okay

SCA AND SECURITY:

Industry Challenges

Introduction

With the Payment Services Directive 2 (PSD2) and the corresponding Regulatory Technical Standard (RTS), there has been an increased focus on the security of the authentication process, both from the regulatory authorities and from the private sector. We at Okay have been involved in this process since before the first drafts of the PSD2, both as a service provider and as a member of the Emerging Payments Association (EPA) in the UK and l'Association du Paiement in France.

When implementing Strong Customer Authentication (SCA), most actors come across a number of challenges. Through conversations with actors across the PSD2 SCA industry, we have identified some issues that are shared throughout the industry, regardless of company size or type of financial institution.

Based on this insight, we have created a list of 8 SCA challenges that we will explore further in this white paper. This paper has been written with issuers and PSPs as the primary audience, but hopefully, it is also of use for anyone interested in the challenges that SCA providers meet when trying to implement a friction-free and security compliant solution.

Challenge 1
SCA Security and 2FA

SCA Security and 2FA

There is no denying that two-factor authentication (2FA) has significantly improved authentication security. Passwords and passwords resetting systems have proven to be a prime target for hackers, and thus simple password protection is a known security risk. Adding a second factor to the authentication process - whether it is possession, inherence or knowledge - does help fight cybercrime, but is it enough to keep it secure? And is 2FA sufficient to cover the requirements of strong customer authentication under PSD2?

Security requirements in the PSD2 RTS

Across the industry, there appears to be a lack of focus on the security of the 2FA process itself. In our opinion, that is an oversight. Securing the process of the 2FA is essential in order to fully protect the authentication.

The RTS does not specify a required level of security around the authentication process, but it does require security implementation on many of the steps. A few of these are:

- Protect the transaction, its information and the authentication
- Protect the dynamic linking
- Prevent new innovative attacks

Without proper security throughout the entire authentication process, one of the factors can easily be broken, and a fraudster will be able to take control of the process and compromise the security of the authentication. An excellent example of this is from Welivesecurity; they discovered a trojan malware that could break into Paypal's security even with 2FA activated. [1]

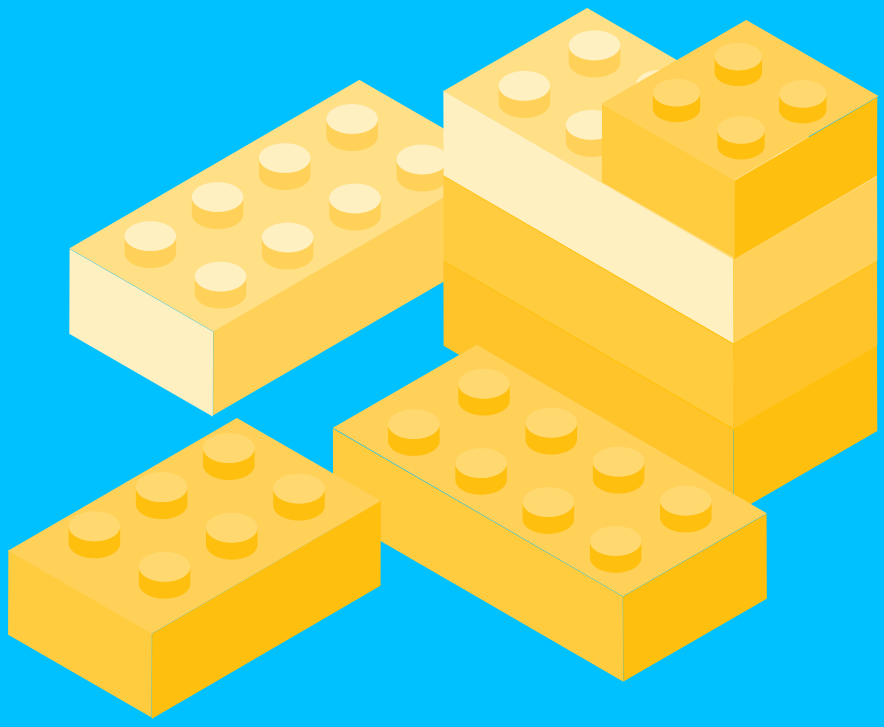
The fight against hackers

As we, the security industry, step up our game, so will the hackers and fraudsters. At Okay, we believe that malware is the most innovative vector for attacks. Either the hackers will try to use malware to start mass attacks, or they will use attacks to specifically target individuals. Company treasurers and individuals who are wealthy in their own right are prime targets for these types of attacks.

To be fully PSD2 SCA compliant the environment where the authentication happens has to be properly secured. Thus, 2FA is just part of the PSD2 SCA solution.

[1] https://www.welivesecurity.com/2018/12/11/android-trojan-steals-money-paypal-accounts-2fa/

Challenge 2
Constraints for a full SCA implementation



Constraints for a full SCA implementation

Authentication continues to be a headache for many within the banking industry. While everybody would like to have a "one-size-fits-all" solution to SCA, the fact is that the authentication use-cases are so complex that it might not be feasible in a cost-efficient and user-friendly way.

