COMP-8920-01 Fund. Of Information Assurance

Final Project Report

FIX Protocol – Exploitation & Mitigation

University of Windsor

Index

Introduction 3
Exploiting FIX 4
Authentication 4
Session Resumption 4
Session Ticket Resumption (FIXS) 4
Connectivity via leased line 5
Using FIXS with older TLS versions 6
Using FIX without TLS 6
Using FIXS with Simple TLS 7
Firewalking 7
Mitigation techniques 8
Vulnerability Specific techniques: 8
1. Session Ticket Resumption 8
2. Firewalking 8
3. TLS specific vulnerabilities 8
Security Perimeter Factors: 8
1. Area of Attack or Attack Vector 8
2. Authentication & Verification 8
3. Encryption 9
4. Integrity 9
5. Layered Architecture for FIX 9
6. Incident Response Plan 9
Bibliography 9
Team 10
Roles & Responsibilities 10

Introduction

In 1992, a group of institutions initiated an effort, FIX - **F**inancial **I**nformation e**X**change, to streamline their trading processes. Their belief that a standard for electronic communication of indications, orders & executions could increase efficiency and benefit the whole financial industry led to creation of an open message standard – FIX. The standard was purposefully controlled by no single entity and was structured keeping in mind differential requirements of each firm.
It has become a common language for global financial markets and has been accepted as a way of trading, given its progress towards becoming an essential part of achieving transparency, efficiency, and minimal cost for trades. FIX was initially devised for pre-trade usage but has over the last decade found its place in post-trade space as well.
FIX provided institutions, brokers, and other market participants a way out from clutter of unnecessary phone calls and scraps of paper. It provided foundation for high quality information sharing and straight through processing. From technologist's perspective, FIX provided an open standard benefiting both development and production efforts for financial industry. Efficient creation of connections with wide range of counter parties was an added perk. As for vendors, FIX has remained vendor neutral while encouraging its use and has used openness as a key to success. No over standardization and no demand for a specific carrier or protocol, left the technical decisions in hands of individual firms and thereby offered flexibility.

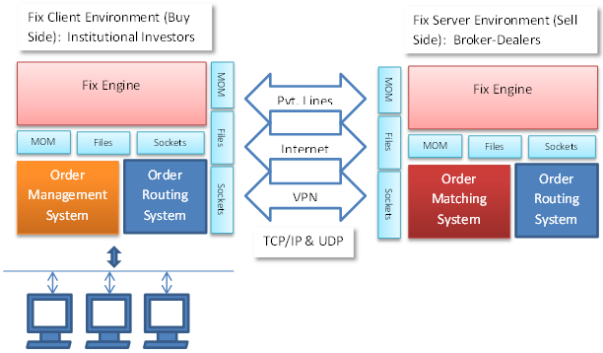

Figure 1-Basic FIX enabled architecture

While FIX brings multiple benefits, its open interface leaves it prone to threats due to multiple vulnerable points. In an attempt to offer flexibility, FIX left choice of security measures open to community. Working in financial sector, especially on FIX based architectures, a lot of instances triggered a curiosity to read and explore the security loopholes, possible backdoors and point of entries for cyberattacks. The NASDAQ hack in October 2010, raised multiple questions regarding the security measures employed in financial sector further raising the surprisingly not often asked question – **whether the globally followed standard – FIX, is as secure as it is transparent; or not**.

The last paper that focused on FIX's security issues was published in 2012 i.e., way before the release of TLS 1.3 and FIXS (an initiative towards cybersecurity from FIX community). This paper focuses on potential threats to FIX based architectures, engines, and financial firms. While primary objective is to list and summarize the vulnerabilities that might be an active entry point for attackers, the paper does not elaborate the methodology for an attack to be performed.

Goal of the research is to highlight vulnerabilities, possible mitigation techniques and security mechanisms that shall be employed to improve the layers of security for a FIX based architecture.

Exploiting FIX

David Goldsmith, CEO of Matasano Security, quoted the following in an interview regarding FIX protocol's security concerns:

"For the most part, when you look under the hood of these protocols, we find almost no means of security"

Most apps that use FIX are written in C and C++, *"which is not always super well-audited code."*

An attack on FIX could be silent and deadly: *"If a hacker was monitoring or viewing [the transactions], you may never know they are there, [He] could take that information and use it to their advantage for insider trading... or to cause significant financial damage."*

Authentication

FIX connections rely primarily and sometimes solely on cleartext FIX tags 49 and 56: SenderCompID and TargetCompID. If the two ids make a valid pair, the session is considered authenticated. For the same reason, FIX protocol is notoriously weak in authentication.

