

# Compteurs SNMP : Forum aux questions

## Contenu

[Introduction](#)

[Questions de compteurs SNMP](#)

[Compteurs SNMP et questions d'équivalent de commande show](#)

[Informations connexes](#)

## Introduction

Ce document apporte des réponses aux questions fréquemment posées, et guide les utilisateurs afin qu'ils trouvent des ressources utiles sur le SNMP, et les problèmes SNMP qui sont liés au matériel de Cisco.

## Questions de compteurs SNMP

### Q. Quel MIB est-ce que je devrais utiliser pour des compteurs d'interface ?

A. La Gestion d'interface au-dessus du SNMP est basée sur deux tables : [ifTable](#) (clients [enregistrés](#) seulement) et son extension, [ifXTable](#) (clients [enregistrés](#) seulement) décrit dans RFC1213/RFC2233. Les interfaces peuvent avoir plusieurs couches, qui dépend des medias, et chaque sous-couche est représentée par une ligne distincte dans la table. Les relations entre la couche et les couches inférieures plus élevées sont décrites dans l'[ifStackTable](#) (les clients [enregistrés](#) seulement). L'[ifTable](#) définit les compteurs de 32 bits pour les octets d'arrivée et sortants ([ifInOctets](#) (clients [enregistrés](#) seulement) /[ifOutOctets](#)), les paquets ([ifInUcastPkts](#) (clients [enregistrés](#) seulement) /[ifOutUcastPkts](#) (clients [enregistrés](#) seulement), [ifInNUcastPkts](#) (clients [enregistrés](#) seulement) /[ifOutNUcastPkts](#) (clients [enregistrés](#) seulement)), les erreurs, et les écarts. L'[ifXTable](#) fournit les compteurs 64-bit semblables, également appelés les compteurs de la grande capacité (HC) : [ifHCInOctets](#) (clients [enregistrés](#) seulement) /[ifHCOutOctets](#) (clients [enregistrés](#) seulement), et [ifHCInUcastPkts](#) (clients [enregistrés](#) seulement) /[ifHCOutUcastPkts](#) (clients [enregistrés](#) seulement).

### Q. Quand les compteurs 64-bit devraient-ils être utilisés ?

A. Les compteurs 64-bit développés adoptés [RFC 2233](#) pour la grande capacité relie dans des quels compteurs de 32 bits ne fournissent pas assez de capacité et enveloppent trop rapide.

À mesure que la vitesse des supports réseau augmente, le temps minimum l'où les contre-bouclages de 32 bits diminue. Par exemple, un flot de 10 Mbits/s des paquets dos à dos et normaux fait envelopper des [ifInOctets](#) dedans juste plus de 57 minutes. Aux 100 Mbits/s, le temps de bouclage minimum est de 5.7 minutes, et à GBP 1, le minimum est de 34 secondes.

**Remarque:** Le bouclage de compteurs SNMP, les compteurs de l'interface de ligne de commande (CLI) ne font pas.

Pour les interfaces qui fonctionnent à 20,000,000 bits par seconde ou à moins (de 20 millions), vous devez utiliser les compteurs de 32 bits d'octet et de paquet. Pour les interfaces qui actionnent plus rapide que 20 millions de bits par seconde, et plus lentement que 650,000,000 bits par seconde, vous devez utiliser les compteurs de 32 bits de paquet et les compteurs 64-bit d'octet. Pour les interfaces qui fonctionnent au bits/seconde 650,000,000 ou au paquet et aux compteurs plus rapides et 64-bit d'octet doit être utilisé.

Également, le logiciel de Cisco IOS® ne prend en charge pas les compteurs 64-bit pour des vitesses d'interface de moins de 20 Mbits/s. Ceci signifie que des compteurs 64-bit ne sont pas pris en charge sur 10 ports Ethernet de mi-bande, seulement 100 Fast Ethernet de mi-bande et d'autres ports à grande vitesse prennent en charge les compteurs 64-bit.

## Q. Quelle version de SNMP est exigée pour questionner les compteurs 64-bit ?

A. Le SNMPv2C ou le SNMPv3 est exigé pour questionner les compteurs 64-bit. SNMPv1 ne prend en charge pas les compteurs 64-bit. Rendez-vous compte que les ifInOctets = .1.3.6.1.2.1.2.2.1.10 est un compteur de 32 bits tandis que la version 64-bit est des ifHCInOctets = .1.3.6.1.2.1.31.1.1.1.6.

Exemple :

### Le Catalyst 5000 utilise le snmpget de HP OpenView, qui se transfère sur SNMPv1

```
# snmpget -c public 14.32.5.18 ifName.1
```

```
ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifName.1 DISPLAY STRING- (ascii) sc0
```

### Requête avec SNMPv1, par défaut pour le snmpget de HP OpenView

```
# snmpget -c public 14.32.5.18 ifHCInOctets.1
```

```
snmpget Agent reported error with variable #1.  
.iso.org.dod.internet.mgmt.mib-2.ifMIB.ifMIBObjects.ifXTable.ifXEntry.  
ifHCInOctets.1
```

```
SNMP Variable does not exist or access is denied.
```

### La même requête avec le SNMPv2C

```
# snmpget -v 2c -c public 14.32.5.18 ifHCInOctets.1
```

```
ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifHCInOctets.1 Counter64 622366215
```

## Q. Quels périphériques de Cisco implémentent les compteurs SNMP 64-bit, particulièrement pour l'IF-MIB ?

A. Ces compteurs SNMP 64-bit de mise en place de périphériques de Cisco :

**Remarque:** Vous devez être un utilisateur enregistré et vous devez être ouvert une session afin de visiter les liens d'ID de bogue Cisco et voir l'informations détaillées sur le bogue.

