Rescuing LoRaWAN 1.0

Workshop CRYPTACUS
Internet of Things

- 20 billion internet-connected things by 2020 [Gartner]
- Main domains
  - smart home (Zigbee, Z-Wave, BLE, DECT ULE, Thread, etc.)
  - eHealth
  - industrial IoT => allegedly
    - the largest volume of things
    - the most sensitive use cases
Internet of Things

- 20 billion internet-connected things by 2020 [Gartner]
- Main domains
  - smart home (Zigbee, Z-Wave, BLE, DECT ULE, Thread, etc.)
  - eHealth
  - industrial IoT => allegedly
    - the largest volume of things
    - the most sensitive use cases

- A proposal for industrial IoT: LoRa (communication layer) & LoRaWAN (security layer)
- Originally conceived by Semtech (Cycleo). Now promoted by LoRa Alliance.
- Deployed in more than 50 countries worldwide: USA (100 cities), Japan, China (300 million people), India (400 million people), France, Netherlands, South Africa, etc.
- Use cases: temperature monitoring, presence detection, remote device on/off switch, etc.
- Current deployed version: v1.0 (this talk).
Architecture

End-devices ➔ Gateway ➔ Network Server ➔ Application Server
Key exchange

End-device (MK)  Network Server (MK)  Application Server

req

ans
Key exchange

1. \( r_{nd_C} \leftarrow \{0,1\}^{16} \)
2. \( \tau_C = MAC_{MK}(id_{AS} \mid id_C \mid r_{nd_C}) \)
3. \( req = id_{AS} \mid id_C \mid r_{nd_C} \mid \tau_C \)
Key exchange

1. \( \text{md}_C \leftarrow \{0,1\}^{16} \)
2. \( \tau_C = \text{MAC}_{\text{MK}}(\text{id}_{AS} | \text{id}_C | \text{md}_C) \)
3. \( \text{req} = \text{id}_{AS} | \text{id}_C | \text{md}_C | \tau_C \)
4. check req
5. \( \text{md}_S \leftarrow \{0,1\}^{24} \)
6. \( \tau_S = \text{MAC}_{\text{MK}}(\text{md}_S | \text{id}_S | \text{addr} | \text{prms}) \)
7. \( \text{ans} = \text{AES}^{-1}_{\text{MK}}(\text{md}_S | \text{id}_S | \text{addr} | \text{prms} | \tau_S) \)
8. check ans
1. $\text{rnd}_C \leftarrow \{0,1\}^{16}$
2. $\tau_C = \text{MAC}_{MK}(\text{id}_{AS} | \text{id}_C | \text{rnd}_C)$
3. req = $\text{id}_{AS} | \text{id}_C | \text{rnd}_C | \tau_C$

4. check req
5. $\text{rnd}_S \leftarrow \{0,1\}^{24}$
6. $\tau_S = \text{MAC}_{MK}(\text{rnd}_S | \text{id}_S | \text{addr} | \text{prms})$
7. ans = $\text{AES}^{-1}_{MK}(\text{rnd}_S | \text{id}_S | \text{addr} | \text{prms} | \tau_S)$

8. check ans

Data encryption key $\text{Ke} = \text{ENC}_{MK}(01 | v)$
Data integrity key $\text{Ki} = \text{ENC}_{MK}(02 | v)$

with $v = \text{rnd}_S | \text{id}_S | \text{rnd}_C | 00..00$
Secure channel

- **End-device (MK)**
- **Network Server (MK)**
- **Application Server**

- **Ke, Ki**
- **data confidentiality (Ke)**
- **data integrity (Ki)**

- **Application frame**
  - `hdr` [pld]_{Ke}
  - `Ki`

- **Network frame**
  - `hdr` [pld]_{Ki}
  - `Ki`
Secure channel

- Encryption: based on AES CCM
  - $A_j (16) = 01 | 00...00 | \text{dir} | \text{addr} (4) | \text{cnt} (4) | 00 | j (1)$
  - $S_j = \text{AES}_K(A_j)$ with $K = \begin{cases} Ke & \text{if application data} \\ Ki & \text{if network data} \end{cases}$
  - ctxt = pld $\oplus (S_0 | .. | S_{n-1})$