For the majority of challenger or neo-banks, everything should seem easy: For most of these, the entire business model is based on the fact that their users can do everything on a smartphone. However, this is not the reality for established banks with a very diverse client base, corporate banks, or even challenger banks working on financial inclusion.

While talking to different actors within the banking industry, a few common security-related use cases connected to SCA have crystallised.

First, we see four common use-cases related to SCA:

- The initial onboarding to the SCA solution
- SCA and transaction verification
- Two-factor authentication itself (2FA)
- Re-enrolment to the SCA solution (lost device or credentials)

These use-cases are common across user types, but the technical environment of the user varies for the different user groups and have an impact on SCA and 2FA. Based on our experience, we define five different technical scenarios that cover almost all users:

Smartphone only

The user has a smartphone, and they have network access. These are the users that need single device authentication like the Okay solution for SCA. How onboarding is done is up to the bank or issuer, but we suggest using either an SMS message, which proves possession, or the traditional letter with OTP. For re-enrolment to the service, we believe SMS by itself is not enough, because of "SIM-swap-fraud": Someone might register a new address on you, then have a new SIM card sent to that address, which then can be used to access your account. Using a single factor (such as knowing the PIN) is not enough.

Smartphone only, but no data coverage

How do you serve a customer who only has access to a smartphone but does not have a data connection to that phone? That is the market of traditional offline eWallet providers, which used to be tied to special hardware in the handset. For SCA and transaction verification it would, of course, be possible for the user to use a dongle, but the question here is; what kind of transaction can the user initiate without data coverage?

Smartphone with no data + PC with network

These users have a computer with network access, but no data connection for their smartphones, which can be the case in some rural areas and industrial settings. Not having a data connection to the phone makes the onboarding and enrolment process challenging. In this scenario, it is not possible to create a link between the smartphone and the account. Not having a data connection also makes re-enrolment even more of a challenge.

A dongle would be an alternative for authentication for these users; however, it is important to note that not all dongles are usable under PSD2. A dynamic linking between the transaction data and any code provided is required to be PSD2 compliant. Thus, a dongle that can communicate with the PC is the only usable alternative under PSD2. Most dongles and card-readers today do not fulfil this requirement.

PC with network + no smartphone, but landline available

This is the "grandmother-with-PC" scenario, but this is also the case for some enterprise customers: The customers have PCs and landlines, but no smartphones are available. That is a use case that can be handled with dongles, but dongles are expensive and inconvenient. An alternative is to use a voice call to a landline. The user has to listen to the details of the transaction, and then either get a transaction authentication number (TAN) from the call to be entered on the PC or a TAN from the PC has to be given during the call.

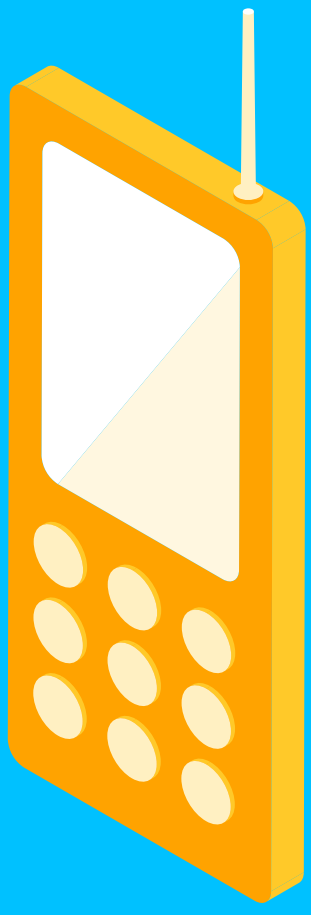
Here we assume that the security of voice calls to a landline is strong enough, which might not always be the case. However, a voice call on a landline would surely be more secure than a text message to a smartphone. In our opinion, a voice call can be strong enough for transaction verification, possibly depending on value, but it is not recommended for onboarding or re-enrolment.

PC with network + low-tech phone - "dumb-phone"

Here we have the same issue as in the previous scenario, but the user has a "dumb-phone" with the option of receiving SMS. Keep in mind that SMS messages can be intercepted and SIM-cards are vulnerable to a "SIM-swap attack". If OTP by SMS is your only option, it is worth looking into providers that can do this in as a secure way as possible.

Of course, in addition to these five scenarios, there are several potential problems with SCA linked to accessibility that are not included in these scenarios, as accessibility requires interfaces that can be misused by malware. If you're curious, you will find more articles on this topic on our website www.okaythis.com.

Challenge 3
SCA for low-tech phones and fallback options



SCA for low-tech phones and fallback options

Today we tend to assume that everybody has a smartphone and is able - and willing - to use it for all tasks, including banking. However, that is not always the case. A significant number of users either won't or can't use a smartphone for their banking needs. How can we offer a PSD2 SCA compliant authentication without a smartphone? Not offering an alternative is not an option.