- Cisco 2500, 2600 Routeurs — Ces Routeurs ne prennent en charge pas les compteurs 64-bit.
- 2950 de Catalyst et 3550 — Le support commence dans le Logiciel Cisco IOS version

- 12.1(11)EA1 puisque l'ID de bogue Cisco [CSCdx67611](#) (clients [enregistrés](#) seulement) et l'ID de bogue Cisco [CSCdw52807](#) (clients [enregistrés](#) seulement).
- Catalyst 2900XL et 3500XL — Le support commence dans la version du logiciel Cisco IOS 12.0(5)WC3 depuis l'ID de bogue Cisco [CSCds45300](#) (clients [enregistrés](#) seulement).
  - Gamme Catalyst 5000 — Depuis la version du logiciel Cisco IOS 3.x. Sur RSM/RSFC, le support commence dans le Logiciel Cisco IOS version 12.1(6)E1 depuis l'ID de bogue Cisco [CSCds50549](#) (clients [enregistrés](#) seulement).
  - Modules atmosphère du Catalyst 5000/6000 — Depuis la version du logiciel Cisco IOS 12.0(14)W05(20), référez-vous à l'ID de bogue Cisco [CSCds07238](#) (clients [enregistrés](#) seulement).
  - WAN OSM de Gigabit Ethernet du Catalyst 6000 — Depuis la version du logiciel Cisco IOS 12.1.12E, référez-vous à l'ID de bogue Cisco [CSCdw64849](#) (clients [enregistrés](#) seulement).
  - Gamme Catalyst 6000 — Toutes les versions logicielles de Cisco IOS. Le support WS-F6K-MSFC et MSM commence dans le Logiciel Cisco IOS version 12.1(8a)E4.
  - Gamme Catalyst 8500 — Le support commence en date de la version du logiciel Cisco IOS 12.0(5)W5(13).
  - Les Routeurs de Cisco 3600, 4000, et Plateformes plus élevées — en date du logiciel de Cisco IOS 12.0(1) et du Logiciel Cisco IOS version 12.0(1)T, se réfèrent à l'ID de bogue Cisco [CSCdj93712](#) (clients [enregistrés](#) seulement) et à l'ID de bogue Cisco [CSCdt58029](#) (clients [enregistrés](#) seulement).
  - Interfaces de Relais de trames — En date du Logiciel Cisco IOS version 12.0(17)S et du Logiciel Cisco IOS version 12.2(4)T3, référez-vous au [Fonction Frame Relay 64-bit Counters](#).
  - Interfaces ATM OC3 — En date du Logiciel Cisco IOS version 12.0(6)T, référez-vous à l'ID de bogue Cisco [CSCdm45357](#) (clients [enregistrés](#) seulement).
  - Interfaces de tunnel — En date du Logiciel Cisco IOS version 12.0(16)S, référez-vous à l'ID de bogue Cisco [CSCdt58029](#) (clients [enregistrés](#) seulement).

**Remarque:** Le logiciel de Cisco IOS ne prend en charge pas les compteurs 64-bit pour des vitesses d'interface de moins de 20 Mb/s. Ceci signifie que des compteurs 64-bit ne sont pas pris en charge sur 10 ports Ethernet de mi-bande. Seulement 100 Fast Ethernet de mi-bande et d'autres ports à grande vitesse prennent en charge les compteurs 64-bit.

### **Q. Les ifInOctets et les compteurs SNMP d'ifOutOctets sont-ils les mêmes que l'exposition relie des compteurs d'entrée/sortie ?**

A. Oui, mais seulement quand le SNMP est activé du temps de démarrage. Si vous mettez sous tension un périphérique de Cisco, alors SNMP d'enable, le début de compteurs SNMP de 0. Ils ne prennent pas automatiquement leurs valeurs de la sortie CLI.

### **Q. Les ifInOctets et les compteurs d'ifOutOctets incluent-ils le temps système de tramage (protocole point-à-point, contrôle de liaison de données de haut niveau) ?**

A. Oui.

### **Q. Sur une interface d'Asynchronous Transfer Mode, les compteurs incluent-ils l'entête de cellule ?**

A. Les compteurs de Mode de transfert asynchrone (ATM) n'incluent pas le temps système atmosphère (les en-têtes de cellule et la remplissage AAL5).

## Q. Pourquoi les compteurs SNMP ne renvoient-ils pas le même nombre que des commandes show CLI ?

A. Un objet SNMP défini comme compteur doit se conformer à [RFC1155](#) :

### "3.2.3.3. Compteur

Ce type de la taille de l'application représente un entier non négatif qui monotoniquement des augmentations jusqu'à ce qu'il atteigne une valeur maximale, quand elle s'enveloppe autour et commence l'augmentation de nouveau de zéro. Cette note spécifie une valeur maximale de  $2^{32}-1$  (décimale 4294967295) pour des compteurs. »

Il n'y a aucune méthode pour remettre à l'état initial un SNMP à l'opposé de zéro sans nécessité de recharger le périphérique.

La sortie de compteur d'une **commande show CLI** peut être remise à l'état initial sur des interfaces puisque les restrictions SNMP ne sont pas présentes.

Les compteurs d'origine d'interface définis dans MIB-2 sont les compteurs de 32 bits. Pour 10 Mbits/s reliait, un compteur de 32 bits pourraient théoriquement s'envelopper en 57 minutes. Il est facile d'éviter des discontinuités avec une telle longue période. Mais pour des 100 Mbits/s, le temps de bouclage théorique minimum est de 5.7 minutes. Pour des interfaces 1 Gbps, il tombe à 34 secondes. A accordé ces périodes sont pour la transmission des paquets normaux dos à dos, un idéal théorique. Néanmoins, plus la vitesse d'interface est élevée, plus il devient pour éviter de manquer un contre- bouclage dur. Comme solution au problème, SMI SNMPv2 a défini un nouveau type d'objet, counter64, pour les compteurs 64-bit. Par conséquent, il y a plusieurs nouveaux compteurs 64-bit définis dans la table d'interface d'extension (ifxTable) définie dans [RFC 1573](#) (plus tard remplacé par [RFC 2233](#) ). [Ceux-ci sont de l'IF-MIB-V1SMI.my](#) (clients enregistrés seulement).

ifHCInOctets (.1.3.6.1.2.1.31.1.1.1.6)	ifHCOctets (1.3.6.1.2.1.31.1.1.1.10)
ifHCInUcastPkts (.1.3.6.1.2.1.31.1.1.1.7)	ifHCOctetsUcastPkts (.1.3.6.1.2.1.31.1.1.1.11)
ifHCInMulticastPkts (.1.3.6.1.2.1.31.1.1.1.8)	ifHCOctetsMulticastPkts (.1.3.6.1.2.1.31.1.1.1.12)
ifHCInBroadcastPkts (.1.3.6.1.2.1.31.1.1.1.9)	ifHCOctetsBroadcastPkts (.1.3.6.1.2.1.31.1.1.1.13)

