THREAT ANALYSIS

SWIFT Systems and the SWIFT Customer Security Program

F-Secure

CONTENTS

1. Introduction 3

2. What is SWIFT? 4

3. Attacks on SWIFT Systems 5

- 3.1 Sonali Bank - $250,000 6
- 3.2 Banco del Austroz - $12,000,000 7
- 3.3 Reports from the Philippines - $Unknown 7
- 3.4 Tien Phong Bank – $1,130,000 (Attempted) 7
- 3.5 The Bank of Bangladesh – $81,000,000 (of $951,000,000) 8
- 3.6 Unnamed Ukrainian Bank - $10,000,000 9
- 3.7 The Far Eastern International Bank - $160,000 (of $60,100,000) 9
- 3.8 The NIC Asia Bank - $580,000 (of $4,400,000) 9
- 3.9 Common Factors 10

4. What is SWIFT CSP? 10

5. Is SWIFT CSP Compliance Enough? 13

- 5.1 Summary 15

6. So, How Can You Better Secure Your Local SWIFT Systems? 16

- 6.1 Predict 17
- 6.2 Prevent 17
- 6.3 Detect 17
- 6.4 Respond 17

7. Conclusion 18

1. INTRODUCTION

The financial sector is and has always been a prime target for crime. In the modern day there is not only the risk of physical attacks, but also cyber attacks. Heist, espionage, and sabotage campaigns – once a threat which could be mitigated with the implementation of strong physical security controls and procedures – can now all be conducted by a wide range of threat actors from anywhere in the world.

Over the past five years there has been a steady increase in cyber attacks on banks and the finance sector as a whole, specifically with regards to the development and execution of advanced targeted attacks against financial messaging services, such as SWIFT. This comes as no surprise. Attackers have come to the realization that focusing their resources on performing a low profile, calculated, and sophisticated attack on a financial institution has the potential for a much higher gain and requires less overall effort than continuously targeting individual customers. As such, these attacks have increased over the years, not only in number, but also in sophistication, with attackers becoming increasingly persistent and adaptive[1] when it comes to bypassing security controls and compromising critical financial systems to achieve their end goals.

Although a number of these attacks appear to be criminal in nature (e.g. the Carbanak gang), some attacks have shown strong links to nation states such as the Lazarus group (reportedly linked to North Korea). This may be an indication that large-scale financial heists may be one of the few remaining methods of obtaining international currency within heavily sanctioned states.

As a countermeasure to the current cyber-threat landscape, SWIFT has implemented the Customer Security Programme (CSP)[2]. This programme requires all SWIFT customers to implement a number of controls defined by SWIFT's Customer Security Controls Framework (CSCF)[3], to which customers must self-attest compliance before 1 January 2018. This framework outlines a collection of security controls to ensure a minimal baseline for security is in place across all customers' local SWIFT deployments.

This document will review a collection of major SWIFT-related breaches that have occurred over the past five years, and analyse which common factors are shared between them. This will be followed by an analysis of the scope and reliability of SWIFT CSP, identifying its strengths as well as its limitations. Finally, MWR will provide recommendations on how to further secure these types of critical payment systems against future attacks.

2. WHAT IS SWIFT?

SWIFT (the Society for Worldwide Interbank Financial Telecommunications) is a secure messaging service used to transmit financial messages between member institutions around the world. SWIFT functions as a member-only cooperative service that is used and trusted by more than 11,000 financial institutions in more than 200 countries and territories around the world.[4]

At its core, SWIFT provides access to the SWIFT messaging network (SWIFTNet) and its four messaging services (FIN, InterAct, FileAct and Browse). However, it also provides the standard for financial messaging and a range of solutions for the security, creation, management, processing and validation of these messages.[5]

SWIFT does not, however, hold responsibility for the security of its customers' local SWIFT infrastructure, although it does provide assistance to ensure customers are able to manage cyber attacks. An example of this is the Customer Security Programme (CSP), which was originally introduced in late 2016.

3. ATTACKS ON SWIFT SYSTEMS

There have been at least eight high-profile attacks on SWIFT systems over the past five years (among many other lower-profile attacks), all resulting in significant financial loss.

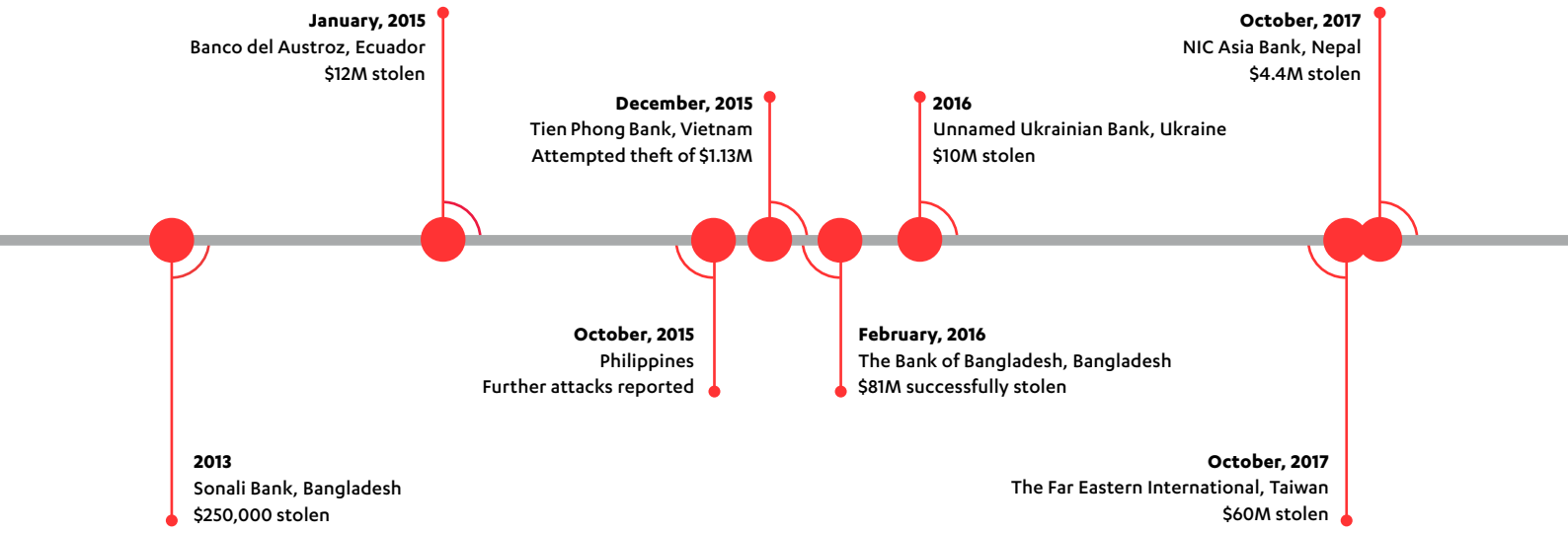


Figure 1 - Timeline: High-profile SWIFT-related attacks

Together, these high-profile attacks alone have resulted in the collective theft of around $167,210,000, with the cyber-attack on the Bank of Bangladesh in 2016 being one of the largest bank heists in history. During this attack, an attempt to steal $951,000,000 was carried out, with $81,000,000 being successfully exfiltrated to banks in the Philippines and then laundered via casinos to fully extract the stolen money from the banking system.

