RMF XP Implementation Guide

Last Change Date:2017-10-02

Table of Contents

1 Abstract

For installations running operating systems other than z/OS, RMF XP provides a solution to monitor the performance of heterogeneous environments. RMF XP supports the following operating systems:

- AIX on System p
- Linux on System x
- Linux on z System

The customization steps that are required to set up RMF XP in the USS environment under z/OS are described in chapter *Cross platform monitoring with RMF XP* of the *RMF User's Guide*.

RMF XP exploits the existing Common Information Model (CIM) instrumentation for AIX and for the Linux distributions (RHEL/SUSE) and does not require any proprietary agent software on the monitored endpoints. The CIM server, as well as the metric providers, are integral parts of the supported AIX and Linux distributions, and therefore no additional software needs to be installed. However, you need to ensure that the CIM servers with their metric providers are properly set up and running on the monitored endpoints. This document helps you in configuring the CIM servers and their metric providers and gives you additional hints and tips on how to optimize your RMF XP configuration.
For further information about installation and configuration of CIM servers and their metric providers, check out the documentation provided with the RHEL/SUSE software packages and in document SC23-6604 Common Information Model Guide located in the AIX Infocenter.

RMF XP can communicate with two different CIM server implementations, the Open Pegasus CIM server (Tog-Pegasus) and the Small Footprint CIM Broker (SFCB). The SFCB is only available for the Linux operating systems. The required configuration steps are dependent on which AIX or Linux distribution and which CIM server implementation is used.

2 Linux CIM Server Setup for SFCB

2.1 RHEL / SUSE

2.1.1 Installation

Install the following packages that are delivered with your distribution:

sblim-gather
sblim-gather-provider
sblim-cmpi-base
sblim-sfcb

For local test also the wbemcli package is recommended:

sblim-wbemcli

Upon installation it is attempted to register the providers. This may fail, and it may be necessary to perform some extra post install steps to register the providers.

- Register the provider delivered in the sblim-cmpi-base package. As root user, issue the script provider-register.sh script, located in directory /usr/share/sblim-cmpi-base. The provider are located in the /usr/share/sblim-cmpi-base directory. Register all providers.
  e.g.

  provider-register.sh -t sfcb -n root/cimv2 -r Linux_Base.registration -m Linux_Base.mof

- Register the provider delivered in the sblim-gather package. As root user, issue the script provider-register.sh script, located in directory /usr/share/sblim-gather directory. The providers are also located in the /usr/share/sblim-gather directory. Register all providers.

  e.g.

  provider-register.sh -t sfcb -n root/cimv2 -r Linux_IPProtocolEndpointMetric.registration -m Linux_IPProtocolEndpointMetric.mof

For RHEL v6, the registration can fail. Refer to Bug 618080 (/usr/lib/sfcb/CIM schema directory does not exist) on how to circumvent the problem.

2.1.2 CIM Server Configuration

The sfcb reads the configuration file (default is /etc/sfcb/sfcb.cfg) at start up, which contains the sfcb options.

When you plan to use the HTTP port of the CIM Server, ensure to allow HTTP connection by using the following option

enableHttp=true

Ensure to configure the same HTTP port for communication, as in your GPM4CIM configuration. E.g. use the following option

httpPort=5988

When you plan to use the HTTPS port of the CIM Server, ensure to allow HTTPS connection by using the following option

enableHttps=true

Ensure to configure the same HTTPS port for communication, as in your GPM4CIM configuration. E.g. use the following option

httpsPort=5989

To allow access without credentials, disable the Authentication by using the following option:
doBasicAuth=false

Depending on your installation size, the response of the sfcb to the GPM4CIM may exceed the maxMsgLen. It may be necessary to increase the maxMsgLen size, by using the following option:
maxMsgLen=10 000 000

2.1.3 Start and stop the CIM Server

sfcb installs an init.d style script to allow you to start, stop and restart the sfcb daemon.