Bien que la prise en charge de base pour les compteurs 64-bit ait été écrite dans le Logiciel Cisco IOS version 11.3, qui commence à partir du Logiciel Cisco IOS version 12.0, seulement les ifHCInOctets (.1.3.6.1.2.1.31.1.1.1.6) et les ifHCOctets (1.3.6.1.2.1.31.1.1.1.10) ont été mis en application pour des sous-interfaces de l'ATM LANE LEC seulement. Pour des Commutateurs de groupe de travail de Catalyst, le contre- support 64-bit a été mis en application dans la version 3.1.

**Remarque:** Vous devez utiliser le SNMPv2C ou le protocole SNMPv3 afin de récupérer en parent 64 objets.

## Compteurs SNMP et questions d'équivalent de commande show

**Q. Ce qui font les Routeurs de Cisco font pour les variables MIB suivantes SNMP : les ifInOctets, des ifInUcastPkts, des ifInNUcastPkts, des ifInDiscards, des ifInErrors, des ifInUnknownProtos, des ifOutOctets, des ifOutUcastPkts, des ifOutNUcastPkts, des ifOutDiscards, des ifOutErrors, et ifOutQLen ?**

A. Voir le ce tableau pour des détails. Ceux-ci sont du [RFC1213-MIB](#) (clients [enregistrés](#) seulement).

<b>ifInNUcastPkts</b> (.1.3.6.1.2.1.2.2.1.12)	Ce sont des comptes de diffusion en entrée et de paquets de multidiffusion.
<b>ifInDiscards</b> (.1.3.6.1.2.1.2.2.1.13)	Ceux-ci sont comptés en tant qu'aucune mémoires tampons comme envisagé dans la commande d' <b>interfaces d'exposition</b> .
<b>ifInErrors</b> (.1.3.6.1.2.1.2.2.1.14)	Ce sont des comptes de toutes les erreurs d'entrée comme envisagé dans la commande d' <b>interfaces d'exposition</b> .
<b>ifInUnknownProtos</b> (.1.3.6.1.2.1.2.2.1.15)	Ceux-ci sont comptés en tant qu'erreurs non classifiées.
<b>ifOutOctets</b> (.1.3.6.1.2.1.2.2.1.16)	Ce sont des comptes du nombre de sortie d'octets par l'interface suivant les indications de la commande d' <b>interfaces d'exposition</b> .
<b>ifOutUcastPkts</b> (.1.3.6.1.2.1.2.2.1.17)	Ce sont des comptes de paquets de diffusion et de multicast en sortie.
<b>ifOutDiscards</b> (.1.3.6.1.2.1.2.2.1.19)	Ceux-ci sont comptés comme suppressions de sortie suivant les indications de la commande d' <b>interfaces d'exposition</b> .
<b>ifOutErrors</b> (.1.3.6.1.2.1.2.2.1.20)	Ceux-ci sont comptés comme erreurs de sortie suivant les indications de la commande d' <b>interfaces d'exposition</b> .
<b>ifOutQLen</b> (.1.3.6.1.2.1.2.2.1.21)	C'est le nombre de paquets permis pour être sur la file d'attente de sortie suivant les indications de la commande d' <b>interfaces d'exposition</b> .

Les variables les ont précédemment répertorié qui n'indiquent pas qu'elles apparaissent dans des **interfaces d'exposition** ne sont pas disponibles n'importe où autre que le SNMP.

## Exemples

Cet exemple utilise des 3640 qui fonctionne avec le logiciel Release12.2(2)T1 de Cisco IOS. La

chaîne en lecture seule de la Communauté (RO) utilisée est publique et la chaîne de la Communauté de lecture-Write(RW) utilisée est privée. Référez-vous à [comment configurer des chaînes de caractères de la communauté SNMP](#) pour plus d'informations sur la façon configurer des chaînes de caractères de la communauté SNMP sur des périphériques.

Cette sortie est typique de la **commande brief de show ip interface** exécutée dans le mode enable :

```
3600#show ip interface brief Interface IP-Address OK? Method Status Prol BRI0/0 unassigned YES
NVRAM administratively down dow FastEthernet0/0 172.16.99.20 YES NVRAM up up Serial0/0
unassigned YES NVRAM down dow Serial0/0.1 unassigned YES unset down dow BRI0/0:1 unassigned YES
unset administratively down dow BRI0/0:2 unassigned YES unset administratively down dow
Serial0/1 unassigned YES NVRAM administratively down dow ATM1/0 unassigned YES NVRAM down dow
ATM1/0.109 10.164.0.46 YES NVRAM down dow Virtual-Templat1 99.99.99.99 YES NVRAM down dow
Loopback0 10.1.10.1 YES NVRAM up up Loopback1 unassigned YES NVRAM up up Loopback101 3.3.3.3 YES
NVRAM administratively down dow Loopback200 4.4.4.14 YES NVRAM administratively down dow
Loopback201 4.4.4.18 YES NVRAM administratively down dow
```

