HOW TO PREVENT A ROBBERY IN ALBANIAN FINANCIAL INSTITUTIONS

Enida Puto
Bank of Albania
ALBANIA
enidaputo@yahoo.com

Kozeta Sevrani
Faculty of Economy - UT
ALBANIA
kozeta.sevrani@unitir.edu.al

ABSTRACT

Cyber-attacks are growing in number and attackers are focusing more deeply inside banks. Financial institutions execute some of the most important, critical and confidential transactions and are more at risk from cyber attacks. Recently, several robberies were made using SWIFT messaging interfaces, which has caused an alarm in the financial world, since SWIFT is the end point to financial transactions for most financial institutions around the world. In this paper we refer to Bangladesh Bank robbery as a case study, analyze what happened from the information available on the Web and come to conclusions about how this incident could have been prevented. Starting from this case, our scope for this paper is to explain how to prevent similar incidents in Albanian financial institutions. In order to do so, we use a list of security areas that usually take place in local infrastructures and compare against the local infrastructures in several Albanian institutions. We then list the vulnerabilities that we found and recommend how to improve them. Also, we found some strong security points. The security of local environments lies beyond SWIFT local infrastructure. The findings and recommendations of this paper can be applied in every system that needs high security.

Keywords: Local, infrastructure, security, institutions, SWIFT.

INTRODUCTION

The Bangladesh Bank robbery took place in February 2016, when instructions to fraudulently withdraw US$ 1 billion from the account of Bangladesh Bank, the central bank of Bangladesh, at the Federal Reserve Bank of New York were issued via the SWIFT network. Five transactions issued by security hackers, worth $101 million and withdrawn from a Bangladesh Bank account at the Federal Reserve Bank of New York, succeeded, with $20 million traced to Sri Lanka (since recovered) and $81 million to the Philippines (about $18 million recovered). The Federal Reserve Bank of New York blocked the remaining thirty transactions, amounting to $850 million, at the request of Bangladesh Bank. [1]

Soon after the reports of the theft from the Bangladesh central bank, a second, apparently related, attack was reported to have occurred on a commercial bank in Vietnam. Both attacks involved malware written to both issue unauthorized SWIFT messages and to conceal that the messages had been sent. [2]

These cases are not isolated. Russian cyber security firm Kaspersky Lab claims Interpol and other agencies estimate that more than $1 billion has been pilfered from 100 financial institutions during the last two years by the Carbanak cybergang located in Russia, Ukraine and China. [3]

A SWIFT transfer is a type of international money transfer sent via the SWIFT international payment network. The SWIFT organization provides a secure network that allows more than 10,000 financial institutions in 212 different countries to send and receive information about financial transactions to each other. The majority of SWIFT system members are banks, but it is also used by trading institutions, money brokers etc. The SWIFT network does not actually transfer funds, but instead it sends payment orders between institutions' accounts, using SWIFT codes. [4]

All these robberies point to a global problem in security of online payments. Financial institutions are, of course, more at risk from these kinds of attacks.

In this paper we have taken SWIFT and the Bangladesh Bank robbery as a case study, and then explain how to secure and prevent similar cases in Albanian financial institutions.
The findings and conclusions of this paper can be applied in every other system and institution which has a focus on security. Since Albanian Banks are connected via SWIFT Network, and also use other financial systems for processing payments and transactions, it is important that every one of them, and every other financial institution in Albania, understands its own role for maintaining a secure local infrastructure, its functions and the preventative measures that that should be taken in order to stay safe in this new world of continuous cyber threats.

While the attackers' sophistication is clearly on the rise, in all cases, they have relied on basic security weaknesses in the targeted customers' perimeter and internal network security. The determination, patience and cunning the attackers are demonstrating makes it more imperative than ever that customers rapidly deploy and maintain all basic cyber hygiene tools and measures. [5] This is an ongoing security challenge for all financial institutions.

