在安全防火墙威胁防御7.4中配置AppID早期数据包检测

目录

简介

本文档介绍如何在Cisco安全防火墙7.4中配置AppID早期数据包检测。

背景-问题（客户要求）

- 通过深度数据包检测进行应用检测可能需要多个数据包来识别流量。
- 有时，在已知应用服务器的IP和/或端口的情况下，可以避免检查其他数据包。

新特性

- 我们创建了新的基于Snort的Lua AppID API，它允许将IP地址、端口和协议映射至各项：
  - 应用协议(service appid)，
  - 客户端应用（客户端appid）和
  - Web应用（负载应用）。
- 可以使用此API在FMC上创建自定义应用检测器，以进行应用检测。
- 激活此检测器后，此新API将允许我们识别会话中第一个数据包上的应用。

功能概述

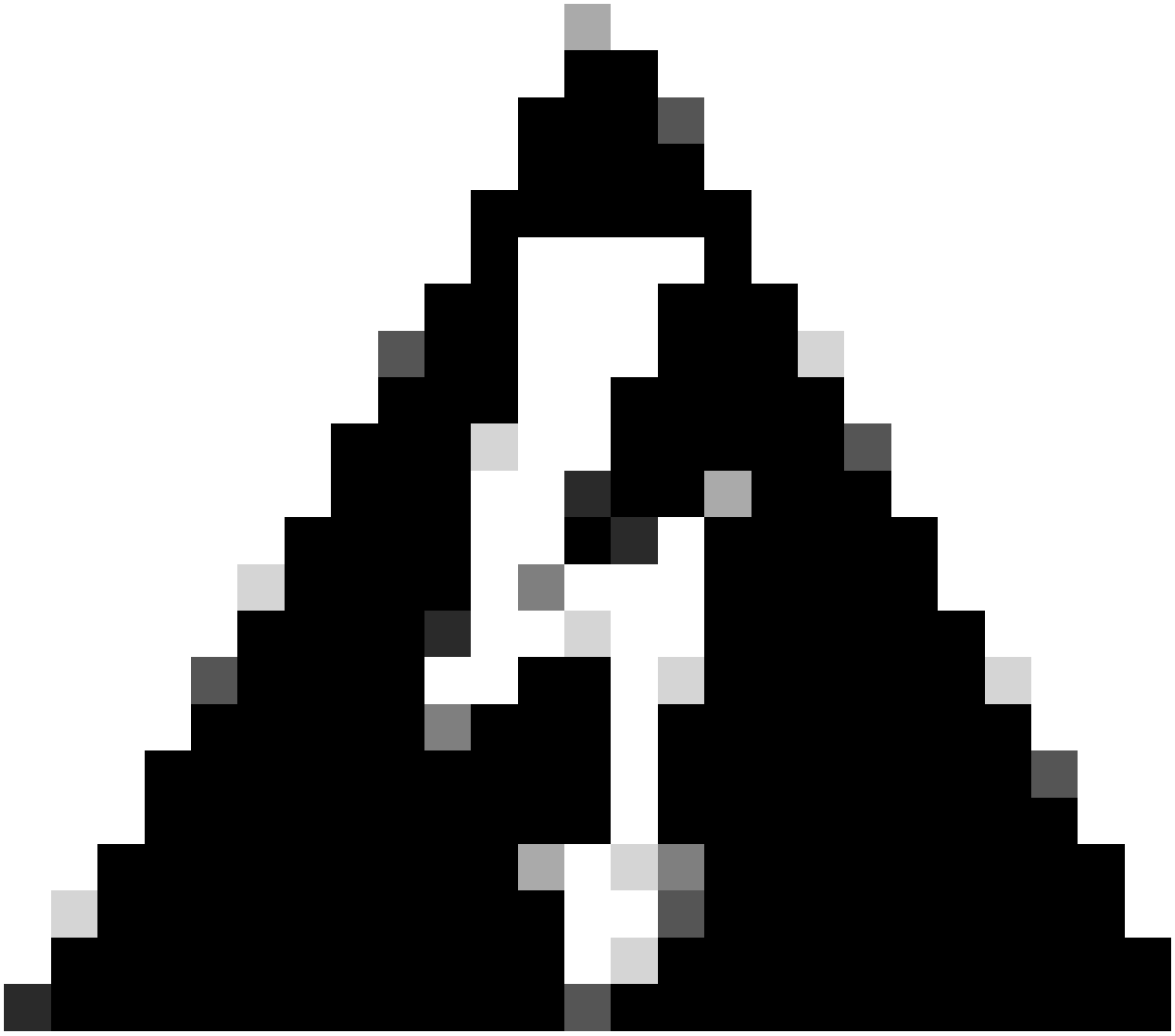
- API标识为：
  - **addHostFirstPktApp** （protocol_appId、client_appId、payload_appId、IP地址、端口、协议、恢复）

