

Security Analysis of Mobile Point-of-Sale Terminals

Mahshid Mehr Nezhad, Elliot Laidlaw, Feng Hao
University of Warwick, UK

Network and System Security 2023



Introduction

Payment Systems:

- Card Present (CP)
- Card Not Present (CNP)

CP Acceptance Terminals:

- Traditionally: Point of Sale (PoS)
- Recently: mobile PoS (mPoS)



mPoS
Terminals

Risks

Ecosystem

Related Work

mPoS Terminals

mobile PoS Terminals: small, compact, low-cost, wireless, easy to configure

Accept various payment methods: Contact, Contactless, QR Code

Accpet various devices: card, mobile, watch, wearables



Introduction

Payment Systems:

- Card Present (CP)
- Card Not Present (CNP)

CP Acceptance Terminals:

- Traditionally: Point of Sale (PoS)
- Recently: mobile PoS (mPoS)



mPoS
Terminals

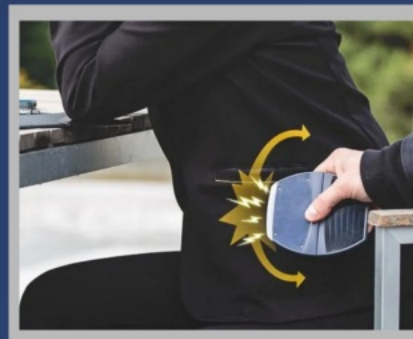
Risks

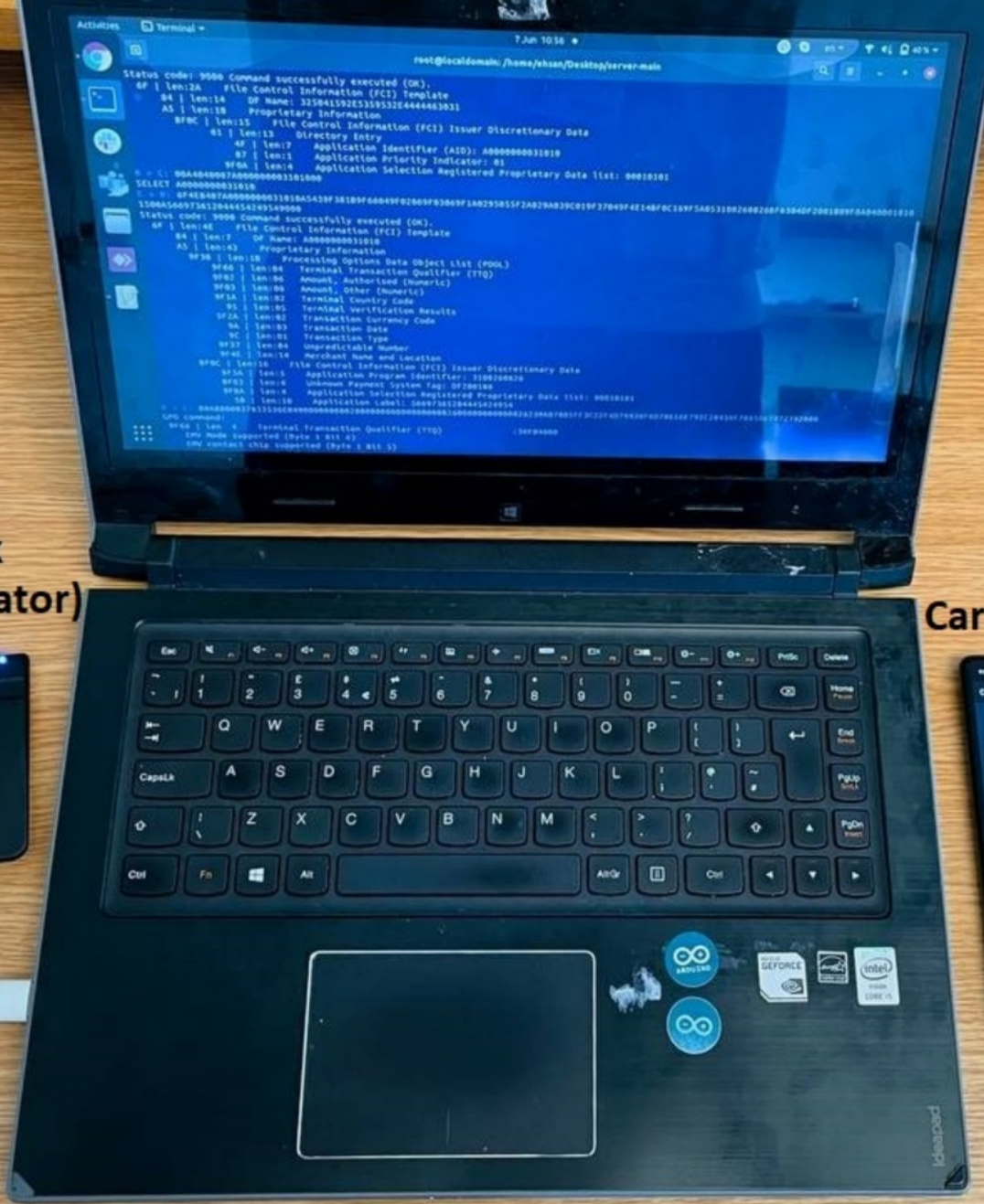
Ecosystem

Related Work

Some Potential Risks

- Lock-screen bypass for mobile payments [22]
- PIN bypass for over the contactless limit [3-5]
- and ...
- Relay (Digital Pickpocketing) [17]





Proxmark
(Terminal Emulator)

Card Emulator

Merchant Phone



Victim's Phone



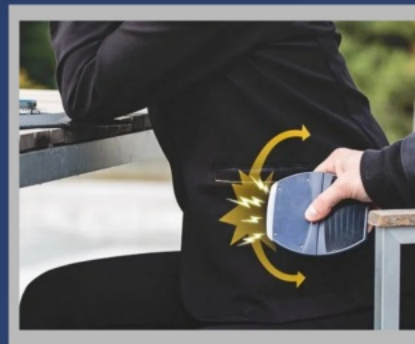
Terminal

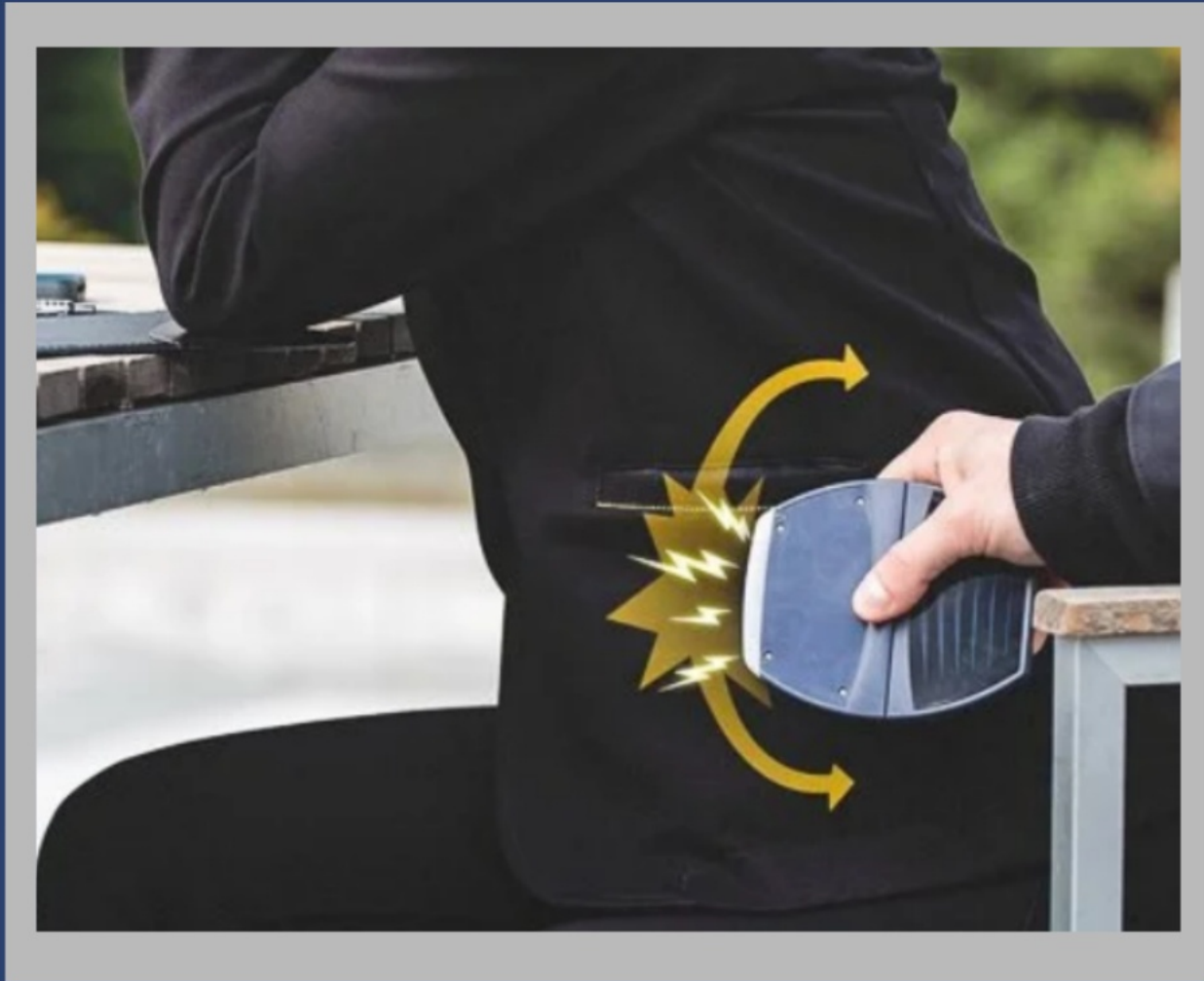


Proxy Server

Some Potential Risks

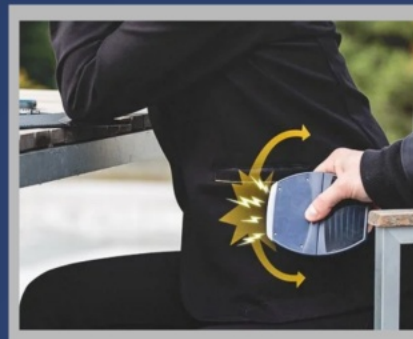
- Lock-screen bypass for mobile payments [22]
- PIN bypass for over the contactless limit [3-5]
- and ...
- Relay (Digital Pickpocketing) [17]





