



R4.2.2 Application Enablement Services Software Only Server, Bundled Server and Client Release Notes

June 2009

INTRODUCTION

This document introduces the Generally Available release of the Application Enablement (AE) Services Release 4.2.2 and describes important notes and known issues. The *Avaya MultiVantage® Application Enablement Services Overview* has a complete list of 4.X features.

What's Changed In AE Services 4.2.2?

- IBM Sametime 8.0.2 is now supported.
- Microsoft OCS 2007 R2 is now supported.
- AE Services 4.2.2 is a bug fix release for AE Services 4.0, 4.0.1, 4.1, 4.2 and 4.2.1.
- AE Services 4.2.2 supports Red Hat Enterprise Linux 4.0 Update 7.
- AE Services 4.2.2 is compatible with the following Bundled Servers:
 - IBM x306 (S8500B)
 - IBM x306m (S8500C)
 - Dell 1950 (S8510)
- JTAPI sample applications have been moved from applet to swing.

SOFTWARE RELEASE VERSIONS

Application Enablement Services Application	File Name
CVLAN Client Linux	cvlan-client-linux-4.2-451.i386.rpm
CVLAN Client Windows	cvlan-client-win32-4.2-451.zip
TSAPI Client Linux	tsapi-client-linux-4.2-338.i386.rpm
TSAPI Client MS Windows	tsapi-client-win32-4.2-451.zip
TSAPI SDK Linux	tsapi-sdk-linux-4.2-451.i386.rpm
TSAPI SDK MS Windows	tsapi-sdk-win32-4.2-451.zip
JTAPI OS Independent Client	jtapi-client-osindependent-4.2-451.zip
JTAPI OS Independent SDK	jtapi-sdk-osindependent-4.2-451.zip
JTAPI MS Windows 32-BIT Client	jtapi-client-win32-4.2-451.zip

Application Enablement Services Application	File Name
JTAPI MS Windows SDK	jtapi-sdk-win32-4.2-451.zip
IPCommunications SDK (System Management Services)	smssvc-sdk-4.2.2.482.zip
IPCommunications SDK DMCC dotnet SDK	dmcc-dotnet-sdk-4.2.2.60.zip
IPCommunications SDK Telephony Services	telsvc-sdk-4.2.2.482.zip
IPCommunications SDK (Device and Media Control/DMCC)(XML)	cmapixml-sdk-4.2.2.482.zip
IPCommunications SDK (Device and Media Control/DMCC)(Java)	cmapijava-sdk-4.2.2.482.zip
Application Enablement Services MIB	aesvcs-product-mibs-4.2.2.482.zip
Server Software for the Software Only Solution	swonly-r4-2-2-31-20090519.iso
Server Software for Bundled Solution	bundled-r4-2-2-31-20090519.iso

IMPORTANT NOTES

AE Services 4.2.2 is compatible with the following Communication Manager Releases and Platforms:

- Communication Manager 3.1.x (G3csi, S8300, S8400, S8500, S87xx)
- Communication Manager 4.x (S8300, S8400, S8500, S87xx)
- Communication Manager 5.0 (S8300, S8400, S8500, S87xx)
- Communication Manager 5.1 (S8300, S8400, S85xx, S87xx)
- Communication Manager 5.2 (S8300, S8400, S85xx, S87xx)

Communication Manager 5.2 is compatible with the following AE Services Releases:

- AE Services 3.1.x
- AE Services 4.x.x

Note: MAPD is NOT supported on Communication Manager 5.x

Release History:

Date	Build	Change(s)
03/2007	47-3	General Availability R4.0
06/2007	50-1	General Availability R4.0.1
12/2007	31-2	General Availability R4.1
04/2008	4.1.16	General Availability R4.1.1 JTAPI Client/SDK
05/2008	19-4	General Availability R4.2
08/2008	20-5	Service Pack R4.2.1
06/2009	31	Service Pack R4.2.2

KNOWN ISSUES AND WORKAROUNDS

- **CVLAN Services Does Not Display Online**

If the AE Server has no CVLAN links administered, the controller status of CVLAN Service will appear as running under CTI-OAM home, but will appear as OFFLINE under the services summary. The service will appear as online after you add a CVLAN link.

- **DLG Links**

DLG links may be OFFLINE after recovery from an abnormal shutdown.

- **DLG Service Does Not Display Online**

If the AE Server has no DLG links administered, the controller status of DLG Service will appear as running under CTI-OAM home, but will appear as OFFLINE under the services summary. The service will appear as online after you add a DLG link.

- **DMCC – New CSTA Private Error Code**

A new CSTA private error code has been added for DMCC clients to resolve a request timeout when the request contains invalid UTF-8. For client applications that use the Java SDK, the error will appear as a `com.avaya.csta.errors.PayloadDecodeException`. Users of the XML SDK will see this as a `CSTAErrorCode` with a `privateError` code value of 8.

- **Modem Lock-Up During Server Reboot**

An issue has been identified with the USB modem MT5634ZBA equipped on the AE Services 4.x bundled servers (running the Linux kernel 2.6.9). The USB modem may lock-up if it is in ringing state during the server reboot process. To avoid modem lock-up, wait for 20 minutes to dial into the modem after every server reboot. Reference PSN002127.

- **OCS Integration and Microsoft Certificate Authorities (CA)**

When using Microsoft as the CA, Microsoft recommends using an Enterprise CA. The Enterprise CA template used to create the AE Services certificate must have the Enhanced Key Usage (EKU) field specified appropriately (Server and Client Auth or neither).

The LCS/OCS AE Services integration uses Mutual TLS (MTLS) to authenticate server to server SIP communication. On an MTLS connection, the server originating a message and the server receiving it exchange certificates from a mutually trusted CA to prove the identity of each server to the other.

The server certificate used for MTLS on both servers must either not specify an Extended Key Usage (EKU) or specify an EKU for Server and Client Auth.

When the EKU is not specified the certificate is not restricted to a particular usage. However when the Key Usage field is specified and the EKU is specified as Server and Client Auth, the certificate can only be used by the server for mutual server and client based authentication purposes. If an EKU with only Server Auth is specified, in this scenario, the connecting server certificate will fail authentication and the MTLS connection will not be established.

The Standalone CA, which may also be used (but is not Microsoft recommended), does not provide configurable templates including some additional features and must adhere to the same certificate generation rules in regards to the EKU field.

Note that this statement doesn't preclude administrators from using non-Microsoft CA's (e.g. VeriSign).

