

Primeton Portal 在 IBM HTTP Server7、WebSphere7 中 代理配置指南

概述

本文从 Primeton Portal 的配置集成出发, 介绍 Primeton Portal 在 AIX 环境下, 与 IBM HTTP Server7 + WebSphere 7 的相关配置过程。

此文档中不会包含有 Primeton Portal 安装配置的过程信息, 如有需要, 请参考《Portal_7.0 集成指南》。

与单点登录相关的内容请参考《SSO 集成指南》 文档。

本文目标

解决 Primeton Portal 安装在 AIX + IBM HTTP Server7 + WebSphere7 中遇到的相关代理配置, Portal 依赖的第三方模块(mod_substitute)缺失的问题。

业务背景

在实施某客户的 Portal 时, 客户要求必须基于以下环境:

1. AIX
2. 使用 IBM HTTP Server 7
3. 使用 WebSphere 7
4. 不允许安装开源软件, 如 Apache

参考资料

《Apache 驱动的 IBM HTTP Server 常见问题.pdf》

《apxs - Apache 扩展工具.pdf》

《Linux 下通过 apxs 命令给 Apache 添加模块.pdf》

《如何向已经安装好的 apache 添加一个模块.pdf》

《使用第三方插件模块扩展 IBM HTTP Server 功能.pdf》

集成步骤

此文档中不再介绍 Primeton Portal 基于独立部署的模式和在已有应用上部署 Portal 的相关安装与配置步骤，详细的操作方式和配置步骤请参考《Portal_7.0 集成指南.pdf》相关章节内容。

IBM HTTP Server 安装

客户已将 IBM HTTP Server 安装在 /usr/IBM/HTTPServer 中，下文提到的 %IHS_HOME% 无特殊说明的情况下是指该目录。

在某些情况下，如果客户没有安装好 IBM HTTP Server 时，建议通过向导形式安装 IBM HTTP Server，这里不再进行介绍。

配置 WebServer

在已经安装好 IBM HTTP Server 和 Portal 应用的情况下，通过访问 Portal 应用所属的 WebSphere Integrated Solutions Console （集成解决方案控制台），如：
<https://IP:AdminPort/ibm/console/login.do>，如下图所示：



图表 1 管理控制台

点击服务器->服务器类型->Web 服务器，进入到 Web 服务器列表页面，如下图所示：



图表 2 Web 服务器

点击创建，进行 Web 服务器的定义，如下图所示：



图表 3 创建新的 Web 服务器定义

输入参数说明如下：

- 1、 服务器名称可以进行自定义，输入一个标识名称即可。
- 2、 类型——由于此次只能够使用 IBM HTTP Server，因此使用默认选择 IBM HTTP Server。
- 3、 主机名——填写 IBM HTTP Server 所在的服务器的 IP 地址。
- 4、 平台——选择 IBM HTTP Server 所在的服务器的平台类型，这里选择为 AIX。

示例配置详细信息如下图所示：

创建新的 Web 服务器定义

使用此页面来创建新的 Web 服务器。

→ 步骤 1: 指定 Web 服务器名称、类型、主机名和平台。

步骤 2: 选择 Web 服务器模板

步骤 3: 为新的 Web 服务器输入属性

步骤 4: 确认新的 Web 服务器

指定 Web 服务器名称、类型、主机名和平台。

指定 Web 服务器名称、类型、主机名和平台。

* 服务器名称
portalWebServer

* 类型
IBM HTTP Server

* 主机名
96. .148

* 平台
AIX

下一步

取消

图表 4 指定 Web 服务器名称、类型、主机名称和平台

点击 **下一步**，进行 Web 服务器模板的选择配置，这里选择默认即可，如下图所示：

创建新的 Web 服务器定义

使用此页面来创建新的 Web 服务器。

步骤 1: 指定 Web 服务器名称、类型、主机名和平台。

→ 步骤 2: 选择 Web 服务器模板

步骤 3: 为新的 Web 服务器输入属性

步骤 4: 确认新的 Web 服务器

选择 Web 服务器模板

选择符合您要创建的服务器的模板。

选择

模板名

类型

描述

☒

IHS

系统

The IHS Web Server Template

上一步

下一步

取消

图表 5 使用 Web 服务器模板

继续点击 **下一步**，指定新的 Web 服务器输入属性信息，如下图所示：

创建新的 Web 服务器定义

使用此页面来创建新的 Web 服务器。

步骤 1: 指定 Web 服务器名称、类型、主机名和平台。

步骤 2: 选择 Web 服务器模板

→ 步骤 3: 为新的 Web 服务器输入属性

步骤 4: 确认新的 Web 服务器

为新的 Web 服务器输入属性

输入 Web 服务器属性。

* 端口

80

* Web 服务器安装位置

/usr/IBM/HTTPServer

* 插件安装位置

/usr/IBM/HTTPServer/Plugins

至 Web 服务器的应用程序映射

全部 ▼

输入 IBM Administration Server 属性。

* 管理服务器端口

8008

* 用户名

* 密码

* 确认密码

☐ 使用 SSL

上一步

下一步

取消

图表 6 输入 Web 服务器输入属性

Web 服务器属性说明：

- 1、 端口——IBM HTTP Server 的访问端口
- 2、 Web 服务器安装位置——IBM HTTP Server 的安装目录
- 3、 插件安装位置——IBM HTTP Server Plugins 的安装目录，通常情况下，该目录处于%IHS_HOME%/Plugins 目录下，如果在对应的目录下未找到，可咨询系统管理员
- 4、 至 Web 服务器的应用程序映射选择“全部”
- 5、 IBM Administration Server 属性中的用户名和密码可向系统管理员获取，管理服务器端口通常为 8008

