How to configure DISMAN-TRACEROUTE (RFC 2925T) via NETSNMP in DGS-3620 FW 1.02.R026?

The OIDs are defined in "**traceRouteCtlTable**" of "**DISMAN-TRACEROUTE-MIB.mib**".

traceRouteCtlTable (1.3.6.1.2.1.81.1.2)

⇨ Setup the TraceRoute action.

|	
Object name	traceRouteCtlTable
Object ID	1.3.6.1.2.1.81.1.2
Module	DISMAN-TRACEROUTE-MIB
Base syntax	Sequence Of traceRouteCtlEntry
Access	Not_Accessible
Status	Current
Sequence	1:traceRouteCtlOwnerIndex - Octet String
	2:traceRouteCtlTestName - Octet String
	3:traceRouteCtlTargetAddressType - Integer
	4:traceRouteCtlTargetAddress - Octet String
	5:traceRouteCtlByPassRouteTable - Integer
	6:traceRouteCtlDataSize - Gauge
	7:traceRouteCtlTimeOut - Gauge
	8:traceRouteCtlProbesPerHop - Gauge
	9:traceRouteCtlPort - Gauge
	10:traceRouteCtlMaxTtl - Gauge
	11:traceRouteCtlDSField - Gauge
	12:traceRouteCtlSourceAddressType - Integer
	13:traceRouteCtlSourceAddress - Octet String
	14:traceRouteCtlIfIndex - Integer
	15:traceRouteCtlMiscOptions - Octet String
	16:traceRouteCtlMaxFailures - Gauge
	17:traceRouteCtlDontFragment - Integer
	18:traceRouteCtlInitialTtl - Gauge
	19:traceRouteCtlFrequency - Gauge
	20:traceRouteCtlStorageType - Integer
	21:traceRouteCtlAdminStatus - Integer
	22:traceRouteCtlMaxRows - Gauge
	23:traceRouteCtlTrapGeneration - 0
	24:traceRouteCtlDescr - Octet String
	25:traceRouteCtlCreateHopsEntries - Integer
	26:traceRouteCtlType - 0
	27:traceRouteCtlRowStatus - Integer
Parent node	traceRouteObjects
First child	traceRouteCtlEntry
Description	Defines the Remote Operations Traceroute Control Table for providing the capability of invoking traceroute from a remote host. The results of traceroute operations can be stored in the traceRouteResultsTable, traceRouteProbeHistoryTable, and the traceRouteHopsTable.

traceRouteResultsTable (1.3.6.1.2.1.81.1.3)

⇨ Check the TraceRoute status and result.

Object name	traceRouteResultsTable
Object ID	1.3.6.1.2.1.81.1.3
Module	DISMAN-TRACEROUTE-MIB
Base syntax	Sequence Of traceRouteResultsEntry
Access	Not_Accessible
Status	Current
Sequence	1:traceRouteResultsOperStatus - Integer
	2:traceRouteResultsCurHopCount - Gauge
	3:traceRouteResultsCurProbeCount - Gauge
	4:traceRouteResultsIpTgtAddrType - Integer
	5:traceRouteResultsIpTgtAddr - Octet String
	6:traceRouteResultsTestAttempts - Gauge
	7:traceRouteResultsTestSuccesses - Gauge
	8:traceRouteResultsLastGoodPath - Octet String
Parent node	traceRouteObjects
First child	traceRouteResultsEntry
Description	Defines the Remote Operations Traceroute Results Table for keeping track of the status of a traceRouteCtlEntry.
	An entry is added to the traceRouteResultsTable when an traceRouteCtlEntry is started by successful transition of its traceRouteCtlAdminStatus object to enabled(1). An entry is removed from the traceRouteResultsTable when its corresponding traceRouteCtlEntry is deleted.

For example

⇨ **DGS-3620** wants to TraceRoute the PC via SNMP as the topology below:

PC: 10.90.90.100   DGS-3620   10.90.90.90/8
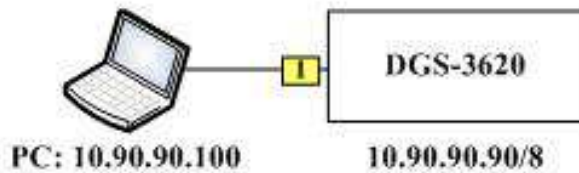
Step 1

⇨ **Create a TraceRoute entry**

We should create an entry name: "**test**" for **traceRouteCtlTable** first. After looking up ASCII Table below, the name "**test**" converts into "**116 101 115 116**"

