

# Échec de l'établissement des connexions TCP lorsque le trafic suit des chemins asymétriques

## Contenu

[Introduction](#)

[Problème](#)

[Schéma de topologie](#)

[Diagnostic](#)

[Solution](#)

[Conclusion](#)

## Introduction

Ce document décrit le problème qui survient lorsque des chemins asymétriques sont utilisés pour le transfert de trafic dans le fabric SD-WAN.

## Problème

Les connexions Secure Shell (SSH) ne peuvent pas être établies sur host2 (hostname - edgeclient2) à partir de host1 (hostname - edgeclient1), mais en même temps SSH fonctionne correctement dans le sens inverse.

```
[root@edgeclient2 user]# ssh user@192.168.40.21
user@192.168.40.21's password:
Last login: Sun Feb 10 13:26:32 2019 from 192.168.60.20
[user@edgeclient1 ~]$
```

```
[root@edgeclient1 user]# ssh user@192.168.60.20
<nothing happens after that>
```

ou

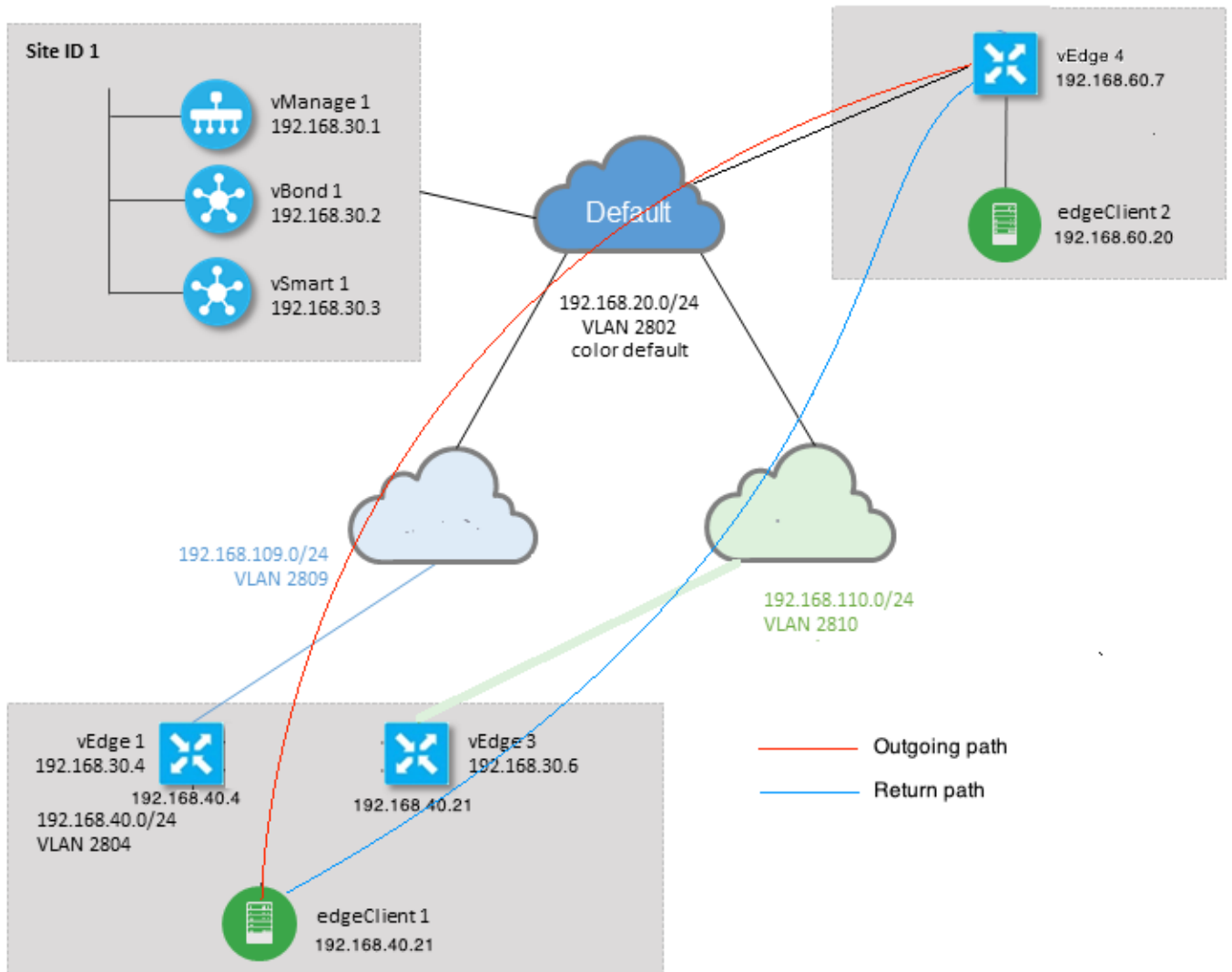
```
[user@edgeclient1 ~]$ ssh user@192.168.60.20
ssh_exchange_identification: Connection closed by remote host
```

