Best Practices Guide for eHealth™ Data Validation

Verifying the eHealth Report data is consistent with the values returned by the polled device.

This guide will provide insight into the procedure to ensure that the values reported by the Concord eHealth™ suite are accurate with the data provided by the SNMP device.

Prepared by:
Sheldon Chapman/Jason Normandin
Concord Technical Support
October 8th, 2004
I. Executive Summary

This best practices guide is designed to introduce the new or established eHealth user to the methodology of validating data within eHealth reports. This guide will provide insight into the procedures to resolve what may appear to be an inconsistency in the data on an eHealth report or to ensure that the values reported by Concord eHealth are accurate. The more experienced user should be able to perform this procedure independently and provide the findings to Concord Technical Support if there is some further discrepancy between the values on a report.

II. Basics of the Data Validation Process

How eHealth obtains and reports the device SNMP data

The eHealth poller process obtains the SNMP performance information from a particular device by performing SNMPGET requests on specific OID’s (object identifiers) within the device’s MIB. The OID’s polled by eHealth for each device are defined within the target element’s MTF (MIB translation file) and PCM (pre-compiled MIB) files.

Once the device responds to the eHealth request, eHealth stores the data within the database. The data is stored as the actual value received from the SNMPGET request in the case of a gauge value or as the delta between the current and previous poll in the case of a counter value. When a report is run against that data, eHealth retrieves the raw values directly from the eHealth database for the particular element(s). The raw data is then calculated into the actual eHealth report variables using the calculation assigned to each variable within the MTF file. The calculation results are then displayed within the eHealth report.

How to perform eHealth data validation

Should a potential data validity issue arise, the data that is being provided from the target device’s MIB objects can be verified externally from the eHealth report engine.
The eHealth installation contains a valuable tool which can be used to troubleshoot data validation issues. The nhSnmpTool is a command line interface that allows users to independently poll a device’s MIB object or objects outside of the eHealth polling and reporting mechanisms. The values returned by the nhSnmpTool utility can then be compared to the values returned from the eHealth polling and reporting mechanisms.

III. Running the nhSnmpTool Command

Run the Element Variable and Configuration Reports

The first step in the data validation process is to obtain the correct MIB OID for the element and variable in question as well as the element’s IP address and index values. These values can be obtained from an Element Configuration and Element Variable reports.

The Element Configuration report will provide the user with the correct IP address and index values for the element in question. The index values are vital in ensuring the correct MIB object is being requested and returned by the nhSnmpTool command.

The Element Variable report will provide the user with the correct MIB OID’s and variable calculations to ensure that the values match those of the eHealth report variables. The index values retrieved from the Element Configuration report are then appended to the OID requested based upon the variable definition provided in the Element Variable report.

These two reports can be run from the eHealth Web Interface using the following process:

1. Log into the eHealth Web interface as a user with appropriate permissions for the element in question
2. Navigate to the Organization tab
3. Expand the Element list and select the element in question
4. Select the EC icon in the right frame to run the Element Configuration report or the EV icon to run the Element Variable report.

These two reports can also be run via a command line interface on the eHealth system using the following process:

1. Determine the element name from either the eHealth poller configuration or eHealth report
2. Login as the $NH_USER and source the shell appropriate nethealthrc file (UNIX only)
3. Issue the nhReport command with the appropriate arguments to generate the report:
   a. nhReport –rptType drilldown –rptName ElementConfiguration –subjName <element name> -ascii $NH_HOME/tmp/elementConfiguration.log
   b. nhReport –rptType drilldown –rptName ElementVariables –subjName <element name> -ascii $NH_HOME/tmp/elementVariable.log
**Determine the MIB Objects to request for the variable in question**

The Element Configuration and Variable reports contain the information needed to formulate the SNMPGET request used to validate the eHealth variable.

1. In the Element Configuration report, note the values for the following:
   - Agent IP address
   - Index 1, 2, 3 & 4

2. In the Element Variable report, note the following:
   - Find the expression for the associated Trend variable in question
   - Determine the correct MIB objects for each variable within that expression

For example, if performing data validation for the eHealth ‘Discarded Frames’ variable, the expression would be:

```
ifInDiscards + ifOutDiscards
```

3. We would then determine that the MIB objects for each of those variables are:

```
ifInDiscards 1.3.6.1.2.1.2.2.1.13.$1
ifOutDiscards 1.3.6.1.2.1.2.2.1.19.$1
```

4. The eHealth element indices would then be substituted into the MIB object definition noted above using the following mapping:
   - $1  index 1
   - $2  index 2
   - $3  index 3
   - $4  index 4

In the above example, the MIB objects required to validate the ‘Discarded Frames’ variable would become:

```
ifInDiscards 1.3.6.1.2.1.2.2.1.13.1
ifOutDiscards 1.3.6.1.2.1.2.2.1.19.1
```

**Run the nhSnmpTool command**

We now have all of the necessary information to use the nhSnmpTool command to validate the eHealth variable. We simply now need to substitute the correct value with each argument required by the nhSnmpTool command.