Some Potential Risks

- Lock-screen bypass for mobile payments [22]
- PIN bypass for over the contactless limit [3-5]
- and ...
- Relay (Digital Pickpocketing) [17]



Introduction

Payment Systems:

- Card Present (CP)
- Card Not Present (CNP)

CP Acceptance Terminals:

- Traditionally: Point of Sale (PoS)
- Recently: mobile PoS (mPoS)



mPoS
Terminals

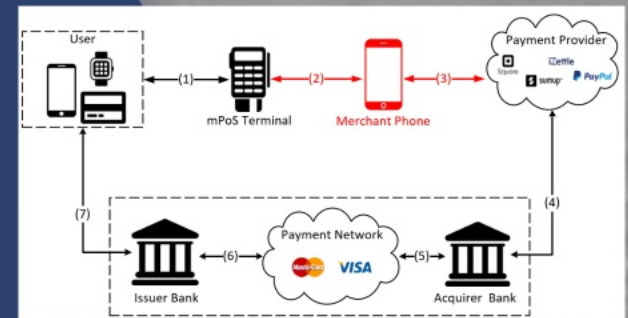
Risks

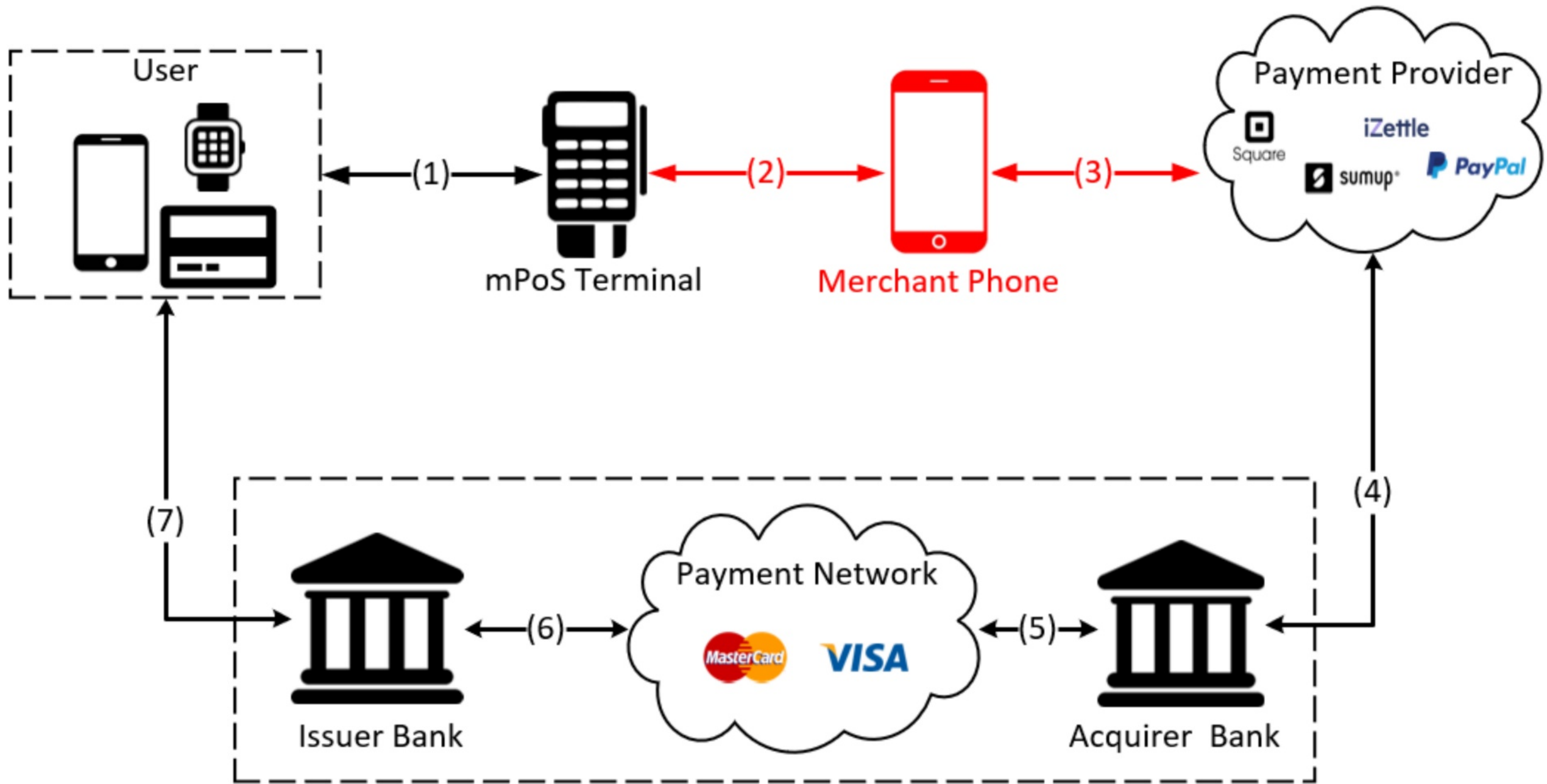
Ecosystem

Related Work

Ecosystem

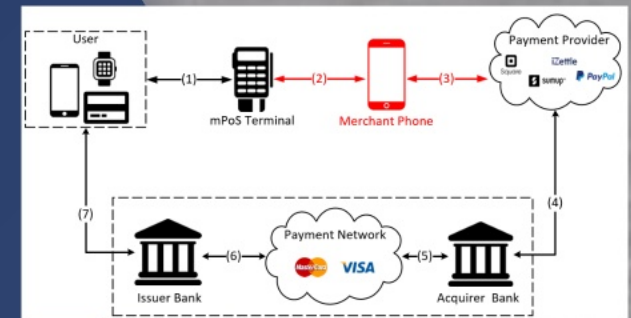
- **Crucial Component: Mobile Phone**
- **Roles:**
 - Communication with mPoS terminal
 - Connection to payment provider
 - Mobile Application
- Proof of concept: **SumUP**





Ecosystem

- **Crucial Component: Mobile Phone**
- **Roles:**
 - Communication with mPoS terminal
 - Connection to payment provider
 - Mobile Application
- Proof of concept: **SumUP**



Introduction

Payment Systems:

- Card Present (CP)
- Card Not Present (CNP)

CP Acceptance Terminals:

- Traditionally: Point of Sale (PoS)
- Recently: mobile PoS (mPoS)



mPoS
Terminals

Risks

Ecosystem

Related Work

Related Work

- | | |
|------|--|
| 2012 | Frisby et. al. [10]: disable the magnetic stripe reader in audio-jack magnetic stripe reader (AMSR) by arbitrary application running and obtain cryptographic keys |
| 2014 | MWR Lab [15]: utilize USB and Bluetooth interfaces, get root access, 1) execute arbitrary command 2) full control over screen ("Try again") |
| 2015 | Mellen et. al. [18]: bypass the encryption by crushing the encryption chip, recording unencrypted swipes and transmit the credit card information to an external server |
| 2018 | Galloway and Yunusov [11]: exploit BLE interface, send arbitrary commands ("please swipe card") and tamper with amounts (Sumup transmitted commands in plaintext!) |

Introduction

Payment Systems:

- Card Present (CP)
- Card Not Present (CNP)

CP Acceptance Terminals:

- Traditionally: Point of Sale (PoS)
- Recently: mobile PoS (mPoS)



mPoS
Terminals

Risks

Ecosystem

Related Work

Security Analysis of Mobile Point-of-Sale Terminals

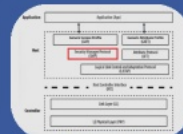
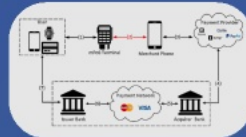
Mahshid Mehr Nezhad, Elliot Laidlaw, Feng Hao
University of Warwick, UK

Network and System Security 2023

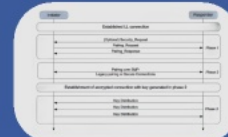


Encryption Security

- Communication: **BLE**
- Protocol **Stack**: Controller, (HCI), Host, Application
- Our interest: **Security Manager Protocol (SMP)**
 - Contains pairing
 - Generates and distributes keys
- Pairing Phases:
 - Phase 1: exchange pairing feature
 - Phase 2: determines pairing mechanism
 - Phase 3: distributes keys