Les démons et clients SSH edgeclient1 et edgeclient2 ont connu des configurations et des connexions correctes qui peuvent être établies à partir du segment LAN local avec succès :

```
vedge4# request execute vpn 40 ssh user@192.168.60.20
user@192.168.60.20's password:
Last login: Sun Feb 10 13:28:23 2019 from 192.168.60.7
[user@edgeclient2 ~]$
```

Toutes les autres applications TCP (Transmission Control Protocol) ont des problèmes similaires.

## Schéma de topologie



## Diagnostic

Ces listes de contrôle d'accès (ACL) ont été configurées et appliquées dans les directions correspondantes sur les interfaces côté service de vEdge1 et vEdge3 :

```
policy
access-list SSH_IN
sequence 10
match
source-ip      192.168.40.21/32
destination-ip 192.168.60.20/32
!
action accept
count SSH_IN
!
!
default-action accept
!
access-list SSH_OUT
sequence 10
match
```

```

source-ip      192.168.60.20/32
destination-ip 192.168.40.21/32
!
action accept
count SSH_OUT
!
!
default-action accept
!
!
```

La liste de contrôle d'accès en miroir a été appliquée sur vEdge4 :

```

policy
access-list SSH_IN
sequence 10
match
source-ip      192.168.60.20/32
destination-ip 192.168.40.21/32
!
action accept
count SSH_IN
!
!
default-action accept
!
access-list SSH_OUT
sequence 10
match
source-ip      192.168.40.21/32
destination-ip 192.168.60.20/32
!
action accept
count SSH_OUT
!
!
default-action accept
!
!
```

La visibilité sur les applications a également été activée sur tous les routeurs vEdge et les flux ont été vérifiés pendant la phase d'établissement de la connexion SSH :

```
vedgel# show app cflowd flows | tab ; show policy access-list-counters
```