Session Resumption

FIX protocol has a dependency on sequence numbers to maintain the session synchronization and ensure that messages are not lost or missed by client or server.

When an active FIX session disconnects due to heartbeat miss or a test message awaited state of either client or server, session resumption is expected using the last active sequence number shared by both ends.

Resuming an encrypted session through a session ID means that the server keeps track of recent negotiated sessions using unique session IDs. This is done so that when a client reconnects to a server with a session ID, the server can quickly look up the session keys and resume the encrypted communication.

Attack1 - Rootkit
Numerous available rootkits can be used to spoof the output of the netstat command, therefore making both client and server vulnerable to believing themselves connected/disconnected while the state can be only known to the rootkit. Rootkit can make a client believe itself to be disconnected, triggering a session resumption from client end making it vulnerable to attacker's reach.

Session Ticket Resumption (FIXS)

Session resumption needed servers to take responsibility for remembering negotiated sessions for prolonged periods which was a critical limitation. For servers which handled multiple concurrent connections per second and had significant amount of load faced scalability issues. The same was applicable for servers that stored sessions information in cache for long periods. Session Ticket Resumption offered a reliable solution to this limitation.

A session ticket is created by encrypting a tuple of session key and the related information using a secret key known only to the server. At the end of a TLS handshake, server sends this ticket to client. Caching is

performed for ticket and the associated session key information by the client, given session tickets are supported.

When session resumption is needed, client sends a handshake message including the session ticket. On receiving a session ticket, server assumes that client wishes to resume an earlier session and therefore recovers the session key by decrypting the ticket and thereby resuming the session.
The session key shared via handshake is the single point of failure for TLS protocol applied over FIX. If an attacker gets access of the key, the information exposed makes all the session tickets vulnerable. This would exploit the "perfect forward secrecy" as proposed by TLS 1.3.

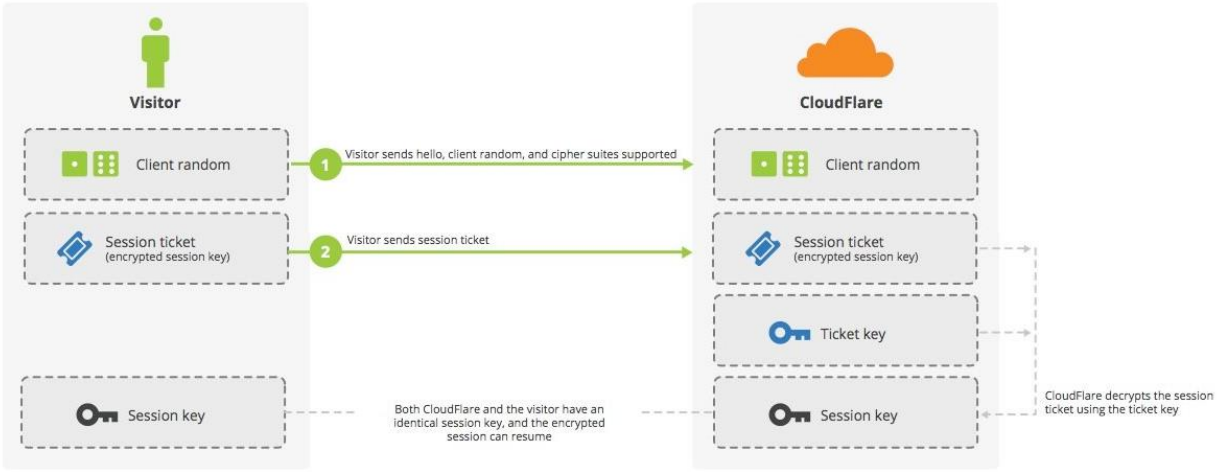

Figure 2 - Session Ticket Resumption Flow

Attack1 – Rootkit
Same as in case of session resumption, it can be applied here too.

Attack2 – MITM
If an attacker gets hold of the key, the same can be used to make a connection with server impersonating the actual client. FIX is used for sensitive financial trading information exchange and a single spoofed message can cause a major loss to a client or even a critical stock market crash.

Attack3 – DOS
An extension to MITM can be to send bulk of messages to server making it crash or become unresponsive to legitimate requests from other clients or connections.

