
Building Secure B2B Applications using SSL

Mike McNamee

e3 Sciences

Mikem@e3sciences.com

Agenda

- Introduction
- Typical Business requirements
- Review of available technologies
- Building a custom SSL Application-where do I start ?

Introduction

- Project to provide secure multi-platform (Win32, UNIX, Mainframe) B2B data movement
 - Review of business requirements
 - Review of existing technologies
 - Determine suitable approaches
- This is a summary of our findings to help others in the same process

Typical Business requirements

- Lets talk, lets do business!



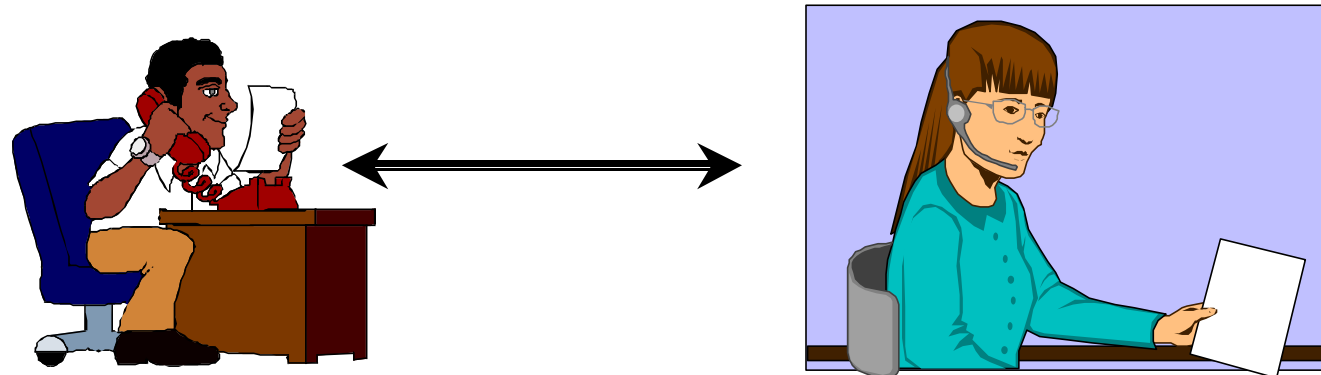
- But How ????

Typical Business requirements

- Types of interaction between organizations over the years:
 - User-to-User
 - User-to-Application
 - Application-to-Application

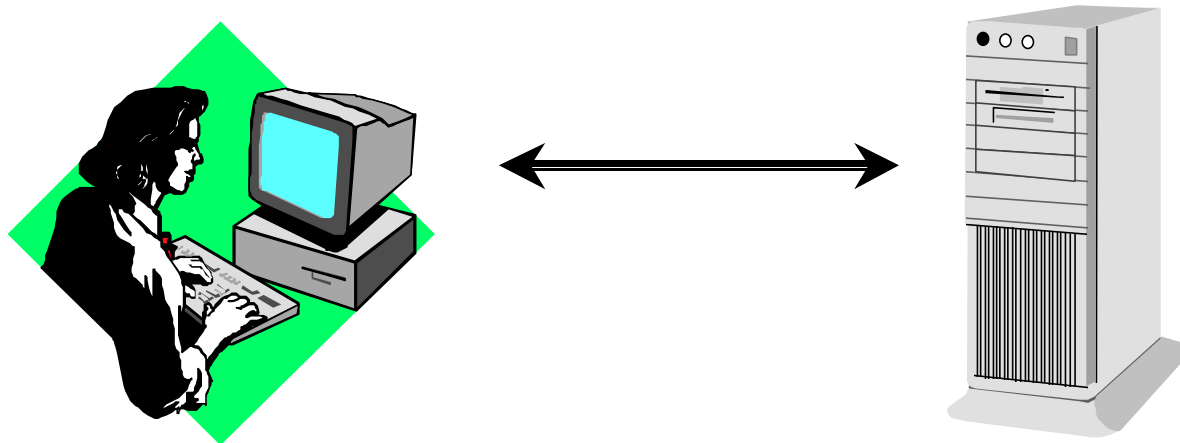
Interaction between Organizations

- User-to-User
 - Transaction is conducted by direct person to person communication in a spoken language
 - Mediums used Telephone, Postal, Email, Fax, Telex



