Dépannage de la gestion des chemins de données par UTD et filtrage des URL

Table des matières

- Introduction
- Informations générales
  - Vue de haut niveau Datapath
    - Du LAN/WAN au conteneur
    - Du conteneur au LAN/WAN
  - Approfondissement Datapath
    - Paquet entrant du côté LAN ou WAN vers le conteneur
    - Paquet entrant du conteneur vers le côté LAN ou WAN
  - Intégration de la journalisation de flux UTD avec Packet-trace
- Prérequis :
  - Vérification de la compatibilité de la version UTD avec IOS XE
  - Vérifier la configuration valide du serveur de noms dans le conteneur
- Problème 1
  - Dépannage
  - Cause première
- Problème 2
  - Dépannage
  - Cause première
- Problème 3
  - Dépannage
    - Étape 1 : collecte de statistiques générales
    - Étape 2 : Examinez le fichier journal de l'application
- Problème 4
  - Dépannage
  - Cause première
- Références

Introduction

Ce document décrit comment dépanner Unified Threat Defense (UTD) également connu sous le nom de Snort et de filtrage d'URL (Uniform Resource Locator) sur les routeurs IOS® XE WAN Edge.

Informations générales

Snort est le système de prévention des intrusions (IPS) le plus déployé au monde. Depuis 2013, la société qui a créé une version commerciale du logiciel Snort, Sourcefire, est rachetée par Cisco. À

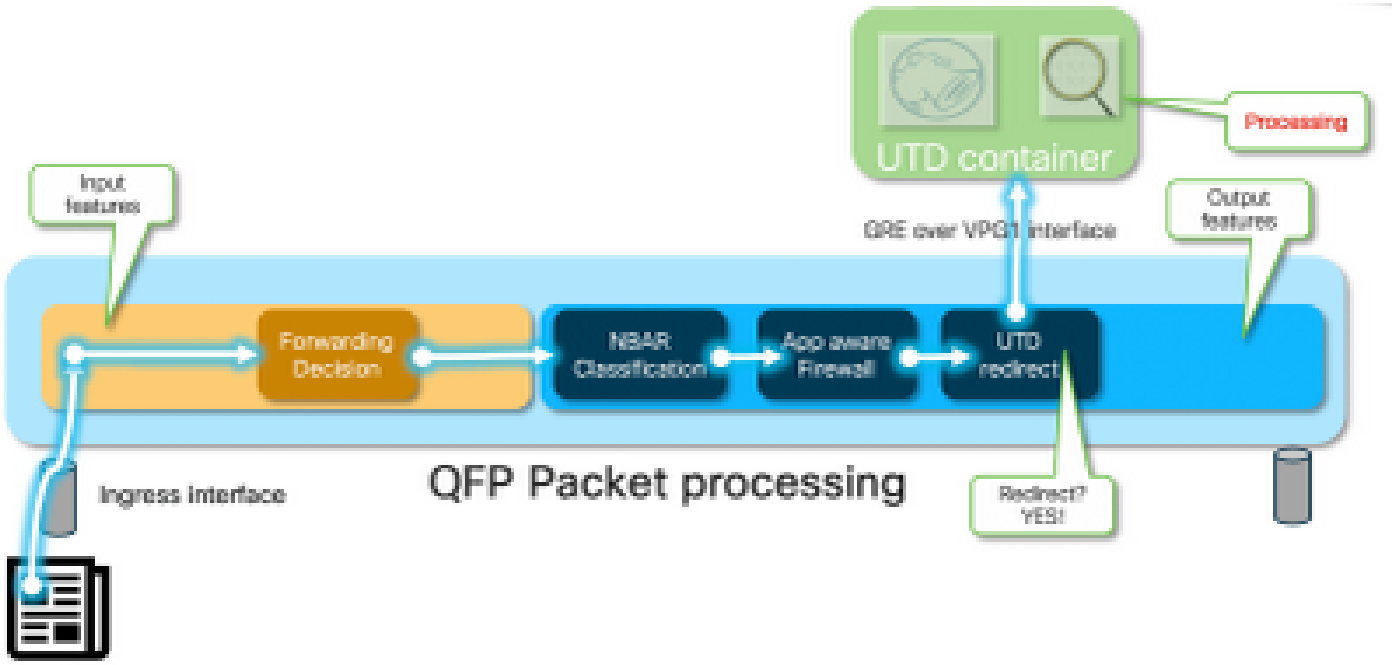
partir de la version 16.10.1 du logiciel SD-WAN IOS® XE, des conteneurs de filtrage UTD/URF ont été ajoutés à la solution SD-WAN de Cisco.

Le conteneur s'enregistre auprès du routeur IOS®XE à l'aide de l'infrastructure app-nav. L'explication de ce processus sort du cadre du présent document.

Vue de haut niveau Datapath

À un niveau élevé, le chemin de données ressemble à ceci :