- 为自定义应用检测器中创建的每个映射创建缓存条目。
- 检查所有传入会话的第一个数据包，查看缓存中是否存在匹配。
- 找到匹配项后，我们会为会话分配相应的api，应用发现过程将停止。
- 即使在API找到匹配项后，用户仍可以选择重新扫描流量。
- respect参数是一个布尔值，指示是否需要重新扫描第一个数据包上找到的应用程序。
- 当重新检测为true时，即使API找到匹配项，应用发现也会继续。
- 在这种情况下，第一个数据包上分配的appid可能会发生变化。

必备条件、支持的平台、许可

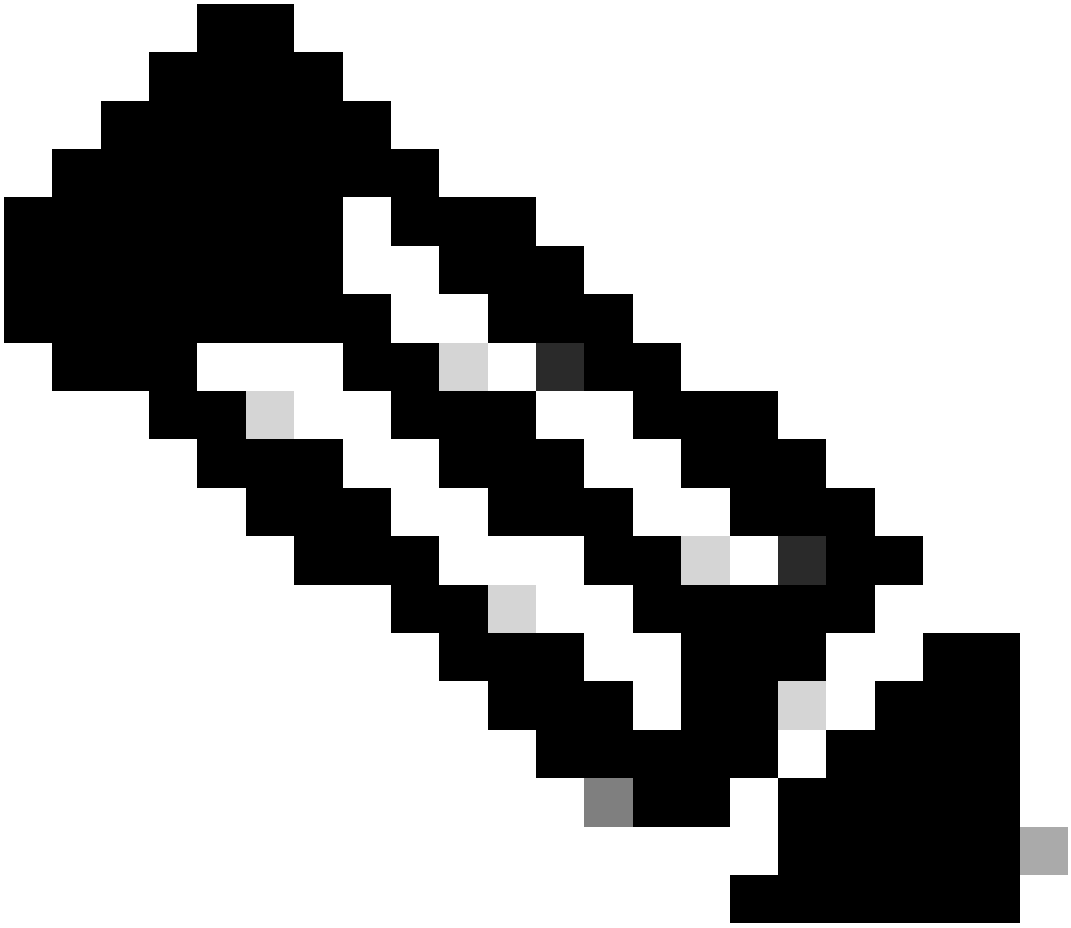
最低软件和硬件平台

应用和最低版本	支持的托管平台和版本	管理员	备注
安全防火墙7.4 使用Snort3	支持FTD 7.4的所有平台	FMC内部部署+ FTD	这是设备端功能；FTD必须在7.4版本上



