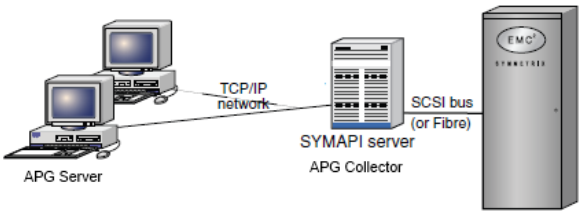
EMC VMAX
Last update 12/11/2013

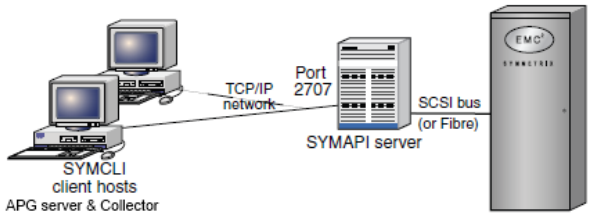
The SolutionPack for EMC VMAX requires EMC Solutions Enabler (SE) 7.6.1.0 installed on your collector server.
Binary can be found on EMC's support website: https://support.emc.com/downloads/2071_Solutions-Enabler

- **Local scenario**



Here, we install only the collector piece directly on the SE server (represented as the SYMAPI server in this diagram. Everything else (APG Frontend, backend and database) is installed on the APG Server. The main advantage is to avoid network calls every time, and configure security between SE software. Disadvantage is the APG footprint on the SYMAPI server. **Using this scenario, you must have 2-4 gatekeeper devices mapped to your APG collector.**

- **Remote scenario**



In this scenario, no APG software is deployed on the SYMAPI server. We install every APG component on the APG server, in addition of EMC Solutions Enabler (SE). The client (APG server) communicates over TCP/IP (port 2706/2707) with the SE SYMAPI server. We recommend mapping 2 additional gatekeeper devices presented to your SYMAPI server.

In order to run in remote mode, you must edit the file "netcnfg" on your client (APG Collector) server. On linux, it is located by default at: /var/symapi/config/netcnfg. For Windows, it is located by default at: C:\Program Files\EMC\SYMAPI\config
Add a line, that contains for example:

```
<YOUR SYMCLI>          -  TCPIP  <SYMAPI short> <SYMAPI ip/hostname> 2707  ANY
```

For example:

```
MYBOX          -  TCPIP  symapi1 10.1.2.3 2707  ANY
```

In this example, the MYBOX string is user defined, and thus could be anything. But you will need to remember it, as it is asked by the ReportPack installer.

On the SYMAPI server, make sure the "storsrvd" service is running. For Unix based installation, you can start it with the command:

```
/opt/emc/SYMCLI/bin/stordaemon start storsrvd
```

When running on local scenario, you might need to grant the “apg” account access. Edit the “API/symapi/config/daemon_users” file (Unix default path is: /opt/APG/emc/API/symapi/config/daemon_users), to add a line for the apg account:

```
apg     <all>
```

And restart the “storapid” deamon/service.

Testing connectivity:

- Unix based installation
    - Local scenario

Connect as the “apg” account using command:

```
[root@testbox APG]# su - apg
-bash-3.2$
```

List connected VMAX arrays:

```
-bash-3.2$ symcfg list

                                S Y M M E T R I X

                                        Mcode    Cache      Num Phys  Num Symm
    SymmID       Attachment  Model      Version  Size (MB)  Devices   Devices

    000194900287 Local       VMAX-1SE   5876       28672         2     12844
    000195700363 Local       VMAX40K    5876       73728         2      2432
    000195700932 Local       VMAX40K    5876       36864         2      1220
    000194900405 Remote      VMAX-1SE   5876       60160         0      2914
```

You can then test other output, like for example the masking view:

```
-bash-3.2$ symaccess -sid 000194900287 list view

Symmetrix ID          : 000194900287

Masking View Name   Initiator Group     Port Group          Storage Group
------------------- ------------------- ------------------- -------------------
BM_lglov177_E75B_C* BM_lglov177_E75B    BM_lglov177_7H0     BM_lglov177_CNA_2
BM_lglov178_E835_C* BM_lglov178_E835    BM_lglov178_7H0     BM_lglov178_CNA_2
CD_lglod220_0E1A    CD_lglod220_0E1A    CD_7F0_8F0          CD_lglod220
CD_lglod220_1514    CD_lglod220_1514    CD_7F0_8F0          CD_lglod220
[…]
```

Note: validate output. In case of missing elements, contact your SYMAPI administrator.

