Mobile Devices Security

On Practical Risks of NFC Payments

Tomáš Rosa

crypto.hyperlink.cz

Mobile Payments 2012, Prague

Part ONE

So, the NFC Is …

NFC at Glance

- NFC stands for *Near Field Communication*

- Device equipped with an NFC controller can work in the following modes:
  - Passive-mode initiator (or just a “reader”)
  - Passive-mode target (or just a “transponder”)
  - Active-mode initiator/target (or just “reader-to-reader”)

13,56 MHz

NFC vs. RFID

- Correct to say NFC is an inductively coupled communication interface that shares many technical features with HF RFID.
  - This goes such far that NFC devices can directly play the role of certain HF RFID transponders or terminals (readers).
    - Vice versa, some existing HF RFID components can fit the definition of particular NFC operational modes.
    - This is happily abused in marketing leaflets.
  - Of course, NFC also shares the general security properties related to communication interception, wormhole phenomenon, etc.

NFC and EMV-CL / ISO 14443

- NFC-equipped device can address contactless smartcards world in two ways:
  - **As a terminal (“reader”)**
    - ISO 14443 A – passive-mode initiator
  - **As a transponder emulator**
    - ISO 14443 A – passive-mode target
    - This is the mode used in all mobile payment applications discussed here.

NFC In Smart Phone OS
(as of Autumn 2012)

- The most systematic treatment can be found in Google Android.
  - Especially since Ice Cream Sandwich (4.0), but it already started with Gingerbread 2.3.3 [43].
  - Clearly, Google strives to become the leader in this area.
- Also interesting support in some BlackBerry devices (e.g. BB 9900 with BB OS API v7.0.0 [47], [59]).
- Apple seems to wait the see how others will eventually do with NFC [44], [45].
  - This stays true after iPhone 5 disclosure [106].
  - External NFC modules can be attached as accessories to iPhone [46].
    - This should principally work for iPad as well.

Part TWO

Here Comes the Smart Phone

Mobile Payment Application (MPA)

- Runs on the Secure Element (SE)
  - That means on a SIM or a comparable IC.
- Performs client transactions via the EMV contactless protocol
  - Through the NFC controller, MPA appears as a regular EMV contactless payement card to the terminal.
  - Although the application protocol offers (slightly) more scenarios, the HF transport layer stays the same!
    - As this layer has to be compatible with EMV CL [9].
- The main security focus is usually here
  - However, MPA has to rely on the Mobile User Application in some cases [96], [101].

Mobile User Application (MUA)

- Runs on the smart phone application processor
  - That means under iOS, Android, etc.
- Should mainly provide user interface and network connectivity for MPA
- Needs to be a trusted code anyway
  - For instance, it manages entering the PIN (passcode) for MPA.
  - Furthermore, it displays the card details for e.g. internet transactions.

Mobile Cards Wallet (MCW)

- Another smart phone application
  - With possible enhancement on the SE side.
- Solves the problem of having multiple contactless cards “loaded” on the same phone
- So, it should be independent on the particular bank
- However, it shall be independent on the particular mobile network provider as well
  - The smart phone OS is the right place!
  - Apple’s Passbook may serve for an illustration.

Part THREE

Jailbreaking and Rooting
- Cautionary Note & Observation

Jailbreak and Root

- Firmware patching aimed at user privileges escalation.
  - Finally, we can have unauthorized applications running with no sandbox and the root account at their disposal.
- On Android, installing a set-uid binary is usually enough.
  - So the term “rooting” [74].
- On iOS, the situation is considerably more complicated.
  - Achieving root privileges is often just the beginning, since the runtime is still under Apple tight control.
  - So the term “jailbreaking” [94].

2root || !(2root) ? Don’t!

- Running highly sensitive applications on rooted or jailbroken devices shall be avoided.
  - Already rooted or jailbroken device definitely makes the attacker’s job easier.
    - In the same way as it already helps in forensics [74], [83].
    - Furthermore, the runtime protection is almost none [94].
    - As you can already see in our EA sniffing experiments.
  - Sometimes, the attacker can even hope to get an access to memory dumps of sleeping processes.
    - Consider the unlocked screen and the ability to run anything as root with no sandbox…

2root || !(2root) ? Do!

- We shall admit, however, the device can get rooted or jailbroken without user’s incentive.
  - In JailbreakMe tools, for instance, it was enough to point the Mobile Safari at innocent-looking page [87].
  - See also another remote attack announced at EuSecWest Pwn2Own contest this Autumn [112].
- Developers, therefore, shall test their applications on such devices!
  - Just to be able to see their applications from other perspective…
  - From the perspective of the enemy.

iKee Worms Hit Jailbreakers in 2009

- Exploited default root password “alpine” in SSH on jailbroken phones.
- iKee.A was merely a joke of Australian hacker.
  - It offended users by Rick Astley pictures.
- iKee.B from Europe (probably different author) was a regular malware [95].
- The whole community of Jailbreakers is still so big to be an attractive target of tailored attacks.

photo by AFP

What Does It Mean Anyway

- Besides obvious warnings, there is one more thing to add.
- Do you wonder whether smart phone OS security can be broken?
  - You do not need to ask anymore.
- The worldwide verified proof is right here.
  - It is the Jailbreak in itself! [94]

So, Be Careful!

