咨询服务器 SNMP读写团体字

SNMP　zhiliao_KzPSC　2018-01-09 发表

问题描述

这是在/etc/snmp/snmpd.conf目录下找到的 SNMP配置文件 麻烦问下 哪个关键字为 读和写的关键字

```
##############################################################################
#
# snmpd.conf:
#   An example configuration file for configuring the ucd-snmp snmpd agent.
#
##############################################################################
#
# This file is intended to only be as a starting point.  Many more
# configuration directives exist than are mentioned in this file.  For
# full details, see the snmpd.conf(5) manual page.
#
# All lines beginning with a '#' are comments and are intended for you
# to read.  All other lines are configuration commands for the agent.

##############################################################################
# Access Control
##############################################################################

# As shipped, the snmpd demon will only respond to queries on the
# system mib group until this file is replaced or modified for
# security purposes.  Examples are shown below about how to increase the
# level of access.

# By far, the most common question I get about the agent is "why won't
# it work?", when really it should be "how do I configure the agent to
# allow me to access it?"
#
# By default, the agent responds to the "public" community for read
# only access, if run out of the box without any configuration file in
# place.  The following examples show you other ways of configuring
# the agent so that you can change the community names, and give
# yourself write access to the mib tree as well.
#
# For more information, read the FAQ as well as the snmpd.conf(5)
# manual page.

####
# First, map the community name "public" into a "security name"

#       sec.name  source          community
com2sec notConfigUser  172.18.64.3  ydzcyzt

####
# Second, map the security name into a group name:

#       groupName      securityModel securityName
group   notConfigGroup v1           notConfigUser
group   notConfigGroup v2c           notConfigUser

####
# Third, create a view for us to let the group have rights to:

# Make at least  snmpwalk -v 1 localhost -c public system fast again.
#       name           incl/excl     subtree         mask(optional)
view    systemview    included   .1.3.6.1.2.1.1
view    systemview    included   .1.3.6.1.2.1.25.1.1
view    all           included   .1
####
```

```
###
# Finally, grant the group read-only access to the systemview view.

#       group          context sec.model sec.level prefix read   write  notif
#access  notConfigGroup ""      any       noauth    exact  systemview none none
#access  notConfigGroup ""      any       noauth    exact  mib2 none none
#ccess  notConfigGroup ""      any       noauth    exact  all  none none

# -----------------------------------------------------------------------------

# Here is a commented out example configuration that allows less
# restrictive access.

# YOU SHOULD CHANGE THE "COMMUNITY" TOKEN BELOW TO A NEW KEYWORD ONLY
# KNOWN AT YOUR SITE.  YOU *MUST* CHANGE THE NETWORK TOKEN BELOW TO
# SOMETHING REFLECTING YOUR LOCAL NETWORK ADDRESS SPACE.

#       sec.name  source          community
#com2sec local     localhost       COMMUNITY
#com2sec mynetwork NETWORK/24      COMMUNITY

#     group.name sec.model  sec.name
#group MyRWGroup  any        local
#group MyROGroup  any        mynetwork
#
#group MyRWGroup  any        otherv3user
#...

#           incl/excl subtree                          mask
#view all    included  .1                               80

## -or just the mib2 tree-

#iew mib2   included  .iso.org.dod.internet.mgmt.mib-2 fc


#               context sec.model sec.level prefix read   write  notif
#access MyROGroup ""      any       noauth    0      all    none   none
#access MyRWGroup ""      any       noauth    0      all    all    all


################################################################################
# Sample configuration to make net-snmpd RFC 1213.
# Unfortunately v1 and v2c don't allow any user based authentification, so
# opening up the default config is not an option from a security point.
#
# WARNING: If you uncomment the following lines you allow write access to your
# snmpd daemon from any source! To avoid this use different names for your
# community or split out the write access to a different community and
# restrict it to your local network.
# Also remember to comment the syslocation and syscontact parameters later as
# otherwise they are still read only (see FAQ for net-snmp).