- Remote scenario

Connect as the “apg” account using command:

```
[root@testbox APG]# su - apg
-bash-3.2$
```

Configure SYMAPI to connect remotely:

```
-bash-3.2$ export SYMCLI_CONNECT=MYBOX
-bash-3.2$ export SYMCLI_CONNECT_TYPE=REMOTE
```

List connected VMAX arrays (you may need to add your SE bin directory into your PATH):

```
-bash-3.2$ symcfg list

                                S Y M M E T R I X

                                      Mcode    Cache      Num Phys  Num Symm
```

SymmID	Attachment	Model	Version	Size (MB)	Devices	Devices
000194900287	Local	VMAX-1SE	5876	28672	2	12844
000195700363	Local	VMAX40K	5876	73728	2	2432
000195700932	Local	VMAX40K	5876	36864	2	1220
000194900405	Remote	VMAX-1SE	5876	60160	0	2914

You can then test other output, like for example the masking view (where `000194900287` is any VMAX in the previous list):

```
-bash-3.2$ symaccess -sid 000194900287 list view

Symmetrix ID          : 000194900287

Masking View Name   Initiator Group     Port Group           Storage Group
------------------- ------------------- -------------------- -------------------
BM_lglov177_E75B_C* BM_lglov177_E75B    BM_lglov177_7H0      BM_lglov177_CNA_2
BM_lglov178_E835_C* BM_lglov178_E835    BM_lglov178_7H0      BM_lglov178_CNA_2
CD_lglod220_0E1A    CD_lglod220_0E1A    CD_7F0_8F0           CD_lglod220
CD_lglod220_1514    CD_lglod220_1514    CD_7F0_8F0           CD_lglod220
[…]
```

Note: validate output. In case of missing elements, contact your SYMAPI administrator.

- Windows based installation
    - Local scenario

Watch4net services runs by default as “Local System account”. In order to have a valid test, you need to execute “cmd.exe” as “Local System account”. To do this, you may require “Windows Sysinternals” to achieve this. Refer to “Windows Sysinternals” for more information. Alternatively, you can change your service configuration of your collector to run as “Administrator” account.

To perform the test, open the “Command Prompt”, and list connected VMAX arrays (you may need to add SE bin directory to your PATH):

```
C:\Users\Administrator>set PATH=%PATH%;C:\Program Files\EMC\SYMCLI\bin

C:\Users\Administrator>symcfg list

                              S Y M M E T R I X

                                       Mcode    Cache      Num Phys  Num Symm
    SymmID       Attachment  Model     Version  Size (MB)  Devices   Devices

    000194900287 Local       VMAX-1SE  5876       28672         2     12844
    000195700363 Local       VMAX40K   5876       73728         2      2432
    000195700932 Local       VMAX40K   5876       36864         2      1220
    000194900405 Remote      VMAX-1SE  5876       60160         0      2914
```

You can then test other output, like for example the masking view:

```
C:\Users\Administrator>symaccess -sid 000194900287 list view

Symmetrix ID          : 000194900287

Masking View Name   Initiator Group     Port Group           Storage Group
------------------- ------------------- -------------------- -------------------
BM_lglov177_E75B_C* BM_lglov177_E75B    BM_lglov177_7H0      BM_lglov177_CNA_2
BM_lglov178_E835_C* BM_lglov178_E835    BM_lglov178_7H0      BM_lglov178_CNA_2
CD_lglod220_0E1A    CD_lglod220_0E1A    CD_7F0_8F0           CD_lglod220
CD_lglod220_1514    CD_lglod220_1514    CD_7F0_8F0           CD_lglod220
[…]
```

Note: validate output. In case of missing elements, contact your SYMAPI administrator.

    - Remote scenario

Watch4net services runs by default as “Local System account”. In order to have a valid test, you need to execute “cmd.exe” as “Local System account”. To do this, you may require “Windows Sysinternals” to achieve this. Refer to “Windows Sysinternals” for more information. Alternatively, you can change your service configuration of your collector to run as “Administrator” account.

To perform the test, open the “Command Prompt”, and list connected VMAX arrays (you may need to add SE bin directory to your PATH):

```
C:\Users\Administrator>set PATH=%PATH%;C:\Program Files\EMC\SYMCLI\bin

C:\Users\Administrator>set SYMCLI_CONNECT=MYBOX

C:\Users\Administrator>set SYMCLI_CONNECT_TYPE=REMOTE
```

List connected VMAX arrays (you may need to add your SE bin directory into your PATH):

```
C:\Users\Administrator>symcfg list

                            S Y M M E T R I X

                                       Mcode    Cache      Num Phys  Num Symm
    SymmID       Attachment  Model     Version  Size (MB)  Devices   Devices

    000194900287 Local       VMAX-1SE  5876       28672         2     12844
    000195700363 Local       VMAX40K   5876       73728         2      2432
    000195700932 Local       VMAX40K   5876       36864         2      1220
    000194900405 Remote      VMAX-1SE  5876       60160         0      2914
```