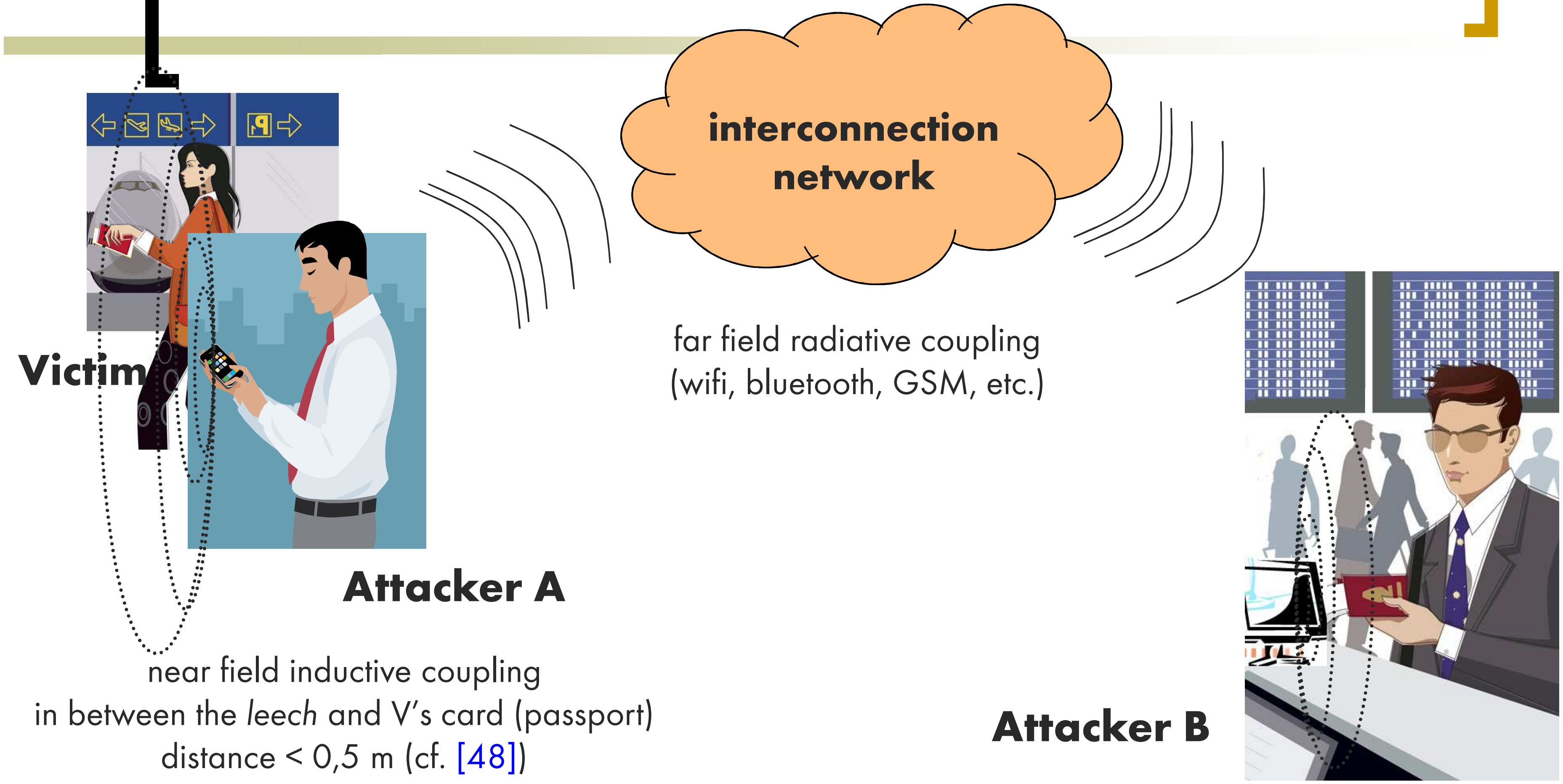
But…

- … what does it mean to “be careful”?
  - Do not participate in pilot projects.
    - Since provisioning profiles open the door for untrusted code execution [94].
  - Avoid Mobile Device Management.
    - Since the mDM server has nearly full control over its enrolled devices [113].
  - Do not visit any untrusted web page.
    - Since web-based exploits are probably never ending story [112].
  - Do not skim untrusted NFC tags.
    - Since this is promising malware vector [107], [111].
  - Et cetera, et cetera, et cetera…

Part FOUR

Attacking Scenarios

Wormhole Attack Illustrated



interconnection network

far field radiative coupling
(wifi, bluetooth, GSM, etc.)

