Security Considerations for Low-Power Constrained Networks

René Struik
(Certicom Research)
E-mail: rstruik@certicom.com
The “Holy Grail”: Security and Ease of Use

“Computer users have been taught for years that computer security systems can’t be effective unless they are complex and difficult to use. In reality, this conventional wisdom is completely wrong.”

— Lorrie Faith Cranor, Carnegie Mellon University

Security technology can make trust lifecycle management intuitive and hidden from the user.

Figure 1. Connecting a laptop to a secured wireless network in 32 seconds. All the user has to do is briefly align the infrared ports of laptop and access point and press the Enter key twice. These are snapshots from a live Network-in-a-Box demonstration.

Ease of Configuration and Reconfiguration

Ease of configuration:
- Merging of networks
- Partitioning of networks
- Device portability and orphaning
- Hand-over of control (remote, backup)
- Synchronization and failure recovery
**March 25, 2010**  
*Feasibility of Crypto on Small Devices*  

**Conventional wisdom:** Symmetric-key cryptographic functionality, let alone public-key cryptographic functionality, are expensive to implement with sensor networks.  

**Status anno 2008:** conventional wisdom challenged for all but most mundane devices.

**Examples:** Bluetooth v2.1, ZigBee Smart Metering, RFID e-Passport.

---

**• ZigBee Smart Energy Profile Certificate Structure:**

- 22 octets: Public Key Device
- 8 octets: DeviceId
- 8 octets: CA Id
- 10 octets: AttributeData
- (total: 48 octets)

---

**• Low-energy hardware implementations:**

<table>
<thead>
<tr>
<th></th>
<th>Smart Sensors ¹</th>
<th>RFID ²</th>
</tr>
</thead>
<tbody>
<tr>
<td>clock frequency</td>
<td>2 MHz</td>
<td>10 MHz</td>
</tr>
<tr>
<td>#gates</td>
<td>~10 kgates</td>
<td>~100 kgates</td>
</tr>
<tr>
<td>CMOS process</td>
<td>130nm</td>
<td>250nm</td>
</tr>
<tr>
<td>Energy exp.:</td>
<td>~ 250 µJ</td>
<td>&lt; 100 µJ</td>
</tr>
<tr>
<td>Computation</td>
<td>signature verify</td>
<td>point multiple</td>
</tr>
</tbody>
</table>

Less than energy expenditure single IEEE 802.15.4 frame!

Sources:

¹Certicom-internal
²SAC 2008 conference
Deployment Scenarios vs. Security Design

Diverse deployment scenarios
- Home Automation  draft-ietf-roll-home-routing-reqs-11
- Building Automation  draft-ietf-roll-building-routing-reqs-09
- Urban Settings  RFC 5548 - Routing Requirements for Urban Low-Power and Lossy Networks (May 2009)
- Industrial Control  RFC 5673 - Industrial Routing Requirements (October 2009)
ZigBee, ISA SP100.11a, “smart grid”, “Internet of Things”, etc.

Actual security design
Unified design that fits these diverse deployment scenarios
- concise set of cryptographic and security mechanisms;
- single security policy framework;
- configuration parameters application-dependent.
This allows for mass-scale production, while still allowing for customization (e.g., as to security services provided, granularity of assurances, used keys, device roles, etc.)

This may require consideration of system perspective, taking into account the entire system and device lifecycle and ease-of-use and ease-of-deployment
Security Architectural Framework: Overview

ACL initialization

CA key initialization

ACL Maintenance

ACL

ACL

CA key initialization

Certificate maintenance

Public key verification

Public key verification

Extracted public key info

Extracted public key info

Authentication, key establishment

Wrapped public key info

Wrapped public key info

Certificate maintenance

Public key verification

Data key repository

Data key repository

Data key maintenance

Data key maintenance

Wrapped data key info

Wrapped data key info

Wrapped data key info

Wrapped data key info

Key info

Data key

Network and down

Network and down

Data key

Key info

Data

Encryptor/decryptor

Encryptor/decryptor

data

data

Upper layers

Upper layers

(Link key, A, B)

