Internet of Things 2024

A combined talk

Alessandro Bruni
Center for Information Security and Trust
IT-University of Copenhagen
2023

Sebastian Büttrich
2024

When it looks like this ...

It is by Sebastian

[or if it is marked like this, ed.]

Plan for this lecture

- Security Goals and Principles
- IoT threat modeling
- IoT security constraints
- Advanced security properties
- IoT security protocols

When you speak of security …

What do you mean?

What do you think others mean?

What are aspects of security?

Discuss!

When you speak of security …

IoT Security

Alessandro Bruni
Center for Information Security and Trust
IT-University of Copenhagen

About me

- Alessandro Bruni, brun@itu.dk
- Associate Professor at the ITU Center for Information Security And Trust
- Chapter leader of OWASP Copenhagen

- Interests: verification of security protocols
  - IoT, Voting, Identity

Trusselsvurdering (2019) – IoT

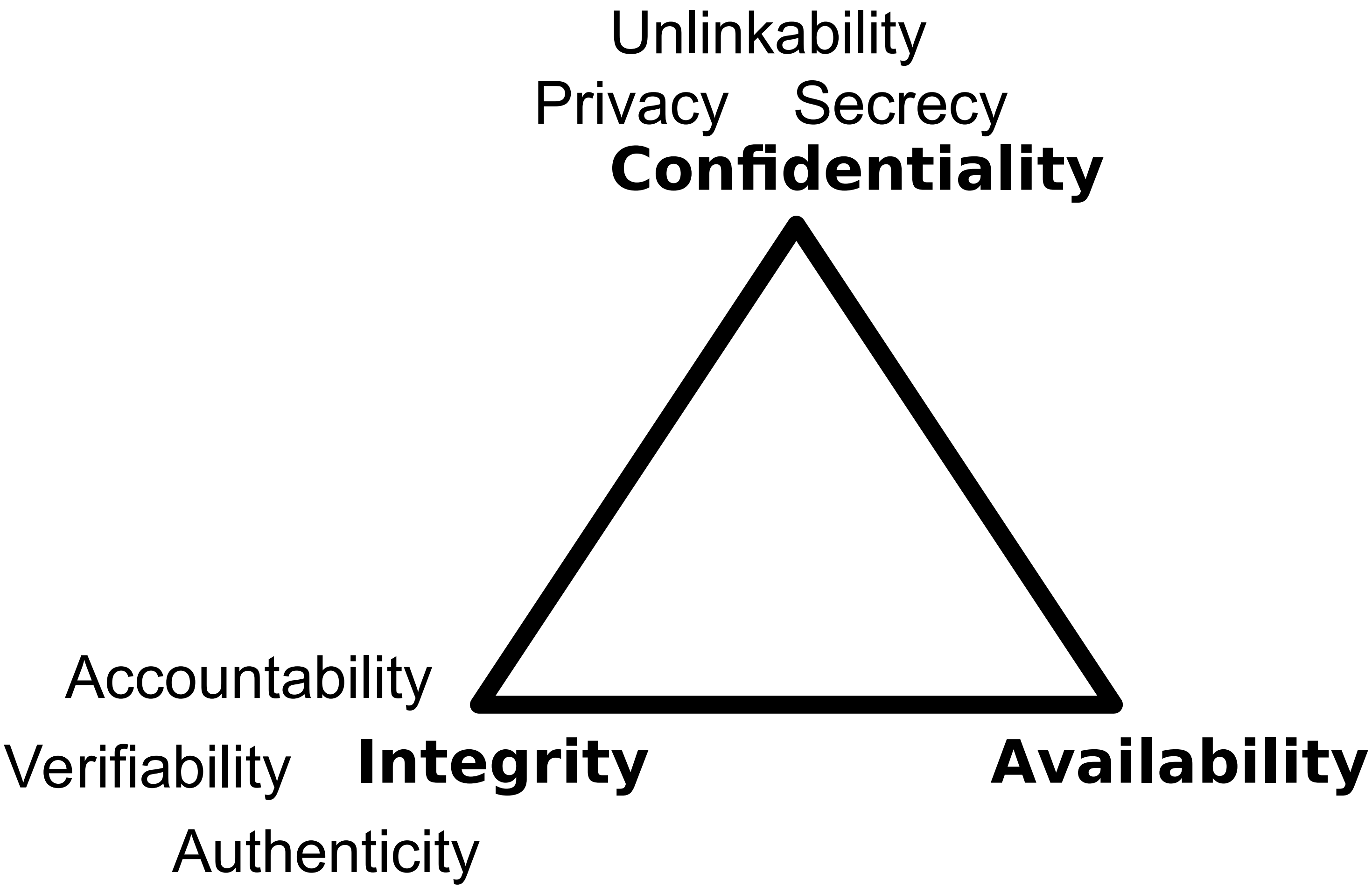
- Cyberangreb kan i stigende grad få konsekvenser i den fysiske verden
- Det stigende antal IoT-enheder kan føre til flere og mere alvorlige cyberangreb
- Et cyberangreb på en IoT-enhed kan påvirke enhedens funktion eller medføre en kompromittering af det netværk, som enheden er installeret i.
- IoT-enheder er ofte sårbare, fordi de bliver udviklet med et specifikt formål for øje, og de netærksfunktioner, som gør det muligt for enheden at kommunikere via internettet, er kun sekundære funktionaliteter.

https://fe-ddis.dk/cfcs/publikationer/Documents/Cybertruslen-mod-Danmark-2019.pdf