Analysis shows that attacks on SWIFT systems are frequently targeted at institutions with less mature security policies and procedures. However, there have been reports of both attempted and successful attacks at institutions of all sizes and levels of security maturity around the world.

A map outlining the geographical location of the attacks we will review in this report can be seen overleaf.

Figure 2 – Map: High profile SWIFT-related attacks

In the wake of the Bangladesh heist there was a shift in attention regarding attacks on SWIFT systems. Since then, a number of links between the tools and techniques used in these attacks and advanced persistent threat actors (APTs), such as the Lazarus group[6], have been speculated on and identified.[7]

The number of successful attacks against these systems shows that SWIFT customers must do more to protect their local infrastructure. However, to effectively defend any system from an attack, there is a pre-requisite to first understand how attackers are targeting these systems.

3.1 Sonali Bank - $250,000

Not much was known regarding the Sonali Bank heist (2013), and until 2016 it was treated as a 'cold case'. However, investigators re-opened the case after the attack on the Bank of Bangladesh in 2016.

It was reported that attackers were able to infect the bank's internal systems with key-logger software that was used to harvest user credentials. These credentials were then used to laterally move through the bank's network in order to gain access to the bank's internal SWIFT systems, where $250,000 worth of SWIFT transactions were made.

1. Attackers deploy a key-logger to steal employee credentials

2. Using stolen credentials, the attackers move laterally across the network to find SWIFT connected systems

3. The attackers then issue an unspecified number of SWIFT transactions

Figure 3 – Attack Path: Sonali Bank heist[8]

3.2 Banco del Austroz - $12,000,000

During the attack on Banco del Austroz (January 2015), attackers stole the credentials of an unnamed bank employee and used these credentials to access the employee's Outlook email account. Using this access, the attackers located cancelled and rejected SWIFT transfer requests, altered their details, and reissued them, resulting in $12,000,000 worth of legitimate transfer requests being sent.

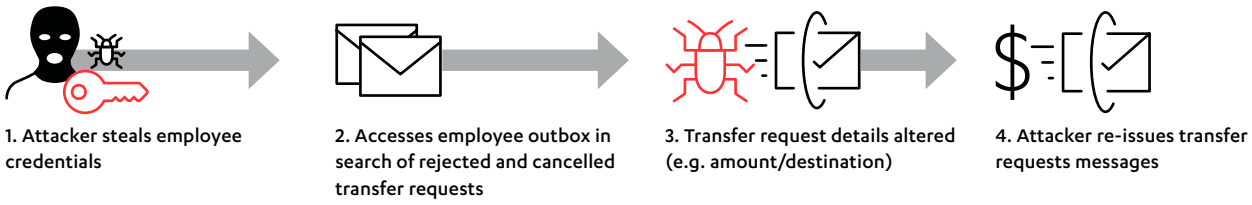


Figure 4 – Attack Path: Banco del Austroz[9]

3.3 Reports from the Philippines - $Unknown

In 2016, reports emerged that a bank in the Philippines had been the victim of an attack in October of 2015. Although this attack occurred two months prior to the failed attack on the Tien Phone Bank in Vietnam (December 2015) and the attack on the Bank of Bangladesh (February 2016), malware samples recovered from all three incidents were linked. Furthermore, these malware samples were found to share similar code with malware used by the APT group Lazarus.[10]

3.4 Tien Phong Bank – $1,130,000 (Attempted)

During the attack on the Tien Phong Bank (December 2015), attackers used malware that specifically targeted the Foxit PDF reader, which was known to be used by the bank employees when viewing SWIFT statements. Attackers were able to install a malicious version of the Foxit PDF reader on employee workstations, which altered statements (when opened) in order to hide evidence of any malicious activity. This malware was found to be installed on infrastructure provided by a third-party vendor, specifically used to provide the bank's connection into the SWIFT messaging network. Through a carefully-planned and sophisticated attack, employees at the Tien Phong Bank identified suspicious SWIFT messages and rapidly contacted all parties involved. This prevented the transfer requests from being completed and the attempt to steal $1,130,000 was halted.

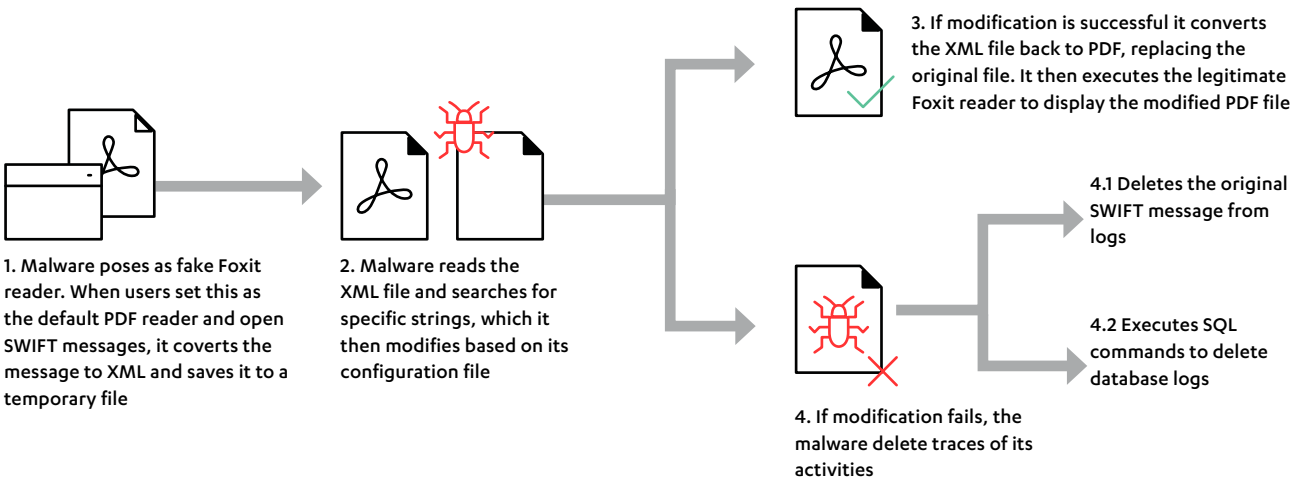


Figure 5 - Execution of malware used in Vietnam hack[11]