- Application frame
  - $[\text{pld}]_{Ke}$
  - $\text{hdr}$

- Network frame
  - $[\text{pld}]_{Ki}$
  - $\text{hdr}$
Secure channel

- **Encryption**: based on AES CCM
  - $A_j(16) = 01 | 00...00 | \text{dir} | \text{addr}(4) | \text{cnt}(4) | 00 | j(1)$
  - $S_j = \text{AES}_K(A_j)$ with $K = \begin{cases} 
  \text{Ke} & \text{if application data} \\
  \text{Ki} & \text{if network data} 
\end{cases}$
  - $\text{ctxt} = \text{pld} \oplus (S_0 | .. | S_{n-1})$

- **MAC**: AES CMAC
  - $B_0(16) = 49 | 00...00 | \text{dir} | \text{addr}(4) | \text{cnt}(4) | 00 | \text{len}(1)$
  - $\tau = \text{MAC}_{K_i}(B_0 | \text{hdr} | \text{ctxt})$

- **Message**: $\text{hdr} | [\text{pld}]_K | \tau$

---

Application frame

Network frame
Attack: end-device disconnection

- $\text{Ke}^* = \text{ENC}_{MK}(01 \mid v^*)$
- $\text{Ki}^* = \text{ENC}_{MK}(02 \mid v^*)$
  with $v^* = y^* \mid \text{id}_S \mid x \mid 00..00$

- $\text{Ke} = \text{ENC}_{MK}(01 \mid v)$
- $\text{Ki} = \text{ENC}_{MK}(02 \mid v)$
  with $v = y \mid \text{id}_S \mid x \mid 00..00$
Attack: end-device disconnection

- Ke* = ENC_{MK}(01 | v*)
- Ki* = ENC_{MK}(02 | v*)
  with v* = y* | id_S | x | 00..00

- The end-device is “disconnected”.
- The NS cannot initiate a new session.
- The end-device may not expect replies from the NS.

If no reply is received within the next ADR_ACK_DELAY uplinks (i.e., after a total of ADR_ACK_LIMIT + ADR_ACK_DELAY), the end-device may try to regain connectivity by switching to the next lower data rate that provides a longer radio range. The end-device will further lower its data rate step by step every time ADR_ACK_DELAY is reached. The ADRACKReq shall not be set if the device uses its lowest available data rate because in that case no action can be taken to improve the link range.

LoRaWAN 1.0.2 specification, § 4.3.1.1, p. 17
Attack: replay or decrypt

- Ke = ENC_{MK}(01 | v)
  Ki = ENC_{MK}(02 | v) 
  with v = rnd_S | id_S | rnd_C | 00..00

- A_j (16) = 01 | 00...00 | dir | addr (4) | cnt (4) | 00 | j (1) 
  S_j = AES_K(A_j) 
  ctxt = pld ⊕ (S_0 | .. | S_{n-1})

- B_0 (16) = 49 | 00...00 | dir | addr (4) | cnt (4) | 00 | len (1) 
  τ = MAC_{Ki}(B_0 | hdr | ctxt)

1. Replay of ans = AES^{-1}_{MK}(rnd_S | id_S | addr | prms | τ_S) 
2. Reuse of rnd_C 

=> Reuse of Ke, Ki, A_j, B_0
Attack: replay or decrypt

- Consequences
  - (downlink) frame replay
  - (uplink) frame decryption:

\[
\begin{align*}
  \text{ctxt} &= \text{pld} \oplus S \\
  \text{ctxt}' &= \text{pld}' \oplus S \\
  \begin{cases}
    \text{ctxt} \oplus \text{ctxt}' &= \text{pld} \oplus \text{pld}'
  \end{cases}
\end{align*}
\]
Attack: replay or decrypt