You can then test other output, like for example the masking view (where `000194900287` is any VMAX in the previous list):

```
C:\Users\Administrator>symaccess -sid 000194900287 list view

Symmetrix ID          : 000194900287

Masking View Name   Initiator Group     Port Group          Storage Group
------------------- ------------------- ------------------- -------------------
BM_lglov177_E75B_C* BM_lglov177_E75B    BM_lglov177_7H0     BM_lglov177_CNA_2
BM_lglov178_E835_C* BM_lglov178_E835    BM_lglov178_7H0     BM_lglov178_CNA_2
CD_lglod220_0E1A    CD_lglod220_0E1A    CD_7F0_8F0          CD_lglod220
CD_lglod220_1514    CD_lglod220_1514    CD_7F0_8F0          CD_lglod220
[…]
```

Note: validate output. In case of missing elements, contact your SYMAPI administrator.

Enabling performance collection:

Performance data collection is now collected through Unisphere for VMAX. Unisphere for VMAX must be setup to collect performance data for the arrays. Follow the Unisphere for VMAX installation guide for help about configuring SPA.

Once the product is configured correctly, data collection must be enable for every desired arrays. Login to Unisphere, and make sure collection is enabled. This should look like this:

The “Real Time” status should show green. A good method to determine if data collection is enabled for your array, is to point your browser to:

https://Unisphere_ip:8443/univmax/restapi/performance/Array/keys

This should give you an XML message, similar to this:

Know issues:

- Performance collection in remote scenario.