3.5 The Bank of Bangladesh – $81,000,000 (of $951,000,000)

The cyber attack on the Bank of Bangladesh (February 2016) was one of the largest heists and most calculated and sophisticated attacks against SWIFT systems to date. Investigations found that the attack had been patiently executed over the period of almost a full year.

Attackers gained access to the bank's internal systems and deployed trusted Windows software to monitor employee activity. Using this initial foothold, attackers were able to move laterally across the bank's internal network in search of SWIFT-connected systems.

Once access to SWIFT systems was obtained, the attackers monitored employee behaviour, stole user credentials, and deployed specifically-designed malware. The malware targeted the SWIFT Alliance Access application, bypassed its security controls, and removed evidence from printed SWIFT messages.

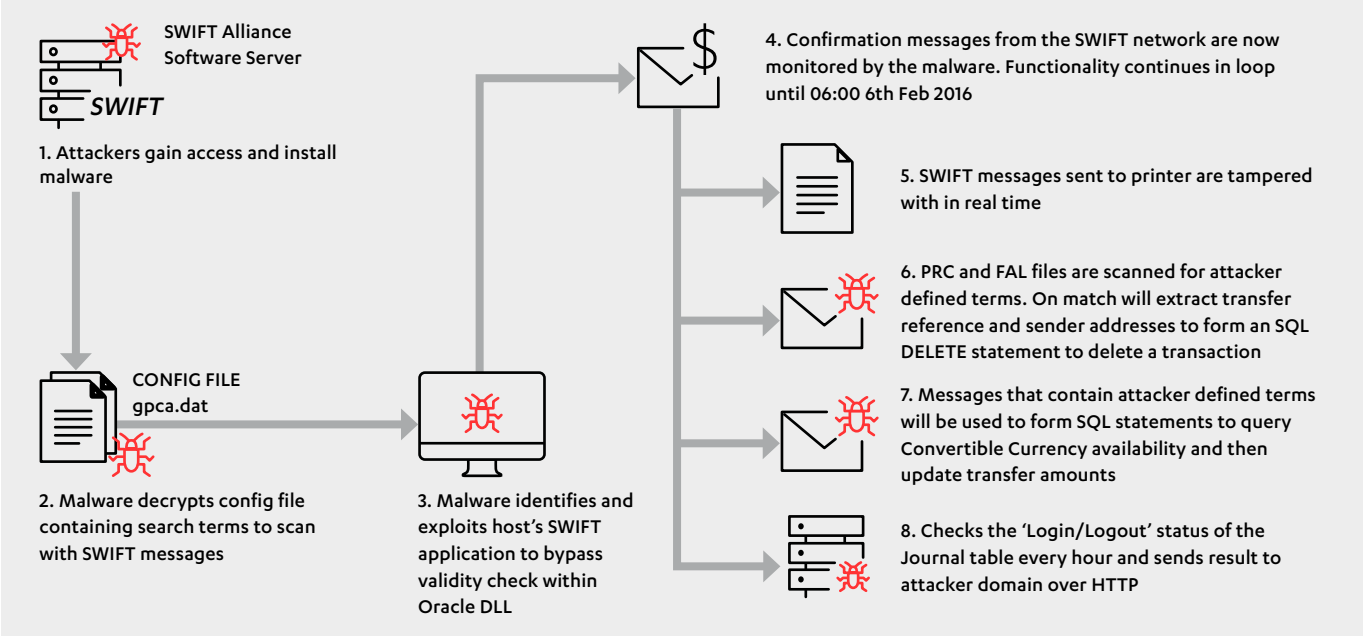


Figure 6 - Overview of the Bangladesh attack[12]

In order to grant the ability to execute database transactions, the malware targeted a specific module that was responsible for managing some of the core functions of the database.

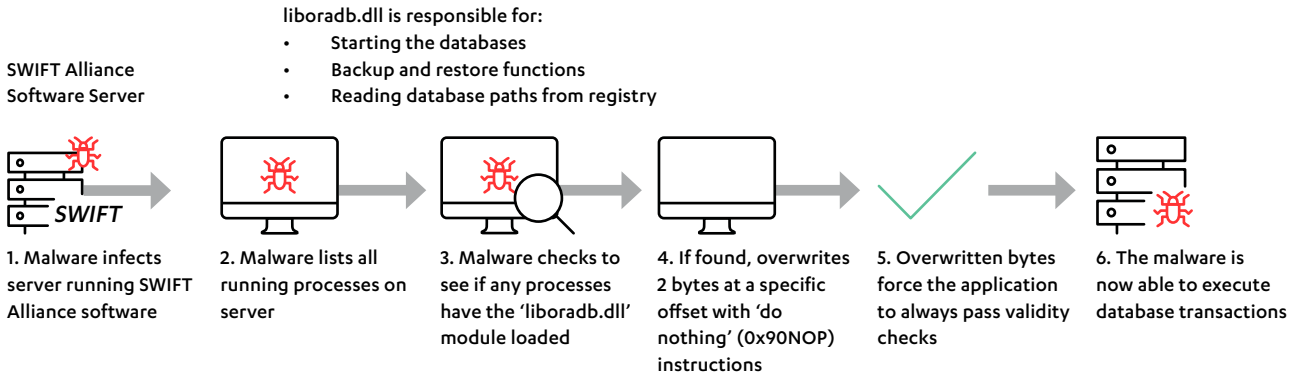


Figure 7 - How the malware exploited liboradb.dll[12]

A total of 35 SWIFT transactions worth $951,000,000 were made. However, only $81,000,000 of this was successfully exfiltrated to bank accounts in the Philippines, while the rest were blocked and recovered due to typos being identified within a number of SWIFT messages sent.