示例配置详细信息如下图所示：

创建新的 Web 服务器定义

使用此页面来创建新的 Web 服务器。

步骤 1: 指定 Web 服务器名称、类型、主机名和平台。

步骤 2: 选择 Web 服务器模板

→ 步骤 3: 为新的 Web 服务器输入属性

步骤 4: 确认新的 Web 服务器

为新的 Web 服务器输入属性

输入 Web 服务器属性。

* 端口

8080

* Web 服务器安装位置

/usr/IBM/HTTPServer

* 插件安装位置

/usr/IBM/HTTPServer/Plugins

至 Web 服务器的应用程序映射

全部 ▾

输入 IBM Administration Server 属性。

* 管理服务端口

8008

* 用户名

ihs

* 密码

●●●

* 确认密码

●●●

☐ 使用 SSL

上一步

下一步

取消

图表 7 输入 Web 服务器属性

点击 **下一步**，确认新的 Web 服务器，如下图所示：



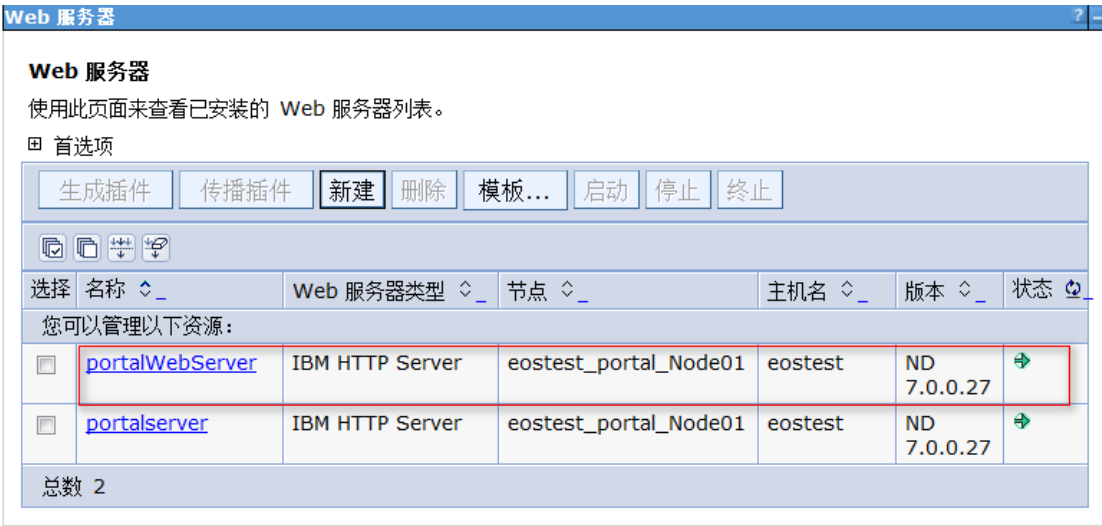
图表 8 确认新的 Web 服务器

点击 **完成** 按钮，进行 Web 服务器的创建，如下图所示：



图表 9 保存主配置信息

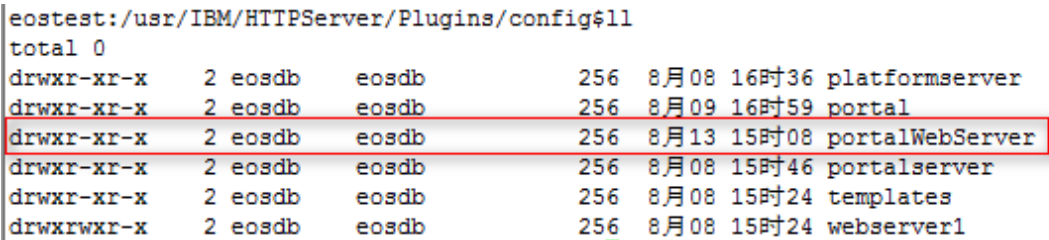
点击 **保存到主配置。**，新的 Web 服务器配置完成。如下图所示：



图表 10 已创建的 Web 服务器

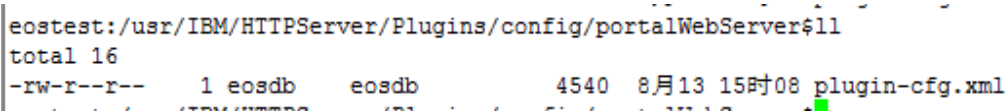
完成 Web 服务器定义后，需要检查新的 Web 服务器相关的插件配置文件是否已经生成到%IHS_HOME%/Plugins/config 目录下，如果没有，则需要生成插件并传播插件，检查步骤如下：

第一步、进入到%IHS_HOME%/Plugin/config 目录下，查看文件列表，如下图所示：



图表 11 查看生成的 Web 服务器配置

portalWebServer 文件夹下应该存在有 plugin-cfg.xml 配置文件，如下图所示：



图表 12 生成的 Web 服务器配置

第二步、如果 Web 服务器名称对应的文件夹下没有 plugin-cfg.xml 配置文件，则需要 在 WebSphere 管理控制台中进行插件的生成和传播，如下图所示：