- To start sfcb, issue the following command as root user:
  /etc/init.d/sfcb start
  or start sfcb service
  /sbin/service sfcb start

- To stop sfcb, issue the following command as root user:
  /etc/init.d/sfcb stop
  or stop sfcb service
  /sbin/service sfcb stop

- To check the sfcb state, issue the following command as root user:
  /etc/init.d/sfcb status
  or check sfcb service
  /sbin/service sfcb status

Alternatively, you can manually start the sfcb daemon by issuing the following command as root user:

/usr/sbin/sfcbd

2.1.4 Start the Gatherer Daemon and the Repository Daemon

The Gatherer Daemon (gatherd) retrieves the performance data. The Repository Daemon (reposd) is the central repository for the data collection and processing. Within the gatherd configuration file, the reposd location can be specified. The gatherd can deliver the performance data to a local reposd or also to a remote reposd. Using this functionality, multiple gatherds' can deliver their data to one central reposd.

The gatherd reads the configuration file (default: /etc/gatherd.conf) at start up, which contains the gatherd options.
To change the repository host of the gatherd, use the following option:

RepositoryHost=localhost

The RepositoryPort option in the gatherd.conf and in the reposd.conf defines the port which is used for communication between the gatherer and repository daemon. Ensure that they match.
e.g. using the default port definition:
RepositoryPort=6363

To start the gatherd, use the Gatherer control program (gatherctl). Issue the command as root user:
/usr/bin/gatherctl

To start the daemon process, type
d
To display the daemon status, type
s
To initialize the daemon, type
i

To quit the gatherctl, type
q

Ensure that the daemon is started and initialized.

Alternatively, you can manually start the gatherd by issuing the following command as root user:
/usr/sbin/gatherd

To start the reposd, use the Repository control program (reposctl). Issue the command as root user:
/usr/bin/reposctl

To start the daemon process, type
d
To display the daemon status, type
s
To initialize the daemon, type
i
To quit the reposctl, type
q

Ensure that the daemon is started and initialized.

Alternatively, you can manually start the reposd by issuing the following command as root user:

/usr/sbin/reposd

3 Linux CIM Server Setup for Tog-Pegasus

3.1 RHEL / SUSE

3.1.1 Installation

Install the following packages that are delivered with your distribution:

tog-pegasus
sblim-gather
sblim-gather-provider
sblim-cmpi-base

For local test also the wbemcli package is recommended:

sblim-wbemcli

Upon installation it is attempted to register the providers. This may fail, and it may be necessary to perform some extra post install steps to register the providers.

- Register the provider delivered in the sblim-cmpi-base package. As root user, issue the script provider-register.sh script, located in directory /usr/share/sblim-cmpi-base. The provider are located in the /usr/share/sblim-cmpi-base directory. Register all providers.
  e.g.

  provider-register.sh -t pegasus -n root/cimv2 -r Linux_Base.registration -m Linux_Base.mof

- Register the provider delivered in the sblim-gather package. As root user, issue the script provider-register.sh script, located in directory /usr/share/sblim-gather directory. The providers are also located in the /usr/share/sblim-gather directory. Register all providers.
  e.g.
  provider-register.sh -t pegasus -n root/cimv2 -r Linux_IPProtocolEndpointMetric.registration
  -m Linux_IPProtocolEndpointMetric.mof

3.1.2 CIM Server Configuration

Use the CIMCONFIG tool to configure the CIM Server as root user. Please note that the cimserver must be active before the CIMCONFIG tool can be used. After a configuration change, the cimserver must be restarted.

When you plan to use the HTTP port of the CIM Server, ensure to allow HTTP connections

cimconfig -s enableHttpConnection=true -p

Ensure to configure the same HTTP port for communication, as in your GPM4CIM configuration

cimconfig -s httpPort=5988 -p

When you plan to use the HTTPS port of the CIM Server, ensure to allow HTTPS connections

cimconfig -s enableHttpsConnection=true -p

Ensure to configure the same HTTPS port for communication, as in your GPM4CIM configuration

cimconfig -s httpsPort=5989 -p

Ensure to enable the repository component of the CIM server to provide CIM object instances by default

cimconfig -s repositoryIsDefaultInstanceProvide=true -p

3.1.3 Start and stop the CIM Server

Tog-Pegasus installs an init.d style script to allow you to start, stop and restart the cimserver.