FORSVARETS EFTERRETNINGSTJENESTE

System, Stakeholders, Assets, Vulnerabilities, Threats, Countermeasures

The CIA Triad



Security Goals

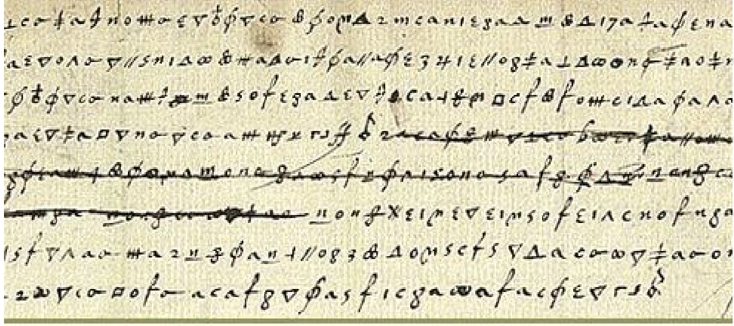
Confidentiality

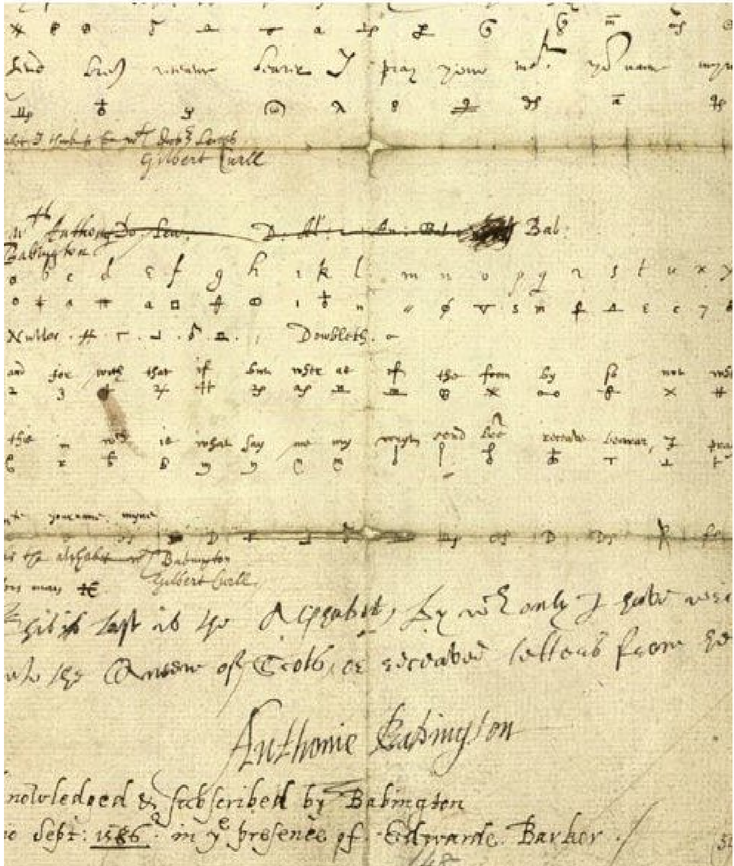
- Attacks: *eavesdropping, man-in-the-middle*

Integrity

- Attacks: *masquerading, message tampering, replaying*

- July 17, 1586: Thomas Phelippes confounds the Babington plot to murder Queen Elisabeth and install Queen Mary as regent.
- He intercepted and decrypted a letter, then added:
- “I would be glad to know the names and qualities of the six gentlemen which are to accomplish the [deed], ...”

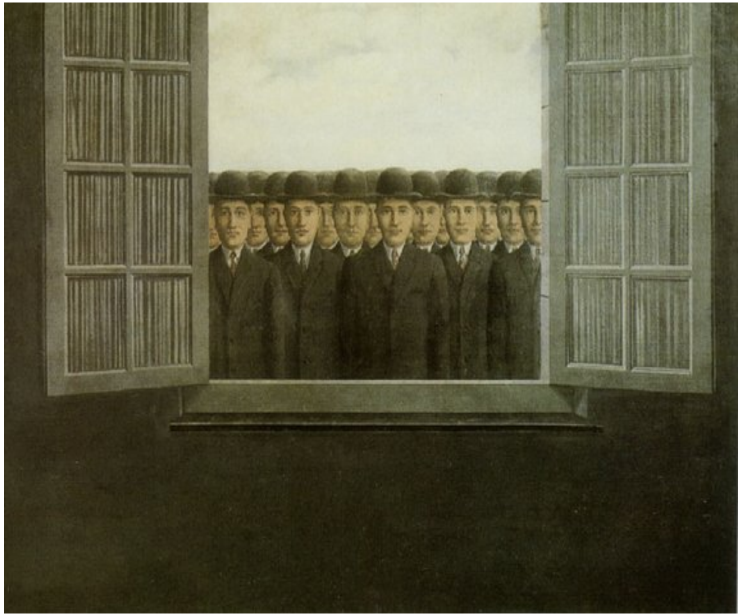




Availability

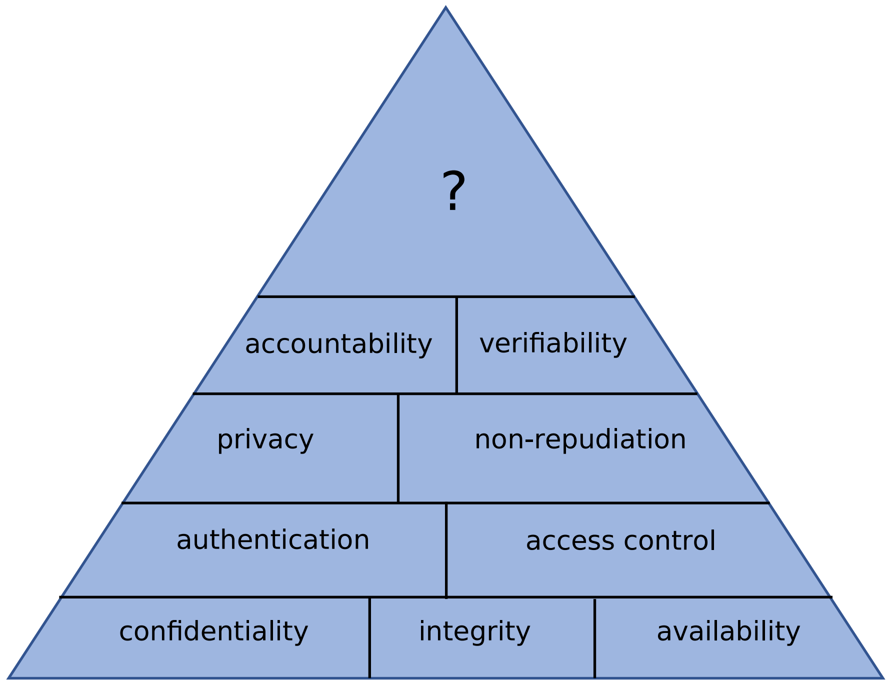
Attacks: *Denial of Service, distributed denial of service*

- December 19-21, 2018: Gatwick drone disruption cost easyJet nearly $20 million



Security Goals

- What does it mean that a system is secure?



