

SNMP-Zähler: Häufig gestellte Fragen

Inhalt

[Einleitung](#)

[Fragen zu SNMP-Zählern](#)

[SNMP-Zähler und show Command Equivalent Questions](#)

[Zugehörige Informationen](#)

Einleitung

Dieses Dokument bietet Antworten auf häufig gestellte Fragen und unterstützt Benutzer bei der Suche nach hilfreichen Ressourcen zu SNMP- und SNMP-Problemen im Zusammenhang mit Cisco Geräten.

Fragen zu SNMP-Zählern

F. Welche MIB sollte ich für Schnittstellenzähler verwenden?

Antwort: Die Schnittstellenverwaltung über SNMP basiert auf zwei Tabellen: [ifTable](#) (nur [registrierte Kunden](#)) und deren Erweiterung, [fallsXTable](#) (nur [registrierte Kunden](#)), [beschrieben in RFC1213/RFC2233](#). Schnittstellen können mehrere Ebenen haben, die von den Medien abhängig sind, und jede Unterschicht wird durch eine separate Zeile in der Tabelle dargestellt. Die Beziehung zwischen der höheren und der unteren Ebene wird in der [ifStackTable](#) beschrieben (nur [registrierte Kunden](#)). Die ifTable definiert 32-Bit-Zähler für ein- und ausgehende Oktette (ifInOctets (nur [registrierte Kunden](#)) /ifOutOctets), Pakete (ifInUcastPkts (nur [registrierte Kunden](#)) /ifOutUcastPkts (nur [registrierte Kunden](#)), wennInNUcastPkts definiert. (nur [registrierte Kunden](#)) /ifOutNUcastPkts (nur [registrierte Kunden](#)), [Fehler und Rückwürfe](#). Die ifXTable stellt ähnliche 64-Bit-Zähler bereit, die auch als Zähler für hohe Kapazität (High Capacity, HC) bezeichnet werden: [ifHCInOctets](#) (nur [registrierte Kunden](#)) /[ifHCOutOctets](#) (nur [registrierte Kunden](#)) und [ifHCInUcastPkts](#) (nur [registrierte Kunden](#)) /[ifHCOutUcastPkts](#) (nur [registrierte Kunden](#)) .

F. Wann sollten 64-Bit-Zähler verwendet werden?

Antwort: [RFC 2233](#) führte erweiterte 64-Bit-Zähler für Schnittstellen mit hoher Kapazität ein, bei denen 32-Bit-Zähler nicht genügend Kapazität bereitstellen und zu schnell umwickeln.

Mit zunehmender Geschwindigkeit von Netzwerkmedien verringert sich die Mindestzeit, in der ein 32-Bit-Counter-Wraps verwendet wird. Ein 10-Mbit/s-Stream von Back-to-Back-Paketen voller Größe veranlasst ifInOctets zum Einwickeln von etwas mehr als 57 Minuten. Bei 100 Mbit/s beträgt die Mindestumschlagzeit 5,7 Minuten, bei 1 Gbit/s 34 Sekunden.

Hinweis: Die Zähler der SNMP-Zähler, die Zähler der Befehlszeilenschnittstelle (CLI), werden nicht eingewickelt.

Für Schnittstellen, die mit 20.000.000 (20 Millionen) Bit pro Sekunde oder weniger arbeiten, müssen Sie 32-Bit-Byte- und Paketzähler verwenden. Für Schnittstellen, die schneller als 20 Millionen Bit pro Sekunde und langsamer als 650.000.000 Bit pro Sekunde arbeiten, müssen 32-Bit-Paketzähler und 64-Bit-Oktett-Zähler verwendet werden. Für Schnittstellen, die mit 650.000.000 Bit/s oder schneller arbeiten, müssen 64-Bit-Paket- und Oktett-Zähler verwendet werden.

Dementsprechend unterstützt die Cisco IOS® Software keine 64-Bit-Zähler für Schnittstellengeschwindigkeiten von weniger als 20 Mbit/s. Das bedeutet, dass 64-Bit-Zähler von 10-Mbit-Ethernet-Ports nicht unterstützt werden, nur 100-Mbit-Fast-Ethernet und andere Hochgeschwindigkeits-Ports 64-Bit-Zähler unterstützen.

F. Welche SNMP-Version ist erforderlich, um 64-Bit-Zähler abzufragen?

Antwort: Zur Abfrage von 64-Bit-Zählern ist SNMPv2c oder SNMPv3 erforderlich. SNMPv1 unterstützt keine 64-Bit-Zähler. Beachten Sie, dass ifInOctets = 0,1.3.6.1.2.1.2.2.1.10 ein 32-Bit-Zähler ist, während die 64-Bit-Version ifHCInOctets = 0,1.3.6.1.2.1.31.1.1.6 ist.