- To start tog-pegasus, issue the following command as root user:
  /etc/init.d/tog-pegasus start
  or start tog-pegasus service
  /sbin/service tog-pegasus start

- To stop tog-pegasus, issue the following command as root user:
  /etc/init.d/tog-pegasus stop
  or stop tog-pegasus service
  /sbin/service tog-pegasus stop

Alternatively, you can manually start the CIM server by issuing the following command as root user:

/usr/sbin/cimserver

3.1.4 Start the Gatherer Daemon and the Repository Daemon

The Gatherer Daemon (gatherd) retrieves the performance data. The Repository Daemon (reposd) is the central repository for the data collection and processing. Within the gatherd configuration file, the reposd location can be specified. The gatherd can deliver the performance data to a local reposd or also to a remote reposd. Using this functionality, multiple gatherer daemons can deliver their data to one central repository daemon.

The gatherd reads the configuration file (default: /etc/gatherd.conf) at start up, which contains the gatherd options. To change the repository host of the gatherd, use the following option:

RepositoryHost=localhost

The RepositoryPort option in the gatherd.conf and in the reposd.conf defines the port which is used for communication between the gatherer and repository daemon. Ensure that they match.
e.g. using the default port definition:
RepositoryPort=6363

To start the gatherd, use the Gatherer control program (gatherctl). Issue the command as root user:
/usr/bin/gatherctl

To start the daemon process, type
d
To display the daemon status, type
s
To initialize the daemon, type
i
To quit the gatherctl, type
q

Ensure that the daemon is started and initialized.

Alternatively, you can manually start the gatherd by issuing the following command as root user:

/usr/sbin/gatherd

To start the reposd, use the Repository control program (reposctl). Issue the command as root user:
/usr/bin/reposctl

To start the daemon process, type
    d
To display the daemon status, type
    s
To initialize the daemon, type
    i
To quit the reposctl, type
    q

Ensure that the daemon is started and initialized.

Alternatively, you can manually start the reposd by issuing the following command as root user:

        /usr/sbin/reposd

3.2 AIX CIM Server Setup

3.2.1 Installation

Install the following packages delivered with AIX:

sysmgt.cimserver.pegasus
: installs the Pegasus CIM Server fileset in the /opt/freeware/cimom/pegasus directory

sysmgt.cim.providers
: installs the base providers for the AIX fileset in the /usr/pegasus/provider directory

sysmgt.cim.smisproviders
: installs the SMI-S providers for the AIX fileset in the /usr/pegasus/provider directory

3.2.2 Configuration

Use the CIMCONFIG tool to configure the CIM Server as root user. Please note that the cimserver must be active before the CIMCONFIG tool can be used. After a configuration change, the cimserver must be restarted.

When you plan to use the HTTP port of the CIM Server, ensure to allow HTTP connections

```
cimconfig -s enableHttpConnection=true -p
```

Ensure to configure the same HTTP port for communication, as in your GPM4CIM configuration

```
cimconfig -s httpPort=5988 -p
```

When you plan to use the HTTPS port of the CIM Server, ensure to allow HTTPS connections

```
cimconfig -s enableHttpsConnection=true -p
```

Ensure to configure the same HTTPS port for communication, as in your GPM4CIM configuration

```
cimconfig -s httpsPort=5989 -p
```

To allow access without credentials, disable the Authentication:

```
cimconfig -s enableAuthentication=false -p
```

3.2.3 Start and stop the CIM Server

The AIX Pegasus is controlled by the System Resource Controller (SRC). The subsystem name is cimsys and it has three subservers: cimserver, cimlistener and CIM_diagd.

- To start cimsys subsystem and its subservers, issue the following command as root user:

  ```
  startsrc -s cimsys
  ```

  To start the CIM Server, issue the following command as root user:

  ```
  startsrc -t cimserver
  ```

- To stop cimsys subsystem and its subservers, issue the following command as root user:

  ```
  stopsrc -s cimsys
  ```

  To stop the CIM Server, issue the following command as root user:

  ```
  stopsrc -t cimserver
  ```

4 Security Setup for SSL Communication

If your installation requires secure communication between GPM4CIM and the CIM servers on the monitored endpoints, you can set up encryption. GPM4CIM does not provide encryption support explicitly, but you can exploit the z/OS Application Transparent Transport Layer Security (AT-TLS) in order to set up secure server-to-server communication.

The following steps describe one possible way to implement secure communication.

1. Generate and export certificates with the RACF RACDCERT command on z/OS

   Ensure that user id *GPM4USER* must be the same user id that was assigned to started task GPM4CIM.