Remark

Some of these goals might be in contradiction with one another

e.g.

the wish to control

vs

the wish for privacy

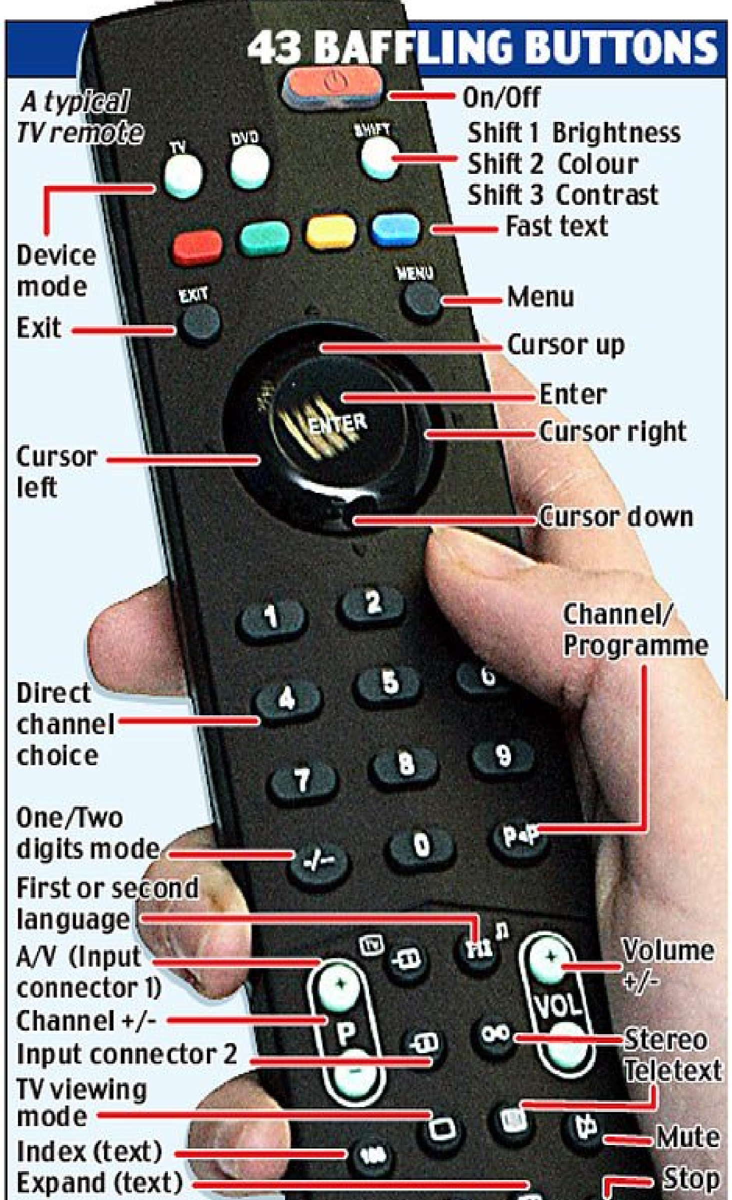
sources

Security is impossibly hard

- You must defend against **all** possible attacks
- Adversary needs to find just **one** attack that works
- No perfect security
  - (...all possible attacks)