Victim

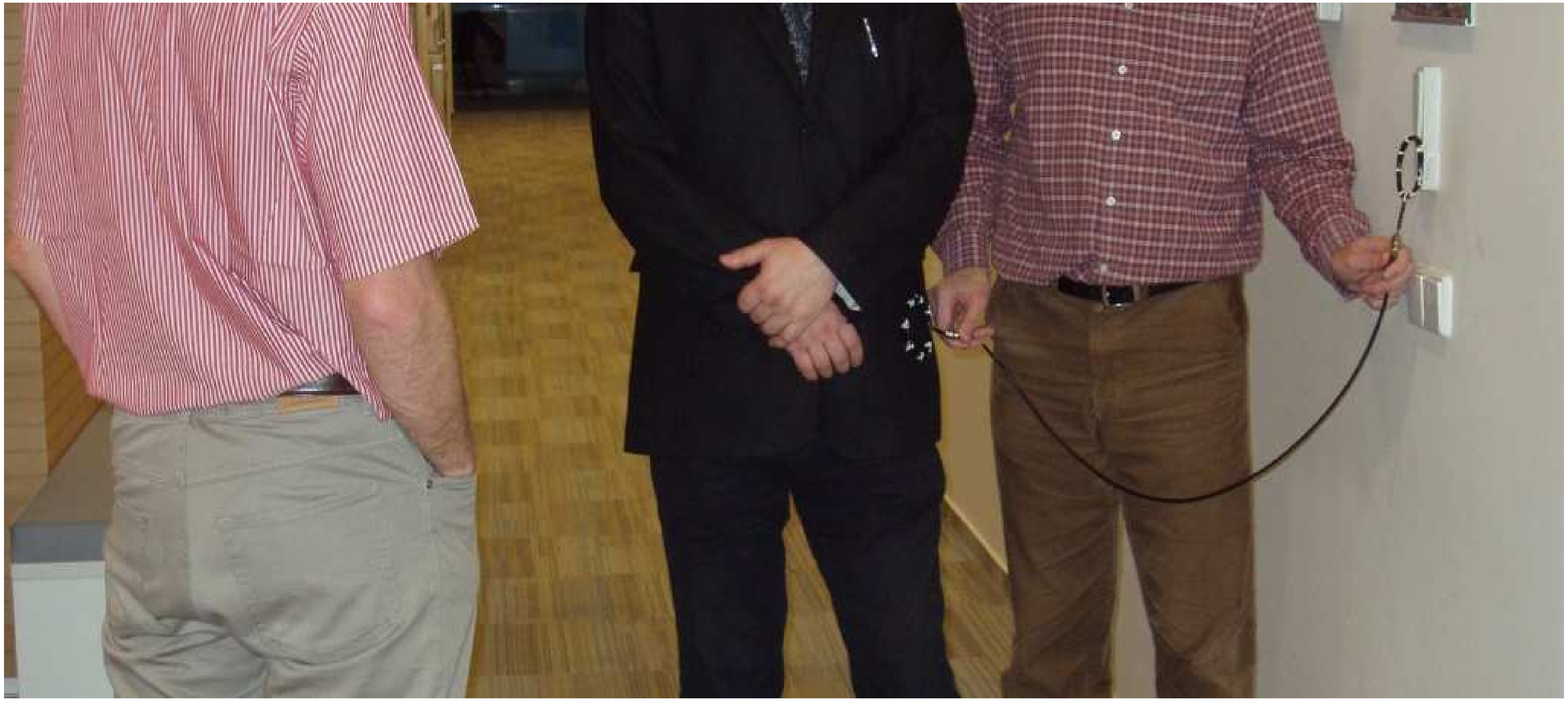
Attacker A

Attacker B

near field inductive coupling
in between the *leech* and V's card (passport)
distance < 0,5 m (cf. [48])

near field inductive coupling
in between the *ghost* and the inspection terminal

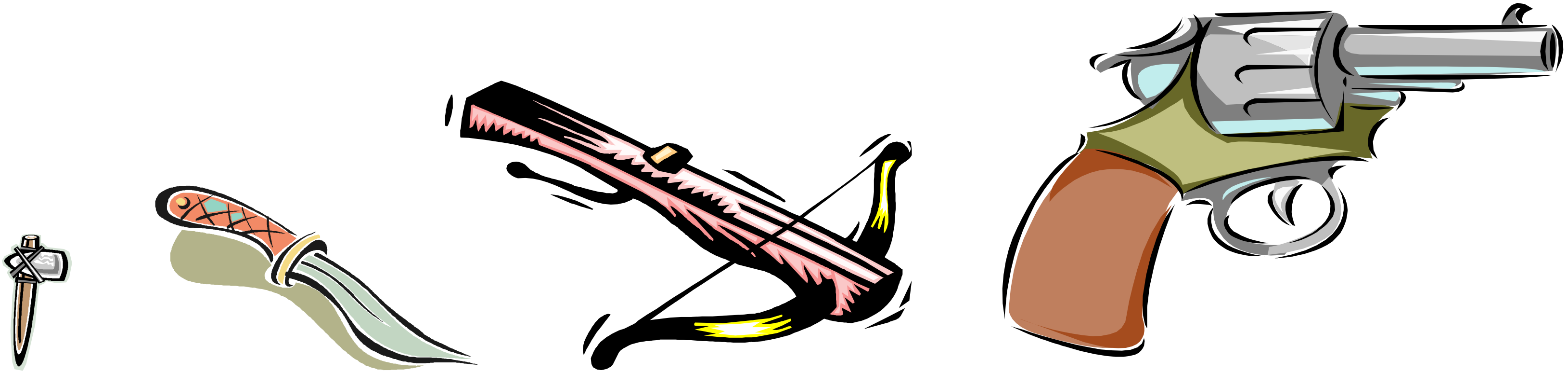
Wormhole In Access Control



Real successful experiment with the DIY wormhole in HF RFID access control.