BLE Protocol Stack

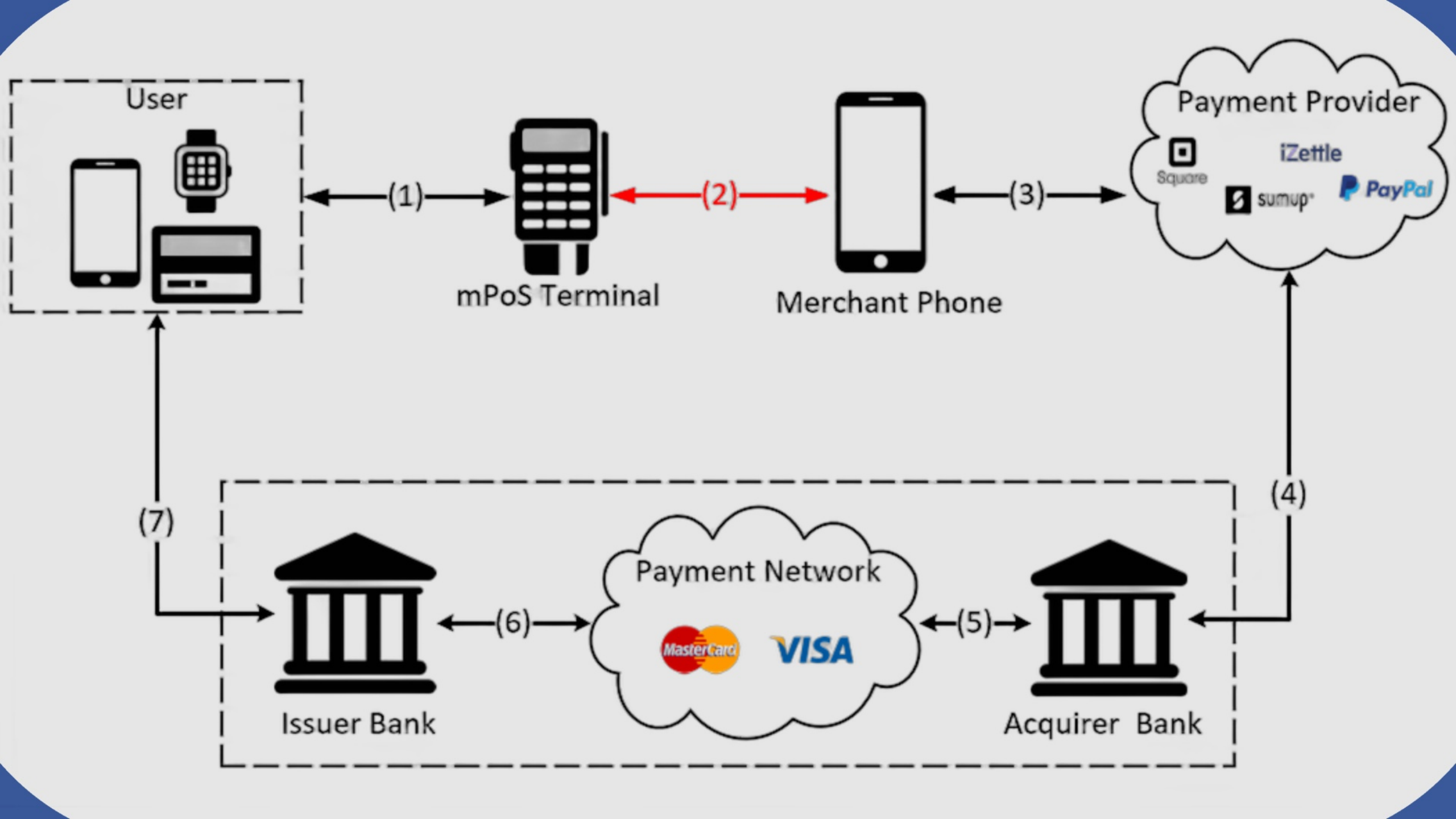


BLE Pairing

Pairing

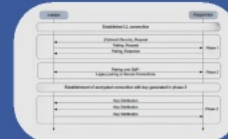
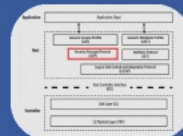
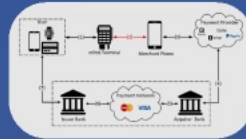
Eavesdropping

Attack



Encryption Security

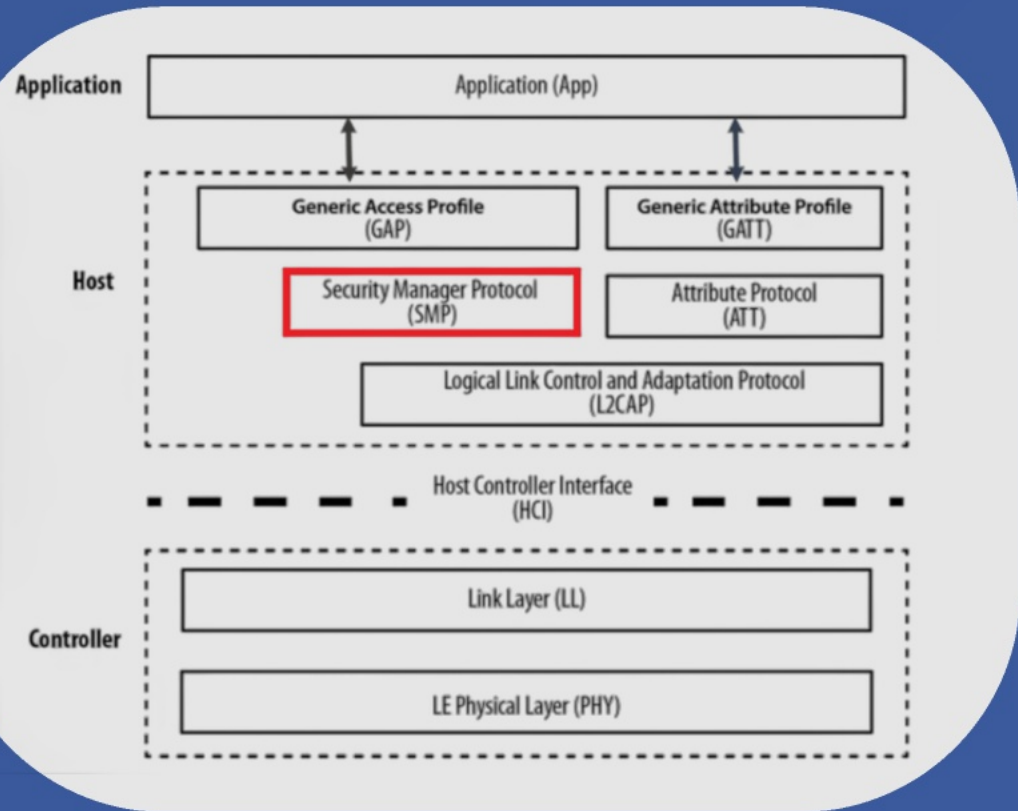
- Communication: **BLE**
- Protocol **Stack**: Controller, (HCI), Host, Application
- Our interest: **Security Manager Protocol (SMP)**
 - Contains pairing
 - Generates and distributes keys
- Pairing Phases:
 - Phase 1: exchange pairing feature
 - Phase 2: determines pairing mechanism
 - Phase 3: distributes keys



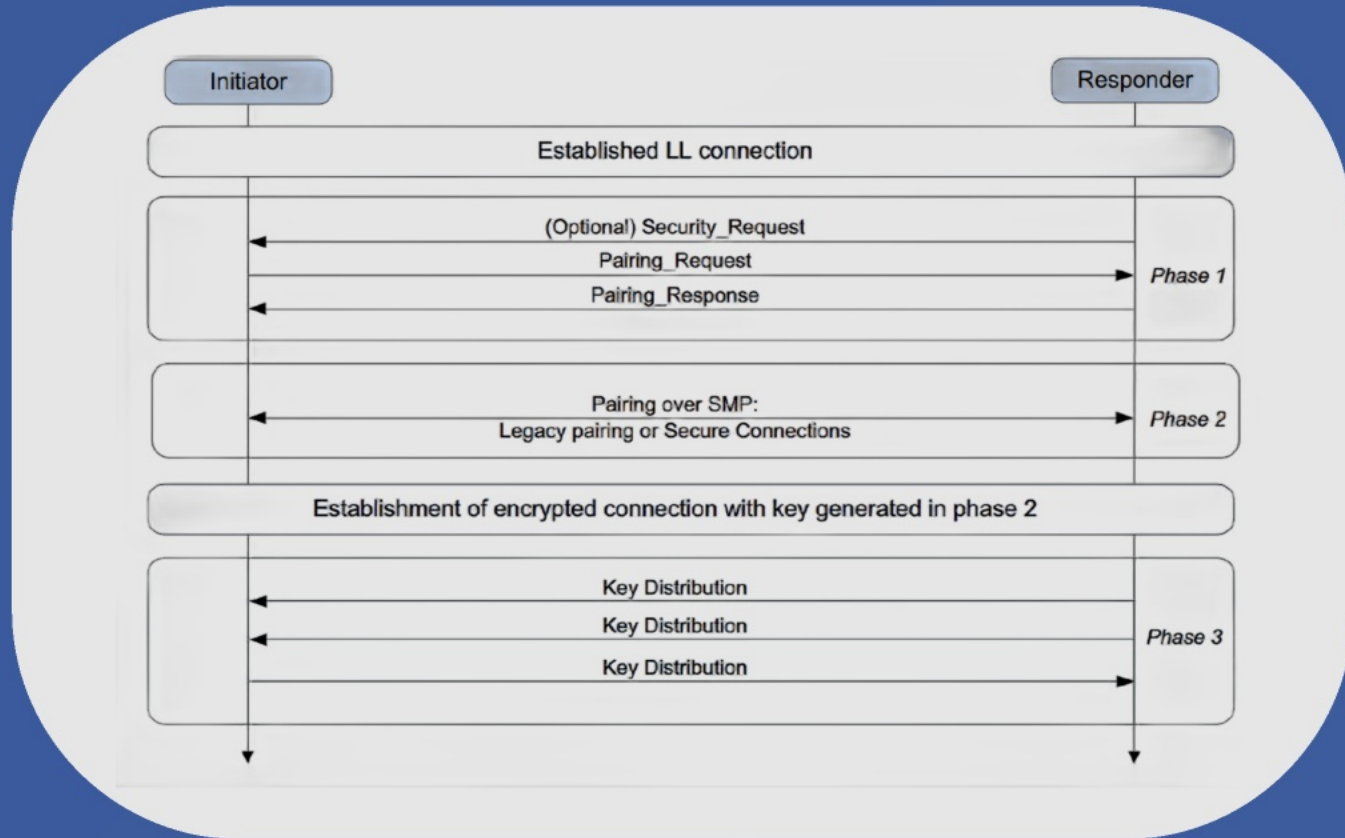
Pairing

Eavesdropping

Attack



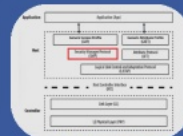
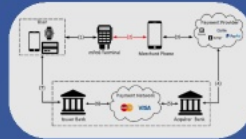
BLE Protocol Stack



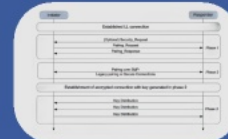
BLE Pairing

Encryption Security

- Communication: **BLE**
- Protocol **Stack**: Controller, (HCI), Host, Application
- Our interest: **Security Manager Protocol (SMP)**
 - Contains pairing
 - Generates and distributes keys
- Pairing Phases:
 - Phase 1: exchange pairing feature
 - Phase 2: determines pairing mechanism
 - Phase 3: distributes keys



BLE Protocol Stack



BLE Pairing

Pairing

Eavesdropping

Attack

Phase 1:

- Pairing Request (I/O, OOB, BF, SC, Key size, ...)
- Pairing Response (I/O, OOB, BF, SC, Key size, ...)