When is a fallback option needed?

There are some typical scenarios where there is an evident need for a fallback option to the smartphone SCA solution:

- Enterprise users that do not want to or cannot use a personal smartphone in their work
- Users who do not have a smartphone
- Situations where there is no wifi or mobile internet
- When there is doubt about the integrity of the user's smartphone

There is no official data on the total size of the non-smartphone market, but numbers as high as 20 % of the entire customer base has been mentioned. 20 % is a significant number of clients for any bank, and the banks have to supply these clients with a fallback alternative for SCA.

What are the fallback options?

So, when a client can't or won't use their smartphone for SCA, what are the technical alternatives and do they keep the authentication process SCA compliant? Let's take a look at the different options and how they stand in terms of compliance:

- The conventional way is to **use a code generator or a dongle**. Traditionally this was a time-based code generator, where you press a button and get a code that authenticates you. The obvious disadvantage is that dongles are expensive and inconvenient to users, particularly since they have to display transaction details in order to support dynamic linking to be SCA compliant.
- Another common mechanism is to use **SMS one-time password (OTP)** sent to the user's dumb-phone. However, SMS is very vulnerable to malware, and using OTPs sent by SMS makes the user susceptible to phishing. OTPs on its own is thus not considered an SCA compliant option.
- A less used option is to **use a voice call to the user**. In the call, the transaction details are repeated to the user, and the user is either asked to enter a TAN from the screen during the call or to enter a TAN from the call on their computer. This is an SCA compliant option.
- An older alternative - that is no longer legal in Europe - is to **use printed listings of TAN codes**. Surprisingly some banks have been using this method until the end of 2019.
- A last, and perhaps obvious, alternative is to **ask the users to go to a physical bank location**, bringing their physical identification with them.

Each of these solutions has disadvantages, but there are situations where an automated call to a landline would be a lifesaver, i.e. if there is an issue with your smartphone.

Challenge 4
The mobile OS headache

The mobile OS headache

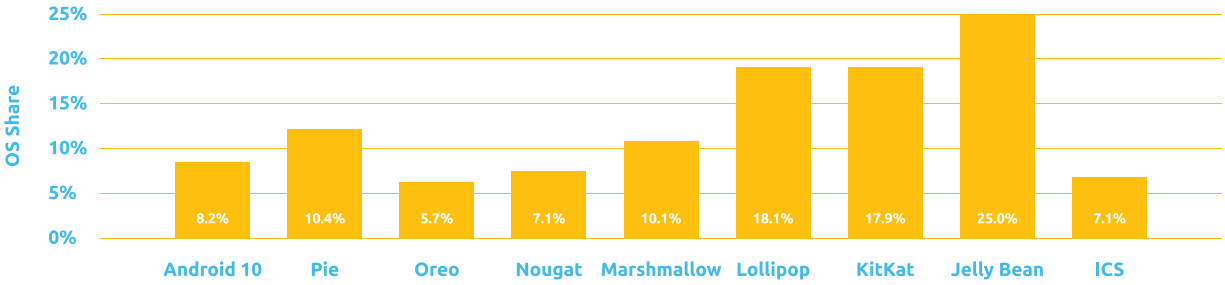
When providing strong customer authentication, smartphones with outdated OSs are one of the major hurdles that have to be addressed. Should these users be "opted out", forced to update or is it possible to find an alternative?

In early February 2020 there was a minor news item that got some traction in the media: WhatsApp, with more than 2 billion users, was ending their support for older versions of Android. The version they ended support for is only 0.3% of the Android devices, but we're still talking about as many as 6 million devices.

Why did the support stop?

It is interesting to speculate why WhatsApp chose to do this. Security is probably one of the main reasons. The version of Android they stopped supporting, version 2.3.7, was released back in December 2011. Since then, there has been a large number of vulnerabilities found in Android, from remote exploits such as "Stagefright" to yet another issue with Bluetooth. Needless to say, these vulnerabilities have not been patched on a 10-year-old version of Android.

There does not seem to be an end to new exploits, and even as recently as February 2020 a new remote exploitable vulnerability was reported, hitting all Samsung Android phones sold since 2014. This type of vulnerability is worrying, as it does not require any user action for the device to potentially be infested with malware.



Android market share for Android OSes 10 months after the launch of Android 10. Older OSes like Jelly Bean initially released in 2012, Lollipop from 2014 and Kitkat from 2013 are still claiming a major percentage of the total market.

