WHITE PAPER

ENSURING SECURITY WITH OPEN APIs

Scott Biesterveld, Lead Solution Architect

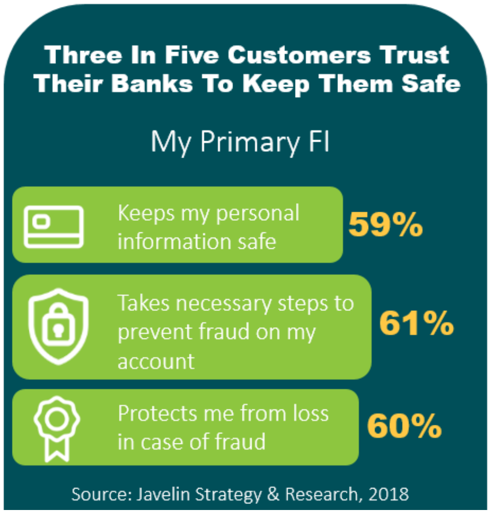
Senthil Senthil, Development Manager

IBS Open APIs

The security features that banks must build into their financial solutions become even more critical with open Application Programming Interfaces (APIs). As financial service firms embrace new open banking technologies, they must focus on security to maintain their trusted provider status with customers. This paper addresses the state of security in banking and then spotlights security procedures and critical functions banks should expect from their API partners.

Industry perceptions on security and secure APIs

As financial services organizations continue to embrace APIs and open banking, they will expand partnerships with third-party technology providers at a rapid pace. The nature of open banking technology optimizes third-party integration to create greater accessibility to various data and features.



Source: Javelin Strategy & Research, 2018

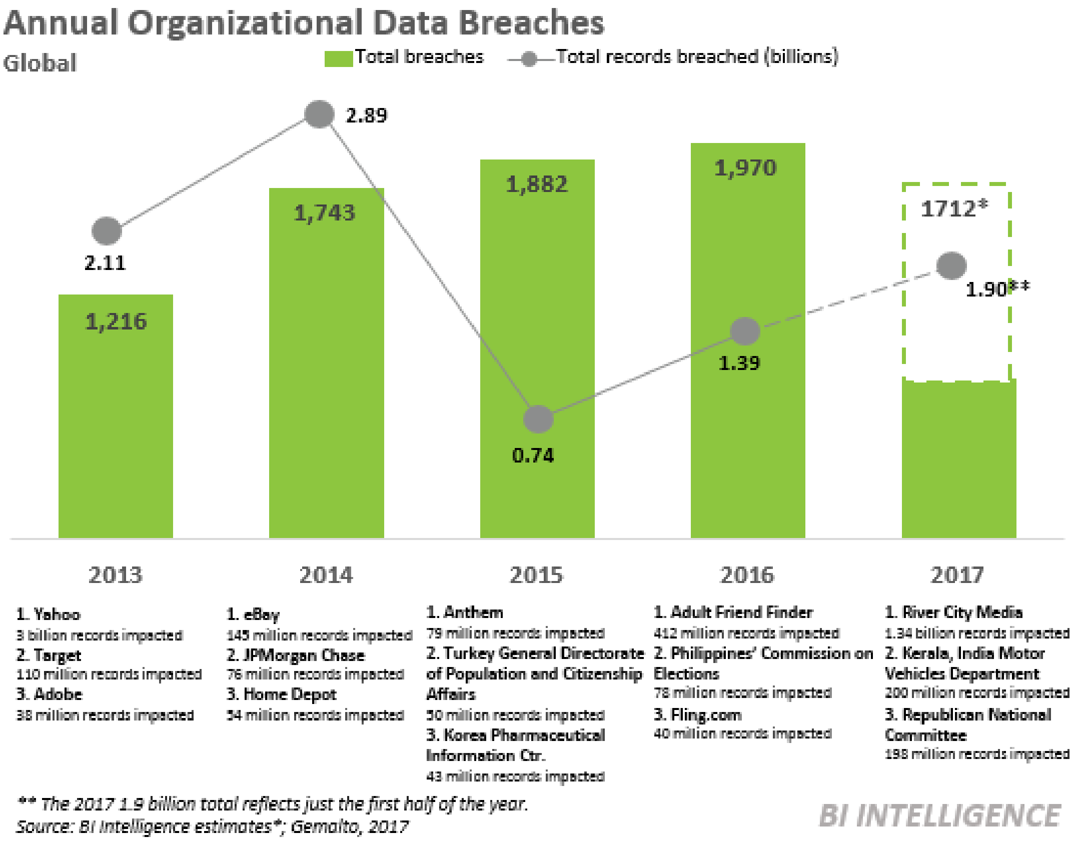
Customers trust their banks

Although a majority of customers trust their banks, at the same time, data breaches, fraud, and security attacks are expanding at their own rapid pace. Technology innovation with APIs must be carefully aligned with security processes and procedures for banks to continue their role as trusted advisor to their customers. According to recent research from Javelin, three in five customers expect their banks to protect their data.[1]

