（如消息、类类型或引用），这些选项不参与规则评估。

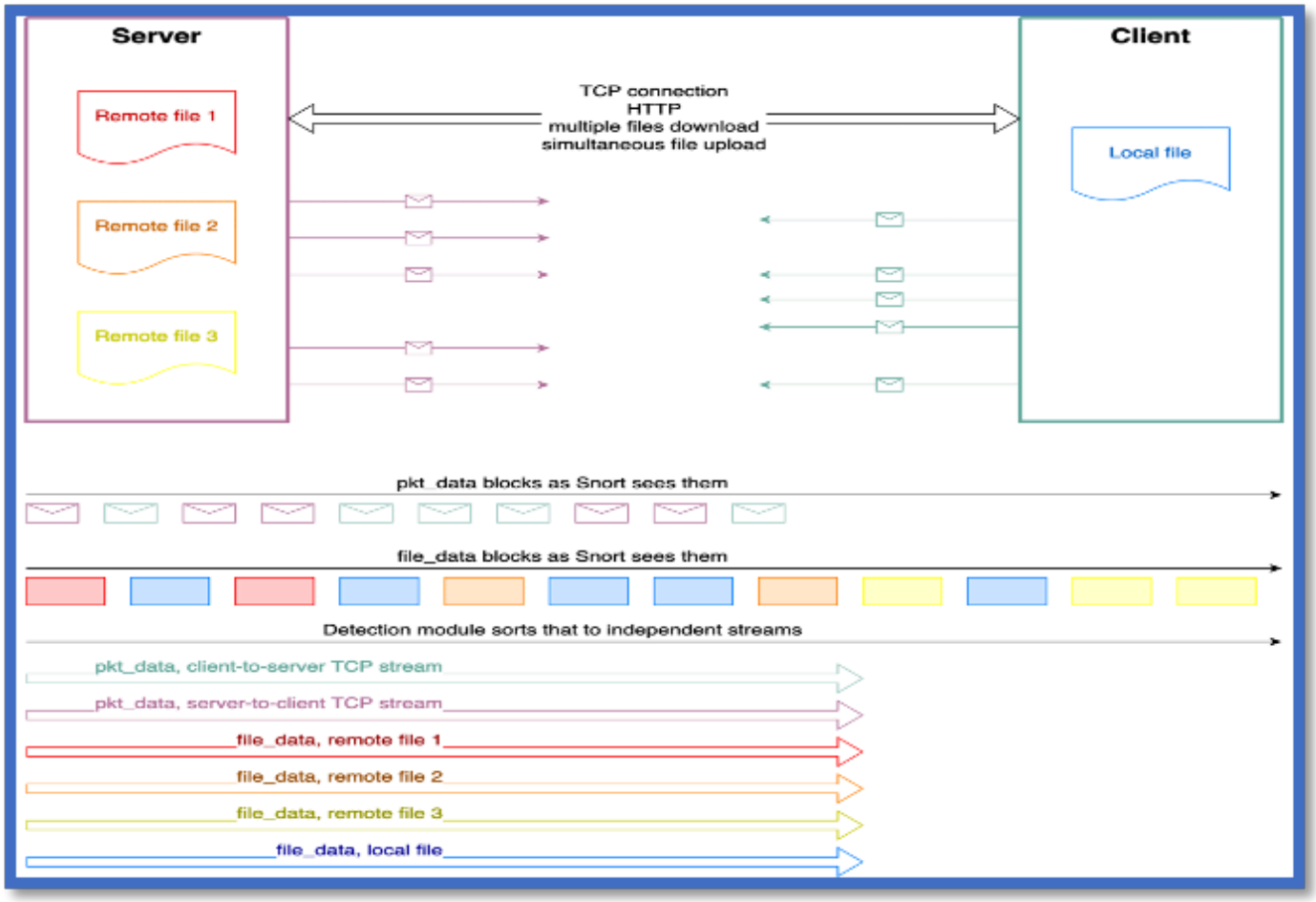
- 协议、源和目标构成规则报头。对于网络流（接受哪个流进行评估），它就像一个过滤器。括号中的所有内容都是规则体。来自规则正文的IPS选项（规则元数据除外）是为数据块评估的选项。它们符合以下声明：
- 选项按从左到右的顺序严格评估。
    1. 可以是两种主要类型之一。
    2. buffer setter时，该选项会为当前数据包选择IPS缓冲区。
- 其他（模式搜索、数学操作、光标操作、flowbit操作）
- 光标用于跟踪所选IPS缓冲区中的位置。