Threats Do Evolve

- They do not magically appear or disappear.
  - They just follow the technology evolution.



For Instance

- We do not have to empower the mobile phone emulated NFC target.
  - This improves the active communication distance significantly.
- We can require a user action before any NFC activity.
  - This lowers the wormhole attack risk.

Another Example

Faulty NFC Stack

- As a complex networking stack, any NFC implementation itself offers vital hacking surface.
  - Recent study [107] shows this gets further amplified by inappropriate default application actions such as automatically following received URLs, etc…
  - See also [111] for another exploit.
- NFC Forum's quick response [108] talks much about security but it addresses a different topic.
  - Paradoxically, adding a lot of cryptographic protocols to the stack actually makes it more error-prone from the implementation hacking viewpoint...
  - This is not to say we shall omit cryptography.
  - This is to say that implementation security needs another kind of treatment.

Part FIVE

Tweaking iOS Peripherals

OFA Scenario

Definition. *Let the On-the-Fly Attack (OFA) be any attacking scenario that assumes the attacker is able to launch their privileged code running on the user's smart phone transparently during the time the legitimate user performs the authentication procedure.*

- Note that this does not strictly call for having the root account access.
- It is more important to bypass the application sandbox barrier.
  - When we can do that then the "mobile" account on iOS or the respective application UID on Android is usually far enough for the OFA attack.

iOS Peripheral Channels

- They are managed by the External Accessory framework [97], [98].
  - Actually, this is a dynamic library that provides streaming Objective-C interface in between application processes and the operating system drivers.
- Communication with external iPhone NFC controllers is provided this way.
  - In particular, this concerns MPA ↔ MUA communication.
  - Even with iPhone 5, there is still no internal NFC controller available.

EA versus OFA

- Recall that EA is just a dynamic library.
  - It is trivial to write a *tweak* for Jailbroken phone that hooks the relevant library methods [83].
  - The tweak then plays the role of MITM in between the application process and the NFC controller.
- Furthermore the data streams provided by External Accessory framework have no implicit data protection [97].
  - Its is up to the application to eventually devise its own cryptographic protocol.

EA Sniffer

- It started as a simple, purely SW-oriented debugging tool.
  - It is a *tweak* that is automatically injected into EA-based application processes via MobileSubstrate [91].
  - Once injected, it echoes the peripheral communication into the system log.
- From security perspective, however, it is a MITM proof-of-concept for EA under OFA.
  - We show a simple session captured for Redpark C2-DB9 bus converter (iDevice ↔ RS 232).
    - http://www.redpark.com/c2db9.html

Demo: Sniffing Redpark Serial
Initialization Phase

```
Rsc Demo[2437] <Warning>: EASniFF> -[EASession initWithAccessory:forProtocol:]
(@@:@@) hooked successfully, was 0x37538c29 now is 0x211a19

Rsc Demo[2437] <Warning>: EASniFF> -initWithAccessory:forProtocol: dispatched for
EASession<0x00187790>, dropping self for sniffer substitution

Rsc Demo[2437] <Warning>: EASniFF> EASessionSniff<0x00187930>
initWithAccessory:<0x00179fc0> protocolString:com.redpark.hobdb9

Rsc Demo[2437] <Warning>: EASniFF> EAInputStream not hooked yet, hooking now

Rsc Demo[2437] <Warning>: EASniFF> -[EAInputStream read:maxLength:] (l@:^CL)
hooked successfully, was 0x375384dd now is 0x21217d

Rsc Demo[2437] <Warning>: EASniFF> -[EAInputStream getBuffer:length:] (c@:^^C^L)
hooked successfully, was 0x375385ed now is 0x2122f5

Rsc Demo[2437] <Warning>: EASniFF> EAOutputStream not hooked yet, hooking now

Rsc Demo[2437] <Warning>: EASniFF> -[EAOutputStream write:maxLength:] (l@:^CL)
hooked successfully, was 0x37537711 now is 0x211ffd
```