If running in remote scenario, you may encounter an issue where performance collection is not available. You will notice in logs, some messages similar to:

```
WARNING  -- [2013-02-22 09:52:05 EST] -- MultipleStreamLogger::log(): 195700363: /opt/APG/Collecting/XML-
Collector/emc-vmax/conf/vmax.sh: line 5: 25732 Segmentation fault      $*
SEVERE   -- [2013-02-22 09:52:05 EST] -- AbstractXmlJob::b(): VMAX-DISKS-STATS: 195700363:
net.sf.saxon.trans.XPathException: org.xml.sax.SAXParseException: XML document structures must start and
end within the same entity.
        at net.sf.saxon.event.Sender.sendSAXSource(Sender.java:418)
        at net.sf.saxon.event.Sender.send(Sender.java:214)
        at net.sf.saxon.IdentityTransformer.transform(IdentityTransformer.java:30)
        at com.watch4net.apg.v2.collector.plugins.xmlcollector.engine.AbstractXmlJob.b(SourceFile:127)
        at com.watch4net.apg.v2.collector.plugins.xmlcollector.engine.AbstractXmlJob.call(SourceFile:33)
        at java.util.concurrent.FutureTask$Sync.innerRun(FutureTask.java:303)
        at java.util.concurrent.FutureTask.run(FutureTask.java:138)
        at java.util.concurrent.Executors$RunnableAdapter.call(Executors.java:439)
        at java.util.concurrent.FutureTask$Sync.innerRun(FutureTask.java:303)
        at java.util.concurrent.FutureTask.run(FutureTask.java:138)
        at
java.util.concurrent.ScheduledThreadPoolExecutor$ScheduledFutureTask.access$301(ScheduledThreadPoolExecuto
r.java:98)
        at
java.util.concurrent.ScheduledThreadPoolExecutor$ScheduledFutureTask.run(ScheduledThreadPoolExecutor.java:
206)
        at java.util.concurrent.ThreadPoolExecutor$Worker.runTask(ThreadPoolExecutor.java:886)
        at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:908)
        at java.lang.Thread.run(Thread.java:662)
Caused by: org.xml.sax.SAXParseException: XML document structures must start and end within the same
entity.
        at
com.sun.org.apache.xerces.internal.util.ErrorHandlerWrapper.createSAXParseException(ErrorHandlerWrapper.ja
va:195)
        at
com.sun.org.apache.xerces.internal.util.ErrorHandlerWrapper.fatalError(ErrorHandlerWrapper.java:174)
        at com.sun.org.apache.xerces.internal.impl.XMLErrorReporter.reportError(XMLErrorReporter.java:388)
        at com.sun.org.apache.xerces.internal.impl.XMLScanner.reportFatalError(XMLScanner.java:1427)
        at
com.sun.org.apache.xerces.internal.impl.XMLDocumentFragmentScannerImpl.endEntity(XMLDocumentFragmentScanne
rImpl.java:905)
        at
com.sun.org.apache.xerces.internal.impl.XMLDocumentScannerImpl.endEntity(XMLDocumentScannerImpl.java:604)
        at com.sun.org.apache.xerces.internal.impl.XMLEntityManager.endEntity(XMLEntityManager.java:1391)
        at com.sun.org.apache.xerces.internal.impl.XMLEntityScanner.load(XMLEntityScanner.java:1763)
        at com.sun.org.apache.xerces.internal.impl.XMLEntityScanner.peekChar(XMLEntityScanner.java:487)
        at
com.sun.org.apache.xerces.internal.impl.XMLDocumentFragmentScannerImpl$FragmentContentDriver.next(XMLDocum
entFragmentScannerImpl.java:2688)
```

```
        at
com.sun.org.apache.xerces.internal.impl.XMLDocumentScannerImpl.next(XMLDocumentScannerImpl.java:647)
        at
com.sun.org.apache.xerces.internal.impl.XMLNSDocumentScannerImpl.next(XMLNSDocumentScannerImpl.java:140)
        at
com.sun.org.apache.xerces.internal.impl.XMLDocumentFragmentScannerImpl.scanDocument(XMLDocumentFragmentSca
nnerImpl.java:511)
        at
com.sun.org.apache.xerces.internal.parsers.XML11Configuration.parse(XML11Configuration.java:808)
        at
com.sun.org.apache.xerces.internal.parsers.XML11Configuration.parse(XML11Configuration.java:737)
        at com.sun.org.apache.xerces.internal.parsers.XMLParser.parse(XMLParser.java:119)
        at com.sun.org.apache.xerces.internal.parsers.AbstractSAXParser.parse(AbstractSAXParser.java:1205)
        at
com.sun.org.apache.xerces.internal.jaxp.SAXParserImpl$JAXPSAXParser.parse(SAXParserImpl.java:522)
        at net.sf.saxon.event.Sender.sendSAXSource(Sender.java:404)
        ... 14 more
---------
org.xml.sax.SAXParseException: XML document structures must start and end within the same entity.
        at
com.sun.org.apache.xerces.internal.util.ErrorHandlerWrapper.createSAXParseException(ErrorHandlerWrapper.ja
va:195)
        at
com.sun.org.apache.xerces.internal.util.ErrorHandlerWrapper.fatalError(ErrorHandlerWrapper.java:174)
        at com.sun.org.apache.xerces.internal.impl.XMLErrorReporter.reportError(XMLErrorReporter.java:388)
        at com.sun.org.apache.xerces.internal.impl.XMLScanner.reportFatalError(XMLScanner.java:1427)
        at
com.sun.org.apache.xerces.internal.impl.XMLDocumentFragmentScannerImpl.endEntity(XMLDocumentFragmentScanne
rImpl.java:905)
        at
com.sun.org.apache.xerces.internal.impl.XMLDocumentScannerImpl.endEntity(XMLDocumentScannerImpl.java:604)
        at com.sun.org.apache.xerces.internal.impl.XMLEntityManager.endEntity(XMLEntityManager.java:1391)
        at com.sun.org.apache.xerces.internal.impl.XMLEntityScanner.load(XMLEntityScanner.java:1763)
        at com.sun.org.apache.xerces.internal.impl.XMLEntityScanner.peekChar(XMLEntityScanner.java:487)
        at
com.sun.org.apache.xerces.internal.impl.XMLDocumentFragmentScannerImpl$FragmentContentDriver.next(XMLDocum
entFragmentScannerImpl.java:2688)
        at
com.sun.org.apache.xerces.internal.impl.XMLDocumentScannerImpl.next(XMLDocumentScannerImpl.java:647)
        at
com.sun.org.apache.xerces.internal.impl.XMLNSDocumentScannerImpl.next(XMLNSDocumentScannerImpl.java:140)
        at
com.sun.org.apache.xerces.internal.impl.XMLDocumentFragmentScannerImpl.scanDocument(XMLDocumentFragmentSca
nnerImpl.java:511)
        at
com.sun.org.apache.xerces.internal.parsers.XML11Configuration.parse(XML11Configuration.java:808)
        at
com.sun.org.apache.xerces.internal.parsers.XML11Configuration.parse(XML11Configuration.java:737)
        at com.sun.org.apache.xerces.internal.parsers.XMLParser.parse(XMLParser.java:119)
        at com.sun.org.apache.xerces.internal.parsers.AbstractSAXParser.parse(AbstractSAXParser.java:1205)
        at
com.sun.org.apache.xerces.internal.jaxp.SAXParserImpl$JAXPSAXParser.parse(SAXParserImpl.java:522)
        at net.sf.saxon.event.Sender.sendSAXSource(Sender.java:404)
        at net.sf.saxon.event.Sender.send(Sender.java:214)
```

The problem is that Solutions Enabler “segfault” in remote mode, when fetching statistics.
Fix: Make sure you are running Solutions Enabler 7.5.1.0 & +.