Du LAN/WAN au conteneur



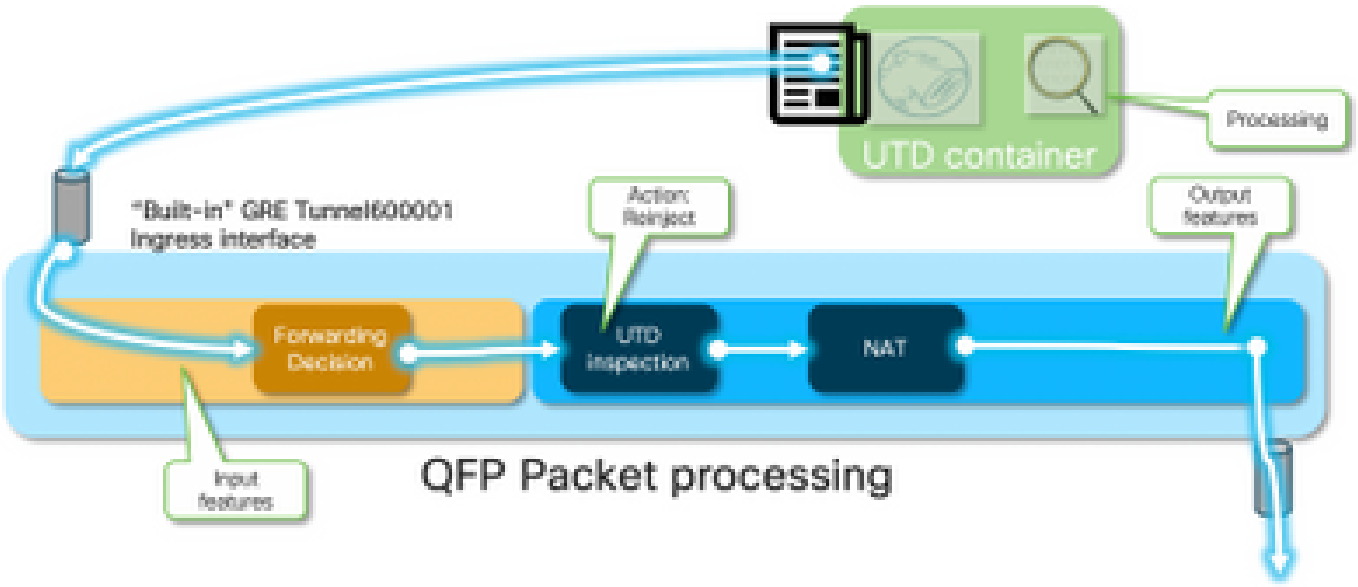
Le trafic provient du côté LAN. Comme IOS® XE sait que le conteneur est en bon état, il détourne le trafic vers le conteneur UTD. La dérivation utilise l'interface VirtualPortGroup1 comme interface de sortie, qui encapsule le paquet à l'intérieur d'un tunnel GRE (Generic Routing Encapsulation).

Le routeur exécute l'action « PUNT » en utilisant « cause : 64 (Service Engine packet) » et envoie le trafic vers le processeur de routage (RP). Un en-tête de point est ajouté et le paquet est envoyé au conteneur à l'aide d'une interface de sortie interne vers le conteneur "[internal0/0/svc_eng:0]"

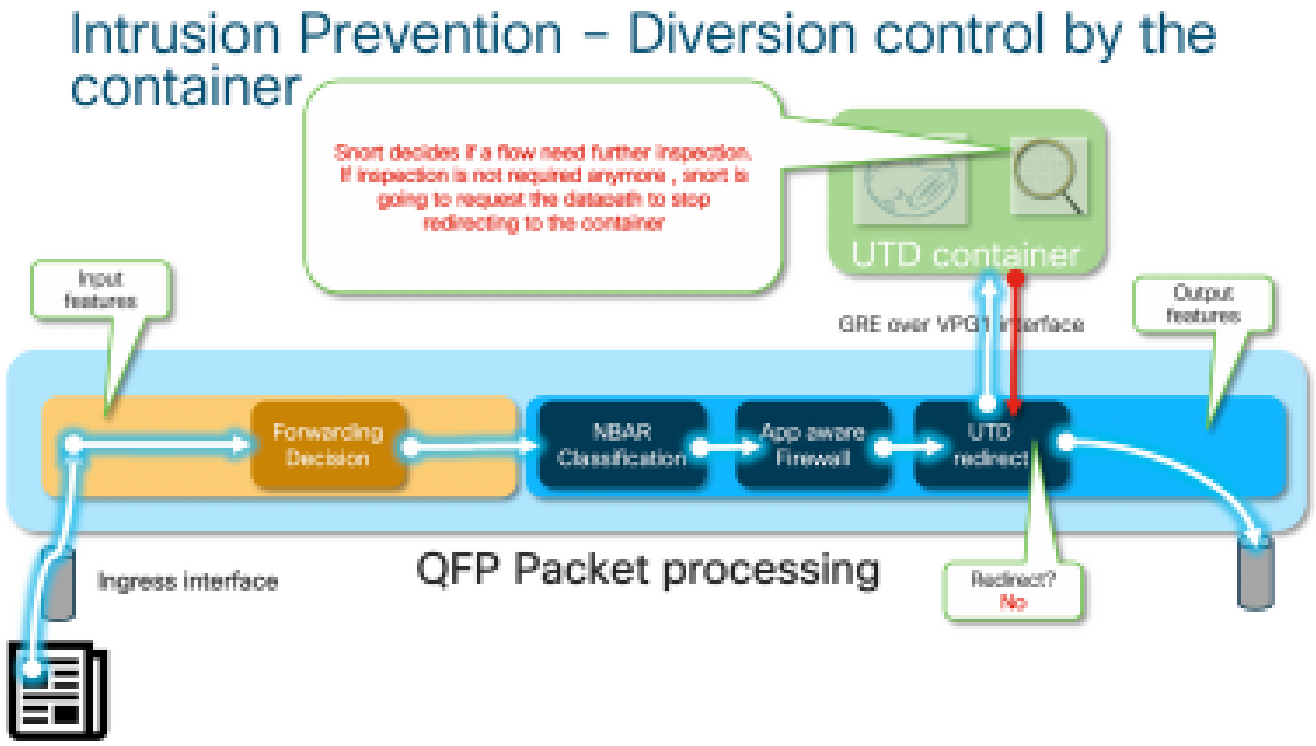
À ce stade, Snort exploite ses préprocesseurs et ses jeux de règles. Le paquet peut être abandonné ou transféré en fonction des résultats du traitement.

Du conteneur au LAN/WAN

En supposant que le trafic n’est pas censé être abandonné, le paquet est renvoyé au routeur après le traitement UTD. Il apparaît sur le processeur de flux quantique (QFP) comme provenant du tunnel6000001. Ensuite, il est traité par le routeur et doit être (si possible) routé vers l'interface WAN.



Le conteneur contrôle le résultat de la déviation dans l'inspection UTD dans le chemin de données IOS® XE.



Par exemple, avec le flux HTTPS, les préprocesseurs sont intéressés de voir les paquets Hello/Hello client du serveur avec la négociation TLS. Par la suite, le flux n'est pas redirigé car l'inspection du trafic chiffré TLS n'a que peu de valeur.

