How to enable UMS ExtraLogging channels

Contents

UMS ExtraLogging

Version	R21sp1 PatchID	API Attributes:	Channel	To Troubleshoot:
v1	ap375053	**isBindResourceExtraLoggingActive** **isJabberIqAuthExtraLoggingActive**	**bindResourceExtraLoggingActive** **jabberIqAuthExtraLoggingActive**	XMPP-Bind issue
v2	ap374519	**isJabberIqRosterExtraLoggingActive** **isRosterFlatExtraLoggingActive**	**jabberIqRosterExtraLoggingActive** **rosterFlatExtraLoggingActive**	Roster issue
v3	ap377180	**isMessageExtraLoggingActive** **isMessageHistoryExtraLoggingActive**	**messageExtraLoggingActive** **messageHistoryExtraLoggingActive**	message-to-message issue (User A o and user B on N2 are chatting)
v4	ap377373	**isReceiverBareJidLBExtraLoggingActive** **isComponentProtocolExtraLoggingActive**	**receiverBareJidLBExtraLoggingActive** **componentProtocolExtraLoggingActive**	node-disconnect, MUC* & XMPP_Bir issues

* Only for system having MUC configured as external component under:

UMS_CLI/System/ProfileTuning/GeneralSettings> g
profileTuningName = mucExternal

According to the test requested by TAC, you will need to enable/disable one or more of the troubleshooting channels using the related API Attributes.

Note the customers will be unable to enable those ExtaLogging IMPLog channels with the original test instructions from those patches.

The original patches release Notes "test instructions" contain improper information, plus the API have change over time to be able to survive restart of the UMS server

Here are the right instruction on using API /gateway/implog

1. To enable extra IMP logging, you need to send a POST HTTP command with the name of the API Attributes matching the channel you need. Here is an example for channel componentProtocolExtraLoggingActive:

```
POST http://xx.xxx.xxx.xxx/gateway/implog/isComponentProtocolExtraLoggingActive
```

2. To verify how the channels configuration, you need to send the following API command:

```
GET http://xx.xxx.xxx.xxx/gateway/implog
```

Response from the server will have the channels listed showing individual configuration:

```
Response: { "status":{ "code":"0300001", "type":"success", "message":"LogState Get
Successfully!" }, "impLogState":{ "bindResourceExtraLoggingActive":false,
"jabberIqAuthExtraLoggingActive":false, "jabberIqRosterExtraLoggingActive":false,
"rosterFlatExtraLoggingActive":false, "messageExtraLoggingActive":false,
"messageHistoryExtraLoggingActive":false, "receiverBareJidLBExtraLoggingActive":false,
"componentProtocolExtraLoggingActive":true } }
```

5. To disable an extra IMP logging channel, you need to send a DELETE HTTP command with the name of the API Attributes matching the channel you need.:

```
DELETE http://xx.xxx.xxx.xxx/gateway/implog/isComponentProtocolExtraLoggingActive
```

Here are examples using CURL commands to use the ExtraLogging IMPLog API:

1) confirm status:

```
curl -X GET -u '<USERID>' http://localhost/gateway/implog/
```

2) enable a channel:

```
curl -X POST -H "Content-Type: application/json" -u '<USERID>'
http://localhost/gateway/implog/isMessageExtraLoggingActive
```

3) disable a channel:

```
curl -X DELETE -H "Content-Type: application/json" -u '<USERID>'
http://localhost/gateway/implog/isMessageExtraLoggingActive
```

USERID:

For all HTTP requests an authorized user is required. Your UMS system already has a script that allows you to set permissions.

For more details, please review this file avaiable from the UMS server:
/usr/local/broadworks/UMS_Rel_21.sp1_1.551/sbin/authorization/README-authorization.txt

IMPORTANT:

You need to configure each UMS server individually to have the channels on both servers, since the API attributes configuration is not synchrozined between them.

Note: After a server restart, the *API Attributes* states remain as it has been previously set.