AMS Confidentiality Performance

Objective

Examine the performance of the new AMS Confidentiality mode.

Background

MQ customers have often requested message payloads to be encrypted at rest to comply with various security mandates.

Initial AMS performance testing often leaves users surprised when the performance is compared with non-AMS messaging because of the overhead of frequent asymmetric key operations. For many users the repeated asymmetric key operations involved in signing each message and encrypting a symmetric key for each intended recipient are more than what they require; they often want the payload encrypted, but don't necessarily need each message to be individually signed nor symmetric keys created for each message.

MQ V9 delivered a new AMS Quality of Protection called 'Confidentiality' which utilizes symmetric keys without message signing to provide a faster way to allow messages to be transferred securely and protect their payload data at rest.

Scenario

A couple of different scenarios will be used to analyze the AMS Performance:

- Single queue – All clients send and receive from a single queue
- Multiple queue – All clients send and receive from one of 21 queues

For this investigation, a 2KB persistent message is used in all tests and each message is encrypted for a single recipient. The messaging scenario is a request responder scenario as featured in the current distributed and appliance performance reports.

A comparison will be made between all of the AMS Quality of Protection settings:

- None – No signing or encryption applied
- Integrity – Message is signed
- Privacy – Message is signed and encrypted
- Confidentiality – Message is encrypted and key may be reused
    - Key reuse setting of 32 will be used

The key reuse setting controls how often the symmetric key that is regenerated. For more information on this and additional detail on the Qualities of protection, please see the IBM Knowledge Center:

http://www.ibm.com/support/knowledgecenter/SSFKSJ_9.0.0/com.ibm.mq.sec.doc/q127085_.htm

The signing algorithm used in the Integrity and Privacy modes is SHA512. The encryption algorithm used for the Privacy and Confidential modes is AES256.

Environment

These tests use 3 x86_64 Linux servers (see Appendix A for their specification); Server 1 hosts the requester clients, Server 2 hosts the QM and Server 3 hosts the responder applications.

When providing a message buffer to the MQGET API to receive your AMS message, ensure the buffer is larger than the expected message size, as the encrypted payload size is much larger than the original message length. For the tests featured in this report a 20KB buffer was supplied.

The version of MQ used in these tests is MQ V9.0.

Results

Single queue

The graph below shows the results from the single queue test:

Note that the CPU represented on the charts in this whitepaper is an average of the two client machines featured in the test; this is of more interest than the server CPU (which is traditionally featured on such graphs) as the costs of encryption and decryption are handled by the client machines.