- **SAMP**

An intermediate SAMP firmware upgrade will be required if SAMP firmware AVAYA_S8500_1_0_SP2_BUILD_17 or lesser is found on AE Services 3.x bundled servers. Manually reload the SAMP firmware to AVAYA_S8500_1_0_SP4 prior to upgrading to AE Services 4.x. Reference the details in PSN0002128.

- **SNMP**

While technically correct, the alarm log and trap information is somewhat misleading in that the mon daemon's configuration file set high CPU utilization to greater than 90% not the 80% as is reported by the logs, traps and alarms.

- **Process to Change the Server IP Address**

If the IP address of an AE Services server is changed without stopping the server or if the IP address is changed and then an attempt is made to set the new address through the web pages without stopping the server service (which is using the connection), an error message will be displayed. The error message will appear on the Local IP web page and indicate that the database entry for the IP address does not match the IP address configured on the server. The proper procedure to change the IP address is as follows:

For AE Services bundled server:

1. Log in as sroot.
2. Issue "service mvap stop"
3. Execute "/opt/mvap/bin/netconfig" to bring up GUI.
4. Enter/Modify IP address(es) per NIC interface (Make sure "Enable" boxes are checked), and save/exit by "OK" button.
5. Re-login as sroot with new IP address if administering remotely. Issue "service network restart" (This step is pre-cautionary).

6. Log into OAM “AE Server Administration” using new IP address of AE SERVICES server. (Note: If OAM web page is not responding in proper time, issue “service tomcat5 restart” and try again).
7. Go to “Administration” -> “Network Configuration” -> “Local IP” and set the new IP address(es) for all “Connectivity” entries.
8. Execute the script “/opt/mvap/bin/setAlarmSvcUpgrade.sh” .
9. Go to “Maintenance” -> “Service Controller” and apply “Restart AE Server” button.
10. Make sure all services are in “Running” state, and connection state to switch(es) is functional.

For AE Services Software-Only server:

1. Log in as root.
2. Issue “/sbin/service mvap stop”
3. *** Customer is responsible for the change, but highly recommend using Linux utility such as /usr/bin/system-config-network if using Redhat Release 4. /etc/hosts file must be updated with new IP address(es). Issue “/sbin/service network restart” (This step is pre-cautionary). ***
4. Log into OAM “AE Server Administration” using new IP address of AE Services server. (Note: If OAM web page is not responding in proper time, issue “/sbin/service tomcat5 restart” and try again).
5. Go to “Administration” -> “Network Configuration” -> “Local IP” and set the new IP address(es) for all “Connectivity” entries.
6. Execute the script “/opt/mvap/bin/setAlarmSvcUpgrade.sh”.
7. Go to “Maintenance” -> “Service Controller” and apply “Restart AE Server” button.
8. Make sure all services are in “Running” state, and connection state to switch(es) is functional.

Note: If the OAM page cannot be accessed, check status of httpd/tomcat5 processes via “/sbin/service httpd status” and “/sbin/service tomcat5 status”, and start if not running.

- **Sametime Upgrade from 8.0 to 8.02 Loses the Telephony Service Provider Policy Setting**

When Sametime is upgraded from 8.0.0 to 8.0.2, the telephony service provider is not enabled as a default. Re-enable the Telephony Service Provider field in the Sametime Policy.

The IBM SPR is SLEE7NKJRJ.

- **Sametime Connect 8.0.2 – Calls Cannot be Made to the Client When “Display Incoming Invitation” is Unchecked**

When a Sametime client unchecks "Preferences->Notifications->Telephony notifications->Display incoming invitation" in Sametime Connect 8.0.2, Sametime calls cannot be made to that particular client. IBM has provided Hotfix # DAMD-7NJKJA.

- **SIP Issues**

- When using 3rd party call control to make a call on a SIP endpoint to a VDN that has a vector step to collect digits after an announcement, the announcement will not be played and the digits entered will not be forwarded.
- When using 3rd party call control to make a direct agent call to a busy agent on a SIP endpoint the call drops. The workaround is to place the call manually.
- When using 3rd party call control to make a call using a TAC, the call will fail on a SIP phone if the Communication Manager does not have a TN2602AP board.
- When using 3rd party call control to answer and place a call on hold on a SIP endpoint any attempt to make another call from that SIP endpoint will fail.
- If Communication Manager does not have a TN2602AP board, the media encryption on the SIP endpoint should be disabled. The SIP endpoint transport type must be set to TCP or UDP. If transport type is set to TLS, the 3rd party call control application may fail during transfer and conference.
- The Single Step Transfer service does not work reliably for SIP stations.
- DTMF Tones are not supported on SIP endpoints.
- User classified call does not generate an ALERTING event over domain control association.
- Going off-hook on a SIP station followed by on-hook does not generate an INITIATED event.
- Using Third party Call control when a call is made from a SIP station, the INITIATED event is slightly delayed as compared to other station types. Subsequent events are not delayed.
- ACD calls that are delivered to SIP endpoints are generating Alerting Event reports that do NOT contain the split/skill extension from the associated call.

- **WebLM Enterprise Model – Using HTTPS**

Run this workaround if all three of the following conditions are true:

1. The master WebLM Server, which hosts the Enterprise License File (ELF), is not co-located with an AE Services Server,
2. Local WebLM servers are co-located with AE Services
3. HTTPS is in use for communication between the master and local WebLM Servers [for example, to push an Allocation License File (ALF) to a local WebLM server on AE Services].

Locate the Enterprise Web Licensing WebLM Patch. The name of the file is ImportCertToWebLM.zip in the patch directory on the Bundled ISO and in the root directory of the SW-Only ISO.

1. Download importCertToWebLm.zip files to your EWL server.
2. Unzip the file
3. Follow the directions in the README to install

- **WebLM Session May Hang**

Performing one of the following actions on WebLM may hang the session.

1. Repeatedly uninstall and install licenses
2. Repeatedly refreshing license page

The current session should be closed and a new session opened.

INSTALLATION NOTES

Sametime

AE Services supports integration with IBM Lotus Sametime Version 7.5.1 and 8.0. The following software from IBM is required. In addition, the Avaya AE Services Sametime zip file is needed to complete the installation and setup of this feature.

- Lotus Sametime
 - Lotus Sametime Server 7.5.1 or 8.0 (Windows only)
 - Lotus Sametime Clients 7.5.1 or 8.0 (Windows, Linux, Mac OS X)
 - Lotus Sametime 8.0 SDK (Windows, Linux, Mac OS X) (works with 7.5.1 and 8.0 clients)
- Avaya AE Services Sametime Integration
 - avaya-aes-sametime.zip can be downloaded from the following Web Site:
http://devconnect.avaya.com/AEServices_LotusSametime

For complete installation and setup instructions, see the following document:

Avaya Application Enablement Services

Release 4.2

Integration Guide for IBM Sametime

RESOLVED ISSUES IN AE SERVICES RELEASE 4.2.2

ASAI Link Service:

Previously, the ASAI Link Service was aging Call Reference Values too quickly. This caused the ASAI Link Service to generate a segmentation fault when it received a message from Communication Manager associated with a Call Reference Value that had been marked as 'stale'.

