

FIX Security White Paper

Revision 1.8

February 5, 2008

DISCLAIMER

THE INFORMATION CONTAINED HEREIN AND THE FINANCIAL INFORMATION EXCHANGE PROTOCOL (COLLECTIVELY, THE "FIX PROTOCOL") ARE PROVIDED "AS IS" AND NO PERSON OR ENTITY ASSOCIATED WITH THE FIX PROTOCOL MAKES ANY REPRESENTATION OR WARRANTY, EXPRESS OR IMPLIED, AS TO THE FIX PROTOCOL (OR THE RESULTS TO BE OBTAINED BY THE USE THEREOF) OR ANY OTHER MATTER AND EACH SUCH PERSON AND ENTITY SPECIFICALLY DISCLAIMS ANY WARRANTY OF ORIGINALITY, ACCURACY, COMPLETENESS, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. SUCH PERSONS AND ENTITIES DO NOT WARRANT THAT THE FIX PROTOCOL WILL CONFORM TO ANY DESCRIPTION THEREOF OR BE FREE OF ERRORS. THE ENTIRE RISK OF ANY USE OF THE FIX PROTOCOL IS ASSUMED BY THE USER.

NO PERSON OR ENTITY ASSOCIATED WITH THE FIX PROTOCOL SHALL HAVE ANY LIABILITY FOR DAMAGES OF ANY KIND ARISING IN ANY MANNER OUT OF OR IN CONNECTION WITH ANY USER'S USE OF (OR ANY INABILITY TO USE) THE FIX PROTOCOL, WHETHER DIRECT, INDIRECT, INCIDENTAL, SPECIAL OR CONSEQUENTIAL (INCLUDING, WITHOUT LIMITATION, LOSS OF DATA, LOSS OF USE, CLAIMS OF THIRD PARTIES OR LOST PROFITS OR REVENUES OR OTHER ECONOMIC LOSS), WHETHER IN TORT (INCLUDING NEGLIGENCE AND STRICT LIABILITY), CONTRACT OR OTHERWISE, WHETHER OR NOT ANY SUCH PERSON OR ENTITY HAS BEEN ADVISED OF, OR OTHERWISE MIGHT HAVE ANTICIPATED THE POSSIBILITY OF, SUCH DAMAGES.

DRAFT OR NOT RATIFIED PROPOSALS (REFER TO PROPOSAL STATUS AND/OR SUBMISSION STATUS ON COVER PAGE) ARE PROVIDED "AS-IS" TO INTERESTED PARTIES FOR DISCUSSION ONLY. IT IS A DRAFT DOCUMENT AND MAY BE UPDATED, REPLACED, OR MADE OBSOLETE BY OTHER DOCUMENTS AT ANY TIME. IT IS INAPPROPRIATE TO USE FPL WORKING DRAFTS AS REFERENCE MATERIAL OR TO CITE THEM AS OTHER THAN "WORKS IN PROGRESS". THE FPL GLOBAL TECHNICAL COMMITTEE WILL ISSUE, UPON COMPLETION OF REVIEW AND RATIFICATION, AN OFFICIAL STATUS ("APPROVED") TO THIS.

No proprietary or ownership interest of any kind is granted with respect to the FIX Protocol (or any rights therein).

Copyright 2003-2008 FIX Protocol Limited, all rights reserved

This document is intended to provide FPL members with some of the common questions and answers regarding computer and network security when using FIX. Its scope is limited to the FIX Protocol and transmission of FIX messages between parties; issues such as security of operating systems, internal applications, databases, etc. are outside the scope of this document.

Secure use of FIX relies on industry standard security infrastructure.

FIX provides a simple and flexible transaction protocol to bring together the buyers and sellers of securities.

The creators and maintainers of the FIX Protocol envision the FIX Protocol being integrated and used in conjunction with existing security mechanisms. Infrastructure controls such as authentication and encryption are provided at a different layer than the FIX Protocol.

Questions and Answers

Is using FIX a risk from a security perspective?

