Global Information Assurance Certification Paper

Jose Alberto Yong

July 1, 2004

IMPLEMENTING SECURITY ON A WEBSPHERE MQ DISTRIBUTED MESSAGING ENVIRONMENT

IMPLEMENTING SECURITY ON A WEBSPHERE MQ DISTRIBUTED MESSAGING ENVIRONMENT

ABSTRACT

The business world is continually adopting new technology to stay competitive and to cater the ever increasingly technical markets and customers. Today, businesses are involved in data processing, e-commerce, and direct online communication with customers, and transferring information over distributed networks, such as the Internet.

As technology evolved, businesses in order to stay competitive and provide their customers with state-of-the-art online services opted for implementing the very best applications with the best platform. However, the challenge was always filling the gap between the best-of-breed applications with the best platform and operating system.

Middleware distributed messaging systems such as WebSphere MQ provide the solution in filling up this technology gap. The beauty of a middleware distributed messaging solution is the ease of integration and support among the best-of-breed applications with the best platforms and operating systems.
This paper will analyze the main security threats to the middleware-messaging environment such as WebSphere MQ and present the industry's best practices in implementing a secure WebSphere MQ distributed messaging environment.

WEBSPHERE MQ OVERVIEW

Websphere MQ was developed to fill the gap that was becoming increasingly evident and costly in the business world – the gap between applications and platforms. It makes good financial and technologically strategic sense to choose applications that best serve your company's needs, both for performance and budget. However, the best application for one area of your business may not be on the same platform or easily integrated with the best application for another area of the company or with a third party business partner.

Instead of having to compromise or pay for custom programming, which thereafter needs to be maintained, WebSphere MQ enables you to choose the very best application for your various business needs with the assurance that they can communicate seamlessly and effectively through middleware. WebSphere MQ is the *middleware* that builds a bridge between the islands that are your applications, remote offices, and databases. As IBM has developed WebSphere MQ, it gradually incorporated and now supports about 35 operating systems. This compatibility and support for so many platforms and the ease of integration with the various applications in the market has earned WebSphere MQ as the leading supplier of messaging middleware.

The trend toward a global economy, more distributed networks, business over the Internet and remote offices make the need for middleware more that crucial. Enterprises need to integrate applications and data from more diverse/complex sources and platforms than ever. Thus, the reliance on middleware has become more critical than ever. WebSphere MQ enables businesses to grow, to integrate data and applications and has become part of the backbone of an enterprise. But every solution needs managing and implementing a secure environment and WebSphere MQ is not an exception.

THE RISK OF OPEN ACCESS

Nowadays, businesses are embracing technology and e-commerce to stay competitive, increase profits and provide better services and products to their customers. In the beginning of the Information Age and the coming of the Information Super Highway, industry experts discussed "access" and how to bring information to the masses. After a decade of impressive technological growth and expansion, we are starting to realize that the technology including WebSphere MQ that is helping us conduct business faster and farther is also putting us at risk. The corporate most valuable asset – ***Information*** – can be

compromised, stolen, misrouted, exposed, or altered.[1] Because of these factors, corporate liability can be enormous if security barriers (Defense in Depth) to potential attacks are not properly implemented and defined.

HOW BIG OF AN ISSUE IS "SECURITY"?

How big of an issue is "security"? Consider the fact that a security breach will cost… time, financial losses, data, your customers, your reputation, your trust, your business.

According to a study by the Federal Bureau of Investigation (FBI) and the Computer Security Institute (CSI), computer security breaches have increased steadily and recovering from attacks is expensive. Last year, the value of stolen proprietary information averaged $2.7 million per occurrence. The year before that, one company reported a $50 million compromise!! [2]Security breaches are split almost 50-50 between external and internal attacks.