Phase 2:

- Pairing mechanism:
 - Legacy Pairing (TK ==> STK ==> LTK)
 - SC: Secure Connction (ECDH: LTK)
- Pairing method:
 - Just Works Unauthenticated (TK=0)
 - Out of Band (OOB)
 - Passkey (TK: 6 digit)
 - Numeric Comparison

I/O: Input/Output

OOB: Out of Band

BF: Bonding Flag

SC: Secure Connection

TK: Temparory Key

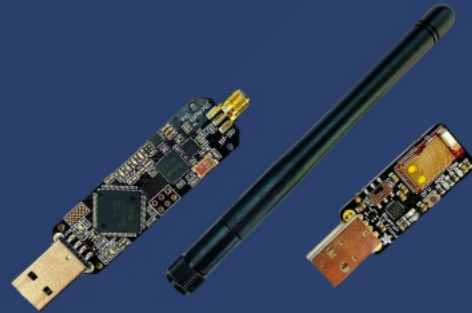
STK: Short Term Key

LTK: Long Term Key

ECDH: Elliptic Curve Diffie–Hellman

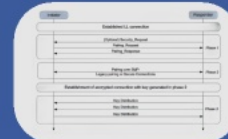
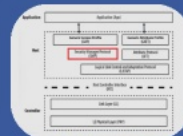
Eavesdropping

- **Threat Model:** malicious merchant or eavesdropper
- **Tools:**
 - HCI Snoop Log
 - BLE Over-the-air Sniffer



Encryption Security

- Communication: **BLE**
- Protocol **Stack**: Controller, (HCI), Host, Application
- Our interest: **Security Manager Protocol (SMP)**
 - Contains pairing
 - Generates and distributes keys
- Pairing Phases:
 - Phase 1: exchange pairing feature
 - Phase 2: determines pairing mechanism
 - Phase 3: distributes keys

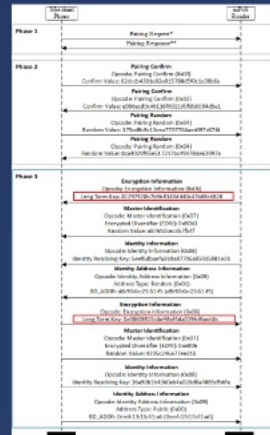


Pairing

Eavesdropping

Attack

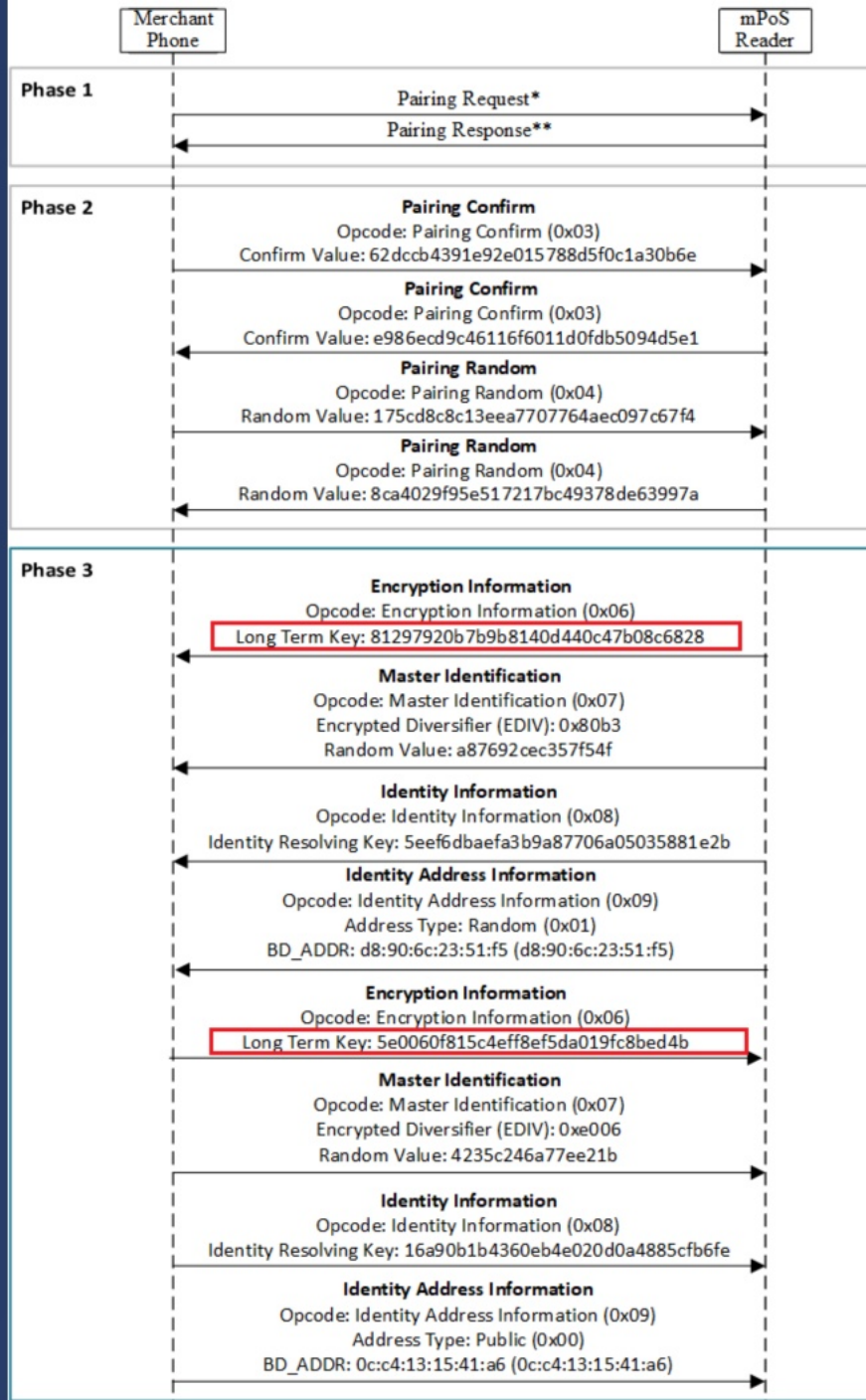
Extract Cryptographic Keys



Field	Pairing Request	Pairing Confirm	Response	Pairing Response	Pairing Response
Code	0x01	0x02	0x03	0x04	0x05
I/O	0x01	0x02	0x03	0x04	0x05
IOE	0x01	0x02	0x03	0x04	0x05
Showing	0x01	0x02	0x03	0x04	0x05
MTSE	1	1	1	1	1
RC	1	1	1	1	1
KP	0	0	0	0	0
Reserved	0x00	0x00	0x00	0x00	0x00
Max Err.	16	16	16	16	16
Initiator Key Distribution					
LTK	1	1	1	1	1
IRK	1	1	1	1	1
CSRK	1	1	1	1	1
Link Key	1	1	1	1	1
Reserved	0x00	0x00	0x00	0x00	0x00
Responder Key Distribution					
LTK	1	1	1	1	1
IRK	1	1	1	1	1
CSRK	1	1	1	1	1
Link Key	1	1	1	1	1
Reserved	0x00	0x00	0x00	0x00	0x00

- Request: Keyboard & Display ~ Response: No I/O
- Pairing: **LE Legacy**
- Key Generation: **Just Works (Unauthenticated)**
- Temporary Key (TK): **Zero**
- Extract **LTK!**
- Crackle: "**Decrypt with LTK**"
 - Input: encrypted file + LTK
 - Output: decrypted file





- Request: Key
- Response: N

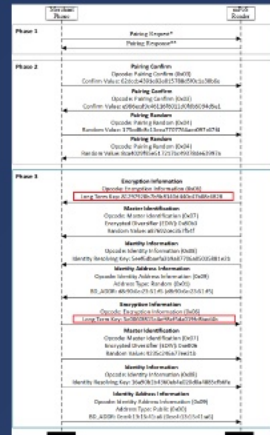
- Pairing

- Key Ge
- (Unaut

- Tempa

Field	Pairing Request Value	Pairing Request Meaning	Pairing Response Value	Pairing Response Meaning
Code	0x01	Pairing Request	0x02	Pairing Response
I/O	0x04	Keyboard/Display	0x03	No I/O
OOB	0x00	NOT Present	0x00	NOT Present
Authentication Request				
Bonding	0x1	Bonding	0x1	Bonding
MITM	1	True	0	False
SC	1	True	0	False
KP	0	False	0	False
Reserved	0x0	-	0x0	-
Max Enc.	16	Max Enc. Size	16	Max Enc. Size
Initiator Key Distribution				
LTK	1	True	1	True
IRK	1	True	1	True
CSRK	1	True	0	False
Link Key	1	True	0	False
Reserved	0x0	-	0x0	-
Responder Key Distribution				
LTK	1	True	1	True
IRK	1	True	1	True
CSRK	1	True	0	False
Link Key	1	True	0	False
Reserved	0x0	-	0x0	-

Extract Cryptographic Keys



Field	Pairing Request	Pairing Confirm	Response	Pairing Response	Pairing Response
Code	0x01	0x02	0x03	0x04	0x05
I/O	0x01	0x02	0x03	0x04	0x05
IOE	0x01	0x02	0x03	0x04	0x05
Showing	0x01	0x02	0x03	0x04	0x05
MTSE	1	1	1	1	1
RC	1	1	1	1	1
KP	0	0	0	0	0
Reserved	0x00	0x00	0x00	0x00	0x00
Max Err.	0x00	0x00	0x00	0x00	0x00
Initiator Key Distribution					
LTK	1	1	1	1	1
IRK	1	1	1	1	1
CRK	1	1	1	1	1
Link Key	1	1	1	1	1
Reserved	0x00	0x00	0x00	0x00	0x00
Responder Key Distribution					
LTK	1	1	1	1	1
IRK	1	1	1	1	1
CRK	1	1	1	1	1
Link Key	1	1	1	1	1
Reserved	0x00	0x00	0x00	0x00	0x00

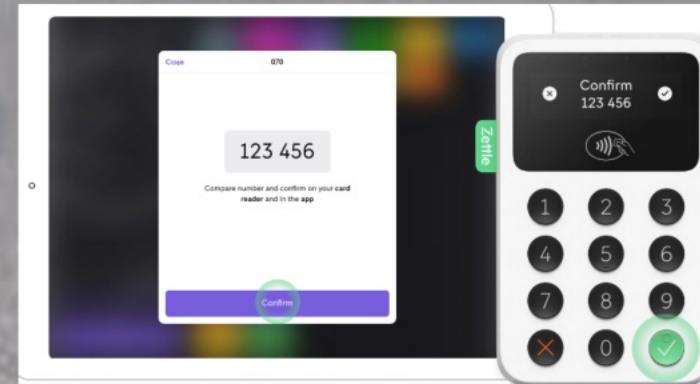
- Request: Keyboard & Display ~ Response: No I/O
- Pairing: **LE Legacy**
- Key Generation: **Just Works (Unauthenticated)**
- Temporary Key (TK): **Zero**
- Extract **LTK!**
- Crackle: **"Decrypt with LTK"**
 - Input: encrypted file + LTK
 - Output: decrypted file



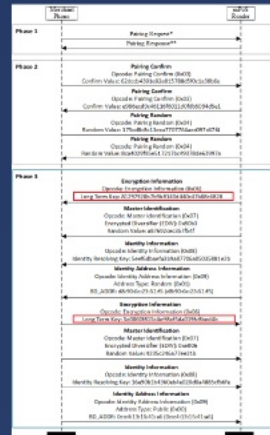
iZettle:

- ✔ Secure Connction
- ✔ Numeric comparison

Not Common Practice!