TIME	EGRESS		INGRESS	SRC	DEST	TCP				TOTAL		
	MIN	MAX				IP	CNTRL	ICMP	TOTAL			
TOTAL	SRC	IP	DEST	PORT	PORT	DSCP	PROTO	BITS	OPCODE	NHOP	IP	PKTS
BYTES	LEN	LEN	START	TIME	EXP	NAME	NAME	NAME	NAME	NAME	NAME	NAME
40	192.168.40.21	192.168.60.20	47866	22	0	6	24	0		192.168.109.7	3	
227	66	87	Sun Feb 17 14:13:25 2019	34		ge0/0	ge0/1					

```

COUNTER
NAME      NAME      PACKETS  BYTES
```

```
-----
SSH_IN   SSH_IN   3         227
SSH_OUT  SSH_OUT  2         140
```

```
vedge3# show app cflowd flows | tab ; show policy access-list-counters
```

```

                                     TCP
TIME      EGRESS  INGRESS
TOTAL    MIN    MAX
VPN SRC IP          DEST IP          SRC  DEST      IP      CNTRL  ICMP
BYTES  LEN  LEN  START TIME     PORT  PORT  DSCP  PROTO  BITS  OPCODE  NHOP IP          PKTS
-----
40      192.168.60.20  192.168.40.21  22    47866  0     6      18    0      192.168.40.21  8
480     60   60   Sun Feb 17 14:14:08 2019  51     ge0/1  ge0/0
```

```

COUNTER
NAME     NAME     PACKETS  BYTES
-----
SSH_IN   SSH_IN   0         0
SSH_OUT  SSH_OUT  7         420
```

```
vedge4# show app cflowd flows | tab ; show policy access-list-counters
```

```

                                     TCP
TIME      EGRESS  INGRESS
TOTAL    TOTAL  MIN    MAX
VPN SRC IP          DEST IP          SRC  DEST      IP      CNTRL  ICMP
BYTES  LEN  LEN  START TIME     PORT  PORT  DSCP  PROTO  BITS  OPCODE  NHOP IP          PKTS
-----
40      192.168.40.21  192.168.60.20  47866  22    0     6      2     0      192.168.60.20  4
240     60   60   Sun Feb 17 14:17:44 2019  37     ge0/2  ge0/0
40      192.168.60.20  192.168.40.21  22    47866  0     6      18    0      192.168.110.6  8
592     74   74   Sun Feb 17 14:17:44 2019  49     ge0/0  ge0/2
```

```

COUNTER
NAME     NAME     PACKETS  BYTES
-----
SSH_IN   SSH_IN   8         592
SSH_OUT  SSH_OUT  4         240
```

Comme vous pouvez le voir à partir de ces sorties, les flux entrants et sortants sont asymétriques. edgeclient1 (192.168.40.21) tente d'établir une session SSH avec edgeclient2 (192.168.60.20) et le trafic entrant passe par vEdge1 et retourne le trafic via vEdge3. À partir des compteurs ACL, vous pouvez également voir que le nombre de paquets entrants et sortants sur vEdge4 ne correspond pas à la somme dans les directions correspondantes sur vEdge1 et vEdge3. En même temps, il n'y a aucune perte de paquet lors du test avec ping :

```
[root@edgeclient1 user]# ping -f 192.168.60.20 -c 10000
PING 192.168.60.20 (192.168.60.20) 56(84) bytes of data.
```

```
--- 192.168.60.20 ping statistics ---
10000 packets transmitted, 10000 received, 0% packet loss, time 3076ms
rtt min/avg/max/mdev = 0.128/0.291/6.607/0.623 ms, ipg/ewma 0.307/0.170 ms
```

```
[root@edgeclient2 user]# ping -f 192.168.40.21 -c 10000
PING 192.168.40.21 (192.168.40.21) 56(84) bytes of data.
```

```
--- 192.168.40.21 ping statistics ---
10000 packets transmitted, 10000 received, 0% packet loss, time 3402ms
rtt min/avg/max/mdev = 0.212/0.318/2.766/0.136 ms, ipg/ewma 0.340/0.327 ms
```

En outre, il est possible de copier des fichiers sur scp/sftp sans problème.

## Solution

Certains paramètres de configuration ou de stratégie de données Deep Packet Inspection (DPI) ont été suspectés au départ, mais aucun d'entre eux n'a été activé :

```
vedge3# show policy from-vsmart
% No entries found.
```

```
vedge1# show policy from-vsmart
% No entries found.
```

Mais finalement, il a été découvert que l'optimisation TCP était activée :

```
vedge1# show app tcp-opt active-flows
```

RX	UNOPT	PROXY	SRC	DEST	EGRESS	INGRESS	TX	
VPN	SRC IP	DEST IP	PORT	PORT	START TIME	NAME	NAME	BYTES
BYTES	TCP STATE	REASON	IDENTITY					
40	192.168.40.21	192.168.60.20	47868	22	Sun Feb 17 14:18:13 2019	ge0_0	ge0_1	314
0	In-progress	-	Client-Proxy					

```
vedge1# show app tcp-opt expired-flows
```