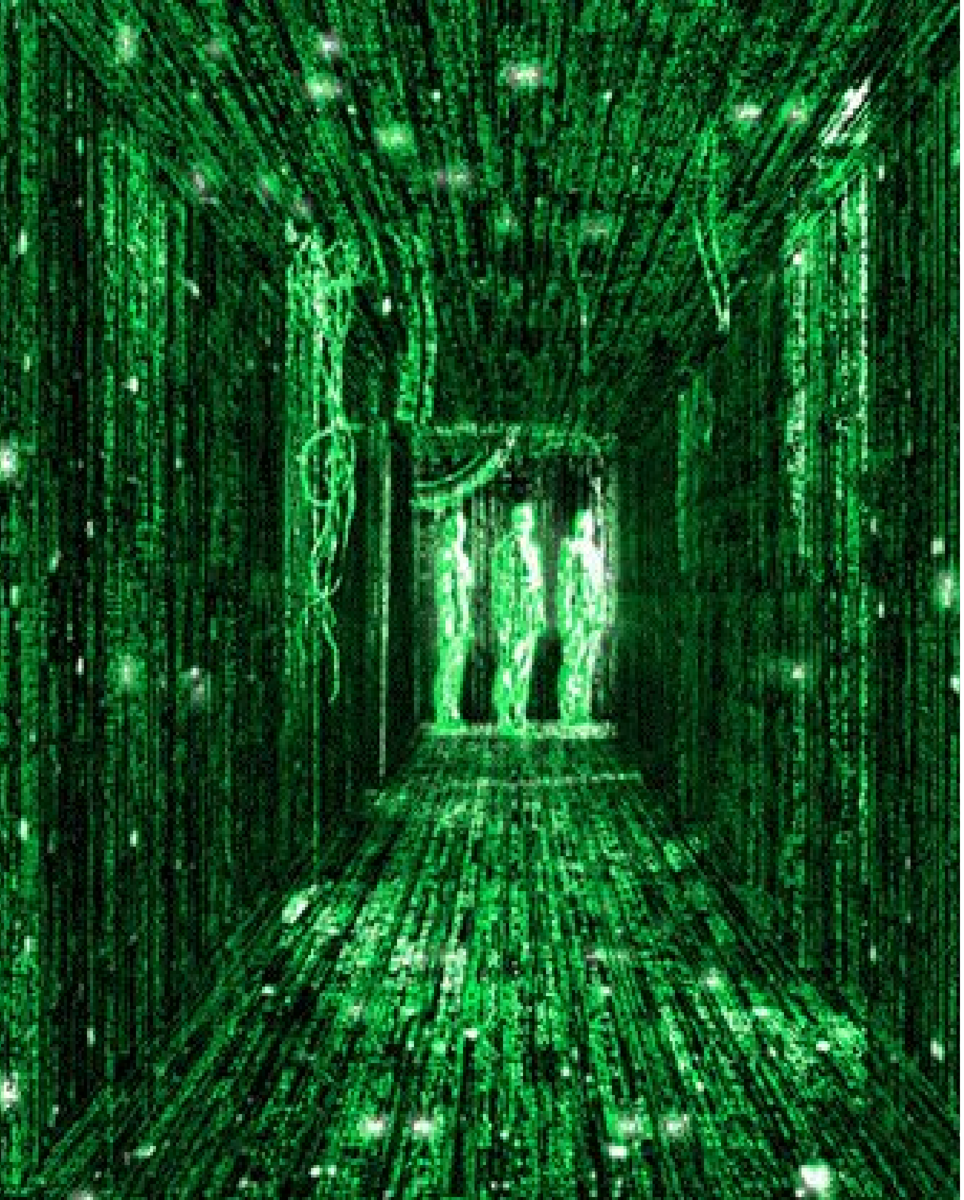
- Security is measured in the resources required of the adversary

Security Principles



Economy of Mechanism

- "Keep it simple."
- aka. "simplicity"
- General engineering principle: Complex designs yields complex failure analysis.



Open design

- “The security of a system should not depend on the secrecy of its protection mechanisms.”

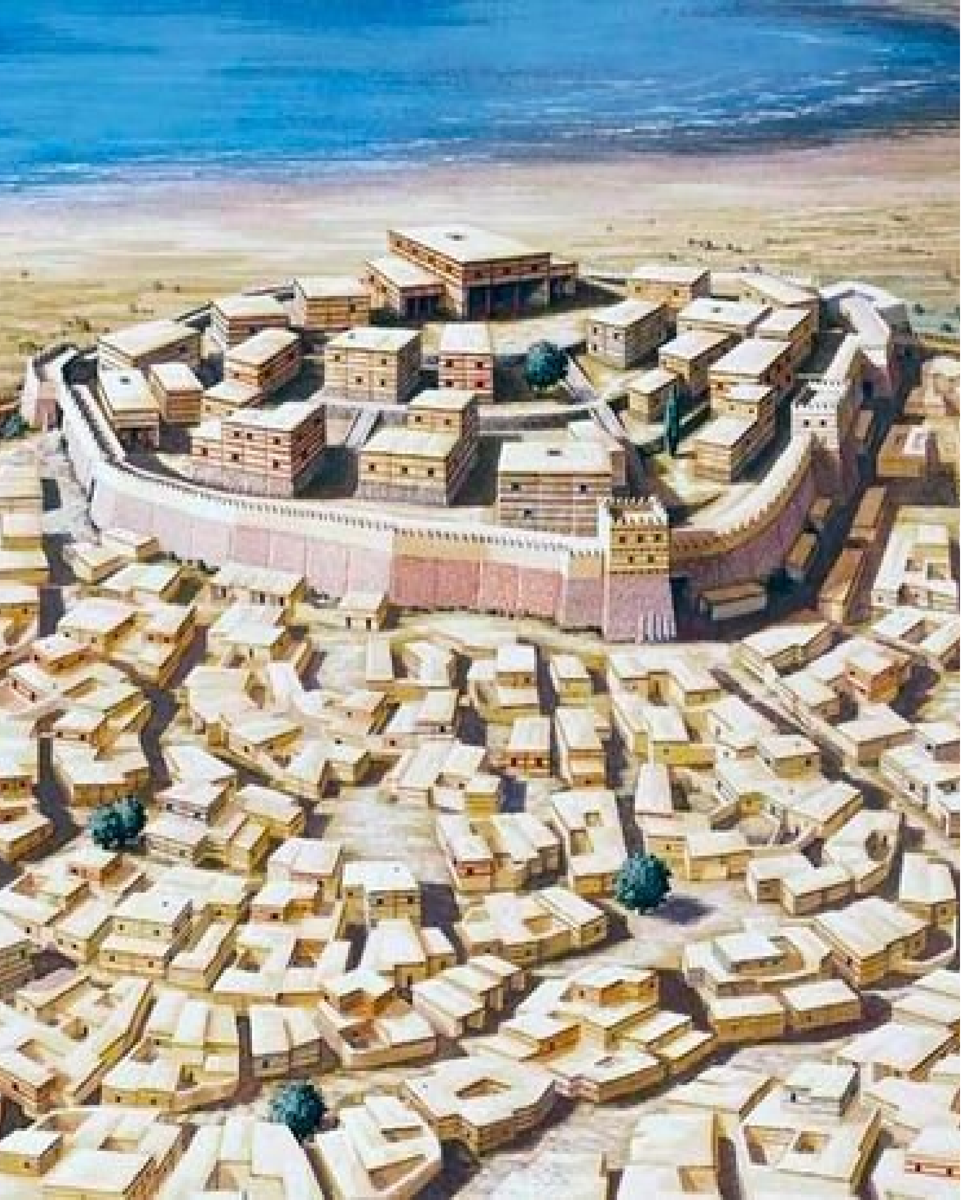
  [Obscurity is not Security, ed.]

