Konfigurieren von AppID Early Packet Detection in Secure Firewall Threat Defense 7.4

Inhalt

Einleitung

In diesem Dokument wird beschrieben, wie Sie AppID Early Packet Detection in Cisco Secure Firewall 7.4 konfigurieren.

Hintergrund - Problem (Kundenanforderungen)

- Die Anwendungserkennung durch Deep Packet Inspection kann mehr als ein Paket zur Identifizierung des Datenverkehrs benötigen.
- In manchen Fällen, in denen die IP-Adresse und/oder der Port für einen Anwendungsserver bekannt sind, können Sie die Überprüfung zusätzlicher Pakete vermeiden.

Neuerungen

- Eine neue Snort-basierte Lua AppID API ermöglicht die Zuordnung von IP-Adresse, Port und Protokoll zu den jeweiligen:
  - Anwendungsprotokoll (Service-Appid),
  - Client-Anwendung (Client-App) und
  - Webanwendung (Payload-Appid).
- Mit dieser API für die Anwendungserkennung können benutzerdefinierte Anwendungsdetektoren auf FMC erstellt werden.
- Sobald dieser Detektor aktiviert ist, ermöglicht uns diese neue API, Anwendungen auf dem ersten Paket in einer Sitzung zu identifizieren.

Funktionsüberblick

- Die API umfasst folgende Komponenten:
  - **addHostFirstPktApp** (protocol_appId, client_appId, payload_appId, IP-Adresse, Port, Protokoll, reinspect)

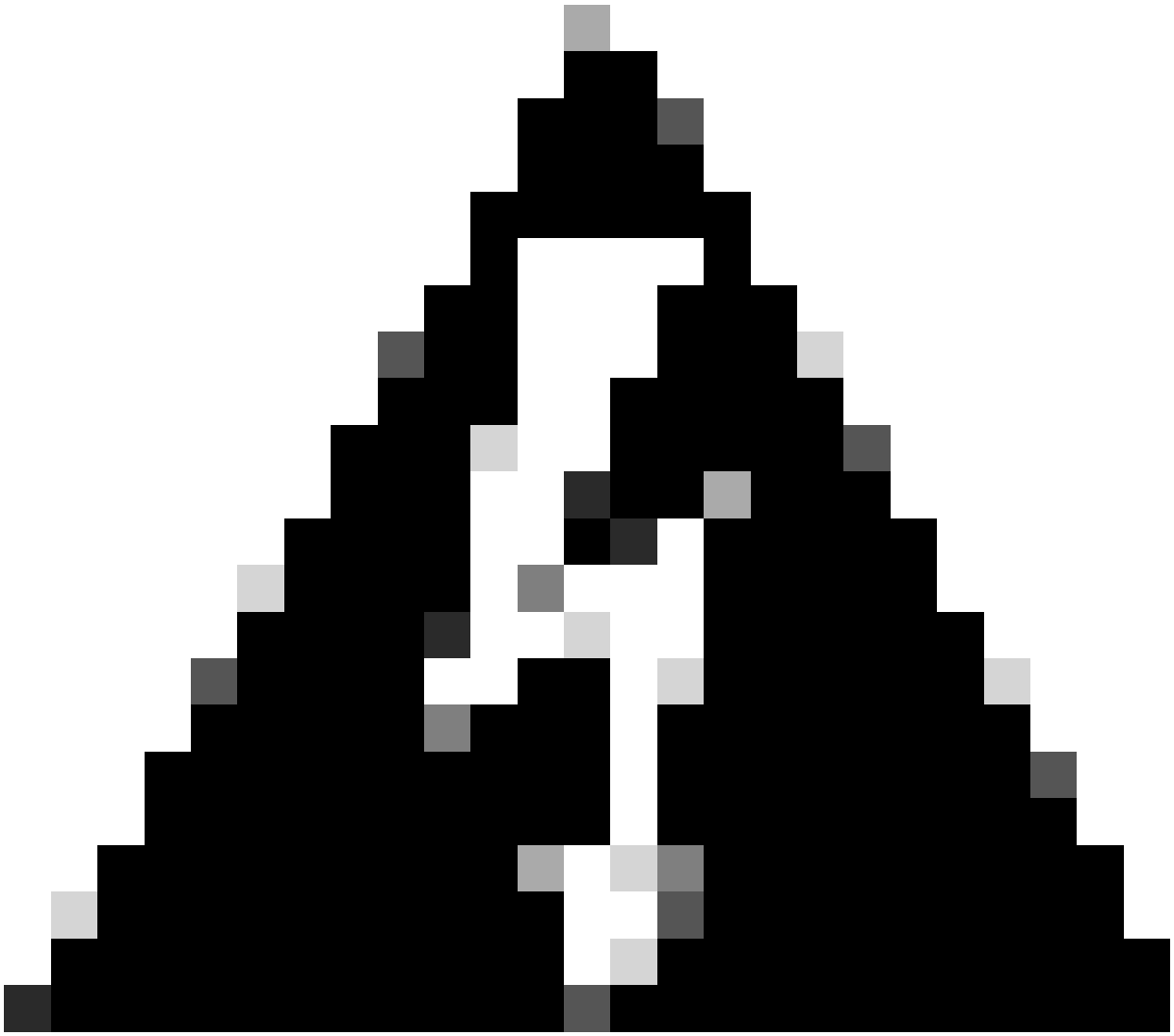
- Für jede im benutzerdefinierten App-Detektor erstellte Zuordnung wird ein Cache-Eintrag erstellt.
- Das erste Paket aller eingehenden Sitzungen wird daraufhin überprüft, ob eine Übereinstimmung im Cache gefunden wurde.
- Sobald eine Übereinstimmung gefunden wurde, weisen wir die entsprechenden Appids für die Sitzung zu, und der App-Erkennungsprozess wird beendet.
- Benutzer haben die Möglichkeit, den Datenverkehr auch dann erneut zu überprüfen, wenn die API eine Übereinstimmung gefunden hat.
- Das Argument reinspect ist ein boolescher Wert, der angibt, ob die im ersten Paket gefundenen Anwendungen erneut geprüft werden müssen oder nicht.
- Wenn eine erneute Überprüfung zutrifft, wird die Anwendungserkennung fortgesetzt, auch wenn die API eine Übereinstimmung findet.
- In diesem Fall können sich die im ersten Paket zugewiesenen Appliances ändern.