One of the difficult aspects of quantifying computer crime is that companies that have been compromised go to great lengths to hide or deny the breach, since the negative publicity can do more damage that the initial break in. Although new legislations such as the California Identity theft law, SB 1386, HIPAA, etc. are making companies more accountable to such kind of computer information breaches. In addition, many times companies are unaware that they are being attacked. In a well publicized case, when the FBI apprehended Carlos Salgado with 80,000 stolen credit card numbers, two of the compromised companies were not even aware of a breach until the FBI contacted them for cooperation in the investigation.[3]

While the Salgado case revealed how lucrative and appealing hacking into corporate systems can be, it is not the only motivator for cyber crime. There are large, organized group of hackers who specialize in exposing security holes and weaknesses, some are looking for financial gain, some are looking for fame or political power, some feel compelled to push the industry to keep improving security techniques and policies.

While the above cases are only a small fraction of the overall security incidences, there is no doubt that "security" is one of the outmost important considerations when implementing a computing system.

---

[1] Candle, Corp. "How to Conquer Security Challenges in Distributed Messaging Environments",March, 2003 http://www.capitalware.biz/dl/docs/mq5guide.zip

[2] 8th Annual CSI/FBI Computer Crime and Security Survey http://www.yle.fi/mot/kj040524/fbiraportti.pdf

[3] "Special Report: Salgado case reveals dark side of electronic commerce" Richard Power, CSI Monthly Newsletter, September 1997.

WHAT YOU NEED TO KNOW ABOUT SECURITY

You need to know the risk and liabilities of you business. The threats to your company's information are real and can range from a reasonable innocent user errors and glitches to malicious hacking. Just as you purchase virus-protection software to prevent and solve problems before there is damage to your data and system, you need to proactively invest in ensuring the security of your data.

A good rule of thumb in securing data is: "*The cost of a security system must be less than the value of the data that is protecting*". This principle dictates that it would not be logical to implement security for subsystems whose data have little or no value.[4]

In a WebSphere MQ environment, several different platforms communicate application commands and your company's critical information through message queue managers. Since messaging is the transmission of technology (data, e-mail, strings, objects) from one system to another across machines or platforms, the "message" becomes the object that must be secured.

RISKS/THREATS AGAINST WEBSPHERE MQ

WebSphere MQ is a messaging middleware transport. It sits above low-level protocols such as SNA and TCP/IP (which can themselves be secured) and acts as the "glue" in a heterogeneous computing environment. The following represents security threats against a WebSphere MQ infrastructure:

Sniffing: Computers with access to a network can record traffic flowing through it. If data or commands are sent unencrypted as WebSphere MQ messages, it is easy for unauthorized people to passively eavesdrop. The keyword here is *passive.* Sniffing is an activity that leaves no trace in a network log.

Sniffing is a threat to confidentiality[5]. Hackers with malicious or fraudulent intent can gain financial or business advantage by being able to sniff message content, particularly if the context of credit card or fund transfers messages

The risk/threat of sniffing becomes even greater if user IDs and passwords are captured. This is because an attacker could then impersonate a legitimate user. It may be possible for an attacker to also use sniffed packets off the network and re-apply them. For example, if a MQ message is an instruction to make a

[4] Cesare San Martino. "Securing an MQ Infrastructure, MQSoftware Resource Center White Paper", July 2002 http://www.mqsoftware.com/products/docs/Securing_an_MQ_infrastructure.pdf

[5] Saida Davies, Hazel Fix, Peter Rhys-Jenkis, Mayumi Kawashima, John Scanlan, Steven Lane. "WebSphere MQ Security in an Enterprise Environment" IBM Redbook Abstract, May 7, 2003 (p. 17) http://www.redbooks.ibm.com/redbooks/pdfs/sg246814.pdf

payment, then its is possible to replay this message multiple times.

Impersonation: The hacker tricks a security system, passing as an authorized user. There are three levels of impersonations:

1. The hacker may be able to impersonate the WebSphere MQ Administrator.
2. The hacker may be able to impersonate a valid queue manager and receive messages
3. The hacker may be able to impersonate a sending queue manager and send messages.