Security breaches a growing problem

And the threats against those customer expectations are very real. Data breaches are industry-agnostic, costly, and here to stay. In 2017, the average data breach cost $3.6 million globally and $7.4 million in the U.S., according to IBM. Those figures include direct costs, such as those tied to identification, containment, and resolution, as well as indirect costs, including customer losses and brand damage, which can be vast.[2] (source Business Intelligence)

[1] Javelin, 2018 Fraud and Security Trends, February 2018
[2] Business Intelligence, Data Breaches Report, February 2018



Annual Organizational Data Breaches

Global

*** The 2017 1.9 billion total reflects just the first half of the year.*
Source: BI Intelligence estimates; Gemalto, 2017*

APIs create opportunities but also can present security challenges

APIs are exploding in popularity because they build on well-understood techniques and leverage some existing infrastructure. However, it is a mistake to think providers can secure APIs using the same methods and technology previously used to secure the conventional, browser-centric Web. While it's true APIs share many of the same threats that plague the Web, they are fundamentally different and have unique risk profiles to manage.

The challenges become more acute as banks look at APIs as a competitive tool to collaborate with their corporate clients in business-to-business (B2B) transactions. Corporate customers are more technically focused, agile, and demanding than in the past. In contrast, banks have less money, less time, and a greater focus on security, which hinders their ability to concentrate on, and deliver, what's needed in today's competitive market.[3]

Need for security from API technology partner

The key to addressing API security is to collaborate with a strong, secure partner—one that understands the regulatory environment, technology, and procedures to make security job one. As banks evaluate API technology partners, it is crucial to understand the security attributes an API partner can and should provide today's financial institutions.

[3] Aite, Corporate Banking API Strategies, May 2017

Roles of an API gateway and API marketplace in providing security

An API gateway and API marketplace provide developers with tremendous flexibility for creating new solutions, while extending security to innovate with confidence.

An API marketplace makes it easier for organizations to find and discover APIs because it is presented in a user-friendly fashion. A developer is no longer limited to simple HTML or PDF documentation offered outside the marketplace, as a developer should have everything they need inside the marketplace. This reduces steps that must be taken and promotes rapid development. An effective API marketplace organizes and categorizes APIs by providing a logical method in which developers can easily find APIs. It also provides the ability to search for an API. One of the most useful marketplace benefits is the ability to try out an API in a “sandbox” environment, allowing a developer to immediately assess if an API meets their organization’s needs, with no risk to the production environment.

API marketplace as an insulator

Banking integrations have traditionally occurred over a private secure network. With an API marketplace, APIs can be insulated, so that only registered users are able to discover APIs. Thus, an API gateway can provide the additional security necessary to expose APIs to the Internet—increasing accessibility and uses for APIs. An API gateway and marketplace working together provide third-party vendors and FinTechs with secure access to a bank’s data in ways that have not been previously available.

Security capabilities an API Gateway should provide

Most current version of communication security (HTTPS/TLS 1.2)

This is the cryptographic protocol used to transport the data over the Internet. Older versions of TLS and versions of the previous protocol SSL are no longer secure enough to be used.

Reverse proxy URL

A reverse proxy URL can provide load balancing and fail over, creating a single point of access to a bank’s APIs and minimizing the risk of potential bad actors consuming the APIs or creating outages such as DDoS attacks. The reverse proxy also provides an additional level of abstraction and control to ensure the smooth flow of network traffic between the clients and the servers.

1. Most current version of communication security (HTTPS/TLS 1.2)
2. A reverse proxy URL
3. Client certificate
4. Advanced authentication models such as OAuth 2.0
5. An authentication/role model
6. Access to a limited set of customers/accounts
7. Encryption above and beyond what is provided by HTTPS/TLS
8. Secure concepts regarding logging

Client certificate

A client certificate makes it improbable for someone to submit requests to the API gateway without having the client certificate. This is accomplished by enforcing the standard wherein all requests must be sent with the client certificate.

Advanced authentication models such as OAuth 2.0

Advanced authentication models such as OAuth 2.0 should also be used. Historically, APIs have used Basic Auth, which required the user to provide his/her user ID/password for each request. The user ID/password is not encrypted or hashed. This created a risk if that user ID/password was compromised; however, that risk was mitigated by requiring transport of the activity over a private line. OAuth 2.0 offers a better solution because it incorporates an access token that expires.