- aka “Kerckhoff’s principle”

- The adversary knows the system (Claude Shannon).

- Systems are hard to build—more scrutiny, less defects.

- Hard case: DRM. The user has the device. Sony compromises(!) consumers machines in 2005.



Minimum exposure

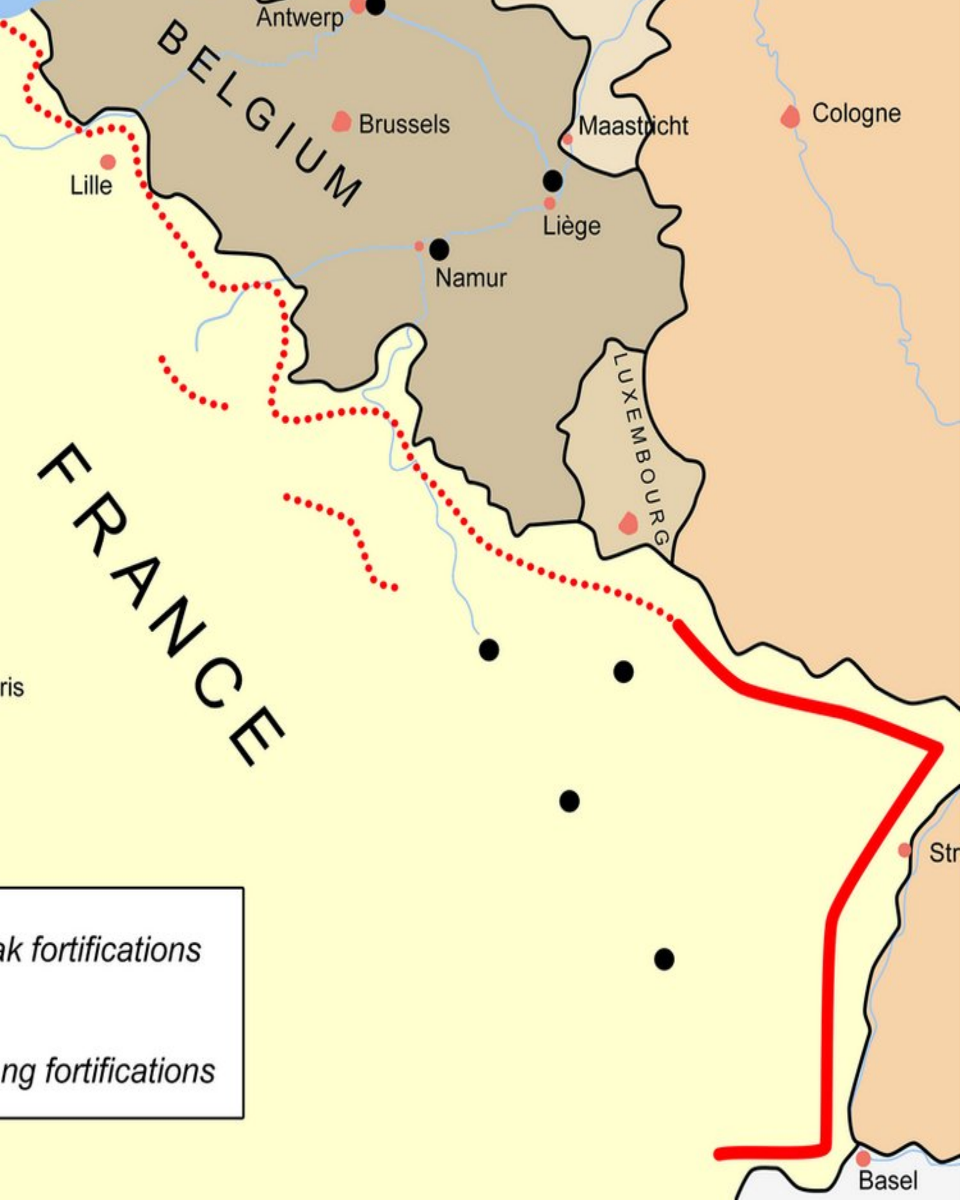
- “Minimise the attack surface a system presents to the adversary.”
- Reduce external interfaces (If you don’t need it, turn it off.)
- Limit information
- Limit window of opportunity.

Least privilege

- “Any component should operate using the least set of privileges necessary.”
- I don't have access to ITU mail servers.
- Keynote does not run as root.