METHODOLOGY

But how did the Bangladesh Bank robbery happen technically? Which line of defense was breached? Since several banks in different countries were included in the incident (Philippines, Sri Lanka and United States), it is still an ongoing investigation for a well-organized cyber crime, but it all started with the compromised local system of Bangladesh Bank. This incident was followed by several others, with attackers using the same methods.
SWIFT has pronounced that its network, software and services have not been compromised – each of incidents took place after a customer suffered security breaches within its locally managed infrastructure. [6]

According to SWIFT, attackers obtained valid credentials the banks use to conduct money transfers over SWIFT and then used those credentials to initiate money transactions as if they were legitimate bank employees. Other reports indicate that lax computer security practices at Bangladesh Bank were to blame: the bank reportedly didn't have firewalls installed on its networks, raising the possibility that hackers may have breached the network and found the credentials stored on the system. Attackers also installed malware on the bank's network to prevent workers from discovering the fraudulent transactions quickly. [7]

From this case we can find that attackers gained access to the system via network security breaches, installed malware and then obtained and used valid user credentials. Which means that this incident could be prevented if the following security controls were in place: installed and well configured firewalls, antimalware protection, logging and monitoring of hosts and applications and multi-factor authentication.

Usually, the main weakness in any network is its client endpoints. Instead of focusing on potential vulnerabilities in provider's core messaging systems and software, hackers are exploiting holes in banks' security measures to attack the banks' connections to the SWIFT network and gain access to the SWIFT and other messaging systems. As each institution is responsible for maintaining the security of its local infrastructure, mandatory levels of endpoint and user security should be a requirement before access is granted. [8]

We monitored several financial institutions in Albania in order to find security vulnerabilities for its most important systems, especially SWIFT where it applied. The security areas in institutions' local infrastructure that we focused during monitoring are:
Perimeter - physical security and conditions, securing the perimeter from unauthorized people;
People, policies and procedures - Access control/Authorization, Authentication control, Monitoring, Business Continuity and Disaster Recovery, Risk assessment;
Network – Secure zones, Firewalls, Routers, VPN's, Demilitarized zone, Data flow, Encrypted Protocols, Active Directory, Intrusion detection system (IDS);
Hosts – Servers and workstations. Operating Systems must be updated and patched. Updated antivirus software should also be available;
Applications – Flaws can be in the design, development, deployment, upgrade and maintenance. Security testing should be performed;
Databases – encrypted stored procedures, compromised credentials, regular backups, storing on tapes. [9]
The methodology used, which is comparison of the most important security areas against each financial institution, is represented in Figure 1.