Connectivity via leased line

Despite the safety offered by leased lines, there are still number of risks to be considered. Example: Intentional line cutting by a hacker or Physical wiretapping i.e., connecting to a company's dedicated line. This year in February'21, NSE – World's top derivative exchange, faced a lease line failure from 2 telecom lines at the same time. While the exchange clearly stated it to be a glitch at operator's end, the case holds the potential to be planned attack considering failure of two telecoms at the very same time.

Another open topic for this 4-hour long trading halt was the missed DR site switch at NSE end, this further suggests a cyberattack that jammed NSE from switching to disaster recovery. Information of the same can be found here & here.

Using FIXS with older TLS versions

Since TLS 1.3 is new and requires multiple integration changes, migration is a challenge most of the financial organizations will prefer not to take a stand.

TLS 1.3 made "forward secrecy" a necessity, thereby rendering useless the solutions where an Intrusion Detection System (IDS) monitors all network traffic by applying TLS decryption (using the private key). For IDS to work in the same manner, critical changes in architecture will be required.

Debugging or troubleshooting in production environments has been a lot dependent on captured traffic. Since TLS 1.3 does not allow decryption, the last resort or blunt solution to multiple performance or operational queries stands down as well.

Usage of stronger cryptography demands revision to both hardware and software requirements especially for embedded systems such as point-of-sale (POS) terminals and large-scale systems already close to their limits, from a system performance and load perspective. Processing power, internal memory or secure key storage that suffice for today's cryptographic requirements, may become too limited to accommodate and support future updates.

For all the above-mentioned reasons, the easier alternative being followed is to continue with TLS 1.2 instead. TLS 1.3 requires both client and server to agree on the same version. If either server or client backs out, they are turned back to the last agreed version hence making application prone to multiple attacks.

To name a few: POODLE, LOGJAM, FREAK, LUCKY13, LUCKYminus20, Sweet32, SLOTH, DROWN, etc.

Using FIX without TLS

Multiple stock exchanges and FIX engine vendors still opt for FIX as the chosen protocol for their connection. There are multiple variations and flavors of FIX available and proposed by FIX Trading Community that do not use or employ TLS.

Without TLS, FIX is prone to multiple attacks and has numerous vulnerabilities to explore. From adding a virus or worm to advanced attacks like DDOS or MITM, FIX protocol is vulnerable at every point since it works on basic characters being sent over network. The authentication mechanism for a FIX session involves simply validating session id which can easily be spoofed.

To name a very basic exploitation, Packet-Sniffing and Spoofing is quite an easy task considering no encryption is applied whatsoever.

Wireshark and Tcpdump can be used to sniff packets easily.

Another major issue here is that once a user has all information from the message, impersonation can easily be performed because for FIX sessions SenderCompID and TargetCompID stay consistent for a very long time.

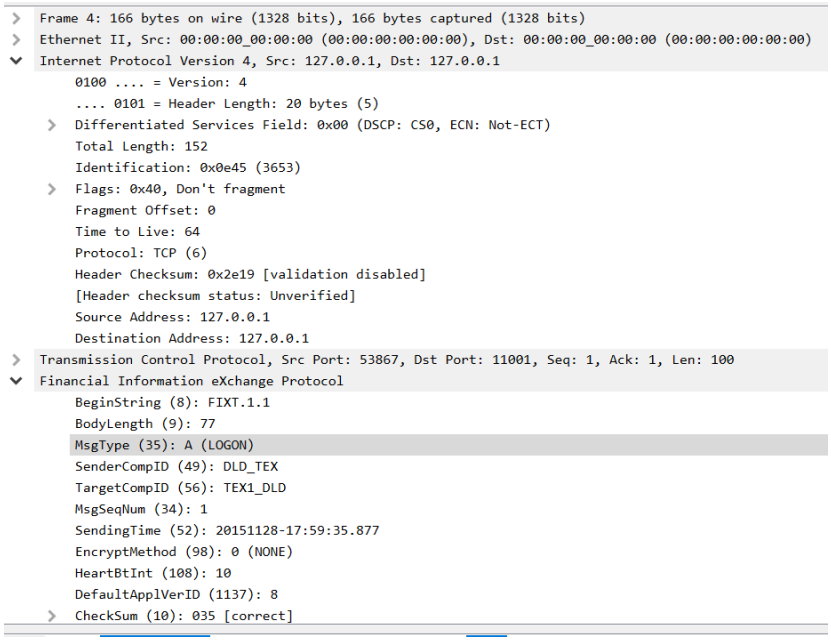

Figure 3 - Wireshark snapshot displaying details of a FIX packet captured in a pcap.