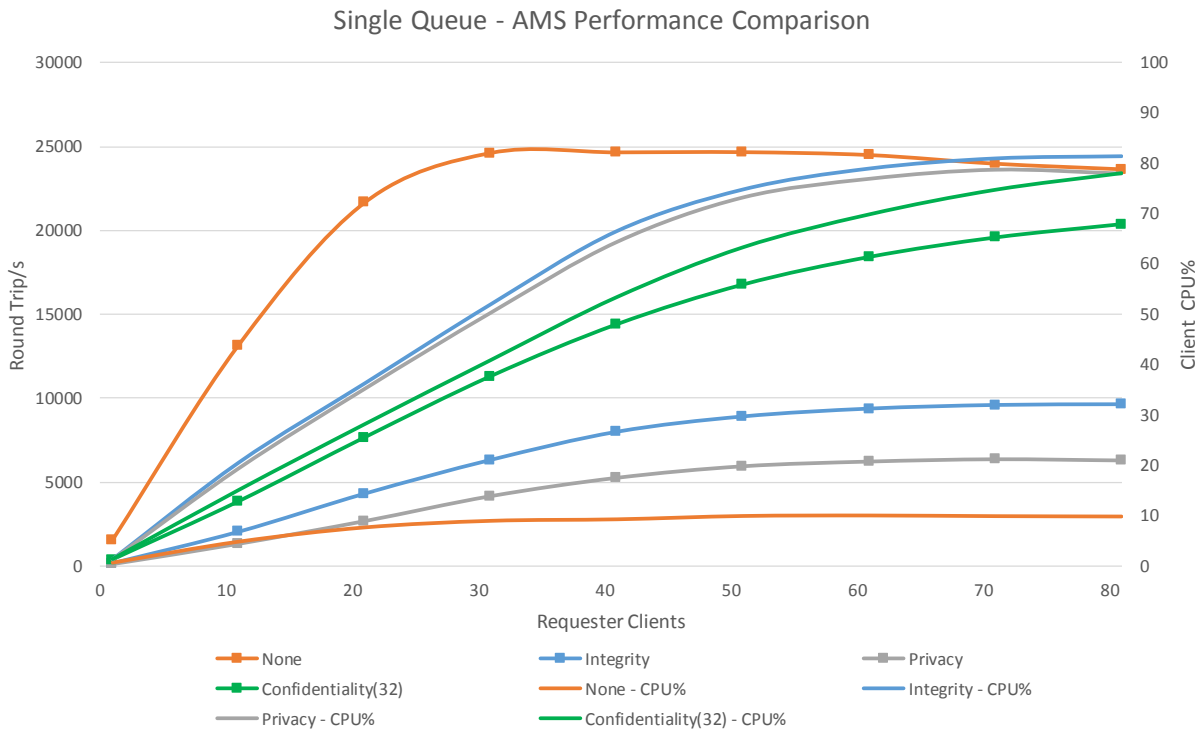

Figure 1 - Single queue AMS Comparison

The performance of AMS Confidentiality is more than 3x faster than the AMS Privacy mode. At 81 clients, the performance of AMS Confidentiality was only 14% slower than when not using AMS (although more CPU was being expended by the clients).

Multiple queue

The graph below shows the results from the multiple queue test:

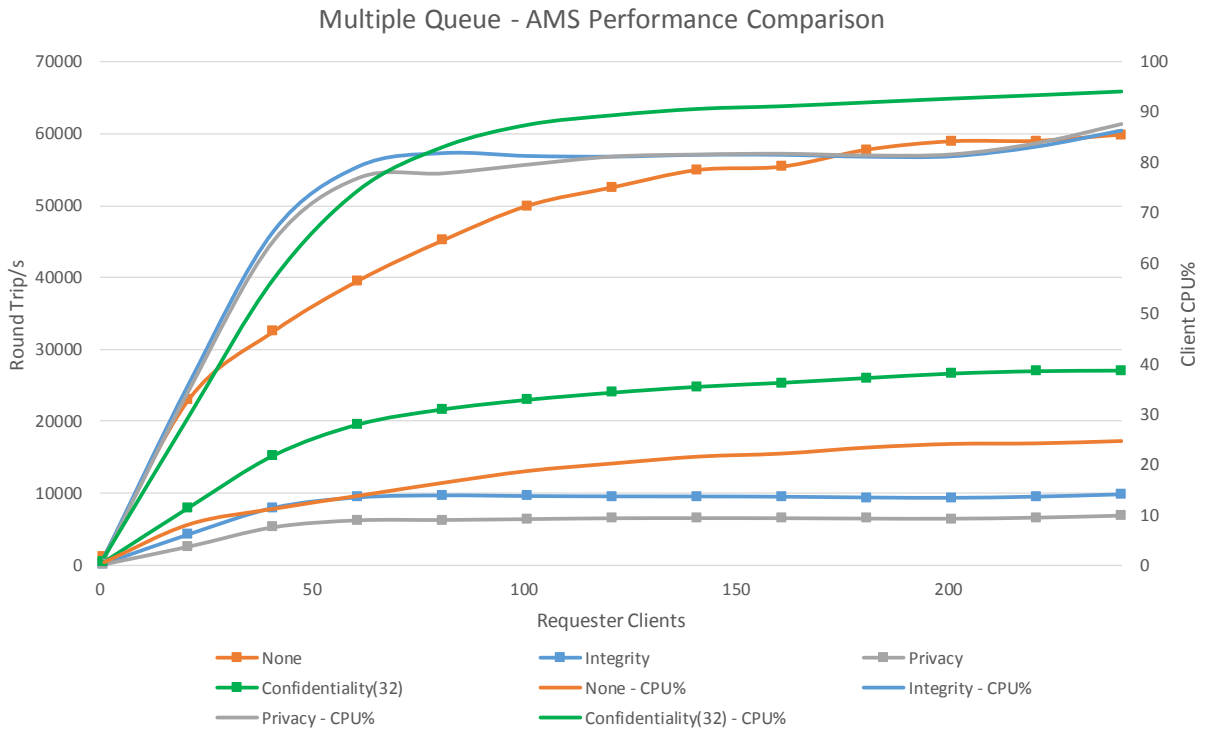

Figure 2 - Multiple Queue AMS Comparison

Adding multiple queues into the scenario has increased the performance of AMS Confidentiality(32) to nearly 4x faster than the AMS Privacy mode. It has also increased the performance of the non AMS scenario (by relieving queue lock) and at 241 clients, the performance of AMS Confidentiality(32) was just under half of the performance when compared with not using AMS.

