how to go through fire and water with MobileSCAN

Vladimir KOTOV

no software engineers were harmed
during the production of the product and this presentation

What is MobileSCAN

- Authorization device that lets access Citadele Online Banking services both via a web browser and the Mobile Application
- Publicly announced on May 25, 2015
- Alternative to traditional authorization devices
  - TAN, Digipass, GO3

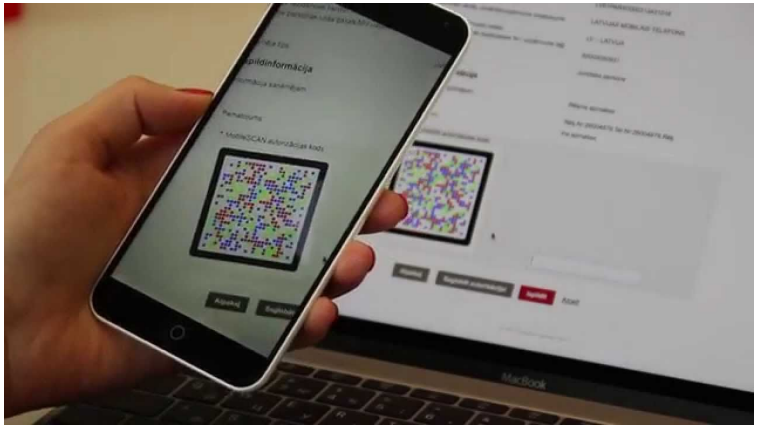
What is MobileSCAN

- Originally utilizes Cronto visual transaction signing solution and DIGIPASS® FOR APPS libraries from Vasco
- Can be activated on any Android or iOS device
- Secured with selected PIN code, Touch ID or Face ID technology

Online Banking experience

In online banking allows clients

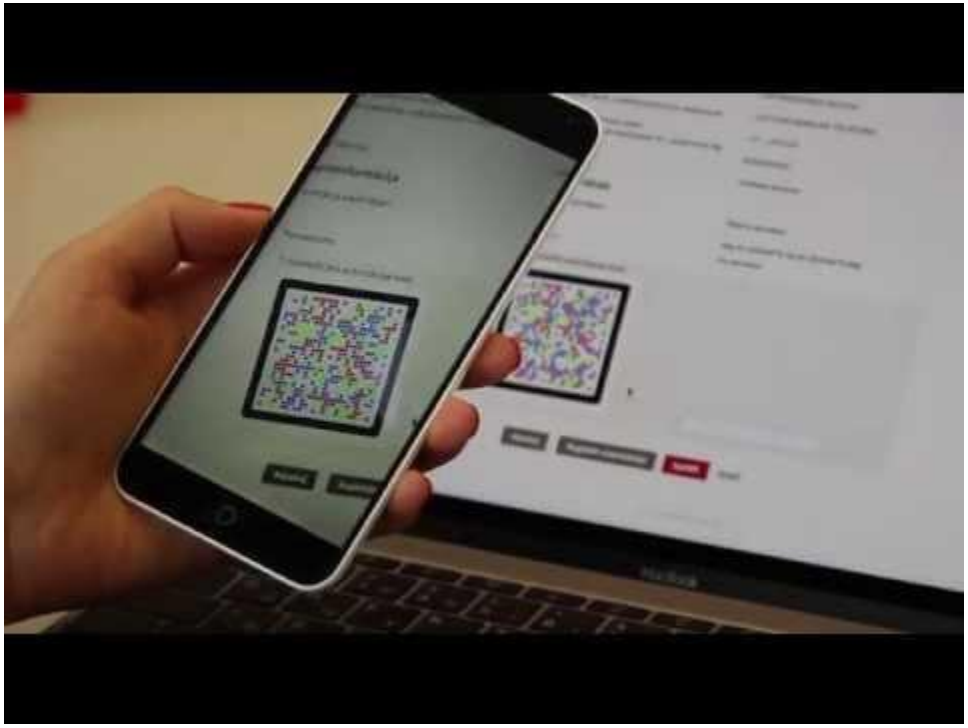
- to login
- to confirm payments and orders
- to authorize via Digi:Link
- to make card transaction