- Consequences
  - (downlink) frame replay
  - (uplink) frame decryption:
    \[
    \text{ctxt} = \text{pld} \oplus S
    \]
    \[
    \text{ctxt}' = \text{pld'} \oplus S
    \]
    \[
    \text{ctxt} \oplus \text{ctxt}' = \text{pld} \oplus \text{pld'}
    \]

- \( \text{Pr[hit]} = 2^{-16} \)
- With \( n \) previous ans messages, \( \text{Pr[hit]} \approx n.2^{-16} = p \)
- The attacker iterates \( k \) times: \( \text{Pr[success]} = 1 - (1 - p)^k \approx k.p \)
- Complexity: \( k \approx 2^{16}/n \) to get \( \text{Pr[success]} \approx 1 \)
- 8 s/key exchange \( \Rightarrow \) 9.1 hours (with \( n = 16 \))

End-device (MK)

\[
\text{rnd}_C = x_0, x_1, \ldots, x_k
\]
\[
\text{rnd}_S = *, *, \ldots, y_k
\]
Attack: replay or decrypt

- Consequences
  - (downlink) frame replay
  - (uplink) frame decryption: $\text{ctxt} = \text{pld} \oplus S$
    $$\text{ctxt}' = \text{pld}' \oplus S$$
    $\text{ctxt} \oplus \text{ctxt}' = \text{pld} \oplus \text{pld}'$

- $\Pr[\text{hit}] = 2^{-16}$
- With $n$ previous ans messages, $\Pr[\text{hit}] \approx n.2^{-16} = p$
- The attacker iterates $k$ times: $\Pr[\text{success}] = 1 - (1 - p)^k \approx k.p$
- Complexity: $k \approx 2^{16}/n$ to get $\Pr[\text{success}] \approx 1$
- 8 s/key exchange $\Rightarrow$ 9.1 hours (with $n = 16$)

- Remark on the duty cycle
  - Not a security mechanism
  - Not applied in all countries
  - Not verified through the LoRa Alliance certification process

---

To fully test the End Device the DUT needs to run a test application (in a test mode) running on top of the MAC layer and must disable any duty cycle restrictions as well as suspend its normal application software. The LoRa Certification testing will not do any duty cycle testing. The required test...
Attack: targeting the NS

- Disconnection and “replay or decrypt” doable against the NS.

- Disconnection
  - The NS must keep track of a “certain number” of previous req messages. => Use of “forgotten” or “unknown” req messages.

- “Replay or decrypt”
  - $|\text{rnd}_S| = 24$ bits => $\Pr[\text{hit}] \approx 2^{-24}$
  - addr is “arbitrarily” generated => $\Pr[\text{hit}] \approx 2^{-49}$
  - The attacker chooses $\text{rnd}_C$ first (then the NS replies).
  - Use of $n$ req messages: $\Pr[\text{success}] \approx \frac{n}{2^{24}}$ (if addr is unchanged)

- Consequences
  - (uplink) frame replay
  - (downlink) frame decryption
Lack of data integrity

- Encryption in CTR mode
  - Change plaintext by flipping ciphertext bits => end-device or AS is deceived
  - Truncate encrypted payload => hide information from end-device or AS
  - Possible payload decryption under assumptions (easier in uplink direction)
Recommendations

- Constraints: keep interoperability between patched and unmodified equipment

- \( r_{nd_S} \) replaced with 24-bit counter (1 counter per end-device)

- \( \text{addr} = H(r_{nd_C} | r_{nd_S} | i_{id_C}) \)

- Key confirmation by NS (using an existing LoRaWAN command)

- Provide end-to-end data integrity (application layer)
Conclusion

- Low cost security => low power attacks
- LoRaWAN 1.0 published *without* security analysis

- Upcoming version: v1.1 (includes some recommendations related to v1.0)
- LoRa Alliance: call for a public review of LoRaWAN 1.1 from the academic community
Thank you
References

[LoRaWAN1.0] N. Sornin, M. Luis, T. Eirich, T. Kramp, O. Hersent. *LoRaWAN Specification* (Jul 2016), LoRa Alliance, version 1.0.2