TX	RX	UNOPT	PROXY	SRC	DEST	START TIME	END
TIMESTAMP	VPN	SRC IP	DEST IP	PORT	PORT	START TIME	END
TIME	BYTES	BYTES	TCP STATE	REASON	IDENTITY	DELETE REASON	
1549819969608	40	192.168.40.21	192.168.60.7	22	56612	Sun Feb 10 18:32:49 2019	Sun
Feb 10 18:36:03 2019		5649	4405	Optimized	-	Server-Proxy	CLOSED
1549820055487	40	192.168.40.21	192.168.60.7	22	56613	Sun Feb 10 18:34:15 2019	Sun
Feb 10 19:07:46 2019		5719	4669	Optimized	-	Server-Proxy	CLOSED
1550408210511	40	192.168.40.21	192.168.60.20	47862	22	Sun Feb 17 13:56:50 2019	Sun
Feb 17 13:56:58 2019		401	0	Optimized	-	Client-Proxy	STATE-TIMEOUT
1550408981634	40	192.168.40.21	192.168.60.20	47864	22	Sun Feb 17 14:09:41 2019	Sun
Feb 17 14:09:49 2019		401	0	Optimized	-	Client-Proxy	STATE-TIMEOUT
1550409205399	40	192.168.40.21	192.168.60.20	47866	22	Sun Feb 17 14:13:25 2019	Sun
Feb 17 14:13:33 2019		227	0	Optimized	-	Client-Proxy	STATE-TIMEOUT
1550409493042	40	192.168.40.21	192.168.60.20	47868	22	Sun Feb 17 14:18:13 2019	Sun
Feb 17 14:18:21 2019		401	0	Optimized	-	Client-Proxy	STATE-TIMEOUT

En outre, dans `debugs ftm tcpopt CONN_TEARDOWN` message peut être vu.

```

vedge1# show log /var/log/tmplog/vdebug tail "-f"
local7.debug: Feb 17 13:56:50 vedge1 FTMD[662]: ftm_tcptopt_flow_add[268]: Created new tcpflow :-
vrid-3 192.168.40.21/47862 192.168.60.20/22
local7.debug: Feb 17 13:56:58 vedge1 FTMD[662]: ftm_tcpd_send_conn_tear_down[388]: Trying to
pack and send the following message to TCPD
local7.debug: Feb 17 13:56:58 vedge1 FTMD[662]: ftm_tcpd_send_conn_tear_down[408]: Sending
following CONN_TD msg
local7.debug: Feb 17 13:56:58 vedge1 FTMD[662]: ftm_tcpd_send_conn_tear_down[413]:
192.168.40.21:47862->192.168.60.20:22; vpn:40; syn_seq_num:4172167164; identity:0; cport_prime:0
local7.debug: Feb 17 13:56:58 vedge1 FTMD[662]: ftm_tcpd_msgq_tx[354]: Transferring size = 66
bytes data
local7.debug: Feb 17 13:56:58 vedge1 FTMD[662]: ftm_tcpd_send_conn_tear_down[416]: Successfully
sent conn_td msg to TCPD
local7.debug: Feb 17 13:56:58 vedge1 FTMD[662]: ftm_tcptopt_propagate_tear_down[1038]: Sent
CONN_TEARDOWN msg to tcpd for existing tcpflow :- vrid-3 192.168.40.21/47862 192.168.60.20/22 ;
identity:CLIENT_SIDE_PROXY . Send Successful !
local7.debug: Feb 17 13:56:58 vedge1 FTMD[662]: ftm_tcptopt_append_expired_err_flow_tbl[958]:
Appending flow vrid-3 192.168.40.21/47862 192.168.60.20/22 to the expired flow table at Sun Feb
17 13:56:58 2019
local7.debug: Feb 17 13:56:58 vedge1 FTMD[662]: ftm_tcptopt_append_expired_err_flow_tbl[980]:
Appending flow vrid-3 192.168.40.21/47862 192.168.60.20/22 to the error flow table at Sun Feb
17 13:56:58 2019
local7.debug: Feb 17 13:56:58 vedge1 FTMD[662]: ftm_tcptopt_flow_delete[293]: Removing tcpflow :-
vrid-3 192.168.40.21/47862 192.168.60.20/22
local7.debug: Feb 17 13:56:58 vedge1 TCPD[670]: handle_upstream_connect[538]: Error - BP NULL
local7.debug: Feb 17 13:56:58 vedge1 FTMD[662]: ftm_tcpd_msg_decode[254]: FTM-TCPD: Received
FTM_TCPD__PB_FTM_TCPD_MSG__E_MSG_TYPE__CONN_CLOSED msg
local7.debug: Feb 17 13:56:58 vedge1 FTMD[662]: ftm_tcpd_handle_conn_closed[139]: FTM-TCPD:
Received CONN_CLOSED for following C->S
local7.debug: Feb 17 13:56:58 vedge1 FTMD[662]: ftm_tcpd_handle_conn_closed[150]:
192.168.40.21:47862->192.168.60.20:22; vpn:40; syn_seq_num:4172167164; identity:0;
cport_prime:47862; bind_port:0
local7.debug: Feb 17 13:56:58 vedge1 FTMD[662]: ftm_tcpd_handle_conn_closed[184]: FTM-TCPD:
Could not find entry in FT for following flow
local7.debug: Feb 17 13:56:58 vedge1 FTMD[662]: ftm_tcpd_handle_conn_closed[185]: vrid-3
192.168.40.21/47862 192.168.60.20/22

