

Configurez la gamme Cisco D9800 avec la Surveillance de réseau simple Protocol

Contenu

[Introduction](#)

[Conditions préalables](#)

[Conditions requises](#)

[Composants utilisés](#)

[Informations générales](#)

[Configurez](#)

[Sous-arbre de accord du MIB CISCO-DMN-DSG-ROOT](#)

[Exemple de configuration du débit symbole de la Manche RF1](#)

[Exemple de configuration avec la validation](#)

[Illustration](#)

[Conclusion](#)

Introduction

Ce document décrit un exemple sur la façon dont configurer un récepteur de Cisco D9859 (utilisé comme exemple) utilisant la Surveillance de réseau simple Protocol (SNMP).

Conditions préalables

Conditions requises

Cisco vous recommande de prendre connaissance des rubriques suivantes :

- L'application Net-SNMP (snmpwalk, snmpget, snmpset, snmptranslate)
- Un ou plusieurs récepteurs de Cisco D9859 (pour cet exemple)

[Composants utilisés](#)

Les informations dans ce document sont basées sur Cisco D9859 sur la version v1.51 utilisant le D9859_Feature_MIB_V1.50.

Les informations contenues dans ce document ont été créées à partir des périphériques d'un environnement de laboratoire spécifique. Tous les périphériques utilisés dans ce document ont démarré avec une configuration effacée (par défaut). Si votre réseau est opérationnel, assurez-vous que vous comprenez l'effet potentiel de toute commande.

[Informations générales](#)

Cet exemple se concentre sur changer le débit symbole de l'entrée RF1. Les mêmes commandes peuvent être utilisées pour changer d'autres paramètres de la configuration. Les identifiants d'objet (OID) ont été obtenus en parcourant Cisco D9859_Feature_MIB disponible sur Cisco.com chez

https://software.cisco.com/download/release.html?mdfid=285007629&softwareid=282766520&release=D9859_MIB%2001.50.00.

D'ailleurs, on le destine pour des opérateurs réseau qui veut configurer leurs récepteurs de Cisco D98xx avec le SNMP et est au courant de la ligne de commandes SNMP aux commandes et sait configurer des récepteurs de Cisco un D98xx.

Configurez

Le châssis de Cisco D9859 peut être configuré par l'intermédiaire du SNMP utilisant le D9859_Feature_MIB. Le MIB n'est pas chargé dans le navigateur SNMP sur Cisco.com. Vous pouvez obtenir un aperçu du MIB à l'aide de la commande de snmptranslate :

Remarque: **CISCO_DMN_DSG_Root.mib** est la racine de l'arborescence comme suggéré par le nom. Utilisez l'ancre - m. /<root_file:/ <child_file pour visualiser le contenu.

Sous-arbre de accord du MIB CISCO-DMN-DSG-ROOT

```
[D9859_Feature_MIB_v1.50 fprosman]$ snmptranslate -m
./CISCO_DMN_DSG_Root.mib:./CISCO_DMN_DSG_Tuning.mib -Pu -Tp
+--iso(1)
|
+--org(3)
|
+--dod(6)
|
+--internet(1)
|
+--directory(1)
|
+--mgmt(2)
| |
| +--mib-2(1)
| |
| | +--transmission(10)
| |
| +--experimental(3)
|
+--private(4)
|
| +--enterprises(1)
| |
| | +--ciscoSPVTG(1429)
| | |
| | | +--ciscoSat(2)
| | | |
| | | | +--ciscoDMN(2)
| | | | |
| | | | | +--ciscoDSGUtilities(5)
```