警告： Snort 2不支持此API。

Snort 3、多实例和HA/群集支持



注意：要求Snort 3作为检测引擎。

FTD	
是否支持多实例？	Yes
支持HA'd设备	Yes
是否支持集群设备？	Yes

使用的组件

本文档中的信息基于以下软件和硬件版本：
· 运行7.4或更高版本的Cisco Firepower威胁防御。

本文档中的信息都是基于特定实验室环境中的设备编写的。本文档中使用的所有设备最初均采用原始（默认）配置。如果您的网络处于活动状态，请确保您了解所有命令的潜在影响。

功能详细信息

功能功能说明

对比此版本之前的版本

在安全防火墙7.3及更低版本中	安全防火墙7.4新增功能
· 用于已知IP/端口/协议组合的应用检测仅在用尽所有其他应用检测机制后作为回退选项可用。 · 基本上，不支持对会话中第一个数据包进行检测。	· 在任何其他应用检测机制之前，评估新的LUA检测器API， · 因此，在7.4中，我们支持检测会话中的第一个数据包。

工作原理

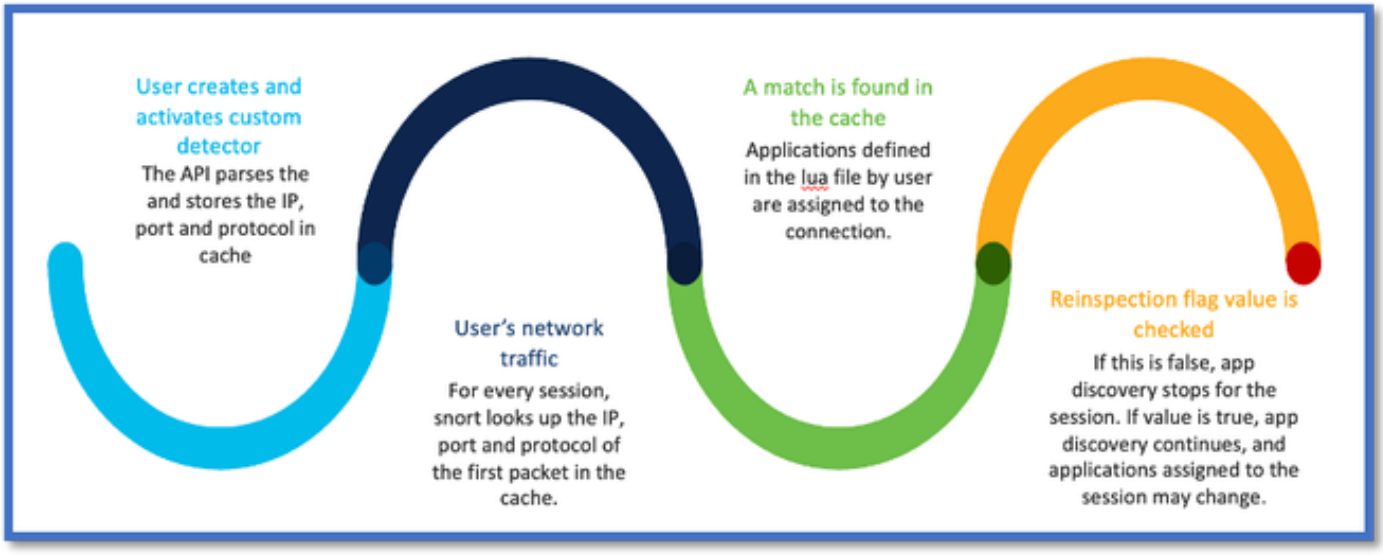
- 创建lua文件：确保该文件位于lua模板中（无语法错误）。此外，请验证文件中指定给API的参数是否正确。