using

- Android or iOS smartphone
- Digipass 780 device

Online Banking experience in 2015



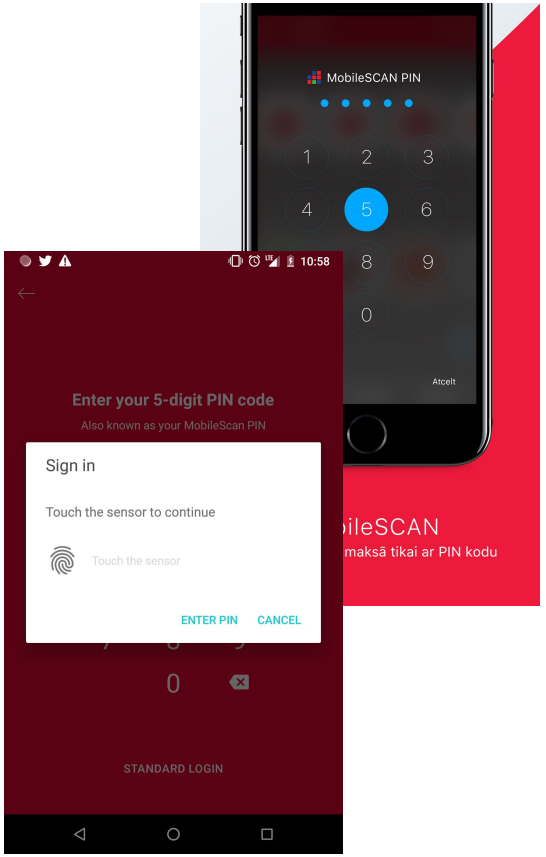
Mobile Application experience

In mobile banking allows clients

- to login
- to confirm payments and orders
- to authorize via Digi:Link

using just 5 digit PIN / Touch ID / Face ID

- no image scanning required



Mobile Application experience in 2016

Business Benefits

- Combat banking threats
  - Phishing attacks
  - Man in the Middle (MitM) and Man in the Browser (MitB)
- Secure high value transactions
  - Additional layer of protection, out-of-band (OOB) validation
- Improve user experience
  - Safety and simplicity to our online and mobile banking users

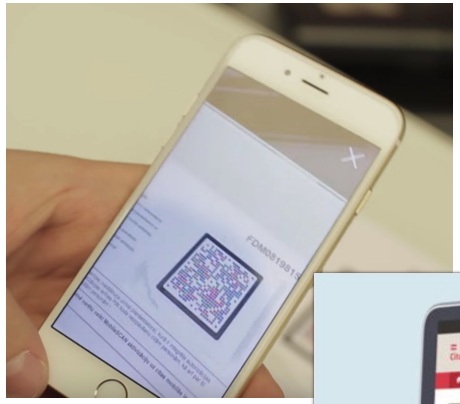
MobileSCAN Evolution

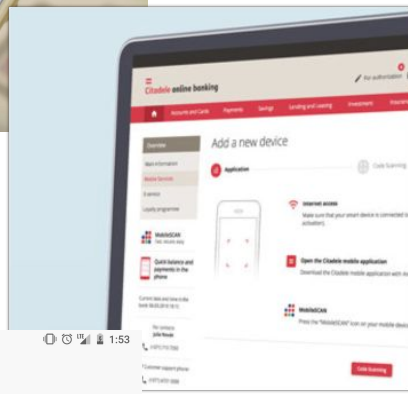
2015 - Activation in branches using printed license activation letters

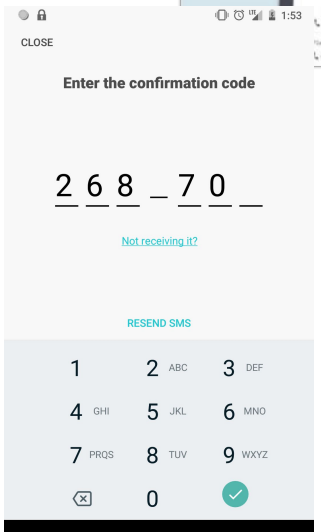
2016 - Device management and MobileSCAN activation in Online Banking