Authentication/role model

An authentication/role model, that can limit the APIs to which an application has access, minimizes the risk that a bad actor can get access across-the-board.

Access to a limited set of customers/accounts

Access to a limited set of customers/accounts should be provided. Typically, an API accesses any or all customers/accounts, but in special cases—where a subset of accounts might be required—limiting access to a specific account subset can significantly reduce the impact of a security breach. This is especially the case when a client is a third-party application fulfilling a specific use case.

Encryption above and beyond what is provided by HTTPS/TLS

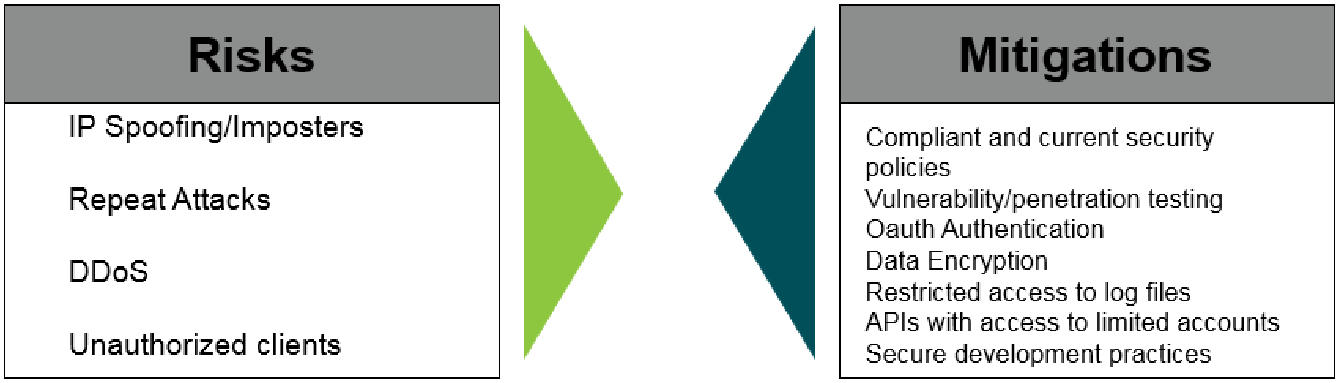
Encryption above and beyond what is provided by HTTPS/TLS should also be incorporated to provide additional security. This requires that the client and the server mutually agree on the cryptographic algorithm and overall approach. For banking APIs, this is beneficial because you only want your intended and vetted clients to be aware of your encryption approach.

Secure concepts regarding logging

REST APIs include customer numbers, account numbers, and other sensitive data within Uniform Resource Identifiers (URIs). Most banks log the URI before invoking the API. Additionally, a partner should log the URI when the request is received. A bad actor could browse the log and begin to put together a story about the activity within the log. Restricting access to logs and encrypting sensitive data within the URI will help minimize risks.

Bank expectations of an API partner

Most banks either cannot afford to put their own API gateway into place or don't have the resources to support one. Therefore, many banks will look for API technology partners to provide those API solutions. Rightfully, banks should have concerns about exposing APIs and subsequently sensitive data to the Internet as well as have an expectation of the API partner's ability to reduce the risk of fraudulent attacks. An API partner should provide robust API solutions that provide a bank with the confidence that risk is minimized and ongoing surveillance is used to mitigate new risks as they arise.