				Values: shortFrame(1), longFrame(2),
notApplicable(3)				
				+++ -R-- String satSignalCnMarginRaw(33)
				Textual Convention: DisplayString
				Size: 0..17
				+++ -R-- String satSignalDvbSQpskErrCount(34)
				Textual Convention: DisplayString
				Size: 0..8
				+++ -R-- String satSignalDvbS2LdpcErrCount(35)
				Textual Convention: DisplayString
				Size: 0..8
				+++ -R-- String satSignalPvErrCount(36)
				Textual Convention: DisplayString
				Size: 0..8
				+++ -R-- EnumVal satSignalFecSyncStatus(37)
				Values: no(1), yes(2)
				+++ -R-- String satSignalPktErrCount(38)
				Textual Convention: DisplayString
				Size: 0..10
				+++inputStatusTable(2)
				+++inputStatusEntry(1)
				Index: inputStatusIndex
				+++ ---- Integer32 inputStatusIndex(1)
				Range: 1
				+++ -R-- EnumVal inputStatusCurInput(2)
				Values: rf(1)
				+++ -R-- EnumVal inputStatusSatLock(3)
				Values: nolock(1), lockminussignal(2),
lockplussignal(3)				
				+++ -R-- EnumVal inputStatusMpgIpLock(4)
				Values: nolock(1), lock(2)
				+++ -R-- String inputStatusInputRate(5)
				Textual Convention: DisplayString
				Size: 0..32
				+++ -R-- String inputStatusNetworkName(6)
				Textual Convention: DisplayString
				Size: 0..64
				+++ -R-- String inputStatusNetworkId(7)
				Textual Convention: DisplayString
				Size: 0..32
				+++ -R-- String inputStatusTransportId(8)
				Textual Convention: DisplayString
				Size: 0..32
				+++ -R-- EnumVal inputStatusScramblingMode(9)
				Values: unknown(1), des(2), dvb(3), biss1(4),
biss2(5), biss3(6)				
				+++ -R-- EnumVal inputStatusTransportError(10)
				Values: notApplicable(1), ok(2), error(3)
				+++ -R-- EnumVal inputStatusAsiLock(11)
				Values: nolock(1), lock(2)
				+++ -R-- EnumVal inputStatusAsiLinkError(12)
				Values: notApplicable(1), ok(2), error(3)
				+++ -R-- EnumVal inputStatusAsiPacketSize(13)
				Values: notApplicable(1),
oneHundredAndEightyEight(2), twoHundredAndFour(3)				
				+++ -R-- String inputStatusLastTuneReason(14)
				Textual Convention: DisplayString
				Size: 0..16
				+++ -R-- String inputStatusCurD985xInput(15)
				Textual Convention: DisplayString
				Size: 0..16

```

+--- -R-- String    inputStatusIpiLinkStatus(16)
|
|    Textual Convention: DisplayString
|    Size: 0..16
+--- -R-- String    inputStatusIpiSignal(17)
|
|    Textual Convention: DisplayString
|    Size: 0..16
+--- -R-- String    inputStatusIpiFecLock(18)
|
|    Textual Convention: DisplayString
|    Size: 0..16
+--- -R-- String    inputStatusIpiPcrLock(19)
|
|    Textual Convention: DisplayString
|    Size: 0..16
+--- -R-- String    inputStatusIpiDelLatency(20)
|
|    Textual Convention: DisplayString
|    Size: 0..16
+--- -R-- IpAddr    inputStatusIpiData1SrcIP(21)
+--- -R-- IpAddr    inputStatusIpiData2SrcIP(22)
+--- -R-- String    inputStatusIpiData1TsType(23)
|
|    Textual Convention: DisplayString
|    Size: 0..16
+--- -R-- String    inputStatusIpiData2TsType(24)
|
|    Textual Convention: DisplayString
|    Size: 0..16
+---siRcvTable(4)
|
+---siRcvOptionTable(1)
|
|    +---siRcvOptionEntry(1)
|    |    Index: siRcvOptionInstance
|    |
|    |    +--- ---- Integer32 siRcvOptionInstance(1)
|    |    |    Range: 1
|    |    +--- -RW- EnumVal    siRcvOptionAcqMode(2)
|    |    |    Values: basic(1), auto(2), custom(3)
|    |    +--- -RW- EnumVal    siRcvOptionReacq(3)
|    |    |    Values: writeOnly(1), yes(2)
|    |    +--- -RW- Integer32 siRcvOptionNetID(4)
|    |    |    Range: 0..65535
|    |    +--- -RW- EnumVal    siRcvOptionInputSel(5)
|    |    |    Values: userCfg(1), swMap(2)
|    |    +--- -RW- EnumVal    siRcvOptionFreqSel(6)
|    |    |    Values: nit(1), userCfg(2)
|    |    +--- -RW- EnumVal    siRcvOptionServListMode(7)
|    |    |    Values: rigorous(1), degraded(2)
|    |    +--- -R-- EnumVal    siRcvOptionUseBAT(8)
|    |    |    Values: no(1), yes(2)
|    |    +--- -RW- EnumVal    siRcvOptionUseNIT(9)
|    |    |    Values: no(1), yes(2)
|    |    +--- -RW- EnumVal    siRcvOptionUseSDT(10)
|    |    |    Values: no(1), yes(2)
|    |    +--- -RW- EnumVal    siRcvOptionUsePAT(11)
|    |    |    Values: no(1), yes(2)
|    |
|    +---siRcvOptionStatusTable(2)
|    |
|    |    +---siRcvOptionStatusEntry(1)
|    |    |    Index: siRcvOptionStatusInstance
|    |    |
|    |    |    +--- ---- Integer32 siRcvOptionStatusInstance(1)
|    |    |    |    Range: 1
|    |    |    +--- -R-- EnumVal    siRcvOptionLastChanReas(2)
|    |    |    |    Values: nit(1), uplinkForceRetune(2),