################################################################################
# Sample configuration to make net-snmpd RFC 1213.
# Unfortunately v1 and v2c don't allow any user based authentification, so
# opening up the default config is not an option from a security point.
#
# WARNING: If you uncomment the following lines you allow write access to your
# snmpd daemon from any source! To avoid this use different names for your
# community or split out the write access to a different community and
# restrict it to your local network.
# Also remember to comment the syslocation and syscontact parameters later as
# otherwise they are still read only (see FAQ for net-snmp).
#

# First, map the community name "public" into a "security name"
#       sec.name        source          community
#com2sec notConfigUser   default         public

# Second, map the security name into a group name:
#       groupName       securityModel   securityName
#group   notConfigGroup  v1              notConfigUser
#group   notConfigGroup  v2c             notConfigUser

# Third, create a view for us to let the group have rights to:
# Open up the whole tree for ro, make the RFC 1213 required ones rw.
#       name            incl/excl       subtree mask(optional)
#view    roview          included        .1
#view    rwview          included        system.sysContact
#view    rwview          included        system.sysName
#view    rwview          included        system.sysLocation
#view    rwview          included        interfaces.ifTable.ifEntry.ifAdminStatus
#view    rwview          included        at.atTable.atEntry.atPhysAddress
#view    rwview          included        at.atTable.atEntry.atNetAddress
#view    rwview          included        ip.ipForwarding
#view    rwview          included        ip.ipDefaultTTL
#view    rwview          included        ip.ipRouteTable.ipRouteEntry.ipRouteDest
#view    rwview          included        ip.ipRouteTable.ipRouteEntry.ipRouteIfIndex
#view    rwview          included        ip.ipRouteTable.ipRouteEntry.ipRouteMetric1
#view    rwview          included        ip.ipRouteTable.ipRouteEntry.ipRouteMetric2
#view    rwview          included        ip.ipRouteTable.ipRouteEntry.ipRouteMetric3
#view    rwview          included        ip.ipRouteTable.ipRouteEntry.ipRouteMetric4
#view    rwview          included        ip.ipRouteTable.ipRouteEntry.ipRouteType
#view    rwview          included        ip.ipRouteTable.ipRouteEntry.ipRouteAge
#view    rwview          included        ip.ipRouteTable.ipRouteEntry.ipRouteMask
#view    rwview          included        ip.ipRouteTable.ipRouteEntry.ipRouteMetric5
#view    rwview          included        ip.ipNetToMediaTable.ipNetToMediaEntry.ipNetToMediaIfIndex
#view    rwview          included        ip.ipNetToMediaTable.ipNetToMediaEntry.ipNetToMediaPhysAddress
#view    rwview          included        ip.ipNetToMediaTable.ipNetToMediaEntry.ipNetToMediaNetAddress
#view    rwview          included        ip.ipNetToMediaTable.ipNetToMediaEntry.ipNetToMediaType
#view    rwview          included        tcp.tcpConnTable.tcpConnEntry.tcpConnState
#view    rwview          included        egp.egpNeighTable.egpNeighEntry.egpNeighEventTrigger
#view    rwview          included        snmp.snmpEnableAuthenTraps
```

```
# Finally, grant the group read-only access to the systemview view.
#       group          context sec.model sec.level prefix read   write  notif
#access  notConfigGroup ""      any       noauth    exact  roview rwview none



###############################################################################
# System contact information
#

# It is also possible to set the sysContact and sysLocation system
# variables through the snmpd.conf file:

syslocation Unknown (edit /etc/snmp/snmpd.conf)
syscontact Root <root@localhost> (configure /etc/snmp/snmp.local.conf)

# Example output of snmpwalk:
#   % snmpwalk -v 1 localhost -c public system
#   system.sysDescr.0 = "SunOS name sun4c"
#   system.sysObjectID.0 = OID: enterprises.ucdavis.ucdSnmpAgent.sunos4
#   system.sysUpTime.0 = Timeticks: (595637548) 68 days, 22:32:55
#   system.sysContact.0 = "Me <me@somewhere.org>"
#   system.sysName.0 = "name"
#   system.sysLocation.0 = "Right here, right now."
#   system.sysServices.0 = 72


###############################################################################
# Logging
#