Source: https://www.androidauthority.com/android-version-distribution-748439/

That opens the door to a different question: How long can we, as users, expect to receive updates for our Android phone? According to an article by The Verge "*Google's contract with Android partners stipulates that they must provide 'at least four security updates' within one year of the phone's launch. Security updates are mandated within the second year as well, though without a specified minimum number of releases.*"

If you work in IT and replace your phone every two years, there is no issue, but for most consumers, buying a new phone every second year is not an option. Discovering that a new phone is the only way to be sure to have a secure device might be an unpleasant surprise to many users. We estimate that as much as ⅔rds of Android phones don't get regular security updates. For iOS, the situation is a bit better; all phones released in the last five years (iPhone 6s and newer) are still receiving updates.

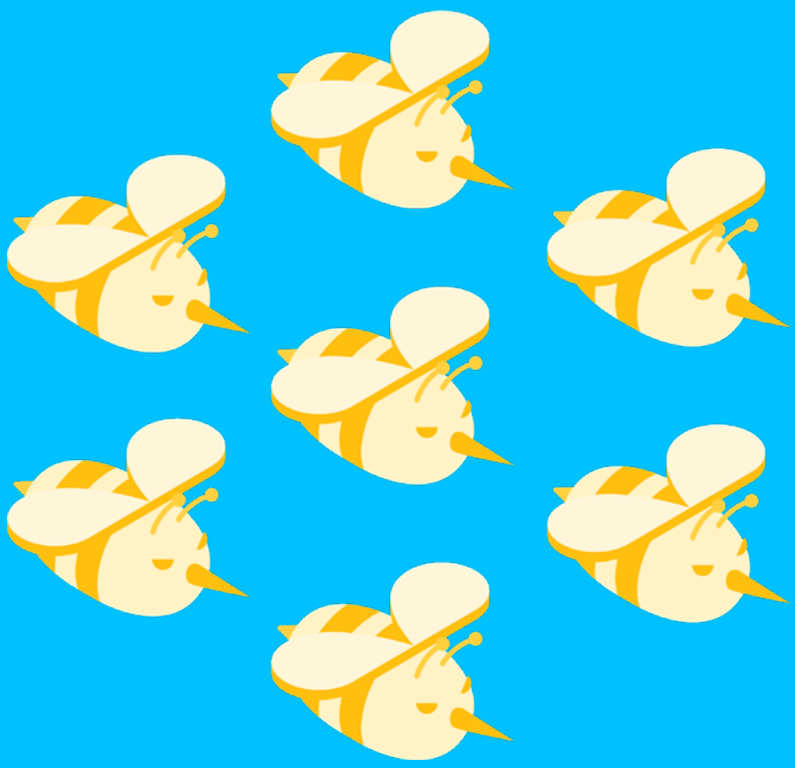
How to deliver SCA on outdated devices?

The lack of security updates on users devices puts banks and fintechs in a bind: What do you do when a large percentage of your user base, potentially even the majority, runs a mobile operating system that is no longer receiving updates?

Today the majority of users are doing their banking on a single device, and the industry cannot expect users to go back to using a PC + phone to do their banking. The two apparent alternatives are to opt-out the users or force them to buy a new phone. Neither case is very appealing. Once you have customers who have deposited money, blocking them from accessing their accounts is at best very inconvenient, and at worst possibly of questionable legality.

The best option is to take a different approach to this issue: One option is to provide a secure execution environment that aims to be as independent of the operating system as possible. You can even assume that the user's device has been infected with malware and that malware is actively trying to attack your application. The aim should be to focus everything on protecting the most sensitive parts of the app: the authentication and transaction verification. This approach can help you - as a bank - to avoid making hard decisions about opting out large numbers of users.

Challenge 5
Innovative malware attacks



Innovative malware attacks

Reading the news there appears to be a new critical security issue almost every week. Some of these get a lot of attention in even daily newspapers, while others go by almost unnoticed. Sadly, the reason that some security issues get a lot of attention is not solely due to the severity of the issue. Often, the marketing of the issue is what determines the media attention. Some security researchers take the time to create a cool logo, and some even develop websites, which increases the chance of media exposure, regardless of the severity.

Even though not all of the malware attacks and security vulnerabilities that get attention are all that interesting, it is still worthwhile to look at the trends in vulnerabilities and how they can be examined.

Malware trends

Over the last few months, there is one particular trend that has gained extra attention; there has been an increase in published attacks on data communication protocols. One example is the attack on the Bluetooth chipset known as BlueFrag, an attack where; "*On Android 8.0 to 9.0, a remote attacker within proximity can silently execute arbitrary code with the privileges of the Bluetooth daemon as long as Bluetooth is enabled. No user interaction is required and only the Bluetooth MAC address of the target devices has to be known.*"[2] But it is not just Bluetooth that is under attack. In late February 2020, the "Kr00k" vulnerability was published. "Kr00k" is a new WiFi vulnerability which can affect the network security of as many as a billion devices.