```

userEntry(3), preset(4)

```

+--- -R-- String      siRcvOptionLastActivated(3)
|
| Textual Convention: DisplayString
|
| Size: 0..32
+--- -R-- EnumVal    siRcvOptionStatusFreqSel(4)
|
| Values: nit(1), userCfg(2)
+--- -R-- EnumVal    siRcvOptionStatusServListMode(5)
|
| Values: rigorous(1), degraded(2)
+--- -R-- EnumVal    siRcvOptionStatusUseBAT(6)
|
| Values: no(1), yes(2)
+--- -R-- EnumVal    siRcvOptionStatusUseNIT(7)
|
| Values: no(1), yes(2)
+--- -R-- EnumVal    siRcvOptionStatusUseSDT(8)
|
| Values: no(1), yes(2)
+--- -R-- EnumVal    siRcvOptionStatusUsePAT(9)
|
| Values: no(1), yes(2)
+---siInfoRxTable(3)
|
| +---siInfoRxEntry(1)
|   | Index: siInfoRxInstance, siInfoRxIdx
|   |
|   +--- ---- Integer32 siInfoRxInstance(1)
|   |   Range: 1
|   +--- ---- Integer32 siInfoRxIdx(2)
|   |   Range: 1..65535
|   +--- -R-- EnumVal    siInfoRxType(3)
|   |   Values: pat(1), cat(2), pmt(3), tsdt(4),
nit(5), nitother(6), sdt(7), sdtother(8), bat(9), aeitpf(10), oeitpf(11), aeitES0(12),
aeitES1(13), oeitES(14), tdt(15), rst(16), st(17), tot(18), dit(19), sit(20), ecmodd(21),
ecmeven(22), emm(23), mpe(24), dpi(25), drt(26), cdt(27), mct(28), mat(29), mit(30), ect(31),
invalidtableid(32)
|
| +--- -R-- String      siInfoRxIDExt(4)
|   | Textual Convention: DisplayString
|   | Size: 0..32
+--- -R-- String      siInfoRxUid(5)
|
| Textual Convention: DisplayString
|
| Size: 0..32
+--- -R-- EnumVal    siInfoRxStatus(6)
|
| Values: none(1), partial(2), full(3),
update(4), timeout(5), lost(6)
|
| +--- -R-- String      siInfoRxVer(7)
|   | Textual Convention: DisplayString
|   | Size: 0..32
+--- -R-- String      siInfoRxPID(8)
|
| Textual Convention: DisplayString
|
| Size: 0..32
+--- -R-- String      siInfoRxSections(9)
|
| Textual Convention: DisplayString
|
| Size: 0..32
+---siInfoTsTable(4)
|
| +---siInfoTsEntry(1)
|   | Index: siInfoTsInstance, siInfoTsIdx
|   |
|   +--- ---- Integer32 siInfoTsInstance(1)
|   |   Range: 1
|   +--- ---- Integer32 siInfoTsIdx(2)
|   |   Range: 1..256
|   +--- -R-- String    siInfoTsId(3)
|   |   Textual Convention: DisplayString
|   |   Size: 0..32
+--- -R-- String      siInfoTsFreq(4)
|
| Textual Convention: DisplayString

```


Exemple de configuration du débit symbole de la Manche RF1

Dans cet exemple, vous changez le débit symbole pour la première entrée rf du récepteur. Notez cela qui émet un SNMP SET sur Cisco D9859 déclenchera un verrouillage de la session de configuration. En conséquence, la modification devra être commise pour être validée dans l'interface et libérer pour verrouiller sur la session. Le verrouillage empêchent également une modification de configuration utilisant l'interface web pendant environ 5 minutes.