3.6 Unnamed Ukrainian Bank - $10,000,000

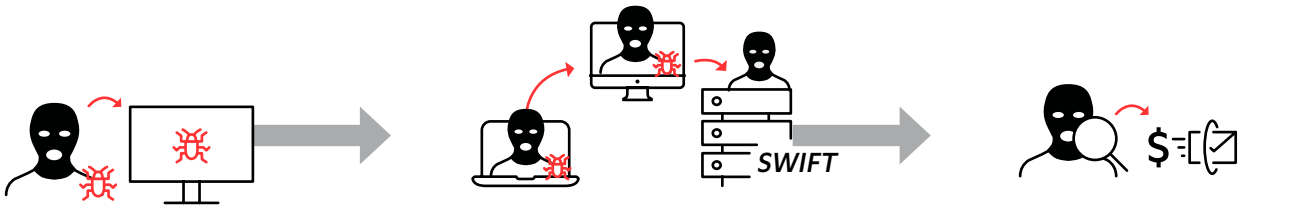
Details regarding the compromise of an unnamed Ukrainian bank (2016) are limited, though it was reported that $10,000,000 was stolen and that the attack was similar to that of the Bank of Bangladesh. It was further reported that this attack was only one of many that the Ukraine and Russia had experienced, resulting in the loss of "hundreds of millions of dollars".[13]

3.7 The Far Eastern International Bank - $160,000 (of $60,100,000)

In October 2017 an attack was carried out against the Far Eastern International Bank. During the heist, attackers used malware similar to that used by the APT group Lazarus, which, as reported, has been linked to multiple attacks on financial institutions around the world.

This malware was used to gain access to and move through the bank's internal network in order to infiltrate SWIFT systems. Attackers then compromised employee credentials and used this information to authenticate to the SWIFT Alliance Messaging Hub and issue a total of $60,100,000 worth of fraudulent transactions. Although it was initially understood that $500,000 was lost, the Financial Supervisory Commission (FSC) reported that the final amount lost by Far Eastern Bank was $160,000.[14]

Following an investigation, it was found that the bank's security posture was not in line with the requirements outlined by Taiwan's banking law. As a result, Taiwan's financial regulator fined the Far Eastern International Bank $266,524, raising the total financial loss of the incident to $426,524.



1. Attackers deploy malware to access the bank internal network

2. Laterally moves across the network in search of SWIFT connected systems

3. Attackers compromise SWIFT operator credentials and use SWIFT software to issue fraudulent transactions

Figure 8 - Attack Path: The Far Eastern International Bank[15]

3.8 The NIC Asia Bank - $580,000 (of $4,400,000)

The most recent attack on SWIFT systems was the attack on the NIC Asia Bank in October 2017.[16] Attackers specifically targeted the bank during the Hindu festival Tihar, one of Nepal's largest holidays.

According to reports, $4,400,000 of fraudulent SWIFT transactions were issued during the attack. However, NIC identified the suspicious activity and informed Nepal Rastra Bank (which is Nepal's central bank), resulting in the recovery of all but $580,000 of the $4,400,000.

At the time of writing, investigations into this attack were still ongoing. KPMG India's forensic team, who were commissioned by NIC Asia Bank to perform a digital investigation, had requested two additional weeks of time to complete their investigations.[17]

3.9 Common Factors

In review of these high-profile attacks, the first common factor is that almost all of these attacks involve the deployment of some type of malware onto a bank's internal systems. Furthermore, we see that attackers frequently pair this with the compromise of user credentials. An overview of the tactics deployed by the attackers has been summarised in figure 9.

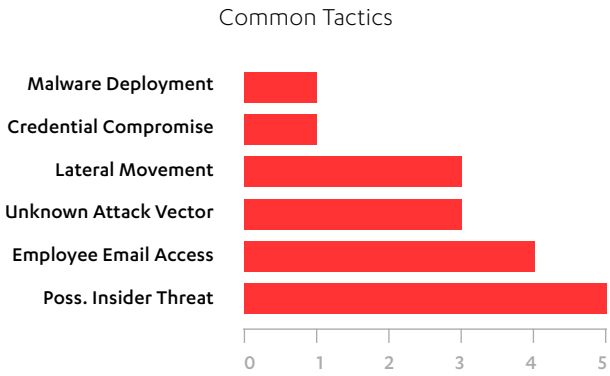


Figure 9 – Overview: Case studies per common threat actor tactic

Overall, it can be concluded that none of the attacks directly compromised the SWIFT network itself, and that they were frequently the result of flaws in the security controls deployed across the targeted bank's IT environments. Furthermore, they are frequently paired with some type of user error; however, this comes as no surprise. In 2014, IBM's Cyber Security Intelligence Index[18] reported that 95% of all incidents recognise "human error" as a contributing factor. This is due to the fact that, regardless of how strong the security of a system is, many security controls can be bypassed by human error. As an example, a password used to access a secure system or resource, no matter how complex, will only remain secure if kept secret. If the password is inadvertently or deliberately disclosed, this information will undermine any authorization and authentication controls implemented to restrict access to that system or resource.

4. WHAT IS SWIFT CSP?

As a countermeasure to the current cyber-threat landscape, SWIFT introduced the Customer Security Program (CSP) to support SWIFT customers in securing their local SWIFT infrastructure. This program requires that all customers implement a set of mandatory and advisory security controls outlined within SWIFT's Customer Security Controls Framework (CSCF).[19]

These controls have been identified by SWIFT based on cyber-threat intelligence and in collaboration with industry experts, and are articulated around three main objectives:

1. Secure Your Environment	2. Know and Limit Access	3. Detect and Respond

These objectives are further broken down into 27 security controls, each of which mitigates one of the specific risks that SWIFT customers face. The main purpose of these controls is to mitigate:

1. The unauthorized sending or modification of financial transactions;
2. The processing of altered or unauthorized SWIFT inbound transactions;
3. Business conducted with an unauthorized counterpart;
4. Breaches of business data, computer systems, or operator details.

Collectively, these 27 controls will create a "Secure Zone" in which – at a minimum – all local SWIFT infrastructure will reside. This will isolate all local SWIFT systems from the wider enterprise network and place an emphasis on the security of all systems within this Secure Zone.

