TrustOTP: Transforming Smartphones into Secure One-Time Password Tokens

He Sun, Kun Sun, Yuewu Wang, and Jiwu Jing

Presented by Fengwei Zhang
Outline

• Introduction
• Motivation
• Architecture
• Implementation
• Evaluation
• Summary
Outline

• Introduction
• Motivation
• Architecture
• Implementation
• Evaluation
• Summary
One-time Password (OTP)

• A password that is valid for only one login session or transaction
  – Not vulnerable to reply attacks
  – Widely used in Two-factor Authentication
  – HOTP (Hash-based OTP)
    • Event triggered, key & counter
  – TOTP (Time-based OTP)
    • Time synchronized, key & clock
  – Hardware token & software App
Existing Solutions

- **Hardware-based**
  - RSA SecurID
  - Yubikey

- **Software-based**
  - Google authenticator
  - McAfee one-time password
Outline

• Introduction
• **Motivation**
• Architecture
• Implementation
• Evaluation
• Summary
Limitation

• Hardware-based --- not flexible
  – Unprogrammable
  – Expensive

• Software-based --- not secure
  – Vulnerable to external attacks
Goals

• Confidentiality
  – Malicious mobile OS cannot compromise the keying material (seed) in the OTP generator
  – It cannot read the OTP

• Reliability and Availability
  – Trusted inputs (e.g., clock time) for the OTP generator
  – Trusted display
  – OTP works even if mobile OS crashes

• Small TCB
TrustZone-related Work

• TrustICE (Sun et al.[1])
  – Isolated Computing Environment in the normal domain
• SeCReT (Jang et al.[2])
  – Secure channel between secure domain and normal application
• Hypervision (Azab et al.[3])
  – Real-time kernel protection in the normal domain
• TrustDump (Sun et al.[4])
  – Reliable Memory Acquisition of the mobile OS
• Smartphone as location verification token for payments (Marforio et al.[5])
• Trusted Language Runtime for trusted applications in the secure domain (Santos et al.[6])
Outline

• Introduction
• Motivation
• Architecture
• Implementation
• Evaluation
• Summary
TrustOTP Architecture

– In the secure domain
– Shared I/O device with the rich OS
– Reliable switch between domains

### Diagram

**Normal Domain**
- Non-secure Permanent Storage
- Non-secure Framebuffer
- Touchscreen Driver

**Rich OS**
- Framebuffer Driver

**Secure Domain**
- Secure Permanent Storage
- Secure Framebuffer
- Secure Display Controller
- Secure Touchscreen Driver
- Secure Clock
- Secure Counters

**TrustOTP**
- OTP Generator
- TOTP
- HOTP

**Reliable Switch**
- User Input of the Rich OS
- Display with Touchscreen
- User Input of TrustOTP
Challenges

• Secure input and display though shared touchscreen
• Reliable switch
• Generator protection
  – Static code
  – Execution environment
• Availability
Outline

• Introduction
• Motivation
• Architecture
  • Implementation
• Evaluation
• Summary
Security Analysis

• Information leakage
  – Generated OTPs
  – Shared keys

• Control flow tampering
  – Code integrity
  – Execution integrity (e.g., Interrupt)

• Denial-of-service
  – Switch between domains
  – Static & dynamic code
  – Display
Boot Sequence

- Secure storage
  - MicroSD card
- Memory Isolation
  - TZASC (TrustZone Address Space Controller)
  - Watermark mechanism
  - Secure boot
- Secure bootloader
  - Non-secure bootloader
  - Rich OS
TrustOTP Triggering

• Reliable switch
  – Non-maskable interrupt (NMI)
    • The rich OS cannot block or intercept
  – Secure Interrupt (FIQ)
    • The rich OS cannot manipulate
  – Interrupt source (configurable)
    • Physical button
    • Timer
OTP Generation

• Hash-based one-time password (HOTP)
  – Key, counter

• Time-based one-time password (TOTP)
  – Key, Clock

Listing 1: OTP Generation Functions

```c
int oath_hotp_generate (const char *secret,
    size_t secret_length,
    uint64_t moving_factor,
    unsigned digits,
    char *output_otp)

int oath_totp_generate (const char *secret,
    size_t secret_length,
    time_t now,
    unsigned time_step_size,
    unsigned digits,
    char *output_otp)
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>secret</td>
<td>the secret key</td>
</tr>
<tr>
<td>secret_length</td>
<td>length of the secret Key</td>
</tr>
<tr>
<td>moving_factor</td>
<td>secure counter in HOTP</td>
</tr>
<tr>
<td>now</td>
<td>secure clock in TOTP</td>
</tr>
<tr>
<td>time_step_size</td>
<td>time period between two TOTPs</td>
</tr>
<tr>
<td>digits</td>
<td>length of the generated OTP</td>
</tr>
<tr>
<td>output_otp</td>
<td>the generated OTP</td>
</tr>
</tbody>
</table>
OTP Display

- Secure I/O
  - Display: IPU (Image Processing Unit) + LCD
  - Input: 4-wire resistive touchscreen
- User-friendly manner
  - Rich OS and TrustOTP run concurrently
  - Watchdog timer
  - 1.5 seconds / cycle
    - 0.5 second for display
    - 1 second for input 2~3 numbers
Outline

- Introduction
- Motivation
- Architecture
- Implementation
- Evaluation
- Summary
Evaluation

• Freescale i.MX53 QSB
  – A Cortex-A8 1GHz processor
  – 1GB DD3 RAM
  – 4GB microSD card
• Monsoon power monitor
  – Power measurement
  – Power logging
TrustOTP Performance

• Before OTP display (60.48 ms)

<table>
<thead>
<tr>
<th>Step</th>
<th>Operation</th>
<th>Time (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Domain Switching</td>
<td>0.002</td>
</tr>
<tr>
<td>2</td>
<td>Context Saving</td>
<td>0.0006</td>
</tr>
<tr>
<td>3</td>
<td>TOTP/HOTP Generation</td>
<td>0.048/0.044</td>
</tr>
<tr>
<td>4</td>
<td>Background Matching</td>
<td>49.85</td>
</tr>
<tr>
<td>5</td>
<td>OTP Drawing</td>
<td>8.029</td>
</tr>
<tr>
<td>6</td>
<td>IPU Check</td>
<td>2.22</td>
</tr>
<tr>
<td>7</td>
<td>Framebuffer Replacement</td>
<td>0.28</td>
</tr>
</tbody>
</table>

• After OTP display (7.52 ms)

<table>
<thead>
<tr>
<th>Step</th>
<th>Operation</th>
<th>Time (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Flushing IPU &amp; Rich OS Recovery</td>
<td>7.47</td>
</tr>
<tr>
<td>2</td>
<td>Domain Switching</td>
<td>0.05</td>
</tr>
</tbody>
</table>
Impact on the Rich OS

- Rich OS vs. TrustOTP
- Anutu
  - CPU & RAM
  - I/O devices
- Vellamo
Power Consumption

- Rich OS
  - Average = 2,128 mW
- TrustOTP running
  - Average = 2,230 mW
- TrustOTP without display
Outline

• Introduction
• Motivation
• Architecture
• Implementation
• Evaluation
• Summary
Summary

• TrustOTP: Hardware-assisted OTP Token on smartphones
  – Security (confidentiality, integrity, availability)
  – Flexibility (various and multiple OTPs)

• Low performance overhead on the Rich OS
  – No need to modify the Rich OS
  – Low power consumption
References