图表 13 生成插件操作

提示生成成功，如下图所示：



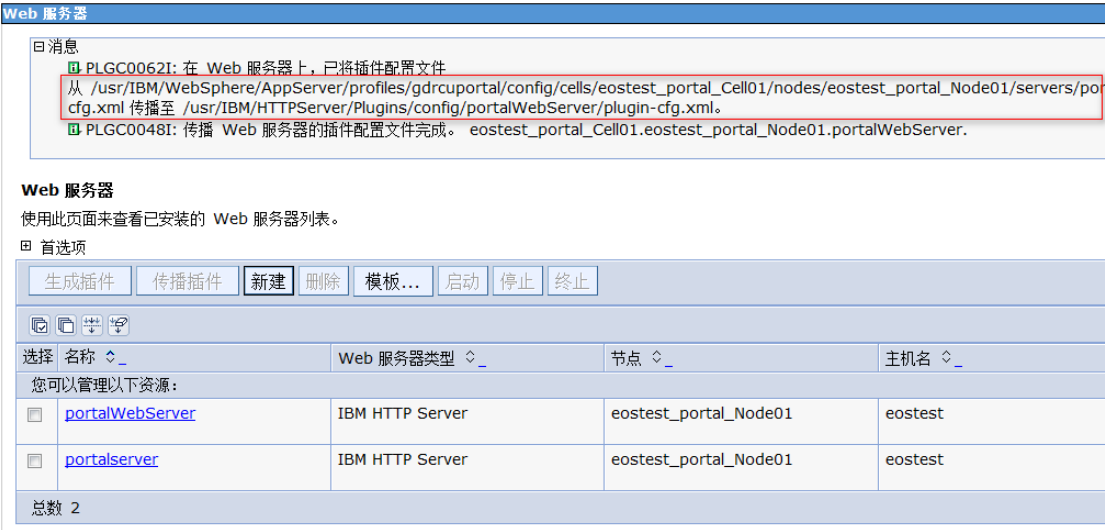
图表 14 生成插件配置

第三步、进行插件的传播，如下图所示：



图表 15 传播插件

插件传播成功，如下图所示：



图表 16 插件传播成功

第三方模块的编译和加载

WebSphere 中的 Web 服务器配置完成后，需要配置 IBM HTTP Server 的相关代理配置。通过在 WebSphere 6 和 WebSphere 7 的检查，Primeton Portal 依赖的 Apache mod_substitute（第三方模块，主要是提供根据正则表达式对响应头中的字符串进行替换）在 IBM HTTP Server 中默认是不包含的，因此需要进行编译和加载。

根据 IBM 的相关帮助说明，IBM HTTP Server 中是能够支持对第三方的模块的，如下截图所示：

我的第三方模块能与 IHS 一起工作吗?

如果某个第三方模块可以和 Apache 一起工作，那么它没有理由不可以与 IHS 一起工作。如果模块已经为 Apache 的某一版本进行了预编译，而此版本 Apache 中的模块幻数与 IHS 中的匹配，则此模块可以装入并且“在盒子外”工作。如果模块对包含不同于 IHS 的模块幻数的 Apache 版本进行了预编译，则与损坏的模块结构有关的错误或不正确的模块幻数将阻止服务器启动。在这种情况下，需要对提供的 IHS 标题文件和核心库重建模块。因为不提供 IHS 的源代码，您可能只能使用通过 IHS 动态装入的模块。

图表 17 第三方模块与 IHS 工作说明

根据 IBM 的相关帮助文档，发现动态模块的构件可通过 Apache 扩展工具（APXS）或随模块提供的配置脚本来进行编译，可参考《使用第三方插件模块扩展 IBM HTTP Server 功能.pdf》文档。

此次操作的过程如下：

第一步、下载 `httpd-2.2.14.tar.bz2` 包，将该包中的 `modules/filters/mod_substitute.c` 文件上传到 `%IHS_HOME%/bin` 目录中，如下图所示：

```
eostest:/usr/IBM/HTTPServer/bin$ls
ab                                httpd
adminctl                         httxt2dbm
apachectl                       ikeyman
apr-1-config                    install
apu-1-config                   installRegistryUtils.sh
apxs                            installver.sh
dbmmanage                      ldapstash
envvars                        logresolve
envvars-std                    mod_substitute.c
genHistoryReport.sh            postinst
genVersionReport.sh           registerproduct.sh
gsk7capiCmd                   rotatelog
gsk7cmd                        setInstallRegistryUtilsEnv.sh
gsk7ver                       setupCmdLine.sh
gsk_envvars                   setupadm
historyInfo.sh                sidd
htcacheclean                  sslstash
htdbm                         verifyinstallver.sh
htpasswd                       versionInfo.sh
```

图表 18 上传 mod_substitute.c 文件

第二步、执行 `apxs` 命令：`apxs -ci mod_substitute.c`，如下图所示：

```
eostest:/usr/IBM/HTTPServer/bin$apxs -ci mod_substitute.c
/usr/IBM/HTTPServer/build/libtool --silent --mode=compile xlc_r -prefer-pic
include -I/usr/IBM/HTTPServer/include -I/usr/IBM/HTTPServer/include -
/usr/IBM/HTTPServer/build/libtool[847]: xlc_r: not found
apxs:Error: Command failed with rc=65536
```

图表 19 使用 apxs 进行模块编译

说明：大部份情况下，执行 `apxs` 命令会提示 `xl_c_r not found`，通过分析错误信息可了解到 `xl_c_r` 指的是 **IBM XL C/C++ Compiler**，提示找不到说明当前执行编译的用户的 `PATH` 中没有添加该编译器的路径。一般情况下，系统管理员在安装完操作系统后，都会安装 **C** 和 **C++** 编译器，可能是 **GCC** 或 **IBM** 标准编译器；此次的环境中安装的为 **IBM XL C/C++ Compiler**，因此找到对应的安装目录，如：`/usr/vac/bin` 目录中，查找 `xl_c_r`，如下图所示：

```

eosdbtest:/usr/vac/bin$ls
CreateExportList  c99_128_r      cc_r      xlc128
c89               c99_128_r4    cc_r4     xlc128_r
c89_128           c99_128_r7    cc_r7     xlc128_r4
c89_128_r         c99_r         cleanpdf  xlc128_r7
c89_128_r4        c99_r4        fixpkg_vacndi  xlc_r
c89_128_r7        c99_r7        gxlc      xlc_r4
c89_r             cc            mergepdf   xlc_r7
c89_r4            cc128         resetpdf   xlchelp
c89_r7            cc128_r      showpdf
c99               cc128_r4     vacndi
c99_128           cc128_r7     xlc
```