(Link key, A, B)

René Struik (Certicom Research)
Security Architectural Framework – Design Aspects

Various aspects, including
– Security Policy and Trust Model
– Configuration and Installation
– Protocol design aspects

Adhoc networks
• No centralized management
• Promiscuous behavior
• Unreliability

Sensor networks
• Low energy consumption
• Low manufacturing cost

Security constraints
• Decentralized key management
• Flexible configuration and trust model
• Low impact key compromise
• Automatic lifecycle management
• Low communication overhead
• Low implementation cost

For details, cf. draft-struik-6lowapp-security-considerations-00
Full stack device, including per-layer and shared parameters

Device-wide parameters

Transport functions

Network functions

Data Link functions

PHY functions

APP functions

Transport parameters

Network parameters

Data Link parameters

PHY parameters

DeviceID

Keying material

Security policies

Security protocols

AES ECC RNG

Trust binding
Full stack device, including per-layer and shared parameters

Device-wide parameters

APP parameters

Transport parameters

Network parameters

PHY parameters

DeviceID

Keying material

Security policies

Security protocols

AES, ECC, RNG

Authorizations (policy state machine)
Acceptability test based on
- Device Id
- Tag Name
- Device Label
- Open enrolment
- Proximity-based techniques

Trust management via device identities

Configuration
Deployment Scenarios

Scenario #1:  
mix-and-match of nodes from different vendors

Scenario #2: 
addition of nodes to operational network

Scenario #3:  
security audit

Scenario #4: 
device repair and replacement (roaming in/out different user sites)

Scenario #5: 
network separation (devices joining wrong network)

Scenario #6:  
thwarting malicious attacks by (former) insiders

Scenario #7: 
thwarting attacks by outsiders via insiders (held at ‘gunpoint’)

Scenario #8: 
addition of subsystem (‘skid’) assembled elsewhere to operational network

\(^1\)Deployment scenarios discussed with ZigBee, ISA SP100.11a user community
Desired Features and Benefits (1)

**Ease of use.** Trust lifecycle management appears the same as that of an unsecured network and relies on
- proper identification of devices (e.g., reading off a label of physical module);
- proper management of device roles (e.g., adding these to, resp. removing these from a white list, e.g., via a workstation GUI).

Thus, trust lifecycle management relies completely on handling of public information.

**Flexibility.** Virtually no restrictions w.r.t. support for
- mix-and-match of devices from different vendors;
- changes to network topology (merging or partitioning of networks, device replacement or addition, addition of pre-assembled subsystem);
- changes to device roles (e.g., smooth hand-over of system manager, security manager roles, via ‘soft reboot’);
- back-up and failure recovery (since management fully relies on public information).
Desired Features and Benefits (2)

Minimize trust dependencies.
- Reduced reliance on trustworthy personnel;
- Virtually no training requirements for operational personnel;
- Virtual removal of trust dependencies between different entities in value chain (whether OEM, vendor, system integrator, installer, or user).
- Ease of security auditability.

Support for flexible deployment and business models.
Network topology changes or device role changes present a ‘clean’ logical separation between state prior to and after such an event (thus, allowing subscription-based services, outsourced management, re-contracting, etc.).

Enforcement of standards compliance. Enforcement possible by only issuing a certificate to devices from vendors that passed conformance testing.

No reliance on configuration tools and out-of-band configuration steps. A configuration tool may be used, but is not strictly necessary for trust enforcement.
Recommended Next Steps

– Validate deployment scenarios of various application domains

– Further consider overarching issues and broadening scope of security document

– Align security design framework with stack layering

– Combine efforts of current security-relevant drafts with CoRE:
  draft-struik-6lowapp-security-considerations-00
  draft-oflynn-core-bootstrapping-00

– Specify protocols that implement security design (existing or new)

– Consider where work should be carried out, since cross-WG effort:
  Security design relevant for 6lowapp, 6lowpan, roll, Smart Grid

Discussion:
– How to realize (re-charter, coordinate with others, etc.)?