Approfondissement Datapath

Du point de vue du traceur de paquets, ces actions vont être vues (192.168.16.254 est un client Web) :

```
debug platform condition ipv4 192.168.16.254/32 both
debug platform condition start
debug platform packet-trace packet 256 fia-trace data-size 3000
```

Paquet entrant du côté LAN ou WAN vers le conteneur

Dans ce scénario particulier, le paquet suivi provient du réseau local. Du point de vue de la redirection, il existe des différences importantes si le flux provient d'un réseau local ou d'un réseau étendu.

Le client tente d'accéder à www.cisco.com sur HTTPS

```
cedge6#show platform packet-trace packet 14
Packet: 14          CBUG ID: 3849209
Summary
  Input     : GigabitEthernet2
  Output    : internal0/0/svc_eng:0
  State     : PUNT 64  (Service Engine packet)
  Timestamp
    Start   : 1196238208743284 ns (05/08/2019 10:50:36.836575 UTC)
    Stop    : 1196238208842625 ns (05/08/2019 10:50:36.836675 UTC)
Path Trace
  Feature: IPV4(Input)
    Input       : GigabitEthernet2
    Output      : <unknown>
    Source      : 192.168.16.254
    Destination : 203.0.113.67
    Protocol    : 6 (TCP)
      SrcPort   : 35568
      DstPort   : 443
  Feature: DEBUG_COND_INPUT_PKT
    Entry       : Input - 0x8177c67c
    Input       : GigabitEthernet2
    Output      : <unknown>
    Lapsed time : 2933 ns
<snip>
```

Le trafic correspondant à la condition est suivi en entrée sur l'interface GigabitEthernet2.

```
  Feature: UTD Policy (First FIA)
    Action         : Divert
    Input interface : GigabitEthernet2
    Egress interface: GigabitEthernet3
  Feature: OUTPUT_UTD_FIRST_INSPECT
    Entry       : Output - 0x817cc5b8
    Input       : GigabitEthernet2
    Output      : GigabitEthernet3
    Lapsed time : 136260 ns
  Feature: UTD Inspection
    Action         : Divert             <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
    Input interface : GigabitEthernet2
```

```
    Egress interface: GigabitEthernet3
  Feature: OUTPUT_UTD_FINAL_INSPECT
    Entry       : Output - 0x817cc5e8
    Input       : GigabitEthernet2
    Output      : GigabitEthernet3
    Lapsed time : 43546 ns
<snip>
```

Sur le Feature Invocation Array (FIA) de sortie de l'interface de sortie, UTD FIA a décidé de renvoyer ce paquet vers le conteneur.

```
  Feature: IPV4_OUTPUT_LOOKUP_PROCESS_EXT
    Entry       : Output - 0x81781bb4
    Input       : GigabitEthernet2
    Output      : Tunnel6000001
<removed>
  Feature: IPV4_OUTPUT_LOOKUP_PROCESS_EXT
    Entry       : Output - 0x81781bb4
    Input       : GigabitEthernet2
    Output      : Tunnel6000001
<removed>
  Feature: IPV4_INPUT_LOOKUP_PROCESS_EXT
    Entry       : Output - 0x8177c698
    Input       : Tunnel6000001
    Output      : VirtualPortGroup1
    Lapsed time : 880 ns
<snip>
```

Le paquet est placé sur le tunnel par défaut Tunnel600001 et est routé à travers l'interface VPG1. À ce stade, le paquet d'origine est encapsulé dans GRE.

```
  Feature: OUTPUT_SERVICE_ENGINE
    Entry       : Output - 0x817c6b10
    Input       : Tunnel6000001
    Output      : internal0/0/svc_eng:0
    Lapsed time : 15086 ns
<removed>
  Feature: INTERNAL_TRANSMIT_PKT_EXT
    Entry       : Output - 0x8177c718
    Input       : Tunnel6000001
    Output      : internal0/0/svc_eng:0
    Lapsed time : 43986 ns
```

Le paquet est transmis en interne au conteneur.

Remarque : des informations supplémentaires sur les composants internes du conteneur sont fournies à titre d'information uniquement. Le conteneur UTD n'est pas accessible via l'interface de ligne de commande normale.

Plus profondément dans le routeur lui-même, le trafic arrive dans un VRF interne sur l'interface de processeur de routage eth2 :

```
[cedge6:/]$ chvrf utd  ifconfig
eth0      Link encap:Ethernet  HWaddr 54:0e:00:0b:0c:02
          inet6 addr: fe80::560e:ff:fe0b:c02/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:137510 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1366614 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:96520127 (92.0 MiB)  TX bytes:96510792 (92.0 MiB)

eth1      Link encap:Ethernet  HWaddr 00:1e:e6:61:6d:ba
          inet addr:192.168.1.2  Bcast:192.168.1.3  Mask:255.255.255.252
          inet6 addr: fe80::21e:e6ff:fe61:6dba/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:2000  Metric:1
          RX packets:1069 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2001 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:235093 (229.5 KiB)  TX bytes:193413 (188.8 KiB)

eth2      Link encap:Ethernet  HWaddr 00:1e:e6:61:6d:b9
          inet addr:192.0.2.2  Bcast:192.0.2.3  Mask:255.255.255.252
          inet6 addr: fe80::21e:e6ff:fe61:6db9/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:2000  Metric:1
          RX packets:2564233 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2564203 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:210051658 (200.3 MiB)  TX bytes:301467970 (287.5 MiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
```

Eth0 est une interface TIPC (Transport Inter Process Communication) connectée au processus IOSd. Le canal OneP passe par-dessus pour transmettre les configurations et les notifications entre le conteneur IOSd et UTD.

De ce que vous êtes concerné, "eth2 [interface de conteneur]" est ponté à "VPG1 [192.0.2.1/192.168.2.2]" sont les adresses poussées par vManage à l'IOS-XE et au conteneur.