Beispiele:

Catalyst 5000 verwendet HP OpenView snmpget, das standardmäßig auf SNMPv1 festgelegt ist.

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

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

Abfrage mit SNMPv1, Standard für HP OpenView snmpget

```
# 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.
```

Dieselbe Abfrage mit SNMPv2c

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

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

F. Welche Cisco Geräte implementieren 64-Bit-SNMP-Zähler, insbesondere für die IF-MIB?

Antwort: Diese Cisco Geräte implementieren 64-Bit-SNMP-Zähler:

Hinweis: Sie müssen ein registrierter Benutzer sein und angemeldet sein, um die Cisco Bug ID-Links aufzurufen und detaillierte Bug-Informationen zu sehen.

- Cisco 2500- und 2600-Router - Diese Router unterstützen keine 64-Bit-Zähler.
- Catalyst 2950 und 3550 - Der Support beginnt in der Cisco IOS Software, Version 12.1(11)EA1, seit Cisco Bug ID [CSCdx67611](#) (nur registrierte Kunden) und Cisco Bug ID [CSCdw528077](#) (nur registrierte Kunden)
- Catalyst 2900XL und 3500XL - Die Unterstützung beginnt in der Cisco IOS Software Release 12.0(5)WC3 seit Cisco Bug ID [CSCds45300](#) (nur registrierte Kunden).
- Catalyst 5000-Serie - Seit Cisco IOS Software Release 3.x. Bei RSM/RSFC beginnt die Unterstützung in Version 12.1(6)E1 der Cisco IOS-Software seit Cisco Bug ID [CSCds50549](#) (nur registrierte Kunden).
- Catalyst 5000/6000 ATM-Module - Seit Cisco IOS Software Release 12.0(14)W05(20), siehe Cisco Bug ID [CSCds07238](#) (nur registrierte Kunden).
- Catalyst 6000 Gigabit Ethernet WAN OSM - Seit Version 12.1.12E der Cisco IOS Software, siehe Cisco Bug ID [CSCdw64849](#) (nur registrierte Kunden).
- Catalyst 6000-Serie - Alle Cisco IOS-Softwareversionen Die Unterstützung für WS-F6K-MSFC und MSM beginnt mit der Cisco IOS Software, Version 12.1(8a)E4.
- Catalyst 8500-Serie - Die Unterstützung beginnt mit der Cisco IOS Software, Version 12.0(5)W5(13).
- Cisco Router der Serien 3600, 4000 und höher - Informationen zu Cisco IOS Software 12.0(1) und Cisco IOS Software Release 12.0(1)T finden Sie unter Cisco Bug ID [CSCdj93712](#) (nur registrierte Kunden) und Cisco Bug ID [CSCdt58029](#) (nur registrierte Kunden).
- Frame Relay-Schnittstellen - Ab der Cisco IOS Software Version 12.0(17)S und der Cisco IOS Software Version 12.2(4)T3 finden Sie Informationen unter [Frame Relay 64-Bit-Zähler](#).
- OC3 ATM-Schnittstellen - Informationen zur Cisco IOS-Softwareversion 12.0(6)T finden Sie unter Cisco Bug ID [CSCdm45357](#) (nur registrierte Kunden).
- Tunnelschnittstellen - Informationen zur Cisco IOS Software-Version 12.0(16)S finden Sie unter Cisco Bug ID [CSCdt58029](#) (nur registrierte Kunden).

Hinweis: Die Cisco IOS Software unterstützt keine 64-Bit-Zähler für Schnittstellengeschwindigkeiten von weniger als 20 Mbit/s. Das bedeutet, dass 64-Bit-Zähler von 10-Mbit-Ethernet-Ports nicht unterstützt werden. Nur 100-Mbit-Fast-Ethernet und andere Hochgeschwindigkeits-Ports unterstützen 64-Bit-Zähler.

F. Sind die Zähler ifInOctets und ifOutOctets für SNMP dieselben wie die Zähler zum Anzeigen der Schnittstellen Ein/Aus?

Antwort: Ja, aber nur, wenn SNMP vom Startzeitpunkt an aktiviert ist. Wenn Sie ein Cisco Gerät einschalten, aktivieren Sie SNMP. Die SNMP-Zähler beginnen von 0. Sie übernehmen ihre Werte nicht automatisch aus der CLI-Ausgabe.