- 创建新的自定义检测器：在FMC上创建新的自定义检测器并上传您的lua文件。激活检测器。

- 运行流量：将匹配自定义应用检测器中定义的IP/端口/协议组合的流量发送到设备。

- 检查连接事件：在FMC上，检查按IP和端口过滤的连接事件。将确定用户定义的应用。

AppID早期数据包检测API工作流程



来自自定义检测器的API字段说明示例

gDetector:addHostFirstPktApp

(gAppIdProto、gAppIdClient、gAppId、0, "192.0.2.1", 443, DC.ipproto.tcp)；

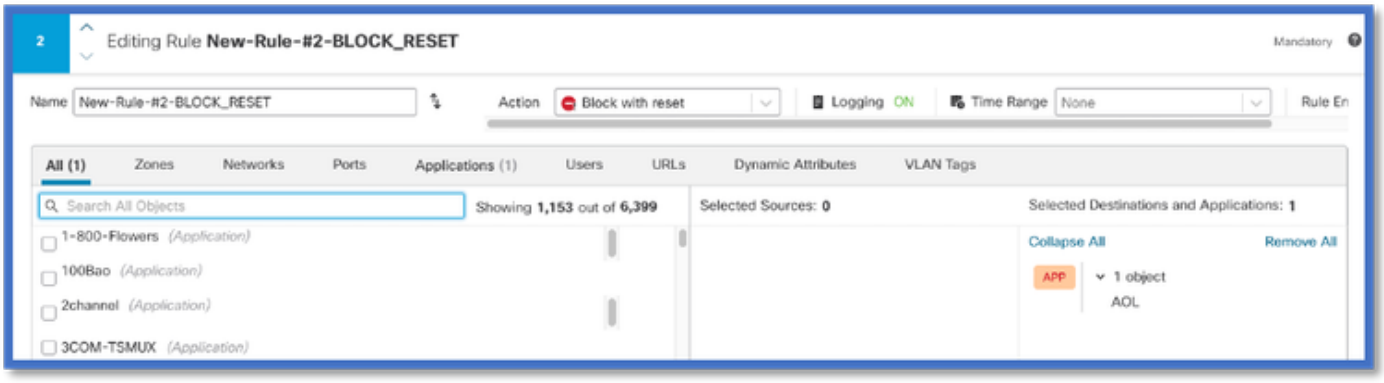
- 突出显示的参数是用户定义的恢复标志、IP地址、端口和协议值。
- 0表示通配符。

参数	说明	预期值
Respect标志	如果用户更愿意检查流量而不是根据IP/端口/协议执行防火墙操作，则可以将恢复标志值启用为1。	0 =禁用恢复或 1 =已启用重新扫描
IP Address	服务器的目标IP（子网中的单个IP或一系列IP）。会话中第1个数据包的目标IP。	192.168.4.198 或 192.168.4.198/24 或 2a03:2880：f103:83：face：b00c：0:25de或 2a03:2880：f103:83：face：b00c：0:25de/32
端口	会话中第1个数据包的目标端口。	0 到 65535

协议	网络协议	TCP/UDP/ICMP

使用案例：如何更快地阻止流量

- 策略视图：阻止应用“AOL”的规则。



- 使用curl测试流量：curl https://www.example.com v/s curl https://192.0.2.1/（TEST的IP地址之一）

<#root>

> curl https://www.example.com/

curl: (35) OpenSSL SSL_connect: SSL_ERROR_SYSCALL in connection to www.example.com:443

> curl https://192.0.2.1/

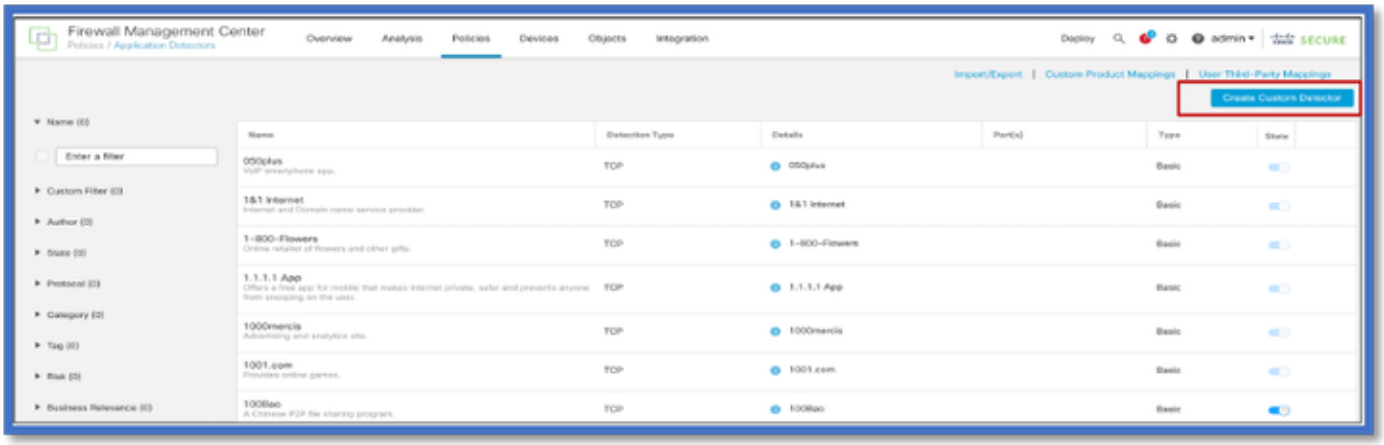
curl: (7) Failed to connect to 192.0.2.1 port 443: Connection refused