Cette sortie est l'objet MIB si **Descr (.1.3.6.1.2.1.2.2.1.2)** pour le routeur précédent, qui est une chaîne de texte qui contient des informations sur l'interface. Ceci donne le nom et la description d'interface comme obtenue, qui utilise la sortie de commande précédente CLI. **ifName (.1.3.6.1.2.1.31.1.1.1)** peut également être utilisé mais **ifDescr** donne la description d'interface avec le nom, où **ifName** donne seulement le nom d'interface.

```
snmpwalk 172.16.99.20 public .1.3.6.1.2.1.2.2.1.2
interfaces.ifTable.ifEntry.ifDescr.1 = ATM1/0
interfaces.ifTable.ifEntry.ifDescr.2 = BRI0/0
interfaces.ifTable.ifEntry.ifDescr.3 = FastEthernet0/0
interfaces.ifTable.ifEntry.ifDescr.4 = Serial0/0
interfaces.ifTable.ifEntry.ifDescr.5 = BRI0/0:1
interfaces.ifTable.ifEntry.ifDescr.6 = BRI0/0:2
interfaces.ifTable.ifEntry.ifDescr.7 = Serial0/1
interfaces.ifTable.ifEntry.ifDescr.8 = Null0
interfaces.ifTable.ifEntry.ifDescr.10 = Foreign Exchange Office 2/0/0
interfaces.ifTable.ifEntry.ifDescr.11 = Foreign Exchange Office 2/0/1
interfaces.ifTable.ifEntry.ifDescr.12 = recEive And transMit 3/0/0
interfaces.ifTable.ifEntry.ifDescr.13 = recEive And transMit 3/0/1
interfaces.ifTable.ifEntry.ifDescr.14 = Loopback0
interfaces.ifTable.ifEntry.ifDescr.15 = Loopback1
interfaces.ifTable.ifEntry.ifDescr.16 = Loopback101
interfaces.ifTable.ifEntry.ifDescr.17 = Loopback200
interfaces.ifTable.ifEntry.ifDescr.18 = Loopback201
interfaces.ifTable.ifEntry.ifDescr.19 = Serial0/0.1
interfaces.ifTable.ifEntry.ifDescr.20 = ATM1/0.109-atm subif
interfaces.ifTable.ifEntry.ifDescr.21 = ATM1/0.109-aal5 layer
interfaces.ifTable.ifEntry.ifDescr.22 = Virtual-Templat1
interfaces.ifTable.ifEntry.ifDescr.23 = Voice Encapsulation (POTS) Peer: 1
interfaces.ifTable.ifEntry.ifDescr.24 = Voice Over IP Peer: 2
interfaces.ifTable.ifEntry.ifDescr.25 = Voice Encapsulation (POTS) Peer: 111
interfaces.ifTable.ifEntry.ifDescr.26 = Voice Over IP Peer: 222
interfaces.ifTable.ifEntry.ifDescr.27 = Voice Over IP Peer: 1234
interfaces.ifTable.ifEntry.ifDescr.28 = Voice Over IP Peer: 300000
interfaces.ifTable.ifEntry.ifDescr.29 = Voice Over FR Peer: 3
interfaces.ifTable.ifEntry.ifDescr.30 = Voice Over IP Peer: 99
interfaces.ifTable.ifEntry.ifDescr.31 = Voice Encapsulation (POTS) Peer: 9
interfaces.ifTable.ifEntry.ifDescr.32 = BRI0/0-Physical
interfaces.ifTable.ifEntry.ifDescr.33 = BRI0/0-Signaling
interfaces.ifTable.ifEntry.ifDescr.34 = BRI0/0:1-Bearer Channel
interfaces.ifTable.ifEntry.ifDescr.35 = BRI0/0:2-Bearer Channel
```

1. ifInDiscards (.1.3.6.1.2.1.2.2.1.13) :snmpwalk 172.16.99.20 public .1.3.6.1.2.1.2.2.1.13

```
interfaces.ifTable.ifEntry.ifInDiscards.1 = Counter32: 0
interfaces.ifTable.ifEntry.ifInDiscards.2 = Counter32: 0
interfaces.ifTable.ifEntry.ifInDiscards.3 = Counter32: 0
interfaces.ifTable.ifEntry.ifInDiscards.4 = Counter32: 0
interfaces.ifTable.ifEntry.ifInDiscards.5 = Counter32: 0
interfaces.ifTable.ifEntry.ifInDiscards.6 = Counter32: 0
interfaces.ifTable.ifEntry.ifInDiscards.7 = Counter32: 0
interfaces.ifTable.ifEntry.ifInDiscards.8 = Counter32: 0
interfaces.ifTable.ifEntry.ifInDiscards.10 = Counter32: 0
interfaces.ifTable.ifEntry.ifInDiscards.11 = Counter32: 0
interfaces.ifTable.ifEntry.ifInDiscards.12 = Counter32: 0
interfaces.ifTable.ifEntry.ifInDiscards.13 = Counter32: 0
interfaces.ifTable.ifEntry.ifInDiscards.14 = Counter32: 0
interfaces.ifTable.ifEntry.ifInDiscards.15 = Counter32: 0
interfaces.ifTable.ifEntry.ifInDiscards.16 = Counter32: 0
interfaces.ifTable.ifEntry.ifInDiscards.17 = Counter32: 0
interfaces.ifTable.ifEntry.ifInDiscards.18 = Counter32: 0
interfaces.ifTable.ifEntry.ifInDiscards.19 = Counter32: 0
interfaces.ifTable.ifEntry.ifInDiscards.20 = Counter32: 0
interfaces.ifTable.ifEntry.ifInDiscards.21 = Counter32: 0
interfaces.ifTable.ifEntry.ifInDiscards.22 = Counter32: 0
interfaces.ifTable.ifEntry.ifInDiscards.23 = Counter32: 0
interfaces.ifTable.ifEntry.ifInDiscards.24 = Counter32: 0
interfaces.ifTable.ifEntry.ifInDiscards.25 = Counter32: 0
interfaces.ifTable.ifEntry.ifInDiscards.26 = Counter32: 0
interfaces.ifTable.ifEntry.ifInDiscards.27 = Counter32: 0
interfaces.ifTable.ifEntry.ifInDiscards.28 = Counter32: 0
interfaces.ifTable.ifEntry.ifInDiscards.29 = Counter32: 0
interfaces.ifTable.ifEntry.ifInDiscards.30 = Counter32: 0
interfaces.ifTable.ifEntry.ifInDiscards.31 = Counter32: 0
interfaces.ifTable.ifEntry.ifInDiscards.32 = Counter32: 0
interfaces.ifTable.ifEntry.ifInDiscards.33 = Counter32: 0
interfaces.ifTable.ifEntry.ifInDiscards.34 = Counter32: 0
```

interfaces.ifTable.ifEntry.ifInDiscards.35 = Counter32: 0 **Les ifInDiscards est zéro pour toutes les interfaces de ce routeur. Si vous comparez ceci au résultat CLI des shows**