The nhSnmpTool command requires the use of the following arguments to perform data validation:

- **-POLLsa**: Toggle continuous polling of OID area specified with –o
- **-o**: Target the specified MIB object or OID instead of the default targets
  - Only 1 OID can be targeted when using the –POLLsa switch
- **-c**: Community string of device
• \textbf{-s} : Seconds between SNMPGET requests
• \textbf{-p} : Port on which the SNMP agent on the target device resides
• IP Address of the target device

Using the procedure documented above, we now have enough information to perform our SNMPGET requests using the \texttt{nhSnmpTool} command:

\begin{verbatim}
   nhSnmpTool \--c \textless{}community string\textgreater{} \--p \textless{}agent port\textgreater{} \--POLLsa \--o \textless{}target OID\textgreater{} \--s \textless{}seconds\textgreater{} \textless{}IP address\textgreater{}
\end{verbatim}

Using our above example, the command could appear as:

\begin{verbatim}
   nhSnmpTool \--c public \--p 161 \--POLLsa \--o 1.3.6.1.2.1.2.1.13.1 \--s 60 10.150.100.10 > $NH_HOME/tmp/ifInDiscards.txt &
\end{verbatim}

* Note: An out file was used to redirect output for later analysis
* Note: The \& symbol was used to run the command in the background
* Note: If polling multiple MIB objects for a particular eHealth variable, multiple commands will need to be issued using the same procedure as documented above.

Allow the \texttt{nhSnmpTool} commands to run over spanning multiple eHealth polling cycles and initiate the utility as close to the beginning of a poll cycle as possible. It is recommended to allow the command to run for approximately 2 hours to facilitate a sample size large enough to ensure accuracy.

\textbf{IV. Compare the Data Returned by \texttt{nhSnmpTool} to the Report Data}

\textit{Normalize the data returned by \texttt{nhSnmpTool} to the eHealth poll cycle data sample}

Review the output files created by the \texttt{nhSnmpTool} commands to determine the values over a given eHealth poll sample. For example, if running the command at 60 seconds interval, 5 \texttt{nhSnmpTool} samples would need to be evaluated to determine the total values reported during an eHealth poll cycle.

For example, given the following output, we would derive a 5 minute poll sample by subtracting the initial counter value from sample 1 from the final counter value in sample 5. This would give us the total sample delta over the 5 minute period matching the eHealth functionality.

\begin{verbatim}
1.3.6.1.2.1.2.1.13.1 Counter 175342
# 10.150.6.94:1.3.6.1.2.1.2.1.13.1
Poll occurred: 12:54:03

1.3.6.1.2.1.2.1.13.1 Counter 179321
# 10.150.6.94:1.3.6.1.2.1.2.1.13.1
Poll occurred: 12:55:03
\end{verbatim}
1.3.6.1.2.1.2.1.13.1 Counter 189321
# 10.150.6.94:1.3.6.1.2.1.2.1.13.1
Poll occurred: 12:56:03
--------------------------------------------------
1.3.6.1.2.1.2.1.13.1 Counter 190052
# 10.150.6.94:1.3.6.1.2.1.2.1.13.1
Poll occurred: 12:57:03
--------------------------------------------------
1.3.6.1.2.1.2.1.13.1 Counter 190787
# 10.150.6.94:1.3.6.1.2.1.2.1.13.1
Poll occurred: 12:58:03
--------------------------------------------------

We would subtract 175342 from 190787 (=15445) to determine the total delta value for the poll closest to 12:58. During the poll cycle closest to 12:58, eHealth would also subtract the previous MIB object value from the current poll’s MIB object value to determine the delta. This is the value that would be inserted into the database for the object value.

The same procedure would be used to determine the value for each remaining poll cycle within the nhSnmpTool output

**Generate the eHealth report to perform the actual data validation**

An ASCII eHealth Trend report should then be run for the same time frame as the nhSnmpTool analysis for the variable in question. Each data point presented within that Trend report will represent a single poll’s MIB object delta calculation. These values should roughly match the values determined in the nhSnmpTool output analysis.