防火墙管理中心演练

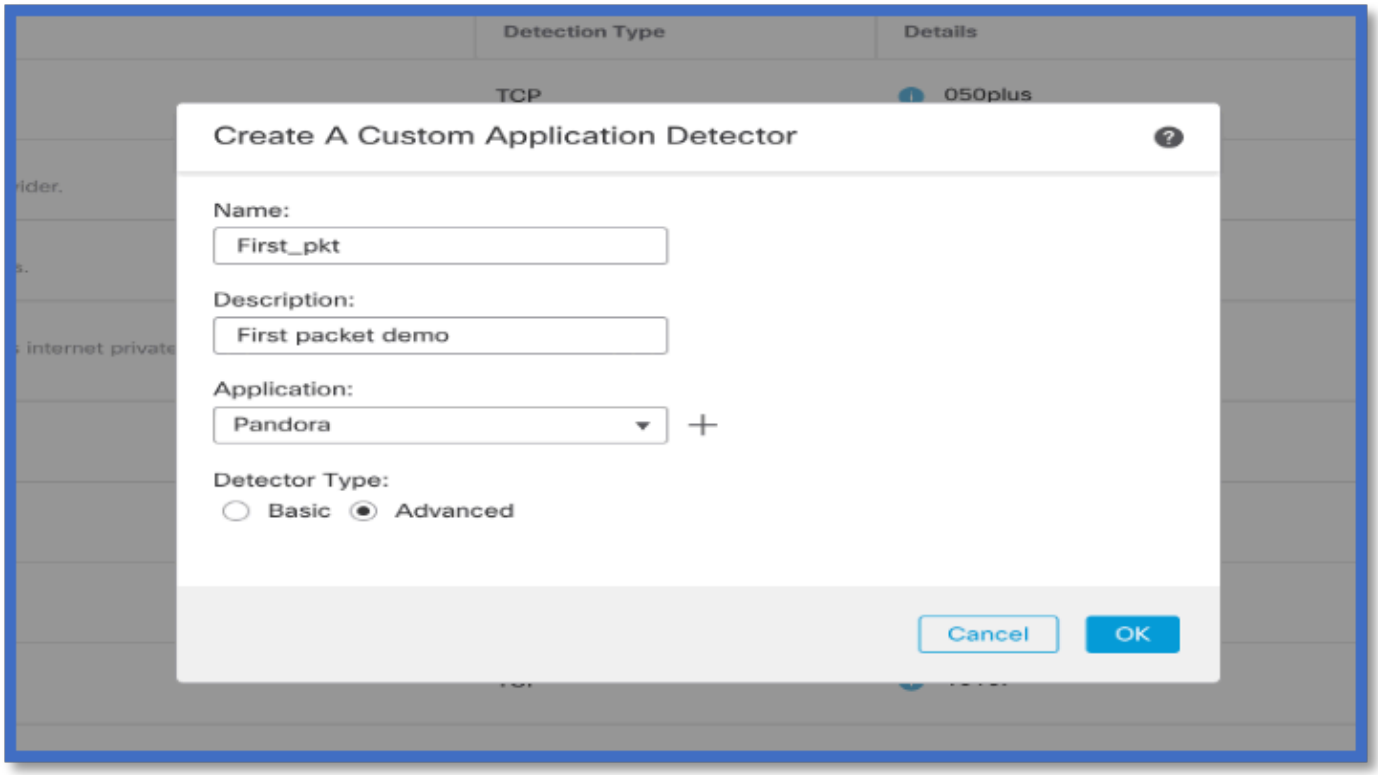
使用API创建自定义检测器的步骤

在FMC上创建新的自定义检测器，可从以下网址获得：

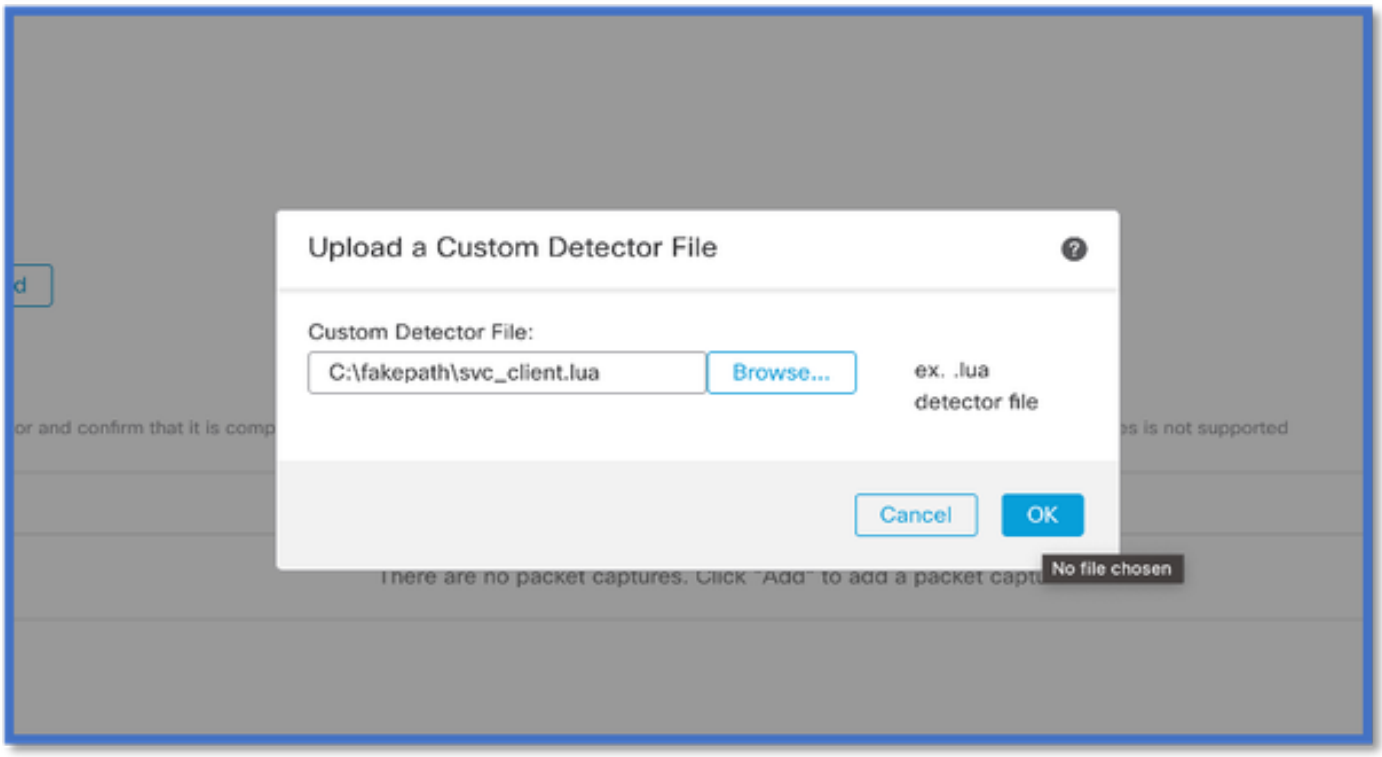
- Policies > Application Detectors > Create Custom Detector .



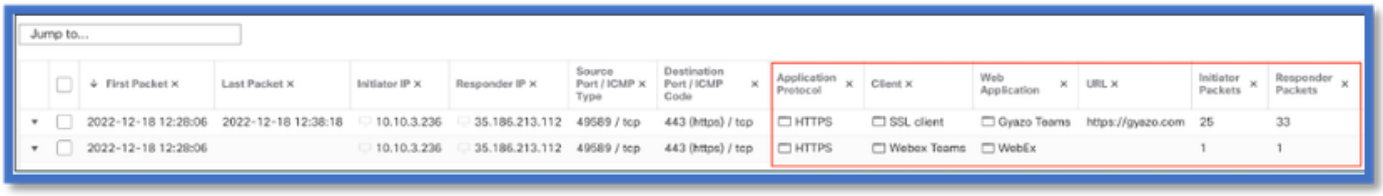
- 定义名称和说明。
  - 从下拉菜单中选择应用。
  - 选择Advanced Detector Type。