# We do not want annoying "Connection from UDP: " messages in syslog.
# If the following option is commented out, snmpd will print each incoming
# connection, which can be useful for debugging.

dontLogTCPWrappersConnects yes

# ----------------------------------------------------------------------------


###############################################################################
# Process checks.
#
#  The following are examples of how to use the agent to check for
#  processes running on the host.  The syntax looks something like:
#
#  proc NAME [MAX=0] [MIN=0]
#
#  NAME:  the name of the process to check for.  It must match
#         exactly (ie, http will not find httpd processes).
#  MAX:   the maximum number allowed to be running.  Defaults to 0.
#  MIN:   the minimum number to be running.  Defaults to 0.

#
```

```
#
#  Examples (commented out by default):
#

#  Make sure mountd is running
#proc mountd

#  Make sure there are no more than 4 ntalkds running, but 0 is ok too.
#proc ntalkd 4

#  Make sure at least one sendmail, but less than or equal to 10 are running.
#proc sendmail 10 1

#  A snmpwalk of the process mib tree would look something like this:
#
# % snmpwalk -v 1 localhost -c public .1.3.6.1.4.1.2021.2
# enterprises.ucdavis.procTable.prEntry.prIndex.1 = 1
# enterprises.ucdavis.procTable.prEntry.prIndex.2 = 2
# enterprises.ucdavis.procTable.prEntry.prIndex.3 = 3
# enterprises.ucdavis.procTable.prEntry.prNames.1 = "mountd"
# enterprises.ucdavis.procTable.prEntry.prNames.2 = "ntalkd"
# enterprises.ucdavis.procTable.prEntry.prNames.3 = "sendmail"
# enterprises.ucdavis.procTable.prEntry.prMin.1 = 0
# enterprises.ucdavis.procTable.prEntry.prMin.2 = 0
# enterprises.ucdavis.procTable.prEntry.prMin.3 = 1
# enterprises.ucdavis.procTable.prEntry.prMax.1 = 0
# enterprises.ucdavis.procTable.prEntry.prMax.2 = 4
# enterprises.ucdavis.procTable.prEntry.prMax.3 = 10
# enterprises.ucdavis.procTable.prEntry.prCount.1 = 0
# enterprises.ucdavis.procTable.prEntry.prCount.2 = 0
# enterprises.ucdavis.procTable.prEntry.prCount.3 = 1
# enterprises.ucdavis.procTable.prEntry.prErrorFlag.1 = 1
# enterprises.ucdavis.procTable.prEntry.prErrorFlag.2 = 0
# enterprises.ucdavis.procTable.prEntry.prErrorFlag.3 = 0
# enterprises.ucdavis.procTable.prEntry.prErrMessage.1 = "No mountd process running."
# enterprises.ucdavis.procTable.prEntry.prErrMessage.2 = ""
# enterprises.ucdavis.procTable.prEntry.prErrMessage.3 = ""
# enterprises.ucdavis.procTable.prEntry.prErrFix.1 = 0
# enterprises.ucdavis.procTable.prEntry.prErrFix.2 = 0
# enterprises.ucdavis.procTable.prEntry.prErrFix.3 = 0
#
#  Note that the errorFlag for mountd is set to 1 because one is not
#  running (in this case an rpc.mountd is, but thats not good enough),
#  and the ErrMessage tells you what's wrong.  The configuration
#  imposed in the snmpd.conf file is also shown.
#
#  Special Case:  When the min and max numbers are both 0, it assumes
#  you want a max of infinity and a min of 1.
#



#
#  You can also have programs run by the agent that return a single
#  line of output and an exit code.  Here are two examples.
#
#  exec NAME PROGRAM [ARGS ...]
#
#  NAME:     A generic name. The name must be unique for each exec statement.
#  PROGRAM:  The program to run.  Include the path!
#  ARGS:     optional arguments to be passed to the program

# a simple hello world

#exec echotest /bin/echo hello world