These controls cover seven specific principles:

1. Restrict Internal Access and Protect Critical Systems from the General IT Environment;
2. Reduce the Attack Surface and Vulnerabilities;
3. Physically Secure your Environment;
4. Prevent Compromise of Credentials;
5. Manage Identities and Segregate Privileges;
6. Detect Anomalous Activity to Systems or Transaction Records;
7. Plan for Incident Response and Information Sharing.

The application of these security controls varies based on what SWIFT infrastructure is located locally within an institution's environment. In recognition of this, SWIFT has grouped architectures into four main models:

A1 – Full Stack
Both the messaging interface and communication interface are within the customer's local environment.

A3 – Connector
Only a software component (e.g. Alliance Lite2) is present within the local infrastructure, which is used to connect to a SWIFT service provider.

A2 – Partial Stack
The messaging interface is within the customer's local environment; however, a service provider manages the communication interface.

B1 - No Local Footprint
No SWIFT-specific infrastructure component is within the customer's local environment.

A diagram representing a Full Stack local SWIFT infrastructure model can be seen below:

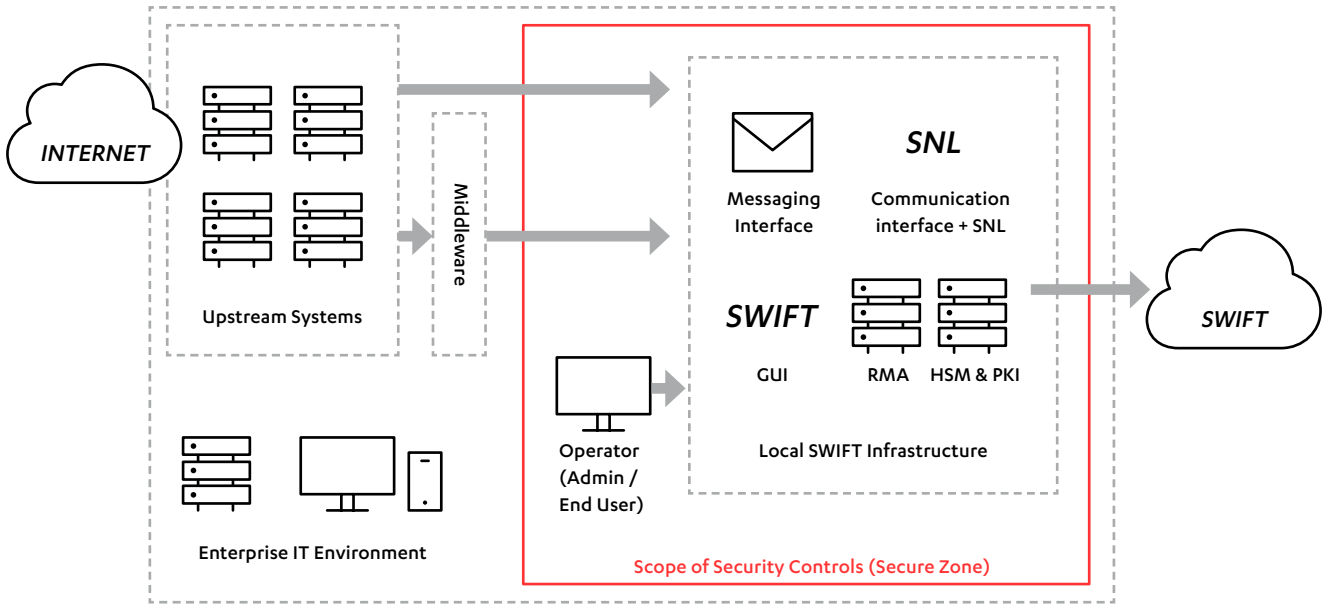


Figure 10 – "Full-Stack" SWIFT infrastructure

The components that are in scope of CSP (as shown in Figure 10) are broken down as follows:

1. **Data Exchange Layer**
   This is the flow of data between the upstream systems/middleware and the local SWIFT infrastructure.

2. **Secure Zone**
   This is a segmented portion of the network, isolating SWIFT systems from the rest of the enterprise environment.

3. **Messaging Interface**
   This is a software product (e.g. Alliance Access) supporting the use of SWIFT's messaging services. This is typically connected directly to the Communication Interface.

4. **Communication Interface**
   This is a software product (e.g. Alliance Gateway) that provides a link between the SWIFT network (SWIFTNet) and the Messaging Interface software.

5. **SWIFTNet Link (SNL)**
   This is a mandatory software product for access to messaging services over a secure IP network (within the above diagram the SNL is part of the Communication Interface).

6. **Connector**
   The Connector is a local software product (e.g. Alliance Lite2 AutoClient) that facilitates communication with a Messaging and/or Communication Interface.

7. **HSM & PKI**
   This is the SWIFT Hardware Security Module and Public Key Infrastructure.

8. **RMA**
   The Relationship Management Application (RMA) is a SWIFT-mandated filter that enables customers to define which counterparties are permitted to send FIN messages to the institution.

9. **Operators**
   Operators are individual end users and administrators who directly interact with the local SWIFT infrastructure.

10. **Operator PCs**
    These are the end users' or administrators' computer devices, used to operate or maintain the local SWIFT infrastructure.

The components which are not in scope of CSP are:

1. **Upstream Systems & Middleware**
   Systems responsible for business logic, transaction generation, and other activities that occur before transmission of data into the local SWIFT infrastructure.

2. **Enterprise IT Environment**
   This is the general IT infrastructure used to support the broad organization (e.g. Mail server, directory services, employee PCs, etc.)

As seen in Figure 10, the scope of the CSCF security controls is highlighted in orange. If implemented to its fullest extent it will effectively isolate all local SWIFT infrastructure from the wider enterprise environment, leaving only the communication channel from the upstream systems and the middleware as an entry point.

5. IS SWIFT CSP COMPLIANCE ENOUGH?

Implementing all mandatory (and advisory) controls specified by CSP will greatly improve the security of your local SWIFT infrastructure and also ensure that all SWIFT customers have a baseline standard for their security. However, it should not be seen as a perfect solution to preventing further attacks. Within the CSP document itself, it is stated that CSP should not be considered an exhaustive approach to security and it does not replace a well-structured security and risk framework. By design, its purpose is to provide a baseline-standard for the security of all local SWIFT systems, only. As such, general attack methodologies can still be applied to the most secure of critical systems.