DEC	HEX	Symbo	DEC	HEX	Symbo	DEC	HEX	Symbo	DEC	HEX	Symbo
32	20	Space	56	38	8	80	50	P	104	68	h
33	21	!	57	39	9	81	51	Q	105	69	i
34	22	"	58	3A	:	82	52	R	106	6A	j
35	23	#	59	3B	;	83	53	S	107	6B	k
36	24	$	60	3C	<	84	54	T	108	6C	l
37	25	%	61	3D	=	85	55	U	109	6D	m
38	26	&	62	3E	>	86	56	V	110	6E	n
39	27	'	63	3F	?	87	57	W	111	6F	o
40	28	(	64	40	@	88	58	X	112	70	p
41	29	)	65	41	A	89	59	Y	113	71	q
42	2A	*	66	42	B	90	5A	Z	114	72	r
43	2B	+	67	43	C	91	5B	[	115	73	s
44	2C	,	68	44	D	92	5C	\	116	74	t
45	2D	-	69	45	E	93	5D	]	117	75	u
46	2E	.	70	46	F	94	5E	^	118	76	v
47	2F	/	71	47	G	95	5F	_	119	77	w
48	30	0	72	48	H	96	60	`	120	78	x
49	31	1	73	49	I	97	61	a	121	79	y
50	32	2	74	4A	J	98	62	b	122	7A	z
51	33	3	75	4B	K	99	63	c	123	7B	{
52	34	4	76	4C	L	100	64	d	124	7C		
53	35	5	77	4D	M	101	65	e	125	7D	}
54	36	6	78	4E	N	102	66	f	126	7E	~
55	37	7	79	4F	O	103	67	g	127	7F	⌂

Then to create an entry via the MIB **traceRouteCtlRowStatus** (1.3.6.1.2.1.81.1.2.1.27)

Object name	traceRouteCtlRowStatus
Object ID	1.3.6.1.2.1.81.1.2.1.27
Module	DISMAN-TRACEROUTE-MIB
Base syntax	Integer
Composed syntax	RowStatus
Access	Read-Create
Status	Current
Value list	1 : active(1)
	2 : notInService(2)
	3 : notReady(3)
	4 : createAndGo(4)
	5 : createAndWait(5)
	6 : destroy(6)

We should assign a index number "1"(ASCII = **49**) to this entry "**test**" (ASCII = **116.101.115.116**), and we have know "test" including "**4**" characters. So the result is:

snmpset -v2c -c private 10.90.90.90 1.3.6.1.2.1.81.1.2.1.27.1.49.4.116.101.115.116 i 4

```
C:\>snmpset -v2c -c private 10.90.90.90 1.3.6.1.2.1.81.1.2.1.27.1.49.4.116.101.1
15.116 i 4
SNMPv2-SMI::mib-2.81.1.2.1.27.1.49.4.116.101.115.116 = INTEGER: 4
```

To check if it is successful by the command:

snmpwalk -v2c -c private 10.90.90.90 1.3.6.1.2.1.81.1.3.1

```
C:\>snmpwalk -v2c -c private 10.90.90.90 1.3.6.1.2.1.81.1.3.1
SNMPv2-SMI::mib-2.81.1.3.1.1.1.49.4.116.101.115.116 = INTEGER: 2
SNMPv2-SMI::mib-2.81.1.3.1.2.1.49.4.116.101.115.116 = Gauge32: 0
SNMPv2-SMI::mib-2.81.1.3.1.3.1.49.4.116.101.115.116 = Gauge32: 0
SNMPv2-SMI::mib-2.81.1.3.1.4.1.49.4.116.101.115.116 = INTEGER: 0
SNMPv2-SMI::mib-2.81.1.3.1.5.1.49.4.116.101.115.116 = ""
SNMPv2-SMI::mib-2.81.1.3.1.6.1.49.4.116.101.115.116 = Gauge32: 0
SNMPv2-SMI::mib-2.81.1.3.1.7.1.49.4.116.101.115.116 = Gauge32: 0
SNMPv2-SMI::mib-2.81.1.3.1.8.1.49.4.116.101.115.116 = Hex-STRING: 00 00 00 00 00
 00 00 00 00 00 00
```

Step 2

⇨ **Specify a Target IP Type**

traceRouteCtlTargetAddressType (1.3.6.1.2.1.81.1.2.1.3)

Object name	traceRouteCtlTargetAddressType
Object ID	1.3.6.1.2.1.81.1.2.1.3
Module	DISMAN-TRACEROUTE-MIB
Base syntax	Integer
Composed syntax	InetAddressType
Access	Read-Create
Status	Current
Value list	1 : unknown(0)
	2 : ipv4(1)
	3 : ipv6(2)
	4 : ipv4z(3)
	5 : ipv6z(4)
	6 : dns(16)
Parent node	traceRouteCtlEntry
First child	None
Description	Specifies the type of host address to be used on the traceroute request at the remote host.