**interfaces fastethernets 0/0** commande, ceci confirme le résultat :3600#show interfaces fastEthernet 0/0 FastEthernet0/0 is up, line protocol is up Hardware is AmdFE, address is 0001.42b4.fe81 (bia 0001.42b4.fe81) Description: testme Internet address is 172.16.99.20/24 MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec, reliability 255/255, txload 1/255, rxload 1/255 Encapsulation ARPA, loopback not set Keepalive set (10 sec) Full-duplex, 100Mb/s, 100BaseTX/FX ARP type: ARPA, ARP Timeout 04:00:00 Last input 00:00:00, output 00:00:00, output hang never Last clearing of "show interface" counters never Queueing strategy: fifo Output queue 0/40, 0 drops; input queue 0/75, 323 drops 5 minute input rate 1000 bits/sec, 2 packets/sec 5 minute output rate 0 bits/sec, 0 packets/sec 1767411 packets input, 178272010 bytes Received 1161500 broadcasts, 0 runts, 0 giants, 0 throttles 0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored 0 watchdog 0 input packets with dribble condition detected 7146925 packets output, 765049281 bytes, 0 underruns(0/0/0) 0 output errors, 0 collisions, 1 interface resets 0 babbles, 0 late collision, 461 deferred 0 lost carrier, 0 no carrier 0 output buffer failures, 0 output buffers swapped out

## 2. ifInErrors (.1.3.6.1.2.1.2.2.1.14) :snmpwalk 172.16.99.20 public .1.3.6.1.2.1.2.2.1.14

```
interfaces.ifTable.ifEntry.ifInErrors.1 = Counter32: 0
interfaces.ifTable.ifEntry.ifInErrors.2 = Counter32: 0
interfaces.ifTable.ifEntry.ifInErrors.3 = Counter32: 0
interfaces.ifTable.ifEntry.ifInErrors.4 = Counter32: 0
interfaces.ifTable.ifEntry.ifInErrors.5 = Counter32: 0
interfaces.ifTable.ifEntry.ifInErrors.6 = Counter32: 0
interfaces.ifTable.ifEntry.ifInErrors.7 = Counter32: 1
interfaces.ifTable.ifEntry.ifInErrors.8 = Counter32: 0
interfaces.ifTable.ifEntry.ifInErrors.10 = Counter32: 0
interfaces.ifTable.ifEntry.ifInErrors.11 = Counter32: 0
interfaces.ifTable.ifEntry.ifInErrors.12 = Counter32: 0
```



```
interfaces.ifTable.ifEntry.ifInErrors.13 = Counter32: 0
interfaces.ifTable.ifEntry.ifInErrors.14 = Counter32: 0
interfaces.ifTable.ifEntry.ifInErrors.15 = Counter32: 0
interfaces.ifTable.ifEntry.ifInErrors.16 = Counter32: 0
interfaces.ifTable.ifEntry.ifInErrors.17 = Counter32: 0
interfaces.ifTable.ifEntry.ifInErrors.18 = Counter32: 0
interfaces.ifTable.ifEntry.ifInErrors.19 = Counter32: 0
interfaces.ifTable.ifEntry.ifInErrors.20 = Counter32: 0
interfaces.ifTable.ifEntry.ifInErrors.21 = Counter32: 0
interfaces.ifTable.ifEntry.ifInErrors.22 = Counter32: 0
interfaces.ifTable.ifEntry.ifInErrors.23 = Counter32: 0
interfaces.ifTable.ifEntry.ifInErrors.24 = Counter32: 0
interfaces.ifTable.ifEntry.ifInErrors.25 = Counter32: 0
interfaces.ifTable.ifEntry.ifInErrors.26 = Counter32: 0
interfaces.ifTable.ifEntry.ifInErrors.27 = Counter32: 0
interfaces.ifTable.ifEntry.ifInErrors.28 = Counter32: 0
interfaces.ifTable.ifEntry.ifInErrors.29 = Counter32: 0
interfaces.ifTable.ifEntry.ifInErrors.30 = Counter32: 0
interfaces.ifTable.ifEntry.ifInErrors.31 = Counter32: 0
interfaces.ifTable.ifEntry.ifInErrors.32 = Counter32: 0
interfaces.ifTable.ifEntry.ifInErrors.33 = Counter32: 0
interfaces.ifTable.ifEntry.ifInErrors.34 = Counter32: 0
```

interfaces.ifTable.ifEntry.ifInErrors.35 = Counter32: 0 Cette sortie prouve qu'il y a seulement une erreur d'entrée pour l'interface **interfaces.ifTable.ifEntry.ifInErrors.7 = Counter32 : 1**. Afin de déterminer quelle interface c'est, comparez-la à la sortie de l'**ifDescr** ci-dessus, qui prouve que c'est de **interfaces.ifTable.ifEntry.ifDescr.7 = Serial0/1**. Exécutez maintenant la commande du **show interfaces serial 0/1** dans le mode enable afin de vérifier le résultat précédent :

```
3600#show interfaces serial 0/1 Serial0/1 is administratively down,
line protocol is down Hardware is DSCC4 Serial Description: atm-dxi test MTU 1500 bytes, BW
2048 Kbit, DLY 20000 usec, reliability 255/255, txload 1/255, rxload 1/255 Encapsulation
ATM-DXI, loopback not set Keepalive not set Last input never, output never, output hang
never Last clearing of "show interface" counters 1w1d Queueing strategy: fifo Output queue
0/40, 0 drops; input queue 0/75, 0 drops 5 minute input rate 0 bits/sec, 0 packets/sec 5
minute output rate 0 bits/sec, 0 packets/sec 0 packets input, 0 bytes, 0 no buffer Received
0 broadcasts, 0 runts, 0 giants, 0 throttles 1 input errors, 0 CRC, 0 frame, 0 overrun, 0
ignored, 1 abort 0 packets output, 0 bytes, 0 underruns 0 output errors, 0 collisions, 0
interface resets 0 output buffer failures, 0 output buffers swapped out 0 carrier
transitions DCD=down DSR=down DTR=down RTS=down CTS=down
```