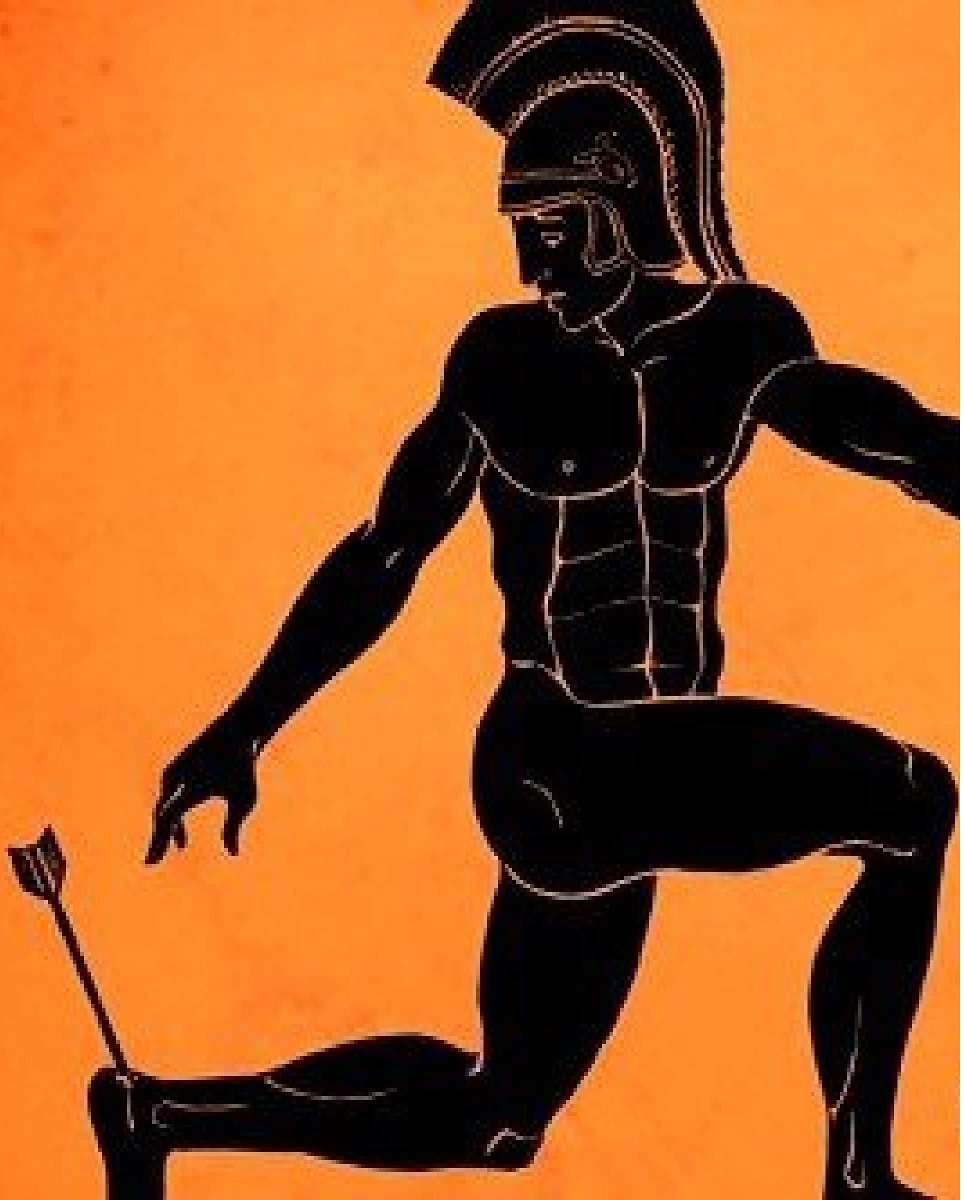
Fail-safe defaults

- “The system should start in and return to a secure state in the event of a failure.”
- Whitelist, blacklist.
- If you lost connectivity to the authentication server, don’t let anyone in while it’s down.
- E.g., whitelist ports for firewalls



Complete mediation

- “Access to any object must be monitored and controlled.”
- The Maginot-line: strong fortifications not extending all the way did not help.
- e.g., OS access control to files can be circumvented if you have access to the physical disk. (Use crypto, then.)



No single point of failure

- "Build redundant security mechanisms whenever feasible."
- aka "defence in depth."