Extract Cryptographic Keys



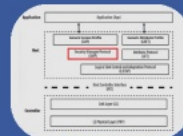
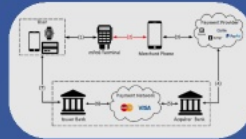
Field	Pairing Request	Pairing Confirm	Response	Pairing Response	Pairing Response
Code	0x01	0x02	0x03	0x04	0x05
I/O	0x01	0x02	0x03	0x04	0x05
CSIS	0x01	0x02	0x03	0x04	0x05
Showing	0x01	0x02	0x03	0x04	0x05
MTSE	1	1	1	1	1
RC	1	1	1	1	1
KP	0	0	0	0	0
Reserved	0x00	0x00	0x00	0x00	0x00
Max Err.	16	16	16	16	16
Initiator Key Distribution					
LTK	1	1	1	1	1
IRK	1	1	1	1	1
CSRK	1	1	1	1	1
Link Key	1	1	1	1	1
Reserved	0x00	0x00	0x00	0x00	0x00
Responder Key Distribution					
LTK	1	1	1	1	1
IRK	1	1	1	1	1
CSRK	1	1	1	1	1
Link Key	1	1	1	1	1
Reserved	0x00	0x00	0x00	0x00	0x00

- Request: Keyboard & Display ~ Response: No I/O
- Pairing: **LE Legacy**
- Key Generation: **Just Works (Unauthenticated)**
- Temporary Key (TK): **Zero**
- Extract **LTK!**
- Crackle: "**Decrypt with LTK**"
 - Input: encrypted file + LTK
 - Output: decrypted file

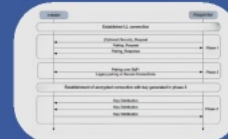


Encryption Security

- Communication: **BLE**
- Protocol **Stack**: Controller, (HCI), Host, Application
- Our interest: **Security Manager Protocol (SMP)**
 - Contains pairing
 - Generates and distributes keys
- Pairing Phases:
 - Phase 1: exchange pairing feature
 - Phase 2: determines pairing mechanism
 - Phase 3: distributes keys



BLE Protocol Stack



BLE Pairing

Pairing

Eavesdropping

Attack

Security Analysis of Mobile Point-of-Sale Terminals

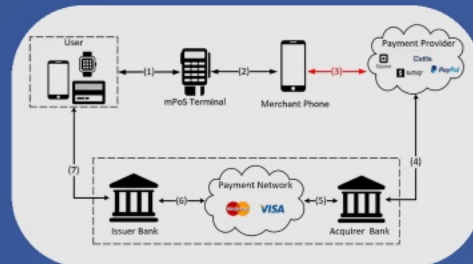
Mahshid Mehr Nezhad, Elliot Laidlaw, Feng Hao
University of Warwick, UK

Network and System Security 2023



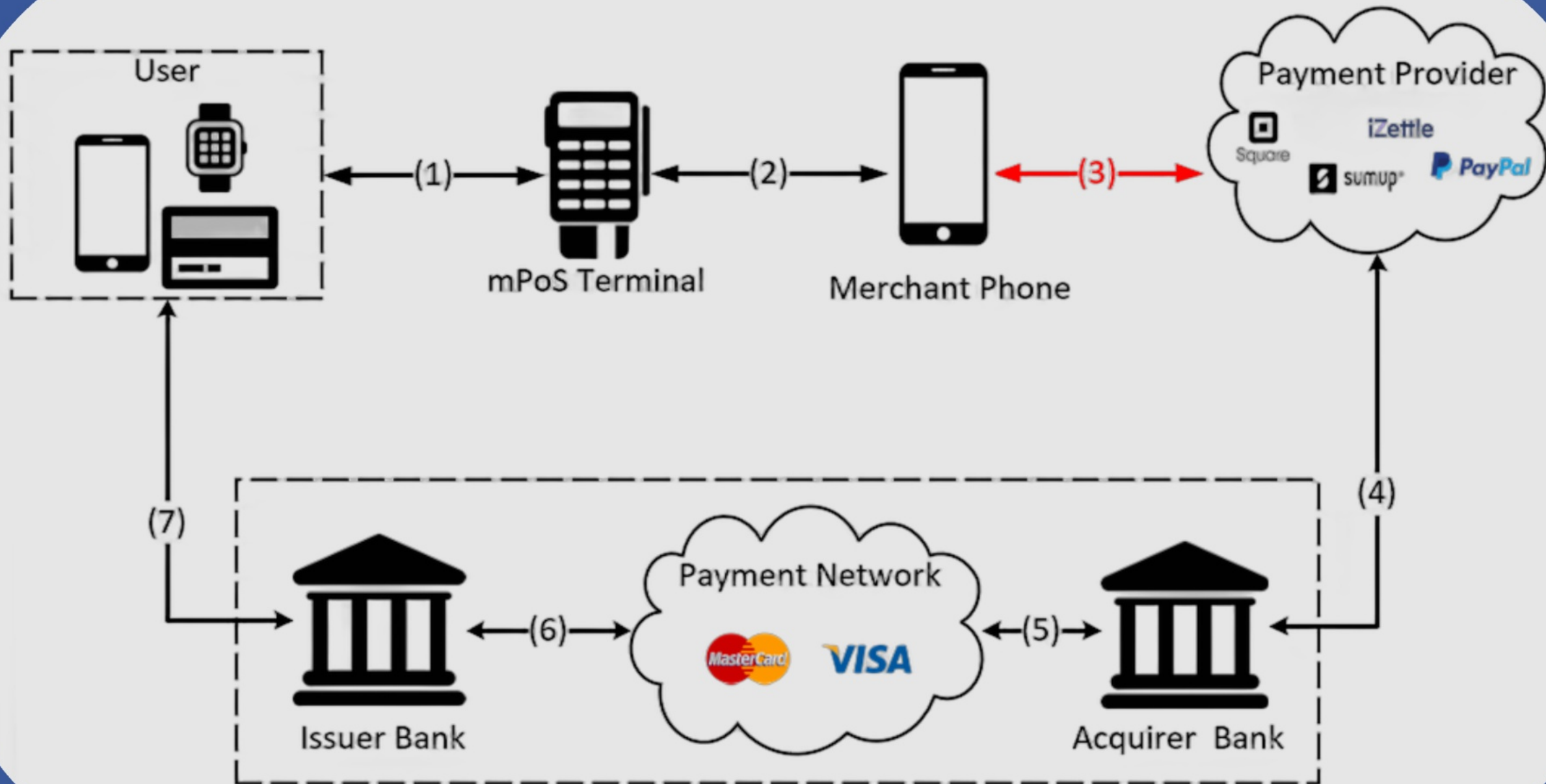
Network Security

- Communication: **HTTPS** (uses TLS)
- Threat model: **man-in-the-middle (MITM)**
- Proxy server: **mitmproxy**
 - intercept and decrypt traffic



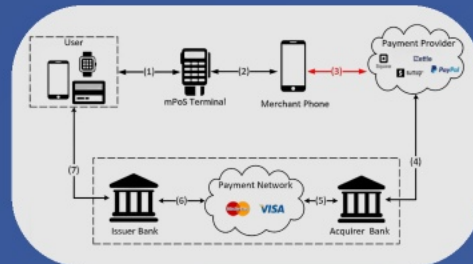
**HTTPS
Interception**

Attack



Network Security

- Communication: **HTTPS** (uses TLS)
- Threat model: **man-in-the-middle (MITM)**
- Proxy server: **mitmproxy**
 - intercept and decrypt traffic

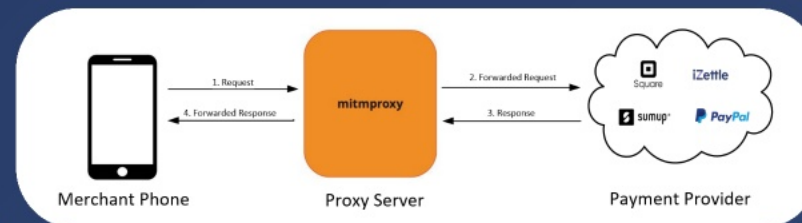


**HTTPS
Interception**

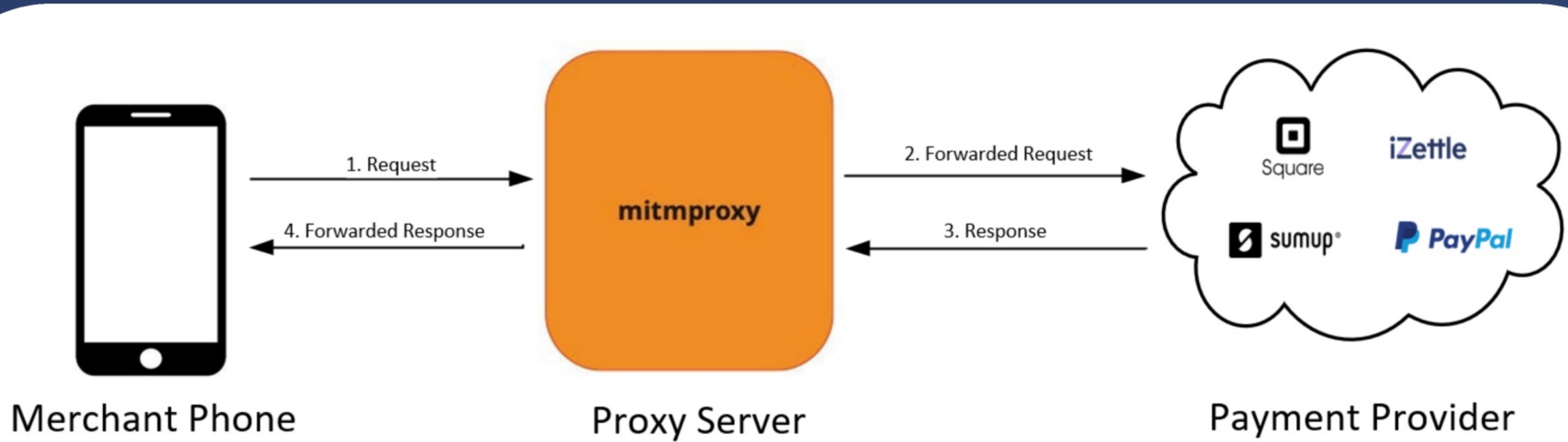
Attack

HTTPS Interception

1. Set up manual **proxy** configuration on the phone
2. Install mitmproxy **Certificate Authority (CA)** on the phone
3. Bypass **Certificate Pinning** (allowing user-added certificate) by modifying the app
4. The modified app now **trusts** the mitmproxy certificate!