For example, the hacker could impersonate a valid queue manager and pass messages. This attack is the most common and easy to perform since by default queue managers do not authenticate and most queue managers in a distributed environment have the administration ID set to ***mqm***.

The hacker may sniff network traffic and steal a valid WebSphere MQ Administrator user ID and password. If a WebSphere MQ message is not digitally signed or encoded with a weak CipherSpec, a hacker can change the message content with malicious or fraudulent intent and/or enter completely new data or commands.

Impersonation can be a threat to all three goals of computer security[6]: **CIA**, **C**onfidentiality, **I**ntegrity and **A**vailability.

Decryption: If WebSphere MQ messages are sent over a public network, hackers may be able to easily capture the encrypted messages[7]. Strong encryption methods are a must in today's world, especially if the messages contain confidential customer's information such as Social Security Numbers, Date of Birth, Names, Credit Card information, etc.

Weak encryption can compromise the WebSphere MQ messages since an attacker can easily decrypt the data in a fairly short time. Decryption is a threat to Confidentiality.

Flooding: Hackers can conduct a denial of service (DOS) attack by sending random stop and delete commands to a command server, thus overloading the network bandwidth and bringing the network to a crawl.

If an attacker has gained access to a WebSphere MQ queue manager, the server may be over utilized, preventing access to other users or greatly affecting

[6] Saida Davies, Hazel Fix, Peter Rhys-Jenkis, Mayumi Kawashima, John Scanlan, Steven Lane. "WebSphere MQ Security in an Enterprise Environment" IBM Redbook Abstract, May 7, 2003 (p. 17)
http://www.redbooks.ibm.com/redbooks/pdfs/sg246814.pdf

[7] Saida Davies, Hazel Fix, Peter Rhys-Jenkis, Mayumi Kawashima, John Scanlan, Steven Lane. "WebSphere MQ Security in an Enterprise Environment" IBM Redbook Abstract, May 7, 2003 (p. 18)
http://www.redbooks.ibm.com/redbooks/pdfs/sg246814.pdf

the performance. The attacker may also be able to flood downstream queue managers by using the communication between queue managers. Flooding is a threat to availability[8].

Technology or Application Weakness: The TCP/IP protocol, some of its applications, and some operating systems were originally designed with the objective of openness, interoperability, and easy communication between computers and applications. Because of this, many of these technologies and applications have inherent security shortcomings.

Security holes in the underlying technologies such as platforms and network protocols directly affect the security of WebSphere MQ.

Company-developed applications such as WebSphere MQ adapters or software purchased from vendors may also contain security weaknesses that a hacker can exploit.

The degree of the damage depends on the nature of the problem. The most common problem is for a system to be shut down.

The problem could be more serious if an attacker is able to access data that they can modify or use to their advantage. Technology and application weaknesses exploited by malicious hackers are threat to all goals of the WebSphere MQ security.

To protect an enterprise, technology users must keep up to date with the vendor's security patches/updates and rely on vendors that command good reputation when it comes to their applications' security.

If a company decides to develop its own application to run on hosts, security must always be on top of the design goals, whether they interface into WebSphere MQ or not[9].

HACKING INTO A WEBSPHERE MQ QUEUE MANAGER

In order for Security Administrators to apply the best defense for their WebSphere MQ infrastructure is to understand the attacker's offense. The following provides a general overview on how an attacker may exploit security holes and gain access to a WebSphere MQ Queue Manager.

[8] Saida Davies, Hazel Fix, Peter Rhys-Jenkis, Mayumi Kawashima, John Scanlan, Steven Lane. "WebSphere MQ Security in an Enterprise Environment" IBM Redbook Abstract, May 7, 2003 (p. 18) http://www.redbooks.ibm.com/redbooks/pdfs/sg246814.pdf

[9] Saida Davies, Hazel Fix, Peter Rhys-Jenkis, Mayumi Kawashima, John Scanlan, Steven Lane. "WebSphere MQ Security in an Enterprise Environment" IBM Redbook Abstract, May 7, 2003 (p. 18) http://www.redbooks.ibm.com/redbooks/pdfs/sg246814.pdf

Via MQ Explorer

In order to connect to a WebSphere MQ queue manager, an attacker will need to know it's IP address, the queue manager name and the port on which it is listening. All of these information can easily be sniffed by a hacker using common industry known sniffing software that are available from the Internet such as Ethereal, PortFlash, NetStumbler, AirSnort, etc. Given network accessibility and enough time to sniff and collect network traffic, a hacker most likely will be able to capture all of the required information to start launching an attack.