CVLAN Service:

- Previously, the OAM CVLAN test did not display results for a specific link.
- Previously, the CVLAN Service would generate a segmentation fault when the "ntpd" daemon reset the time on the AE Server backward by 1 hour.
- Previously, if the CVLAN Service was not able to contact WebLM after a reboot of the AE Server, the CVLAN Service would go offline.

CVLAN SDK:

- Additional protection against improper shutdown has been provided. Previously, customers using some applications were receiving errors and failing due to improper shutdown. The CVLAN client library now detects this condition and no unexpected errors are returned.
- CVLAN secure connections were not working. Customers who received patch to fix this issue could not longer establish secure connections with CVLAN. This problem was fixed with a subsequent patch to Release 4.2.1 and is fixed in Release 4.2.2. Note: This fix requires both a client and server upgrade.

DLG Service:

Previously, if the DLG Service was not able to contact WebLM after a reboot of the AE Server, the DLG Service could default to a 30-Day Temporary license.

DMCC:

- Previously, if a DMCC endpoint was registered in "dependent" or "independent" mode, when it went off-hook, AE Services would send a flood of HookSwitchEvents to the client application. This issue has been resolved, so that only one HookSwitchEvent is sent to the client.
- If a DMCC endpoint has subscribed for media events, it will receive a MediaStartEvent from AE Services when a call to the endpoint is initiated. Previously, the MediaStartEvent incorrectly reported the far-end RTP socket address:port combination to be the same as the RTCP socket address:port. This issue has been resolved, so that the RTP and RTCP are each reported as having a unique address:port combination

- Previously, if two applications were invoking GetCallInformation at the same time, with the same invoke ID, it was possible to experience a race condition whereby one of the applications would not receive a response to their request. This could happen in the case of a duplicated application with two instances, each of which is always invoking the same requests as a result of receiving the same events.
- Previously, in certain cases with calls involving more than 2 parties, DMCC would incorrectly process a snapshot call response and throw an Exception. This Exception was not handled cleanly, and all Call Control monitors would be lost as a result of this error. The application would also never see the Snapshot Call response and would timeout
- Previously, when an Active/LDAP directory contained invalid UTF-8 for a number of phone numbers, the DMCC service CstaPacketFormat.decode would throw a PacketDecodeException. Since DMCC did not know the type of the message sent by the client, DMCC would not send a response and the request timed out.
- Previously, AE Services 4.x did not support SIP UPDATE message. Beginning with AE Services 4.2.1 Patch 2 and 4.2.2, AE Services will support SIP UPDATE message with Microsoft OCS 2007 R2 integrations.
- Microsoft OCS 2007 R2 uses the SIP UPDATE message rather than RE-INVITE to refresh its session resulting in sessions expiring within 30 minutes.

JTAPI Client/SDK:

- Previously, under some circumstances, the JTAPI client library would throw a java.util.ConcurrentModificationException while processing a removeCallObserver() request and, as a result, would shutdown the provider.
- Previously, under some circumstances, the query AgentTerminal.getAgents() would return NULL when it should have returned an array of agent objects.
- The JTAPI client library has reduced the number of internal queries it performs, improving its performance.
- Previously, there was no limit to the number of threads that JTAPI would create for the purpose of delivering events to observers. This could adversely affect performance as the number of threads increased. JTAPI now has a property 'maxThreadPoolSize' to configure the number of threads created for this purpose and its default value is 20.

- Previously, under some circumstances, JTAPI would create a TerminalConnection object for a connection at an external device when it should have created a Connection object.
- Previously, when a call that came through a VDN being monitored by a JTAPI application was transferred to a second VDN that is being monitored by a different application, the first JTAPI application would not clean up its data structures for the call.

Office Communicator 2007:

Previously, when an outbound inter-switch call was placed, the name of the called party was displayed when the call was ringing. When the called party answered, the name would change to “Unidentified Caller”. This has been fixed so that the name of the called party is displayed (instead of “Unidentified Caller”) when the called party answers.

NIC Configurations Not Saved

In AE Services 3.x releases, the NIC configurations (auto negotiation, speed, and duplex) were not preserved in AE Services Database. Starting with AE Services 4.0, the NIC interface can be configured via Web OA&M page and the settings are stored in database. For those servers upgraded from 3.x, all the NIC interfaces have to be re-configured via Web OA&M page.

Sametime:

- Previously starting or restarting the Presence Service during business hours resulted in undue competition on the AE Services server between the Conference Service and Presence Service. It was therefore advised that the administrator only start or restart the Presence Service after business hours. The undue competition for resources has been eliminated in AE Services Release 4.2.2.
- Previously the name of a TSAPI switch link used by the AE Services IBM Sametime Integration could not contain any uppercase characters. (This is the name administered in Administration > Switch Connections in the AE Services OA&M.) This restriction has been removed in AE Services Release 4.2.2.

Security:

The following Red Hat Linux security issues have been incorporated into Release 4.2.2:

- [RHSA-2008:0607-01] Important: kernel security and bug fix update
- [RHSA-2008:0967-01] Moderate: httpd security and bug fix update
- [RHSA-2009:0014-01] Important: kernel security and bug fix update
- [RHSA-2008:0055-01] Important: kernel security and bug fix update
- [RHSA-2008:0110-01] Moderate: openldap security update