F. Umfasst die ifInOctets-Zähler und die ifOutOctets-Zähler Framing-Overhead (Point-to-Point-Protokoll, High-Level Data Link Control)?

Antwort: Ja.

F. Enthalten die Zähler auf einer Schnittstelle des asynchronen Übertragungsmodus den Zellenheader?

Antwort: Die Zähler für den asynchronen Übertragungsmodus (ATM) enthalten keinen ATM-Overhead (Zellenheader und AAL5-Padding).

F. Warum geben SNMP-Zähler nicht dieselbe Nummer zurück wie CLI-Befehle show?

Antwort: Ein als Zähler definiertes SNMP-Objekt muss [RFC1155](#) entsprechen:

"3.2.3.3. Zähler

Dieser anwendungsweite Typ stellt eine nicht negative ganze Zahl dar, die monoton ansteigt, bis sie einen maximalen Wert erreicht, wenn sie umhüllt und wieder von Null ansteigt. In diesem Memo wird für Zähler ein Höchstwert von $2^{32}-1$ (4294967295 Dezimalstellen) angegeben."

Es gibt keine Methoden zum Zurücksetzen eines SNMP-Zählers auf Null, ohne dass das Gerät neu geladen werden muss.

Die Zählerausgabe eines CLI **show**-Befehls kann auf Schnittstellen zurückgesetzt werden, da die SNMP-Einschränkungen nicht vorhanden sind.

Die in MIB-2 definierten ursprünglichen Schnittstellenindikatoren sind 32-Bit-Zähler. Bei einer 10-Mbit/s-Schnittstelle könnte ein 32-Bit-Zähler theoretisch in 57 Minuten umwickeln. Es ist leicht, Diskontinuitäten mit solch langer Zeit zu vermeiden. Für 100 Mbit/s beträgt die theoretische Wrap-Zeit jedoch mindestens 5,7 Minuten. Für 1-Gbit/s-Schnittstellen sind es 34 Sekunden. Diese Zeiten gelten für die Übertragung von Back-to-Back-Paketen in voller Größe, ein theoretisches Ideal. Je höher die Schnittstellengeschwindigkeit, desto schwieriger wird es, das Verpassen eines Zählerwickels zu vermeiden. Als Lösung für dieses Problem definierte SNMPv2 SMI einen neuen Objekttyp, counter64, für 64-Bit-Zähler. Aus diesem Grund sind in der in [RFC 1573](#) definierten Schnittstellentabelle (ifxTable) mehrere neue 64-Bit-Zähler definiert (die später durch [RFC 2233](#) ersetzt wird). Diese stammen von der [IF-MIB-V1SMI.my](#) (nur [registrierte](#) Kunden).

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)	ifHCOUcastPkts (.1.3.6.1.2.1.31.1.1.1.11)
ifHCInMulticastPkts (.1.3.6.1.2.1.31.1.1.1.8)	ifHCOMulticastPkts (.1.3.6.1.2.1.31.1.1.1.12)
ifHCInBroadcast-Pkte (.1.3.6.1.2.1.31.1.1.1.9)	ifHCOBroadcast-Pkte (.1.3.6.1.2.1.31.1.1.1.13)

Obwohl die grundlegende Unterstützung für 64-Bit-Zähler in Version 11.3 der Cisco IOS-Software geschrieben wurde, die mit Version 12.0 der Cisco IOS-Software beginnt, sind nur ifHCInOctets (.1.3.6.1.2.1.31.1.1.1.6) und HCOctets (1.3.6.1.2.1.31.1.1.1.1) 1.10) wurden nur für ATM LANE LEC Sub-Schnittstellen implementiert. Für Catalyst-Workgroup-Switches wurde die Unterstützung von 64-Bit-Zählern in Version 3.1 implementiert.

Hinweis: Sie müssen das SNMPv2c- oder SNMPv3-Protokoll verwenden, um Zählerobjekte mit 64 Objekten abzurufen.

SNMP-Zähler und show Command Equivalent Questions

F. Was tun Cisco Router für die folgenden SNMP-MIB-Variablen? ifInOctets, ifInUcastPkts, ifInNUcastPkts, ifInDiscards, ifInErrors, ifInUnknownProtos,

ifOutOctets, ifOutUcastPkts, ifOutNUcastPkts, ifOutDiscards Fehler und ifOutQLen?