   - Generate and activate a Certificate-authority certificate which is necessary to create a self- signed certificate:

```
RACDCERT  CERTAUTH -
                GENCERT -
          SUBJECTSDN(OU('organizational-unit-name1') -
                O('organization-name') -
                C('country')) -
          NOTAFTER(DATE(2040-12-31)) -
          SIZE(512) -
          KEYUSAGE(CERTSIGN) -
          WITHLABEL('GPM4CACE')

 SETROPTS RACLIST(DIGTCERT) REFRESH
```

   - Generate and activate a self- signed User certificate:

```
RACDCERT  ID(GPM4USER) GENCERT -
      SUBJECTSDN(CN('common-name') -
            OU(''organizational-unit-name1','organizational-unit-name2') -
            O('organization-name') -
            C('country')) -
      WITHLABEL('GPM4CLCE') -
      SIZE(512) -
      NOTAFTER(DATE(2040-12-31)) -
      SIGNWITH(CERTAUTH   -
            LABEL('GPM4CACE'))

 SETROPTS RACLIST(DIGTCERT) REFRESH
```

   - Create a keyring which needs to be specified in the AT-TLS configuration :

```
 RACDCERT ID(GPM4USER) ADDRING(RMFGPM4RING)

SETROPTS RACLIST(DIGTRING) REFRESH
```

   - Connect the certificate-authority certificate to the keyring:

```
 RACDCERT ID(GPM4USER) -
            CONNECT(CERTAUTH LABEL('GPM4CACE') -
            RING(RMFGPM4RING) -
             )

 SETROPTS RACLIST(DIGTRING) REFRESH
```

- Connect the User certificate to the keyring:

```
RACDCERT CONNECT(ID(GPM4USER) -
       LABEL('GPM4CLCE') -
       RING(RMFGPM4RING) -
        DEFAULT -
       ) -
       ID(GPM4USER)

 SETROPTS RACLIST(DIGTRING) REFRESH
```

- Export the certificate-authority certificate:

```
RACDCERT CERTAUTH EXPORT (LABEL('GPM4CACE')) -
                DSN('HLQ.GPM4CACE.CERT.PFX') -
                FORMAT(PKCS12DER) -
                PASSWORD('TESTPWD')
```

- Export the User certificate:

```
RACDCERT ID(GPM4USER) EXPORT (LABEL('GPM4CLCE')) -
              DSN('HLQ.GPM4CLCE.CERT.PFX') -
              FORMAT(PKCS12DER) -
              PASSWORD('TESTPWD')
```

2. The previously generated certificate needs to be transferred to the Linux or AIX endpoints and modified with the OPENSSL utility.

   On the endpoints, you have to convert and split the certificate with the following OPENSSL commands:

   - Convert and extract the certificate-authority certificate:

     openssl pkcs12 -in *HLQ.GPM4CACE.CERT.PFX* -out *gpm.cacert.pem* -nokeys

   - Convert and extract the User certificate:

     openssl pkcs12 -in HLQ.GPM4CLCE.CERT.PFX -out gpm.clcert.pem -clcerts -nokeys

   - Convert and extract the User Key:

     openssl pkcs12 -in HLQ.GPM4CLCE.CERT.PFX -out gpm.key.pem -nocerts

   - Remove the header from the certificate files:

     openssl x509 -in *gpm.cacert.pem* -out *gpm.cacert.nopw.pem*

     openssl x509 -in *gpm.clcert.pem* -out *gpm.clcert.nopw.pem*

   - *Remove the password and header from the key file:*

     openssl rsa -in *gpm.key.pem* -out *gpm.key.nopw.pem*

3. Modify the CIM server configuration on the endpoints
   - Tog-Pegasus
     Enable SSL communication on the endpoints by using the CIMCONFIG commands:
     cimconfig -s sslClientVerificationMode=disabled -p
     cimconfig -s enableHttpsConnection=true

     On SLES systems only:
     cimconfig -s sslTrustStoreUserName=root -p
     cimconfig -s sslKeyFilePath=./gpm.key.nopw.pem -p
     cimconfig -s sslCertificateFilePath=./gpm.clcert.nopw.pem -p

     Since parameters sslTrustStore, sslCertificateFilePath, and sslKeyFilePath can not be dynamically changed on RHEL systems, replace the complete Tog-Pegasus file.pem and client.pem with the key and certificate generated within RACF on z/OS:

     cp -f gpm.key.nopw.pem /etc/Pegasus/file.pem
     cp -f gpm.clcert.nopw.pem /etc/Pegasus/server.pem

     On Linux Systems user authentication is forced by Tog-Pegasus. With the actual available Linux distributions, it is not possible to perform user/password authentication with certificates and keys. Hence, a user and password has to be specified as GPM4CIM environment definitions:
     GPM_CIMUID=
     GPM_CIMPWD=

   To create an user with password in the linux system use useradd and the passwd command.