WebSphere MQ has a feature that allows for remote administration. This is done via the MQ Explorer supplied with WebSphere MQ for Windows server. You right mouse click on the queue managers and type the name of the queue manager and it's IP address. The MQ Explorer then tries to create a channel between the queue manager and sends commands to the remote machine's queue manager.

If proper security was implemented for the administration queue, then a hacker would not be able to access the queue via the MQ Explorer. However, failure to properly secure the administration queue would give the hacker a free reign into your WebSphere MQ infrastructure.

Via the Web Administration Tool

Another way for hackers to crack into the WebSphere MQ queue manager is by attempting access through the web interface to manage MQ. Many MQ administrators may inadvertently installed this feature thus creating the risk for hackers to gain access using this mode.

Hackers can readily access IBM supplied tools for remote WebSphere MQ administration and perform a port scan to detect if port 8081 is active on a MQS machine. Port 8081 is the default port for the browser remote admin tool.

Once hackers are able to remotely access the WebSphere MQ queue manager, they will attempt to login to the machine. It is good practice for a System Administrator to change the default user ID and password upon installation of the application. Failure to do so is to grant hackers the green light into your system

Via a Channel

IBM creates default objects during the WebSphere MQ installation. Very few customers delete these default definitions as they are used to define a new object (if an user leave out a few parameters during the define, MQ picks up the remainder from the default definitions).

A hacker can assume that a channel existed in the MQS machine named SYSTEM.DEF.RECEIVER. Channel names need to be identical at both ends of

a connection for a channel to start, and with the sender-receiver channel one needs to be a sender and one needs to be the receiver – both with the same name.

A hacker using a laptop with Websphere MQ and connected to the network then may attempt to delete the SYSTEM.DEF.RECEIVER channel and re-create it as a SENDER channel. If no security planning was implemented during installation and the default settings were selected, then the attack may be successful, thus the channel would start on its first attempt.

This security breach would leave the hacker with access to send messages to the WebSphere MQ queues. Message formats and names could be "harvested" from a packet sniffer, allowing an attacker to use this mechanism to construct their own MQ messages and send them to the target MQ queues.

ADDITIONAL WEBSPHERE MQ EXPLOITATIONS.

Queue Manager Aliasing

WebSphere MQ series support a feature known as "Queue Manager Aliasing". If for example we have three queue managers - Que1, Que2 and Que3 and there are channels between Que1 and Que2 and Que2 and Que3, then one can send a message from Que1 to Que3 through the Que2 queue manager, making the Que2 queue manager in effect participant in the exchange.

The way it works is that in our example, a message would be created on the Que1 machine and the name of the remote queue manager for the message would be set to "Que3" the message would then be sent to Que2. The Que2 queue manager, when it receives the message, looks at the target queue manager and says "This is not for me, this is for someone called "Que3"" – and promptly pops it in the transmission queue that send it off to Que3, without anyone ever knowing it. Similar actions can be performed by creating ones own transmission headers and writing messages directly into the transmission queue.

Not defining queue manager alias' is one way to prevent this problem (although it does not stop the writing directly to the transmission queue problem), but some components of WebSphere MQ – such as clusters define these objects automatically, making the problem appear.

The only viable solution to these two problems is really to implement a "Message Firewall". No commercial products exist that do this, however sample prototype code can be found in several WebSphere MQ resource websites to provide ideas and from it implement their own implementation.