Voraussetzungen, unterstützte Plattformen, Lizenzierung

Software- und Hardware-Mindestanforderungen

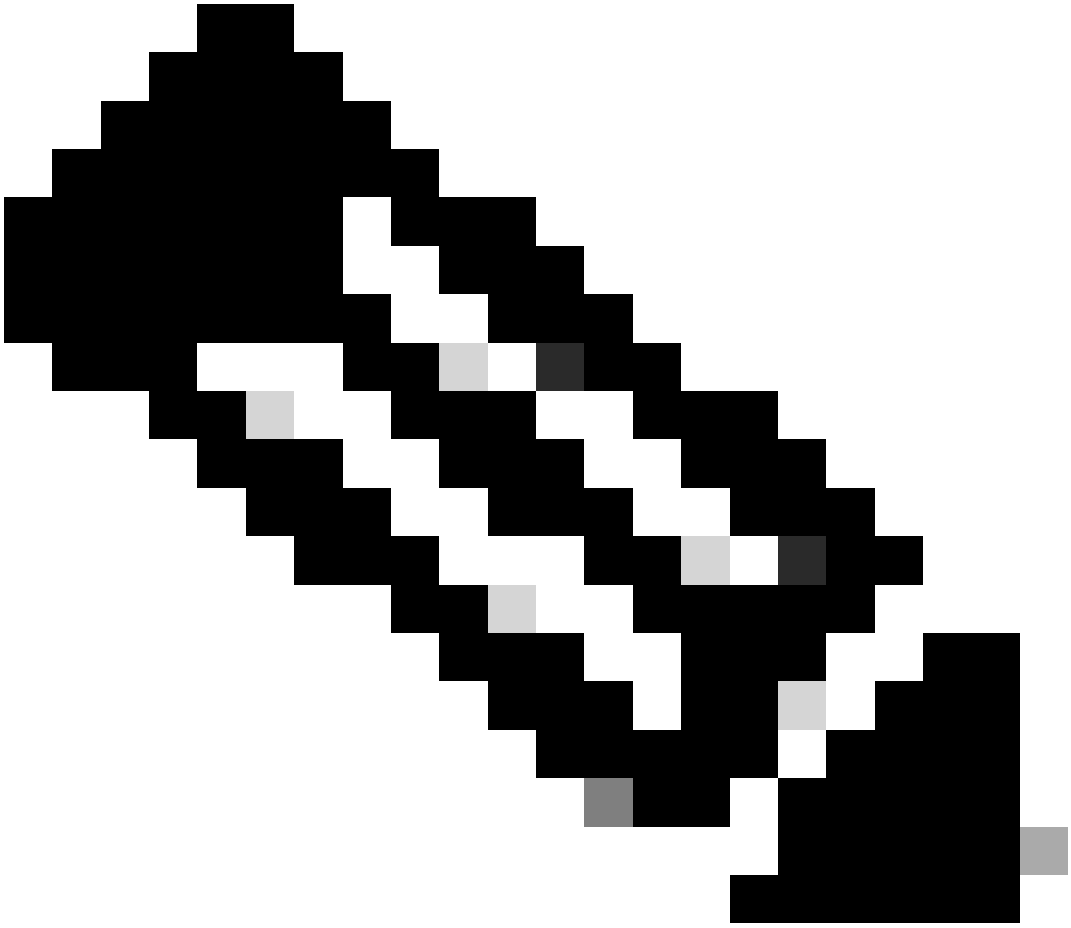
Anwendungs- und Mindestversion	Unterstützte verwaltete Plattform(en) und Version	Manager	Hinweise

Sichere Firewall 7.4 Verwenden von Snort3	Alle Plattformen, die FTD 7.4 unterstützen	FMC vor Ort + FTD	Dies ist eine geräteseitige Funktion. FTD muss auf Version 7.4 installiert sein.



Warnung: Snort 2 unterstützt diese API nicht.

Unterstützung von Snort 3, Multi-Instance und HA/Clustering



Hinweis: Erfordert, dass Snort 3 die Erkennungs-Engine ist.