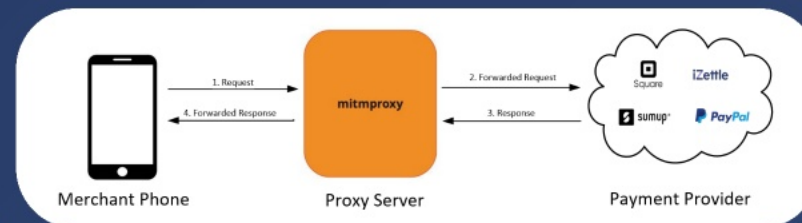


The modified app now **trusts** the mitmproxy certificate!



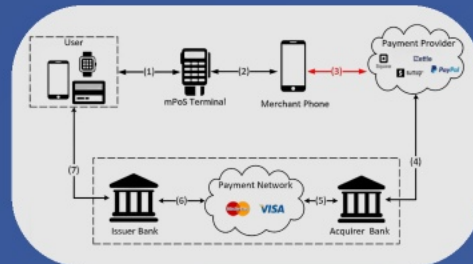
HTTPS Interception

1. Set up manual **proxy** configuration on the phone
2. Install mitmproxy **Certificate Authority (CA)** on the phone
3. Bypass **Certificate Pinning** (allowing user-added certificate) by modifying the app
4. The modified app now **trusts** the mitmproxy certificate!



Network Security

- Communication: **HTTPS** (uses TLS)
- Threat model: **man-in-the-middle (MITM)**
- Proxy server: **mitmproxy**
 - intercept and decrypt traffic




**HTTPS
Interception**

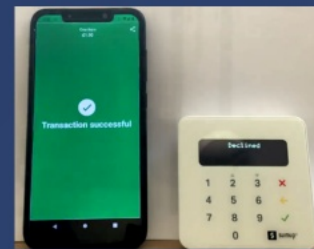
Attack

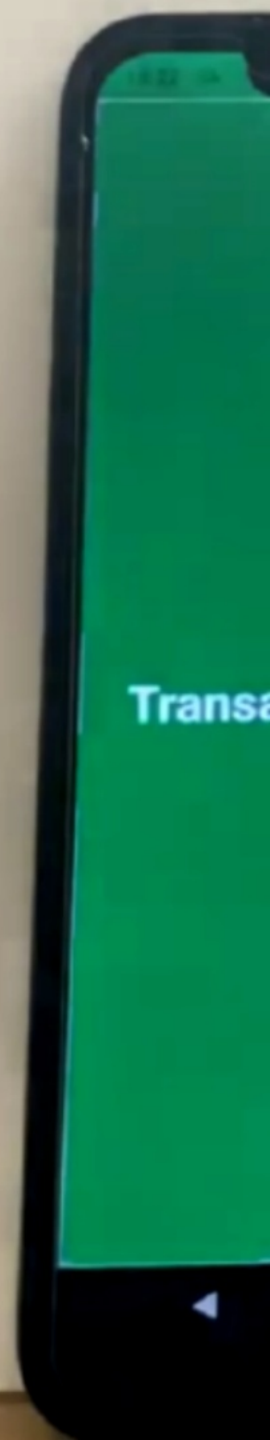
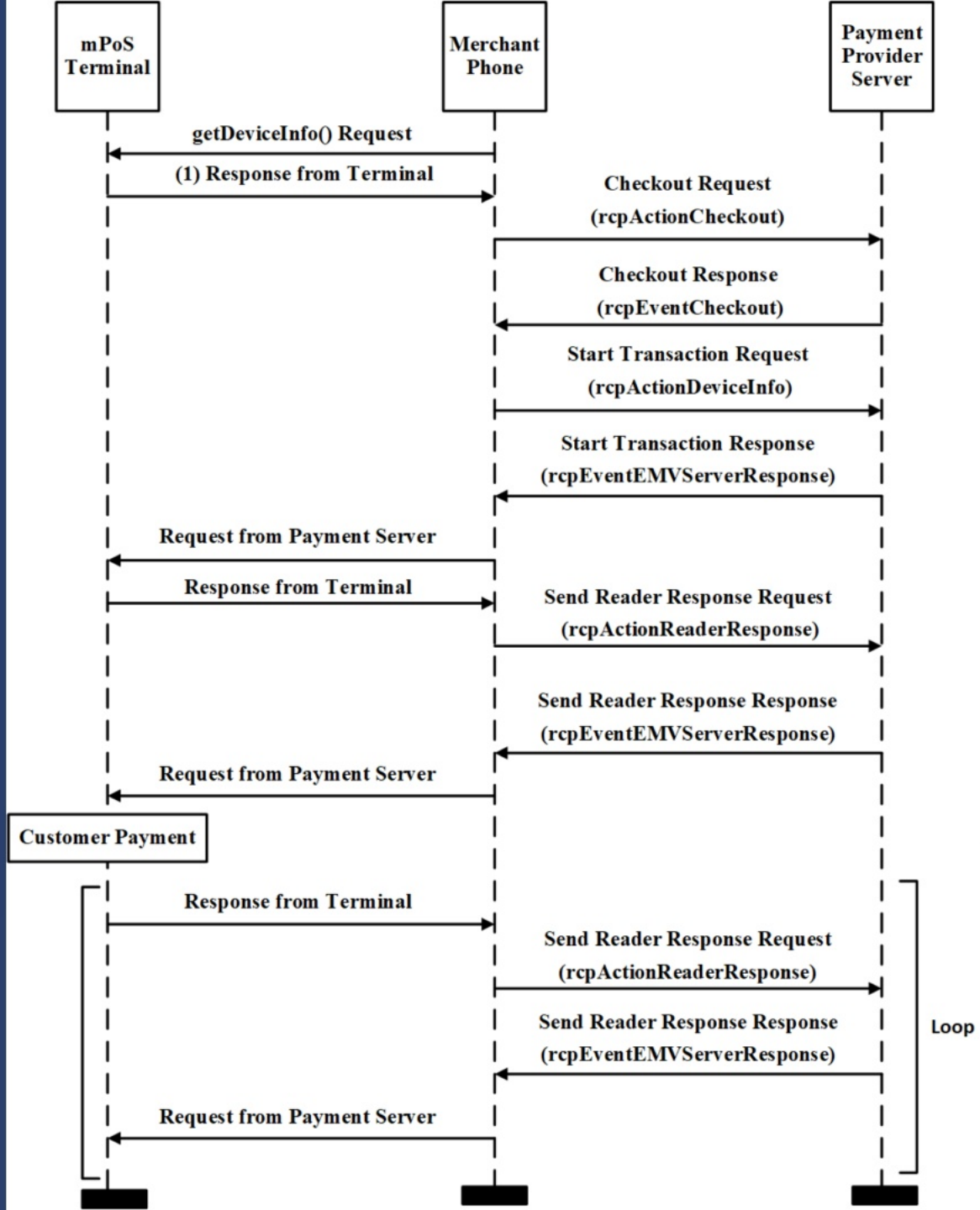
Tampering Attack

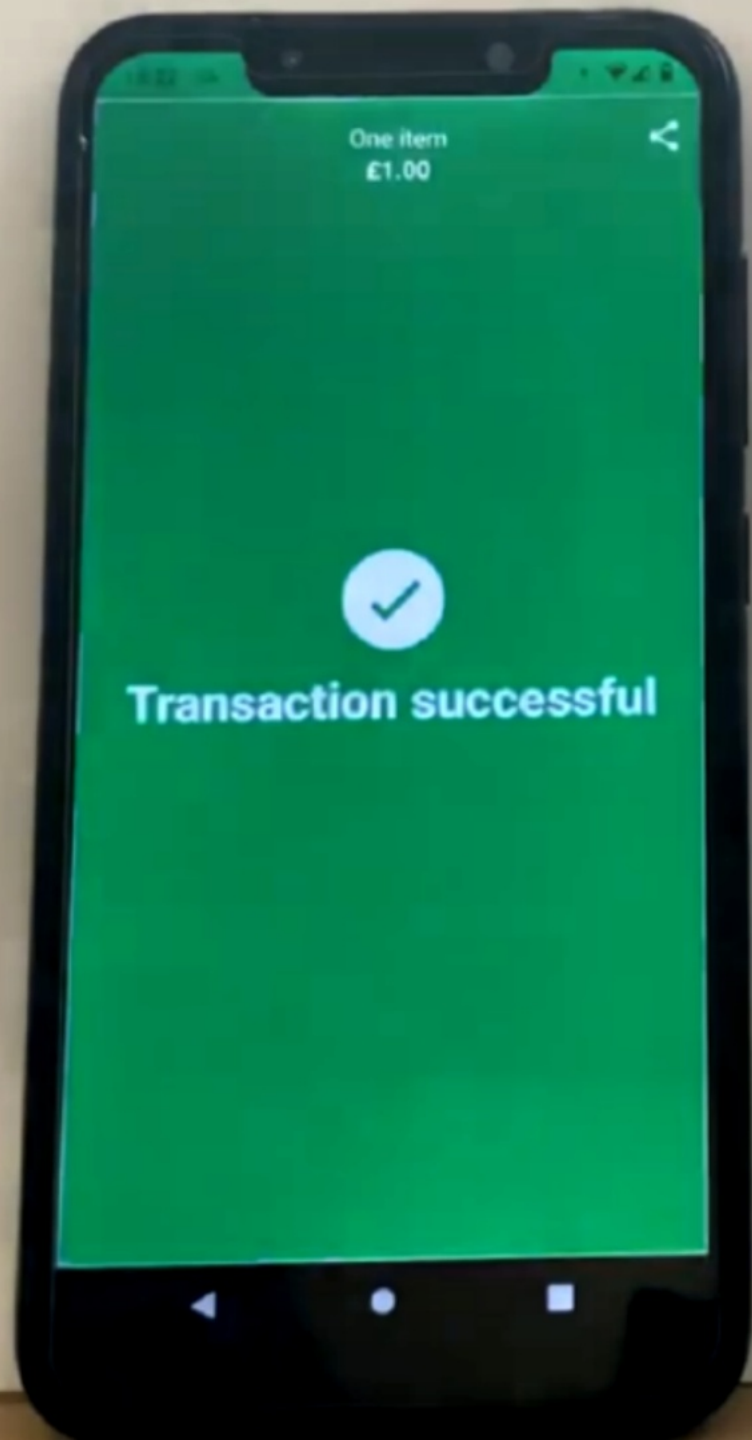
- **Tampering** with the (protected) messages
- Command "**PINPLUS SHOW DEFAULT MESSAGE**" coded in **plain text** Hexadecimal
- Inserting **arbitrary commands** to force the terminal to change the displayed message ("**Declined!**")
- **Challenge:** Protected messages are rejected
- **Solution:** Send "leave_protected_session" command first!