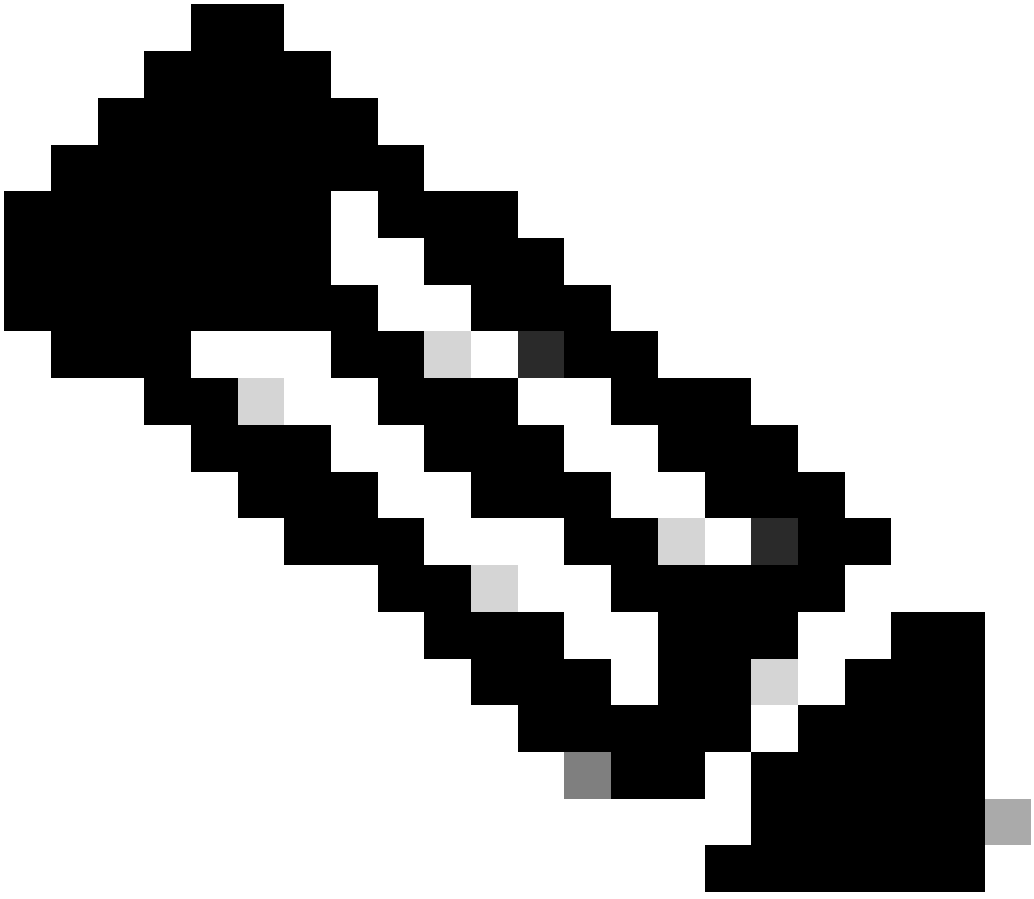
- 上传Detection Criteria下的Lua文件。保存并激活检测器。



已禁用Respect Enabled v/s



- 这两个事件显示连接开始时间v/s和连接结束时间（启用重新检查时）。



注意：注意事项：

1. “HTTPS、Webex和Webex团队”在连接开始时由API标识。由于重新检查是真实的，因此应用发现会继续，并且appId会更新为“HTTPS、SSL客户端和Gyazo团队”。

2. 注意发起方和响应方数据包的数量。常规应用检测方法需要比API更多的数据包。

故障排除/诊断

诊断概述

- 系统支持应用识别调试中添加新日志，以指示第1个数据包检测API是否发现任何应用。
- 日志还会显示用户是否选择重新检查流量。
- 用户上传的lua检测器文件的内容可以在FTD的/var/sf/appid/custom/lua/<UUID>下找到。
- 在激活检测器时，lua文件中的任何错误都会转储到/var/log/messages文件中的FTD上。