3. **ifOutOctets (.1.3.6.1.2.1.2.2.1.16)** :snmpwalk 172.16.99.20 public .1.3.6.1.2.1.2.2.1.16

```
interfaces.ifTable.ifEntry.ifOutOctets.1 = Counter32: 98
interfaces.ifTable.ifEntry.ifOutOctets.2 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutOctets.3 = Counter32: 765470674
interfaces.ifTable.ifEntry.ifOutOctets.4 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutOctets.5 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutOctets.6 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutOctets.7 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutOctets.8 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutOctets.10 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutOctets.11 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutOctets.12 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutOctets.13 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutOctets.14 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutOctets.15 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutOctets.16 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutOctets.17 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutOctets.18 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutOctets.19 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutOctets.20 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutOctets.21 = Counter32: 98
interfaces.ifTable.ifEntry.ifOutOctets.22 = Counter32: 0
```



```

interfaces.ifTable.ifEntry.ifOutOctets.23 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutOctets.24 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutOctets.25 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutOctets.26 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutOctets.27 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutOctets.28 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutOctets.29 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutOctets.30 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutOctets.31 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutOctets.32 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutOctets.33 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutOctets.34 = Counter32: 0

```

Si vous comparez le résultat précédent à la sortie de l'ifDescr, ceci indique : **interfaces.ifTable.ifEntry.ifOutOctets.1 = Counter32 : 98** correspond à **interfaces.ifTable.ifEntry.ifDescr.1 = ATM1/0** **interfaces.ifTable.ifEntry.ifOutOctets.3 = Counter32 : 765470674** correspond à **interfaces.ifTable.ifEntry.ifDescr.3 = FastEthernet0/0** **interfaces.ifTable.ifEntry.ifOutOctets.21 = Counter32 : 98** correspond à la couche **interfaces.ifTable.ifEntry.ifDescr.21 = ATM1/0.109-aal5C** c'est la sortie des interfaces d'exposition CLI commandent pour chacune des interfaces

```

3600#show interfaces atM 1/0 ATM1/0 is down,
line protocol is down Hardware is RS8234 ATMOC3 MTU 4470 bytes, sub MTU 4470, BW 155000
Kbit, DLY 80 usec, reliability 5/255, txload 1/255, rxload 1/255 Encapsulation ATM,
loopback not set Encapsulation(s): AAL5 1024 maximum active VCs, 1 current VCCs VC idle
disconnect time: 300 seconds Last input never, output 1wld, output hang never Last clearing
of "show interface" counters never Input queue: 0/75/0/0 (size/max/drops/flushes); Total
output drops: 0 Queueing strategy: None 5 minute input rate 0 bits/sec, 0 packets/sec 5
minute output rate 0 bits/sec, 0 packets/sec 0 packets input, 0 bytes, 0 no buffer Received
0 broadcasts, 0 runts, 0 giants, 0 throttles 0 input errors, 0 CRC, 0 frame, 0 overrun, 0
ignored, 0 abort 2 packets output, 98 bytes, 0 underruns 0 output errors, 0 collisions, 2
interface resets 0 output buffer failures, 0 output buffers swapped out 3600#show
interfaces fastEthernet 0/0 FastEthernet0/0 is up, line protocol is up Hardware is AmdFE,
address is 0001.42b4.fe81 (bia 0001.42b4.fe81) Description: testme Internet address is
172.16.99.20/24 MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec, reliability 255/255, txload
1/255, rxload 1/255 Encapsulation ARPA, loopback not set Keepalive set (10 sec) Full-
duplex, 100Mb/s, 100BaseTX/FX ARP type: ARPA, ARP Timeout 04:00:00 Last input 00:00:00,
output 00:00:00, output hang never Last clearing of "show interface" counters never
Queueing strategy: fifo Output queue 0/40, 0 drops; input queue 0/75, 323 drops 5 minute
input rate 2000 bits/sec, 3 packets/sec 5 minute output rate 1000 bits/sec, 1 packets/sec
1772214 packets input, 178767841 bytes Received 1164210 broadcasts, 0 runts, 0 giants, 0
throttles 0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored 0 watchdog 0 input packets
with dribble condition detected 7149179 packets output, 765450524 bytes, 0 underruns(0/0/0)
0 output errors, 0 collisions, 1 interface resets 0 babbles, 0 late collision, 461 deferred
0 lost carrier, 0 no carrier 0 output buffer failures, 0 output buffers swapped out La

```

sortie des ifOutOctets n'apparie pas le CLI sorti pour les shows interfaces fastethernets 0/0 commande, mais elle est semblable. C'est parce qu'il pourrait y a un retard quand les

```

3600#show interfaces atM
1/0.109 ATM1/0.109 is down, line protocol is down Hardware is RS8234 ATMOC3 Description:
pvc Internet address is 10.164.0.46/30 MTU 4470 bytes, BW 2250 Kbit, DLY 80 usec,
reliability 5/255, txload 1/255, rxload 1/255 Encapsulation ATM 0 packets input, 0 bytes 2
packets output,98 bytes 0 OAM cells input, 77093 OAM cells output AAL5 CRC errors : 0 AAL5
SAR Timeouts : 0 AAL5 Oversized SDUs : 0 AAL5 length violation : 0 AAL5 CPI Error : 0

```

#### 4. ifOutDiscards (.1.3.6.1.2.1.2.2.1.19) :snmpwalk 172.16.99.20 public .1.3.6.1.2.1.2.2.1.19

```

interfaces.ifTable.ifEntry.ifOutDiscards.1 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutDiscards.2 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutDiscards.3 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutDiscards.4 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutDiscards.5 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutDiscards.6 = Counter32: 0

```

```
interfaces.ifTable.ifEntry.ifOutDiscards.7 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutDiscards.8 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutDiscards.10 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutDiscards.11 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutDiscards.12 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutDiscards.13 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutDiscards.14 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutDiscards.15 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutDiscards.16 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutDiscards.17 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutDiscards.18 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutDiscards.19 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutDiscards.20 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutDiscards.21 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutDiscards.22 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutDiscards.23 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutDiscards.24 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutDiscards.25 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutDiscards.26 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutDiscards.27 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutDiscards.28 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutDiscards.29 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutDiscards.30 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutDiscards.31 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutDiscards.32 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutDiscards.33 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutDiscards.34 = Counter32: 0
```

interfaces.ifTable.ifEntry.ifOutDiscards.35 = Counter32: 0 **Les ifOutDiscards est zéro pour toutes les interfaces. Avec les shows interfaces fastethernets 0/0 commande comme**