Antwort: Einzelheiten finden Sie in dieser Tabelle. Diese stammen von der [RFC1213-MIB](#) (nur registrierte Kunden).

ifInNUcastPkts (.1.3.6.1.2.1.2.2.1.12)	Dies sind die Anzahl der eingehenden Broadcast- und Multicast-Pakete.
ifInDiscards (.1.3.6.1.2.1.2.2.1.13)	Diese werden als keine Puffer gezählt, wie im Befehl show interfaces dargestellt.
ifInErrors (.1.3.6.1.2.1.2.2.1.14)	Dies sind die Zählungen aller Eingabefehler, die im Befehl show interfaces angezeigt werden.
ifInUnknownProtos (.1.3.6.1.2.1.2.2.1.15)	Diese werden als nicht klassifizierte Fehler gezählt.
ifOutOctets (.1.3.6.1.2.1.2.2.1.16)	Dies sind die Anzahl der Bytes, die von der Schnittstelle ausgegeben werden, wie im Befehl show interfaces dargestellt.
ifOutUcastPkts (.1.3.6.1.2.1.2.2.1.17)	Dies sind die Anzahl der ausgehenden Broadcast- und Multicast-Pakete.
ifOutDiscards (.1.3.6.1.2.1.2.2.1.19)	Diese werden als Ausgabeverwerfen gezählt, wie im Befehl show interfaces dargestellt.
ifOutErrors (.1.3.6.1.2.1.2.2.1.20)	Diese werden als Ausgabefehler gezählt, wie im Befehl show interfaces dargestellt.
ifOutQLen (.1.3.6.1.2.1.2.2.1.21)	Dies ist die Anzahl der Pakete, die zur Ausgabewarteschlange gehören dürfen, wie im Befehl show interfaces dargestellt.

Die zuvor aufgelisteten Variablen, die nicht angeben, dass sie in **Show Interfaces** erscheinen, sind nirgendwo anders als SNMP verfügbar.

Beispiele

In diesem Beispiel wird ein 3640 verwendet, der mit der Cisco IOS Software Release 12.2(2)T1 ausgeführt wird. Der verwendete Read-Only (RO) Community String ist öffentlich, und der verwendete Read-Write (RW) Community String ist privat. Weitere Informationen zum Konfigurieren von SNMP-Community-Strings auf Geräten finden Sie unter [Konfigurieren von SNMP-Community-Strings](#).

Diese Ausgabe ist typisch für den Befehl **show ip interface brief**, der im Aktivierungsmodus ausgeführt wird:

```
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-Template1	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

Diese Ausgabe ist das MIB-Objekt, wenn **Descr (.1.3.6.1.2.1.2.2.1.2)** für den vorherigen Router, d. h. eine Textzeichenfolge, die Informationen über die Schnittstelle enthält. Dadurch werden der Schnittstellenname und die Beschreibung wie angegeben angegeben, wobei die vorherige Befehlsausgabe der CLI verwendet wird. **ifName (.1.3.6.1.2.1.31.1.1.1)** kann auch verwendet werden, aber **ifDescr** gibt die Schnittstellenbeschreibung zusammen mit dem Namen an, wobei **ifName** nur den Schnittstellennamen angibt.

```
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-Template1
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
```

Der Wert **ifInDiscards** ist für alle Schnittstellen dieses Routers gleich Null. Wenn Sie dies mit dem CLI-Ergebnis des Befehls **show interfaces fastEthernet 0/0** vergleichen, bestätigt dies das Ergebnis:

```
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
```

Diese Ausgabe zeigt, dass nur ein Eingabefehler für die Schnittstelle vorhanden ist. **interface.ifTable.ifEntry.ifInErrors.7 = Counter32: 1**. Um zu bestimmen, welche Schnittstelle diese ist, vergleichen Sie sie mit der Ausgabe von **ifDescr** oben, die anzeigt, dass diese von **interfaces.ifTable.ifEntry.ifDescr.7 = Serial0/1** stammt. Führen Sie jetzt den Befehl **show interfaces serial 0/1** im privilegierten Modus aus, um das vorherige Ergebnis zu überprüfen:

```
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
interfaces.ifTable.ifEntry.ifOutOctets.35 = Counter32: 0
```

Wenn Sie das vorherige Ergebnis mit der Ausgabe von `ifDescr` vergleichen, bedeutet dies: `interfaces.ifTable.ifEntry.ifOutOctets.1 = Counter32: 98` entspricht `interfaces.ifTable.ifEntry.ifDescr.1 = ATM1/0` `interfaces.ifTable.ifEntry.ifOutOctets.3 = Counter32: 765470674` entspricht `interfaces.ifTable.ifEntry.ifDescr.3 = FastEthernet0/0` `interfaces.ifTable.ifEntry.ifOutOctets.21 = Counter32: 98` entspricht `interfaces.ifTable.ifEntry.ifDescr.21 = ATM1/0.109-aal5` layer Dies ist die Ausgabe des CLI-Befehls `show interfaces` für jede der vorherigen Schnittstellen, die im Aktivierungsmodus ausgeführt wurden:

```
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 lwld, 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
```

Die Ausgabe des **ifOutOctets** stimmt nicht mit der CLI-Ausgabe für den Befehl **show interfaces FastEthernet 0/0** überein, ist jedoch ähnlich. Dies liegt daran, dass es zu einer Verzögerung kommen kann, wenn die Schnittstellen abgefragt werden und der CLI-Befehl ausgeführt wird.