Si vous exécutez tcpdump, vous pouvez voir le trafic encapsulé GRE allant au conteneur. L'encapsulation GRE inclut un en-tête VPATH.

```
[cedge6:/]$  chvrf utd  tcpdump -nNvvvXi eth2    not udp
tcpdump: listening on eth2, link-type EN10MB (Ethernet), capture size 262144 bytes
06:46:56.350725 IP (tos 0x0, ttl 255, id 35903, offset 0, flags [none], proto GRE (47), length 121)
    192.0.2.1 > 192.0.2.2: GREv0, Flags [none], length 101
```

```
    gre-proto-0x8921
    0x0000:  4500 0079 8c3f 0000 ff2f ab12 c000 0201  E..y.?.../......
    0x0010:  c000 0202 0000 8921 4089 2102 0000 0000  .......!@.!.....
    0x0020:  0000 0000 0300 0001 0000 0000 0000 0000  ................
    0x0030:  0004 0800 e103 0004 0008 0000 0001 0000  ................
    0x0040:  4500 0039 2542 4000 4011 ce40 c0a8 10fe  E..9%B@.@..@....
    0x0050:  ad26 c864 8781 0035 0025 fe81 cfa8 0100  .&.d...5.%......
    0x0060:  0001 0000 0000 0000 0377 7777 0363 6e6e  .........www.cnn
    0x0070:  0363 6f6d 0000 0100 01                   .com.....
```

Paquet entrant du conteneur vers le côté LAN ou WAN

Après le traitement Snort (en supposant que le trafic ne doit pas être abandonné), il est réinjecté dans le chemin de transfert QFP.

```
cedge6#show platform packet-trace packet 15
Packet: 15          CBUG ID: 3849210
Summary
  Input     : Tunnel6000001
  Output    : GigabitEthernet3
  State     : FWD
```

Tunnel600001 est l'interface de sortie du conteneur.

```
  Feature: OUTPUT_UTD_FIRST_INSPECT_EXT
    Entry       : Output - 0x817cc5b8
    Input       : GigabitEthernet2
    Output      : GigabitEthernet3
    Lapsed time : 2680 ns
  Feature: UTD Inspection
    Action         : Reinject
    Input interface : GigabitEthernet2
    Egress interface: GigabitEthernet3
  Feature: OUTPUT_UTD_FINAL_INSPECT_EXT
    Entry       : Output - 0x817cc5e8
    Input       : GigabitEthernet2
    Output      : GigabitEthernet3
    Lapsed time : 12933 ns
```

Comme le trafic a déjà été inspecté, le routeur sait qu'il s'agit d'une réinjection.

```
  Feature: NAT
    Direction   : IN to OUT
    Action      : Translate Source
    Steps       :
    Match id    : 1
```

```
    Old Address : 192.168.16.254 35568
    New Address : 172.16.16.254  05062
```

Le trafic reçoit la fonction NAT et se dirige vers Internet.

```
  Feature: MARMOT_SPA_D_TRANSMIT_PKT
    Entry       : Output - 0x8177c838
    Input       : GigabitEthernet2
    Output      : GigabitEthernet3
    Lapsed time : 91733 ns
```

Intégration de la journalisation de flux UTD avec Packet-trace

IOS-XE 17.5.1 a ajouté l'intégration de la journalisation de flux UTD avec packet-trace, où la sortie path-trace inclura un verdict UTD. Verdict peut être l'un des éléments suivants, par exemple :

- Le paquet que UTD décide de bloquer/d'alerter pour Snort
- autoriser/abandonner pour URLF
- bloquer/autoriser AMP

Pour les paquets qui n'ont pas les informations de verdict UTD, aucune information de journalisation de flux n'est consignée. Notez également qu'il n'y a pas de journalisation du verdict de réussite/d'autorisation IPS/IDS en raison d'un impact négatif potentiel sur les performances.

Pour activer l'intégration de la journalisation des flux, utilisez le modèle de module complémentaire CLI avec :

```
utd engine standard multi-tenancy
utd global
  flow-logging all
```

Exemple de résultat pour différents verdicts :

Délai de recherche URL :

```
show platform packet-trace  pack all | sec Packet: | Feature: UTD Inspection
Packet: 31          CBUG ID: 12640
Feature: UTD Inspection
   Action                   : Reinject
   Input interface          : GigabitEthernet2
   Egress interface         : GigabitEthernet3
   Flow-Logging Information :
    URLF Policy ID          : 1
    URLF Action             : Allow(1)
```

```
    URLF Reason             : URL Lookup Timeout(8)
```

Reputation et verdict URLF Autoriser :

```
Packet: 21          CBUG ID: 13859
  Feature: UTD Inspection
    Action                  : Reinject
    Input interface         : GigabitEthernet3
    Egress interface        : GigabitEthernet2
    Flow-Logging Information :
     URLF Policy ID         : 1
     URLF Action            : Allow(1)
     URLF Reason            : No Policy Match(4)
     URLF Category          : News and Media(63)
     URLF Reputation        : 81
```

Reputation et verdict URLF Block :

```
Packet: 26          CBUG ID: 15107
  Feature: UTD Inspection
    Action                  : Reinject
    Input interface         : GigabitEthernet3
    Egress interface        : GigabitEthernet2
    Flow-Logging Information :
     URLF Policy ID         : 1
     URLF Action            : Block(2)
     URLF Reason            : Category/Reputation(3)
     URLF Category          : Social Network(14)
     URLF Reputation        : 81
```

Prérequis :

Vérification de la compatibilité de la version UTD avec IOS XE

```
cedge7#sh utd eng sta ver
UTD Virtual-service Name: utd
IOS-XE Recommended UTD Version: 1.10.33_SV2.9.16.1_XEmain
IOS-XE Supported UTD Regex: ^1\.10\.([0-9]+)_SV(.*)_XEmain$
UTD Installed Version: 1.0.2_SV2.9.16.1_XE17.5 (UNSUPPORTED)
```

Si « NON PRIS EN CHARGE » s'affiche, la mise à niveau du conteneur est requise comme première étape avant de

Vérifier la configuration valide du serveur de noms dans le conteneur

Certains services de sécurité, tels qu'AMP et URLF, nécessitent que le conteneur UTD puisse résoudre les noms des

```
cedge:/harddisk/virtual-instance/utd/rootfs/etc]$ more resolv.conf
nameserver 208.67.222.222
nameserver 208.67.220.220
nameserver 8.8.8.8
```

Problème 1

Par conception, Unified Thread Defense doit être configuré avec l'exemple d'utilisation Direct Internet Access (DIA)

Dépannage

Consultez toujours le fichier journal du conteneur.

```
cedge6#app-hosting move appid utd log to bootflash:
Successfully moved tracelog to bootflash:
iox_utd_R0-0_R0-0.18629_0.20190501005829.bin.gz
```

Copie le fichier journal sur la mémoire flash elle-même.