**exemple, cette commande produit ce résultat :3600#show interfaces fastEthernet 0/0**

```
FastEthernet0/0 is up, line protocol is up Hardware is AmdFE, address is 0001.42b4.fe81
(bia 0001.42b4.fe81) Description: testme Internet address is 172.16.99.20/24 MTU 1500
bytes, BW 100000 Kbit, DLY 100 usec, reliability 255/255, txload 1/255, rxload 1/255
Encapsulation ARPA, loopback not set Keepalive set (10 sec) Full-duplex, 100Mb/s,
100BaseTX/FX ARP type: ARPA, ARP Timeout 04:00:00 Last input 00:00:00, output 00:00:00,
output hang never Last clearing of "show interface" counters never Queueing strategy: fifo
Output queue 0/40, 0 drops; input queue 0/75, 323 drops 5 minute input rate 1000 bits/sec,
2 packets/sec 5 minute output rate 1000 bits/sec, 1 packets/sec 1774581 packets input,
179005552 bytes Received 1165620 broadcasts, 0 runts, 0 giants, 0 throttles 0 input errors,
0 CRC, 0 frame, 0 overrun, 0 ignored 0 watchdog 0 input packets with dribble condition
detected 7150259 packets output, 765645035 bytes, 0 underruns(0/0/0) 0 output errors, 0
collisions, 1 interface resets 0 babbles, 0 late collision, 461 deferred 0 lost carrier, 0
no carrier 0 output buffer failures, 0 output buffers swapped out
```

5. ifOutErrors (.1.3.6.1.2.1.2.2.1.20) :snmpwalk 172.16.99.20 public .1.3.6.1.2.1.2.2.1.20

```
interfaces.ifTable.ifEntry.ifOutErrors.1 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutErrors.2 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutErrors.3 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutErrors.4 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutErrors.5 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutErrors.6 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutErrors.7 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutErrors.8 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutErrors.10 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutErrors.11 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutErrors.12 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutErrors.13 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutErrors.14 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutErrors.15 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutErrors.16 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutErrors.17 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutErrors.18 = Counter32: 0
```

```

interfaces.ifTable.ifEntry.ifOutErrors.19 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutErrors.20 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutErrors.21 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutErrors.22 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutErrors.23 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutErrors.24 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutErrors.25 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutErrors.26 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutErrors.27 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutErrors.28 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutErrors.29 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutErrors.30 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutErrors.31 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutErrors.32 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutErrors.33 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutErrors.34 = Counter32: 0
interfaces.ifTable.ifEntry.ifOutErrors.35 = Counter32: 0

```

**Les ifOutErrors est zéro pour toutes les interfaces. Avec les shows interfaces fastethernets 0/0 commande comme exemple, cette commande produit ce résultat :**

```

3600#show interfaces fastEthernet 0/0
FastEthernet0/0 is up, line protocol is up Hardware is AmdFE, address is 0001.42b4.fe81
(bia 0001.42b4.fe81) Description: testme Internet address is 172.16.99.20/24 MTU 1500
bytes, BW 100000 Kbit, DLY 100 usec, reliability 255/255, txload 1/255, rxload 1/255
Encapsulation ARPA, loopback not set Keepalive set (10 sec) Full-duplex, 100Mb/s,
100BaseTX/FX ARP type: ARPA, ARP Timeout 04:00:00 Last input 00:00:00, output 00:00:00,
output hang never Last clearing of "show interface" counters never Queueing strategy: fifo
Output queue 0/40, 0 drops; input queue 0/75, 323 drops 5 minute input rate 0 bits/sec, 1
packets/sec 5 minute output rate 0 bits/sec, 0 packets/sec 1776187 packets input, 179154616
bytes Received 1166778 broadcasts, 0 runts, 0 giants, 0 throttles 0 input errors, 0 CRC, 0
frame, 0 overrun, 0 ignored 0 watchdog 0 input packets with dribble condition detected
7150781 packets output, 765744231 bytes, 0 underruns(0/0/0) 0 output errors, 0 collisions,
1 interface resets 0 babbles, 0 late collision, 461 deferred 0 lost carrier, 0 no carrier 0
output buffer failures, 0 output buffers swapped out

```

## 6. ifOutQLen (.1.3.6.1.2.1.2.2.1.21) :snmpwalk 172.16.99.20 public .1.3.6.1.2.1.2.2.1.21

```

interfaces.ifTable.ifEntry.ifOutQLen.1 = Gauge32: 0
interfaces.ifTable.ifEntry.ifOutQLen.2 = Gauge32: 0
interfaces.ifTable.ifEntry.ifOutQLen.3 = Gauge32: 0 interfaces.ifTable.ifEntry.ifOutQLen.4
= Gauge32: 0 interfaces.ifTable.ifEntry.ifOutQLen.5 = Gauge32: 0
interfaces.ifTable.ifEntry.ifOutQLen.6 = Gauge32: 0 interfaces.ifTable.ifEntry.ifOutQLen.7
= Gauge32: 0 interfaces.ifTable.ifEntry.ifOutQLen.8 = Gauge32: 0
interfaces.ifTable.ifEntry.ifOutQLen.10 = Gauge32: 0
interfaces.ifTable.ifEntry.ifOutQLen.11 = Gauge32: 0
interfaces.ifTable.ifEntry.ifOutQLen.12 = Gauge32: 0
interfaces.ifTable.ifEntry.ifOutQLen.13 = Gauge32: 0
interfaces.ifTable.ifEntry.ifOutQLen.14 = Gauge32: 0
interfaces.ifTable.ifEntry.ifOutQLen.15 = Gauge32: 0
interfaces.ifTable.ifEntry.ifOutQLen.16 = Gauge32: 0
interfaces.ifTable.ifEntry.ifOutQLen.17 = Gauge32: 0
interfaces.ifTable.ifEntry.ifOutQLen.18 = Gauge32: 0
interfaces.ifTable.ifEntry.ifOutQLen.19 = Gauge32: 0
interfaces.ifTable.ifEntry.ifOutQLen.20 = Gauge32: 0
interfaces.ifTable.ifEntry.ifOutQLen.21 = Gauge32: 0
interfaces.ifTable.ifEntry.ifOutQLen.22 = Gauge32: 0
interfaces.ifTable.ifEntry.ifOutQLen.23 = Gauge32: 0
interfaces.ifTable.ifEntry.ifOutQLen.24 = Gauge32: 0
interfaces.ifTable.ifEntry.ifOutQLen.25 = Gauge32: 0
interfaces.ifTable.ifEntry.ifOutQLen.26 = Gauge32: 0
interfaces.ifTable.ifEntry.ifOutQLen.27 = Gauge32: 0
interfaces.ifTable.ifEntry.ifOutQLen.28 = Gauge32: 0
interfaces.ifTable.ifEntry.ifOutQLen.29 = Gauge32: 0
interfaces.ifTable.ifEntry.ifOutQLen.30 = Gauge32: 0
interfaces.ifTable.ifEntry.ifOutQLen.31 = Gauge32: 0

