

# Adobe® Media Server 5

## Release Notes

Welcome to the Adobe® Media Server 5 release! This document contains late-breaking product information and updates concerning this release.

[Minimum system requirements](#)

[What's new in this release](#)

[Important Changes in this release](#)

[Issues fixed in this release](#)

[Known Issues](#)

[Install your software](#)

[Uninstall your software](#)

[Other resources](#)

### Minimum system requirements

The Adobe Flash Media Server 5 system requirements are:

#### Supported operating systems

- Linux CentOS 5.8 64 bit

#### Hardware requirements

- 3.2GHz Intel® Pentium® 4 processor (dual Intel Xeon® or faster recommended)
- 64-bit operating systems: 4GB of RAM (8GB recommended)
- 1Gb Ethernet card recommended (multiple network cards and 10Gb also supported)

### What's new in this release

Adobe Media Server is available in following configurations

- Adobe Media Server **Starter** (formerly Adobe Flash Media Development Server)
- Adobe Media Server **Standard** (formerly Adobe Flash Media Streaming Server)
- Adobe Media Server **Professional** (formerly Adobe Flash Media Interactive Server)
- Adobe Media Server **Extended** (formerly Adobe Flash Media Enterprise Server)
- Adobe Media Gateway is a bundled software available with your license

## **New Content Protection options for HLS streaming to native Apple iOS applications**

Content protection and Digital Rights Management (DRM) workflows for streaming to both Flash using HTTP Dynamic Streaming (HDS) and native Apple iOS applications using HTTP Live Streaming (HLS). With this feature Adobe Media Server can dynamically segment, encrypt and deliver standard MP4 assets using the HLS format with Adobe Access DRM policies on native Apple iOS applications (using the free Adobe Access Objective C library for Apple iOS). For more details on this feature, please refer to online documentation.

Adobe Media Server 5 can also support content protection of HLS video streams using Adobe Access protection without requiring a separate license server. This unique functionality already available for HDS format enables ultimate scalability and cost management. Details on PHLS and PHDS features are available as part of Adobe Media Server online documentation.

## **Just in Time content protection enhancements**

Adobe Media Server 5 enables real time packaging and encryption of your media for HTTP delivery to both Adobe Flash and Apple HLS file formats. This means that your single live source publishing need not be duplicated to serve Adobe Flash & Apple HLS formats, which significantly reduces your storage and stream management requirements. Using dynamic and real time encryption, Adobe Media Server 5 can prepare and encrypt a single live source of F4F recording and deliver the encrypted segments over HTTP to both HDS (Flash) and HLS (Apple iOS).

## **HLS Offline Segmenter**

A new HLS offline segmenter utility converts MP4 video files into the HLS file format and M3u8 manifest for http streaming to devices that support HLS such as the Apple iPad. When used with Adobe Access, the HLS packager can encrypt the HLS file for playback in native IOS applications using the Adobe Access iOS client SDK.

## **Important Changes in this release**

Adobe Media Server 5 has introduced a few important branding and configuration-level changes. We recommend all users to go through below list of changes and prepare their environment accordingly.

- **Executable name changes**

Below is list of executable which has undergone name changes

- fmsmaster is renamed to amsmaster
- fmsadmin is renamed to amsadmin
- fmscore is renamed to amscore
- fmsedge is renamed to amsedge

- **Installer Scripts name changes**

Below is list of Installer Scripts which has undergone name changes

- fmsmgr is renamed to amsmgr
- installFMS is renamed to installAMS
- uninstallFMS is renamed to uninstallAMS

- **Tools name changes**

Below is list of Tools which has undergone name changes

- fmscheck is renamed to amscheck
- fms\_adminConsole.swf is renamed to ams\_adminConsole.swf

- **Service name changes**

Below is list of services which has undergone name changes

- fmsadmin is renamed to amsadmin
- fms is renamed to ams

- **Configuration files name changes**

- fms.ini is now renamed to ams.ini

- **Other important changes**

- Default installed directory on Linux is now /opt/adobe/ams
- Default User name is changed from “fms” to “ams”
- Registered Services reference location on Linux is changed from “/etc/adobe/fms/services” to “/etc/adobe/ams/services”

## Issues fixed in this release

3165076	Some valid license keys were not being recognized with Windows version of AMS.
3157234	In the livepkgr application, republishing of a stream fails if the stream starts with 0 timestamp. We have introduced new configuration named AdjustForRecordingRollover to take care of such scenario. When enabled along with AssumeAbsoluteTime, the server will assume that incoming live streams will always maintain the timestamps based on an absolute clock across server and stream restarts. By default AdjustForRecordingRollover is true but set to false in livepkgr application. As consequence of this setting, the default livepkgr application will append to an existing stream on republish.
3146826	Recording from Flash Player from MAC machine results in corrupted FLV.
3126588	iOS 5.0: Sample page flickers and disappears leaving just "Get Adobe Flash Player" button.