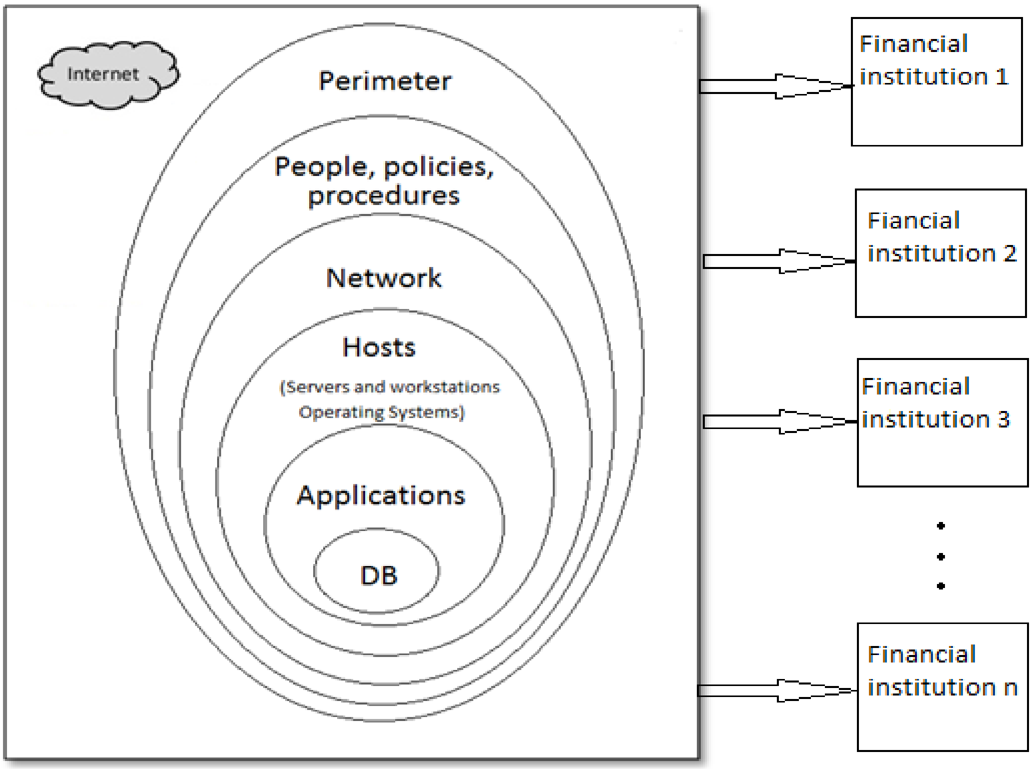

Figure 1 – Comparison of the most important security areas against each financial institution

Our findings and recommendations are discussed next.

RESULTS AND DISCUSSIONS

Below we have highlighted the areas that need the most improvement in order to maintain a safe local environment and we also describe how to improve these areas. We have listed the vulnerabilities from most critical to less.

A. The most critical vulnerability found is the lack of multi-factor authentication.
Multi-factor authentication is one of the most important security controls to implement.
How to improve: Multi-factor authentication must be implemented at least at one of the following: 1. Operator PC log-in, 2. Operator access to jump server or 3. Operator log-in process to the messaging interface.
Multi-factor authentication requires two or more of the following authentication factors:
Knowledge factor, for example, a password;
Possession factor, for example, USB tokens or one-time password generators using mobile phone;
Inherence factor, for example, fingerprint, retina scans or voice recognition.
We recommend: at least: two-factor authentication (password and one-time password generator using mobile phone) while logging-in messaging interface.

B. Other critical vulnerability found, almost as critical as multi-factor authentication is the lack of Security Updates.
Usually attackers use known security vulnerabilities, for which the vendor has released an update.
How to improve: All hardware and software (including servers, network devices, operator PCs, operating systems, messaging applications, antivirus and antimalware protection etc), should be upgraded with software updates, should be supported and within the supported product lifecycle window of the vendor and timelines should be established for applying patches.
We recommend: the use of CVSS (Common Vulnerability Scoring System) Version 3 Critical (9.0+ score): applied within 1 month of release and other scoring applied within 2 months of release for applying updates.

C. Another important vulnerability found is the lack of logical access control.
How to improve: Sometimes a person has more roles and responsibilities that should. Accounts are defined according to the documented security principles of:

> Need-to-know access: Access is given only for operators who have a continuing requirement to access needed information for their defined tasks. Access to other system functions is disabled.
>
> Least privilege: User and administrator privileges are controlled and accounts are granted only the privileges that are required for normal, routine operation. Additional privileges are only granted on a temporary basis.
>
> Segregation of duties: 4-Eyes principles are enforced and sensitive duties are separated which means that some roles cannot be represented by the same individual.
>
> Account Review: Privileges are revoked when an employee changes roles or leaves the organization.

We recommend: applying logical access controls described above and reviewing them at least once a year.

D. Next widespread vulnerability is data flow without secure mechanism.
How to improve: Financial institutions have several payment systems, usually end pointing to SWIFT. Confidentiality, integrity, and authentication mechanisms should be implemented to protect data flows from operator-to-application and application-to-application.
We recommend: the use of one-way TLS for operator-to-application data flow and two-way TLS, local authentication keys, cryptographic algorithm AES for application-to-application data flow.