Any computer-to-computer communication comes with security risks. These risks are increased when communicating with external parties and networks. However, external electronic communication is a fundamental requirement in the financial markets. Therefore, all participants must employ appropriate security measures and diligently manage security risk.

The FIX Protocol leaves the choice of appropriate security measures open to the user community.

Why doesn't the FIX Protocol provide its own security mechanisms?

Developing security technology and protocols, and developing financial transaction management protocols, require different skill sets. The core expertise of the FIX Protocol organization is financial transaction management. The Global Information Security Subcommittee believes that the best way to improve security for the FIX protocol is to rely on standard security technology. This is deemed a lower risk strategy when compared to creating comparable security capabilities within the FIX Protocol itself, and doing so allows for adoption of emerging security technology.

Can Internet hackers compromise my FIX connections?

Firms should not leave their FIX gateways inadvertently or insecurely exposed on the Internet where they would be susceptible to attacks by Internet hackers. Internet traffic should always be segregated from internal networks, including those networks that are used to transmit FIX messages using firewall technology, with the exception of deliberate and carefully planned instances where Internet FIX traffic is allowed.

Do I need to use encryption for my FIX session?

The use of encryption is considered essential on open networks like the Internet. However, encryption is often not used on private networks, such as dedicated leased lines. It often is also not used on extranets that have been secured, such as VPNs or

point to point extranet providers serving the financial community. Using FIX over a private network or secured extranet runs a risk of someone eavesdropping on the network traffic, viewing the FIX transactions as they travel over the network. However, such risk is mitigated by traditional security means of restricting access to points of presence where communication equipment is housed. Extranet providers can employ different methods to provide varying levels of security; their users should assure themselves that the methods employed are adequate prior to deciding not to employ encryption on their FIX connections.

What are the advantages and disadvantages of using the Internet for FIX?

Use of the Internet for FIX sessions often comes at a significantly lower cost than use of private networks and extranets. It also eliminates the provisioning delays sometimes associated with private networks.

However, use of the Internet for FIX traffic has several drawbacks.

Some form of encryption, while sometimes considered optional over private networks, is required to protect the confidentiality of Internet traffic. This can add latency to FIX messages, and complicate configuration and maintenance of FIX gateways.

No one party owns the Internet, hence no one party can be held responsible for Internet security and availability. The path data takes between firms often is not fixed, and may travel through several different Internet providers and access points. As such, it is difficult to receive any meaningful guarantee of uptime, bandwidth or latency for end to end connectivity. Unlike the Internet, private networks and secured extranets often include Service Level Agreements concerning uptime, latency and bandwidth.

Reliance on the Internet to conduct business carries with it the possibility of Denial of Service (DoS) attacks, in which an attacker attempts to disrupt business by flooding a firm's network with data. While defenses against such attacks exist, they are not foolproof. These risks are reduced by use of private networks or secured extranets.

Many firms make considerable investments in high speed, low latency Internet connectivity. Two firms with high speed Internet connectivity might find better latency and bandwidth by using the Internet instead of a private network or secured extranet. These factors can be weighed against the lack of guarantees, the possibility of unexpected connectivity degradation, and the overall exposure.

Each firm should carefully consider the cost, benefit, and risk of using various network transport options for FIX. For example, firms may decide against use of the Internet for mission critical FIX sessions, and may or may not choose to use the Internet for FIX sessions of less critical importance, such as testing, certification, or infrequently used production sessions, or as emergency backups. Alternately, some firms have security policies that require the use of hardware based encryption technology over private leased lines.

How can I encrypt FIX messages?

Originally, several methods of encrypting the contents within a FIX message using standard industry encryption algorithms were employed. An Application Note is available on the FIX Protocol website that describes the most common of these approaches, PGP/DES-MD5. However, advances, primarily in computational power, have reduced the effectiveness of the DES encryption algorithm used, and these are no longer

recommended as a best practice for encrypting FIX messages. The use of these methods has been deprecated as of FIX 5.0 SP1; use of these fields is discouraged.

As a best practice, use of existing technologies that sit beneath the FIX session layer and encrypt the communications transport layer itself is highly recommended.