3118394	Incorrect duration is written for a partial TS in the m3u8, formed at the start of the recording.
3113419	Currently when you present a URL in OSMF as X and server send a redirect as Y in response. OSMF player constructs the bootstrap/drm/fragments uri by resolving the relative base url with X and not Y. This creates playback issue when user is using FMS 4.5 redirect functionality for redirecting older style HDS Playback Request URI to newer style. Though this is a player issue in 5.0 we are providing workaround so that user is not affected by this OSMF bug( <a href="http://bugs.adobe.com/jira/browse/FM-1558">http://bugs.adobe.com/jira/browse/FM-1558</a> )
3109854	When incoming key frame interval matches the disk cleanup interval exactly, disk management may fail to cleanup.
3097945	FMS HDS recordings would fail if repeat/backward timestamps were encountered in an RTMP publish. Now they are disregarded with a warning.
3089129	application.clearStreams() does not clear f4f streams.
3074802	[Windows only] When one enables PRTMP with SWFv via SSAS, the Whitelist folder is wrongly mapped to the default folder irrespective of the folder mentioned in the SSAS
3064933	Extra audio sample descriptors led to small audio pops on encoded HDS content.
3056883	When a user attempts to stream a pre-encrypted file through PRTMP, it resulted in assert failures. Now it will log a debug message.
3056110	HLS streaming module dropping segments in case of compacted bootstrap table.
3048722	Errors in core log while republishing into livepkgr from FP publisher with SS & NellMoser/Speex codecs.
3019603	NetStream object cannot be used to play a different stream when PRTMP is enabled. If a NetStream is used to play a single stream, and is reused to play a different stream later, the second play will fail. To work around this problem, a new NetStream object can be created for each stream play instead of reusing the same NetStream for multiple streams. We have also confirmed that playing multiple streams on the same NetStream with client side playlist still works as expected.
2963728	AMS crashes when some incorrect certificates are given for encryption parameters
2947283	The releaseStream() function in livepkgr app is called without sanitizing the stream name argument.
2939847	Some MPEG4 files with wrong size for mdat box were not playing. Now this causes a warning but the file still plays.
2930714	There needs to be functionality to remove recorded content once application unloads if a user chooses to do so.
2908311	VOD sample validation constraints of url are modified so that they don't fail depended on query parameters.

## Known Issues

The following are known issues in this release:

3060434	Letting RTMPT traffic proxying through Apache reduces the quality of service for RTMPT connections.
---------	---

3052723	<p>As of 4.5.2 release, RTMPT can be tunneled through Apache listening on port 80. This is to allow both RTMPT as well as HDS to work over port 80. As a result, it means that RTMP[e] connections on port 80 would fail to connect. This may affect Edge/Origin environments where clients connect to the Edge server over RTMPT on port 80. The Edge server would try to connect to the Origin on the same port. However, the protocol between Edge and Origin is RTMP (or RTMPS), and hence would fail to connect.</p> <p>To avoid this, you could do one of the following:</p> <ol style="list-style-type: none"> <li>1. Configure the Origin to have FMS listen on 80; this means that you cannot run both RTMPT and HDS over port 80.</li> <li>2. Configure <code>&lt;RouteEntry&gt;*:80;*:1935&lt;/RouteEntry&gt;</code> in the Edge server's Vhost.xml to remap connections to 80, to 1935 instead.</li> <li>3. Rewrite the Edge client connection url to replace port 80 with 1935 so that the Edge will attempt to use 1935 instead of the port the client connected on.</li> </ol>
2944919	<p>When doing HDS and HLS live streaming, if audio or video is received after the first fragment has been created can lead to a "Track not found" error that results in a recording error. To ensure this does not happen, audio and video should start at the same time.</p>
2873133	<p>Calling <code>NetStream.send()</code> immediately after receiving <code>NetStream.Publish.Start</code> status message may intermittently fail due to a race condition when initializing the Stream object in server-side ActionScript on the receiving server.</p>
2674905	<p>In an n-tier deployment where multiple live streams with different bitrates are being multi-point published from the ingest node to a downstream node, (and synchronized keyframes as required for seamless switching when dynamic streaming), if the streams are being recorded at the downstream node with the intention of utilizing them as DVR streams, the value of "AssumeAbsoluteTime" configuration in application.xml should be set to "true" for the application at the downstream node where the recording is taking place.</p>
2501913	<p>Authorization and Access plugins may result in compilation warnings. These warnings are harmless. Removing the warnings now, would require an interface change, and will break plugins built using the current interface. To maintain compatibility we have decided not to fix these warnings at this time. This will allow users to use their existing plugins without having to rebuild and retest.</p>
2476494	<p>When enabling stream logging via the admin commands <code>logLiveStream</code> and <code>logNetStream</code>, the log directory is always created in the default log directory under the FMS installation directory, regardless of what configured in <code>fms.ini</code> and <code>Logger.xml</code></p>
2426933	<p>The File Plugin API <code>getAttributes()</code> may be called before the <code>E_FILENAME_TRANSFORM</code> event has been processed. As a result, the wrong file attributes may be returned.</p>
2275665	<p><code>sc-stream-bytes</code> measures the bytes per stream sent from server to client for a particular client. Note that when compared with <code>sc-bytes</code>, which measures all bytes sent from server to client for a particular client, users may notice a discrepancy. There are a number of situations that could trigger this discrepancy, but users are advised to take note of this if trying to use <code>sc-stream-bytes</code> for billing purposes. Billing should be done based on <code>sc-bytes</code> when possible.</p>
1630621	<p>Very low frame rate H.264 videos may take a long time to start if the buffer is set too</p>