FTD	
Mehrere Instanzen unterstützt?	Ja
Unterstützung durch HA-Geräte	Ja
Unterstützt durch Cluster-Geräte?	Ja

Verwendete Komponenten

Die Informationen in diesem Dokument basierend auf folgenden Software- und Hardware-Versionen:

- Cisco Firepower Threat Defense mit Version 7.4 oder höher

Die Informationen in diesem Dokument beziehen sich auf Geräte in einer speziell eingerichteten Testumgebung. Alle Geräte, die in diesem Dokument benutzt wurden, begannen mit einer gelöschten (Nichterfüllungs) Konfiguration. Wenn Ihr Netzwerk in Betrieb ist, stellen Sie sicher, dass Sie die möglichen Auswirkungen aller Befehle kennen.

Funktionsdetails

Beschreibung der Funktionsmerkmale

Vergleich mit Vorgängerversion

In Secure Firewall 7.3 und niedriger	Neu bei Secure Firewall 7.4
· Die Anwendungserkennung für eine bekannte IP/Port/Protocol-Kombination war erst als Fallback-Option verfügbar, nachdem alle anderen Mechanismen zur Anwendungserkennung ausgeschöpft waren. · Im Wesentlichen wurde die Erkennung auf dem ersten Paket in einer Sitzung nicht unterstützt.	· Die neue API für den Lua-Detektor wird vor allen anderen Mechanismen zur Erkennung von Anwendungen ausgewertet. · In Version 7.4 unterstützen wir daher die Erkennung des ersten Pakets in einer Sitzung.

So funktioniert es

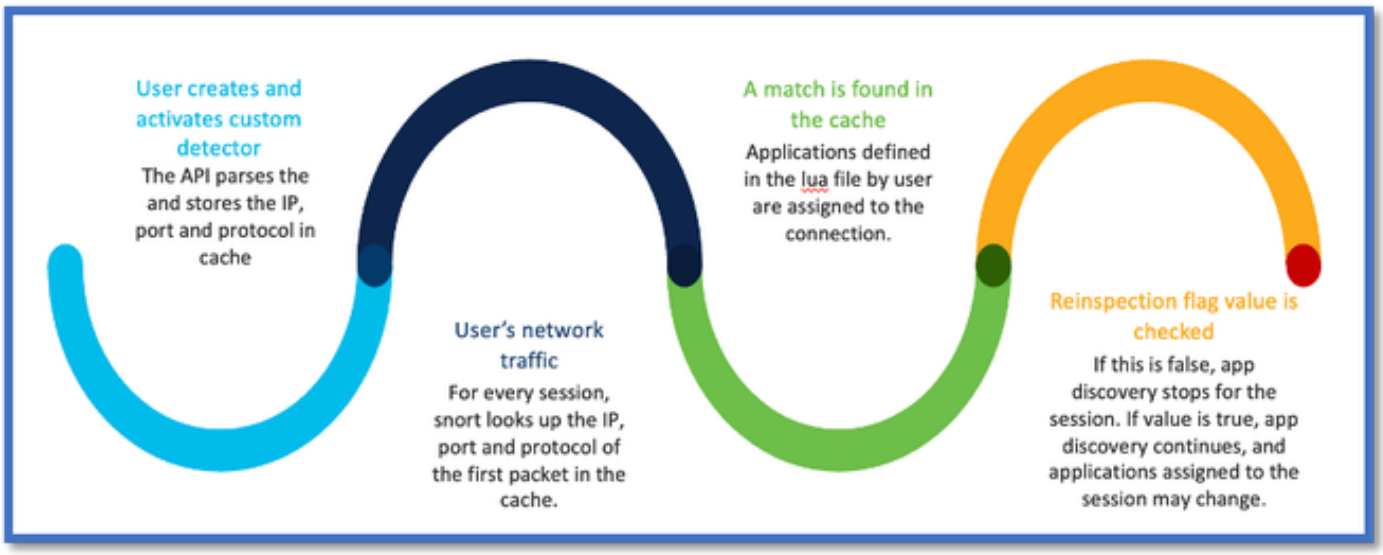
- Erstellen einer Lua-Datei: Stellen Sie sicher, dass sich die Datei in der Lua-Vorlage befindet (keine Syntaxfehler). Überprüfen Sie auch, ob die Argumente, die der API in der Datei gegeben wurden, richtig sind.

- Erstellen Sie einen neuen benutzerdefinierten Detektor: Erstellen Sie einen neuen benutzerdefinierten Detektor auf FMC und laden Sie Ihre Lua-Datei in es. Aktivieren Sie den Detektor.

- Datenverkehr ausführen: Datenverkehr, der der im benutzerdefinierten App-Detektor definierten IP/Port/Protokoll-Kombination entspricht, wird an das Gerät gesendet.

- Verbindungsereignisse überprüfen: Überprüfen Sie auf FMC die Verbindungsereignisse, die nach IP und Port gefiltert wurden. Benutzerdefinierte Anwendungen werden identifiziert.