Another trend that has gotten some attention lately is the activities of national state actors, such as when Jeff Bezos' mobile phone allegedly got hacked by the Saudi Government using a video file sent on WhatsApp. The use of malware for state espionage is, in fact, not a new development, and many reading this might remember the NSA ANT catalogue leaked in 2013. In October 2019, BlackBerry published a good overview that sheds light on how state and state-sponsored Advanced Persistent Threat (APT) groups have been, and still are, using malware for espionage.[3]

Staying up to date on security issues and fixes

For the reported security issues, the most important source is the NIST National Vulnerability Database, where the so-called CVE numbers are registered. This database can be a bit hard to navigate, but there are some aggregators out there that are more user-friendly. One example is CVE Details, which has reports for both Google and Apple, that makes it easy to compare reported security issues. Looking at overviews such as these can give you an impression of how the different vendors are doing with regards to security.

Fixes to the issues can also be investigated. For Android, the official monthly security bulletins is a good place to start. That is the official list of fixes in the latest version of Android, often with a lot of detail, even down to the changes in code to solve the issue. The official fixes for May 2020 included a critical bug which let an attacker impersonate any app without requiring

[2] https://insinuator.net/2020/02/critical-bluetooth-vulnerability-in-android-cve-2020-0022/
[3] https://threatvector.cylance.com/en_us/home/mobile-malware-and-apt-espionage-prolific-pervasive-and-cross-platform.html

root access and a large number of fixes for privilege escalation, which can be used to gain root access. One of the fixes (CVE-2020-0103) could even be triggered remotely, simply by receiving an MMS, or by a web browser. Earlier fixes from 2020 include new issues which can be used to intercept touch events (do stuff like steal PINs and passwords), new vulnerabilities in Bluetooth which could be used to remotely gain access to a device (BlueFrag attack), and more. Apple does have a similar page, but here you have to click in on the individual product updates and look for the CVE numbers to see what is fixed.

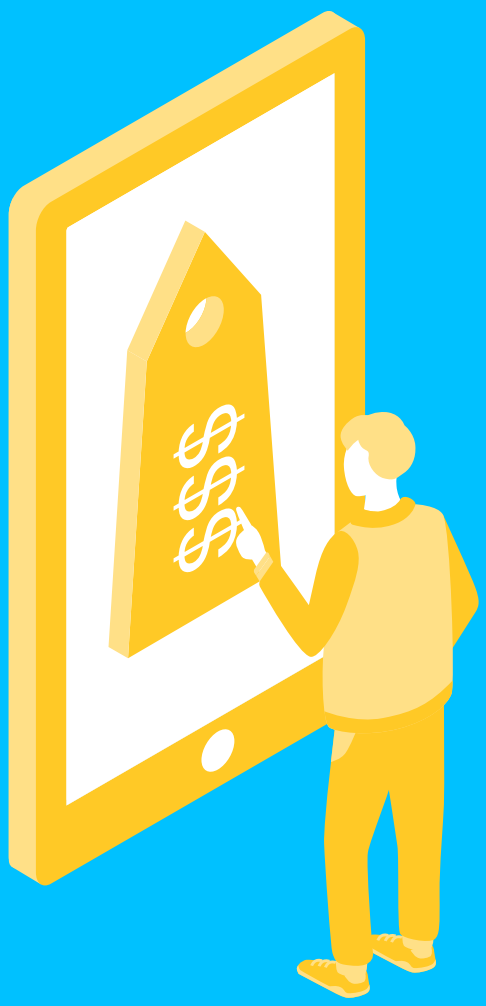
Security issues opening doors for malware

The descriptions used in the security bulletins can be terse and too fond of acronyms. Still, one thing is clear: Most of the security fixes are for the elevation of access, which can often let malware gain root access on a device. In addition, there are usually some which can be used for remotely running code on a device as well.

In other words: There are security issues which can be used to run code on a device remotely, and there are other issues which can let that code run with full access to the device. Also, the exact modification in the code to solve the issue is reported in full. For any user that has a device that is still receiving updates this is not an issue, but how does this affect people who have a device that is simply too old to run the latest security update?

In practice being open about security issues can also help criminals create new and innovative forms of malware, or let national state actors spy on their citizens or other governments. The lack of easy solutions to this issue is one of the reasons Okay made the conscious choice to focus on transaction security, with the hope that at least we can try to help within one single sector.

Challenge 6
The cost of SCA Integration



The cost of SCA Integration

Minimizing integration costs and total cost of ownership is a concern voiced by most issuers when implementing an SCA project. Of course, any IT project involves integration and integration costs; SCA is no different in that regard, but it does have its own cost drivers. In this section, we look at ways for issuers to reduce SCA integration costs as much as possible.

Focus on the transaction, not the app

As an issuer, you have little control over the OS updates of your customers, something that triggers quite a few issues when dealing with security. When scoping requirements, a good place to start is to define what you as an issuer want to protect. When it comes to PSD2 SCA, issuers must protect the transaction - or the challenge - and its authentication process. Security should thus be focused on this particular point.

Review PSD2 SCA fundamentals

- A separate execution environment
- Is dynamic linking enforced?