图表 20 xlc_r

第三步、将 `xl_c_r` 的路径添加到用户的 `.profile` 中，在系统用户的 `HOME` 目录中使用 `ls -la` 命令，如下图所示：

```

eosdbtest:/home/eosdb$ls -la
total 64904
drwxr-xr-x  8 eosdb eosdb      4096  8月13 10时09 .
drwxr-xr-x  8 bin   bin      256  6月18 10时14 ..
-rw-r--r--  1 eosdb eosdb      972  8月06 10时18 .eosreg
drwxr-xr-x  3 eosdb eosdb      256  6月18 17时42 .java
-rwxr----- 1 eosdb eosdb      971  8月13 15时41 .profile
-rw-----  1 eosdb eosdb    260082  8月13 15时49 .sh_history
drwxr-xr-x  3 eosdb eosdb      256  7月10 14时25 .tuscany
-rw-----  1 eosdb eosdb      179  8月13 15时41 .vi_history
drwxrwxrwx  5 eosdb eosdb      256  7月10 14时01 EOS6.6
drwxr-xr-x  3 eosdb eosdb      256  7月10 11时25 PMT
drwxr-xr-x  6 eosdb eosdb     4096  8月01 10时11 eosdb_ddl
-rw-r--r--  1 eosdb eosdb    32942080  8月06 17时56 httpd-2.2.14.tar
drwxr-xr-x  3 eosdb eosdb      256  7月10 13时46 workspace
```

图表 21 编辑.profile 文件

使用 `vi` 编辑 `.profile` 文件，在 `PATH` 中添加 `/usr/vac/bin/` 路径，在 `Linux` 系统中，`PATH` 分隔符为 “:”（不带双引号），如下图所示：

```

PATH=/usr/bin:/etc:/usr/sbin:/usr/ucb:$HOME/bin:/usr/bin/X11:/sbin:/usr/IBM
/HTTPServer/bin:/usr/vac/bin/;

export PATH
export LANG=zh_CN

alias cls='clear'
alias ll='ls -l'
alias la='ls -al'
alias topm='cd /usr/IBM/WebSphere/AppServer/profiles/pm'
alias todefault='cd /usr/IBM/WebSphere/AppServer/profiles/pm/installedApps/
eostest_pm_Cell01/default.ear/default.war'
alias tologs='cd /usr/IBM/WebSphere/AppServer/profiles/pm/logs/server1'
alias tobin='cd /usr/IBM/WebSphere/AppServer/profiles/pm/bin'
alias toplatform='cd /usr/IBM/WebSphere/AppServer/profiles/gdrcuplatform'
alias toportal='cd /usr/IBM/WebSphere/AppServer/profiles/gdrcuportal'

if [ -s "$MAIL" ]          # This is at Shell startup.  In normal
then echo "$MAILMSG"      # operation, the Shell checks
fi                        # periodically.

# The following three lines have been added by IBM DB2 instance utilities.
".profile" 26 行, 971 个字符

```

图表 22 编辑.profile 文件，增加路径

编辑完成后，保存并退回，建议重新打开窗口进行模块的编译。

第四步，重新编译模块，如下图所示：

```

eostest:/usr/IBM/HTTPServer/bin$apxs -ci mod_substitute.c
/usr/IBM/HTTPServer/build/libtool --silent --mode=compile xlc_r -prefer-pic
include -I/usr/IBM/HTTPServer/include -I/usr/IBM/HTTPServer/include -
/usr/IBM/HTTPServer/build/libtool --silent --mode=link xlc_r -o mod_substit
/usr/IBM/HTTPServer/build/instdso.sh SH_LIBTOOL='/usr/IBM/HTTPServer/build/
rm -f /usr/IBM/HTTPServer/modules/mod_substitute.so
/usr/IBM/HTTPServer/build/libtool --mode=install cp mod_substitute.la /usr/
cp .libs/libmod_substitute.so.0.0.0 /usr/IBM/HTTPServer/modules/libmod_subs
(cd /usr/IBM/HTTPServer/modules && rm -f libmod_substitute.so.0 && ln -s li
(cd /usr/IBM/HTTPServer/modules && rm -f libmod_substitute.so && ln -s libm
cp .libs/mod_substitute.lai /usr/IBM/HTTPServer/modules/mod_substitute.la
cp .libs/libmod_substitute.a /usr/IBM/HTTPServer/modules/libmod_substitute.
ranlib /usr/IBM/HTTPServer/modules/libmod_substitute.a
chmod 644 /usr/IBM/HTTPServer/modules/libmod_substitute.a
-----
Libraries have been installed in:
  /usr/IBM/HTTPServer/modules

If you ever happen to want to link against installed libraries
in a given directory, LIBDIR, you must either use libtool, and
specify the full pathname of the library, or use the '-LLIBDIR'
flag during linking and do at least one of the following:
  - add LIBDIR to the 'LIBPATH' environment variable
    during execution
  - use the '-Wl,-blibpath:LIBDIR:/usr/lib:/lib' linker flag

See any operating system documentation about shared libraries for
more information, such as the ld(1) and ld.so(8) manual pages.
-----
chmod 755 /usr/IBM/HTTPServer/modules/mod_substitute.so
eostest:/usr/IBM/HTTPServer/bin$

```

图表 23 处理重新编译

编译完成后, 在 %IHS_HOME%/modules 目录下可查找到对应的模块文件, 如下图所示:


```

eostest:/usr/IBM/HTTPServer/modules$ls
WebSphereCE      mod_cgid.so      mod_log_config.so
debug            mod_dav.so       mod_logio.so
httpd.exp        mod_dav_fs.so   mod_mem_cache.so
mod_actions.so   mod_deflate.so  mod_mime.so
mod_afpa_cache.so mod_dir.so      mod_mime_magic.so
mod_alias.so     mod_disk_cache.so mod_negotiation.so
mod_asis.so      mod_env.so      mod_proxy.so
mod_auth_basic.so mod_expires.so  mod_proxy_connect.so
mod_authn_anon.so mod_ext_filter.so mod_proxy_ftp.so
mod_authn_dbm.so  mod_fastcgi.so  mod_proxy_http.so
mod_authn_default.so mod_file_cache.so mod_reqtimeout.so
mod_authn_file.so mod_filter.so    mod_rewrite.so
mod_authnz_ldap.so mod_headers.so  mod_setenvif.so
mod_authz_dbm.so  mod_ibm_admin.so mod_speling.so
mod_authz_default.so mod_ibm_ldap.so  mod_status.so
mod_authz_groupfile.so mod_ibm_ssl.so  mod_substitute.so
mod_authz_host.so mod_imagemap.so mod_unique_id.so
mod_authz_user.so  mod_include.so  mod_userdir.so
mod_autoindex.so  mod_info.so     mod_usertrack.so
mod_cache.so      mod_ldap.so     mod_vhost_alias.so