SYSTEM.COMMAND.INPUT

The queue used by WebSphere MQ command server (used for Administration) on Windows and Unix machines is named SYSTEM.ADMIN.COMMAND.QUEUE it accepts commands in a format called PCF which requires programming skills to create.

On a MVS environment, however, the command server accepts plain English commands such as STOP QMGR and the name of the queue is different – it's called SYSTEM.COMMAND.INPUT. Thus, proper security administration must be implemented to block an attack.

CLOSING SECURITY HOLES IN WEBSPHERE MQ SERIES

1. **Secure Default and Auto Channels.**
   Many successful attacks to WebSphere MQ are conducted by exploiting the fact that MQSeries installation creates default channels that most customers never delete or secure.

   When one creates a new channel, the system requires the user to input a few mandatory parameters, and then optionally input one or more of the optional parameters. If some optional parameters are not specified, the system obtains default settings from the default definition created at installation time.

   If these default definitions have been deleted, the queue manager will, in some instances require all of the optional parameters to be specified, and in other instances it will not allow the objects to be defined at all. That's why customers need to be extremely careful when deleting these objects, because it may be impossible to re-define them when you need to create a new object (such as a channel). One production solution to this problem is to have a runmqsc (the command line utility) script prepared that is capable of defining ALL of the MQSeries objects used by a queue manager. Such a script is often used for disaster recovery purposes and there are a number of support packs http://www.software.ibm.com/ts/mqseries that allow such a script to be dynamically created from a running MQSeries system.

   If the need arises to create a new object, the queue manager can be deleted and recreated from the script – although this would require that the queue manager be shut down for a short period of time.

   The alternative is to set the MCA user parameter on the at risk default channel definitions, bearing in mind that this parameter is subject to attack, however, such an attack will be noted in the AMQERR01 log (as

failed attempts to start a channel), thus enterprises may be able to monitor this log to detect unauthorized login attempts. In addition, it is recommended to delete the definitions that allow for channels to be automatically defined (SYSTEM.AUTO.*).

2. **Secure Administrative Queues.**
   For enterprises running on a Windows NT/2000 environment, a good practice to secure WebSphere MQ is to secure the SYSTEM.ADMIN.QUEUE . This can be easily done using the OAM (Object Authority Manager) SETMQAUT command. Access should only be granted to system administrators and monitoring applications that make use of them. This will instantly stop casual users from being able to remotely manage these machines while allowing only authorized to perform this function.

3. **Use OAM MQ Security.**
   The OAM (Object Authority Manager) is a command line driven application that allows the administrators to set and view permission associated with WebSphere MQ objects. Queues for example can be secured to the point that only certain users can be granted read and write access. The main problem with the OAM is that is fairly cumbersome to implement, thus many users rarely use it since it requires the administrator to create lengthy list associated with specific queue. Newer versions of WebSphere MQ makes this process easier to administer (because it allows wild cards – e.g. only let user A and B access queues that start with SYSTEM*)[10].

4. **Delete Default Items**
   As indicated earlier, the WebSphere MQ installation process places many default objects on a system that if not secured, can later compromise the infrastructure and be a target of an attack. As a rule of thumb in security nowadays, systems administrators must delete all default object that may plug a security hole in the infrastructure and carefully open only those objects/resources that are needed. Objects that start with the SYSTEM.DEFAULT should be carefully analyzed for their usage. If in doubt why an object is present, rename it and see if the system requires having it back, if not, delete it[11].

5. **Never have a Default Blank MCAUSER**

[10] Saida Davies, Hazel Fix, Peter Rhys-Jenkis, Mayumi Kawashima, John Scanlan, Steven Lane. "WebSphere MQ Security in an Enterprise Environment" IBM Redbook Abstract, May 7, 2003 (p. 260)
http://www.redbooks.ibm.com/redbooks/pdfs/sg246814.pdf
[11] Saida Davies, Hazel Fix, Peter Rhys-Jenkis, Mayumi Kawashima, John Scanlan, Steven Lane. "WebSphere MQ Security in an Enterprise Environment" IBM Redbook Abstract, May 7, 2003 (p. 260)
http://www.redbooks.ibm.com/redbooks/pdfs/sg246814.pdf

When defining channels, WebSphere MQ defaults "blank" for MCAUSER[12]. It is good security practice to disable this setting and define a proper MCAUSER.