Index	Length	Request	Response
1	10	0000000000	0000000000
2	10	0000000000	0000000000
3	10	0000000000	0000000000
4	10	0000000000	0000000000
5	10	0000000000	0000000000
6	10	0000000000	0000000000
7	10	0000000000	0000000000
8	10	0000000000	0000000000
9	10	0000000000	0000000000
10	10	0000000000	0000000000
11	10	0000000000	0000000000
12	10	0000000000	0000000000
13	10	0000000000	0000000000
14	10	0000000000	0000000000
15	10	0000000000	0000000000
16	10	0000000000	0000000000
17	10	0000000000	0000000000
18	10	0000000000	0000000000
19	10	0000000000	0000000000
20	10	0000000000	0000000000
21	10	0000000000	0000000000
22	10	0000000000	0000000000
23	10	0000000000	0000000000
24	10	0000000000	0000000000
25	10	0000000000	0000000000
26	10	0000000000	0000000000
27	10	0000000000	0000000000
28	10	0000000000	0000000000
29	10	0000000000	0000000000
30	10	0000000000	0000000000
31	10	0000000000	0000000000
32	10	0000000000	0000000000
33	10	0000000000	0000000000
34	10	0000000000	0000000000
35	10	0000000000	0000000000
36	10	0000000000	0000000000
37	10	0000000000	0000000000
38	10	0000000000	0000000000
39	10	0000000000	0000000000
40	10	0000000000	0000000000
41	10	0000000000	0000000000
42	10	0000000000	0000000000
43	10	0000000000	0000000000
44	10	0000000000	0000000000
45	10	0000000000	0000000000
46	10	0000000000	0000000000
47	10	0000000000	0000000000
48	10	0000000000	0000000000
49	10	0000000000	0000000000
50	10	0000000000	0000000000
51	10	0000000000	0000000000
52	10	0000000000	0000000000
53	10	0000000000	0000000000
54	10	0000000000	0000000000
55	10	0000000000	0000000000
56	10	0000000000	0000000000
57	10	0000000000	0000000000
58	10	0000000000	0000000000
59	10	0000000000	0000000000
60	10	0000000000	0000000000
61	10	0000000000	0000000000
62	10	0000000000	0000000000
63	10	0000000000	0000000000
64	10	0000000000	0000000000
65	10	0000000000	0000000000
66	10	0000000000	0000000000
67	10	0000000000	0000000000
68	10	0000000000	0000000000
69	10	0000000000	0000000000
70	10	0000000000	0000000000
71	10	0000000000	0000000000
72	10	0000000000	0000000000
73	10	0000000000	0000000000
74	10	0000000000	0000000000
75	10	0000000000	0000000000
76	10	0000000000	0000000000
77	10	0000000000	0000000000
78	10	0000000000	0000000000
79	10	0000000000	0000000000
80	10	0000000000	0000000000
81	10	0000000000	0000000000
82	10	0000000000	0000000000
83	10	0000000000	0000000000
84	10	0000000000	0000000000
85	10	0000000000	0000000000
86	10	0000000000	0000000000
87	10	0000000000	0000000000
88	10	0000000000	0000000000
89	10	0000000000	0000000000
90	10	0000000000	0000000000
91	10	0000000000	0000000000
92	10	0000000000	0000000000
93	10	0000000000	0000000000
94	10	0000000000	0000000000
95	10	0000000000	0000000000
96	10	0000000000	0000000000
97	10	0000000000	0000000000
98	10	0000000000	0000000000
99	10	0000000000	0000000000
100	10	0000000000	0000000000






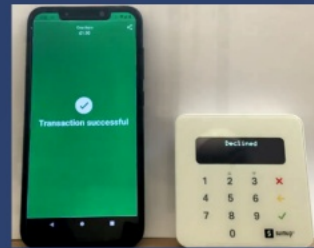


Tampering Attack

- **Tampering** with the (protected) messages
- Command "**PINPLUS SHOW DEFAULT MESSAGE**" coded in **plain text** Hexadecimal
- Inserting **arbitrary commands** to force the terminal to change the displayed message ("**Declined!**")
- **Challenge:** Protected messages are rejected
- **Solution:** Send "leave_protected_session" command first!



Index	Length	Request	Response
1	10	0000000000	0000000000
2	10	0000000000	0000000000
3	10	0000000000	0000000000
4	10	0000000000	0000000000
5	10	0000000000	0000000000
6	10	0000000000	0000000000
7	10	0000000000	0000000000
8	10	0000000000	0000000000
9	10	0000000000	0000000000
10	10	0000000000	0000000000
11	10	0000000000	0000000000
12	10	0000000000	0000000000
13	10	0000000000	0000000000
14	10	0000000000	0000000000
15	10	0000000000	0000000000
16	10	0000000000	0000000000
17	10	0000000000	0000000000
18	10	0000000000	0000000000
19	10	0000000000	0000000000
20	10	0000000000	0000000000



Tampering Attack: Barbie Version!

Not a barbie girl, but stuck in the barbie world!




Not a barbie girl, but stuck in the barbie world!

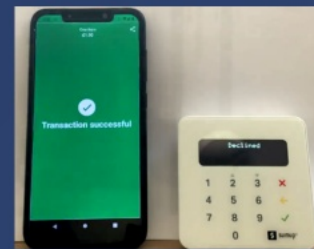


Tampering Attack

- **Tampering** with the (protected) messages
- Command "**PINPLUS SHOW DEFAULT MESSAGE**" coded in **plain text** Hexadecimal
- Inserting **arbitrary commands** to force the terminal to change the displayed message ("**Declined!**")
- **Challenge:** Protected messages are rejected
- **Solution:** Send "leave_protected_session" command first!

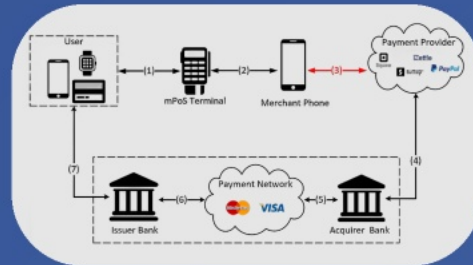


Index	Length	Request	Response	Status
1	10	0000000000	0000000000	Success
2	10	0000000000	0000000000	Success
3	10	0000000000	0000000000	Success
4	10	0000000000	0000000000	Success
5	10	0000000000	0000000000	Success
6	10	0000000000	0000000000	Success
7	10	0000000000	0000000000	Success
8	10	0000000000	0000000000	Success
9	10	0000000000	0000000000	Success
10	10	0000000000	0000000000	Success
11	10	0000000000	0000000000	Success
12	10	0000000000	0000000000	Success
13	10	0000000000	0000000000	Success
14	10	0000000000	0000000000	Success
15	10	0000000000	0000000000	Success
16	10	0000000000	0000000000	Success
17	10	0000000000	0000000000	Success
18	10	0000000000	0000000000	Success
19	10	0000000000	0000000000	Success
20	10	0000000000	0000000000	Success
21	10	0000000000	0000000000	Success
22	10	0000000000	0000000000	Success
23	10	0000000000	0000000000	Success
24	10	0000000000	0000000000	Success
25	10	0000000000	0000000000	Success
26	10	0000000000	0000000000	Success
27	10	0000000000	0000000000	Success
28	10	0000000000	0000000000	Success
29	10	0000000000	0000000000	Success
30	10	0000000000	0000000000	Success
31	10	0000000000	0000000000	Success
32	10	0000000000	0000000000	Success
33	10	0000000000	0000000000	Success
34	10	0000000000	0000000000	Success
35	10	0000000000	0000000000	Success
36	10	0000000000	0000000000	Success
37	10	0000000000	0000000000	Success
38	10	0000000000	0000000000	Success
39	10	0000000000	0000000000	Success
40	10	0000000000	0000000000	Success
41	10	0000000000	0000000000	Success
42	10	0000000000	0000000000	Success
43	10	0000000000	0000000000	Success
44	10	0000000000	0000000000	Success
45	10	0000000000	0000000000	Success
46	10	0000000000	0000000000	Success
47	10	0000000000	0000000000	Success
48	10	0000000000	0000000000	Success
49	10	0000000000	0000000000	Success
50	10	0000000000	0000000000	Success
51	10	0000000000	0000000000	Success
52	10	0000000000	0000000000	Success
53	10	0000000000	0000000000	Success
54	10	0000000000	0000000000	Success
55	10	0000000000	0000000000	Success
56	10	0000000000	0000000000	Success
57	10	0000000000	0000000000	Success
58	10	0000000000	0000000000	Success
59	10	0000000000	0000000000	Success
60	10	0000000000	0000000000	Success
61	10	0000000000	0000000000	Success
62	10	0000000000	0000000000	Success
63	10	0000000000	0000000000	Success
64	10	0000000000	0000000000	Success
65	10	0000000000	0000000000	Success
66	10	0000000000	0000000000	Success
67	10	0000000000	0000000000	Success
68	10	0000000000	0000000000	Success
69	10	0000000000	0000000000	Success
70	10	0000000000	0000000000	Success
71	10	0000000000	0000000000	Success
72	10	0000000000	0000000000	Success
73	10	0000000000	0000000000	Success
74	10	0000000000	0000000000	Success
75	10	0000000000	0000000000	Success
76	10	0000000000	0000000000	Success
77	10	0000000000	0000000000	Success
78	10	0000000000	0000000000	Success
79	10	0000000000	0000000000	Success
80	10	0000000000	0000000000	Success
81	10	0000000000	0000000000	Success
82	10	0000000000	0000000000	Success
83	10	0000000000	0000000000	Success
84	10	0000000000	0000000000	Success
85	10	0000000000	0000000000	Success
86	10	0000000000	0000000000	Success
87	10	0000000000	0000000000	Success
88	10	0000000000	0000000000	Success
89	10	0000000000	0000000000	Success
90	10	0000000000	0000000000	Success
91	10	0000000000	0000000000	Success
92	10	0000000000	0000000000	Success
93	10	0000000000	0000000000	Success
94	10	0000000000	0000000000	Success
95	10	0000000000	0000000000	Success
96	10	0000000000	0000000000	Success
97	10	0000000000	0000000000	Success
98	10	0000000000	0000000000	Success
99	10	0000000000	0000000000	Success
100	10	0000000000	0000000000	Success



