How to use SNMP to set IP ACL on DGS-3100

■ **CLI command:**

> create access_profile profile_id 1 ip source_ip_mask 255.255.255.255

■ **SNMP Command:**

Step 1) before next step, use the following OID (from MIB "*qosclimib.mib*" -> "*rlQosFreeIndexesValue.1*") to get free

"**acl**" **index,** for this time the free index is "**3**":

> snmpwalk -v2c -c private 10.90.90.90 1.3.6.1.4.1.171.10.94.89.89.88.25.1.2.**1**
>
> SNMPv2-SMI::enterprises.171.10.94.89.89.88.25.1.2.1 = INTEGER: **3**

Step 2) using the MIB "*qosclimib.mib*" -> "*rlQosTupleTable*", create and go an ACL profile with L3 IPv4 +

Source_IP_ Mask 255.255.255.255, (but not finish the creating yet, you need to finish the whole step 1~4):

> snmpset -v2c -c private 10.90.90.90 1.3.6.1.4.1.171.10.94.89.89.88.5.1.2.**3** i **2**
>
> 1.3.6.1.4.1.171.10.94.89.89.88.5.1.4.**3** x **0000000000000000**
>
> 1.3.6.1.4.1.171.10.94.89.89.88.5.1.5.**3** i **4**

2 : ip-src (2)

4: createAndGo (4)

3: the key value is used for creating an entry; it's a free ACL index from step 1

1.3.6.1.4.1.171.10.94.89.89.88.5.1.4 = using for the type with masks (Octectstring), see detail at NOTE 1.

NOTE 1:

For "*rlQosTupleTable*", there are two tuple values; the actual mask value is stored in rlQosTupleValue1 or

rlQosTupleValue2 according to type:

1) rlQosTupleValue1 : 1.3.6.1.4.1.171.10.94.89.89.88.5.1.3

Value1 is defined as an integer and it is used to configure integer tuples

2) rlQosTupleValue2 : 1.3.6.1.4.1.171.10.94.89.89.88.5.1.4

Value2 is defined as octet string and it is used for tuples with masks (for example mac address + wildcard).

Step 3) using the MIB "*qosclimib.mib*" -> "*rlQosAclTable*", to create and go the ACL index and type:

> snmpset -v2c -c private 10.90.90.90 1.3.6.1.4.1.171.10.94.89.89.88.7.1.3.**100001** i 2
>
> 1.3.6.1.4.1.171.10.94.89.89.88.7.1.2.**100001** x **41434c31**
>
> 1.3.6.1.4.1.171.10.94.89.89.88.7.1.4.**100001** i 4

100001: means ACL1

41434c31: The hex of "ACL1"

Step 4) before next step, use the following OID (from MIB "*qosclimib.mib*" -> "*rlQosFreeIndexesValue.3*") to get free

"**acl**" **index**, the free index is "**40**":

> snmpwalk -v2c -c private 10.90.90.90 1.3.6.1.4.1.171.10.94.89.89.88.25.1.2
>
> SNMPv2-SMI::enterprises.171.10.94.89.89.88.25.1.2.3 = INTEGER: **40**

Step 5) Final step, using the MIB "*qosclimib.mib*" -> "*rlQosAceTidxTable*", to set the command with the free index value from step 3:

```
snmpset -v2c -c private 10.90.90.90 1.3.6.1.4.1.171.10.94.89.89.88.31.1.3.100001.40 i 1
1.3.6.1.4.1.171.10.94.89.89.88.31.1.4.100001.40 i 1
1.3.6.1.4.1.171.10.94.89.89.88.31.1.5.100001.40 i 0
1.3.6.1.4.1.171.10.94.89.89.88.31.1.6.100001.40 i 3
1.3.6.1.4.1.171.10.94.89.89.88.31.1.14.100001.40 i 4
```

100001: means ACL1

40: free "acl" index number from step 4.

3: the same value from step1, which stand for the free ACL index of this example.

1.3.6.1.4.1.171.10.94.89.89.88.31.1.5 = using when creates ACL profile type is "IP (0)"; see the detail at NOTE 2.

NOTE 2:

 For "*rlQosAceTidxTable*", there are 8 TidxTuple values; the index pointed in *rlQosAceTidxTuple<n>* should match *rlQosTupleIndex* (see NOTE 1), and separate into two parts according to *rlQosAclType*:

  1) If rlQosAclType = 'mac' :

   rlQosAceTidxTuple1 (1.3.6.1.4.1.171.10.94.89.89.88.31.1.5) = Source Mac Mask

   rlQosAceTidxTuple2 (1.3.6.1.4.1.171.10.94.89.89.88.31.1.6) = Destination Mac Mask

   rlQosAceTidxTuple3 (1.3.6.1.4.1.171.10.94.89.89.88.31.1.7) = VLAN

   rlQosAceTidxTuple4 (1.3.6.1.4.1.171.10.94.89.89.88.31.1.8) = 802.1p

   rlQosAceTidxTuple5 (1.3.6.1.4.1.171.10.94.89.89.88.31.1.9) = Ethernet Type

  2) If rlQosAclType = 'IP'

   rlQosAceTidxTuple1 (1.3.6.1.4.1.171.10.94.89.89.88.31.1.5) = according to ACE type (IP, TCP, UDP, ICMP, IGMP, IPv6, etc…).