   Additionally when the tog-pegasus for Linux is installed, a wbem file is installed in the /etc/pam.d directory. This file will list the PAM security modules used for OpenPegasus. Per default, the file points to pam_access.so and to the related access.conf file.
   /etc/pam.d/wbem
   ```
   account    required     pam_access.so accessfile=/etc/Pegasus/access.conf
   ```

   The access.conf file controls access to the Pegasus WBEM Network services by users with the PAM pam_access module. The specified GPM_CIMUID must be added for wbemNetwork access. For local testing purposes, it makes also sense to add the user to wbemLocal access.
   /etc/Pegasus/access.conf

   ```
   # Pegasus PAM Access Rules:
   # 1. The Remote host user access rule:
   -: ALL EXCEPT pegasus root gpm4cim: wbemNetwork
   #
   #
   # 2. The Local host user access rule:
   #
   -: ALL EXCEPT pegasus root gpm4cim: wbemLocal
   ```

   - SFCB
     Enable SSL communication in the SFCB configuration file (default is /etc/sfcb/sfcb.cfg) on the endpoints:

     enableHttps: true
     sslClientCertificate: ignore
     sslKeyFilePath: ./hlq.gpm.key.nopw.pem
     sslCertificateFilePath: ./hlq.gpm.cacert.nopw.pem

4. Add the following definitions to your AT-TLS configuration file on the z/OS System where GPM4CIM is running:

```
TTLSGroupAction             XXGRP
        {
                TTLSEnabled                 On
                Trace                       0
        }

TTLSRule                GPM4CIMRULE
        {
                RemotePortRange                 5989
                JOBNAME                         GPM4*
                Direction                       Outbound
                TTLSGroupActionRef              XXGRP
                TTLSEnvironmentActionRef        GPM4CIMACT
        }

TTLSEnvironmentAction       GPM4CIMACT
        {
                HandshakeRole           Client
                TTLSEnvironmentAdvancedParms
                        {
                        ClientAuthType          PassThru
                        }
                TTLSKeyringParms
                        {
                        KEYRING                 RMFGPM4RING
                        }
        }
```

5 RMF XP Hints and Tips

5.1 RMF XP Interval Control

- Keep in mind that RMF XP sets the interval length and start-time for monitoring on AIX configurations according to the current setting of the INTERVAL parameter in the platform specific configuration file
- In order to ensure accurate monitoring on Linux, you need to ensure that the gathering interval is correctly set in the SampleInterval parameter of the gatherd configuration file /etc/gatherd.conf. In addition, the local time on the monitored Linux endpoints needs to be set correctly.

5.2 Debugging Aids

You can activate tracing for various components

5.2.1 AT-TLS on z/OS

Add the following definition to your AT-TLS configuration file on the z/OS System where GPM4CIM is running:

```
TTLSGroupAction             XXGRP
        {
                TTLSEnabled                 On
                Trace                       255
        }
```

5.2.2 Pegasus Configuration

Here you can modify all trace options with cimconfig commands

traceLevel

0 Tracing is switched off.

1 Severe trace and log messages (if traceComponents is set to LogMessages)

2 Basic logic flow trace messages, minimal data detail (default)

3 Intra function logic flow and moderate data detail

4 High data detail

5 High data detail + Function entry/exit

Example: cimconfig -s traceLevel=3 -c

TraceComponents

The following table lists the available components:

Authentication
Authorization
CIMExportRequestDispatcher
CIMOMHandle
CMPIProvider
CMPIProviderInterface
CQL
Config
ControlProvider
DiscardedData
Dispatcher
ExportClient
Http
IndicationGeneration
IndicationHandler
IndicationReceipt
IndicationService
L10N
Listener
LogMessages
MessageQueueService
ObjectResolution
OsAbstraction
ProviderAgent
ProviderManager
Repository
SSL
Server
Shutdown
StatisticalData
Thread
UserManager
WQL
WsmServer
Xml
XmlIO

Example: cimconfig -s traceComponents=XmlIO -c

traceFacility

The traceFacility property specifies the target facility to which trace messages are written:

File	The trace messages are written to the file specified by traceFilePath.
Log	The trace messages are written to the logging facility using a logging priority TRACE. (The logLevel property must be set to TRACE.)