The peak throughput achieved for the AMS Confidentiality(32) measurement at 241 clients was just over 27,000 round trip/s. The request/responder scenario utilizes a request and a reply queue, so for each round trip, 2 message puts and 2 message gets occurs. For a single put/single get scenario, the peak performance that you might obtain in the same environment is over 54,000 msg/sec.

AMS policy comparison

For the comparisons in this whitepaper, a key reuse value of 32 was chosen; this is considered a reasonable compromise in the tradeoff between the costs of key renegotiation (i.e. scenario performance) and security. A set of data was collected that demonstrates how the performance linearly varies with the key reuse configuration. The chart below uses a scenario whereby each requester and responder thread send and receive from a separate queue; performance results were recorded when using up to 21 queues with 21 requester clients and 21 responder threads:

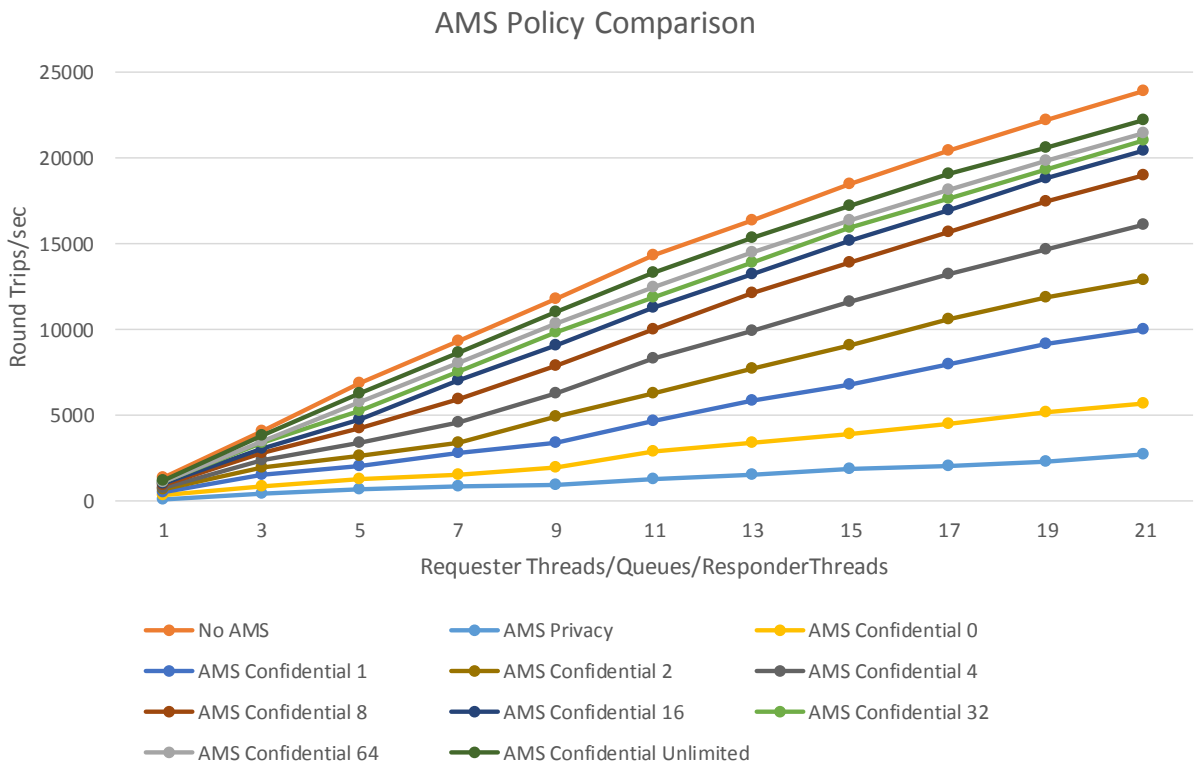

Figure 3 - AMS Policy Comparison

AMS Confidentiality(32) provides 94% of the performance of AMS Confidential(Unlimited), but will only reuse the same symmetric key for 32 messages.