La commande suivante permet d'afficher le journal :

```
cedge6#  more /compressed iox_utd_R0-0_R0-0.18629_0.20190501005829.bin.gz
```

L'affichage du journal révèle :

```
2019-04-29 16:12:12 ERROR: Cannot resolve host api.bcti.brightcloud.com: Temporary failure in name reso
2019-04-29 16:17:52 ERROR: Cannot resolve host api.bcti.brightcloud.com: Temporary failure in name reso
2019-04-29 16:23:32 ERROR: Cannot resolve host api.bcti.brightcloud.com: Temporary failure in name reso
2019-04-29 16:29:12 ERROR: Cannot resolve host api.bcti.brightcloud.com: Temporary failure in name reso
2019-04-29 16:34:52 ERROR: Cannot resolve host api.bcti.brightcloud.com: Temporary failure in name reso
2019-04-29 16:40:27 ERROR: Cannot resolve host api.bcti.brightcloud.com: Temporary failure in name reso
```

Par défaut, vManage provisionne un conteneur qui utilise le serveur OpenDNS [208.67.222.222 et 208.67.220.220]

Cause première

Le trafic DNS (Domain Name System) pour résoudre **api.bcti.brightcloud.com** est abandonné quelque part dans le

Problème 2

Dans un scénario où les sites Web de la catégorie Ordinateur et Infos Internet sont censés être bloqués, la requête HT

Dépannage

Comme expliqué précédemment, le trafic est acheminé vers le conteneur. Lorsque ce flux est encapsulé dans l'en-têt

Dans ce scénario, l'adresse IP du client est 192.168,16.254. Dépannons la gestion des reniflements par le conteneur l

```
debug platform condition ipv4 192.168.16.254/32 both
debug platform condition feature utd controlplane submode serviceplane-web-filtering level verbose
debug platform condition start
```

Cet ensemble de commandes indique à IOS-XE de marquer le trafic en provenance ou à destination de 192.168.16.2

LSMPI punt header	Outer IP header (e.g. 192.0.2.x)	GRE header	vPath header (conditional debug flag is here)	Inner (original) IP packet

Snort débogue uniquement ce flux spécifique tandis que d'autres sont traités normalement.

À ce stade, vous pouvez demander à l'utilisateur de déclencher le trafic du client vers www.cisco.com.

L'étape suivante consiste à récupérer les journaux :

```
app-hosting move appid utd log to bootflash:
```

Dans le cas du trafic HTTP, le préprocesseur Snort HTTP détecte l'URL dans la requête get.

```
2019-04-26 13:04:27.773:(#1):SPP-URL-FILTERING UTM preprocessor p->src_port = 39540, p->dst_port = 80
2019-04-26 13:04:27.793:(#1):SPP-URL-FILTERING UTM preprocessor p->src_port = 80, p->dst_port = 39540
2019-04-26 13:04:27.794:(#1):SPP-URL-FILTERING UTM preprocessor p->src_port = 39540, p->dst_port = 80
2019-04-26 13:04:27.794:(#1):SPP-URL-FILTERING UTM preprocessor p->src_port = 39540, p->dst_port = 80
2019-04-26 13:04:27.794:(#1):SPP-URL-FILTERING got utmdata_p
2019-04-26 13:04:27.794:(#1):SPP-URL-FILTERING HTTP Callback, direction = 00000080

2019-04-26 13:04:27.795:(#1):SPP-URL-FILTERING White list regex match not enabled
2019-04-26 13:04:27.795:(#1):SPP-URL-FILTERING Black list regex match not enabled
2019-04-26 13:04:27.795:(#1):SPP-URL-FILTERING URL database Request: url_len = 12, msg overhead 12 url:
2019-04-26 13:04:27.795:(#1):SPP-URL-FILTERING Send to URL database: req_id=0x10480047
2019-04-26 13:04:27.795:(#1):SPP-URL-FILTERING Sent to URL database 24 bytes
2019-04-26 13:04:27.795:(#1):SPP-URL-FILTERING Send to URL database done, idx: 71, URL: www.cisco.com/
2019-04-26 13:04:27.795:(#1):SPP-URL-FILTERING Received from URL database 24 bytes
2019-04-26 13:04:27.816:(#1):SPP-URL-FILTERING UTM preprocessor p->src_port = 80, p->dst_port = 39540
2019-04-26 13:04:27.816:(#1):SPP-URL-FILTERING Found UTMData at 0x007f8d9ee80878, action = 0000000a
2019-04-26 13:04:27.816:(#1):SPP-URL-FILTERING Utm_verdictProcess: vrf_id 1, category 0x63, score 81 <<
2019-04-26 13:04:27.816:(#1):SPP-URL-FILTERING Category 0x3f  <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
2019-04-26 13:04:27.816:(#1):SPP-URL-FILTERING index = 63, action = 1  <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
2019-04-26 13:04:27.816:(#1):SPP-URL-FILTERING Blocking category = 0x3f  <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
```