E. Lack of network segregation leads to compromised security.
How to improve: In order to restrict access and connectivity to the system that should be secured, a secure zone must be implemented. All components in the secure zone should be protected equivalently, or may be implemented additional segregation between components of the secure zone. Secure zone should be protected by physical or virtual transport layer firewalls. No 'allow any' firewall rules are used and rules should be reviewed at least annually.
We recommend: that end users outside this secure zone connect with the system via a jump server. Also, internet access from systems within the secure zone is highly restricted or blocked.

F. Another vulnerability found is the lack of vulnerability scanning.
How to improve: Secure zone and operator PC systems (or jump servers) are scanned for vulnerabilities using a reputable scanning tool in order to exploit known security vulnerabilities.
We recommend: that this process should be regular, if not possible in real-time. A response plan to address the found vulnerabilities should also be in place.
G. Vulnerability: Perimeter – Not Enough Physical Security
How to improve: Physical access to the secure zone, where passwords and tapes are stored physically, sensitive equipment etc., should not only be restricted to the authorized personnel, but also be traceable via video surveillance and physical access logs. As for logical access, physical access should be revoked when an employee changes roles or leaves the organization.
We recommend: that physical access is reviewed at least once a year.

Strong Security Points
We found that institutions in scope, complied well with:
Software integrity: A software integrity check integrated into the messaging interface application is performed at startup and at regular intervals on SWIFT messaging interface. Software integrity checks provide a detective control for unexpected modification to software. This is a very useful utility that we recommend to be implemented in every application that needs high security.
Database integrity: A database integrity check integrated into the messaging interface application is performed at regular intervals on databases that record SWIFT transactions. Database integrity checks provide a detective control against unexpected modification to records within the database. This is a very useful utility that we recommend to be implemented in every database that needs high security. This utility could also be integrated into the databases products.
Anti-malware software: Anti-malware software from reputable vendors were installed in user PC’s and servers and were up-to-date.
Logging and monitoring: Logging and monitoring tools were in place and personnel frequently reviewed logs.

Business continuity procedures: We found that financial institutions had in place business continuity procedures according to the best known practices.

CONCLUSIONS

Like every financial institution around the world, Albanian financial institutions should be alerted about the growing number and criticality of cyber-attacks.

Since SWIFT is the world's leading provider of secure financial messaging services and the end point system for most financial institutions globally, the local infrastructures of clients hosting its systems should be particularly secured and the security approach to this system can be taken as an example for other systems.

All security points and controls that we have mentioned in ‘’Material and methods’’ part of this paper, should be implemented according to the best known practices.

In this paper we conclude in our findings (weakest and strongest security points) about in scope Albanian financial institutions.

The most important vulnerabilities that we found are: lack of multi-factor authentication, lack of security updates, lack of logical access control, data flow without secure mechanism, lack of network segregation, lack of vulnerability scanning and not enough physical security.

We recommend how to improve these vulnerabilities, but a lot of other information can be found according to best practices on these security points.

Some of the strongest security points that we found are: software integrity, database integrity, anti-malware software, logging and monitoring, business continuity procedures. As we have mentioned above, the findings and recommendations of this paper can be applied in every system that is focused on high information security.

REFERENCES

[1] https://en.wikipedia.org/wiki/Bangladesh_Bank_robbery
[2] https://en.wikipedia.org/wiki/2015–2016_SWIFT_banking_hack
[3] https://www.forbes.com/sites/jonmarkman/2016/06/03/world-banks-reel-from-digital-robberies/#1dd8662af933
[4] https://transferwise.com/us/blog/everything-you-need-to-know-about-swift-network.
[5] http://www.silicon.co.uk/security/microsoft-malwarebytes-intel-227747?inf_by=5a77937a671db8e3558b4acc
[6] https://www.swift.com/myswift/customer-security-programme-csp.
[7] https://www.wired.com/2016/05/insane-81m-bangladesh-bank-heist-heres-know/
[8] http://searchsecurity.techtarget.com/tip/SWIFT-network-communications-How-can-bank-security-be-improved.
[9] Puto, E,. Sevrani, K,. (2015) Database Security for Financial Institutions, HERTSPO2015