	short.
1611644	The OnImageData event within H264 streams is not supported.
3146582	HLS Failover - In case there are fragment discontinuities (which may happen if key-frames are not found at fragment boundaries resulting into fragments bigger than one fragment duration) then the healthy packager may return a 503 after failover resulting in playback stall
3142085	HLS Failover - If recording is not yet started on one packager, requests for TS URLs directed to this packager won't failover (the packager would return a 400 instead of a 503)

## Install your software

### *To install Adobe Media Server on Linux:*

1. Log in as a root user (required to install Adobe Media Server).
2. Open a shell window and switch to the directory with the installation file, AdobeMediaServer\_5\_WWE.tar.gz
3. Unzip and untar the installation file. A directory with the installation program is created.
4. Switch to the directory that was just unzipped.
5. Start the installation program with the following command: ./installAMS
6. The installation program starts and displays a welcome message.
7. Press Enter to start the installation. Follow the installation instructions on your screen. During the process you will be asked to
8. Enter a serial number. If you don't enter a serial number, Adobe Media Server Starter installs.
9. Enter the installation directory and ports which the server will use
10. Enter an administrative user name and password for the first valid server administrator. These values are written to the ams.ini file which is located inside of the conf folder after the installation.
11. Enter a user for Adobe Media Server processes to run as. The default is the "ams" user. (The user you select is also the owner of the Adobe Media Server files.) Your choices are written to the ams.ini file. You can edit the ams.ini file to modify this and other security properties later, if needed.

12. Review the summary of the installation options you have chosen, which are displayed in the installer.
13. To start the server manually, go to the installation directory (default is /opt/adobe/ams/). Use the command ".amsmgr server ams start" to start Adobe Media Server and "./amsmgr adminserver start" to start Adobe Media Administration Server.
14. The installation is complete. If you configured it to start automatically, the Adobe Media Server service starts.

## **Uninstall your software**

### ***To uninstall Adobe Media Server from a Linux computer:***

1. Log on to the server where Adobe Media Server was installed.
2. Switch to the root user, or a user with root permissions. Normally you would use su - root to switch to the root user.
3. At the UNIX shell prompt, enter cd /opt/adobe/ams.
4. By default, /opt/adobe/ams is the directory where Adobe Media Server is installed. If you installed the server in a different directory, replace /opt/adobe/ams with the actual installation location.
5. Execute the uninstall script ./uninstallAMS.
6. Follow the displayed uninstall instructions.

### **Other resources**

[Adobe Media Server Documentation Center](#)  
[Adobe Media Server Product Home](#)  
[Adobe Media Server Developer Center](#)  
[Adobe Media Server End User License Agreement](#)  
[Adobe Media Server Online Forums](#)  
[Adobe Media Server Hands-on Training](#)  
[Adobe Media Server User Group](#)  
[Flash Video Streaming Service](#)  
[Flash Media Solutions Providers](#)  
[Flash Player Product Home](#)

Copyright © 2012 Adobe Systems Incorporated.