AppID Early Packet Detection API-Workflow



API-Feldbeschreibung aus Beispiel für benutzerdefinierten Detector

gDetector:addHostFirstPktApp

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

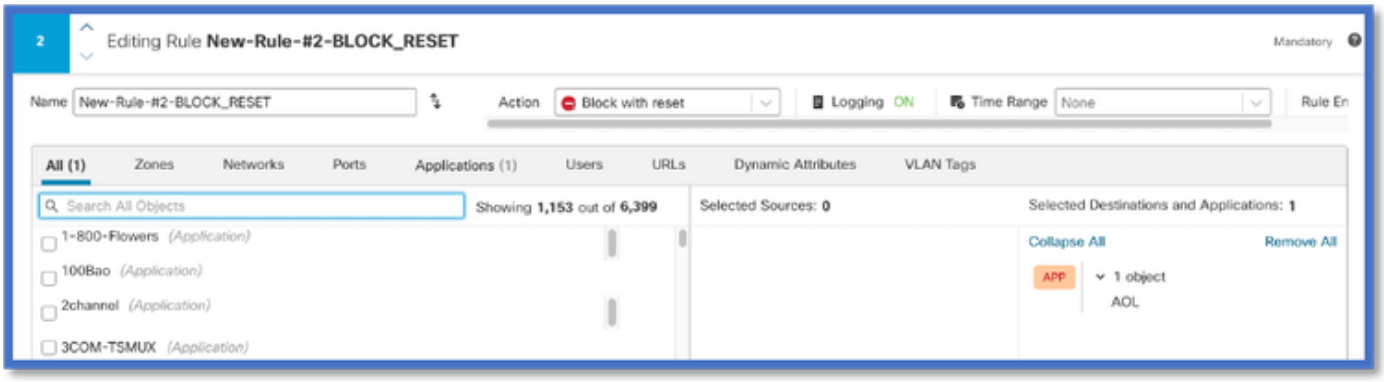
- Die hervorgehobenen Argumente sind die benutzerdefinierten Werte für das Re-Spect-Flag, die IP-Adresse, den Port und das Protokoll.
- 0 steht für einen Platzhalter.

Argumente	Erläuterung	Erwartete Werte
Flag erneut prüfen	Wenn ein Benutzer den Datenverkehr lieber inspizieren möchte, anstatt eine Firewall-Aktion auf der Grundlage von IP/Port/Protokoll durchzuführen, kann er den Wert für die erneute Überprüfung auf 1 aktivieren.	0 = Überprüfung deaktiviert oder 1 = Erneute Überprüfung aktiviert

IP-Adresse	Ziel-IP (eine oder mehrere IPs in einem Subnetz) des Servers. Ziel-IP-Adresse des ersten Pakets in einer Sitzung.	192.168.4.198 ODER 192.168.4.198/24 ODER 2a03:2880:f103:83:face:b00c:0:25de ODER 2a03:2880:f103:83:face:b00c:0:25de/32
Anschluss	Ziel-Port des 1. Pakets in einer Sitzung.	0 bis 65535
Protokolle	Netzwerkprotokoll	TCP/UDP/ICMP

Anwendungsfall: Schnellere Blockierung von Datenverkehr

- Richtlinienansicht: Sperrregel für die Anwendung "AOL".



- Testen des Datenverkehrs mit curl with: curl https://www.example.com v/s curl https://192.0.2.1/ (eine der IP-Adressen von TEST)