  [if one line breaks,

  have a 2nd one, ed.]
- Key technique: separation of duty

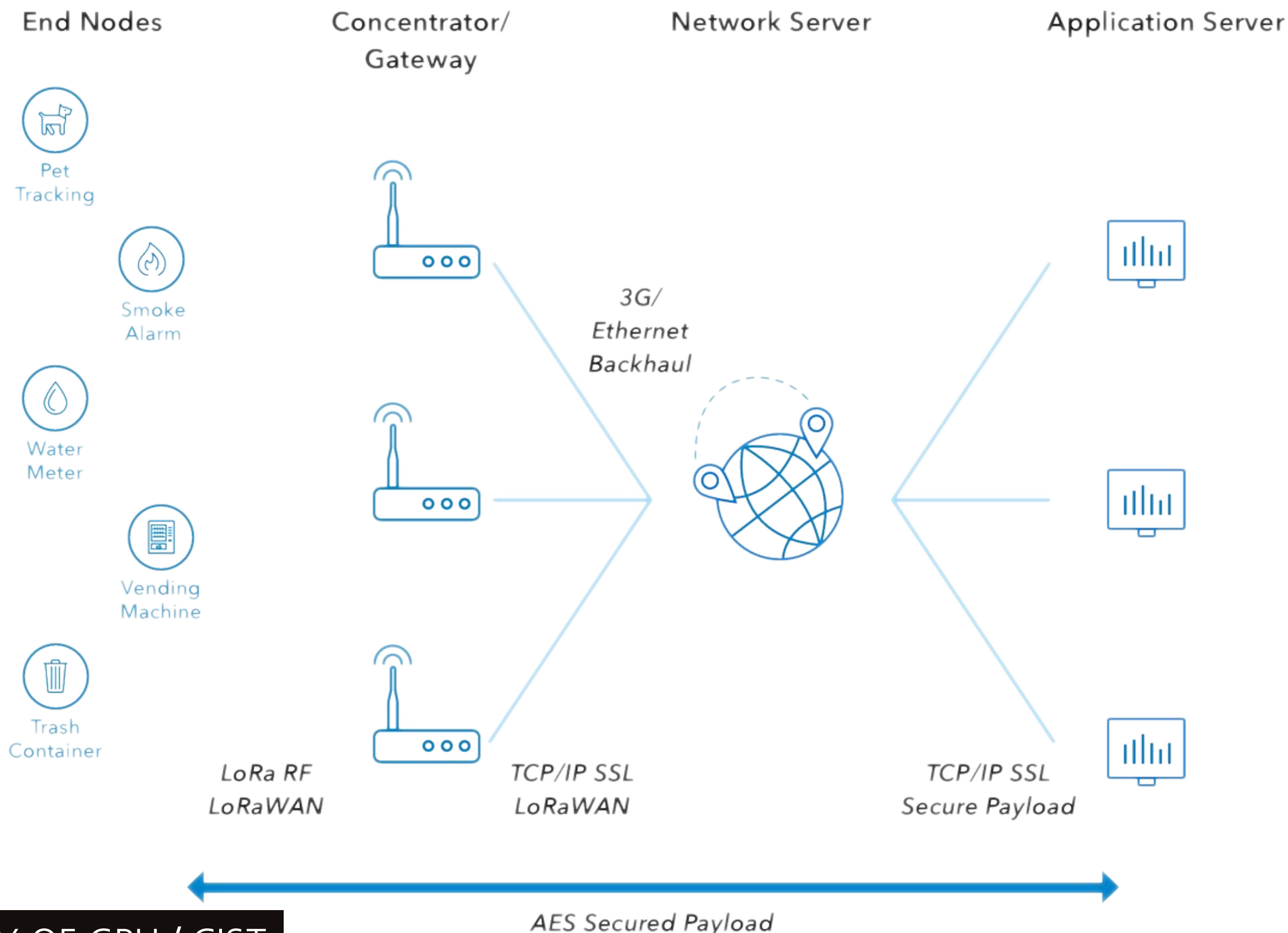
Psychological acceptability

- “Design usable security mechanisms”
- …let users circumvent them
- Help the user to make the *right* choice

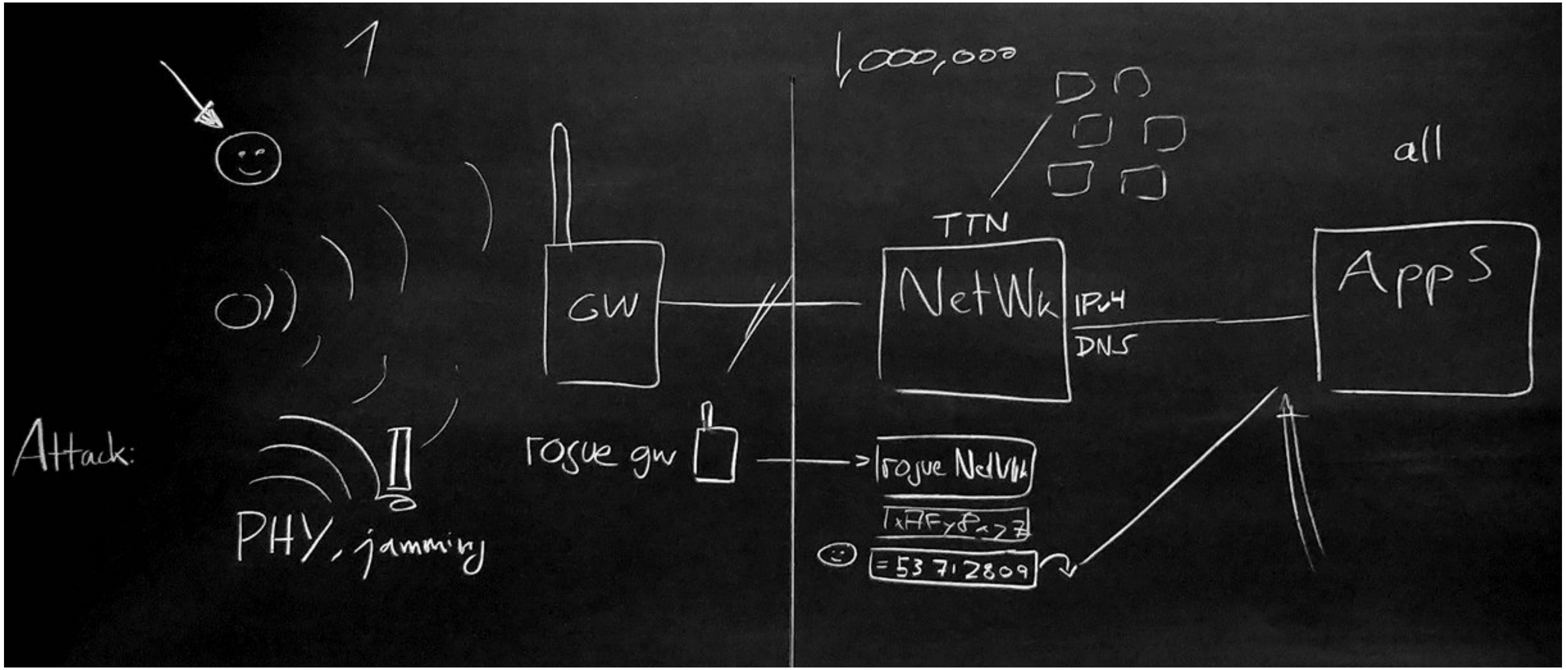
  [what is not easy
  will be circumvented, ed.]

IoT Threat Modeling

IoT threat modeling



Hypothetical Scenario



IoT Attack Surface Areas

END DEVICES GATEWAYS NETWORK SRV APPLICATION SRV

Device Firmware
- Hardcoded credentials
- Sensitive information disclosure
- Sensitive URL disclosure

Local Data Storage
- Unencrypted data
- Data encrypted with known keys
- Lack of data integrity checks

Administrative CLI
- Injection
- Denial of Service
- Unencrypted Services
- Poorly implemented encryption
- Test/Development Services
- Buffer Overflow
- UPnP
- Vulnerable UDP Services
- DoS

Network Traffic
- LAN
- LAN to Internet
- Short range
- Non-standard

-party Backend APIs
- encrypted PII sent
- crypted PII sent
- vice information leaked
- cation leaked

Vendor Backend APIs
- Inherent trust of cloud or mobile application
- Weak authentication
- Weak access controls
- Injection attacks

U
- Update verification
- Maliciou
- Missing
- No man

Ecosystem Communication
- Health checks
- Heartbeats
- Ecosystem commands
- Deprovisioning
- Pushing updates

one
- Decommissioning system
- Lost access procedures

Cloud Web Interface
- SQL injection
- Cross-site scripting
- Cross-site Request Forgery
- Username enumeration
- Weak passwords
- Account lockout
- Known default credentials
- Transport encryption
- Insecure password recovery mechanism
- Two-factor authentication