These two points alone can substantially drive your costs up, depending on the chosen solutions. However, these are mandatory requirements to comply with PSD2 RTS SCA.

A few things to keep in mind when choosing an SCA solution:

- **How is the secure environment implemented, if at all?**
  If it is not implemented, you need to develop it yourself or integrate a third-party solution.
- **What security mechanisms are used to protect the execution of the transaction authentication?**
  The mechanisms must be strong enough to protect the transaction information and detect transaction/code tampering attempts during the execution of the process.
- **How is the code secure execution environment updated, if at all??**
  The execution environment should receive all the information related to a specific transaction, and possibly dynamically update the executable code used to display the SCA UI and specific security mechanisms.
- **Is the dynamic linking functionality part of the solution?**
  There are a few requirements from article 5 of the RTS; the payer must be made aware of the specific amount of the transaction and the identity of the payee; a code specific to the particular transaction must be created.
- **What security measures are offered to protect the dynamic linking?**
  The security measures are needed to protect and ensure “the confidentiality, authenticity and integrity” of the information displayed to the payer throughout the process.
- **Can the SCA user experience be updated with the rest of the app?**
  There is always a bit of conflict between the “UX” designers and the security experts, where the UX designers want to do everything frictionless, while the security experts are focused on making sure of the security.

Avoid the SDK jam

Users of mobile banking are demanding; to satisfy these demands, issuers are continually releasing new features that are either native to the app or delivered via an SDK. That is certainly the case with security solutions. With the layering and multiplication of SDKs, maintenance can become an issue.

Likely, as an issuer, you cannot do without an SDK, especially if the security service is externalised. In this case, it is important to source a "light" SDK when it comes to transaction and authentication security.

Ideally, you should source a solution that natively provides both security **and** multiple-factor authentication (MFA) methods in a single SDK. That is particularly important if you start an SCA project from scratch. That will both save you integration cost and help keep your SDK clean. The MFA methods should not solely be based on receiving an OTP SMS as that is vulnerable to attacks. If OTP by SMS is chosen, the issuer needs to check the device SIM card to prevent SIM-swap attacks or ensure that the device is the right one via a device fingerprinting feature. You should also consider if a SIM check and device fingerprinting by themselves can replace a costly SMS OTP.

On-premise or delivered from cloud?

Although Software as a Service (SaaS) can be seen as a preferred method to reduce the complexity and cost of an SCA project, some issuers opt for an on-premise solution. The size of the organization, together with its maturity and culture, often plays an integral part in the decision process when making this choice.

Obviously, the more you bring in-house, the more it will cost in terms of integration and maintenance. Regardless of delivery method, there are a few points you should investigate before making the final decision:

- What is the API? Is it based on the right technology?
- Is the SDK well documented?
- With the right documentation, any engineer – regardless of seniority – should be able to rapidly get acquainted with the technology, which will speed up the time to market of the app.
- What is the ecosystem of the security vendor?
- Are there any "connectors" or plans to have integration with other solutions?

The ecosystem

The ecosystem surrounding SCA is both rich and complex. When looking at SCA projects, there is rarely one solution that fits all use-cases.

When evaluating an SCA security solution provider, it is vital to explore whether they have connections with:

- Other multi-factor-authentication providers
- Access Control Server providers for payment SCA
- Payment process providers who have built their own authentication servers
- Anti-fraud solutions
- Banking-as-a-service vendors

Challenge 7
SCA enrolment and re-enrolment

SCA enrolment and re-enrolment

Issuers, banks and eMoney wallet providers have all made one thing clear: Enrolment and re-enrolment to the SCA solution are both critical stages in the SCA process. If the enrolment of a client is compromised in any way, all SCA challenges for that client can be compromised, hence a majority if the security focus is on that part of the journey.

When talking about enrolment and re-enrolment security, a good place to start is to look into some actual fraud scenarios:

The SIM swap attack: This is a type of account takeover fraud where the criminal gets a new SIM card for a user. Often this only takes going to a phone store and saying "I lost my phone, can you give me a new SIM card for my number?" If the store does not check the ID properly, the criminal will end up with a valid SIM card for the phone number. Back in 2019, the CEO of Twitter had his Twitter account stolen with this method.

Call centre phishing: In this attack the criminals are calling into the call centre, claiming to be a customer who has lost their credit card or forgotten their password. The same criminal can make multiple calls to the same call centre and continuously learn more and more about their victim. Call centre agents (at least for the good companies) are trained to keep customers happy, so there are even examples where they have sent credit cards to the criminal's address.

Attacks on changing your address or phone number through an app or website: This is a form of attack that can be done with malware. The malware can call functions to change the address or phone number of the user, which can be used to impersonate the user or to take control of their account.