Network Security

- Communication: **HTTPS** (uses TLS)
- Threat model: **man-in-the-middle (MITM)**
- Proxy server: **mitmproxy**
 - intercept and decrypt traffic



**HTTPS
Interception**

Attack

Security Analysis of Mobile Point-of-Sale Terminals

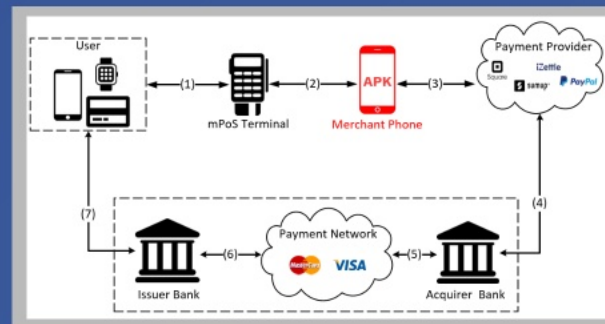
Mahshid Mehr Nezhad, Elliot Laidlaw, Feng Hao
University of Warwick, UK

Network and System Security 2023



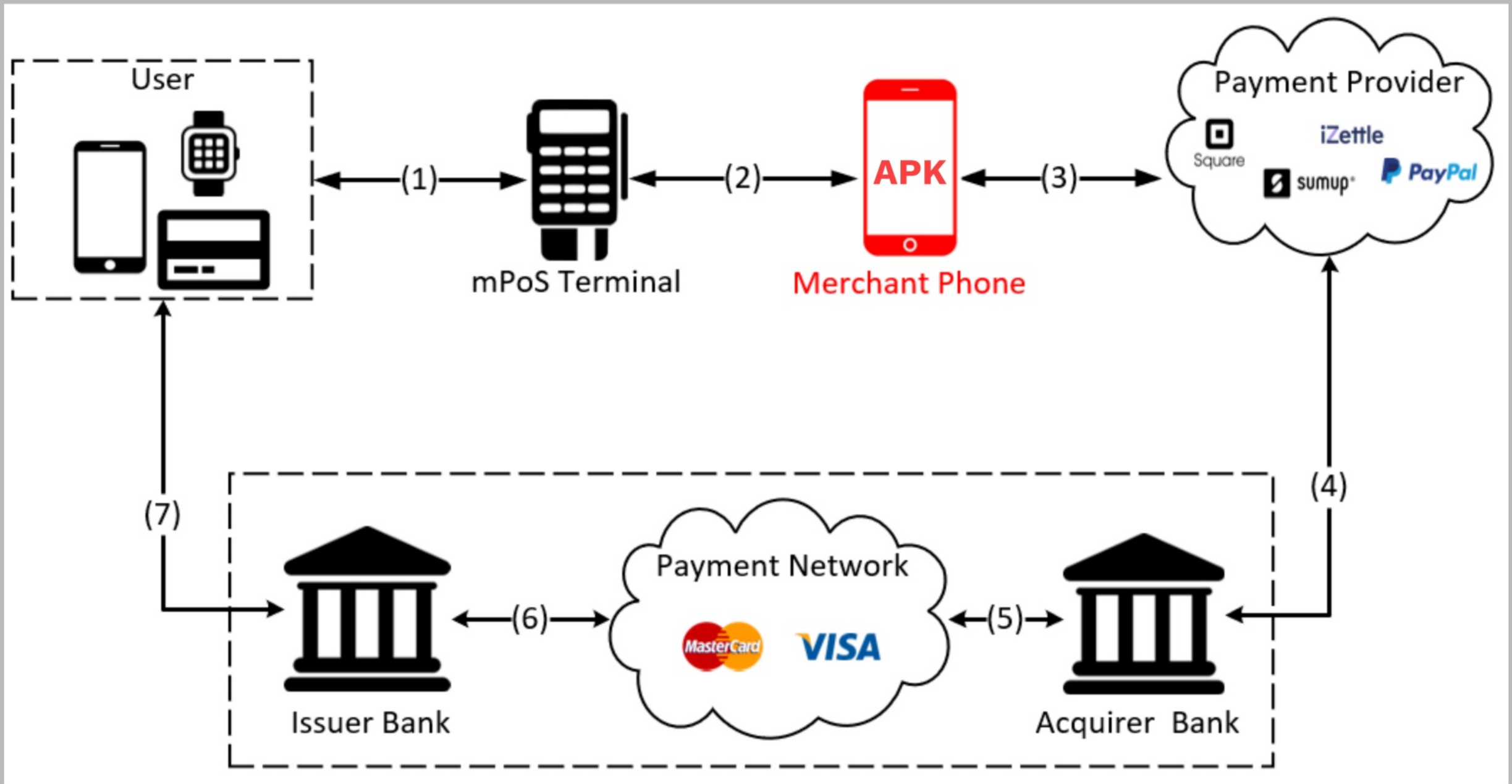
Software Security

- Mobile application: manages the terminal
- Reverse-engineering to identify vulnerabilities in the code
- proof of concept: Android phone, APK



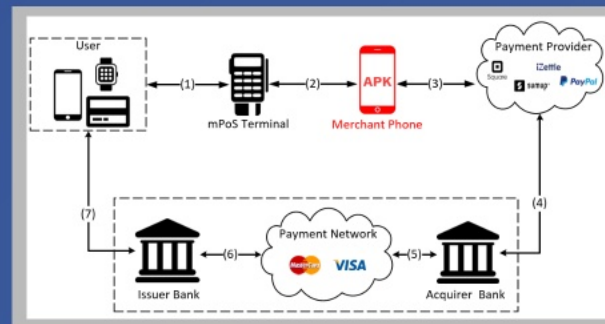
Reverse
Engineering

Attacks



Software Security

- Mobile application: manages the terminal
- Reverse-engineering to identify vulnerabilities in the code
- proof of concept: Android phone, APK



Reverse
Engineering

Attacks

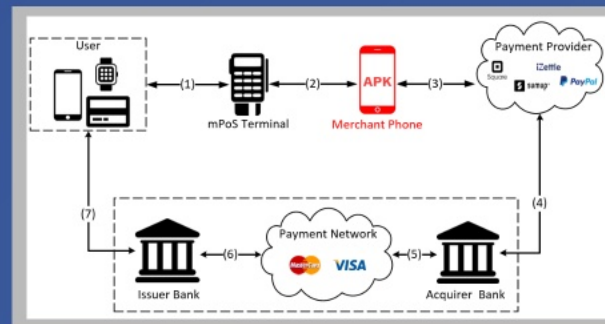
Reverse Engineering Steps

1. Download genuine APK
2. Decompile the APK
 - apktool: Smali code (main)
 - Java decompiler: Java code (complementary)
3. Make Modifications
4. Recompile the APK (e.g, apk-mitm)
5. Sign the APK (e.g, uber-apk-signer)
6. Re-install compromised App!



Software Security

- Mobile application: manages the terminal
- Reverse-engineering to identify vulnerabilities in the code
- proof of concept: Android phone, APK



Reverse
Engineering

Attacks

Attacks

1. Bypass Certificate Pinning

- Replace the application's network security config to allow user-added certs.
- Modify the code to disable cert. pinning implementation

2. Bypass Protected Messages

- Leave Protected Session

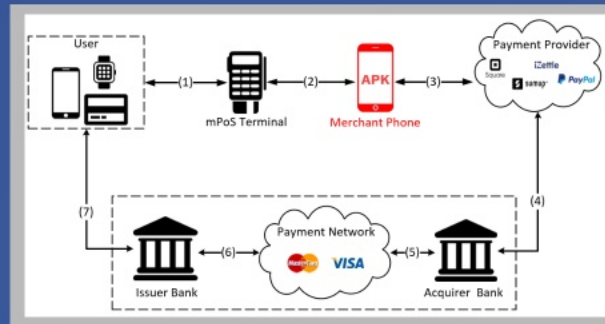
3. Disable Security Features: Beep Sound

- Find "AudioManager" Class ==> "PlaySoundEffect" method
- Modify or remove
- Sounds are muted!

4. ...

Software Security

- Mobile application: manages the terminal
- Reverse-engineering to identify vulnerabilities in the code
- proof of concept: Android phone, APK



Reverse
Engineering

Attacks

Security Analysis of Mobile Point-of-Sale Terminals

Mahshid Mehr Nezhad, Elliot Laidlaw, Feng Hao
University of Warwick, UK

Network and System Security 2023



Conclusion

- mPOS terminals can be vulnerable in various ways
- The involvement of merchant's phone can make it worse!

**Potential
Solutions**

What's next?

Thanks!

Potential Solutions

- Secure Pairing on BLE
- Code Obscuring
- Anti-tampering
- Abuse Detection

- Requires Further Research!

Conclusion

- mPOS terminals can be vulnerable in various ways
- The involvement of merchant's phone can make it worse!

**Potential
Solutions**

What's next?

Thanks!

What's next?

Tap-to-phone!

- Potential Solution
- Susceptible to risks
- Requires further research



Conclusion

- mPoS terminals can be vulnerable in various ways
- The involvement of merchant's phone can make it worse!

**Potential
Solutions**

What's next?

Thanks!

Conclusion

- mPoS terminals can be vulnerable in various ways
- The involvement of merchant's phone can make it worse!

**Potential
Solutions**

What's next?

Thanks!

Thank you!

Any questions?

 Mahshid.mehr-nezhad@warwick.ac.uk

 @MahshiidMehr

 @mahshidmehr

Conclusion

- mPoS terminals can be vulnerable in various ways
- The involvement of merchant's phone can make it worse!

**Potential
Solutions**

What's next?

Thanks!

Security Analysis of Mobile Point-of-Sale Terminals

Mahshid Mehr Nezhad, Elliot Laidlaw, Feng Hao
University of Warwick, UK

Network and System Security 2023