```
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
```

Der Wert ifOutDiscards ist für alle Schnittstellen null. Mit dem Befehl show interfaces fastEthernet 0/0 erzeugt dieser Befehl dieses Ergebnis:

```
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
```

Der ifOutErrors ist für alle Schnittstellen gleich Null. Mit dem Befehl show interfaces fastEthernet 0/0 erzeugt dieser Befehl dieses Ergebnis:

```
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
```

Der ifOutQLen ist für alle Schnittstellen gleich Null. Mit dem Befehl **show interfaces fastEthernet 0/0** als Beispiel:

```
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

```

F. Welche Beziehung besteht zwischen den `show interfaces`-Anweisungen ohne Puffer und Verwerfungen bei Eingabewarteschlangen? Warum gibt die InDiscards von SNMP keine Puffer-Anzahl und keine Eingabe-Warteschlange verloren, während die OutDiscards von SNMP Ausgabewarteschlangen-Drops verursachen?

Antwort: `locfInputQueueDrops/ifInDiscards` arbeiten anders als `locfOutputQueueDrops/ifOutDiscards`. Die `ifInDiscards` zählen die Anzahl der Pakete, die weggeworfen werden, wenn eine Systemressource wie ein Puffer fehlt. Dies ist in der Regel eine Teilmenge der `locfInputQueueDrops`. Oft sehen Sie `locfInputQueueDrops = ifInDiscards`. `locfInputQueueDrops` zählt jedoch auch die Anzahl der verworfenen Pakete, da sie die Grenze für die Eingabewarteschlange erreichen. Im Allgemeinen sehen Sie `locfInputQueueDrops > ifInDiscards`.

Zusammenfassung

`locfInputQueueDrops = Queue Limit Drops + No Buffer Drops`
`ifInDiscards = No Buffer Drops` (und ist eine Teilmenge von `locfInputQueueDrops`)

Die `locfOutputQueueDrops`-Methode und `ifOutDiscards` sind immer gleich, wenn sie dieselben Ereignisse zählen. Diese Ereignisse erreichen das Grenzwert für die Ausgabewarteschlange und verfügen nicht über einen Hardware-Tx-Puffer, wenn ein Paket von einer Schnittstelle auf eine andere schnell umgeschaltet wird. Die OIDs der vorherigen MIB-Objekte sind:

Von OLD-CISCO-INTERFACES-MIB (nur registrierte Kunden)	Von RFC1213-MIB (nur registrierte Kunden)
<code>locfInputQueueDrops = .1.3.6.1.4.1.9.2.2.1.1.26</code>	<code>ifInDiscards = 0,1.3.6.1.2.1.2.2.1.13</code>
<code>locfOutputQueueDrops = 0,1.3.6.1.4.1.9.2.2.1.1.27</code>	<code>ifOutDiscards = 0,1.3.6.1.2.1.2.2.1.19</code>

F. Kann ich auf einem Router keine Puffer abfragen?

Antwort: Ja. Sie können nach `ifInDiscards` fragen, um keine Puffer abzufragen.

F. Wie kann ich Warteschlangenbegrenzungen auf einem Router abfragen?

Antwort: Bei der Verwendung von SNMP kann der Befehl `show interfaces` die einzelnen Elemente, die in die Ausgabepfähle fallen, nicht auseinanderbrechen.

Stellen Sie sich die folgenden neuen Informationen über den Ausgabeverwerfungszähler vor:

Eingabe-Drops = Warteschlangenlimit-Drops + Throttling-Drops + vollständige Verwerfen der SMT-Warteschlange + RSRB-Verwerfen + keine Pufferübergänge

Außerdem werden SNMP-Zähler nie gelöscht, selbst wenn die Schnittstellen gelöscht werden.

[Zugehörige Informationen](#)

- [Technische Tipps zu IP-Anwendungsservices](#)
- [Technischer Support und Dokumentation für Cisco Systeme](#)