As the SNMPGET requests performed by the nhSnmpTool command are not issued at the same time as the eHealth poller’s SNMPGET requests, there may be a slight difference in the values reported by nhSnmpTool and the eHealth reports. These slight differences are anticipated and do not signify a variation in data validity.

**Utilize the nhExportData command to perform additional report data validation**

It is often necessary to utilize the nhExportData command to verify the report data from the eHealth reporting engine. The data returned by the nhExportData command will be generated via the same mechanisms as a standard eHealth report but will not be passed.
1. Through the logic that groups samples together into larger data buckets as is the case with group reports
2. Through the graphing engine that actual creates the eHealth report graph
3. Through our data analysis engine that pre-analyzes report data in the case of Health or Service Level Report values.

This makes this utility very useful when evaluating group reports or reports displaying pre-analyzed data.

The nhExportData command will generate an ASCII results file listing the variable values for the requested element(s), group, variable(s) and time range. The resulting ASCII values can then be compared to both the standard eHealth report as well as the device data received directly from the MIB using the nhSnmpTool command.

The nhExportData command requires several basic arguments:

- **subjName**: (Required) Specifies the name of one or more elements or a single group for which to export statistics data.
- **subjType**: Specifies whether you want to export statistics data for one or more elements or for a single group of elements
- **groupType**: Specifies one of the following group types for the elements for which you want to export statistics data: all (default), lanWan, system, response, multiTechnology, routerSwitch, qos, remoteAccess, application.
- **fromDate**: Specifies the starting date for which you want to export statistics data.
- **toDate**: Specifies the ending date for which you want to export data
- **outFile**: Specifies the name of the DDO file to which you want to export the statistics data.
- **vars**: Specifies one or more names of Trend report variables for which you want to export statistics data.

Using these basic arguments you can simple export the data for any group or element(s) using any available Trend variable. It must be noted however that the variables values expected by the vars argument are not the same as the Trend report variable name but are instead the variable symbol identified within the variable.sys or variable.usr file. These files are located within the $NH_HOME/db/data directory and can be used to lookup the symbol from the basic Trend variable name.

For example, to determine the correct symbol values for the ‘Discarded Frames’ Trend variable, we could search the variable.sys file. The symbol value will be contained in the last column of that file.

```
ehealth:gargamel:4 : cat $NH_HOME/db/data/variable.sys | grep "Discarded Frames"
```

```
26| Discarded Frames                |Discarded Frames|discardedFrames
```

As you can see in the above example, the correct symbol value for the ‘Discarded Frames’ Trend variable is ‘discardedFrames’.

So using this example, we could export the ‘Discarded Frames’ variable data for the element router-RH-enet-port-1 element from Jan 1, 2004 until Jan 30th 2004 using the following syntax:
V. What to do if an Actual Discrepancy is Determined

Should the above procedure document a genuine data validity issue, the following information should be provided to Concord Technical Support to perform a technical analysis to verify if a Concord bug may be present or if the cause is a configuration or other issue:

- The nhSnmpTool output files created in section III
- The eHealth ASCII report created in section IV
- The eHealth poller.cfg configuration file or the output of nhExportConfig command
- Any custom MTF or PCM files used by the element in question
- The current eHealth version and patch level
- nhExportData output for the same time frame, element and variable as analyzed
- The output file created by the nhGetSupportInfo utility for the ‘reports’ component
  - nhgetSupportInfo –type “reports”

Once this information has been received by Concord Technical Support, we will review the above information and contact you with any additional troubleshooting steps that need to occur.

VI. Additional Resources

- Concord Communications Technical Knowledgebase solutions (http://search.support.concord.com)
- How to use the nhSnmpTool command
  - PrimusTrain264
- How to determine how eHealth calculates values in reports - eHealth 5.0.2 and later
- TS9830
  - How to determine how eHealth calculates values in reports - eHealth 4.8 and earlier
    - TS8316
- TS12378
  - How to verify polling is occurring for a particular element.
  - TS84
- TS8273
  - How to find the MTF file that relates to the Poller Configuration UI 'Agent Type' field
  - TS84
- TS5367
  - How does eHealth calculate availability
- TS7596
  - How does eHealth calculate latency
  - TS5367
  - How does Concord calculate bandwidth utilization
    - TS7596