```

**Vous pouvez voir ici un exemple de fonctionnement correct de l'optimisation TCP (message CONN\_EST visible) :**

```

vedge3# show log /var/log/tmplog/vdebug tail "-f -n 0"
local7.debug: Feb 17 15:41:13 vedge3 FTMD[657]: ftm_tcpd_msg_decode[254]: FTM-TCPD: Received
FTM_TCPD__PB_FTM_TCPD_MSG__E_MSG_TYPE__CONN_CLOSED msg
local7.debug: Feb 17 15:41:13 vedge3 FTMD[657]: ftm_tcpd_handle_conn_closed[139]: FTM-TCPD:
Received CONN_CLOSED for following C->S
local7.debug: Feb 17 15:41:13 vedge3 FTMD[657]: ftm_tcpd_handle_conn_closed[150]:
192.168.40.21:47876->192.168.60.20:22; vpn:40; syn_seq_num:2779178897; identity:0;
cport_prime:47876; bind_port:0
local7.debug: Feb 17 15:41:15 vedge3 FTMD[657]: ftm_tcpd_msg_decode[258]: FTM-TCPD: Received
FTM_TCPD__PB_FTM_TCPD_MSG__E_MSG_TYPE__CONN_EST msg
local7.debug: Feb 17 15:41:15 vedge3 FTMD[657]: ftm_tcpd_handle_conn_est[202]: FTM-TCPD:
Received CONN_EST for following C->S
local7.debug: Feb 17 15:41:15 vedge3 FTMD[657]: ftm_tcpd_handle_conn_est[213]:
192.168.40.21:47878->192.168.60.20:22; vpn:40; syn_seq_num:2690847868; identity:0;
cport_prime:47878; bind_port:0
local7.debug: Feb 17 15:41:15 vedge3 FTMD[657]: ftm_tcptopt_flow_add[268]: Created new tcpflow :-
vrid-3 192.168.40.21/47878 192.168.60.20/22

```

## Conclusion

L'optimisation TCP nécessite que les flux soient symétriques, donc pour résoudre ce problème, soit l'optimisation TCP doit être désactivée (**pas d'optimisation vpn 40 tcp**), soit la **stratégie de données doit être créée pour forcer les flux TCP à emprunter le même chemin dans les deux directions**. Pour plus d'informations à ce sujet, reportez-vous à la section Traffic Symmetrix for DPI du [SD-WAN Design Guide](#), page 23.