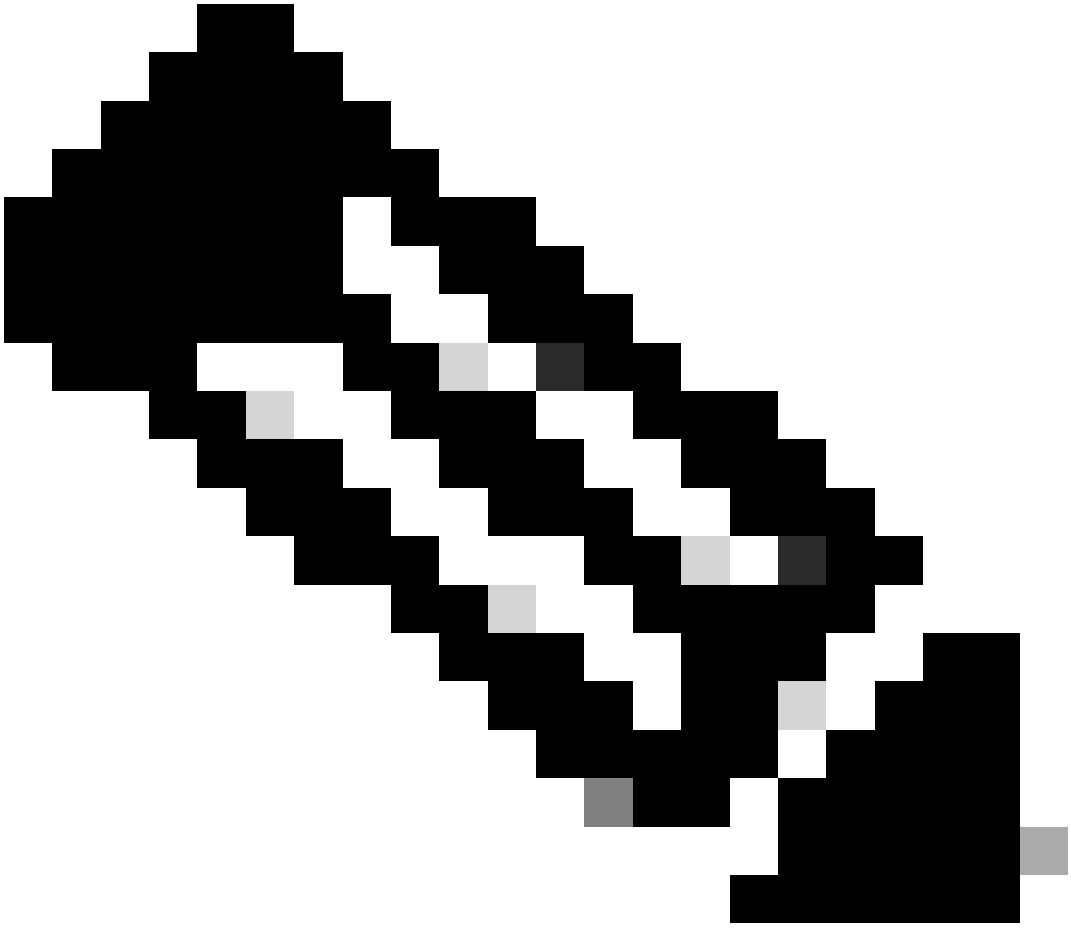
- 选项可以是：
    1. “absolute”，表示它不依赖于光标位置
    2. “relative”，表示从光标位置开始计算
- 如果某个选项尝试将光标设置为选定IPS缓冲区之外，则该选项会失败，并且整个规则不匹配（由于缺少数据）
- 最后一点是检测模块的局限性。如果Snort可以拥有无限的资源，当数据可用时（更多有线数据包到达），它会缓存所有查看到的数据，以反复评估规则。

# 数据流和IPS缓冲区

- 数据流是来自同一源的连续格式字节流。它是一种支持状态评估的新概念。块之间的规则评估必须在同一逻辑数据（无论是文件、纯TCP数据流还是JavaScript文本）内完成。