   rlQosAceTidxTuple2 (1.3.6.1.4.1.171.10.94.89.89.88.31.1.6) = Source IP Mask

   rlQosAceTidxTuple3 (1.3.6.1.4.1.171.10.94.89.89.88.31.1.7) = Destination IP Mask if ICMP or IGMP, else Source Port Mask.

   rlQosAceTidxTuple4 (1.3.6.1.4.1.171.10.94.89.89.88.31.1.8) = ICMP/IGMP Type if ICMP or IGMP, else Destination IP Mask.

   rlQosAceTidxTuple5 (1.3.6.1.4.1.171.10.94.89.89.88.31.1.9) = ICMP Code if ICMP or IGMP, else Destination Port Mask.

   rlQosAceTidxTuple6 (1.3.6.1.4.1.171.10.94.89.89.88.31.1.10) = if ACL type is IP then Dscp

   rlQosAceTidxTuple7 (1.3.6.1.4.1.171.10.94.89.89.88.31.1.11) = TCP Flag Mask

- **CLI command:**

  Source IP, Deny :

  > config access profile profile_id 1 add access_id 1 ip source_ip 10.0.192.178 port 4 deny

- **SNMP Command:**

  Step 1) before next step, use the following OID (from MIB "**qosclimib.mib**" -> "**rlQosFreeIndexesValue.1**") to get free

  "acl" index, for this time the free index is "**1**":

  ```
  snmpwalk -v2c -c private 10.90.90.90 1.3.6.1.4.1.171.10.94.89.89.88.25.1.2.1
  SNMPv2-SMI::enterprises.171.10.94.89.89.88.25.1.2.1 = INTEGER: 1
  ```

  Step 2) using MIB "**qosclimib.mib**" -> "**rlQosTupleTable**", to add the new access id 1with **Src.IP 10.0.192.178**

  ```
  snmpset -v2c -c private 10.90.90.90 1.3.6.1.4.1.171.10.94.89.89.88.5.1.2.1 i 2
  1.3.6.1.4.1.171.10.94.89.89.88.5.1.4.1 x 0A00C0B200000000
  1.3.6.1.4.1.171.10.94.89.89.88.5.1.5 1 i 4
  ```

  2 : ip-src (2)

  0A00C0B200000000: Src.IP address 10.0.192.178 in Hex

  4: createAndGo (4)

  1: the key value is used for creating a entry, it's a free ACL index from step 1

  Step 3) using the MIB "**qosclimib.mib**" -> "**rlQosAclTable**", to

  ```
  snmpset -v2c -c private 10.90.90.90 1.3.6.1.4.1.171.10.94.89.89.88.7.1.2.1 x 41434c3131
  1.3.6.1.4.1.171.10.94.89.89.88.7.1.3 1 i 2 1.3.6.1.4.1.171.10.94.89.89.88.7.1.4 1 i 4
  ```

  41434c3131: hex of "ACL11", means access_id 1

  1: corresponds to rule_id, access_id 1

  2: ip(2)

  4: createAndGo (4)

  Step 4) Before next step, use the following OID to get free **"acl" index**, for this time, the free index is "**60**":

  ```
  snmpwalk -v2c -c private 10.90.90.90 1.3.6.1.4.1.171.10.94.89.89.88.25.1.2.3
  SNMPv2-SMI::enterprises.171.10.94.89.89.88.25.1.2.3 = INTEGER: 60
  ```

  Step 5) using MIB "**qosclimib.mib**" -> "**rlQosAceTidxTable**", to set the command with the free index from step 3:

  ```
  snmpset -v2c -c private 10.90.90.90 1.3.6.1.4.1.171.10.94.89.89.88.31.1.4.1.60 i 1
  1.3.6.1.4.1.171.10.94.89.89.88.31.1.5.1.60 i 0
  1.3.6.1.4.1.171.10.94.89.89.88.31.1.6.1.60 i 1
  1.3.6.1.4.1.171.10.94.89.89.88.31.1.3.1.60 i 2
  1.3.6.1.4.1.171.10.94.89.89.88.31.1.14.1.60 i 4
  ```

  60: free "acl" index number from step 4

  1: ip (1)

  1: free "acl" index number from step 1

  2 : deny (2)

4: createAndGo (4)

Step 6) Using another MIB "*rlActionAcl.mib*" -> "*rlActionAclTable*",

```
snmpset -v2c -c private 10.90.90.90 1.3.6.1.4.1.171.10.94.89.89.130.1.1.2.1 x 1000
1.3.6.1.4.1.171.10.94.89.89.130.1.1.3.1 i 1 1.3.6.1.4.1.171.10.94.89.89.130.1.1.4.1 i 0
1.3.6.1.4.1.171.10.94.89.89.130.1.1.5.1 i 0 1.3.6.1.4.1.171.10.94.89.89.130.1.1.6.1 i 4
```

1000: port 4

1: none(1)

4: createAndGo (4)

0: The two fields are not used for this command, so the value is 0