```

```

interfaces.ifTable.ifEntry.ifOutQLen.32 = Gauge32: 0
interfaces.ifTable.ifEntry.ifOutQLen.33 = Gauge32: 0
interfaces.ifTable.ifEntry.ifOutQLen.34 = Gauge32: 0
interfaces.ifTable.ifEntry.ifOutQLen.35 = Gauge32: 0

```

**L'ifOutQLen est zéro pour toutes les interfaces. Avec les `shows interfaces fastethernets 0/0` commande comme exemple**

```

:3600#show interfaces fastEthernet 0/0 FastEthernet0/0 is up, line protocol is up Hardware
is AmdFE, address is 0001.42b4.fe81 (bia 0001.42b4.fe81) Description: testme Internet
address is 172.16.99.20/24 MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec, reliability
255/255, txload 1/255, rxload 1/255 Encapsulation ARPA, loopback not set Keepalive set (10
sec) Full-duplex, 100Mb/s, 100BaseTX/FX ARP type: ARPA, ARP Timeout 04:00:00 Last input
00:00:00, output 00:00:00, output hang never Last clearing of "show interface" counters
never Queueing strategy: fifo Output queue 0/40, 0 drops; input queue 0/75, 323 drops 5
minute input rate 0 bits/sec, 1 packets/sec 5 minute output rate 0 bits/sec, 0 packets/sec
1776912 packets input, 179225431 bytes Received 1167240 broadcasts, 0 runts, 0 giants, 0
throttles 0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored 0 watchdog 0 input packets
with dribble condition detected 7151102 packets output, 765796341 bytes, 0 underruns(0/0/0)
0 output errors, 0 collisions, 1 interface resets 0 babbles, 0 late collision, 461 deferred
0 lost carrier, 0 no carrier 0 output buffer failures, 0 output buffers swapped out

```

**Q. Que les relations entre les déclarations d'interfaces d'exposition ne sont-elles aucune mémoire tampon Et perte de file d'attente d'entrée ? Pourquoi les inDiscards du SNMP donnent-ils à aucune mémoires tampons des comptes et pas des pertes de file d'attente d'entrée, alors que les outDiscards du SNMP donnent des pertes de file d'attente de sortie ?**

A. Le travail loclflInputQueueDrops/iflInDiscards différemment que loclflOutputQueueDrops/ifOutDiscards. Les iflInDiscards compte le nombre de paquets qui sont jetés faute de ressource système telle qu'une mémoire tampon. C'est généralement un sous-ensemble des loclflInputQueueDrops. Vous voyez souvent des loclflInputQueueDrops = des iflInDiscards. Mais, les loclflInputQueueDrops compte également le nombre de paquets abandonnés parce qu'ils frappent la limite de file d'attente d'entrée. Tellement généralement, vous voyez des loclflInputQueueDrops > des iflInDiscards.

### Résumé

les loclflInputQueueDrops = la limite de file d'attente chute + aucun iflInDiscards d'abandons de mémoire tampon = aucun abandons de mémoire tampon (et est un sous-ensemble de loclflInputQueueDrops)

Les loclflOutputQueueDrops et les ifOutDiscards sont toujours égaux quand ils comptent les mêmes événements. Ces événements frappent la limite de file d'attente de sortie, et n'ont pas une mémoire tampon de tx de matériel quand un paquet fastswitched d'une interface à l'autre. Les OID des objets précédents MIB sont ceux-ci :

<a href="#">D'OLD-CISCO-INTERFACES-MIB</a> (clients <a href="#">enregistrés</a> seulement)	<a href="#">De RFC1213-MIB</a> (clients <a href="#">enregistrés</a> seulement)
locflInputQueueDrops = .1.3.6.1.4.1.9.2.2.1.1.26	iflInDiscards = .1.3.6.1.2.1.2.2.1.13
locflOutputQueueDrops = .1.3.6.1.4.1.9.2.2.1.1.27	ifOutDiscards = .1.3.6.1.2.1.2.2.1.19

**Q. Est-ce que je peux ne voter aucune mémoire tampon sur un routeur ?**

A. Oui. Vous pouvez voter pour des iflnDiscards afin de ne voter aucune mémoire tampon.

**Q. Comment est-ce que je vote des baisses de limite de file d'attente sur un routeur ?**

A. Avec l'utilisation du SNMP, il n'y a aucune manière pour les **interfaces d'exposition** commandent d'éclater les différents éléments qui entrent dans les suppressions de sortie.

Considérez ces nouvelles informations sur ce qui entre dans les suppressions de sortie contre- :

Les baisses de limite de suppressions d'entrée = de file d'attente + étranglant chute + baisses de file d'attente SMT de pleines + RSRB relâche + aucun abandons de mémoire tampon

En outre, des compteurs SNMP ne sont jamais effacés, même si les interfaces sont effacées.

## [Informations connexes](#)

- [Conseils techniques de Services d'applications IP](#)
- [Support et documentation techniques - Cisco Systems](#)