- 一般来说，检测模块接收的数据块可以：
  - 来自不同的IPS缓冲区（例如，pkt_data和file_data不同）
  - 属于另一个流
  - 不形成数据流（从原始数据包生成的缓冲区）
  - 不形成连续流(ICMP、UDP)
  - 不按顺序排列（HTTP部分响应）
  - 包含重复数据（累积块，如http_inspect.script_detection或HTTP分块响应）
- 检测模块可以只对同一数据流中的块进行排序并串联起来，否则，评估过程会发现交织块产生的不必要干扰。





注意：此处的示例显示HTTP客户端同时上传和下载多个文件的情况。

- 目前，只有两个IPS缓冲区可以表示数据流：pkt_data和file_data，其中：
  1. pkt_data形成TCP协议的两个数据流（客户端到服务器和服务器到客户端方向）
  2. file_data必须为文件、MIME附件和其他协议数据（如HTTP HTML页面和/或其他内容类型）形成流
- 状态评估仅在数据流内完成。

## 规则延续

- 前面的部分以一条语句结尾，该语句指出，如果IPS选项将光标置于当前IPS缓冲区之外，则会出现不匹配。但是，当IPS缓冲区形成数据流时，状态签名评估功能会逐步进入，并将规则评估上下文保存在Snort流对象中。保存的评估上下文（状态）称为规则延续。状态签名评估会延迟规则的最终裁决，直到有更多的数据可用。
- 规则延续包括三个主要部分：IPS缓冲区名称、缓冲区源和目标游标位置（缓冲区源是数据流的唯一标识符）。
- 当数据块由检测模块处理时，后续操作会发生：-

    - 状态签名评估会创建规则延续，并将其附加到以下情况：
        - IPS选项（byte_jump、content、pcre或更新游标位置的任何其他内容）将游标设置在当前IPS缓冲区之后
        - 当前IPS缓冲区支持数据流。
        - 当前IPS缓冲区会形成数据流。
- 状态签名评估会撤销刚刚创建的规则连续性，并在以下情况下将其从流中删除：
    - IPS规则已在当前数据块上触发（该规则与块的其他位置匹配）
- 状态签名评估拒绝待处理的规则继续，并在以下情况下将其从流中删除：
    - IPS缓冲区不形成连续数据流(例如，数据块中有重复数据，或者存在间隙（部分数据丢失或块顺序混乱）。
- 状态签名评估在以下情况下使用新数据更新目标光标位置：
    - 来自规则连续性的缓冲区源与选定的缓冲区源相同
    - IPS缓冲区形成连续数据流
- 状态签名评估在以下情况下将规则继续发送回IPS规则引擎：
    1. 目标光标定位点在选定IPS缓冲区内（也就是说，它最终收到了完成规则评估所需的所有数据）。

### 用户配置

- 由于规则持续占用内存，因此Snort无法存储无限数量的规则。有一个配置选项用于控制限制：
    1. Detection.max_continuations_per_flow = 1024：流中同时存储的最大连续数{ 0：65535 }
- 当有状态签名评估达到限制时，它会使用新规则替换最旧的规则延续。
- 驻留在流中的最旧规则继续时间太长，这意味着它仍然不满足恢复规则评估的条件。
- 此外，还有大量挂接计数可用于微调IPS规则（必须成为主要焦点）和限制（如果需要）：
    1. detection.cont_creations：创建的连续总数（总和）
    2. detection.cont_recalls：已召回的连续总数（总和）
    3. detection.cont_flows：使用连续值的流总数(sum)
    4. detection.cont_evals：符合条件的连续项总数（总和）
    5. detection.cont_matches：匹配的总数(sum)
    6. detection.cont_mismatches：不匹配的连续项总数(sum)
    7. detection.cont_max_num：每个流同时持续数的峰值（最大）
    8. detection.cont_match_distance：匹配连续跳转的总字节数(sum)
    9. detection.cont_mismatch_distance：不匹配连续跳过的字节总数(sum)