En cas de trafic https, le DNS de destination a été extrait du serveur hello par le préprocesseur HTTPS

```
2019-05-01 00:56:18.870:(#1):SPP-URL-FILTERING UTM preprocessor p->src_port = 35322, p->dst_port = 443
2019-05-01 00:56:18.886:(#1):SPP-URL-FILTERING UTM preprocessor p->src_port = 443, p->dst_port = 35322
2019-05-01 00:56:18.887:(#1):SPP-URL-FILTERING UTM preprocessor p->src_port = 35322, p->dst_port = 443
2019-05-01 00:56:18.887:(#1):SPP-URL-FILTERING UTM preprocessor p->src_port = 35322, p->dst_port = 443
2019-05-01 00:56:18.903:(#1):SPP-URL-FILTERING UTM preprocessor p->src_port = 443, p->dst_port = 35322
2019-05-01 00:56:18.906:(#1):SPP-URL-FILTERING UTM preprocessor p->src_port = 443, p->dst_port = 35322
2019-05-01 00:56:18.906:(#1):SPP-URL-FILTERING UTM preprocessor p->src_port = 35322, p->dst_port = 443
2019-05-01 00:56:18.907:(#1):SPP-URL-FILTERING UTM preprocessor p->src_port = 443, p->dst_port = 35322
2019-05-01 00:56:18.907:(#1):SPP-URL-FILTERING UTM preprocessor p->src_port = 443, p->dst_port = 35322
2019-05-01 00:56:18.907:(#1):SPP-URL-FILTERING UTM preprocessor p->src_port = 443, p->dst_port = 35322

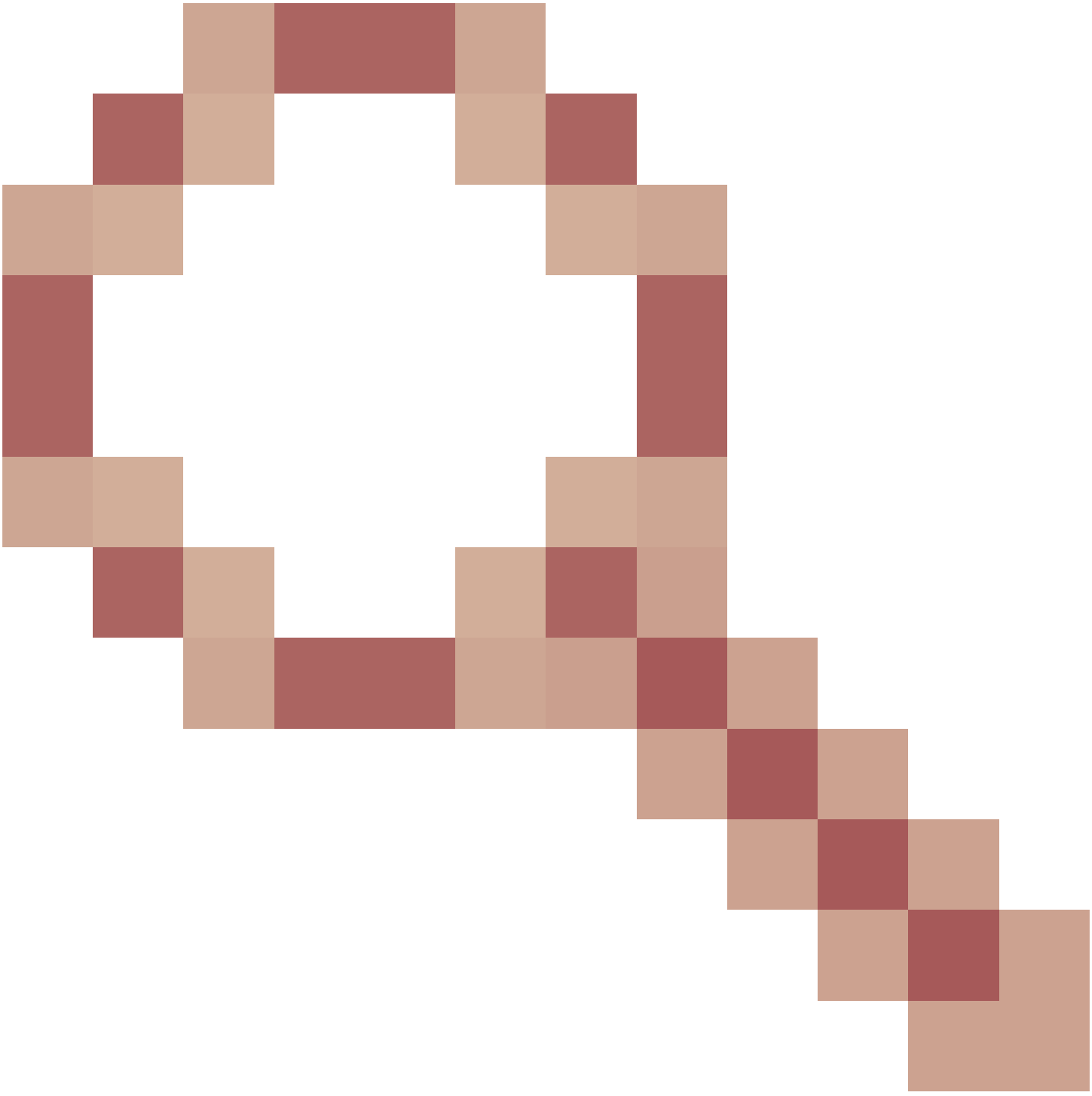
2019-05-01 00:56:18.908:(#1):SPP-URL-FILTERING UTM preprocessor p->src_port = 443, p->dst_port = 35322
2019-05-01 00:56:18.908:(#1):SPP-URL-FILTERING utm_sslLookupCallback
2019-05-01 00:56:18.908:(#1):SPP-URL-FILTERING got utmdata_p
2019-05-01 00:56:18.909:(#1):SPP-URL-FILTERING White list regex match not enabled
2019-05-01 00:56:18.909:(#1):SPP-URL-FILTERING Black list regex match not enabled
2019-05-01 00:56:18.909:(#1):SPP-URL-FILTERING URL database Request: url_len = 11, msg overhead 12 url:
2019-05-01 00:56:18.909:(#1):SPP-URL-FILTERING Send to URL database: req_id=0x10130012
2019-05-01 00:56:18.909:(#1):SPP-URL-FILTERING Sent to URL database 23 bytes
2019-05-01 00:56:18.909:(#1):SPP-URL-FILTERING Send to URL database done, idx: 18, URL: www.cisco.com
2019-05-01 00:56:18.909:(#1):SPP-URL-FILTERING UTM preprocessor p->src_port = 443, p->dst_port = 35322
2019-05-01 00:56:18.910:(#1):SPP-URL-FILTERING Found UTMData at 0x007f1d9c479640, action = 00000008
2019-05-01 00:56:18.910:(#1):SPP-URL-FILTERING Verdict very late, in queryig state 2, idx=18
2019-05-01 00:56:18.910:(#1):SPP-URL-FILTERING UTM preprocessor p->src_port = 443, p->dst_port = 35322

2019-05-01 00:56:18.910:(#1):SPP-URL-FILTERING Found UTMData at 0x007f1d9c479640, action = 00000009
2019-05-01 00:56:18.910:(#1):SPP-URL-FILTERING Verdict very late, in queryig state 2, idx=18 <<<<<<<<<<
2019-05-01 00:56:18.909:(#1):SPP-URL-FILTERING Received from URL database 24 bytes
```