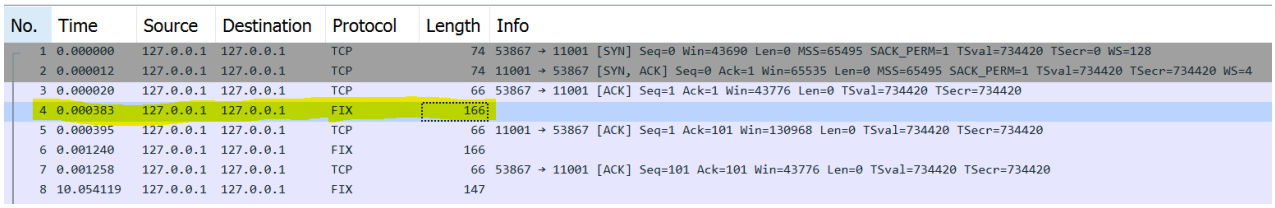

Figure 4 - Wireshark snapshot for a FIX packet recorded in pcap.

Using FIXS with Simple TLS

Simple TLS in terms of FIXS means only server authentication will be performed using certificates. This again leaves the client end vulnerable to penetration and attacks.

Firewalking

Firewalking disguises port scans and is equivalent of tracerouting. It sends TCP or UDP packets configured with TTL set at one hop next than the victim firewall to perform probing. The packet is forwarded to the next hop with a TTL of 0, when it bypasses the gateway. The hop is expected to log “exceeded in transit” message and drop the packet. Successive probe packets can help determine access information configured on a firewall.

Firewalk is a tool that can be used to find a firewall’s vulnerabilities and to track & trace network’s router hops behind a firewall.

Once breached, server is prone to multiple cyberattacks.

Mitigation techniques

As per Kevin Mitnick, an American computer security consultant, author, and convicted hacker, "Companies spend millions of dollars on firewalls, encryption and secure access devices, and it's money wasted; none of these measures address the weakest link in the security chain."

While 40% of attackers are script kiddies, there is a decent range of cyber criminals that don't focus on walls. They target and hammer the weakest entry points. For financial trading sector, network is the most vulnerable point of entry and needs a complete layered security mechanism.

For mitigating risk in financial trading sector, vulnerability specific techniques along with a security perimeter shall be employed.

Vulnerability Specific techniques:

1. Session Ticket Resumption

To avoid the risk of a threat agent getting hold of the ticket, session tickets shall be randomly generated, frequently rotated, and distributed to servers without touching the persistent storage.

2. Firewalking

Firewalking can be prevented by blocking all the outgoing TTL Exceeded Transit packets from leaving the network. Another defense technique to prevent Firewalking is by using Network Address Translation (NAT) or any proxy server to hide the address in the internal network.

3. TLS specific vulnerabilities

Release of TLS 1.3 has helped close numerous vulnerabilities in protocol's older versions. While the protocol added restrictions that require major changes in infrastructure, its addition over time can save organizations from cyberattacks that can easily penetrate and the trading systems and result in major economy crashes if the financial information gets spoofed.

Security Perimeter Factors:

1. Area of Attack or Attack Vector

Area of attack shall be minimized by exposing only necessary ports to registered networks by employing strict firewalls or routing table rules and implicitly distrusting any networks not originally configured. Restricting exposure to trusted extranets only is of utmost importance from security point of view.

2. Authentication & Verification

Multiple authentication factors shall be employed. A combination of FIX tags – (SenderCompID, TargetCompID, MiFid, Account, Source IP/port), can act as a strong authentication factor when not employing FIXS.

A FIX tag can easily be guessed and spoofed while a certificate is comparatively difficult to spoof and can help preventing MITM attacks. FIX secured with TLS can help verify the application. In case of FIXS (FIX-over-TLS), instead of using single TLS, mutual TLS with leaf certificate pinning shall be employed to ensure a better security layer for FIX connection.

3. Encryption

Most of the FIX traffic is clear text even though secured by a VPN or leased line perimeter which only provide protection of data over that network but not in the whole path. For the same reason, protocol encryption shall be leveraged most of the times. There are multiple FIX engines being released every now and then with TLS support and shall therefore be opted for.