# Run a shell script containing:
#
# #!/bin/sh
# echo hello world
# echo hi there
# exit 35
#
# Note:  this has been specifically commented out to prevent
# accidental security holes due to someone else on your system writing
# a /tmp/shtest before you do.  Uncomment to use it.
#
#exec shelltest /bin/sh /tmp/shtest

# Then,
# % snmpwalk -v 1 localhost -c public .1.3.6.1.4.1.2021.8
# enterprises.ucdavis.extTable.extEntry.extIndex.1 = 1
# enterprises.ucdavis.extTable.extEntry.extIndex.2 = 2
# enterprises.ucdavis.extTable.extEntry.extNames.1 = "echotest"
# enterprises.ucdavis.extTable.extEntry.extNames.2 = "shelltest"
# enterprises.ucdavis.extTable.extEntry.extCommand.1 = "/bin/echo hello world"
# enterprises.ucdavis.extTable.extEntry.extCommand.2 = "/bin/sh /tmp/shtest"
# enterprises.ucdavis.extTable.extEntry.extResult.1 = 0
# enterprises.ucdavis.extTable.extEntry.extResult.2 = 35
# enterprises.ucdavis.extTable.extEntry.extOutput.1 = "hello world."
# enterprises.ucdavis.extTable.extEntry.extOutput.2 = "hello world."
# enterprises.ucdavis.extTable.extEntry.extErrFix.1 = 0
# enterprises.ucdavis.extTable.extEntry.extErrFix.2 = 0

# Note that the second line of the /tmp/shtest shell script is cut
# off.  Also note that the exit status of 35 was returned.

# ---------------------------------------------------------------------------
```

```
###############################################################################
# disk checks
#

# The agent can check the amount of available disk space, and make
# sure it is above a set limit.

# disk PATH [MIN=100000]
#
# PATH:  mount path to the disk in question.
# MIN:   Disks with space below this value will have the Mib's errorFlag set.
#        Default value = 100000.

# Check the / partition and make sure it contains at least 10 megs.

#disk / 10000

# % snmpwalk -v 1 localhost -c public .1.3.6.1.4.1.2021.9
# enterprises.ucdavis.diskTable.dskEntry.diskIndex.1 = 0
# enterprises.ucdavis.diskTable.dskEntry.diskPath.1 = "/" Hex: 2F
# enterprises.ucdavis.diskTable.dskEntry.diskDevice.1 = "/dev/dsk/c201d6s0"
# enterprises.ucdavis.diskTable.dskEntry.diskMinimum.1 = 10000
# enterprises.ucdavis.diskTable.dskEntry.diskTotal.1 = 837130
# enterprises.ucdavis.diskTable.dskEntry.diskAvail.1 = 316325
# enterprises.ucdavis.diskTable.dskEntry.diskUsed.1 = 437092
# enterprises.ucdavis.diskTable.dskEntry.diskPercent.1 = 58
# enterprises.ucdavis.diskTable.dskEntry.diskErrorFlag.1 = 0
# enterprises.ucdavis.diskTable.dskEntry.diskErrorMsg.1 = ""

# ---------------------------------------------------------------------------


###############################################################################
# load average checks
#

# load [1MAX=12.0] [5MAX=12.0] [15MAX=12.0]
#
# 1MAX:   If the 1 minute load average is above this limit at query
#         time, the errorFlag will be set.
# 5MAX:   Similar, but for 5 min average.
# 15MAX:  Similar, but for 15 min average.

# Check for loads:
#load 12 14 14


# % snmpwalk -v 1 localhost -c public .1.3.6.1.4.1.2021.10
# enterprises.ucdavis.loadTable.laEntry.loadaveIndex.1 = 1
# enterprises.ucdavis.loadTable.laEntry.loadaveIndex.2 = 2
# enterprises.ucdavis.loadTable.laEntry.loadaveIndex.3 = 3
# enterprises.ucdavis.loadTable.laEntry.loadaveNames.1 = "Load-1"
# enterprises.ucdavis.loadTable.laEntry.loadaveNames.2 = "Load-5"
# enterprises.ucdavis.loadTable.laEntry.loadaveNames.3 = "Load-15"
# enterprises.ucdavis.loadTable.laEntry.loadaveLoad.1 = "0.49" Hex: 30 2E 34 39
# enterprises.ucdavis.loadTable.laEntry.loadaveLoad.2 = "0.31" Hex: 30 2E 33 31
# enterprises.ucdavis.loadTable.laEntry.loadaveLoad.3 = "0.26" Hex: 30 2E 32 36
# enterprises.ucdavis.loadTable.laEntry.loadaveConfig.1 = "12.00"
# enterprises.ucdavis.loadTable.laEntry.loadaveConfig.2 = "14.00"
# enterprises.ucdavis.loadTable.laEntry.loadaveConfig.3 = "14.00"
# enterprises.ucdavis.loadTable.laEntry.loadaveErrorFlag.1 = 0
# enterprises.ucdavis.loadTable.laEntry.loadaveErrorFlag.2 = 0
# enterprises.ucdavis.loadTable.laEntry.loadaveErrorFlag.3 = 0
# enterprises.ucdavis.loadTable.laEntry.loadaveErrMessage.1 = ""
# enterprises.ucdavis.loadTable.laEntry.loadaveErrMessage.2 = ""
# enterprises.ucdavis.loadTable.laEntry.loadaveErrMessage.3 = ""

# ---------------------------------------------------------------------------