<#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/
```

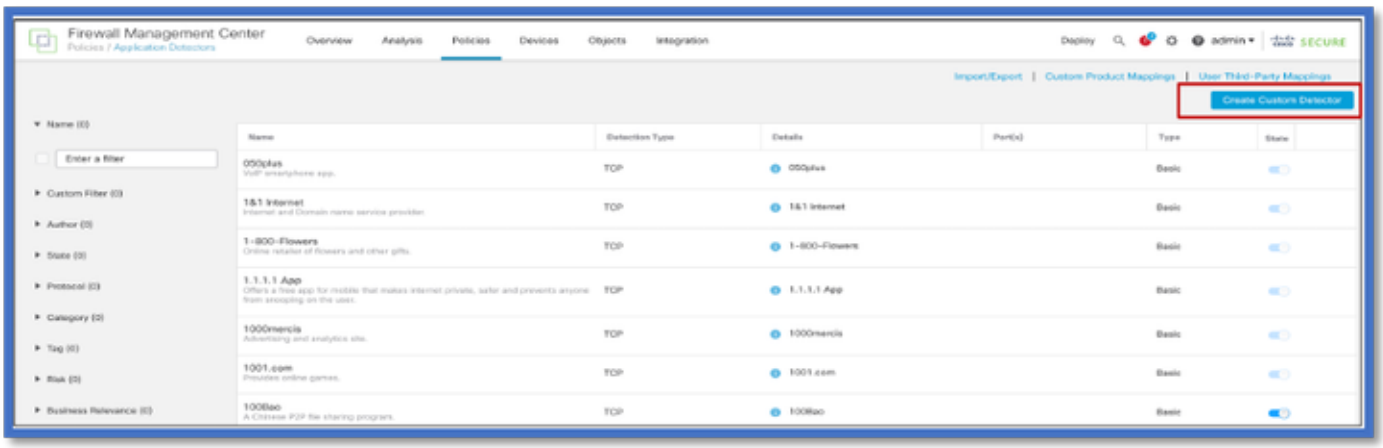
```
curl: (7) Failed to connect to 192.0.2.1 port 443: Connection refused
```

Firewall Management Center - exemplarische Vorgehensweise

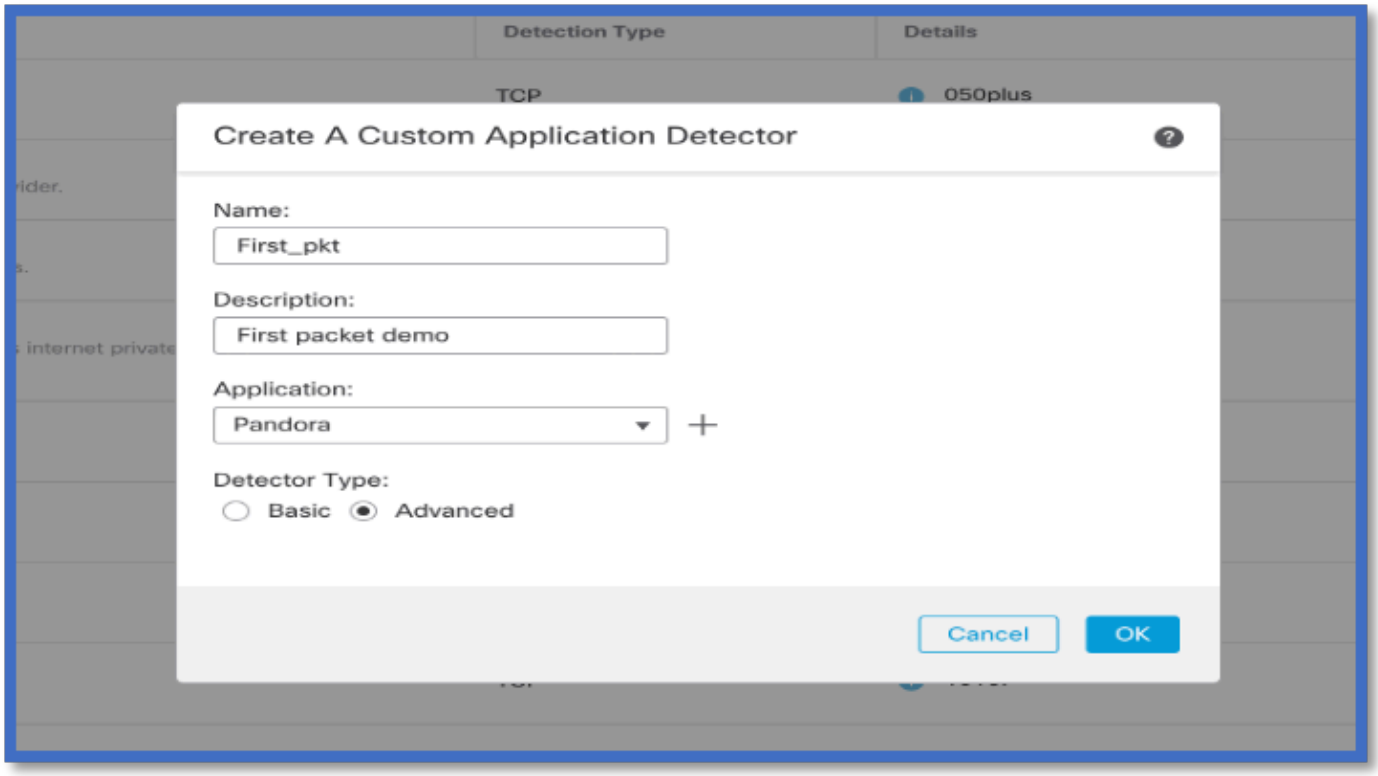
Schritte zum Erstellen eines benutzerdefinierten Detektors mithilfe der API

Erstellen Sie einen neuen benutzerdefinierten Detektor auf dem FMC aus:

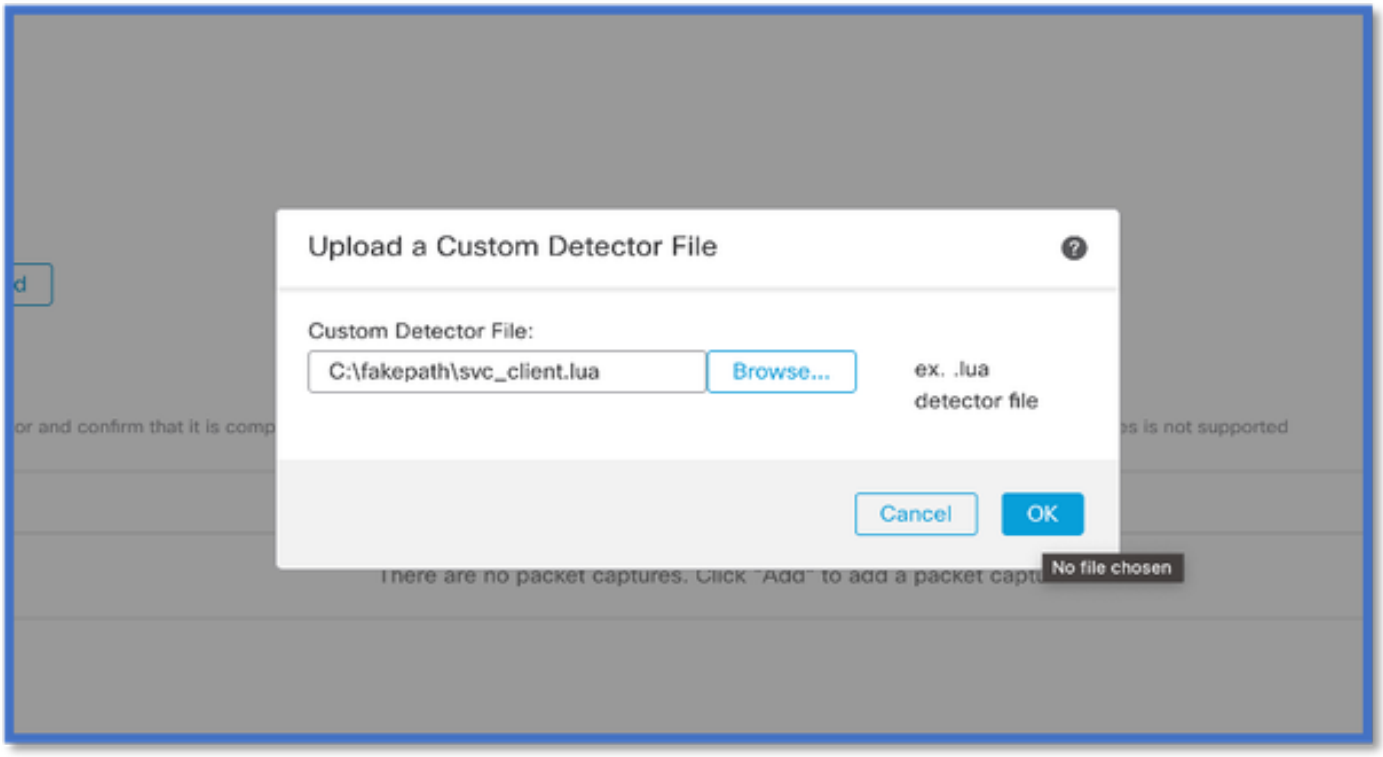
- Policies > Application Detectors > Create Custom Detector .



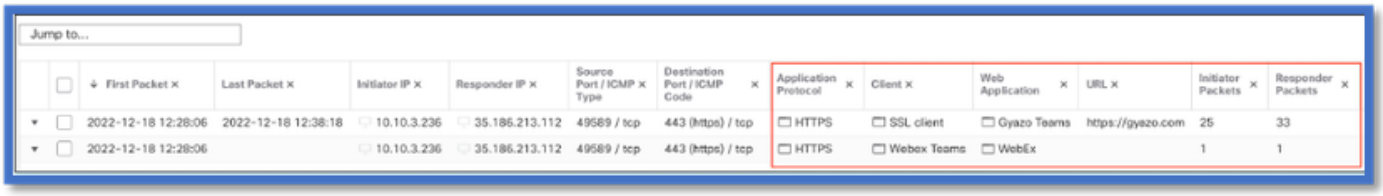
- Definieren Sie einen Namen und eine Beschreibung.
  - Wählen Sie die Anwendung aus dem Dropdown-Menü aus.
  - Wählen Sie Erweiterter Erkennungstyp aus.



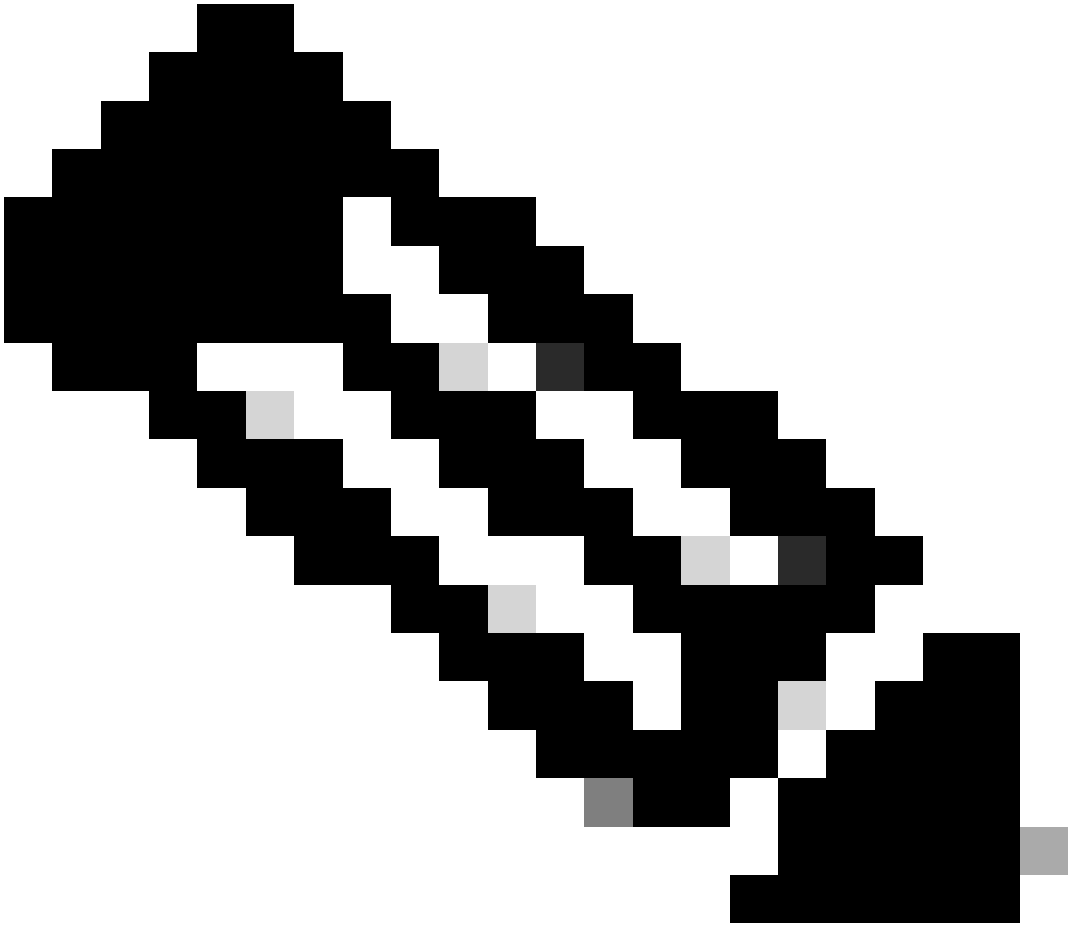
- Laden Sie die Lua-Datei unter "Erkennungskriterien" hoch. Speichern und aktivieren Sie den Detektor.



Respect aktiviert/deaktiviert



- Die beiden Ereignisse zeigen den Beginn der Verbindung v/s das Ende der Verbindung an, wenn die erneute Überprüfung aktiviert ist.



Hinweis: Wichtige Punkte:

1. Die "HTTPS-, Webex- und Webex-Teams" werden zu Beginn der Verbindung durch die API identifiziert. Da eine erneute Überprüfung zutrifft, wird die App-Erkennung fortgesetzt, und die App-IDs werden auf "HTTPS, SSL Client and Gyazo Teams" aktualisiert.