```

图表 24 查看编译结果

同时，在 %IHS_HOME%/conf/httpd.conf 文件中会自动添加模块加载配置，如下图所示：

```

# mod_whatkilledus.
EnableExceptionHook On

# mod_whatkilledus will record information about the current request
# and connection to the error log at the time of a crash. This is
# important information for diagnosing the cause of the crash.
LoadModule whatkilledus_module modules/debug/mod_whatkilledus.so

# mod_net_trace will record actual data sent/received from the client
# and on proxy connections, even for SSL connections. Unlike an IP
# trace, interaction with the platform network APIs can be seen.
# The following example configuration can be activated by uncommenting
# the LoadModule directive.
#LoadModule net_trace_module modules/debug/mod_net_trace.so
<IfModule mod_net_trace.c>
NetTraceFile /tmp/nettrace
NetTrace client * dest file event senddata=65535 event recvddata=65535
</IfModule>

LoadModule was_ap22_module /usr/IBM/HTTPServer/Plugins/bin/mod_was_ap22_htt
p.so
LoadModule substitute_module modules/mod_substitute.so
WebSpherePluginConfig /usr/IBM/HTTPServer/Plugins/config/platformserver/plu
gin-cfg.xml

```

图表 25 模块加载配置

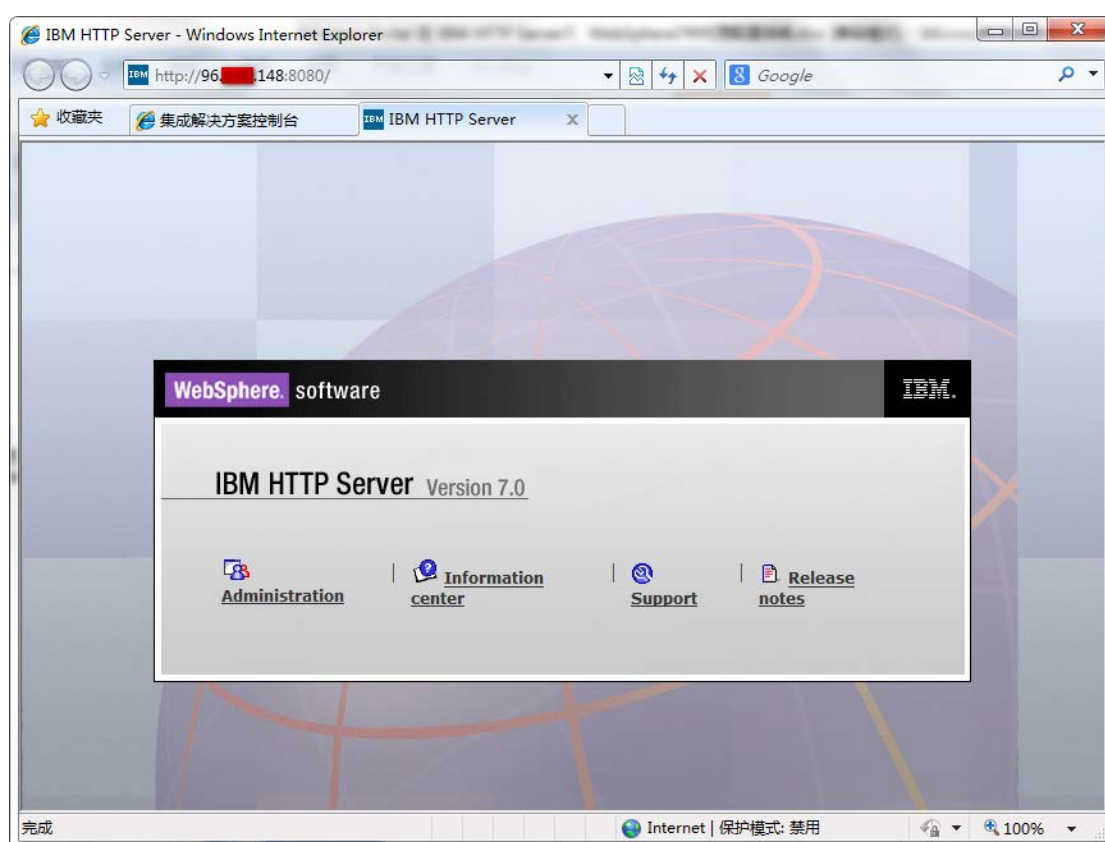
至此，通过 IBM HTTP Server 提供的 apxs 工具（其实是 Apache 的 apxs 工具，向 Apache 致敬！）进行第三方模块的编译和加载已经全部完成。

第五步、验证 IBM HTTP Server 启动是否正常，使用 `apachectl restart` 重启 IBM HTTP Server。如下图所示：

```
eostest:/usr/IBM/HTTPServer/bin$apachectl restart
httpd not running, trying to start
eostest:/usr/IBM/HTTPServer/bin$
```

图表 26 重启 IHS

通过浏览器访问 IBM HTTP Server，如下图所示：



图表 27 通过浏览器访问 IBM HTTP Server

IBM HTTP Server 代理配置

在 IBM HTTP Server 中进行代理配置并无太大差别，如下步骤所示：

第一步、在 `%HIS_HOME%/conf` 目录下编辑 `httpd.conf` 文件，如下图所示：

```

<VirtualHost *:8080>
  #Header add MyHeader "%D %t"
  SetOutputFilter INFLATE
  AddOutputFilterByType SUBSTITUTE text/html .jsp
  Substitute "s|<head>|<head><script>var WidgetContext=top.WidgetContext;if(WidgetContext){WidgetContext.Widget.headStart(window);}</script>|i"
  Substitute "s|</head>|<script>var WidgetContext=top.WidgetContext;if(WidgetContext){WidgetContext.Widget.headEnd(window);}</script></head>|i"
  Substitute "s|</body>|<script>varWidgetContext=top.WidgetContext;if(WidgetContext){WidgetContext.Widget.bodyEnd(window);}</script></body>|i"

  ProxyPass /default/ http://96.148.9082/default/
  ProxyPassReverse /default/ http://96.148.9082/default/

  ProxyPass /portal/ http://96.148.9081/portal/
  ProxyPassReverse /portal/ http://96.148.9081/portal/
</VirtualHost>