###############################################################################
# Extensible sections.
#

# This alleviates the multiple line output problem found in the
# previous executable mib by placing each mib in its own mib table:

# Run a shell script containing:
#
# #!/bin/sh
# echo hello world
# echo hi there
# exit 35
#
# Note:  this has been specifically commented out to prevent
# accidental security holes due to someone else on your system writing
# a /tmp/shtest before you do.  Uncomment to use it.
#
# exec .1.3.6.1.4.1.2021.50 shelltest /bin/sh /tmp/shtest

# % snmpwalk -v 1 localhost -c public .1.3.6.1.4.1.2021.50
# enterprises.ucdavis.50.1.1 = 1
# enterprises.ucdavis.50.2.1 = "shelltest"
# enterprises.ucdavis.50.3.1 = "/bin/sh /tmp/shtest"
# enterprises.ucdavis.50.100.1 = 35
# enterprises.ucdavis.50.101.1 = "hello world."
# enterprises.ucdavis.50.101.2 = "hi there."
# enterprises.ucdavis.50.102.1 = 0
```

```
# Now the Output has grown to two lines, and we can see the 'hi
# there.' output as the second line from our shell script.
#
# Note that you must alter the mib.txt file to be correct if you want
# the .50.* outputs above to change to reasonable text descriptions.

# Other ideas:
#
# exec .1.3.6.1.4.1.2021.51 ps /bin/ps
# exec .1.3.6.1.4.1.2021.52 top /usr/local/bin/top
# exec .1.3.6.1.4.1.2021.53 mailq /usr/bin/mailq

# ----------------------------------------------------------------------------


##############################################################################
# Pass through control.
#

# Usage:
#   pass MIBOID EXEC-COMMAND
#
# This will pass total control of the mib underneath the MIBOID
# portion of the mib to the EXEC-COMMAND.
#
# Note:  You'll have to change the path of the passtest script to your
# source directory or install it in the given location.
#
# Example:  (see the script for details)
#           (commented out here since it requires that you place the
#           script in the right location. (its not installed by default))

# pass .1.3.6.1.4.1.2021.255 /bin/sh /usr/local/local/passtest

# % snmpwalk -v 1 localhost -c public .1.3.6.1.4.1.2021.255
# enterprises.ucdavis.255.1 = "life the universe and everything"
# enterprises.ucdavis.255.2.1 = 42
# enterprises.ucdavis.255.2.2 = OID: 42.42.42
# enterprises.ucdavis.255.3 = Timeticks: (363136200) 42 days, 0:42:42
# enterprises.ucdavis.255.4 = IpAddress: 127.0.0.1
# enterprises.ucdavis.255.5 = 42
# enterprises.ucdavis.255.6 = Gauge: 42
#
# % snmpget -v 1 localhost public .1.3.6.1.4.1.2021.255.5
# enterprises.ucdavis.255.5 = 42
#
# % snmpset -v 1 localhost public .1.3.6.1.4.1.2021.255.1 s "New string"
# enterprises.ucdavis.255.1 = "New string"
#

# For specific usage information, see the man/snmpd.conf.5 manual page
# as well as the local/passtest script used in the above example.
```

解决方法

rwcommunity public 127.0.0.1 ===>读写community

rocommunity public 127.0.0.1 ===>only read community

答案来自于 **风清扬**