2. Beachten Sie die Anzahl der Initiator- und Responder-Pakete. Für reguläre Methoden zur Anwendungserkennung sind deutlich mehr Pakete erforderlich als für die API.

Fehlerbehebung/Diagnose

Diagnoseübersicht

- Neue Protokolle werden beim Debuggen der Anwendungserkennung für die Systemunterstützung hinzugefügt, um anzugeben, ob Anwendungen von der ersten Paketerkennungs-API gefunden wurden.
- Die Protokolle zeigen auch an, ob der Benutzer die erneute Überprüfung des Datenverkehrs gewählt hat.
- Der Inhalt der vom Benutzer hochgeladenen Datei für den Lua-Detektor ist auf der FTD unter /var/sf/appid/custom/lua/<UUID> zu finden.
- Alle Fehler in der lua-Datei werden zum Zeitpunkt der Aktivierung des Detektors in der Datei /var/log/messages auf der FTD abgelegt.

CLI: System Support Application-Identification-Debug

<#root>

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

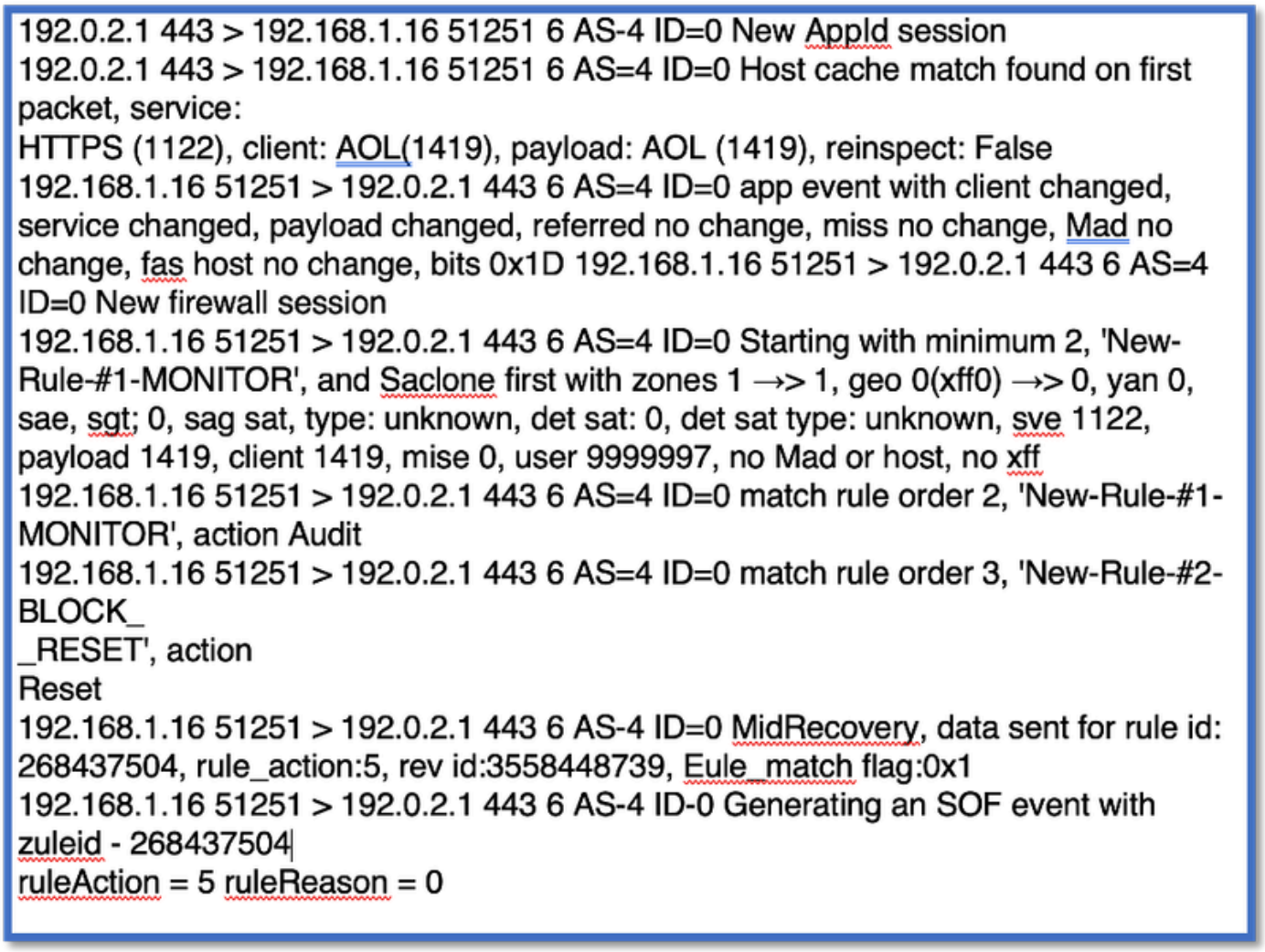
192.0.2.1 443 -> 192.168.1.16 51251 6 AS=4 ID=0 Host cache match found on first packet, service: HTTPS(1

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



Standort der AppID Lua Detektoren Inhalt