Demo: Sniffing Redpark Serial

Simple Loopback Test

```
Rsc Demo[2437] <Warning>: EASniFF> EAOutputStream<0x0de8b910> wrote 30 B (of 30)
Rsc Demo[2437] <Warning>: EASniFF> <0de8b910> 0000: ab cd 1a 10 48 65 6c 6c 6f 20
45 78 74 65 72 6e | ....Hello Extern
Rsc Demo[2437] <Warning>: EASniFF> <0de8b910> 0010: 61 6c 41 63 63 65 73 73 6f 72
79 21 0d 0a       | alAccessory!..

Rsc Demo[2437] <Warning>: EASniFF> EAInputStream<0x0de8b830> read 20 B
Rsc Demo[2437] <Warning>: EASniFF> <0de8b830> 0000: ab cd 10 10 48 65 6c 6c 6f 20
45 78 74 65 72 6e | ....Hello Extern
Rsc Demo[2437] <Warning>: EASniFF> <0de8b830> 0010: 61 6c 41 63
| alAc

Rsc Demo[2437] <Warning>: EASniFF> EAInputStream<0x0de8b830> read 14 B
Rsc Demo[2437] <Warning>: EASniFF> <0de8b830> 0000: ab cd 0a 10 63 65 73 73 6f 72
79 21 0d 0a       | ....cessory!..
```

Part SIX

PIN on POS vs. PIN on Mobile

PIN on Mobile (PoM)

- Apparently, the PIN can be captured under OFA scenario.
  - Stealth techniques can make this harder, but there is no bullet-proof concept [83].
  - Perhaps, TrustZone will make this better [100].
- On the other hand – we already need PoM anyway.
  - For instance, to access passcode protected data on VISA MPA [101].
  - It really does not matter whether the attacker steals the PIN during cardholder verification or when the user accesses e.g. passcode protected card details.

PIN on POS (PoP)

- We shall admit POS can be compromised as well.
  - There already were convincing proof-of-concept attacks [99], [109].
- As POS installations are growing rapidly, the situation will hardly get better with time.
  - So, it is not wise to assume that PoP is a universally secure approach forever.

PoM or PoP?

- There is no universally best approach.
  - The new threats on PoM do not cancel out existing threats on PoP!
- Probably, we need PoM anyway.
  - There is no better authentication of MUA agent to MPA, now.
  - Recall, the attacker does not care *why* the user enters the PIN as long as they do so.

PoM or PoP?

- There needs to be a risk analysis done on application by application basis.
  - We shall consider supporting both PoM and PoP with no discrimination.
  - Any imbalance introduced then shall be clearly justified.
    - Does it really eliminate the risk?
    - Does it introduce any new threat?
    - What is the total risk in such unbalanced system?
  - We shall not overrate existing user experience!
    - Smart phone applications show clearly that users are eager to adopt new habits just because of their fancy implementation.

Conclusion

- As usual, it is unnecessary to achieve the maximum security ever possible.
  - We shall be just ahead of criminals.
- To keep this margin, we shall mainly pay attention to the smart phone security, now.
  - PIN on POS vs. PIN on Mobile is really a side issue.
  - We need to have a secure computing platform anyway to keep mobile payments safe.

Thank You For Attention

Tomáš Rosa

crypto.hyperlink.cz

References I

(extended)

1. Axelson, J.: *USB Complete: Everything You Need to Develop USB Peripherals*, 3rd Ed., Lakeview Research LLC, 2005
2. Beth, T. and Desmedt, Y.: *Identification Tokens – Or: Solving the Chess Grandmaster Problem*, In Proc. of CRYPTO ’90, pp. 169-176, Springer-Verlag, 1991
3. Brands, S. and Chaum, D.: *Distance-Bounding Protocols,* In Proc. of EUROCRYPT ’93, pp. 344–359, Springer-Verlag, 1994
4. Desmedt, Y.: *Major Security Problems with the ‘Unforgeable’ (Feige)-Fiat-Shamir Proofs of Identity and How to Overcome Them*, SecuriCom ’88, SEDEP Paris, pp. 15-17, 1988
5. Desmedt, Y., Goutier, C., and Bengio, S.: *Special Uses and Abuses of the Fiat-Shamir Passport Protocol*, In Proc. of CRYPTO ’87, pp. 16-20, Springer-Verlag, 1988
6. *Development of a Logical Data Structure – LDS for Optional Capacity Expansion Technologies*, ICAO, ver. 1.7, 2004
7. Dobkin, D.: *The RF in RFID: Passive UHF RFID in Practice*, Elsevier Inc., 2008
8. Drimer, S. and Murdoch, S.-J.: *Relay Attack on Card Payment – Vulnerabilities and Defences*, Conference 24C3, December 2007

References II

(extended)

9. EMV Contactless Specifications for Payment Systems, *EMV Contactless Communication Protocol Specification*, v. 2.2, July 2012
10. Finke, T. and Kelter, H.: *Abhörmöglichkeiten der Kommunikation zwischen Lesegerät und Transponder am Beispiel eines ISO14443-Systems*, BSI - German Federal Office for Information Security, 2005
11. Finkenzeller, K.: *RFID Handbook – Fundamentals and Applications in Contactless Smart Cards and Identification*, John Willey and Sons Ltd., 2003
12. Francillon, A., Danev, B., and Čapkun, S.: *Relay Attacks on Passive Keyless Entry and Start Systems in Modern Cars*, IACR ePrint Report 2010/332, 2010
13. Hancke, G.: *Practical Eavesdropping and Skimming Attacks on High-Frequency RFID Tokens*, Journal of Computer Security, accepted to be published 2010
14. Hancke, G.: *Eavesdropping Attacks on High-Frequency RFID Tokens*, 4th Workshop on RFID Security (RFIDSec), July 2008
15. Hancke, G.: *Practical Attacks on Proximity Identification Systems (Short Paper)*, In Proc. of IEEE Symposium on Security and Privacy, pp. 328-333, 2006
16. Hancke, G.-P.: *A Practical Relay Attack on ISO 14443 Proximity Cards*, Tech. Report, 2005