Ici, vous ne voyez pas la page de blocage en cours de déclenchement puisque le logiciel ne rapporte pas les résultats

Cause première

CSCvo77664



"Le filt

Problème 3

Dans ce scénario, par intermittence, les sessions de navigation Web qui devraient être autorisées par le filtrage des U

Dépannage

Étape 1 : collecte de statistiques générales

Remarque Cette sortie de commande est réinitialisée toutes les 5 minutes

<#root>

cedge7#

show utd engine standard statistics internal

```
*************Engine #1*************
<removed>
================================================================================
HTTP Inspect - encodings (Note: stream-reassembled packets included): <<<<<<<<<< generic layer7 HTTP st
    POST methods:                          0
    GET methods:                           7
    HTTP Request Headers extracted:        7
    HTTP Request Cookies extracted:        0
    Post parameters extracted:             0
    HTTP response Headers extracted:       6
    HTTP Response Cookies extracted:       0
    Unicode:                               0
    Double unicode:                        0
    Non-ASCII representable:               0
    Directory traversals:                  0
    Extra slashes ("//"):                  0
    Self-referencing paths ("./"):         0
    HTTP Response Gzip packets extracted: 0
    Gzip Compressed Data Processed:        n/a
    Gzip Decompressed Data Processed:      n/a
    Http/2 Rebuilt Packets:                0
    Total packets processed:               13
<removed>
================================================================================
SSL Preprocessor:    <<<<<<<<<< generic layer7 SSL statistics
   SSL packets decoded: 38
          Client Hello: 8
          Server Hello: 8
           Certificate: 2
           Server Done: 6
   Client Key Exchange: 2
   Server Key Exchange: 2
         Change Cipher: 10
              Finished: 0
    Client Application: 2
    Server Application: 11
                 Alert: 0
  Unrecognized records: 11
  Completed handshakes: 0
        Bad handshakes: 0
      Sessions ignored: 4
    Detection disabled: 1

<removed>
UTM Preprocessor Statistics < URL filtering statistics including
--------------------------
URL Filter Requests Sent:             11
URL Filter Response Received:         5
Blacklist Hit Count:                  0
```

```
Whitelist Hit Count:                  0

Reputation Lookup Count:              5
Reputation Action Block:              0
Reputation Action Pass:               5
Reputation Action Default Pass:       0
Reputation Action Default Block:      0
Reputation Score None:                0
Reputation Score Out of Range:        0

Category Lookup Count:                5
Category Action Block:                0
Category Action Pass:                 5
Category Action Default Pass:         0
Category None:                        0

UTM Preprocessor Internal Statistics
------------------------------------
Total Packets Received:               193
SSL Packet Count:                     4
Action Drop Flow:                     0
Action Reset Session:                 0
Action Block:                         0
Action Pass:                          85
Action Offload Session:               0
Invalid Action:                       0
No UTM Tenant Persona:                0
No UTM Tenant Config:                 0

URL Lookup Response Late:             4  <<<<< Explanation below
URL Lookup Response Very Late:        64 <<<<< Explanation below
URL Lookup Response Extremely Late:   2  <<<<< Explanation below
Response Does Not Match Session:      2  <<<<< Explanation below
No Response When Freeing Session:     1
First Packet Not From Initiator:      0
Fail Open Count:                      0
Fail Close Count :                    0

UTM Preprocessor Internal Global Statistics
-------------------------------------------
Domain Filter Whitelist Count:        0
utmdata Used Count:                   11
utmdata Free Count:                   11
utmdata Unavailable:                  0
URL Filter Response Error:            0
No UTM Tenant Map:                    0
No URL Filter Configuration :         0
Packet NULL Error :                   0

URL Database Internal Statistics
--------------------------------
URL Database Not Ready:               0
Query Successful:                     11
Query Successful from Cloud:          6  <<< 11 queries were succesful but 6 only are queried via brigh
Query Returned No Data:               0  <<<<<< errors
Query Bad Argument:                   0  <<<<<< errors
Query Network Error:                  0  <<<<<< errors
URL Database UTM disconnected:        0
URL Database request failed:          0
URL Database reconnect failed:        0
URL Database request blocked:         0
URL Database control msg response:    0
```

```
URL Database Error Response:          0
================================================================================
Files processed: none
================================================================================
```

- "late request" - représente le certificat HTTP GET ou HTTPS client/serveur [où SNI / DN peut être extrait pou
- « demandes très tardives » : indique une sorte de compteur d'abandon de session dans lequel d'autres paquets d
- « demandes extrêmement tardives » - lorsque la requête de session vers Brightcloud a été réinitialisée sans TCP ou UTF

Dans ce scénario, nous voyons des compteurs mondiaux qui mettent en évidence une situation malsaine.

