snmpwalk(1) - Linux man page

Name

snmpwalk - retrieve a subtree of management values using SNMP GETNEXT requests

Synopsis

snmpwalk [APPLICATION OPTIONS] [COMMON OPTIONS] [OID]

Description

snmpwalk is an SNMP application that uses SNMP GETNEXT requests to query a network entity for a tree of information.

An object identifier (OID) may be given on the command line. This OID specifies which portion of the object identifier space will be searched using GETNEXT requests. All variables in the subtree below the given OID are queried and their values presented to the user. Each variable name is given in the format specified in ***variables****(5)*.

If no OID argument is present, **snmpwalk** will search the subtree rooted at SNMPv2-SMI::mib-2 (including any MIB object values from other MIB modules, that are defined as lying within this subtree). If the network entity has an error processing the request packet, an error packet will be returned and a message will be shown, helping to pinpoint why the request was malformed.

If the tree search causes attempts to search beyond the end of the MIB, the message "End of MIB" will be displayed.

Options

-Cc

Do not check whether the returned OIDs are increasing. Some agents (LaserJets are an example) return OIDs out of order, but can complete the walk anyway. Other agents return OIDs that are out of order and can cause **snmpwalk** to loop indefinitely. By default, **snmpwalk** tries to detect this behavior and warns you when it hits an agent acting illegally. Use **-Cc** to turn off this check.

-CE {OID}

End the walk at the specified OID, rather than a simple subtree. This can be used to walk a partial subtree, selected columns of a table, or even two or more tables within a single command.

-Ci

Include the given OID in the search range. Normally **snmpwalk** uses GETNEXT requests starting with the OID you specified and returns all results in the MIB subtree rooted at that OID. Sometimes, you may wish to include the OID specified on the command line in the printed results if it is a valid OID in the tree itself. This option lets you do this explicitly.

-CI

In fact, the given OID will be retrieved automatically if the main subtree walk returns no useable values. This allows a walk of a single instance to behave as generally expected, and return the specified instance value. This option turns off this final GET request, so a walk of a single instance will return nothing.

-Cp

Upon completion of the walk, print the number of variables found.

-Ct

Upon completion of the walk, print the total wall-clock time it took to collect the data (in seconds). Note that the timer is started just before the beginning of the data request series and stopped just after it finishes. Most importantly, this means that it does not include snmp library initialization, shutdown, argument processing, and any other overhead.
In addition to these options, **snmpwalk** takes the common options described in the ***snmpcmd***_(1)_ manual page.

Examples

The command:

```
snmpwalk -Os -c public -v 1 zeus system
```

will retrieve all of the variables under system:

```
sysDescr.0 = STRING: "SunOS zeus.net.cmu.edu 4.1.3_U1 1 sun4m"
sysObjectID.0 = OID: enterprises.hp.nm.hpsystem.10.1.1
sysUpTime.0 = Timeticks: (155274552) 17 days, 23:19:05
sysContact.0 = STRING: ""
sysName.0 = STRING: "zeus.net.cmu.edu"
sysLocation.0 = STRING: ""
sysServices.0 = INTEGER: 72
```

(plus the contents of the sysORTable).

The command:

```
snmpwalk -Os -c public -v 1 -CE sysORTable zeus system
```

will retrieve the scalar values, but omit the sysORTable.

See Also

snmpcmd(1), **snmpbulkwalk**(1), **variables**(5).

Referenced By

arpsnmp(8), **mbrowse**(1), **snmpcheck**(1), **snmpget**(1)