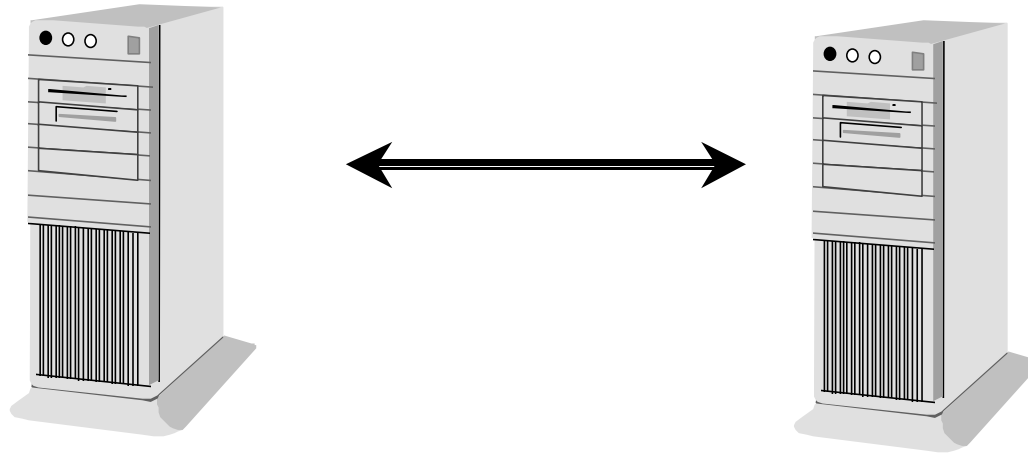
Interaction between Organizations

- User-to-Application
 - User connects to application system owned by another organization to request service



Interaction between Organizations

- Application-to-Application
 - Applications systems directly exchange data to request service in a controlled and secure manner



- Of all methods this method offers the greatest benefit as it's totally automated

Interaction between Organizations

- The Application-to-Application model is what is generally meant by B2B communication
- This is not a new technology :
 - EDI has offered a solution for many years
 - Costs and infrastructure made EDI difficult
- Universal Internet access provides infrastructure
 - Users need to select technology to use for B2B applications

Review of available technologies

- Email/Messaging
- Virtual Private Network
- Web Browser/Server and associated technologies
- Secure File Transfer
- Toolkits

Email/Messaging

- Easily accessible technology
- Most email packages provide good security
- Can be automated with MAPI
- Store-and-forward means poor control
- Not designed for application-to-application communications



Virtual Private Network

- Provide encrypted tunnel between two locations
- Data can be securely exchanged in a number of ways:
 - File copy/transfer
 - Microsoft Networking (SMB) works easily between systems, non MS systems can use Samba
 - FTP

Web Browser/Server etc.

- Provide easy secure data movement as standard
 - Java URL class
 - IE WinInet APIs
 - Java SSE (part of HotJava)
- Designed for user to application transactions
 - Application must emulate a user
 - Application must run under a browser
 - Connectionless protocol

Secure File Transfer

- Move data files between business partners
- File movement provides easy to use API, no extensive training required
- Not transaction oriented
- Easy to use solution

Toolkits

- Stand alone toolkits are available for users to develop their own SSL applications :
 - BSAFE SSL from RSA
 - Open SSL
 - System SSL on IBM OS/390
 - SSE from Sun Microsystems (Java)

Building an SSL Application

- We needed to build an SSL application. Our requirements :
 - Multi -Platform support (Unix, Win32, IBM OS/390)
 - Minimum pre-requisites
 - Cost effective SSL implementation
 - Cost should not be an obstacle

Building an SSL Application

- Web Browser/Server provides easy encryption but not well back-end applications
 - Must run under browser
 - Must use HTTP Post/Get primitives
- Many stand alone SSL implementations have disadvantages to :
 - Are proprietary with limited platform support
 - costly licensing

Building an SSL Application

- We chose :
 - OpenSSL is platform independent - provides standard API across many systems. It's free also!
 - System SSL for IBM OS/390
 - IBM implementation uses built-in Crypto Co-processor, can manage >2000 SSL transactions a second!