A Virtual Private Network (VPN) can be created between parties using encryption in software or network hardware. A VPN encrypts the data being transmitted between two parties. With this model, security is external to the FIX software used by both parties; the software does not actively participate in the security protocol and needs no modification or support for encryption.

Another alternative is to create a “tunnel” through the Internet or other network using Secure Sockets Layer (SSL) or Transport Layer Security (TLS). Applications that use FIX can incorporate SSL and TLS libraries directly. Several commercial libraries exist, as well as at least one open source library, OpenSSL. FIX applications can also connect to a proxy application within their firm, which then establishes an SSL or TLS connection at another firm. With this approach, the FIX application itself does not need to be modified to support encryption. One example is the open source software program Stunnel.

What if I want to encrypt only password fields?

In markets where regulators require that passwords be encrypted, but do not extend these requirements to orders, encrypted passwords may be passed using the fields EncryptedPassword and EncryptedPasswordLen, defined in FIX 5.0 SP1, in the Logon and UserRequest messages. Support for changing passwords is also provided in these messages.

The field EncryptedPasswordMethod is used to define the method for encrypting passwords. At present, FIX Protocol Ltd. has not defined any such methods; implementers may create proprietary methods which may later be submitted for standardization.

Care should be taken to note what security encrypted passwords provide, and what protection they do not provide. Simply encrypting only passwords is not a control against network eavesdropping of transactions. Depending upon the design of the encryption method, it may or may not be a control against replay attacks. Each firm should evaluate the benefits and risks associated with encrypting only passwords in determining their security policies.

When should I use additional security with the FIX Protocol?

At a minimum, a firewall should always be used regardless of the type of network you are using. Firms should adopt stringent security measures including encryption when sending FIX messages over the Internet. If your business can be compromised by someone eavesdropping on your FIX communications, and such communications are potentially vulnerable to eavesdropping, then some form of encryption is recommended. Potentially vulnerable connections include but are not limited to the Internet and network links that are managed by unknown or untrusted parties.

How do I secure a FIX session?

The following technologies are available for use in securing a FIX session.

- Private leased lines traversing secure facilities
- Virtual Private Network (VPN) using network protocols such as IPsec
- Use the FIX Protocol with software-based security (e.g. PKI software like Secure Sockets Layer (SSL) or Transport Layer Security (TLS) libraries, Stunnel, etc.), referred to as Tunneling. Be aware that SSL and TLS should be configured only to use strong encryption and authenticate both parties. Other less secure options supported by SSL and TLS, including weaker encryption algorithms and less stringent authentication models, should be avoided. Sometimes SSL or TLS libraries may allow a user to override certain errors, such as client software that presents a dialog box that allows a user to choose to accept an invalid or expired certificate; this practice should be disabled to prevent compromises of security policies.
- Use the FIX Protocol with hardware-based security (e.g. separate encryption hardware, link-based encryption)
- Use of a specialized network provider that provides security as part of their value-added service

What basic security steps should I take when implementing FIX?

1. When connecting to external entities via the Internet, use a VPN or some form of tunneling technology
2. Consider use of hardware-based encryption technology
3. Where keys or certificates are employed to secure a FIX session, they should be stored securely
4. Implement procedures to revoke and change keys or certificates in the event of actual or suspected compromise
5. Limit the access to systems and networks to essential personnel, and maintain procedures whereby access is revoked when no longer needed
6. Log all access to the FIX systems if possible
7. Use properly configured firewalls
8. Secure all TCP ports on the external-facing side of the firewall, and open ports only as needed for counterparty connections
9. Separate networks for general Internet and email communications from networks that carry business transactions, such as FIX messages
10. Restrict access to networks that carry business transactions
11. Regularly audit all security procedures and system and network configuration

How can I authenticate counterparties using FIX?

The most basic counterparty authentication is the verification of the FIX field SenderCompID. It is conceivable that a party may attempt to send the SenderCompID of

another firm, accidentally or deliberately, thereby “spoofing” the other firm’s identity. In practice, this is unlikely; proper security procedures should prevent Internet hackers from connecting to a FIX system, and an authorized firm using its own network connections to spoof the identity of a competing firm would promptly be detected and incur severe consequences. Still, firms should exercise due diligence in preventing such spoofing from occurring.