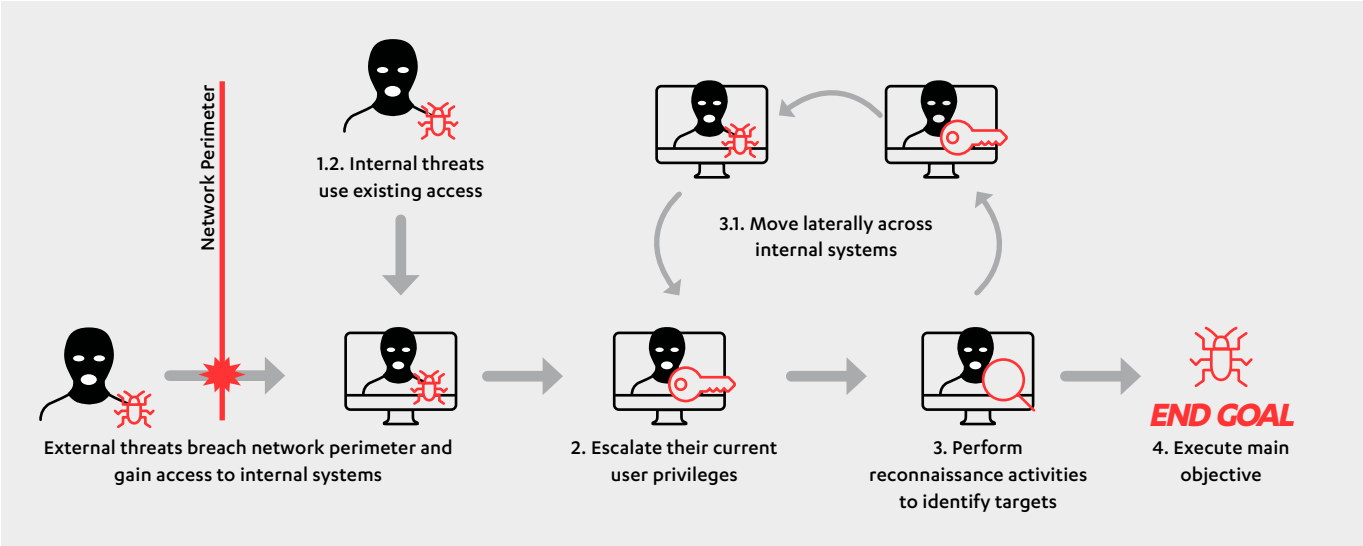


Figure 11 - General Attack Methodology

> SWIFT systems are, and will remain, high-profile targets for all threat actors operating with financial motivations. Regardless of the implementation of standard or advanced security controls, there is still a high risk that these systems will have flaws that will be identified, targeted, and exploited by persistent threat actors. This is a consequence of the complex nature of the infrastructure deployed within financial institutions.

The main focus of CSP is to isolate all SWIFT systems into a secure zone. Without this type of security, attackers could have the opportunity to access SWIFT systems from a number of locations within the general enterprise network. The implications of such an environment are represented below:

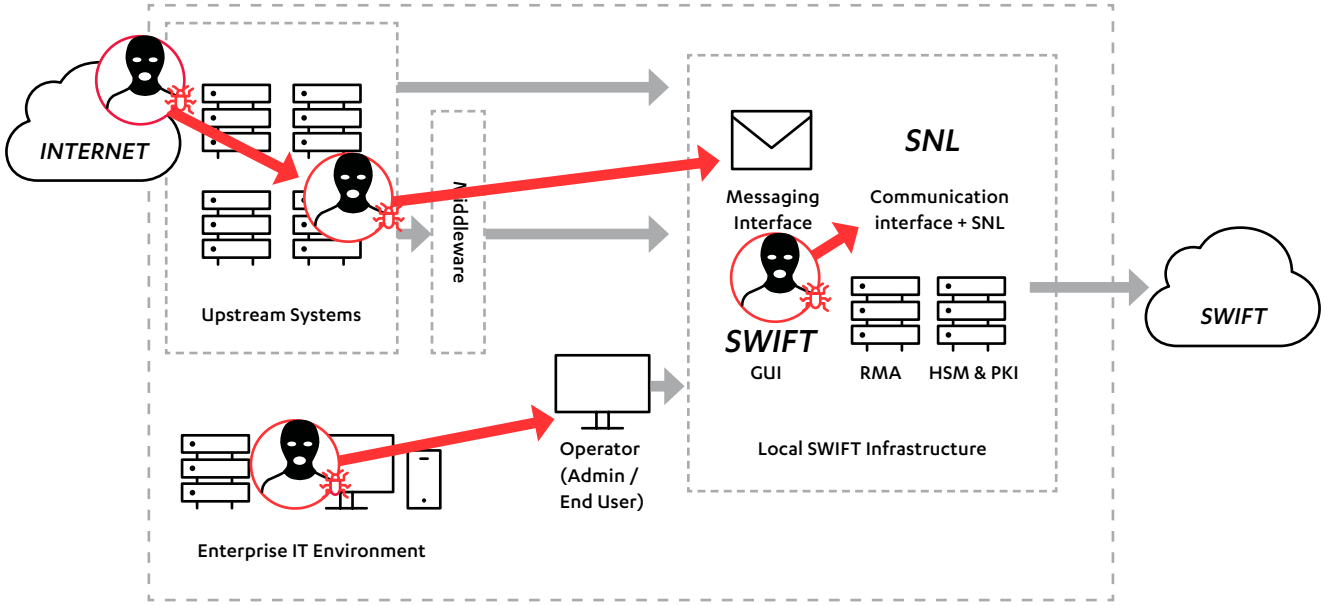


Figure 12- Attack Vectors: Weak Security

With the implementation of all controls within CSCF, the attack surface of the SWIFT infrastructure is considerably reduced, removing a number of attack paths that could previously be exploited to access key parts of these systems. However, these controls do not render SWIFT systems impenetrable, as there must always be a connection from within the local SWIFT infrastructure (Secure Zone), to the upstream systems. This is the weakest link in the security of all local SWIFT deployments.