- [RHSA-2007:0966-01] Important: perl security update
- [RHSA-2007:1052-01] Critical: pcre security update
- [RHSA-2007:1038-01] Moderate: openldap security and enhancement update
- [RHSA-2007:1045-01] Moderate: net-snmp security update
- [RHSA-2007:0969-01] Moderate: util-linux security update
- [RHSA-2007:1003-02] Moderate: openssl security and bug fix update
- [RHSA-2007:0747-02] Moderate: httpd security, bug fix, and enhancement update
- [RHSA-2007:0737-02] Moderate: pam security, bug fix, and enhancement update
- [RHSA-2007:0703-02] Moderate: openssh security and bug fix update
- [RHSA-2007:0387-02] Moderate: tcpdump security and bug fix update
- [RHSA-2008:0167-01] Moderate: kernel security and bug fix update
- [RHSA-2008:0180-01] Critical: krb5 security update
- [RHSA-2008:0665-01] Moderate: Updated kernel packages for Red Hat Enterprise Linux 4.7
- [RHSA-2008:0836-02] libxml2 security update
- [RHSA-2008:0508-01] Important: kernel security and bug fix update
Severity
- [RHSA-2007:1068-01] Important: pcre security update
- [RHSA-2007:0203] unzip security and bug fix update hyoungjoolee
- [RHSA-2007:0391] file security update
- [RHSA-2007:0968-01] Critical: pcre security update
- [RHSA-2008:0529-01] Moderate: net-snmp security update
- [RHSA-2007:0795-01] Moderate: cyrus-sasl security and bug fix update
- [RHSA-2007:0488] kernel security update
- [RHSA-2008:0893-01] Moderate: bzip2 security update
- [RHSA-2007:0220] GCC security and bug fix update
- [RHSA-2007:0662] http security update
- [RHSA-2007:0562] krb5 security update
- [RHSA-2008:0715-01] Low: nss_ldap security and bug fix update
- [RHSA-2008:0884-01] Important: libxml2 security update
- [RHSA-2008:0003-01] Moderate: e2fsprogs security update
- [RHSA-2008:0032-01] Important: libxml2 security update
- [RHSA-2008:0006-01] Moderate: httpd security update
- [RHSA-2008:0617-01] Moderate: vim security update
- [RHSA-2008:0972-01] Important: kernel security and bug fix update
- [RHSA-2008:0237-01] Important: kernel security and bug fix update
- [RHSA-2009:0004-01] Important: openssl security update
- [RHSA-2008:0971-01] Important: net-snmp security update
- [RHSA-2008:0988-01] Important: libxml2 security update
- [RHSA-2007-0326] tomcat security update
- [RHSA-2008:0946-01] Moderate: ed security update

- [RHSA-2009:0020-01] Moderate: bind security update
- [RHSA-2008:0780-01] Low: coreutils security update
- [RHSA-2008:0583-01] Important: openldap security update

SMS:

- SMS has added enable/disable support for the ip-reg-tti command on SAT via the new IPRegTTI model.
- SMS previously could not parse an '=' as a parameter to a model's field. This was observed in various fields of the Vector model which used the '=' and similar comparison type characters for fields data. SMS now uses a special character sequences to represent these types of characters. For more details please consult the Vector model documentation in the SMS SDK.
- The IP Video field can now be accessed via the Station model. For more details please consult the Station model documentation in the SMS SDK.
- Previously the RegisteredIPStations model was missing the required fields to retrieve data on Shared Control Devices. For more details please consult the RegisteredIPStations model documentation in the SMS SDK.
- Talk-path string was added to the StationStatus model in SMS. Users can now query SMS for RTP information on active calls (codec, packet size, etc). For more details please consult the StationStatus model documentation in the SMS SDK.
- The Extension field can now be accessed via the agent model. For more details please consult the agent model documentation in the SMS SDK.

TSAPI Service:

- Previously, a duplicate key database insertion error sometimes occurred when processing a "User Status" request from the TSAPI Service Details OAM page.
- Previously, the TSAPI Service did not always take the appropriate action if an error occurred while sending a message to a client.
- Previously, the Conferenced events generated for a single-step conference operation that included a predictive call did not always contain all of the connections on the call.

- Previously, in some scenarios involving stations with bridged appearances, it was only possible to perform a single single-step transfer operation; subsequent single-step transfer operations would fail. Note that the single-step transfer call service is used by Microsoft Office Communicator.
- Previously, in some single-step transfer call scenarios involving stations with bridged appearances, the TSAPI Service would provide the wrong calling device in TSAPI events. Note that the single-step transfer call service is used by Microsoft Office Communicator.
- Previously, in some predictive call scenarios the TSAPI Snapshot Call service would not return all of the parties on the call.
- Previously, the TSAPI Service would indicate that the reason for Original Call Information in an event was OR_NEW_CALL when there was no Original Call Information in the event. Now the TSAPI Service indicates reason OR_NONE when there is no Original Call Information in an event.
- Previously, in some scenarios the TSAPI Snapshot Call service would indicate that the local connection state of an alerting bridged appearance was CS_NULL when it should have indicated that its connection state was CS_ALERTING.
- Previously, in some scenarios the TSAPI Snapshot Call service would indicate that the local connection state of a party on the call was CS_NONE when it should have indicated that its connection state was CS_CONNECTED.
- Previously, the TSAPI Service would not always provide the correct Universal Call ID (UCID) in TSAPI events after a call had been conferenced or transferred.
- Previously, the TSAPI Service did not always provide the correct Universal Call ID in the TSAPI Conferenced and Transferred events.
- Previously, in some conference and transfer scenarios involving bridged appearances, the TSAPI Service Snapshot Call service reported the incorrect Device ID Type for some stations on the call.
- Previously, the Universal Call ID provided in the TSAPI Delivered event resulting from a Single Step Transfer service request was incorrect.
- Previously, in some call scenarios, the TSAPI Service would incorrectly report the Local Connection State of some call parties as CS_ALERTING (alerting) instead of CS_NULL (bridged).

- Previously, in some transfer and conference call scenarios, the TSAPI Service did not correctly indicate the existence of some bridged parties on the call.
- Previously, the agent event filters for a cstaMonitorDevice() request were not always set correctly.
- Within the TSAPI Service, an audit occurs at regular intervals. If this audit determines that there is a problem processing requests for one of the TSAPI CTI links, it will reject any outstanding requests for that CTI link. Previously, these outstanding requests were rejected with CSTA Universal Failure error RESOURCE OUT OF SERVICE. Now these outstanding requests are rejected with CSTA Universal Failure error RESOURCE LIMITATION REJECTION.
- The TSAPI Services uses a new mechanism to control how its threads are scheduled to run.
- Previously, the TSAPI Service would log the warning message "sendToLink:asai_send() failed, asai_errno= -4" when this warning was not appropriate.
- Within the TSAPI Service, trace messages related to the delivery of private data PDUs has been improved.
- Previously, under some scenarios, the TSAPI Service would create a core file during shutdown.
- Previously, in some scenarios, the TSAPI Service could crash when releasing a license.
- Performance of the TSAPI Service has been improved for service requests that do not require a license.
- The TSAPI Service has been enhanced to allow a pool of TSAPI user licenses to be reserved during startup. Reserving user licenses at startup can significantly improve the performance of the TSAPI Service in some customer environments.
- A memory allocation issue within the TSAPI Service has been fixed.
- The TSAPI Service no longer generates the following WARNING messages:
 - Undocumented ASAI cause (0,29) for C_DENIAL(6).
 - Undocumented ASAI cause (0,29) for C_DROP(4)
 - Undocumented ASAI cause (0,41) for C_VQ_CONF(33)