In these three attack scenarios, the criminal focuses on the re-enrolment phase. Attacks on re-enrolment are typically made to gain access to an existing account. A typical goal for these attacks is to steal all or part of the account balance. Attacks on enrolment, on the other hand, typically target financial institutions that offer loans, to take out a loan in somebody else's name. In both enrolment and re-enrolment use cases, the attackers target the know-your-customer (KYC) routines of a company.

Creating a security anchor used for SCA

A useful way to think of enrolment security is that when enrolling or re-enrolling a user on a device, you create a security-anchor. As long as that anchor is intact, it can be used to verify the device. This anchor can thus be used as one of the two authentication factors required to do SCA. If the anchor is lost, or the security of the anchor is in question, you might have to do a re-enrollment, as you might be left with just a single factor, which is not enough to do proper SCA.

In the PSD2 text itself the word "enrol" is not mentioned. However, it is mentioned in Article 21 of the RTS, where there is a requirement that "*The association via a remote channel of the payment service user's identity with the personalised security credentials and with authentication devices or software shall be performed using strong customer authentication.*" Of course, using SCA would mitigate the three fraud scenarios described above. But, how exactly can one do SCA if the user has lost their device?

Challenge 8
SCA for corporate transactions

SCA for corporate transactions

One of the biggest challenges when introducing Strong Customer Authentication (SCA) is to render the purchasing experience fast and frictionless to end-users. When looking at the requirements for speed and convenience, corporations and private users would be at the opposite end of the scale: The amounts transferred by corporate payments can be much larger, so security is paramount. For corporate payments, friction does indeed suggest more security. So how does this all translate to SCA?

Corporate transfers are an exemption in the PSD2 RTS

Article 17 of the Regulatory Technical Standards (RTS) states that SCA is not required for secure corporate payments as long as the following conditions are met:

- Dedicated payment processes or protocols are used
- The dedicated processes or protocols are only made available to payers who are not consumers
- National competent authorities are satisfied that dedicated corporate processes and protocols are sufficiently secure

Corporate payments are more complex than consumers' payments, so it is not a surprise that they should constitute an exemption. However, the amounts at stake in corporate transfers are of such value that the transfers are a prime target for all the "white-collar" fraudsters of this world.

What does corporate payment look like?

Payments at the corporate level most often involve quite a few people. It starts with the purchasing team onboarding new suppliers and their payments details into the corporation ERPs. The treasurer and their team will then prepare the payment orders to settle invoices. These orders will rely on the supplier's IBAN and company registration number. After preparing the payment order via a corporate portal, banking portal or dedicated corporate payment tool, the treasurer will have to have the payment validated by the CFO, or CEO, Chairman, General secretary etc. depending on the amount and the payment validation process of the company. The actual transfer can then take place.

With thousands or tens of thousands of suppliers, corporate payments can be a headache for corporations and a new bonanza for the white-collar fraudsters. Supplying companies change banks, create subsidiaries, merge etc. There are many occasions that will require the corporation to change the IBAN/registration number couple.

Where are the security threats?

Hackers focus on stealing the identifier and password of key players in the payment process, i.e. treasurers and CFOs. Passwords – even sophisticated ones – can be broken and is for that reason considered a weak point in corporate payment security.

Passwords can be stolen by malware delivered by email, or via social-engineering attacks like phishing. Vulnerabilities linked to hardware and software may also be exploited if upgrade routines are not well in place. Last but not least, the increasing usage of IoT offers new vectors of attack to fraudsters to break into the information systems of corporations.

Where would the fraudsters specifically attack?

Once the fraudsters get hold of the 'keys", they will get direct access to the corporation portal and start to wreak havoc by redirecting payments. As we could see above, many points in the process can be attacked.

One very subtle attack is to change the IBAN in the IBAN/registration couple. The entire finance team believes it is paying the right company while the money is diverted to another account across Europe. In a matter of a few seconds, with instant payment initiatives, the money can be moved out of Europe, with no hope that it will ever be recovered.

Another attack is to trick a supplier into changing the contact information for a corporation. The supplier will then send the invoice to the attacker who can modify it and then pass it on to the corporation. The corporation will then receive an invoice which looks valid except for the payment details. In situations like that, SCA would be useful to validate the identity of a supplier.

Corporations have to up their game

With the amounts at stake and the ingenuity level of fraudsters, corporations just cannot do without strong and innovative security. Although smartphones are not widely used in corporate payments today, they are still an excellent medium to combine possession and inherence factors as required for strong customer authentication. SCA verification can be used at each connection to the enterprise portal, i.e. to the "vault" where IBAN/ registrations numbers are kept. Some corporations may not trust the embedded biometric capabilities of phones and will most likely look for military-grade biometrics solutions as well.

There are security vendors that provide a wide array of biometrics solutions to get rid of passwords on PCs and mobiles alike. That can enhance security at access points to sensitive information. A smartphone can be used as a second factor at an authentication point or used to validate the transaction by the CFO or the CEO when they cannot access the corporate portal via their PCs. However, two factors of authentication might be not enough; security also has to provide malware resistance at the point of validation, especially on a smartphone.