While not limited to the TCP protocol, the vast majority of FIX sessions use TCP, so FIX authentication over TCP will be addressed. Even in the absence of encryption, authentication can be achieved by associating the business-level identity of a counterparty (the FIX SenderCompID) to an IP address or range of IP addresses used by that counterparty, and validating that the IP address matches the associated SenderCompID.

IP addresses themselves can be spoofed or forged. This threat is greatest over the Internet, and it is one of the reasons why encryption is considered essential when using the Internet. However, even IP addresses on private leased lines can be spoofed. Proper network and firewall configuration should be employed to reduce or prevent spoofing. For example, if a specific IP address is used by a counterparty, it is associated with a FIX SenderCompID, and a leased line; a firewall should allow that IP address over that line only, and prohibit it from being used on leased lines to other counterparties, and, especially, the Internet. Routers themselves should be secured so that firewall rules cannot be modified. Access lists, route filters, authentication keys, etc. should be used to prevent dynamic routing protocols from rerouting data over unauthorized paths.

These are only a few examples; network security is a complex field outside the scope of this document. Firms should use best practices to prevent IP address spoofing.

How can I authenticate counterparties without encryption?

The initiator, or client, generally connects to a previously configured IP address or list of IP addresses for primary and backup servers, via a previously configured TCP port(s). Provided the network guarantees a secure path to the counterparty, then simple verification of the other party’s SenderCompID should suffice.

The acceptor, or server, has a more complex job. It may listen for counterparty connections on one or more TCP ports. One of two general approaches is often employed:

1. The FIX server may have a configured list of allowed IP addresses associated with a given SenderCompID. If the client’s IP address is not on that list, then the connection should be dropped immediately, without sending a Logout message or incrementing FIX sequence numbers, and a warning or alert should be generated.
2. FIX servers that cannot verify IP addresses should be configured to use one unique TCP port per FIX session, and verify that for each port only one configured SenderCompID is used. Then, a network firewall or proxy server will validate that the configured client IP address for one party may connect only to that party’s assigned server IP address on its assigned TCP port. Care should be exercised to prevent outside traffic from bypassing the firewall and connecting to the server directly.

How can I authenticate counterparties with network encryption?

Hardware or software network security should be transparent to the FIX systems involved, and the procedures above should apply. Appropriate measures should be taken to prevent firms from bypassing the encryption and authentication systems and connecting directly to the FIX system.

How can I authenticate counterparties with SSL or TLS?

With SSL or TLS, each party has an X.509 certificate, and can validate the validity of the counterparty's certificate. Specifics of this process, and how to associate X.509 certificates with FIX SenderCompID values, are outside the scope of this document.

A FIX engine that has built-in support for SSL or TLS should be able to verify that the engine only accepts certificates owned by the firm's counterparties, and that the SenderCompID used in any given session corresponds with the identity of the counterparty as established by the certificate.

A FIX initiator or client that uses an external proxy, like Stunnel, would connect to the IP address and port of the Stunnel proxy without encryption. Stunnel would, in turn, be configured to validate that the inbound connection came from the client FIX system. It would connect to the IP address of the FIX server, or another proxy like Stunnel running at the site of the FIX server, using SSL or TLS encryption and authentication. Stunnel should be configured to reject the connection if the remote counterparty is not using an expected certificate. On success, it would accept unencrypted data from the client, encrypt it, and send it to the server; encrypted responses from the server would be unencrypted and sent to the client.

A FIX acceptor, or server, that does not have built-in support for SSL or TLS, but relies on an external proxy like Stunnel, should be configured to use one unique TCP port per FIX session and verify that for each port only the one configured SenderCompID is used. Stunnel is configured to accept an SSL or TLS-encrypted connection and validate that the IP address and certificate match what is expected. Upon successful matching, Stunnel will then initiate an unencrypted connection to the FIX server's TCP port corresponding with that counterparty's identity. It would accept encrypted data from the client, unencrypt it, and send it to the server; unencrypted responses from the server would be encrypted and sent to the client.