```

图表 28 增加代理配置

具体的内容如下：

```

<VirtualHost *:8080>
  #Header add MyHeader "%D %t"
  SetOutputFilter INFLATE
  AddOutputFilterByType SUBSTITUTE text/html .jsp
  Substitute "s|<head>|<head><script>var
WidgetContext=top.WidgetContext;if(WidgetContext){WidgetContext.Widget.headStart(window
);}</script>|i"
  Substitute "s|</head>|<script>var
WidgetContext=top.WidgetContext;if(WidgetContext){WidgetContext.Widget.headEnd(window);
}</script></head>|i"
  Substitute "s|</body>|<script>varWidgetContext=top.WidgetContext;if(WidgetContext){WidgetContext.Wi
dget.bodyEnd(window);}</script></body>|i"

  ProxyPass /default/ http://IP:9082/default/
  ProxyPassReverse /default/ http:// IP:9082/default/

  ProxyPass /portal/ http:// IP:9081/portal/
  ProxyPassReverse /portal/ http:// IP:9081/portal/
</VirtualHost>

```

第二步、检查加载的 Web 服务器的插件配置文件 `plugin-cfg.xml` 中是否已经增加了 IBM HTTP Server 的监听端口。打开 `%IHS_HOME%/Plugins/portalWebServer` 目录下的 `plugin-cfg.xml` 文件，如下图所示：

```

<Log LogLevel="Error" Name="/usr/IBM/HTTPServer/Plugins/logs/portalWebServer/http_plugin.log"/>
<Property Name="ESIEnable" Value="true"/>
<Property Name="ESIMaxCacheSize" Value="1024"/>
<Property Name="ESIInvalidationMonitor" Value="false"/>
<Property Name="ESIEnableToPassCookies" Value="false"/>
<Property Name="ESICacheidFull" Value="false"/>
<Property Name="PluginInstallRoot" Value="/usr/IBM/HTTPServer/Plugins/" />
<VirtualHostGroup Name="default_host">
  <VirtualHost Name="*:9081"/>
  <VirtualHost Name="*:80"/>
  <VirtualHost Name="*:9444"/>
  <VirtualHost Name="*:5060"/>
  <VirtualHost Name="*:5061"/>
  <VirtualHost Name="*:443"/>
</VirtualHostGroup>

```

图表 29 增加端口

增加 <VirtualHost Name="*:8080"/> 的端口，如下图所示：

```

<Log LogLevel="Error" Name="/usr/IBM/HTTPServer/Plugins/logs/portalWebServer/http_plugin.log"/>
<Property Name="ESIEnable" Value="true"/>
<Property Name="ESIMaxCacheSize" Value="1024"/>
<Property Name="ESIInvalidationMonitor" Value="false"/>
<Property Name="ESIEnableToPassCookies" Value="false"/>
<Property Name="ESICacheidFull" Value="false"/>
<Property Name="PluginInstallRoot" Value="/usr/IBM/HTTPServer/Plugins/" />
<VirtualHostGroup Name="default_host">
  <VirtualHost Name="*:9081"/>
  <VirtualHost Name="*:80"/>
  <VirtualHost Name="*:8080"/>
  <VirtualHost Name="*:9444"/>
  <VirtualHost Name="*:5060"/>
  <VirtualHost Name="*:5061"/>
  <VirtualHost Name="*:443"/>
</VirtualHostGroup>

```

图表 30 增加端口设置

第三步，重新启动 IBM HTTP Server 进行测试，如下图所示：



图表 31 通过 IBM HTTP Server 访问接入的 Portal 应用



图表 32 通过 IBM HTTP Server 访问 Portal

多个应用接入的情况下 IBM HTTP Server 中插件配置文件加载的问题

Portal 在接入多个应用（如 CRM、HR、财务系统等）的情况下，需要对每个不同的应用添加 Web 服务器，IBM HTTP Server 的 httpd.conf 中只能够加载一个 Web 服务器的插件配置文件（如果配置多个，则只会加载最后一个 plugin-cfg.xml）。如果多个应用安装在同一台应用服务器的不同 Profile 的情况下，则需要对不同 Profile 下的应用的插件配置文件进行合并（plugin-cfg.xml）。

例如 Portal 应用的 Web 服务器名称为 portalserver，而综合管理平台的服务器名称为 platformserver，如下图所示：

单元 = eostest_portal_Cell01, 概要文件 = gdrcuportal [关闭页面](#)

Web 服务器

Web 服务器
使用此页面来查看已安装的 Web 服务器列表。

首选项

生成插件

传播插件

新建

删除

模板...

启动

停止

终止



选择	名称	Web 服务器类型	节点	主机名	版本
<input type="checkbox"/>	portalserver	IBM HTTP Server	eostest_portal_Node01	eostest	ND 7.0

总数 1

图表 33 gdrcuportal 概要的 Web 服务器

单元 = eostest_gdrcuplatform_Cell01, 概要文件 = gdrcuplatform [关闭页面](#)

Web 服务器

Web 服务器
使用此页面来查看已安装的 Web 服务器列表。

首选项

生成插件

传播插件

新建




删除

模板...