Effect of poorly sized receive buffer

If a buffer that is not adequately sized for performing the MQ Get operation, poor performance can result from the MQ client having to perform multiple message retrievals from the MQ QM. The graph below takes the AMS Confidentiality(32) result from the multiple queue scenario and compares it with the result if a message buffer of just 2K had been provided:

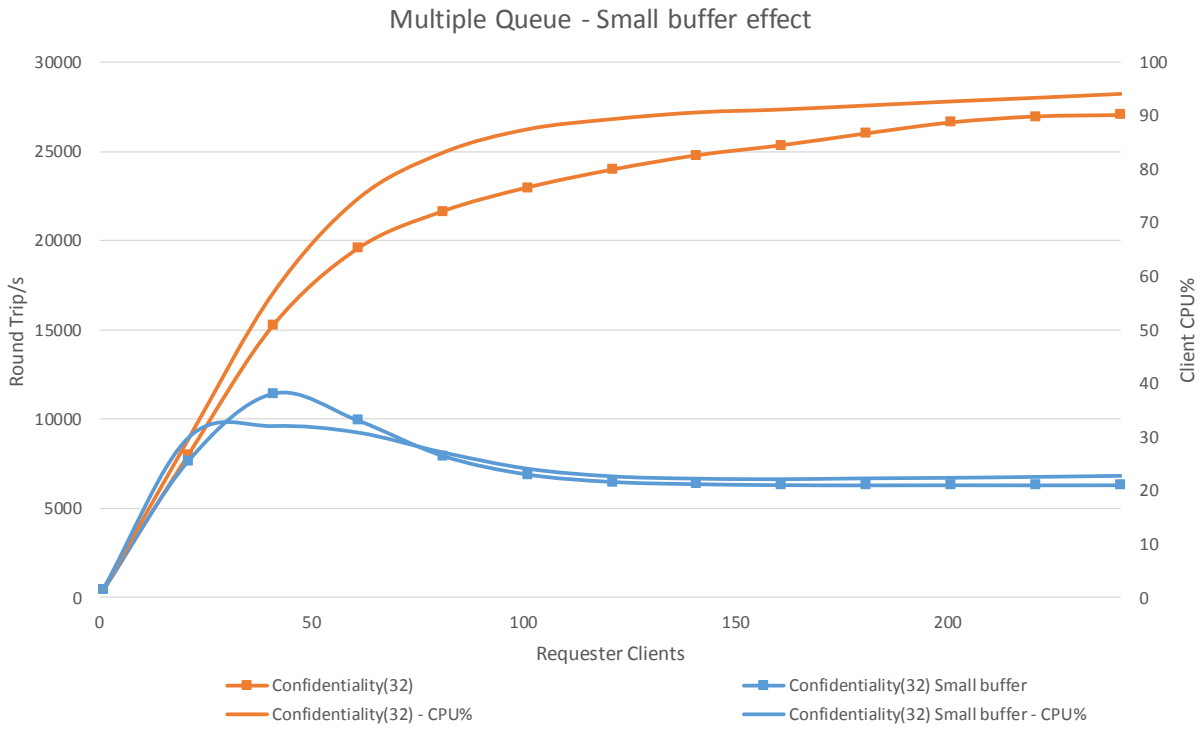

Figure 4 - Effect of small receive buffer

Conclusions

The new AMS Confidentiality mode provides end to end security for your message payload so that it is protected in transit and whilst at rest in the Queue Managers filesystem. It reduces asymmetric key encryption and thus can offer faster performance than previously supported AMS policies. It also allows the user to define how often to reuse the same symmetric key for payload encryption, thus providing flexibility in the choice between key regeneration frequency and performance.

Author

The author of this whitepaper is Sam Massey who works in the MQ Performance Team at the IBM UK Laboratory, Hursley. If you have any questions or comments on this paper, please contact him at smassey@uk.ibm.com

Appendix A

The three machines used for the performance tests in this report have the following specification:

Category	Value
Machine	x3550 M5
OS	Red Hat Enterprise Linux Server 7.2
CPU	2x12 (2.6Ghz)
RAM	128GB RAM
Network	10Gb/40Gb Ethernet
Disks	2x 480GB SSD
RAID	ServeRAID M5210 (4GB Flash RAID cache)