Since PC's IP is 10.90.90.100, we should set the target IP type to "IPv4", the command is:

*snmpset -v2c -c private 10.90.90.90 1.3.6.1.2.1.81.1.2.1.3.1.49.4.116.101.115.116 **i** **1***

```
C:\>snmpset -v2c -c private 10.90.90.90 1.3.6.1.2.1.81.1.2.1.3.1.49.4.116.101.11
5.116 i 1
SNMPv2-SMI::mib-2.81.1.2.1.3.1.49.4.116.101.115.116 = INTEGER: 1
```

To check if it is successful by the command:

snmpwalk -v2c -c private 10.90.90.90 1.3.6.1.2.1.81.1.3.1

```
C:\>snmpwalk -v2c -c private 10.90.90.90 1.3.6.1.2.1.81.1.3.1
SNMPv2-SMI::mib-2.81.1.3.1.1.1.49.4.116.101.115.116 = INTEGER: 2
SNMPv2-SMI::mib-2.81.1.3.1.2.1.49.4.116.101.115.116 = Gauge32: 0
SNMPv2-SMI::mib-2.81.1.3.1.3.1.49.4.116.101.115.116 = Gauge32: 0
SNMPv2-SMI::mib-2.81.1.3.1.4.1.49.4.116.101.115.116 = INTEGER: 1
SNMPv2-SMI::mib-2.81.1.3.1.5.1.49.4.116.101.115.116 = Hex-STRING: 00 00 00 00

SNMPv2-SMI::mib-2.81.1.3.1.6.1.49.4.116.101.115.116 = Gauge32: 0
SNMPv2-SMI::mib-2.81.1.3.1.7.1.49.4.116.101.115.116 = Gauge32: 0
SNMPv2-SMI::mib-2.81.1.3.1.8.1.49.4.116.101.115.116 = Hex-STRING: 00 00 00 00 00
 00 00 00 00 00 00
```

Step 3

⇨ **Specify a Target IP Address**

traceRouteCtlTargetAddress (1.3.6.1.2.1.81.1.2.1.4)

Object name	traceRouteCtlTargetAddress
Object ID	1.3.6.1.2.1.81.1.2.1.4
Module	DISMAN-TRACEROUTE-MIB
Base syntax	Octet String
Composed syntax	InetAddress
Access	Read-Create
Status	Current
Value list	1 : 0..255
Parent node	traceRouteCtlEntry
First child	None
Description	Specifies the host address used on the traceroute request at the remote host. The host address type can be determined by the examining the value of the corresponding traceRouteCtlTargetAddressType index element.

A value for this object MUST be set prior to transitioning its corresponding traceRouteCtlEntry to active(1) via traceRouteCtlRowStatus. |

Firstly, we should convert the decimal IP address (10.90.90.100) into the hexadecimal IP address (0x0a.0x5a.0x5a.0x64), and then configure the IP address as the command:

*snmpset -v2c -c private 10.90.90.90 1.3.6.1.2.1.81.1.2.1.4.1.49.4.116.101.115.116 x **0a5a5a64***

```
C:\>snmpset -v2c -c private 10.90.90.90 1.3.6.1.2.1.81.1.2.1.4.1.49.4.116.101.11
5.116 x 0a5a5a64
SNMPv2-SMI::mib-2.81.1.2.1.4.1.49.4.116.101.115.116 = STRING: "
ZZd"
```