启动

停止

终止



选择	名称	Web 服务器类型	节点	主机名
<input type="checkbox"/>	platformserver	IBM HTTP Server	eostest_gdrcuplatform_Node01	eostes

总数 1

图表 34 gdrcuplatform 概要的 Web 服务器

两个不同的 Web 服务器中对应的插件配置文件目录如下图所示：

```

eostest:/usr/IBM/HTTPServer/Plugins/config$ll
total 0
drwxr-xr-x  2 eosdb  eosdb          256  8月08 16时36 platformserver
drwxr-xr-x  2 eosdb  eosdb          256  8月09 16时59 portal
drwxr-xr-x  2 eosdb  eosdb          256  8月08 15时46 portalserver
drwxr-xr-x  2 eosdb  eosdb          256  8月08 15时24 templates
drwxrwxr-x  2 eosdb  eosdb          256  8月08 15时24 webserver1

```

图表 35 查看插件配置文件

IBM HTTP Server 的 httpd.conf 中加载的插件配置文件只会加载其中一个 Web 服务器的插件配置文件，如下图所示：

```

LoadModule was_ap22_module /usr/IBM/HTTPServer/Plugins/bin/mod_was_ap22_http.so
LoadModule substitute_module modules/mod_substitute.so
WebSpherePluginConfig /usr/IBM/HTTPServer/Plugins/config/platformserver/plugin-cfg.xml

```

图表 36 加载插件配置

通过查找 IBM 的相关帮助文档说明，需要将其他的 概要文件的插件配置合并到某一个概要文件的插件配置文件中，本次示例是将 Portal 的 Web 服务器配置合并到综合管理平台的 Web 服务器插件配置文件中，操作步骤如下：

第一步、复制每个概要文件的 plugin-cfg.xml 文件到某个工作目录中。

第二步、将每个 plugin-cfg.xml 文件的多个副本合并成单个文件，命名为 plugin-cfg.xml，然后用合并后的 plugin-cfg.xml 替换掉 IBM HTTP Server 目录中配置文件加载的 plugin-cfg.xml 文件。要合并文件，务必从不同的 plugin-cfg.xml 文件中捕获以下这些配置项：

A. 将每个功能部件概要文件配置文件中 <VirtualHost> 元素添加到合并后文件中的 <VirtualHostGroup Name="default_host"> 元素块。例如，以下这组虚拟主机值代表单个功能部件概要文件的一组值。您的端口号将根据配置而与这里的不同：

```

<VirtualHostGroup Name="default_host">
  <VirtualHost Name="*:9081"/>
  <VirtualHost Name="*:80"/>
  <VirtualHost Name="*:9444"/>
  <VirtualHost Name="*:5060"/>
  <VirtualHost Name="*:5061"/>
  <VirtualHost Name="*:443"/>
</VirtualHostGroup>

```

B. 将 <ServerCluster> 元素从每个功能部件概要文件的配置文件复制到合并后

的文件。例如：

```
<ServerCluster CloneSeparatorChange="false" GetDWLMTable="false"
IgnoreAffinityRequests="true" LoadBalance="Round Robin"
Name="server1_eostest_portal_Node01_Cluster" PostBufferSize="64" PostSizeLimit="-1"
RemoveSpecialHeaders="true" RetryInterval="60" ServerIOTimeoutRetry="-1">
    <Server ConnectTimeout="0" ExtendedHandshake="false" MaxConnections="-1"
Name="eostest_portal_Node01_server1" ServerIOTimeout="0" WaitForContinue="false">
        <Transport Hostname="eostest" Port="9081" Protocol="http"/>
        <Transport Hostname="eostest" Port="9444" Protocol="https">
            <Property Name="keyring"
Value="/usr/IBM/HTTPServer/Plugins/config/portalserver/plugin-key.kdb"/>
            <Property Name="stashfile"
Value="/usr/IBM/HTTPServer/Plugins/config/portalserver/plugin-key.sth"/>
        </Transport>
    </Server>
</ServerCluster>
```

C. 将 **<UriGroup>** 元素从每个功能部件概要文件的配置文件复制到合并后的文件。例如：

```
<UriGroup Name="default_host_server1_eostest_portal_Node01_Cluster_URLs">
    <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/ivt/*"/>
    <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid"
Name="/portal/AboutPortlet/*"/>
    <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid"
Name="/portal/*"/>
    <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid"
Name="/snoop/*"/>
    <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/hello"/>
    <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid"
Name="/hitcount"/>
    <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="*.jsp"/>
    <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="*.jsv"/>
    <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="*.jsw"/>
    <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid"
Name="/j_security_check"/>
    <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid"
Name="/ibm_security_logout"/>
    <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid"
Name="/servlet/*"/>
    <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid"
```



```
Name="/sso-server/*/"/>
</UriGroup>
```

D. 将 <Route> 元素从每个功能部件概要文件的配置文件复制到合并后的文件。

例如：

```
<Route ServerCluster="server1_eostest_portal_Node01_Cluster"
UriGroup="default_host_server1_eostest_portal_Node01_Cluster_URIs"
VirtualHostGroup="default_host"/>
```