Getting Started

- Download software from www.openssl.org
- Run Make file to build software
 - Builds 1 Runtime libraries and 2 stub files
 - libeay32.dll
 - libeay32.lib
 - ssleay32.lib
- Generate Private/Public key pair and Server certificate signing request (PKCS#10)
 - `openssl req -newkey 1024 -config openssl.cnf -out newreq`

Getting Started

- Check your certificate request using
 - openssl req -text -in req

Certificate Request:

Data:

Version: 0 (0x0)

**Subject: C=UK, ST=Glos, L=Stonehouse, O=e3 Sciences,
OU=Development, CN=
MikeMcNamee/Email=mikem@e3sciences.com**

Subject Public Key Info:

Public Key Algorithm: rsaEncryption

RSA Public Key: (1024 bit)

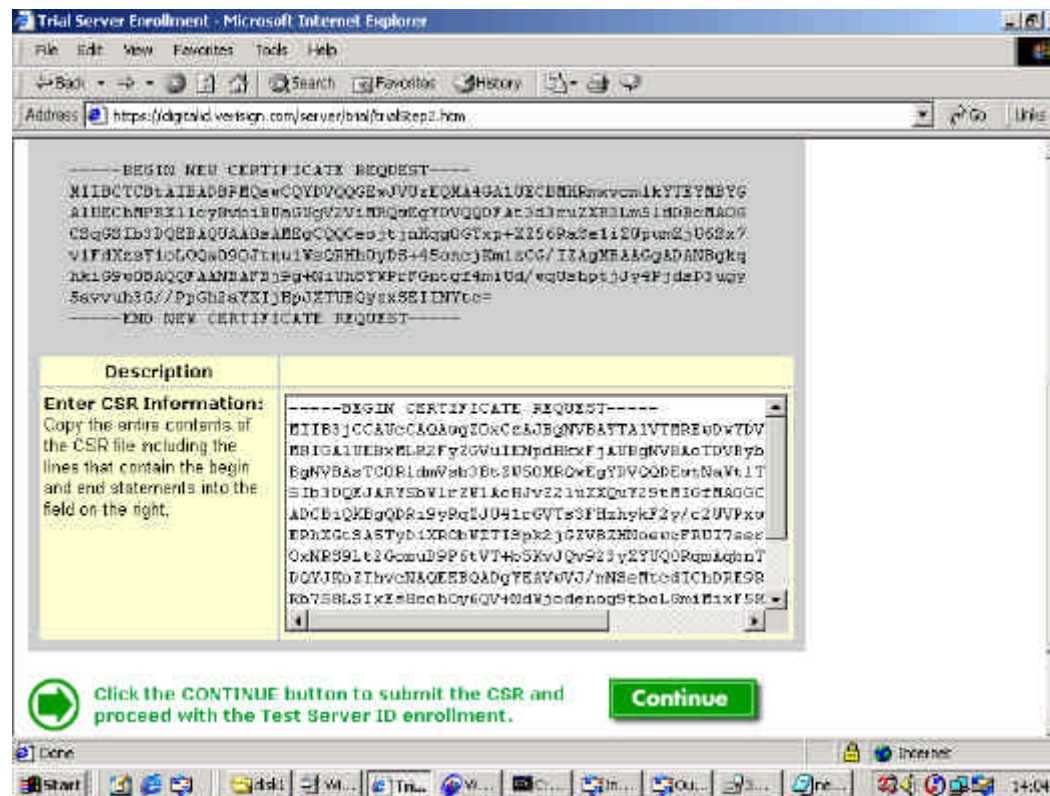
Modulus (1024 bit):

00:aa:57:5a:fb:80:34:08:5a:36:26:6b:37:b5:3a:

.....

Getting Started

- Get certificate generated by a CA such as Verisign:



Getting Started

- Certificate arrives by email
- Certificate is generated by Verisign test CA
 - Follow instructions in email to get CA Certificate
- Write both to disk
 - Program references location of files

Getting Started

- Write a client program a sample is provided in OpenSSL demos\ssl\ cli.cpp and serv.cpp:
- “Hello World” TYPE samples supplied and available in electronic format from e3Sciences Web Site www.e3sciences.com
 - OpenSSL and OS/390 samples available

Getting Started

- On OS/390 SSL operates under Unix System Services (Unix part of OS)
 - Database of keys and certificates maintained by key management utility gskkyman
 - Instructions in System SSL Programming Guide and Reference SC24-5877-01



?

?

?

?

?

?

?

?

?

?

Questions ?

?

?

?

?

?

?

?

?

?

?

?

?

?

?