To check if it is successful by the command:

snmpwalk -v2c -c private 10.90.90.90 1.3.6.1.2.1.81.1.3.1

```
C:\>snmpwalk -v2c -c private 10.90.90.90 1.3.6.1.2.1.81.1.3.1
SNMPv2-SMI::mib-2.81.1.3.1.1.1.49.4.116.101.115.116 = INTEGER: 2
SNMPv2-SMI::mib-2.81.1.3.1.2.1.49.4.116.101.115.116 = Gauge32: 0
SNMPv2-SMI::mib-2.81.1.3.1.3.1.49.4.116.101.115.116 = Gauge32: 0
SNMPv2-SMI::mib-2.81.1.3.1.4.1.49.4.116.101.115.116 = INTEGER: 1
SNMPv2-SMI::mib-2.81.1.3.1.5.1.49.4.116.101.115.116 = STRING: "
ZZd"
SNMPv2-SMI::mib-2.81.1.3.1.6.1.49.4.116.101.115.116 = Gauge32: 0
SNMPv2-SMI::mib-2.81.1.3.1.7.1.49.4.116.101.115.116 = Gauge32: 0
SNMPv2-SMI::mib-2.81.1.3.1.8.1.49.4.116.101.115.116 = Hex-STRING: 00 00 00 00 00
 00 00 00 00 00 00
```

(About the string displays "**ZZd**", please refer to **Notice** at **page 8** of this document.)

Step 4

⇨ **DGS-3620 starts to TraceRoute PC (10.90.90.100)**

```
DGS-3620-28SC:admin#traceroute 10.90.90.100

 <10 ms  10.90.90.100

Trace complete.
```

Step 5

⇨ **Enable TraceRoute status before querying the result**

traceRouteCtlAdminStatus (1.3.6.1.2.1.81.1.2.1.21)

Object name	traceRouteCtlAdminStatus
Object ID	1.3.6.1.2.1.81.1.2.1.21
Module	DISMAN-TRACEROUTE-MIB
Base syntax	Integer
Composed syntax	INTEGER
Access	Read-Create
Status	Current
Value list	1 : enabled (1)
	2 : disabled (2)
Parent node	traceRouteCtlEntry
First child	None
Description	Reflects the desired state that an traceRouteCtlEntry should be in:
	enabled (1) - Attempt to activate the test as defined by this traceRouteCtlEntry.
	disabled (2) - Deactivate the test as defined by this traceRouteCtlEntry.
	Refer to the corresponding traceRouteResultsOperStatus to determine the operational state of the test defined by this entry.

To enable it by the command:

*snmpset -v2c -c private 10.90.90.90 1.3.6.1.2.1.81.1.2.1.21.1.49.4.116.101.115.116 **i 1***

```
C:\>snmpset -v2c -c private 10.90.90.90 1.3.6.1.2.1.81.1.2.1.21.1.49.4.116.101.1
15.116 i 1
SNMPv2-SMI::mib-2.81.1.2.1.21.1.49.4.116.101.115.116 = INTEGER: 1
```

Step 6

⇨ **Query the TraceRoute result**

To check the result by the command:

snmpwalk -v2c -c private 10.90.90.90 1.3.6.1.2.1.81.1.3.1

```
C:\>snmpwalk -v2c -c private 10.90.90.90 1.3.6.1.2.1.81.1.3.1
SNMPv2-SMI::mib-2.81.1.3.1.1.1.49.4.116.101.115.116 = INTEGER: 2
SNMPv2-SMI::mib-2.81.1.3.1.2.1.49.4.116.101.115.116 = Gauge32: 1
SNMPv2-SMI::mib-2.81.1.3.1.3.1.49.4.116.101.115.116 = Gauge32: 1
SNMPv2-SMI::mib-2.81.1.3.1.4.1.49.4.116.101.115.116 = INTEGER: 1
SNMPv2-SMI::mib-2.81.1.3.1.5.1.49.4.116.101.115.116 = STRING: "
ZZd"
SNMPv2-SMI::mib-2.81.1.3.1.6.1.49.4.116.101.115.116 = Gauge32: 1
SNMPv2-SMI::mib-2.81.1.3.1.7.1.49.4.116.101.115.116 = Gauge32: 1
SNMPv2-SMI::mib-2.81.1.3.1.8.1.49.4.116.101.115.116 = Hex-STRING: 00 00 00 00 00
  00 00 00 00 00 00
```

Notice

We find there is a problem on Net-SNMP when it displays the Target IP address. Follow the standard (**RFC 4001**), we know to input the Target IP address by the hex value (0x0a5a5a64), but Net-SNMP displays this value by the ASCII code (0a=line feed, 5a=Z, 64=d). Hence we can see this string value is "ZZd". This issue has released on

http://www.net-snmp.org/wiki/index.php/DISMAN-TRACEROUTE-MIB

We also query this value by some MIB Browsers, like **D-View Compiler** and **MG-SOFT**, and all works correctly, **D-View Compiler** (Target IP Address: 10.90.90.100)

MG-SOFT (Target IP Address: 10.90.90.100)