References III

(extended)

17. Hancke, G.: *Research Homepage*, http://www.rfidblog.org.uk/research.html
18. Hancke, G.-P. and Kuhn, M.-G.: *An RFID Distance Bounding Protocol*, In SecureComm ’05, pp. 67-73, IEEE Computer Society, 2005
19. Hlaváč, M. and Rosa, T.: *A Note on the Relay Attacks on e-passports: The Case of Czech e-passports*, IACR ePrint Report 2007/244, 2007
20. ICAO - International Civil Aviation Organization, http://www.icao.int/
21. *Identity Theft - MIFARE Campus Card Skimming Attack (EN titles)*, http://www.youtube.com/watch?v=NW3RGbQTLhE
22. *Identity Theft - Prague Citizen Card Skimming Attack (CZ titles)*, http://www.youtube.com/watch?v=Yxvy_eGK5r4
23. Jelínek, L.: *Jádro systému Linux - Kompletní průvodce programátora*, Computer Press, a.s., Brno 2008
24. Kasper, T.: *Embedded Security Analysis of RFID Devices*, Diploma Thesis, Ruhr-University Bochum, July 2006

References IV

(extended)

25. Kfir, Z. and Wool, A.: *Picking Virtual Pockets using Relay Attacks on Contactless Smartcard Systems*, IACR ePrint Report 2005/052, 2005
26. Kirschenbaum, I. and Wool, A.: *How to Build a Low-Cost, Extended-Range RFID Skimmer*, USENIX 2006
27. Lee, Y.: *Antenna Circuit Design for RFID Applications*, Application Note 710, Microchip Tech. Inc., 2003
28. libnfc.org - Public platform independent Near Field Communication (NFC) library, www.libnfc.org
29. Long range HF RFID demonstrator – DEMO90121LR, Melexis, http://www.melexis.com/General/General/DEMO90121LR_662.aspx
30. Menezes, A.-J., van Oorschot, P.-C., and Vanstone, S.-A.: *Handbook of Applied Cryptography*, CRC Press, 1996
31. Myslík, J.: *Elektromagnetické pole - základy teorie*, BEN - technická literatura, Praha 1998
32. *Overview of Technical NFC Documents,* includes PN53x documentation catalogue, NXP, March 2009, http://www.nxp.com/documents/other/nfc_documentation_overview.pdf

References V

(extended)

33. *PKI for Machine Readable Travel Documents offering ICC Read-Only Access*, IACO, ver. 1.1, 2004
34. *S2C Interface for NFC*, Survey VI.0, Philips, 2005
35. PC/SC Workgroup Specifications, http://www.pcscworkgroup.com/specifications/overview.php
36. Rosa, T.: *PicNic - Yet Another Emulator/Spyware for HF RFID*, technical project 2008 – 2010, http://crypto.hyperlink.cz/picnic.htm
37. Rosa, T.: *SCL3710 USB Dongle Config-based SHORT-CIRCUIT Found*, libnfc developers forum, 2010, http://www.libnfc.org/community/topic/194/scl3710-usb-dongle-configbased-shortcircuit-found/
38. Rosa, T.: *Passive Target Mode Initialization *Without* Secondary Reader*, libnfc developers forum, 2010, http://www.libnfc.org/community/topic/200/passive-target-mode-initialization-without-secondary-reader/
39. Vinculum-I device datasheet, application notes, drivers, and prototyping boards, http://www.ftdichip.com
40. Weiss, M.: *Performing Relay Attacks on ISO 14443 Contactless Smart Cards using NFC Mobile Equipment*, Master’s Thesis in Computer Science, Fakultät Für Informatik, Der Technischen Universität München, May 2010
41. Fleisch, D.: *A Student’s Guide to Maxwell’s Equations*, Cambridge University Press, New York 2008.