- The CRITICAL error message "cannot get memory from msg heap" has been changed to a WARNING error message, and the text has been changed to: "there is not enough space available in the message buffer for event <event-class-name>(<event-class>) <event-type-name>(<event-type>)."
- Previously, when message tracing was enabled for the TSAPI Service, it was possible for the TSAPI Service to crash when tracing the confirmation message for an extension status value query.
- When message tracing is enabled for the TSAPI Service, the TSAPI Service now includes the private data associated with an ACS Open Stream or ACS Open Stream Confirmation event in the trace output.
- Previously, when message tracing was enabled for the TSAPI Service, Unicode event fields were not properly output in the trace file.
- Previously, when message tracing was enabled for the TSAPI Service, the ATT_SET_AGENT_STATE_CONF event was not properly output in the trace file.
- Previously, on SIP trunk, far-end redirection caused Communication Manager to change trunk party IDs which caused G3PD to send the Delivered Event to the wrong party.
- Previously, if TSAPI clients tried to open more than the maximum number of streams (2,500), an error was returned. This was proper behavior; however, a system resource was not freed as it should be. If the client applications continued to attempt to connect and the resource was exhausted, then the TSAPI service failed to create new monitors to accept new connections (even after other streams are closed), and to update TSAPI OAM pages. This happened after approximately 1,500 such failed requests.

The following error log entry indicates that 2,500 open streams have been exceeded:

```
ERROR:CRITICAL:TSAPI:TSERVER:../ClnMsg.cpp/106 93 Max
tcp connections (2500) is exceeded
```

The following error log entries (each entry applies to separate failure types) may indicate that the resource has been exhausted and the TSAPI service must be restarted:

```
ERROR:WARNING:  
TSAPI:MVAPLicense::acquireTSAPIUsers:getNumAcquired  
failed: Internal error:socket()
```

```
ERROR:CRITICAL:TSAPI:TSERVER:DriverService.cpp/229 93  
accept failed 20
```

```
ERROR:WARNING:TsrvCmdUtility:main:Receive from server  
failed rc= 0, errno= 2
```

```
ERROR:WARNING:TsrvCmdUtility:main:Receive from server  
failed rc= -1, errno= 62
```

DOCUMENTATION UPDATES AND CORRECTIONS

- **Avaya MultiVantage® Application Enablement Services
Installation and Upgrade Guide for a Software-Only Offer
Release 4.2**

Chapter 2: Installing the Linux platform software

Section: Optimizing the Linux software for AE Services

Step 4 on page 16 reads:

“On the Firewall Configuration screen, the Security Enhanced Linux (SELinux) features are active by default. You must disable SELinux or AE Services will not work correctly. Locate the **Enable SELinux** option and select **Disabled**.

! CAUTION:

If you fail to disable SELinux before you install the AE Services software, some AE Services will not start and other problems will occur. To resolve the AE Services problems, you must disable SELinux and then reboot the server. For more information about SELinux, see [Administering SELinux](#) on page 17”

The step should read:

“On the Firewall Configuration screen, the Security Enhanced Linux (SELinux) features are active by default. You must disable SELinux **and the firewall as part of the initial AE Services installation or upgrade process**. Locate the **Enable SELinux** option and select **Disabled**. **In addition, locate the **Enable Firewall** option and select **Disabled**. After the installation or upgrade is complete, the firewall can be properly configured as specified in step 8.**

! CAUTION:

If you fail to disable SELinux **and the firewall** before you install the AE Services software, some AE Services will not start and other problems will occur. To resolve the AE Services problems, you must disable SELinux **and the firewall** and then reboot the server. For more information about SELinux, see [Administering SELinux](#) on page 17 **and page 94 for firewall information.**”

- **Avaya MultiVantage® Application Enablement Services Device, Media and Call Control API Java Programmers Guide Release 4.1 and 4.2**

Chapter 3: Writing a client application

Section: Recovery

Subsection: Transfer Monitor Objects

This section describes how to use the TransferMonitorObjects request for client high availability: enabling the client application to request AE Services to move established registrations and monitors to a different client session. The section does not, however, discuss the use of the TransferProxy service which allows a standby application instance to generate Listener objects for the purposes of receiving events on these already established monitors. For an example of how to use TransferProxy, please see SessionManagement App.java in the SDK sample applications.

- **Avaya Application Enablement Services Release 4.2 Integration Guide for IBM Lotus Sametime**

Chapter 1: System Overview

Section: Known Issues

The following issues have been resolved in AE Services Release 4.2.2:

- Starting or restarting the Presence Service during business hours resulted in the Presence Service competing with the Conference Service for resources on the AE Services server. It was advised that the administrator start or restart the Presence Service after business hours only, in order to provide the best response time to the users of the Conference Service.
- The TSAPI switch links used by the AE Services IBM Sametime Integration must have lowercase names in previous releases. That is, when you administer a switch connection in AE Services (**Administration > Switch Connections**), the connection name must use lowercase characters.
- **Avaya MultiVantage® Application Enablement Services Administration and Maintenance Guide Release**

Chapter 9: Certificate Management

Only one Server Certificate can be installed. Currently, the documentation allows installation of multiple Third Party Server Certificates on AE Services; however, the related functionality is not supported. For example, secure connection only supports one AE Services Server Certificate. Uploading multiple third party Server Certificates is restricted.

- **Avaya MultiVantage® Application Enablement Services Device, Media and Call Control API Java Programmers' Guide Release 4.1 and 4.2**

Chapter 3: Writing a Client Application

Section – Media Encryption

Subsection – Encrypting and Decrypting the RTP Stream

The text in this subsection should be replaced by the following:

1. ENCRYPTING AND DECRYPTING THE RTP STREAM

The encryption transmit and receive keys, along with the roll over counter (ROC) plus the RTP header's SSRC and sequence number, are used to calculate the Initialization Vector.

1.1 ROLL OVER COUNTER (ROC)

The ROC (initially set to zero) is a 32-bit unsigned integer which records how many times the 16-bit RTP sequence number (SEQ) has been reset to zero within the same SSRC (after incrementing up through 65,535). Unlike the sequence number (SEQ), which your secure RTP implementation (SRTP) extracts from the RTP packet header, the ROC is maintained by the SRTP implementation. The ROC must also be knowledgeable of the SSRC that is included in the header of each RTP packet. The SSRC is a 32 bit randomly chosen value in an RTP packet that is used to represent the synchronization source (RFC1889). From one MediaStartEvent to the next MediaStopEvent, the SSRC will remain the same. If the SSRC changes this will be an indication that a new RTP stream has started. When this situation occurs the ROC must be reset to zero.