Um zu überprüfen, ob der Lua Detector mit dieser neuen API auf dem Gerät/FTD vorhanden ist, können Sie überprüfen, ob die addHostFirstPktApp API in den beiden Anwendungsdetektor-Ordnern verwendet wird:

1. VDB AppID-Detektoren -/var/sf/appid/odp/lua

2. Benutzerdefinierte Detektoren -/var/sf/appid/custom/lua

Beispiel:grep addHostFirstPktApp * in jedem Ordner.

Beispielprobleme:

- Ausgabe: Benutzerdefinierter Lua-Detektor nicht auf FMC aktiviert.

Zu prüfender Ort: /var/sf/appid/custom/lua/

Erwartetes Ergebnis: Hier muss eine Datei für jeden auf dem FMC aktivierten benutzerdefinierten App-Detektor vorhanden sein.

Überprüfen Sie, ob der Inhalt mit der hochgeladenen Lua-Datei übereinstimmt.

- Problem: Die hochgeladene Lua-Detektordatei weist Fehler auf.

Zu prüfende Datei: /var/log/messages on FTD

Fehlerprotokoll:

<#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

Schritte zur Fehlerbehebung

Problem: Anwendungen wurden nicht korrekt für den an die benutzerdefinierte IP-Adresse und den benutzerdefinierten Port gehenden Datenverkehr identifiziert.

Schritte zur Fehlerbehebung:

- Stellen Sie sicher, dass der Lua-Detektor richtig definiert und auf dem FTD aktiviert ist.
  - Überprüfen Sie den Inhalt der Lua-Datei auf dem FTD, und stellen Sie sicher, dass bei der Aktivierung keine Fehler angezeigt werden.