注意：请勿将更多的概要文件映射到 **IBM HTTP Server**，否则将覆盖刚刚编辑过的文件。

两个节点（一个节点上安装了 Portal，另一个节点上安装了综合管理平台黄色背景部份）的合并 plugin-cfg.xml 文件示例。

```
<?xml version="1.0" encoding="UTF-8"?><!--HTTP server plugin config file for the webserver
eostest_gdrcuplatform_Cell01.eostest_gdrcuplatform_Node01.platformserver generated on
2013.08.08 at 03:58:58 PM GMT+08:00-->
<Config ASDisableNagle="false" AcceptAllContent="false"
AppServerPortPreference="HostHeader" ChunkedResponse="false" FIPSEnable="false"
FailoverToNext="false" HTTPMaxHeaders="300" IISDisableFlushFlag="false"
IISDisableNagle="false" IISPluginPriority="High" IgnoreDNSFailures="false"
KillWebServerStartUpOnParseErr="false" MarkBusyDown="false"
OS400ConvertQueryStringToJobCCSID="false" RefreshInterval="60" ResponseChunkSize="64"
SSLConsolidate="true" TrustedProxyEnable="false" VHostMatchingCompat="false">
  <Log LogLevel="Error"
Name="/usr/IBM/HTTPServer/Plugins/logs/platformserver/http_plugin.log"/>
    <Property Name="ESIEnable" Value="true"/>
    <Property Name="ESIMaxCacheSize" Value="1024"/>
    <Property Name="ESIInvalidationMonitor" Value="false"/>
    <Property Name="ESIEnableToPassCookies" Value="false"/>
    <Property Name="ESICacheidFull" Value="false"/>
    <Property Name="PluginInstallRoot" Value="/usr/IBM/HTTPServer/Plugins/" />
    <VirtualHostGroup Name="default_host">
      <VirtualHost Name="*:9081"/>
      <VirtualHost Name="*:9082"/>
      <VirtualHost Name="*:80"/>
      <VirtualHost Name="*:8080"/>
      <VirtualHost Name="*:9444"/>
      <VirtualHost Name="*:9445"/>
      <VirtualHost Name="*:5060"/>
      <VirtualHost Name="*:5061"/>
```

```

    <VirtualHost Name="*:443"/>
  </VirtualHostGroup>
  <ServerCluster
    CloneSeparatorChange="false"
    GetDWLMTable="false"
    IgnoreAffinityRequests="true"
    LoadBalance="Round"
    Robin"
    Name="server1_eostest_gdrcuplatform_Node01_Cluster"
    PostBufferSize="64"
    PostSizeLimit="-1"
    RemoveSpecialHeaders="true"
    RetryInterval="60"
    ServerIOTimeoutRetry="-1">
    <Server
      ConnectTimeout="0"
      ExtendedHandshake="false"
      MaxConnections="-1"
      Name="eostest_gdrcuplatform_Node01_server1"
      ServerIOTimeout="0"
      WaitForContinue="false">
      <Transport Hostname="eostest" Port="9082" Protocol="http"/>
      <Transport Hostname="eostest" Port="9445" Protocol="https">
      <Property
        Name="keyring"
        Value="/usr/IBM/HTTPServer/Plugins/config/platformserver/plugin-key.kdb"/>
      <Property
        Name="stashfile"
        Value="/usr/IBM/HTTPServer/Plugins/config/platformserver/plugin-key.sth"/>
      </Transport>
    </Server>
  </ServerCluster>
  <ServerCluster
    CloneSeparatorChange="false"
    GetDWLMTable="false"
    IgnoreAffinityRequests="true"
    LoadBalance="Round"
    Robin"
    Name="server1_eostest_portal_Node01_Cluster"
    PostBufferSize="64"
    PostSizeLimit="-1"
    RemoveSpecialHeaders="true"
    RetryInterval="60"
    ServerIOTimeoutRetry="-1">
    <Server
      ConnectTimeout="0"
      ExtendedHandshake="false"
      MaxConnections="-1"
      Name="eostest_portal_Node01_server1"
      ServerIOTimeout="0"
      WaitForContinue="false">
      <Transport Hostname="eostest" Port="9081" Protocol="http"/>
      <Transport Hostname="eostest" Port="9444" Protocol="https">
      <Property
        Name="keyring"
        Value="/usr/IBM/HTTPServer/Plugins/config/portalserver/plugin-key.kdb"/>
      <Property
        Name="stashfile"
        Value="/usr/IBM/HTTPServer/Plugins/config/portalserver/plugin-key.sth"/>
      </Transport>
    </Server>
  </ServerCluster>
  <UriGroup Name="default_host_server1_eostest_gdrcuplatform_Node01_Cluster_URIs">
    <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/ivt/*"/>
    <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/snoop/*"/>
    <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/hello"/>
    <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/hitcount"/>
    <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="*.jsp"/>
    <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="*.jsv"/>
    <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="*.jsw"/>
    <Uri
      AffinityCookie="JSESSIONID"
      AffinityURLIdentifier="jsessionid"
      Name="/j_security_check"/>
    <Uri
      AffinityCookie="JSESSIONID"
      AffinityURLIdentifier="jsessionid"

```

```

Name="/ibm_security_logout"/>
  <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/servlet/*"/>
  <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/default/*"/>
  <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid"
Name="/workspace/*"/>
</UriGroup>
  <UriGroup Name="default_host_server1_eostest_portal_Node01_Cluster_URIs">
  <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/ivt/*"/>
  <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid"
Name="/portal/AboutPortlet/*"/>
  <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/portal/*"/>
  <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/snoop/*"/>
  <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/hello"/>
  <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/hitcount"/>
  <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="*.jsp"/>
  <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="*.jsw"/>
  <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="*.jsw"/>
  <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid"
Name="/j_security_check"/>
  <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid"
Name="/ibm_security_logout"/>
  <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/servlet/*"/>
  <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid"
Name="/sso-server/*"/>
</UriGroup>
  <Route ServerCluster="server1_eostest_portal_Node01_Cluster"
UriGroup="default_host_server1_eostest_portal_Node01_Cluster_URIs"
VirtualHostGroup="default_host"/>
  <Route ServerCluster="server1_eostest_gdrcupplatform_Node01_Cluster"
UriGroup="default_host_server1_eostest_gdrcupplatform_Node01_Cluster_URIs"
VirtualHostGroup="default_host"/>
  <RequestMetrics armEnabled="false" loggingEnabled="false" rmEnabled="false"
traceLevel="HOPS">
    <filters enable="false" type="URI">
      <filterValues enable="false" value="/snoop"/>
      <filterValues enable="false" value="/hitcount"/>
    </filters>
    <filters enable="false" type="SOURCE_IP">
      <filterValues enable="false" value="255.255.255.255"/>
      <filterValues enable="false" value="254.254.254.254"/>
    </filters>
    <filters enable="false" type="JMS">
      <filterValues enable="false" value="destination=aaa"/>
    </filters>

```

```
<filters enable="false" type="WEB_SERVICES">  
  <filterValues enable="false" value="wsdlPort=aaa:op=bbb:nameSpace=ccc"/>  
</filters>  
</RequestMetrics>  
</Config>
```