Étape 2 : Examinez le fichier journal de l'application

Le logiciel Unified Thread Detection va enregistrer les événements dans le fichier journal de l'application.

```
cedge6#app-hosting move appid utd log to bootflash:
Successfully moved tracelog to bootflash:
iox_utd_R0-0_R0-0.18629_0.20190501005829.bin.gz
```

Cette opération extrait le fichier journal de l'application conteneur et l'enregistre sur la mémoire flash elle-même.

La commande suivante permet d'afficher le journal :

```
cedge6#  more /compressed iox_utd_R0-0_R0-0.18629_0.20190501005829.bin.gz
```

Remarque : dans le logiciel IOS-XE version 20.6.1 et ultérieure, il n'est plus nécessaire de déplacer ma

L'affichage du journal révèle :

```
.....
2020-04-14 17:47:57.504:(#1):SPP-URL-FILTERING txn_id miss match verdict txn_id 245 , utmdata txn_id  0
2020-04-14 17:47:57.504:(#1):SPP-URL-FILTERING txn_id miss match verdict txn_id 248 , utmdata txn_id  0
2020-04-14 17:47:57.504:(#1):SPP-URL-FILTERING txn_id miss match verdict txn_id 249 , utmdata txn_id  0
2020-04-14 17:47:57.504:(#1):SPP-URL-FILTERING txn_id miss match verdict txn_id 250 , utmdata txn_id  0
2020-04-14 17:47:57.660:(#1):SPP-URL-FILTERING txn_id miss match verdict txn_id 251 , utmdata txn_id  0
2020-04-14 17:47:57.660:(#1):SPP-URL-FILTERING txn_id miss match verdict txn_id 254 , utmdata txn_id  0
2020-04-14 17:47:57.660:(#1):SPP-URL-FILTERING txn_id miss match verdict txn_id 255 , utmdata txn_id  0
```

```
2020-04-14 17:48:05.725:(#1):SPP-URL-FILTERING txn_id miss match verdict txn_id 192 , utmdata txn_id  0
2020-04-14 17:48:37.629:(#1):SPP-URL-FILTERING txn_id miss match verdict txn_id 208 , utmdata txn_id  0
2020-04-14 17:49:55.421:(#1):SPP-URL-FILTERING txn_id miss match verdict txn_id 211 , utmdata txn_id  0
2020-04-14 17:51:40 ERROR: Cannot send to host api.bcti.brightcloud.com: Connection timed out
2020-04-14 17:52:21 ERROR: Cannot send to host api.bcti.brightcloud.com: Connection timed out
2020-04-14 17:52:21 ERROR: Cannot send to host api.bcti.brightcloud.com: Connection timed out
2020-04-14 17:53:56 ERROR: Cannot send to host api.bcti.brightcloud.com: Connection timed out
2020-04-14 17:54:28 ERROR: Cannot send to host api.bcti.brightcloud.com: Connection timed out
2020-04-14 17:54:29 ERROR: Cannot send to host api.bcti.brightcloud.com: Connection timed out
2020-04-14 17:54:37 ERROR: Cannot send to host api.bcti.brightcloud.com: Connection timed out
....
```

- "ERREUR : envoi impossible à l'hôte api.bcti.brightcloud.com" - signifie que la session d'interrogation à Brigl
  Afin de démontrer le problème, l'utilisation d'EPC [Embedded Packet Capture] permettrait de visualiser le pro

- "SPP-URL-FILTERING txn_id miss match verdict" - Cette condition d'erreur nécessite un peu plus d'explicati

Problème 4

Dans ce scénario, IPS est la seule fonctionnalité de sécurité activée dans UTD et le client rencontre des problèmes de

Dépannage

Pour résoudre ce problème de chemin de données, commencez par capturer les paquets à partir de l'hôte TCP qui re

```
edge#show platform packet-trace summ
Pkt   Input                    Output                   State  Reason
0     Gi0/0/1                  internal0/0/svc_eng:0    PUNT   64  (Service Engine packet)
1     Tu2000000001             Gi0/0/2                  FWD
2     Gi0/0/2                  internal0/0/svc_eng:0    PUNT   64  (Service Engine packet)
3     Tu2000000001             Gi0/0/1                  FWD
4     Gi0/0/1                  internal0/0/svc_eng:0    PUNT   64  (Service Engine packet)
5     Tu2000000001             Gi0/0/2                  FWD
6     Gi0/0/1                  internal0/0/svc_eng:0    PUNT   64  (Service Engine packet)
7     Tu2000000001             Gi0/0/2                  FWD
8     Gi0/0/2                  internal0/0/svc_eng:0    PUNT   64  (Service Engine packet)
9     Gi0/0/2                  internal0/0/svc_eng:0    PUNT   64  (Service Engine packet)
```

Le résultat ci-dessus indique que les paquets numéros 8 et 9 ont été déviés vers le moteur UTD, mais qu'ils n'ont pas

```
edge#show utd engine standard statistics internal
<snip>
Normalizer drops:
           OUTSIDE_PAWS: 0
             AHEAD_PAWS: 0
           NO_TIMESTAMP: 4
                BAD_RST: 0
```

```
      REPEAT_SYN: 0
     WIN_TOO_BIG: 0
        WIN_SHUT: 0
         BAD_ACK: 0
      DATA_CLOSE: 0
  DATA_NO_FLAGS: 0
     FIN_BEYOND: 0
```

Cause première

La cause principale du problème est due à un mauvais comportement de la pile TCP sur les imprimantes. Lorsque l'o

Références

- Guide de configuration de la sécurité : Unified Threat Defense

À propos de cette traduction
Cisco a traduit ce document en traduction automatisée vérifiée par une personne dans le cadre d'un service mondial permettant à nos utilisateurs d'obtenir le contenu d'assistance dans leur propre langue.

Il convient cependant de noter que même la meilleure traduction automatisée ne sera pas aussi précise que celle fournie par un traducteur professionnel.

À propos de cette traduction
Cisco a traduit ce document en traduction automatisée vérifiée par une personne dans le cadre d'un service mondial permettant à nos utilisateurs d'obtenir le contenu d'assistance dans leur propre langue.

Il convient cependant de noter que même la meilleure traduction automatisée ne sera pas aussi précise que celle fournie par un traducteur professionnel.