Memory	The trace messages are written to a memory buffer. It can be found in a memory dump by searching for the eye-catcher "PEGASUSMEMTRACE" The buffer is organized in a wrap around manner. All messages do have a CR/LF. The last message can be identified by a trailing eye-catcher “*EOTRACE*”.

Example; cimconfig -s traceFacility=Memory -c

traceMemoryBufferKbytes

The traceMemoryBufferKbytes property specifies the size of the memory trace facility in kBytes (1024 bytes). The minimum is 16kB.The default is 10240kB. This property is a planned configuration property and cannot be changed dynamically. It becomes active after a restart of OpenPegasus.

Exapmle: cimconfig -s traceMemoryBufferKbytes=20480 -p

traceFilePath

The traceFilePath property specifies the output file if the traceFacility is set to File.
If the file is specified using a relative path, the file is created relative to PEGASUS_HOME.

Example: cimconfig -s traceFilePath=/tmp/Pegasus.trc -c

5.2.3 SFCB Configuration

sfcbd [trace-*option*]

Options
-t, --trace-components=NUM

Activate component-level tracing messages, where *NUM* is an OR-ed bitmask integer defining which component to trace.
Specifying "-t ?" will list all the components and their associated integer bitmask

Configuration File

sfcbd reads the configuration file /etc/sfcb/sfcb.cfg (or the file specified with the -c option) at start up. The configuration file contains option : value pairs, one per line.

traceLevel Specify the trace level for sfcb. Can be overridden by setting environment variable SFCB_TRACE_LEVEL. Default:*0*.

traceMask Specify the trace mask for sfcb. Can be overridden by the command line option –trace-components. Default: *0*.

traceFile Specify the trace file for sfcb. Can be overridden by setting environment variable SFCB_TRACE_FILE. Default: *stderr*

Environment

SFCB_TRACE Specifies the level of trace/debug messages for sfcb. Valid values are 0 (no trace messages), or 1 (key trace messages only) to 4 (all messages). A default value of 1 will be assumed, if this variable is not set. [Note: SFCB_TRACE level is used in conjunction with sfcb's component-level tracing to control the level of detail of trace messages to display within each component]

SFCB_TRACE_FILE *By default sfcb trace messages are written to STDERR. Setting this environment variable causes the trace messages to be written to a file instead.*

You can verify the CIM server configuration by using the following command line interface in the USS environment of the z/OS system where GPM4CIM is running:

cimcli ei cim_basemetricvalue -l <*hostname:port*> *-u userid -p password* > *values.txt*

You can also verify the CIM server configuration by using the following command line interface on the system where the CIM server is running, when wbemcli is installed:

wbemcli ei http://*userid*:*password*@localhost/root/cimv2:cim_basemetricvalue>*values.txt*

Depending on the CIM server setup, user id and password are mandatory or optional parameters.

5.3 Environment Variables

You can optionally set the following environment variables in the GPM4CIM environment file (Default: gpm4cim.env). Please note that those variables have a application-wide scope and cannot be set individually for particular Linux or AIX endpoints.

- Credentials

  Can be used as workaround for Linux systems with Pegasus CIM. Credentials cannot be deactivated (in contrast to SFCB CIM)

  GPM_CIMUID=
  GPM_CIMPWD=

- Timeout

  Since the default for the initial connect is 60 seconds, this can cause significant delays when multiple systems cannot be reached.

  GPM_CIMTIMEOUT=10

- Include/Exclude Metric Categories from gathering

  For performance reasons, you can reduce the amount of data monitored which especially applies to the Process metric category.