Securing the access point or validation point is a must, but there are other ways to combat corporate payment fraud beyond authentication security. One example of this is a community of finance directors who share IBAN/ registration couple data to combat supplier payment fraud. Artificial intelligence applied to this data and community sharing effect makes it possible to raise alarms on a potential fraudulent data couple.

Corporate payment is, and will continue to be, a prime target for fraudsters. Although it is complex to break into the payment process, the amounts at stake are significant enough for fraudsters to continue their attacks and make the treasurers of this world paranoid about security.

Final remarks

Conclusion

PSD2, RTS and SCA is a comprehensive "package" with many requirements that go beyond multi-factor authentication; MFA is just the tip of the SCA ice-berg visible to end-users. Given the diversity of use-cases generated by SCA, issuers need to analyse their customer base and build an SCA solution that works with the limitations of their different use-cases. That means integrating several solutions or maybe even developing some custom solutions.

User profiles and the use-cases will have an impact on the fallback options an issuer chooses to stay compliant with SCA for all of its users. Whatever the choices, there is an impact on integration costs that issuers will try to mitigate.

When implementing an SCA solution, issuers will have to look out for two specific challenges:

- **Be as independent as possible from the OS**:
  This will enable users with an outdated OS to continue using your app securely.
- **Implement solutions that will be future-proof**:
  Choose solutions that will guard against new innovative attacks, such as malware.

Enrolment and re-enrolment are, while outside of the SCA scope, a crucial part of the authentication process. As an issuer, it is paramount to have a good KYC routine in place.

Last but not least, corporate transactions are a clear target for fraudsters. Although they are an exemption from the RTS - and part of the June 2019 opinion from the EBA - the amounts at stake are such that they require SCA-like security.

SCA for the end-user

When first introduced, there was a lot of discussion in our industry about PSD2 SCA. eCommerce merchants (and their acquirers) were most vocal, fearing that the friction created by SCA would harm basket conversation and generally impact their businesses. With the responsibility shift that came with PSD2, issuers can be tempted to trigger SCA for each transaction. That adds an extra layer of friction for the end-user, so this could be a reasonable fear from merchants.

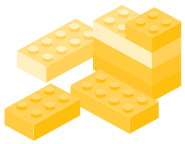
We believe consumers' adverse reaction to this friction is **THE** big challenge to solve. But in our humble opinion, this could be solved by educating consumers. SCA was introduced to protect the consumers and keep the fraudsters of this world at bay, but this is insight most consumers do not have. If we, as an industry, take the responsibility of educating end-users of how the added layer of security protects their values and ID they might not be as reluctant.

We hope this has given you insight into the many SCA challenges issuers and PSPs in Europe will face when engaging in SCA projects. If you believe we missed any key challenges, we'll be very happy to hear from you at hello@okaythis.com.

How does Okay approach these SCA challenges?

Challenge 1 | SCA Security and 2FA

At Okay we are paranoid about security. We assume that there will be an attack on our security; it is just a matter of when. This mindset has been the driving force when building our SCA solution with security mechanisms to protect the authentication process and the transaction.



Challenge 2 | Constraints for a full SCA implementation

We focus not only on the "easy" challenger bank scenario where everyone has a smartphone, but we also talk to banks and financial institutions where a significant percentage of their customers don't have a smartphone.

Challenge 3 | SCA for low-tech phone users and fallbacks

While Okay's core product is a smartphone-based SCA solution, we also provide a solution for authentication and authorization through a voice call to the user.

Challenge 4 | The mobile OS headache

The Okay SCA solution is built with a secure execution environment that aims to be as independent of the device operating system as possible. We assume that the user's device has been infected with malware and that malware is trying to attack the application.

Challenge 5 | Innovative malware attacks

The pace of technology appears to go faster and faster for each year, and the computing power of your mobile phone would most likely have been considered a supercomputer back in the 1990s. At Okay we try to plan and protect against attacks which would have been unthinkable just a few years ago.

Challenge 6 | The cost of SCA integration

The strategy when designing the Okay SCA solution was to reduce costs as much as possible. As a result, we are building our partner network as we strive to take the cost of integration out of the equation for our clients.

Challenge 7 | Enrolment and re-enrolment

At Okay we focus on protecting the process around the SCA challenge itself (e.g. from malware). However, we recognise that enrolment and re-enrolment is a must-have for issuers. To facilitate this need, Okay has decided to strengthen issuers' KYC both through a partnership to check SIM-cards further and by implementing mechanisms to simplify re-enrollment as part of a KYC process.

Challenge 8 | SCA for corporate transactions

We believe that strong, dedicated 2FA solutions combined with our malware resistance mechanisms on smartphones can offer near to "bulletproof" security to corporations whenever authentication is required across different channels - desktops, tablets and smartphones.

okaythis.com
hello@okaythis.com