- Überprüfen Sie die Ziel-IP-Adresse, den Port und das Protokoll des ersten Pakets in der Datenverkehrssitzung.
  - Er kann mit den im Lua-Detektor definierten Werten übereinstimmen.

- Überprüfen Sie System-support-application-identification-debug.

    - Suchen Sie nach der Zeile Host cache match found on first packet. Wenn diese fehlt, weist dies darauf hin, dass die API keine Übereinstimmung gefunden hat.

Einzelheiten zu Einschränkungen, häufige Probleme und Problemumgehungen

In Version 7.4 gibt es keine Benutzeroberfläche für die Verwendung der API. UI-Unterstützung wird in zukünftigen Versionen hinzugefügt.

Revisionsverlauf

Revision	Veröffentlichungsdatum	Kommentare
1.0	18. Juli 2024	Erstveröffentlichung

Informationen zu dieser Übersetzung

Cisco hat dieses Dokument maschinell übersetzen und von einem menschlichen Übersetzer editieren und korrigieren lassen, um unseren Benutzern auf der ganzen Welt Support-Inhalte in ihrer eigenen Sprache zu bieten. Bitte beachten Sie, dass selbst die beste maschinelle Übersetzung nicht so genau ist wie eine von einem professionellen Übersetzer angefertigte. Cisco Systems, Inc. übernimmt keine Haftung für die Richtigkeit dieser Übersetzungen und empfiehlt, immer das englische Originaldokument (siehe bereitgestellter Link) heranzuziehen.