When using external proxies like Stunnel, care should be taken to prevent counterparties from bypassing Stunnel or connecting to the wrong side. For example, if Stunnel is used to accept SSL or TLS FIX connections, care should be taken to prevent a client who knows the actual server's IP address and port from bypassing Stunnel by connecting directly to the server. Or, if Stunnel is used to accept unencrypted inbound FIX sessions and initiate outbound SSL or TLS FIX connections, measures should be taken to prevent an improperly configured client or an unauthorized user from connecting to Stunnel, causing Stunnel to make a secure connection using the firm's X.509 certificate to a counterparty, and using the firm's certificate to authenticate the malicious or improperly configured user's FIX messages.

If I use SSL or TLS, what considerations should be made regarding certificate management?

Within SSL or TLS, each party proves their own identity, and authenticates their counterparty's identity, via X.509 certificates. A full explanation of X.509 certificate management is outside the scope of this document.

An X.509 certificate consists of several parts, which include:

- A private key, which should be generated by the user and must be kept secret.
- A public key, which is generated at the same time as the private key. This key is disclosed to one's counterparty when an SSL or TLS connection is established.
- Information identifying the owner of the certificate.
- A digital signature which is issued by a Certificate Authority (CA).

The security of the system relies heavily on the ability of the owner of the certificate to protect the private key and keep it confidential. Gaining possession of the private key can lead to an unauthorized person successfully impersonating the legitimate certificate holder.

The CA, by issuing a signature that becomes part of the certificate, vouches for the identity of the user. Note that the CA never needs to know the private key of the user, nor should the CA generate the user's public and private key. Rather, the user should generate the public and private key and create a Certificate Signing Request (CSR) that is sent to the CA. Once the CA validates the user's identity, it will sign the certificate and transmit the signed certificate back to the user.

Certificates have specified expiration times. Certificates must be renewed prior to certificate expiration, otherwise the connection should fail. This must be watched diligently; allowing a certificate to expire without renewing it can lead to an unexpected outage. The CA itself has a certificate, and should the CA's certificate expire, any signature on a user's certificate is no longer valid. In some cases, authenticity is proved through a chain of CA's, with one party vouching for the next; should any certificate in the chain expire, the user's certificate, likewise, expires.

In order to validate a certificate, one's counterparty must have a current, trusted copy of the CA's certificate. Both parties must agree upon which CA or CA's to trust to issue the certificates used by each end of the FIX session. Several models for this exist:

- A trusted third party could be used to sign a certificate.
- Each party could operate their own CA.
- One party could operate a CA and sign their own certificate, as well as the certificate of the other party. (Note that this is not the same as generating the other party's public and private keys, mentioned above.)
- A certificate can be self-signed, where the owner acts as his or her own CA.

Depending on the authentication mechanism used, a CA may or may not have the ability to generate certificates that could spoof the identity of the party; therefore choice of the CA is important.

Authentication mechanisms and associated caveats include:

- **Subject Canonical Name (CN)** – Each certificate contains a field called CN within the Subject part of the certificate. This could represent the identity of the user. For example, a sell-side doing business with Buy-Side One might require that only certificates with CN="BUYSIDE1" be accepted, and that these correspond to FIX CompID "BUYS1". With this model, certificate expiration can be handled easily. Buy-Side One can request a new certificate from the CA, and as long as CN="BUYSIDE1", and the CA certificate is still valid, then the sell-side will not need to make configuration changes. Security in this model depends on every CA the sell-side trusts; if any of them issued a certificate with CN="BUYSIDE1" to another party, either deliberately, accidentally, or because their own security was compromised, then that party could impersonate Buy-Side One. The sell-side must be careful not to choose to trust any CA who might issue such a certificate to another party. Many SSL and TLS libraries ship with default CA's; if they are not trusted, then they should be disabled.
- **Issuer and Serial Number** – Buy-Side One informs the sell-side that they will be using a certificate issued by "Big CA Inc." with serial number "1234". The sell-side then configures their SSL or TLS software to require these parameters. In this case, when Buy-Side One renews their certificate, they must coordinate with the sell-side, since the renewed certificate will likely have a later serial number. However, security now does not depend upon what additional certificates the CA chooses to sign; rather, it only requires that, in this example, Big CA Inc, does not reissue a new certificate with serial number "1234". This is always true with normal operating practices, however it may not be true should the CA's internal security be compromised.
- **Require specific certificates** – Buy-Side One gives the sell-side a copy of the public portion of their certificate. Sell-Side One then verifies on connection that the certificate presented, and the configured certificate, match. This option is used by Stunnel. As with Issuer and Serial Number, renewing a certificate requires coordination between parties. However, it does not rely on the security of the CA.