References VI

(extended)

42. Pelly, N. and Hamilton, J.: *How to NFC*, Google I/O 2011, http://developer.android.com/videos/index.html#v=49L7z3rxz4Q
43. http://developer.android.com/guide/topics/nfc/index.html
44. Ankeny, J.: *Apple forgoes NFC m-payment integration with new iOS 5*, October 4, 2011, http://www.fiercemobilecontent.com/story/apple-forgoes-nfc-m-payment-integration-new-ios-5/2011-10-04
45. Evans, J.: *NFC: How Apple's iPhone gains on 'Google Wallet' plan*, October 26, 2011, http://blogs.computerworld.com/19162/nfc_how_apples_using_google_for_the_iphone_wallet
46. http://www.icarte.ca/
47. Francis, L., Hancke, G., Mayes, K., and Markantonakis, K.: *Practical Relay Attack on Contactless Transactions by Using NFC Mobile Phones*, Cryptology ePrint Archive: Report 2011/618
48. Rosa, T.: *RFID Wormholes – the Case of Contactless Smart Cards*, SmartCard Forum 2011
49. http://crypto.hyperlink.cz/cryptoprax.htm

References VII

(extended)

50. Courtois, N.-T.: *The Dark Side of Security by Obscurity and Cloning MiFare Classic Rail and Building Passes Anywhere, Anytime*, rev. May 2009, http://eprint.iacr.org/2009/137
51. Garcia, F.-D., et al.: *Dismantling MIFARE Classic*, ESORICS 2008, pp. 97-114, 2008
52. Garcia, F.-D., et al.: *Wirelessly Pickpocketing a Mifare Classic Card*, IEEE S&P 09, May 2009
53. MIFARE MF1 IC S50, Philips Semiconductors, Rev. 5.1, May 2005
54. Nohl, K., et al.: *Reverse-Engineering a Cryptographic RFID Tag*, USENIX 2008
55. http://code.google.com/p/crapto1/
56. Specification Q5B – ASIC for RFID, SID TAG Switzerland, SOKYMAT s.a., 2001
57. http://www.nfc-forum.org/specs/
58. Felt, A.-P., Finifter, M., Chin, E., Hanna, S., and Wagner, D.: *A Survey of Mobile Malware in the Wild*, SPSM’11, October 17, 2011
59. http://www.blackberry.com/developers/docs/7.0.0api/

References VIII

(extended)

60. Bachman, J.: *iOS Applications Reverse Engineering*, Swiss Cyber Storm, 2011
61. Bédrune, J.-B. and Sigwald, J.: *iPhone Data Protection in Depth*, HITB Amsterdam, 2011
62. Blazakis, D.: *The Apple Sandbox*, Black Hat DC, 2011
63. Breeuwsma, M.-F., de Jongh, M., Klaver, C., van der Knijff, R., and Roeloffs, M.: *Forensic Data Recovery from Flash Memory*, Small Scale Digital Device Forensics Journal, Vol. 1, No. 1, June 2007
64. Breeuwsma, M.-F.: *Forensic Imaging of Embedded Systems Using JTAG (boundary-scan)*, Digital Investigation 3, pp. 32 - 42, 2006
65. Chin, E., Felt, A.-P., Greenwood, K., and Wagner, D.: *Analyzing Inter-Application Communication in Android*, MobiSys'11, 2011
66. Dhanjani, N.: *New Age Application Attacks Against Apple's iOS (and Countermeasures)*, Black Hat Barcelona, 2011
67. Dubuisson, O.: *ASN.1 - Communication Between Heterogeneous Systems*, Morgan Kaufmann Academic Press, 2001
68. Enck, W., Octeau, D., McDaniel, P., and Chaudhuri, S.: *A Study of Android Application Security*, Proc. of the 20th USENIX Security Symposium, 2011
69. Fairbanks, K.-D., Lee, C.-P., and Owen III, H.-L.: *Forensics Implications of Ext4*, Proc. of the Sixth Annual Workshop on Cyber Security and Information Intelligence Research, ACM, 2010

References IX

(extended)

70. Felt, A.-P., Finifter, M., Chin, E., Hanna, S., and Wagner, D.: *A Survey of Mobile Malware in the Wild*, SPSM'11, 2011
71. Halbronn, C. and Sigwald, J.: *iPhone Security Model & Vulnerabilities*, HITB KL, 2010
72. Hay, R. and Amit, Y.: *Android Browser Cross-Application Scripting*, CVE-2011-2357, IBM Rational Application Security Research Group, 2011
73. Heider, J. and Boll, M.: *Lost iPhone? Lost Passwords!*, Fraunhofer SIT Report, cf. also [82], 2011
74. Hoog, A.: *Android Forensics – Investigation, Analysis and Mobile Security for Google Android*, Elsevier, 2011
75. HOTP: *An HMAC-Based One-Time Password Algorithm*, RFC 4226, 2005
76. Jaden and Pod2G: *How To: Install GNU Debugger (GDB) On The iOS 5 Firmware Generation*, iJailbreak, February 24, 2012, http://www.ijailbreak.com/cydia/how-to-install-gnu-debugger-gdb-on-ios-5/
77. Menezes, A.-J., van Oorschot, P.-C., and Vanstone, S.-A.: *Handbook of Applied Cryptography*, CRC Press, 1996
78. Miller, C. and Iozzo, V.: *Fun and Games with Mac OS X and iPhone Payloads*, Black Hat Europe, 2009
79. Miller, C. and Zovi, D.-A.-D.: *The Mac Hacker's Handbook*, Wiley Publishing, Inc., 2009

References X

(extended)