## 故障排除

该功能是对现有检测过程的增强，因此无法显式进行故障排除。如果检测失败，则必须检查规则、配置或流量。

## 问题示例

### 问题：说明

- 假设签名必须同时检查文件的开始及其尾部。
- 例如，在此结构的目标文件（标头、正文、元数据）中，我们需要查看其元数据中是否有值为0的值。

- 文件字节：e1 f3 22 03 7f ff xx xx ... xx 01 00 02 00 ，其中
    - e1 f3 22 03 – 4个字节（幻数），用于标识文件类型
    - 7f ff - 2个字节（表示正文大小）
    - xx ... xx - 32kb的部分数据
    - 01 00 02 00 – 4字节的元数据，采用标记值格式（每个1字节）
- IPS规则如下所示：alert file ( file_data； content："|e1f32203|"，fast_pattern；byte_jump：2,0，relative； content："00"，within：4， relative； sid： 1； )
    - 其中
        - 文件协议确保规则仅接受重建数据包（原始数据包不参与状态签名评估）
        - 'file_data选项选择可形成流的文件数据缓冲区
        - 第1个内容选项是快速模式，它会检查幻数（如果这是预期文件类型）
        - byte_jump选项读取文件正文大小并在文件正文上跳转
        - 第2个内容选项在参数限制搜索深度内对元数据值执行最终检查，并使该选项成为相对选项。

## 问题：解决方案

规则将按以下方式评估：

第1个数据包（大小为8kB）携带文件报头和正文的一部分：

1. 已选择IPS buffer file_data。光标指向第0字节e1。
2. fast pattern选项匹配并设置光标位置，紧跟在幻数后面，指向字节7f。
3. byte_jump选项读取两个字节的文件体大小。这两个字节更新游标。然后，byte_jump计算超过32768个字节的跳跃。
4. 状态签名评估可创建规则延续，其中需要24578字节(32768 -（8kB - 4字节报头- 2字节正文大小））。
5. 整个规则不匹配，因为byte_jump选项不能将游标位置设置得那么远。

在第二个数据包（大小为16kB）上，携带文件正文部分：

1. 状态签名评估将看到挂起的规则延续。
2. 它按名称选择缓冲区，并看到file_data可用且新数据大小为16384。
3. 更新后的游标显示仍需要8194字节(24578 - 16384)
4. 规则未恢复。

第3个数据包（大小为8198个）上携带文件正文部分和元数据：

1. 状态签名评估将看到挂起的规则延续。
2. 它按名称选择缓冲区，并看到file_data可用且新数据大小为8198。
3. 更新后的游标显示缓冲区有足够的数据，游标位置是8194。
4. 状态签名评估将删除规则的连续性。

5. 状态签名评估从第2个内容选项恢复规则评估，光标指向字节01。
6. content选项在搜索到的第2个字节上查找匹配项。
7. 整个规则最终触发。

# 限制详细信息和常见问题

## 限制和其他注意事项

- 由于状态签名评估实施，Snort会在重新加载其配置时丢弃所有待处理的规则继续。请注意，尽管丢弃了规则，但规则继续操作仍会占用Snort内存，直到下一个数据块发送到检测模块。
- 状态评估中IPS规则的规则延迟功能的作用与常规规则评估相同。总结了不同数据块上规则部分的评估时间。如果时间超过限制，则规则评估会执行短路，提前退出。
- Flowbits操作保留其含义，但它们仍然执行类似“静态”的选项。
  在当前已知环境中执行flowbit设置/清除/测试操作。因此，如果在规则继续中评估flowbit选项，该选项会考虑当前环境（flowbits集），而不是规则开始评估时的环境。

此外，规则编写器必须注意快速模式位置。

即使它可以在规则的任何部分，也会在整个规则之前评估fast-pattern选项。它触发规则评估。对于基于状态签名评估的规则，这意味着规则持续点必须位于fast-pattern选项之后。
此外，IPS规则的评估中可有多个规则连续性（一个接一个，而不是同时）。由于规则正文中的任何选项都可以继续，因此规则编写器可以使用同一IPS规则在数据流中的不同位置执行额外检查。

关于此翻译
思科采用人工翻译与机器翻译相结合的方式将此文档翻译成不同语言，希望全球的用户都能通过各自的语言得到支持性的内容。

请注意：即使是最好的机器翻译，其准确度也不及专业翻译人员的水平。

Cisco Systems, Inc. 对于翻译的准确性不承担任何责任，并建议您总是参考英文原始文档（已提供链接）。