In Java, this would be something like:

```
// Increment the ROC whenever the sequence number rolls over
incomingReadSSRC = rtpHdr.ssrc;
incomingSeqNum = rtpHdr.seqNum;
if (currentReadSSRC == incomingReadSSRC) {
    if (incomingSeqNum > currentSeqNum) {
        // Do nothing
    } else if (incomingSeqNum < (currentSeqNum - 100)) {
        // Sequence number has probably rolled over
        ++readROC;
    } else {
        // out of sequence RTP packet - ignore it
        return 0;
    }
} else if (incomingReadSSRC == prevReadSSRC) {
    // very late RTP packet from previous call - ignore it
    return 0;
}
```

```

    } else {
        // New SSRC (i.e. new call) - reset ROC
        readROC = 0;
        prevReadSSRC = currentReadSSRC;
        currentReadSSRC = incomingReadSSRC;
    }

```

currentSeqNum = incomingSeqNum;

and similarly for writeROC.

1.2 CREATING THE ENCRYPTION KEYS USING THE PSEUDO RANDOM FUNCTION

The pseudo random function PRF_n(key, x) produces a bit string of length “n” from a string “x” which is encrypted using the encryption key named “key”. The AE Services Symmetric algorithm mode is “ECB” with no padding.

```

private static final byte XeRx[] = {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0};
String algorithm = "aes";
String modeECB = "ECB";
String modeCTR = "CTR";
String padding = "NoPadding";
Cipher cipher = null;
SecretKey key = null;
byte[] KeRx = null;

// Set up and initialize JCE Engine
try {
    String cipherSpec = algorithm + "/" + modeECB + "/" + padding;
    cipher = Cipher.getInstance(cipherSpec);
} catch (Exception e) {
    e.printStackTrace();
}

// Generate the symmetric key using the JCE Engine and the readMasterKey from the
MediaStart
// event and then use the Avaya XeRx value to calculate the KeRx value
key = new SecretKeySpec(readMasterKey, algorithm);

/***** Calculate the KeRX value *****/
try {
    cipher.init(Cipher.ENCRYPT_MODE, key);
    KeRx = cipher.doFinal(XeRx);
} catch (IllegalStateException e) {
    // Attempting to encrypt before the cipher has been initialized.
    // Probably a race condition resulting in the 1st packet not being encrypted.
    // Ignore for now.
} catch (Exception e) {
    e.printStackTrace();
}

```

```
//Print
dump0x("KeRx", KeRx, 0,KeRx.length);
```

And, similarly for KeTx based on the writeMasterKey.

Once the media receive and transmit encryption keys (KeTx and KeRx) are created, they will be used within the AE Services algorithm to encrypt and decrypt the RTP stream.

1.3 CREATING THE INITIALIZATION VECTORS (IV)

The ROC, together with the media encryption keys from the MediaStartEvent, the SSRC and the RTP header sequence number are used to calculate the IV for each direction (transmit & receive):

```
private static final byte XsRx[] = {0, 0, 0, 0, 0, 0, 0, 2, 0, 0, 0, 0, 0, 0, 0};
byte[] KsRx = new byte[14];
ByteBuffer ssrcBuffer = ByteBuffer.allocate(16);
ByteBuffer KsBuffer = ByteBuffer.allocate(16);
ByteBuffer iBuffer = ByteBuffer.allocate(16);

// Convert the RTP header sequence number to bytes
byte[] seqHex = convert2Bytes(seq);

// Clear the local 16-byte buffers used in the IV calculation
ssrcBuffer.clear();
KsBuffer.clear();
iBuffer.clear();

// Softphone PRF version
// Set up and initialize JCE Engine, and generate the symmetric key
// using the JCE Engine and the readMasterKey (from the MediaStart
// event). Then use the Avaya XsRx value to calculate the KsRx value
try {
    cipher.init(Cipher.ENCRYPT_MODE, key);
    KsRx = cipher.doFinal(XsRx);
} catch (IllegalStateException e) {
    // Attempting to encrypt before the cipher has been initialized.
    // Probably a race condition resulting in the 1st packet not being encrypted.
    // Ignore for now.
} catch (Exception e) {
    e.printStackTrace();
}

//Print
dump0x("KsRx", KsRx, 0,KsRx.length);

KsBuffer.put(KsRx, 0, KsRx.length-2);
```

And, similarly for KsTx based on the writeMasterKey.

Next, we can continue to calculate the read buffer IV (ivRx):

```
// Grab the SSRC from the RTP header and populate the ssrcBuffer
ssrcBuffer.position(4);
ssrcBuffer.putInt(ssrc);

// Set up and populate the iBuffer - this will currently
// make the roll over counter to be zero always
iBuffer.position(8);
iBuffer.putInt(readROC);
iBuffer.put(seqHex[2]);
iBuffer.put(seqHex[3]);

byte[] ssrcBytes = ssrcBuffer.array();
byte[] ksBytes = KsBuffer.array();
byte[] iBytes = iBuffer.array();
byte[] ivRx = new byte[16];

// XOR all 3 buffers
for (int ii = 0; ii<ivRx.length; ii++) {
    ivRx[ii] = (byte)(ssrcBytes[ii] ^ ksBytes[ii] ^ iBytes[ii]);
}

// Print
dump0x("ivRx", ivRx, 0,ivRx.length);
```

and similarly for calculating the write buffer IV (ivTx).

1.4 DECRYPTING THE MEDIA PAYLOAD

In the draft SRTP specification, the encryption algorithm is defined as AE Services in counter mode (CTR and NoPadding). The generated IVs, along with the KeRx or KeTx, can be used to secure the RTP stream during each transmit and receive operation.

First obtain the data from the RTP packet:

```
// get the RTP packet from the read socket
ByteBuffer dst = rtpPacket.getPacket();

// Point to payload start and obtain the encrypted payload
int hLength = 12; (RTP header size)
pLength = dst.limit() - hLength; (RTP payload size)

dst.position(hLength);
byte[] cipherData = new byte[pLength];
dst.get(cipherData, 0, pLength);
```

Note: cipherData will be used below by the cipher.