6. **ALTUSER ID**
   Use of an alternate user ID in the MQMD should be carefully reviewed since it may allow a hacker to subvert object authority[13].

7. **Blank User Ids**
   RACF allows user IDs to be blank. Access to these user IDs must be carefully controlled and may be deleted, if not in use.

   In addition default security time-outs in certain MQ platforms (i.e. Z/OS, etc.) is large Customers should consider setting this to a shorter time to minimize the security risk exposure[14].

8. **Limit Production access to MQExplorer, PQEdit and similar utilities**
   Although these tools are great for developers to manipulate messages, they must be securely and restricted in production systems by placing them in secure directories. This would prevent any casual internal user/hacker access to these tools and potentially compromise the MQ infrastructure.

9. **Automate Dead Letter Queue Management**
   Dead Letter Queue's (DLQ's) are the first line of defense in a WebSphere MQ infrastructure that may be indicative that an attack is being performed against the infrastructure. Trivial Denial of Service (DOS) attacks can be constructed that simply send messages to unknown queues – which the queue manager will promptly place on its dead letter queue[15],

   A good security practice would be to implement a dead letter queue handling to stop the DLQ to be a single point of failure for a DLQ attack (Imagine if an attacker initiates a DOS attack against the MQ infrastructure, the DLQ would be filled in seconds, resulting in a queue manager shut down). A simple DLQ handler can be implemented in such

[12] Saida Davies, Hazel Fix, Peter Rhys-Jenkis, Mayumi Kawashima, John Scanlan, Steven Lane. "WebSphere MQ Security in an Enterprise Environment" IBM Redbook Abstract, May 7, 2003 (p. 261)
http://www.redbooks.ibm.com/redbooks/pdfs/sg246814.pdf
[13] Saida Davies, Hazel Fix, Peter Rhys-Jenkis, Mayumi Kawashima, John Scanlan, Steven Lane. "WebSphere MQ Security in an Enterprise Environment" IBM Redbook Abstract, May 7, 2003 (p. 262)
http://www.redbooks.ibm.com/redbooks/pdfs/sg246814.pdf
[14] Saida Davies, Hazel Fix, Peter Rhys-Jenkis, Mayumi Kawashima, John Scanlan, Steven Lane. "WebSphere MQ Security in an Enterprise Environment" IBM Redbook Abstract, May 7, 2003 (p. 262)
http://www.redbooks.ibm.com/redbooks/pdfs/sg246814.pdf
[15] Burnie Blakeley, Harry Harris and Rhys Lewis. "How to Develop and Integrate WebSphere MQ Messaging Applications", March, 2003
http://www.capitalware.biz/dl/docs/mq1guide.zip

a way that spills messages to a log file pair, allowing a queue manager to continue to operate.

10. Deploy Credential Based Security

Security experts talk about "two factor" authentication[16]. This requires that a principal (a person, or program) have two distinct factors – something that they know and something that they possess. The thing that they possess is usually called their credentials, the best-known form of which is usually called a digital certificate (CA). A good example would be an ATM system, the card is the thing they possess, and the PIN is the thing that they know.

Digital Certificate based authentication provides strong identification and authentication of principals – a principal in a WebSphere infrastructure would be a bridge or the application that is used to send/receive messages. Combined with other technologies such as asymmetric (Private Key-Public) encryption and symmetric (i.e. DES) encryption, they provide the basis for Authentication, Non-Repudiation, Privacy and Integrity when communicating via messaging[17].

Public Key Infrastructure (PKI) is used to describe a number of tasks that need to be carried out, including mechanisms for generation, distribution and management of public keys used in digital certificated (CA).