Exemple de configuration avec la validation

1. Lisez la configuration en cours utilisant le snmpget.
2. Établissement d'une configuration utilisant la commande de snmpset.
3. Validez l'état actuel utilisant le SNMP.
4. État actuel de Validatethe dans l'interface web.
5. Soumission de la modification utilisant le SNMP.
6. Validant l'état dans l'interface web de nouveau.

Illustration

Étape 1. Lisez la configuration en cours utilisant la commande de SNMP GET, le débit symbole est accessible à l'OID .1.3.6.1.4.1.1429.2.2.5.5.2.1.1.4.1

```
[D9859_Feature_MIB_v1.50 fprosman]$ snmpget -v2c -c public "IP address"  
1.3.6.1.4.1.1429.2.2.5.5.2.1.1.4.1 SNMPv2-SMI::enterprises.1429.2.2.5.5.2.1.1.4.1 = INTEGER:  
28000
```

Étape 2. Établissement de la nouvelle configuration.

Changer antérieurement la configuration que nous devons d'abord commuter au bon canal rf dans la liste de saisie, cette exécution est réalisent en plaçant une valeur entière sur l'OID .1.3.6.1.4.1.1429.2.2.5.5.1.1.0 :

Le Management Information Base SNMP (MIB) définit ce qui suit :

- 1 = ASI (interface de série asynchrone)
- 2 = RF1
- 3 = RF2
- 4 = RF3
- 5 = RF4
- 6 = IPI (IP entré)
- 255 = aucun

Dans ce cas le RF1 a été sélectionné :

```
[D9859_Feature_MIB_v1.50 fprosman]$ snmpget -v2c -c public "IP address"
.1.3.6.1.4.1.1429.2.2.5.5.1.1.0 SNMPv2-SMI::enterprises.1429.2.2.5.5.1.1.0 = INTEGER: 2 <<<<<< 2
indicates RF1 (the values are explained under activeTuningInput of the CISCO_DMN_DSG_Tuning.mib
file)
```

Établissement de la modification :

```
[D9859_Feature_MIB_v1.50fprosman]$ snmpset -v2c -c public "IP Address"
1.3.6.1.4.1.1429.2.2.5.5.2.1.1.4.1 i 27000 SNMPv2-SMI::enterprises.1429.2.2.5.5.2.1.1.4.1 =
INTEGER: 27000
```

Étape 3. Validez l'état actuel utilisant le SNMP.

C'est où nous pouvons être fallacieux, issueing un OBTENIR après que le POSITIONNEMENT affiche la valeur étant changée :

```
[D9859_Feature_MIB_v1.50fprosman]$ snmpget -v2c -c public "IP address"
1.3.6.1.4.1.1429.2.2.5.5.2.1.1.4.1 SNMPv2-SMI::enterprises.1429.2.2.5.5.2.1.1.4.1 = INTEGER:
27000
```

Étape 4. Validatie l'état actuel dans l'interface web.

Suivant les indications de l'image, vous pouvez voir que le Web UI ne reflète pas la modification :

▼ Tuning

Tuning	
Downlink Frequency	<input type="text" value="11.2"/> (GHz)
Symbol Rate	<input type="text" value="2.8"/> (MS/s)
FEC	<input type="button" value="Auto"/> ↕

Modification non engagée

Vous devez demander au périphérique pour valider la modification en plaçant un entier utilisant la commande de snmp get.

Le MIB définit ceci :

- 1 = sauvegarde et fin
- 2 = ignorent et se ferment
- 3 = écrivent seulement

Étape 5. Sauvegardez la modification :

```
[D9859_Feature_MIB_v1.50 fprosman]$ snmpset -v2c -c public "IP Address"
.1.3.6.1.4.1.1429.2.2.5.6.2.0 i 1 SNMPv2-SMI::enterprises.1429.2.2.5.6.2.0 = INTEGER: 1
```

Étape 6. Une deuxième validation sur l'exposition du Web UI que la bonne valeur s'est reflétée.

▼ Tuning

Tuning	
Downlink Frequency	11.2 (GHz)
Symbol Rate	2.7 (MS/s)
FEC	Auto ↕
Modulation	DVB-S ↕

Modification commise

Conclusion

Cet exemple affiche que cela configurer les récepteurs de Cisco D9859 par le SNMP est possible mais ce des modifications doivent être commises pour être imposées.