Then decrypt the data obtained from the RTP packet:

```
// Decrypt the data
try {
    String cipherSpec = algorithm + "/" + modeCTR + "/" + padding;
    cipher = Cipher.getInstance(cipherSpec);
} catch (Exception e) {
    e.printStackTrace();
}

byte[] plainTextResult = null;
SecretKey secKeRx = new SecretKeySpec(KeRx, algorithm);

try {
    IvParameterSpec ivSpec = new IvParameterSpec(ivRx);
    cipher.init(Cipher.DECRYPT_MODE, secKeRx, ivSpec);
    plainTextResult = cipher.doFinal(cipherData);
} catch (IllegalStateException e) {
    // Attempting to decrypt before the cipher has been initialized.
    // Probably a race condition resulting in the 1st packet not being
    // decrypted. Ignore for now.
} catch (Exception e) {
    e.printStackTrace();
}

dump0x("plainText", plainTextResult, 0, plainTextResult.length);
```

Finally below are the utility methods used in the code:

```
private static byte[] convert2Bytes(int num) {
    byte[] seq = new byte[4];
    seq[0] = (byte)((num >> 24) & 0xff);
    seq[1] = (byte)((num >> 16) & 0xff);
    seq[2] = (byte)((num >> 8) & 0xff);
    seq[3] = (byte)(num & 0xff);

    return seq;
}

public static void dump0x(String label, byte[] data, int start, int stop) {
    byte[] b = data;
    StringBuffer sb = new StringBuffer();
    int value;

    for(int j = start; j < stop; j++) {
        value = b[j] & 0xff;
        if (j % 6 == 0)
            sb.append("\n");
        sb.append((value < 16 ? " ", (byte)0x0 : " ", (byte)0x0)
            + Integer.toHexString(b[j] & 0xff));
    }
}
```

```

    }
    System.out.println(label+": "+sb.toString());
}

```

1.5 TEST DATA

In order to validate your code, we present here some test data against which you may run your decipher code. Following that is the expected output.

1.5.1 INPUT DATA

```

// From the MediaStart event, we get

byte[] readMasterKey =
{ (byte)0xaa, (byte)0xaa, (byte)0xaa, (byte)0xaa,
(byte)0xaa, (byte)0xaa, (byte)0xaa, (byte)0xaa,
(byte)0xaa, (byte)0xaa, (byte)0xaa, (byte)0xaa,
(byte)0xaa, (byte)0xaa, (byte)0xaa, (byte)0xaa
};

// From the RTP packet, we get
int ssrc = 987011809;
int seq = 3;
int roc = 0;

// And the data to decipher is:

cipherData = {(byte)0xbb, (byte)0xbb, (byte)0xbb, (byte)0xbb };

```

1.5.2 EXPECTED OUTPUT

Ke-Rx:

```

(byte)0xba, (byte)0xeb, (byte)0xc6, (byte)0x18, (byte)0xa5,
(byte)0x5c, (byte)0x35, (byte)0x1f, (byte)0x25, (byte)0xce,
(byte)0xdf, (byte)0x37, (byte)0xbf, (byte)0x70, (byte)0xf3,
(byte)0x90

```

Ks-Rx:

```

(byte)0x98, (byte)0x12, (byte)0xf4, (byte)0x3c, (byte)0x17,
(byte)0xc5, (byte)0xd4, (byte)0x0e, (byte)0xe3, (byte)0x8f,
(byte)0x09, (byte)0xe1, (byte)0x7f, (byte)0xa8, (byte)0xba,
(byte)0xb7

```

IV-Rx:

```

(byte)0x98, (byte)0x12, (byte)0xf4, (byte)0x3c, (byte)0x2d,
(byte)0x11, (byte)0x4e, (byte)0xef, (byte)0xe3, (byte)0x8f,

```

```
(byte)0x09, (byte)0xe1, (byte)0x7f, (byte)0xab, (byte)0x00,
(byte)0x00

// And the deciphered data should be:

plainData = {(byte)0x05, (byte)0x43, (byte)0x2a, (byte)0x3d };
```

- **Avaya MultiVantage® Application Enablement Services Management Console, Operations, Administration and Maintenance (OAM) On-Line Help**

- **OAM "Server Certificates" Help Page:**

The Server Certificate page elements table currently reads for the Import and Delete elements:

Import	<p>Select Import when you want to import a server certificate (generated by the CA) to the AE Server certificate store.</p> <p>When you click Import, OAM displays the Server Certificate Import page.</p> <p>IMPORTANT: After you import a Server Certificate you must restart the Web Server. For more information, see Service Controller.</p>
Delete	<p>Select a certificate, and click Delete to delete a server certificate.</p> <p>IMPORTANT: After you delete a Server Certificate you must restart the Web Server. For more information, see Service Controller.</p>

The table should read:

Import	<p>Select Import when you want to import a server certificate (generated by the CA) to the AE Server certificate store.</p> <p>When you click Import, OAM displays the Server Certificate Import page.</p> <p>IMPORTANT: After you import a Server Certificate you must restart the Web Server and AE Server. For more information, see Service Controller.</p>
Delete	<p>Select a certificate, and click Delete to delete a server certificate.</p> <p>IMPORTANT: After you delete a Server Certificate you must restart the Web Server and AE Server. For more information, see Service Controller.</p>

- **OAM "Server Certificate Import" Help Page:**

Currently, the online help ends at step 6:

6. Select Maintenance > Service Controller, and from the Service Controller page, click **Restart Web Server**.

In addition to step 6, step 7 should be performed:

6. Select Maintenance > Service Controller, and from the Service Controller page, click **Restart Web Server**.

7. Select Maintenance > Service Controller, and from the Service Controller page, click **Restart AE Server**.

DOCUMENTATION UPDATES FROM THE AE SERVICES

R4.2.1 RELEASE NOTES

Reprinted here for ease of use:

- **Avaya MultiVantage® Application Enablement Services
Installation and Upgrade Guide for a Bundled Server Release
4.2**

Section - Setting up the AE Server for remote access

On page 35 of 100 states:

"2. Make a note of this IP address. You will need to use it when you edit the /etc/hosts file and the /etc/ppp/options.ttyUSB0 file on the AE Server."

Editing the /etc/ppp/options.ttyUSB0 or /etc/ppp/options.ttyACM0 file, depending on the type of modem:

The section should read:

"2. Make a note of this IP address. You will need to use it when you edit the appropriate files on the AE Server for the specific modem used."

- **Avaya MultiVantage® Application Enablement Services
Administration and Maintenance Guide**
 - **Chapter 1: Administering Communication Manager for AE Services**

- **Section - Enabling Processor Ethernet Support**

On the S8300 and S8400 media servers, Processor Ethernet support is enabled by default.

On the S85xx media server, Processor Ethernet support is not enabled by default.

To enable AE Services Processor Ethernet support on the S85xx, first enable the Processor Ethernet feature on the System Parameters Customer Options form, then administer the processor interface with the add ip-interface procr command.

