MassTransit Cipher Compatibility

Group Logic, Inc.
November 7, 2008

MassTransit 6.0 provides support for AES-128 and AES-256 encryption algorithms that are recognized as industry standard. These algorithms are natively supported by the version of OpenSSL library currently utilized in MassTransit (0.9.8g). However, there are certain issues related to using these ciphers that had to be resolved in order to preserve MassTransit 6.0 compatibility with older versions.

OpenSSL implements 3 separate secure protocols – SSLv2, SSLv3 and TLSv1. Each protocol has a set of encryption algorithms assigned to it in such a way that using a specific algorithm for a secure session requires using that algorithm's respective protocol. For instance, the RC4-40 encryption algorithm requires SSLv2 whereas the RC4-128 and 3DES encryption algorithms require TLSv1. This aspect has not been an issue in MassTransit so far as OpenSSL is able to automatically pick the appropriate protocol during a handshake based on the encryption algorithm specification selected for the session. That is not the case with the AES-128 and AES-256 encryption algorithms which require TLSv1 – those algorithms are not even recognized by OpenSSL unless TLSv1 is explicitly selected as the protocol for a secure session. This can cause handshake failures when encryption algorithms selected on the server and the client sides in MassTransit are not using the same protocol.

This issue is addressed to a certain extent in MassTransit 6.0 by providing legacy compatibility mode when setting up secure listens on the server side as well as providing comprehensive error messages when connection attempt fails due to incompatible encryption algorithms. When using MassTransit in mixed environments with both 6.0 and earlier versions present, however, reasons for secure handshake failures may not be apparent.

The following table provides information on encryption algorithm compatibility and should be used as a guideline when setting up secure communications in MassTransit (* indicates legacy compatibility mode enabled on the server side):

Client / Server	5.x RC4-40	5.x RC4-128	5.x 3DES	6.0 RC4-40	6.0 RC4-128	6.0 3DES	6.0 AES-128	6.0 AES-256
5.x RC4-40	Yes	Yes	Yes	Yes	Yes	Yes	No	No
5.x RC4-128	No	Yes	Yes	No	Yes	Yes	No	No
5.x 3DES	No	No	Yes	No	No	Yes	No	No
6.0 RC4-40*	Yes	Yes	Yes	Yes	Yes	Yes	No	No
6.0 RC4-128*	No	Yes	Yes	No	Yes	Yes	No	No
6.0 3DES*	No	No	Yes	No	No	Yes	No	No
6.0 RC4-128	No	No	No	No	Yes	Yes	Yes	Yes
6.0 3DES	No	No	No	No	No	Yes	Yes	Yes
6.0 AES-128	No	No	No	No	No	No	Yes	Yes
6.0 AES-256	No	No	No	No	No	No	No	Yes