2016 - Quick Authorization with PIN in Mobile application

2018 - MobileSCAN activation via SMS, Touch ID and Face ID support







Sources of Challenges

- ECB: Recommendations for security of Internet Payments
- EBA: Final guidelines for security of Internet Payments
- EU: Payment services directive (PSD2)
- EBA: Regulatory Technical Standards on **strong customer authentication** and secure communication under PSD2

Challenge 1
Strong Customer Authentication

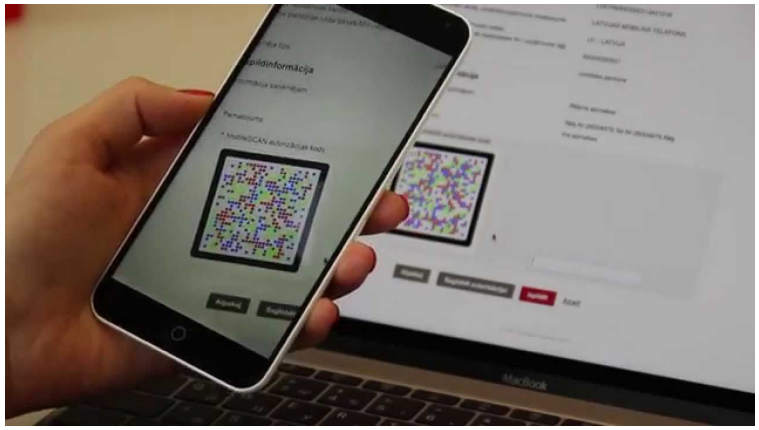
When?

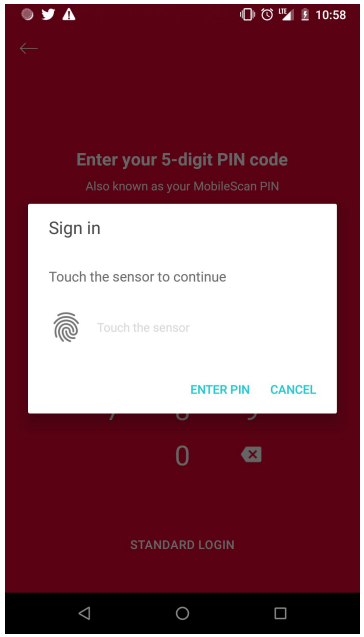
- Payer accesses online payment account
- Payer initiates payment
- Payer carries out action through remote channel that may imply fraud

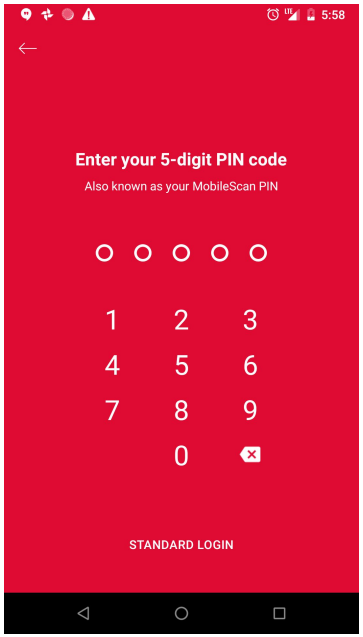
How?

- Generate one-time authentication code using two or more independent elements

Strong Customer Authentication with MobileSCAN







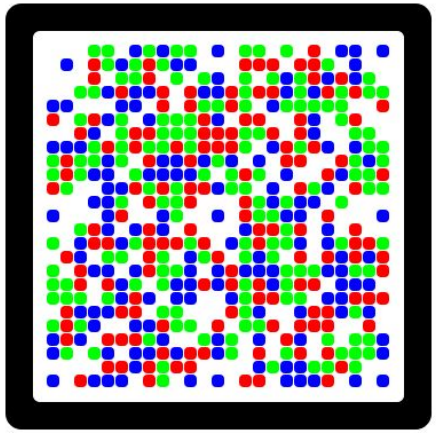
Challenge 2
Dynamic Linking or Transaction Authentication