Follow this procedure:

1. Type **display system-parameters customer-options**.
 - Verify Processor Ethernet is enabled.
 - Processor Ethernet must be enabled before proceeding to the next task.

Note: Processor Ethernet support on the S85xx requires Communication Manager 3.1 or later.

2. Type **add ip-interface procr**

Important: Processor Ethernet support on the S85xx limits the CTI Message rate to 240 messages per second duplex.

- **Section - Enabling AE Services**

Enabling AE Services refers to administering the transport link between Communication Manager and AE Services.

<4.2.1 Release Notes> Step 2 in the current documentation needs to be changed to the following:

2. Complete Page 1 of the IP SERVICES form, as follows:

- In the Service Type field, type **AESVCS**.
- In the Local Node field type the appropriate entry, as follows:
 - On an S8300, S8400, S85xx*; type **procr**.

Note: *Make sure Processor Ethernet support has been enabled on the S85xx.

- On all other systems, type **<nodename>** (where **<nodename>** is the name of the CLAN board you will use on an S8400, S85xx, S87xx, or DEFINITY Server Csi.

You can locate node names by typing **display node-names ip** and checking the Local Node field on the IP NODE NAMES form.

- In the Local Port field, accept the default (**8765**).

Note: If you are adding more than one CLAN for AE Services, repeat Step 2 for each CLAN you add.

New Configuration Changes for Bridged Appearance Alert for Microsoft Office Live Communications (introduced in R4.2.1)

When administering AE Services for usage with Microsoft Office Live Communications Server a Bridged Alert Configuration (Bridged Alert Config) feature in AE Services OAM has been added to control the number of bridged appearances that appear on an assistant's desktop -- Microsoft Office Communicator will display a pop-up conversation window whenever a call appears for a bridged appearance. See section "*New Configuration Changes for Bridge Appearance Alert for Microsoft Office Live Communications*" in the release notes for details on how to administer this feature.

OAM Admin Guide for Bridged Appearance Alert Configuration

When you are administering AE Services for usage with Microsoft Office Live Communications Server you can use the Bridged Alert Configuration (Bridged Alert Config) feature in AE Services OAM to control the number of bridged appearances that appear on an assistant's desktop -- Microsoft Office Communicator will display a pop-up conversation window whenever a call appears for a bridged appearance.

By default, bridged call appearances generate an alert. This means that an assistant whose phone is administered with a bridged call appearance will receive an alert pop-up whenever an executive receives a call. The idea of managing or restricting bridged alerts comes into play when an assistant is administered with multiple bridged call appearances. In effect you are setting up a filter, based on the mapping of an assistant to an executive, or group of executives using the AE Services OAM web interface.

Access the Bridged Appearance Alert Configuration link

1. Login to OAM as a user with System Administrator privileges.
2. Select CTI OAM Administration→Administration→Bridged Alert Config
3. OAM shall display "Bridged Appearance Alert Configuration" page.

Alerts for Bridged Appearances Not Blocked By Default

The checkbox labeled "Block Bridged Appearance Alerts" is unchecked (default setting). Calls to bridged appearances on any assistant station WILL cause an Office Communicator incoming call alert window for the bridged extension to be displayed. If desired, exceptions may be made such that specific assistants will NOT receive alerts for specific bridged lines. The assistants in the "Rule Exceptions" list will not receive a call alert window for the specified executives' lines.

1. To add an assistant to the "Rule Exceptions" list, see the section **Add Bridged Appearance Rule Exception**.
2. To edit an assistant in the "Rule Exceptions" list, see the section **Edit a Bridged Appearance Rule Exception**.

Alerts for Bridged Appearances Blocked By Default

The checkbox labeled "Block Bridged Appearance Alerts" is checked. Calls to bridged appearances on any assistant station will NOT cause an Office Communicator incoming call alert window for the bridged extension to be displayed. If desired, exceptions may be made such that specific assistants WILL receive alerts for specified bridged lines. The assistants in the "Rule Exceptions" list will receive a call alert window for the specified executives' lines.

1. To add an assistant to the "Rule Exceptions" list, see the section **Add Bridged Appearance Rule Exception**.

2. To edit an assistant in the “Rule Exceptions” list, see the section **Edit a Bridged Appearance Rule Exception**.

Add Bridged Appearance Rule Exception

1. Click the Add button
2. Add Bridged Appearance Rule Exceptions page will be loaded.
3. In the “Assistant Login ID” field, enter the login id of the assistant.
4. In the “Executive Login ID” field, enter the login id of the executive.
5. If the assistant is associated to multiple executives, click the “Add Executive” button, another text box will be displayed for entering the other executive login IDs.
6. Click the “Apply Change” button
7. On the confirmation page verify each TelUri and User Name is correct.
8. If not correct, click the “Cancel” button to edit the entered data, otherwise click “Apply”
9. The Bridged Alert Configuration page will be reloaded with the new data.

Edit a Bridged Appearance Rule Exception

1. Select the rule exception that you want to modify
2. Click the Edit button
3. Edit Bridged Appearance Rule Exceptions page will be loaded
4. Modify the Executive Login ID fields, or click the “Add Executive” button to add additional executives.
5. Click the “Apply Change” button
6. On the confirmation page verify each TelUri and User Name is correct.
7. If not correct, click the “Cancel” button to edit the entered data, otherwise click “Apply”
8. The Bridged Alert Configuration page will be reloaded with the new data.

Delete a Bridged Appearance Rule Exception

1. Select the rule exception that you want to delete
2. Click the Delete button
3. The Confirmation page is loaded
4. If you really want to delete this rule, click the “Apply” button, otherwise click the “Cancel” button

View the details of a Bridged Appearance Rule Exception

1. Select the rule that you want to view
2. The corresponding detail page will be displayed, which includes the Login ID, User Name and TelUri of an Assistant and the associated Executives.

Synchronize the data stored by AE Services with LDAP

1. Click the “Synchronize” button.
2. The Confirmation page will be loaded.
3. If you want to continue the synchronization, click the “Apply” button. This function will synchronize the rule exception data in the AE Services OAM database with the customers LDAP data. Once the process completes, if an assistant does not exist in LDAP anymore, the rule exception for the assistant will be removed; In addition any associated executives will be removed. If an executive does not exist anymore, the executive will be removed from the corresponding rule exception list.

TEG (Terminating Extension Group)

Currently, TEG is treated the same as a bridged appearance call.

1. Assign the group number to a MOC user.
 - Group number is the extension of the executive.
 - Member number is the extension of the assistant.
2. Repeat the steps in the pervious sections:
 - **Add Bridged Appearance Rule Exception**
 - **Edit a Bridged Appearance Rule Exception**