Why Message Level Security?

Messaging that does not involve humans

Command & control scenarios

Large MQ networks: difficult to prove security of messages

Against message injection / message modification / message viewing

Data subject to standards compliance (PCI, HIPAA, etc)

Credit card data protected by PCI
Confidential government data
Message Level Protection

- Assurance that messages have not been altered in transit
  - When issuing payment information messages, ensure the payment amount does not change before reaching the receiver

- Assurance that messages originated from the expected source
  - When processing control messages, validate the sender

- Assurance that messages can only be viewed by intended recipient(s)
  - When sending confidential information
WebSphere MQ Advanced Message Security

Sending App

MQ Msg

&@Ja^!

Receiving App

MQ Msg

&@Ja^!
WebSphere MQ Advanced Message Security

- Provides additional security services over and above base MQ

- App → App data protection for **point to point** messaging
  - Asymmetric cryptography used to protect individual messages

- Administratively controlled policies applied to queues
  - Command line
  - MQ Explorer

- Non-invasive
  - No changes required to MQ applications
WMQ vs WMQ AMS

- **WebSphere MQ**
  - Authentication (local OS for server apps or peer authenticated SSL for client apps)
  - Authorisation (OAM on distributed, RACF on z/OS)
  - Integrity (SSL for channels)
  - Privacy (SSL for channels)

- **WebSphere MQ AMS**
  - Integrity (Digital signature of message content)
  - Privacy (Message content encryption)
WMQ + AMS v7.0.1 Architecture

MQ Svr App
MQ Java App
MQ Client App

API Intercept
JMS Intercept
Client Intercept

OK?

y/n

MCA

Queue Manager

Object Authority Manager
Administration: Distributed

- Command line tools
  - `setmqsp1` : Set message protection policy
    - `-m QMGR`
    - `-p Policy_Name`
    - `-s Signing_Algorithm`
    - `-a Authorised Signers`
    - `-e Encryption_Algorithm`
    - `-r Message_Recipients`
  
  - `dspmqsp1` : Display message protection policies
    - `-m QMGR`
    - `[-export]`
    - `[-p Policy_Name]`
Protection Policies

<table>
<thead>
<tr>
<th>Policy Name</th>
<th>Signing Algorithm</th>
<th>Encryption Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q1</td>
<td>SHA1</td>
<td>AES256</td>
</tr>
<tr>
<td>Q2</td>
<td>SHA1</td>
<td>AES256</td>
</tr>
<tr>
<td>MyQueue</td>
<td>SHA1</td>
<td>AES256</td>
</tr>
</tbody>
</table>

Double click policy to view properties

Delete... Properties... New...

Right click node to create a new policy

Pops up a dialogue asking the user to supply a DN.

Name of policy is not editable once created.

Toleration group is extended content and may not be present in the initial release.

Removes the selected DNs from the list below.

If selected, at least one DN is required. If deselected, a DN is not required.

Note – the Policy Name is the same as the queue name to which it applies.

Toleration
- Strictly apply this policy to all messages
- Apply policy to protected messages, but tolerate unprotected messages

Encryption algorithm:
- NONE

Applications put directly to the queue protected by this policy (uncheck this option if the queue is only accessed by another queue manager)

List the distinguished names (DNs) that are permitted message originators. Only messages with signatures containing one of these DNs will be accepted.

Add Signature DN... Remove

Valid signature DN
- cn=Robert Smith,ou=IBM Software Group,o=IBM,c=UK
- cn=Lisa Jones,ou=IBM Software Group,o=IBM,c=UK

List the permitted message recipients. Messages will only be readable if encrypted for one of these recipients.

Add Recipient DN... Remove

Permitted recipient
- cn=Robert Smith,ou=IBM Software Group,o=IBM,c=UK
- cn=Lisa Jones,ou=IBM Software Group,o=IBM,c=UK

Toleration
- Strictly apply this policy to all messages
- Apply policy to protected messages, but tolerate unprotected messages
WebSphere MQ AMS : Security Policy

Signature Algorithm
- MD5
- SHA1

Encryption Algorithm
- RC2
- DES
- 3DES
- AES128
- AES256

Acceptable signer(s)
Applicable when signing messages

Message recipient(s)
Applicable when encrypting and signing messages
Interceptors

Server
- API Exit
  - Application
  - MQ API (mqm lib)
  - API Exit
  - QMGR

Client
- Library Replacement
  - Application
  - Replacement mqic lib
  - Renamed MQIC
  - Channel Agent
  - QMGR

JMS
- JMQI Intercept
  - JMS Application
  - JMS
  - JMQI Intercept
  - JMQI
  - Channel Agent
  - QMGR
WebSphere MQ AMS

Alice

Sending App

AMS_QM

APP.Q

Bob

Receiving App
1. Install AMS Interceptor
1. Install AMS Interceptor
2. Create public / private key pairs
WebSphere MQ AMS

1. Install AMS Interceptor
2. Create public/private key pairs
3. Copy recipient's public key
WebSphere MQ AMS

1. Install AMS Interceptor
2. Create public/private key pairs
3. Copy recipient's public key
4. Define protection policy for the queue
WebSphere MQ AMS: Integrity Message Format

Original MQ Message

<table>
<thead>
<tr>
<th>Message Properties</th>
</tr>
</thead>
<tbody>
<tr>
<td>Message Data</td>
</tr>
</tbody>
</table>

AMS Signed Message

<table>
<thead>
<tr>
<th>Message Properties</th>
</tr>
</thead>
<tbody>
<tr>
<td>PDMQ Header</td>
</tr>
<tr>
<td>PKCS #7 Envelope</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Message Data</th>
</tr>
</thead>
</table>

| Signature    |
WebSphere MQ AMS: Privacy Message Format

**Original MQ Message**
- Message Properties
- Message Data

**AMS Encrypted Message**
- Message Properties
- PDMQ Header
- PKCS #7 Envelope
  - Key encrypted with certificate
  - Data encrypted with key
  - Message Data
  - Signature
Key Points

- **MQ AMS dates**:
  - Released: 8\textsuperscript{th} Oct 2010
  - 7.0.1.1 Released: 14\textsuperscript{th} April 2011
    - Added support for cryptographic hardware acceleration
  - 90 day Trial version available to download

- **Platform support**
  - Same as MQ 7.0.1 (except IBM i)
  - Works with MQ 6 & MQ 7 queue managers (JMS interceptor requires v7 jars)
Thank You

Questions ?