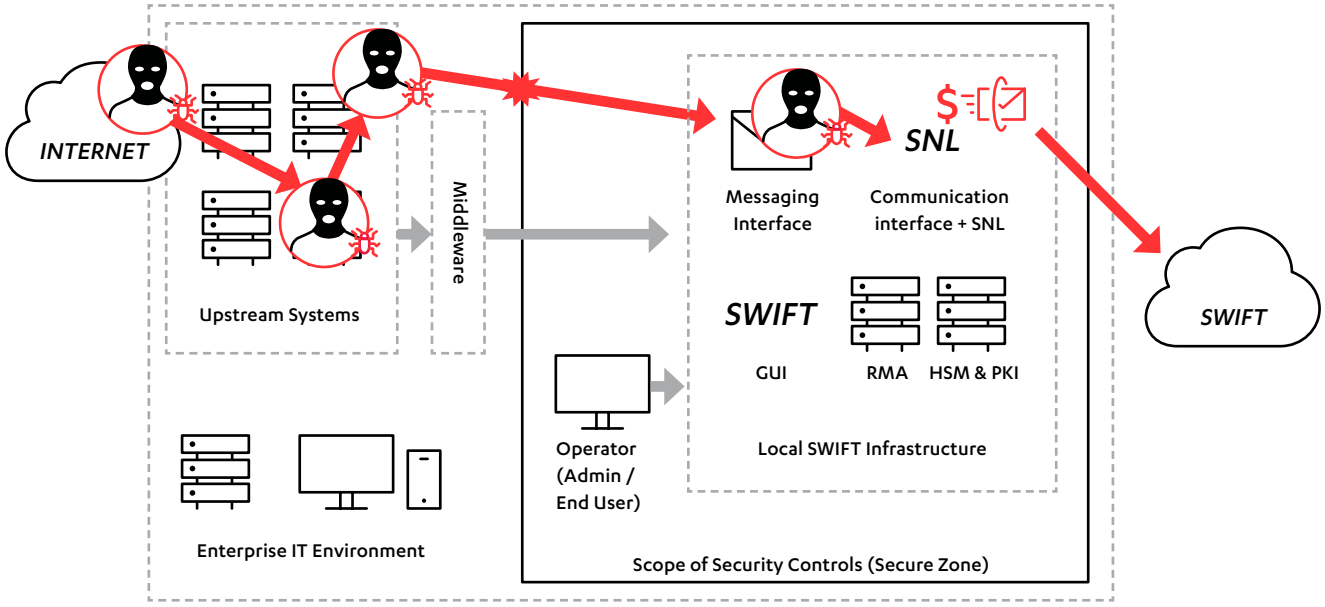
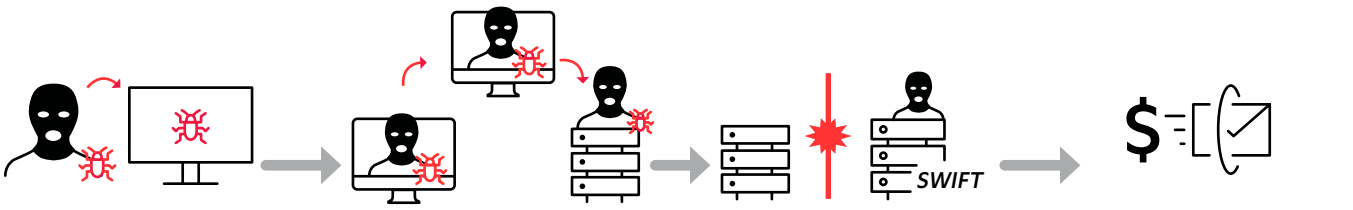


Figure 13 – Attack Vector: SWIFT CSP

The attack described in Figure 13 can be broken down and generalised by the following diagram:



1. Attackers compromise Bank network via vulnerabilities within external facing systems, or via existing access

2. Attackers move laterally across network in search of upstream SWIFT systems

3. Attackers exploit upstream systems to breach the perimeter of the SWIFT Secure Zone, gaining access to the Bank's local SWIFT infrastructure

4. Attackers then exploit SWIFT users and/or applications to issue fraudulent transactions

Figure 14 – Hypothetical Attack Path of a SWIFT Infrastructure Compromise

Analysis of this attack shows that the methodology from Figure 11 still applies, even with all CSP security controls in place. A mapping of the above attack to the methodology is as follows:

In order to compromise the internal network attackers could:

1. Compromise the network perimeter and establish a foothold within the local network;
2. Escalate their current privileges (e.g. via system exploits or by obtaining user credentials);
3. Perform reconnaissance activities to identify the next target system;
4. Repeat to move laterally across the network in search of the end goal (SWIFT upstream systems).

The process is then repeated to further compromise the isolated SWIFT systems:

1. The attackers breach the SWIFT network perimeter and establish a foothold within the network;
2. Escalate privileges in order to gain access to SWIFT system functionality;
3. Perform reconnaissance to understand how transactions can be performed and authorised;
4. And finally, execute their end goal (submission of fraudulent transactions).

5.1 Summary

In summary, by analysing these systems, MWR has identified in theory – and observed in the wild – a number of possible attack paths that could be exploited by a motivated attacker to compromise an institution's local SWIFT infrastructure. However, a significant number of these attack paths were no longer possible following the implementation of the security controls outlined by CSP's CSCF. SWIFT CSP will significantly improve the security posture of an institution's local SWIFT deployment. However, it is a compliance challenge, and therefore cannot be relied upon alone to mitigate and prevent the compromise of these complex payment systems.

6. SO, HOW CAN YOU BETTER SECURE YOUR LOCAL SWIFT SYSTEMS?

SWIFT CSP compliance will ensure that your local SWIFT systems are hardened and isolated within a "Secure Zone". However, there will remain a number of upstream systems within financial institutions that can be used to action payments through SWIFT, which do not reside within the scope of SWIFT CSP's "Secure Zone". As such, financial institutions must:

- Predict
- Prevent
- Detect
- Respond

6.1 Predict

It is key that financial institutions begin by understanding and mapping out the possible attack paths an attacker could take when attempting to compromise their enterprise network and local SWIFT infrastructure. This process begins at the SWIFT systems and works backwards towards the enterprise network perimeter in order to identify which systems communicate with the SWIFT infrastructure and the administration procedures surrounding these systems.
Furthermore, all systems and applications deployed within the institution must be subject to frequent security assessment and penetration tests. A number of attempted (and successful) attacks on financial systems are never publicly reported, and as such organizations are advised to build trusted relationships with other local and international financial organizations to share information on tactics and tooling.

6.2 Prevent

Once these attack paths have been identified, an analysis of the steps an attacker would need to take to complete these paths should be ascertained. The controls surrounding each of these steps should then be assessed to confidently determine whether or not they would prevent such actions. This process should include security assessments of all controls along the path, as well as establishing an understanding of the legitimate use cases for all components. Financial institutions should also establish a strong understanding of which permissions and actions privileged users have access to, and how an attacker could subvert or abuse these privileges. If these actions are necessary and cannot be prevented, monitoring and detection of malicious behaviour should be implemented.