When using Stunnel, one should observe the following precautions:

- Stunnel needs the "-v 3" option, which is described as "Require and verify certificates against locally installed certificates."
- When authenticating certificates, Stunnel looks in a directory specified with the "-a" certificate_dir option. Using one global directory for all counterparties means that any counterparty can impersonate any other counterparty. It is a better practice to maintain a directory per counterparty, and run a separate Stunnel instance that uses this directory.

- It often is best to use the “-S 0” option which is defined as “ignore all default sources” and will disable all default CA’s. One can then explicitly choose an appropriate CA by creating a file with that CA’s certificate and specifying that file using the “-A” Certificate Authority File option.

More information on Stunnel arguments is available at the following URL:
<http://www.stunnel.org/faq/args.html>

What considerations should be made in developing or testing FIX software?

Best practices should be used in constructing and testing both FIX session and application software. A comprehensive guide to writing secure software is outside the scope of this document. However, all FIX software, including application software, should be designed, developed, and tested so that accidental or malicious malformed data do not cause undesired operation or security vulnerabilities. These can include, but are not limited to:

- Buffer overflows, such as copying data into buffers without checking that the buffers are adequately sized, including a terminating NULL character if necessary
- FIX formatting errors, such as improper handling of binary data fields in FIX (which may contain 0 or ASCII <SOH>) and tags without values
- Data type errors, such as attempting to parse a text or binary string inappropriately placed in an integer or floating point field
- Data range errors, such as passing 0 or a negative number into a field that represents share quantities, or passing invalid enumeration values
- Security errors, such as accepting messages sent prior to a valid Logon, or failing to detect if the counterparty’s SenderCompID changes during the course of the FIX session

Where can I find more information on FIX and security?

FPL has security related documentation available on the FIX website.

<http://www.fixprotocol.org/specifications/TechDoc-InfoSecurity>

Links to technical terms used in this document

Provided below are links to terms introduced in this document. These links are for reference purposes only.

Term	Abbreviation	Definition
Buffer overflow		http://en.wikipedia.org/wiki/Buffer_overflow
Certificate Authority	CA	http://en.wikipedia.org/wiki/Certificate_authority
Certificate Signing Request	CSR	http://en.wikipedia.org/wiki/Certificate_signing_request