A basic digital certificate is an electronic string of bits issued to a principal by a Certificate Authority (CA), a CA supposedly checks supplied credentials (a driver's license for example) and issues a certificate that they electronically sign as being issued by them. These are good for a period of time (an hour, a day, a year, etc.) after which they expire.

Deploying credential-based security in a WebSphere MQ infrastructure can be implemented using different approaches. Design and implementation modes will depend on the criticality of the data to be protected, numbers of MQ messages processed, channels' protocol, project budget, etc. Credential based security can be deployed using Link-oriented security or end-to-end security.

- ***WebSphere MQ Link-Oriented Security***
  A link-oriented security solution on WebSphere MQ will be a MQ Channel solution. Typically, exits will be used that are provided by

[16] Paul de Graaff, IBM Field Technical Sales Specialist. "Cross Platform Security using IBM's Websphere; take the Security Challenge"
www.cgisecurity.com/lib/pauld.pdf

[17] Craggs, Steve. "MQSoftware EAI Survival Guide, How to Manage the Complex Middleware and EAI Terrain".
http://www.mqsoftware.com/product/resource.jsp

the application at this level (Security exit, Msg exit and Send/Receive exit). Security, in practice, will involve all components at the communication channel level.

WebShere MQ provides a relevant set of channel exits[18], which can be used to implement link-oriented security solutions.

The Security Exit is used once per session. It is therefore particularly useful for implementing the PEA protocol.

The Message Exit is used once for each message crossing the channel. It is therefore the ideal exit for implementing cryptographic services such as data encryption and authentication.

The Send/Receive exit is a lower level exit where is possible to handle the individual physical blocks into which a message is broken down before being send, This is also a good point for applying cryptographic operations on data. Compared with the message exit, however, it poses performance problems, since the cryptographic functions are more efficient if they operate on a larger portion of data. It is, however, the only practical solution to secure a Client Server relationship, since the MQ client does not have a message exit, only a S/R exit.

In addition both S/R and message exits are suitable for performing compression operations.

If compression to the data is to be carried out, one must pay attention to the order in which the operations are carried out. Compression must be carried BEFORE encryption, and therefore decryption will take place before decompression.

Applying encryption to a compressed text provides cryptographic benefits. Since encryption is performed on a text whose characters are now distributed statistically more evenly, it makes the job of the cryptanalyst trying to perform a frequency analysis attack even more difficult.[19]

- ***WebSphere MQ End-to-End Security***
  An End-to-End (E2E) solution on WebSphere MQ will typically intervene at an application level and will concern this level or higher

[18] Stuart C. Jones. "WebSphere MQ Security White Paper. A white paper on implementing WebSphere MQ security", April, 1999
http://www.capitalware.biz/dl/docs/mqseries_security_white_paper.zip
[19] Cesare San Martino. "Securing an MQ Infrastructure, MQSoftware Resource Center White Paper", July 2002. (p. 9)
http://www.mqsoftware.com/products/docs/Securing_an_MQ_infrastructure.pdf

level of the actual user. Deploying E2E security with digital certificates and encryption is the very best way to ensure message integrity.

Whether a transparent solution or non-transparent solution is chosen, functionally one will have to do the following

Sending Side:

- Compression call (optional)
- Cryptographic call
- Message PUT

Receiving Side:

- Message GET
- Decryption
- Decompression (if necessary)

To implement E2E solutions, it is absolutely recommended to use digital signature functionality in conjunction with the most accepted standards for this purpose: PKCS #12 and S/MINE.

Several cryptographically tools are available commercially, with which these functions can be implemented. In general, all tools now allow digital signature to be implemented with a single high-level call, which in turn results in several internal cryptographic calls. In the most complete case of encryption and signature, these operations are:

- Generation of a symmetric session key
- Encryption of the data with the symmetric key
- Encryption of the symmetric key with the public key of the target recipient
- Generation of a MD
- Encryption of the MD with a private key, thus creating the MAC
- Enveloping of the data obtained in one of the standard "Digital Envelopes"
- To the above are added the operations of access to the private key (usually on a token) and retrieval of the correspondent's public key.