80. Oudot, L.: *Planting and Extracting Sensitive Data Form Your iPhone's Subconscious*, HITB Amsterdam, 2011
81. PKCS #1 v2.1: *RSA Cryptography Standard*, RSA Laboratories, June 14, 2002
82. Toomey, P.: *"Researchers Steal iPhone Passwords In 6 Minutes" - True, But Not the Whole Story*, Security Blog, http://labs.neohapsis.com/2011/02/28/researchers-steal-iphone-passwords-in-6-minutes-true-but-not-the-whole-story/ , 2011
83. Zdziarski, J.: *Hacking and Securing iOS Applications*, O'Reilly Media, January 25, 2012
84. Zovi, D.-A.-D.: *Apple iOS 4 Security Evaluation*, Black Hat USA, 2011

References XI

(extended)

85. http://developer.android.com
86. http://developer.apple.com
87. http://theiphonewiki.com
88. http://thomascannon.net/blog/2011/02/android-lock-screen-bypass/
89. http://www.bbc.co.uk/news/technology-15635408
90. http://www.cycript.org
91. http://www.iphonedevwiki.net
92. http://nakedsecurity.sophos.com/2009/11/08/iphone-worm-discovered-wallpaper-rick-astley-photo/

93. *iOS App Programming Guide*, Apple Developer Guide, Apple Inc., 2011

94. Miller, C., Blazakis, D., Zovi,D.-D., Esser, S., Iozzo, V., and Weinmann, R.-P.: *iOS Hacker's Handbook*, Wiley, May 8, 2012

References XII

(extended)

95. Porras, P., Saidi, H., and Yegneswaran, V.: *An Analysis of the iKee.B (Duh) iPhone Botnet*, Computer Science Laboratory, SRI International, December 2009, http://mtc.sri.com/iphone/

96. *Mobile MasterCard PayPass M/Chip*, Issuer Implementation Guide, 30 December 2011, MasterCard Worldwide

97. *External Accessory Programming Topics*, Apple Developer Guide, Apple Inc., 2011

98. Maskrey, K.: *Building iPhone OS Accessories – Use the iPhone Accessories API to Control and Monitor Device*, Apress, 2010

99. Drimer, S., Murdoch, S.-J., and Anderson, R.: *Security Failures in Smart Card Payment Systems: Tampering the Tamper-Proof*, 25C3, December 27-30, Berlin, Germany, 2008

100. Rosa, T.: *The Decline and Dawn of Two-Factor Authentication on Smart Phones*, Information Security Summit 2012, http://crypto.hyperlink.cz/

References XIII

(extended)

101. *Mobile Application enabled for VISA payWave (MAV) – Requirements and recommendations*, version 1.0, VISA, March 2012

102. Pfeiffer, F., Finkenzeller, K., and Biebl, E.: *Theoretical Limits of ISO/IEC 14443 type A RFID Eavesdropping Attacks*, Smart SysTech 2012 - European Conference on Smart Objects, Systems and Technologies, Munich, June 2012

103. Finkenzeller, K., Pfeiffer, F., and Biebl, E.: *Range Extension of an ISO/IEC 14443 type A RFID System with Actively Emulating Load Modulation*, RFID-Systech, Dresden, May 2011

104. *ARM Security Technology - Building a Secure System using TrustZone Technology*, whitepaper, ARM Limited, 2009

105. http://www.arm.com/products/processors/technologies/trustzone.php, retrieved Sep 12th, 2012

106. Burns, C.: *Why the iPhone 5 needs no NFC, wireless charging, or localized haptic feedback*, Sep 12th, 2012, http://www.slashgear.com/why-the-iphone-5-needs-no-nfc-wireless-charging-or-localized-haptic-feedback-12247301/

107. Miller, C.: *Exploring the NFC Attack Surface*, August 13th, 2012, (a white paper for BlackHat presentation "*Don't Stand So Close to Me*" on July 25, 2012)

References XIV

(extended)

108. Clark, S.: *Forum responds to Black Hat presentation on NFC vulnerabilities*, August 1, 2012, http://www.nfcworld.com/2012/08/01/317100/forum-responds-to-black-hat-presentation-on-nfc-vulnerabilities/

109. Barisani, A., Bianco, D., Laurie, A., and Franken, Z.: *Chip & PIN is definitely broken - Credit Card skimming and PIN harvesting in an EMV world*, Hack In The Box, Amsterdam, 2011

110. Bond, M., Choudary, O., Murdoch, S.-J., Skorobogatov, S., and Anderson, R.: *Chip and Skim: cloning EMV cards with the pre-play attack*, announced at CHES 2012

111. *Android NFC Stack Vulnerability*, Pwn2Own at EuSecWest 2012, http://labs.mwrinfosecurity.com/blog/2012/09/19/mobile-pwn2own-at-eusecwest-2012/

112. *iOS WebKit Vulnerability*, Pwn2Own at EuSecWest 2012, https://www.certifiedsecure.com/news/72

113. Schuetz, D.: *The iOS MDM Protocol*, August 3th, 2011 (a whitepaper for BlackHat presentation “*Inside Apple’s MDM Black Box*”)