Data Encryption Standard	DES	http://en.wikipedia.org/wiki/Data_Encryption_Standard
Denial of Service	DoS	http://en.wikipedia.org/wiki/Denial_of_service
Digital Signature		http://en.wikipedia.org/wiki/Digital_signature
EncryptMethod (tag 98)		Field used in the FIX logon message (http://www.fixprotocol.org/specifications/fix5.0fiximate/Msg11.htm) to specify the type of encryption used to encode message content. The following types of encryption have been used with the FIX protocol. 1 - PKCS (Proprietary) 2 - DES (ECB Mode) 3 - PKCS / DES (Proprietary) 4 - PGP / DES (Defunct) 5 - PGP / DES-MD5 (See app note on FIX web site) 6 - PEM / DES-MD5 (see app note on FIX web site) http://www.fixprotocol.org/specifications/fix5.0fiximate/Field98.htm
Firewall		http://www.howstuffworks.com/firewall.htm http://en.wikipedia.org/wiki/Firewall_(networking)
Internetworking Protocol	IP	http://en.wikipedia.org/wiki/Internet_Protocol
IP security	IPsec	http://en.wikipedia.org/wiki/Ipsec
Message-Digest algorithm 5	MD5	http://en.wikipedia.org/wiki/MD5
OpenSSL		http://en.wikipedia.org/wiki/OpenSSL
PGP/DES-MD5		In 1997 the PGP/DES-MD5 encryption approach was designed by Morgan Stanley [EncryptMethod (tag 98)=5]. PGP/DES-MD5 is not an industry standard, rather it is a customized technique combining three data security standards with PGP. PGP is used to strongly encrypt the DES key for that session in logon, the standard DES is used to encrypt messages, and the MD5 hash function is used as a secure checksum. Using this algorithm, portions of the FIX message were encrypted and sent within the SecureData (tag 91) field.
Pretty Good Privacy	PGP	http://en.wikipedia.org/wiki/Pretty_Good_Privacy
Secure Sockets Layer	SSL	http://en.wikipedia.org/wiki/Secure_Sockets_Layer http://www.networkworld.com/details/473.html
STunnel		http://en.wikipedia.org/wiki/Stunnel
TCP Port		http://en.wikipedia.org/wiki/TCP_and_UDP_port

Transmission Control Protocol	TCP	http://en.wikipedia.org/wiki/Transmission_Control_Protocol
Transport Layer Security	TLS	http://en.wikipedia.org/wiki/Transport_Layer_Security
Tunneling		http://en.wikipedia.org/wiki/Tunneling_protocol
Virtual Private Network	VPN	http://en.wikipedia.org/wiki/Virtual_Private_Network http://computer.howstuffworks.com/vpn.htm
X.509 certificate		http://en.wikipedia.org/wiki/X.509

Revision History

Revision	Date	Author/Editor	Revision Comments
1.8	February 5, 2008	Ryan Pierce, Lehman Brothers	Incorporated comment received during 2 week review period following January 22, 2008 Global Information Security Subcommittee meeting.
1.7	January 22, 2008	Ryan Pierce, Lehman Brothers	Captured comments from January 22, 2008 Global Information Security Subcommittee meeting.
1.6	January 8, 2008	Ryan Pierce, Lehman Brothers	Captured comments from January 8, 2008 Global Information Security Subcommittee meeting.
1.5	December 11, 2007	Ryan Pierce, Lehman Brothers	Captured comments from December 11, 2007 Global Information Security Subcommittee meeting, including upcoming FIX 5.0 SP1 features.
1.4	November 27, 2007	Ryan Pierce, Lehman Brothers	Captured comments from November 27, 2007 Global Information Security Subcommittee meeting.
1.3	November 13, 2007	Ryan Pierce, Lehman Brothers	Captured comments from November 13, 2007 Global Information Security Subcommittee meeting.
1.2	October 2, 2007	Ryan Pierce, Lehman Brothers	Captured comments from October 2, 2007 Global Information Security Subcommittee meeting.
1.1	October 1, 2007	Eugene Rabkin, FPL Program Office, editor Ryan Pierce, Lehman Brothers	Cleanup and language changes by FPL Program Office. Includes language changes from the September 18, 2007 meeting of the Global Information Security Subcommittee. Includes additional information regarding

			possible benefits of the Internet, manual override of security policy, and X.509 certificates from the September 4, 2007 meeting of the Global Information Security Subcommittee.
1.0	August 20, 2007		Initial draft for public comment.

Contributors

The following people assisted in the preparation of this white paper.

Scott Atwell, American Century Investments
Jennifer Bayuk, Bear Stearns and Co. Inc.
Clive Browning, Rapid Addition, Ltd.
Robert Craddock, E*TRADE Capital Markets
Chia-Ling Lee, UBS Investment Bank
Sean Leviser, Reuters
Lloyd Hession, BT Radianz
Kevin Houstoun, HSBC
Jim Northey, The LaSalle Technology Group, LLC
Ryan Pierce, Lehman Brothers
Eugene Rabkin, FPL Program Office
Lisa Taikitsadaporn, Brook Path Partners, Inc.