In case a TLS supporting FIX engine is not an option, tools such as Stunnel can be used to add TLS encryption for existing applications. One of the well-known brokers – Trading Technologies, released their FIX specs with Stunnel as a mandatory requirement further increasing the security perimeter.

4. Integrity

All network hops and points of entry shall be monitored to ensure data integrity. Loosing a component to an attacker is potentially equivalent to entire environment becoming a target. It is important to recognize when a system is under attack. Intrusion detection systems shall be deployed and constantly monitored to ensure recognition of an attack at an early stage. A compromised system might still have a chance of making out alive, given the status of system is known. Proactive monitoring can help avoid critical system breakdowns.

5. Layered Architecture for FIX

Instead of having direct connections between client and server, secured FIX-engines shall act as middleman to read the counterparty FIX messages and send them to the main trading engine/environment. This would ensure that no malicious data coming from the direct connections enters the main infrastructure without being filtered or verified by FIX engine.

6. Incident Response Plan

Disaster recovery mechanisms shall be planned beforehand and employed when required. While mitigating a risk is crucial, ensuring that recovery mechanism is also thought of will help recover faster and reduce overall impact.

Bibliography

https://www.fixtrading.org/implementation-guide/

https://www.fixsim.com/sample-fix-messages

https://www.sans.org/reading-room/whitepapers/threats/exploiting-financial-information-exchange-fix-protocol-33964

https://leased-line-comparison.co.uk/leased-lines-secure/

https://www.thesslstore.com/blog/tls-1-3-banking-industry-working-undermine-encryption/

https://wiki.wireshark.org/SampleCaptures#Financial_Information_eXchange_.28FIX.29

https://ims.ul.com/tls-13-impact-financial-industry-part-1

https://ims.ul.com/ul-security-blog/tls-13-relevance-financial-industry-and-actions-take-part-2

https://eprint.iacr.org/2019/228.pdf

https://www.teldat.com/blog/en/dtls-security-udp-tls-heartbleed/

https://www.venafi.com/blog/importance-forward-secrecy-tls-13

https://www.venafi.com/blog/tls-13-visibility-extension-proposal-banks-want-more-visibility-so-do-cyber-criminals

https://www.cs.bham.ac.uk/~garciaf/publications/spinner.pdf

https://www.hso.co.uk/leased-lines/leased-line/vpn-vs-leased-line-comparing-the-security

https://www.ftfnews.com/cybersecurity-comes-to-the-fix-protocol/20024

https://blog.gigamon.com/2019/07/25/tls-1-3-its-benefits-are-real-but-so-are-the-drawbacks/

https://www.schneier.com/blog/archives/2017/12/security_vulner_10.html

https://wiki.crashtest-security.com/harden-tls-session-resumption

https://mailarchive.ietf.org/arch/msg/tls/KQIyNhPk8K6jOoe2ScdPZ8E08RE/

https://blog.compass-security.com/2017/06/about-tls-perfect-forward-secrecy-and-session-resumption/

https://thefinanser.com/2007/08/claims-that-fix.html/

https://www.fixtrading.org/fix-releases-cybersecurity-guidelines-waters-tech-article/

http://fixulate.blogspot.com/p/vulnerabilities.html

https://www.fnlondon.com/articles/fix-protocol-trading-community-frets-over-cybersecurity-20150511

https://www.darkreading.com/risk/hacking-capitalism/d/d-id/1128865

https://blog.cloudflare.com/tls-session-resumption-full-speed-and-secure/

https://www.cloudinsidr.com/content/known-attack-vectors-against-tls-implementation-vulnerabilities/#:~:text=TLS%20vulnerabilities%20are%20a%20dime,connection%20renegotiation%2C%20and%20session%20resumption.

https://www.fixtrading.org/standards/fixs/

Team

- Srishti Jain (Student ID: 110026562)

- Siddharth Paliwal (Student ID: 110036256)

Roles & Responsibilities

1. Researching TLS vulnerabilities - Srishti

2. Researching FIX-Protocol specific vulnerabilities - Srishti & Siddharth

3. Researching Exploitation schemes & tools - Siddharth

4. Researching Mitigation techniques for discovered vulnerabilities - Srishti & Siddharth

Note : Both will be working on same task occasionally and by purpose to ensure an important vulnerability or mitigation solution is not skipped.