Risks

IP Spoofing/Imposters

Repeat Attacks

DDoS

Unauthorized clients

Mitigations

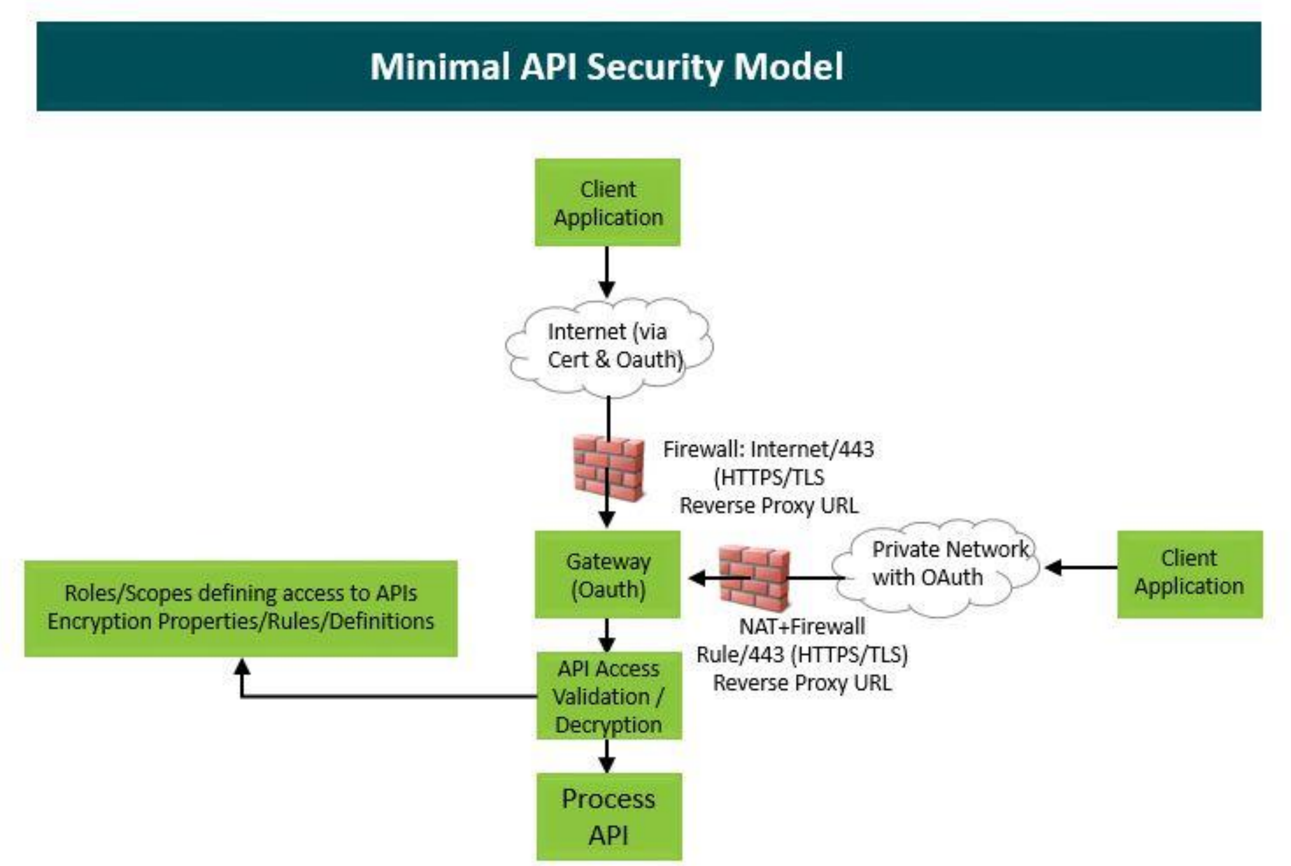
Compliant and current security policies
Vulnerability/penetration testing
Oauth Authentication
Data Encryption
Restricted access to log files
APIs with access to limited accounts
Secure development practices

Points to consider, questions to ask when evaluating API security

- Are current secure connectivity protocols being used and enforced?
- Are secure connectivity protocols reviewed regularly and updated accordingly?
- Is the API gateway regularly scanned for vulnerability and penetration?
- Are advanced authentication methods used?
- Is access to APIs and sensitive data granted only to authorized users?
- Does the authentication model allow the ability to restrict the user to subsets of the APIs?
- Are APIs available that provide access to a limited set or range of accounts?
- Is there an easy and quick way to revoke access if unauthorized access is discovered?
- What is the duration of the OAuth access token being used and can the bank declare the desired duration?
- Is encryption used for additional protection of sensitive data?
- Are log files securely protected?
- Is access to log files restricted?
- Are passwords and other sensitive data stored in log files?
- Does the API Gateway provide advanced features such as orchestration and single commit?

Bringing security components together

Ensuring security with an open technology like APIs is never an easy task. Banks need to partner with full-service technology firms that bring secure procedures, testing, security technology, and an API gateway and marketplace to support API development. A minimal API security model is shown below, highlighting how the main security components of a comprehensive API solution should fit together.


Minimal API Security Model
Client Application
Internet (via Cert & Oauth)
Firewall: Internet/443 (HTTPS/TLS Reverse Proxy URL
Gateway (Oauth)
Private Network with OAuth
Client Application
NAT+Firewall Rule/443 (HTTPS/TLS) Reverse Proxy URL
Roles/Scopes defining access to APIs Encryption Properties/Rules/Definitions
API Access Validation / Decryption
Process API

Contact Us

For additional information on IBS Open APIs, and the security supporting this type of financial solution, contact Amit.Aggarwal@fisglobal.com or call 414.815. 1182.