The use of standard cryptographic tools guarantees that all the operations outlined above can be performed with a single call. In addition, it is recommended to make the cryptographic calls flexible

and relatively independent of the application programmer through the use of “virtual codes”[20]

[20] Cesare San Martino. “Securing an MQ Infrastructure, MQSoftware Resource Center White Paper”, July 2002 (p. 10) http://www.mqsoftware.com/products/docs/Securing_an_MQ_infrastructure.pdf

CONCLUSION

As more e-commerce sites are adopting and embracing middleware distributed messaging systems like WebSphere MQ as part of their strategic e-commerce solution, the need to properly secure these systems will grow accordingly.

Implementing and deploying a security solution for WebSphere MQ, one needs to be aware both of the security technology as well as of the MQ related technology, data involved and architecture.

Defining and understanding the specific security objectives are the keys to properly secure a WebSphere MQ infrastructure. Without clear guidelines or objectives, users will definitely leave security holes in the infrastructure that malicious attackers will most likely gain unauthorized access to the data and compromise the business.

Another aspect in deploying security in a WebSphere MQ environment, is the need to take into account that deploying a security solution implies designing a combined security/MQ architecture; the functional behavior of the whole system will depend not only on the chosen specific functions but also largely on the implemented architecture.

As a final note, the approach of "***defense-in-depth***[21], " which advocates the use of multiple layers of protection to guard against failure of a single security component must always be taken into consideration when implementing security not only to a WebSphere MQ environment, but also to all computing subsystems. No single "Silver Bullet" application will be able to 100% protect and secure a system, but rather a combination of best-of-the breed applications, appliances and hardware will be needed to properly secure an enterprise's infrastructure.

[21] Cole, Eric. Fossen, Jason. Northcutt, Stephen. Pomeranz, Hal. "SANS Security Essentials and the CISSP 10 Domains". SANS Press, 2003.

References

Craggs, Steve. “MQSoftware EAI Survival Guide, How to Manage the Complex Middleware and EAI Terrain”.
http://www.mqsoftware.com/product/resource.jsp

8th Annual CSI/FBI Computer Crime and Security Survey
http://www.yle.fi/mot/kj040524/fbiraportti.pdf

Cole, Eric. Fossen, Jason. Northcutt, Stephen. Pomeranz, Hal. “SANS Security Essentials and the CISSP 10 Domains”. SANS Press, 2003.

Candle, Corp. “How to Conquer Security Challenges in Distributed Messaging Environments”,March, 2003
http://www.capitalware.biz/dl/docs/mq5guide.zip

“Special Report: Salgado case reveals dark side of electronic commerce” Richard Power, CSI Monthly Newsletter, September 1997.

Burnie Blakeley, Harry Harris and Rhys Lewis. “How to Develop and Integrate WebSphere MQ Messaging Applications”, March, 2003
http://www.capitalware.biz/dl/docs/mq1guide.zip

Stuart C. Jones. “WebSphere MQ Security White Paper. A white paper on implementing WebSphere MQ security”, April, 1999
http://www.capitalware.biz/dl/docs/mqseries_security_white_paper.zip

Paul de Graaff, IBM Field Technical Sales Specialist. “Cross Platform Security using IBM's Websphere; take the Security Challenge”
www.cgisecurity.com/lib/pauld.pdf

Cesare San Martino. “Securing an MQ Infrastructure, MQSoftware Resource Center White Paper”, July 2002
http://www.mqsoftware.com/products/docs/Securing_an_MQ_infrastructure.pdf

Saida Davies, Hazel Fix, Peter Rhys-Jenkis, Mayumi Kawashima, John Scanlan, Steven Lane. "WebSphere MQ Security in an Enterprise Environment" IBM Redbook Abstract, May 7, 2003
http://www.redbooks.ibm.com/redbooks/pdfs/sg246814.pdf