A strong focus should also be placed on establishing controls that prevent malware execution. Furthermore, these controls should be redundant in the event one fails or is bypassed. e.g.:

- **Mail Gateway** – Highly-restrictive controls; file types limited to only those necessary; signature detection of malware; sandbox malware detonation;
- **Endpoint Devices** – Software whitelisting to prevent arbitrary binary execution; control of script and macro execution;
- **Account Control** – Removing privileges wherever possible and adopting a "just in time" / "minimal effective access" approach to authentication, supported with multi-factor authentication.

6.3 Detect

Recovery from these types of cyber heists is highly dependent on a timely response, facilitated by an efficient attack detection strategy. Discovering that a compromise has occurred when reading an end-of-day report is of little use. It is crucial that financial institutions implement robust logging of all key servers within the environment and maintain visibility of servers and endpoint devices through endpoint detection and response (EDR) technologies.
MWR further recommends that institutions adopt a threat-hunting approach to detection and ensure that threat hunters are familiar with payment systems, as well as all known attacks against SWIFT systems. This should include prioritisation of the endpoints (including jump hosts) that are used by privileged users as these are the endpoints that are likely to be targeted by advanced threat actors during an attack.

6.4 Respond

When prevention fails, it is these detection and response capabilities that will ultimately determine the overall financial impact of an institution's local SWIFT infrastructure being compromised. Therefore, it is important that resources be placed in establishing a mature detection and response strategy surrounding your SWIFT deployment and its upstream systems. The main goal of this is to efficiently contain and recover from an attack.

Regular incident response exercises should be conducted by financial institutions to ensure that the policies and procedures in place facilitate rapid response to an incident. This should include table-top exercises to test these procedures, as well as full incident response run-throughs based on SWIFT systems. Attack case studies such as the heist of the Bank of Bangladesh should be mapped to the organization's systems and the response team should work through this to establish if an investigation could have been rapidly conducted on their systems in the event of a similar attack.

7. CONCLUSION

In the present day cyber-threat landscape, attacks on financial and SWIFT systems are the focus of advanced persistent threats (APT). As Willie Sutton reportedly stated in 1952[20], when asked why he robbed banks, "that's where the money is". With huge quantities of money for the taking, attackers are becoming considerably more sophisticated, persistent, and resourceful. In some of the most high-profile attacks, threat actors have frequently deployed specifically-designed malware and used advanced tactics to achieve their goal of performing fraudulent financial transactions. In response to this, SWIFT has introduced CSP to help protect its global SWIFT community from this threat.

However, we have found that SWIFT's CSP is a compliance challenge, which by nature is a rigid, linear process. Compliance does not ensure or imply security, as security itself is a fluid, cyclical process – always adapting and changing. CSP focuses on the security of SWIFT infrastructure alone; however, MWR has observed that attackers may shift their resources into targeting upstream systems and/or the users who operate within them. Furthermore, MWR has observed opportunities in the wild for suitably positioned attackers to leverage these systems to perform a successful attack. Whilst SWIFT's CSP recommends a number of excellent hygiene measures, focusing solely on the SWIFT payment systems will simply push attackers to target other parts of the organization.

As with most compromises, the root cause will frequently remain human error, whether this error be made by administrators in a configuration file, developers in their application code, or employees being deceived into opening a malicious email attachment. For this reason, MWR recommends an approach that builds on top of SWIFT CSP compliance to further strengthen the security posture of an institution as a whole, including its local SWIFT infrastructure. This methodology is rooted in establishing a strong understanding of how modern threat actors target financial institutions, mapping this understanding to the organization, and selecting appropriate preventative, detective and responsive measures.

A 'point in time' approach to security will never succeed against an adaptive and persistent threat. The cyber-threat landscape is always shifting, and so only by turning the proposed methodology into a recurring practice can financial institutions and other organizations hope to secure themselves against future threats.

References

1. https://www.reuters.com/article/us-usa-cyber-swift-exclusive/exclusive-swift-confirms-new-cyber-thefts-hacking-tactics-idUSKBN1412NT
2. https://www.swift.com/myswift/customer-security-programme-csp
3. https://www.swift.com/myswift/customer-security-programme-csp/security-controls
4. https://www.swift.com/about-us
5. https://www.swift.com/our-solutions/global-financial-messaging/payments-cash-management/interbank-payments
6. http://baesystemsai.blogspot.co.uk/2017/10/taiwan-heist-lazarus-tools.html
7. https://www.symantec.com/connect/blogs/swift-attackers-malware-linked-more-financial-attacks
8. http://uk.reuters.com/article/us-cyber-heist-bangladesh/exclusive-bangladesh-probes-2013-hack-for-links-to-central-bank-heist-idUKKCN0YG2UT
9. https://www.nettitude.com/wp-content/uploads/2016/12/Nettitude-SWIFT-Threat-Advisory-Report-client.pdf
10. https://www.symantec.com/connect/blogs/swift-attackers-malware-linked-more-financial-attacks
11. http://blog.trendmicro.com/trendlabs-security-intelligence/high-profiled-cyber-theft-against-banks-targeted-swift-systems/
12. http://baesystemsai.blogspot.co.uk/2016/04/two-bytes-to-951m.html?m=1
13. https://www.kyivpost.com/article/content/ukraine-politics/hackers-steal-10-million-from-a-ukrainian-bank-through-swift-loophole-417202.html
14. https://www.reuters.com/article/us-far-eastern-fine/taiwans-far-eastern-international-fined-t8-million-over-swift-hacking-incident-idUSKBN1E60Y3
15. http://baesystemsai.blogspot.co.uk/2017/10/taiwan-heist-lazarus-tools.html
16. https://www.bankinfosecurity.com/report-attackers-hacked-nepalese-banks-swift-server-a-10437
17. https://thehimalayantimes.com/business/kpmg-team-seek-time-draw-conclusion-nic-asia-bank-case/
18. https://media.scmagazine.com/documents/82/ibm_cyber_security_intelligenc_20450.pdf
19. https://www.accesspay.com/wp-content/uploads/2017/09/SWIFT_Customer_Security_Controls_Framework.pdf
20. https://quoteinvestigator.com/2013/02/10/where-money-is/


F-Secure