  Only applicable for AIX metrics with more than one instance (1=ON, 0=Off):

GPM_NETWORK_PORT=1
GPM_LOCAL_FILE_SYSTEM=1
GPM_PROCESS=1
GPM_LOGICAL_PROCESSOR=1
GPM_DISK=1

For Linux, the CIM provider modules that are located in directory /usr/lib/gather/rplug or /usr/lib64/gather/rplug can be deactivated using the reposctl utility:

librepositoryIPProtocolEndpoint.so
librepositoryKvm.so
librepositoryLocalFileSystem.so
librepositoryNetworkPort.so
librepositoryOperatingSystem.so
librepositoryProcessor.so
librepositoryUnixProcess.so
librepositoryXen.so
librepositoryzCEC.so
librepositoryzCH.so
librepositoryzECKD.so
librepositoryzLPAR.so

This can be done by unloading a specific provider module with the following command sequence:

```
[root /]# reposctl
s
Status initialized, 12 plugins and 137 metrics.

u librepositoryUnixProcess.so
s
Status initialized, 11 plugins and 117 metrics.
```

If you decide to deactivate CIM provider modules within the repository daemon, IBM recommends to deactivate the corresponding modules within the gatherer daemons on the various endpoints in the same way. This will help to optimize the system performance of RMF XP data gathering. These modules are located in directory /usr/lib/gather/mplug or /usr/lib64/gather/mplug and can be deactivated using the gatherctl utility.

libmetricIPProtocolEndpoint.so
libmetricKvm.so
libmetricLocalFileSystem.so
libmetricNetworkPort.so
libmetricOperatingSystem.so
libmetricProcessor.so
libmetricUnixProcess.so
libmetricXen.so
libmetriczCEC.so
libmetriczCH.so
libmetriczECKD.so
libmetriczLPAR.so

5.4 Enable additional z/Linux metrics gathering:

zCEC and zLPAR Metrics:

needs mounted hypervisor filesystem with the following command:

mount –t s390_hypfs none /sys/hypervisor/s390/

zChannelMetrics:

cm_enable must be issued in advance: echo 1 > /sys/devices/cssx/cm_enable
where x is the id of the channel subsystem, e.g. css0
In order to check the channel subsystem status issue the following command:
cat /sys/devices/css0/cm_enable
Result: enabled = 1, disabled = 0

zECKD (Enhanced Count Key Data) DASD Metrics:

cmb_enable must be issued in advance: echo 1 > /sys/bus/ccw/devices/x.x.xxxx/cmb_enable
where x.x.xxx is the specific device id, e.g. 0.0.aaf8
You can use the “lsccs“ command to list the devices in your system.
Sample output:

```
Device   Subchan.  DevType CU Type Use  PIM PAM POM  CHPIDs
----------------------------------------------------------------------
0.0.aaf8 0.0.0fa5  3390/0a 3990/e9 yes  c0  c0  ff   40440000 00000000
0.0.ab14 0.0.0fc1  3390/0c 3990/e9 yes  c0  c0  ff   40440000 00000000
0.0.f700 0.0.16e7  1732/01 1731/01 yes  80  80  ff   0c000000 00000000
0.0.f701 0.0.16e8  1732/01 1731/01 yes  80  80  ff   0c000000 00000000
0.0.f702 0.0.16e9  1732/01 1731/01 yes  80  80  ff   0c000000 00000000
0.0.f7fe 0.0.17d7  1731/01 1731/01      80  80  ff   0c000000 00000000
```

In order to check the device status issue the following command:
cat /sys/bus/ccw/devices/0.0.aaf8/cmb_enable
Result: enabled = 1, disabled = 0

5.5 RMF XP Environments tested by IBM

5.5.1 Linux

	Red Hat Enterprise Linux Server release 5.6 (Linux on System x, Linux on z System)	*Red Hat Enterprise Linux Server release 6.0 (Linux on System x, Linux on z System)*	*Red Hat Enterprise Linux Server release 7.0 (Linux on System x, Linux on z System)*	*SUSE Linux Enterprise Server 11 (Linux on System x, Linux on z System)*	*SUSE Linux Enterprise Server 12 (Linux on System x, Linux on z System)*
tog-Pegasus	*X*	*X*	*X*		
SFCB				*X*	*X*

5.5.2 AIX

	AIX 6.1.0	*AIX 7.1.0*
Pegasus Version 2.9.0	*X*	*X*