CLI：系统支持应用识别调试

<#root>

192.0.2.1 443 -> 192.168.1.16 51251 6 AS=4 ID=0 New AppId session

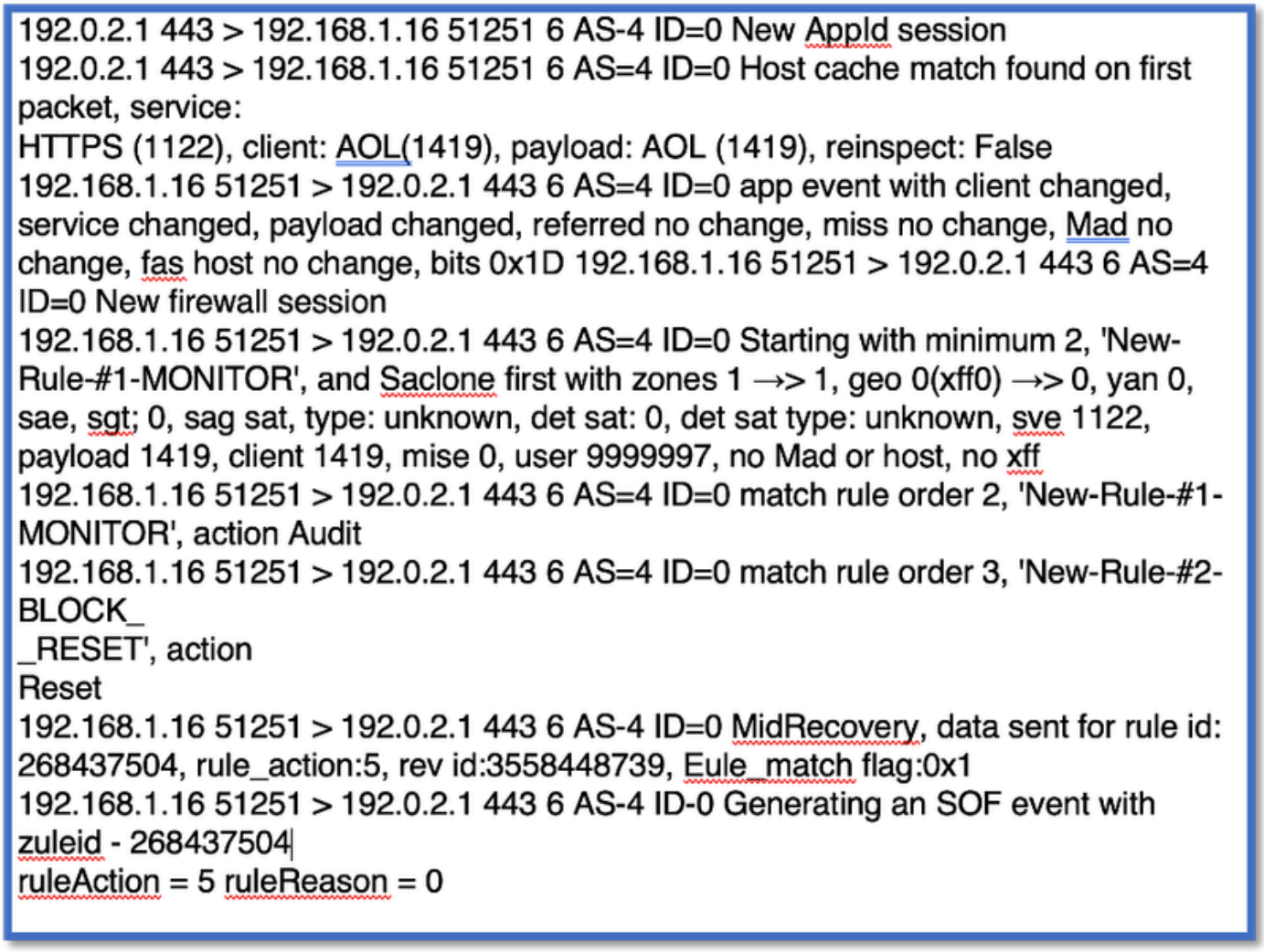
192.0.2.1 443 -> 192.168.1.16 51251 6 AS=4 ID=0 Host cache match found on first packet, service: HTTPS(1

```
192.168.1.16 51251 -> 192.0.2.1 443 6 AS=4 ID=0 app event with client changed, service changed, payload
192.168.1.16 51251 -> 192.0.2.1 443 6 AS=4 ID=0 New firewall session
192.168.1.16 51251 -> 192.0.2.1 443 6 AS=4 ID=0 Starting with minimum 2, 'New-Rule-#1-MONITOR', and Src
192.168.1.16 51251 -> 192.0.2.1 443 6 AS=4 ID=0 match rule order 2, 'New-Rule-#1-MONITOR', action Audit
```

192.168.1.16 51251 -> 192.0.2.1 443 6 AS=4 ID=0 match rule order 3, 'New-Rule-#2-BLOCK_RESET', action Re

192.168.1.16 51251 -> 192.0.2.1 443 6 AS=4 ID=0 MidRecovery data sent for rule id: 268437504, rule_acti
192.168.1.16 51251 -> 192.0.2.1 443 6 AS=4 ID=0 Generating an SOF event with rule_id = 268437504 ruleAc

192.168.1.16 51251 -> 192.0.2.1 443 6 AS=4 ID=0 reset action



AppID Lua检测器内容的位置

要确认Device/FTD上是否存在具有此新API的Lua检测器，您可以查看addHostFirstPktApp API是否正用于2个应用检测器文件夹：

1. VDB AppID检测器-/var/sf/appid/odp/lua

2. 自定义检测器-/var/sf/appid/custom/lua

例如：在每个文件夹中grep addHostFirstPktApp * 。

问题示例：

- 问题：未在FMC上激活自定义Lua检测器。

检查位置： /var/sf/appid/custom/lua/

预期结果：FMC上激活的每个自定义应用检测器必须在此存在一个文件。验证内容是否与上传的lua文件匹配。

- 问题：上传的lua检测器文件出错。

要检查的文件： /var/log/messages on FTD

错误日志:

<#root>

Dec 18 14:17:49 intel-x86-64 SF-IMS[15741]:

Error - appid: can not set env of Lua detector /ngfw/var/sf/appid/custom/lua/6698fbd6-7ede-11ed-972c-d12

故障排除步骤

问题：没有为流向用户定义的IP地址和端口的流量正确识别应用。

故障排除的步骤：

- 验证是否在FTD上正确定义并激活了LUA检测器。
  - 验证FTD上lua文件的内容，并检查激活时是否显示错误。
- 检查流量会话中第一个数据包的目的IP、端口和协议。
  - 它可以与lua检测器中定义的值相匹配。
- 检查system-support-application-identification-debug。
  - 查找行如果Host cache match found on first packet. 缺少此行，则表明该API未找到任何匹配。

限制详细信息、常见问题和解决方法

在7.4中，没有使用API的UI。未来版本中将增加UI支持。

修订历史纪录

修订版	发布日期	备注
1.0	2024年7月18日	首次公开发布

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）。