- For payment authentication, amount and payee need to be included into generation of authentication code
- Confidentiality, authenticity and integrity of the information displayed to the payer and through all phases of authentication

What **Y**ou
See **I**s **W**hat
You **S**ign

Solution
CRONTO code

- Encrypted High-Definition Color QR Code consisting of a matrix of colored dots
- If the image has not been tampered with, the customer is then presented with critical transaction information, decoded securely from the visual cryptogram image
- Effectively counter "Trojan" and "Man in the Browser" attacks

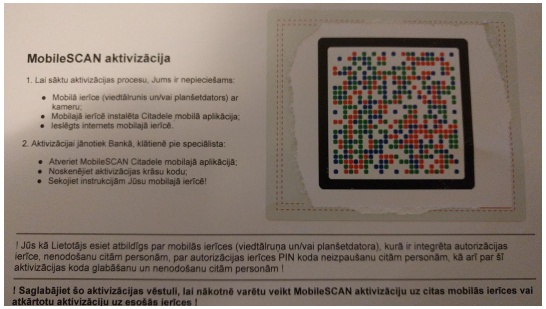


CRONTO and VASCO Secure Channel

- Secure transaction data signing links the authentication code to the transaction key details
- End-to-end protection of communication between user and Bank
- Visual validation feature provides users a quick and convenient check of each transaction
- Omni-channel user experience (Online Banking, Digi:Link, 3DSecure)

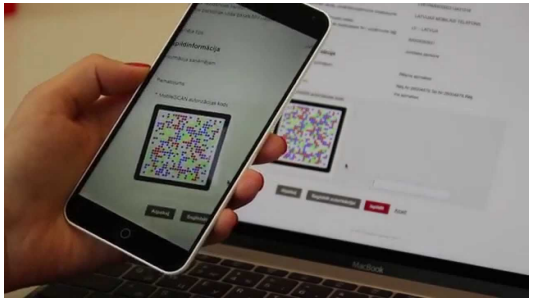
Lessons Learned

- Scanning of Cronto codes in general is more reliable and faster than QR
- Never sacrifice on printer quality
- Cronto codes are scanned better from displays than from paper



Lessons Learned

- Data must be limited to 200 characters if supposed to be scanned by hardware device
- People may do crazy things (e.g. scan Cronto using default Camera application)
- Logging into Online Banking site from mobile device with MobileSCAN is really hard



Challenge 3
Protecting possession element against cloning

The use of elements categorized as possession shall be subject to measures designed to prevent replication of the elements

How to protect mobile apps against cloning?

- Include device-specific data into calculation of authentication code
- Encrypt authentication key using key in Secure Element of device
- Encrypt authentication key using knowledge element (e.g. PIN)

Challenge 4
Mobile device security

Mechanisms to ensure that the software or device has not been altered by the payer or mechanisms to mitigate the consequences of such alteration

- Combination of OS sandboxing and root detection
- Use Runtime Application Self-Protection (RASP)

Solution
Root Check

1. We could probably build our own check …

2. Root Detection SDK
   - Detects if an application is running on a rooted Android device or on a jailbroken iOS device.
   - Detects potentially unsafe devices

Android rooting methods (some)

- *Magisk (systemless root)*
- *SuperSU*
- *KingRoot*
- *TowelRoot*
- *Framaroot*
- *Root master (Root大师)*
- *360 Root (360超级ROOT)*
- *Root Ghost (Root Genius)*
- *...*

Lessons Learned
Mobile device security

- Amount of unsafe Android devices is higher than iOS :)

- Sometimes phone vendors care less about proper security

```
String buildTags = android.os.Build.TAGS;
buildTags.contains("test-keys"); // true
```

The Android tree includes *test-keys* under `build/target/product/security`.

Since the test-keys are publicly known, anybody can sign their own .apk files with the same keys, which may allow them to replace or **hijack system apps built into your OS image**.

It is critical to sign any publicly released or deployed Android OS image with a special set of *release-keys* that only you have access to.

Thank you!

Q&A