Mobile Application
- Implicitly trusted by device or cloud
- Username enumeration
- Account lockout
- Known default credentials
- Weak passwords
- Insecure data storage
- Transport encryption
- Insecure password recovery mechanism
- Two-factor authentication

Source: https://www.owasp.org/index.php/IoT_Attack_Surface_Areas

The OWASP Top-10 Bingo

Source: https://www.owasp.org/images/1/1c/OWASP-IoT-Top-10-2018-final.pdf

Certification (Standards)

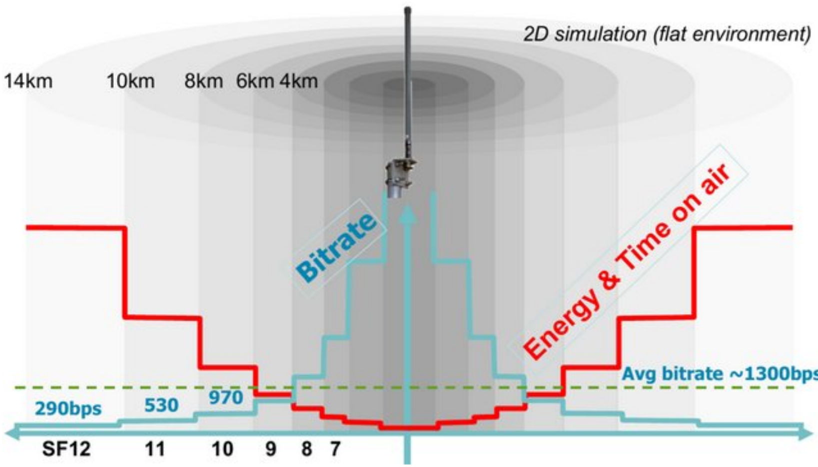
FIPS, IETF, Common Criteria (Protection Profiles) etc.

Good source of inspiration

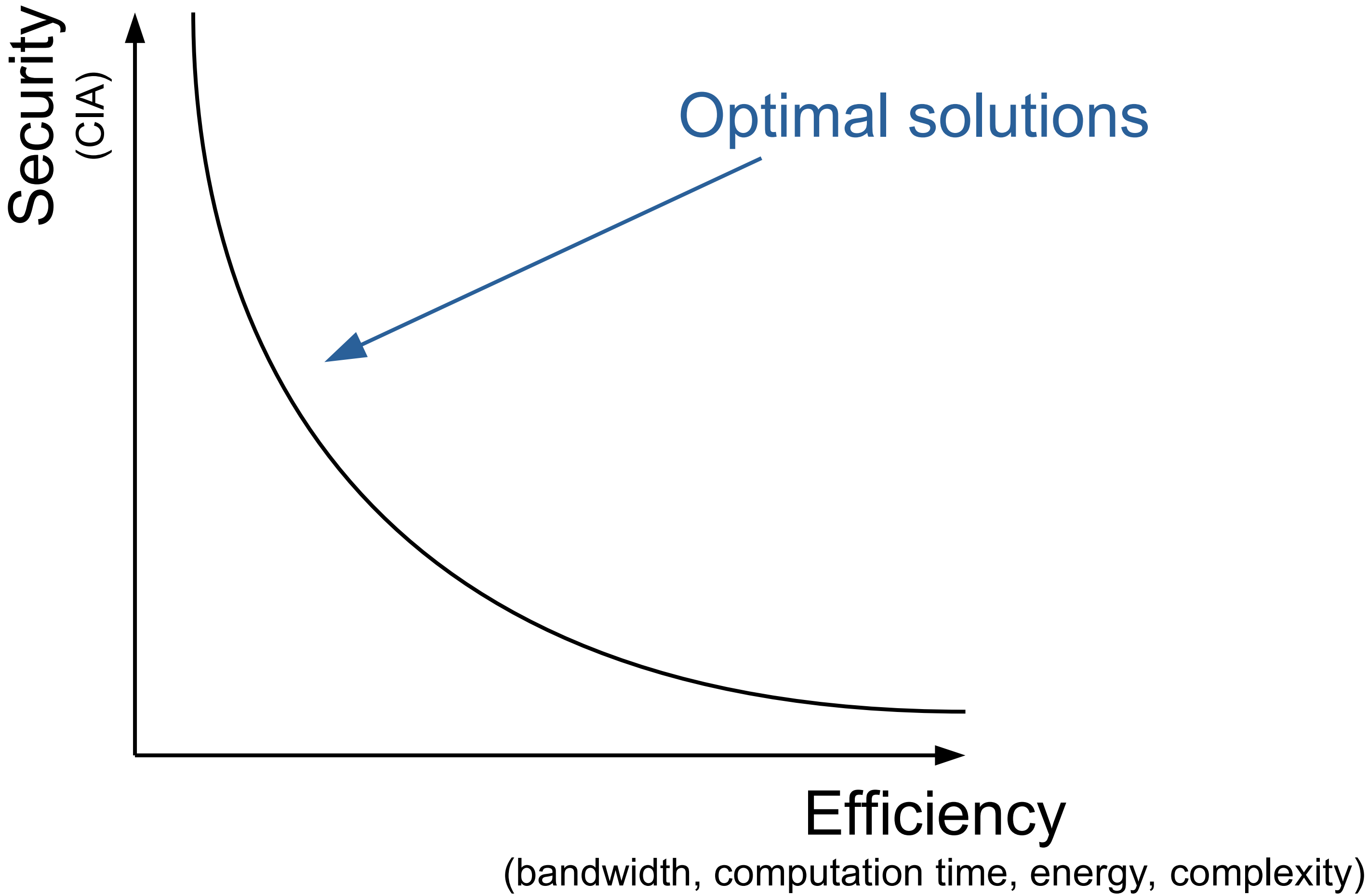
Internet Engineering Task Force

Federal Information Processing Standards, by
National Institute of Standards and Technology (NIST)

IoT Security Constraints



Pareto Frontiers in IoT



Does your message fit the frame?

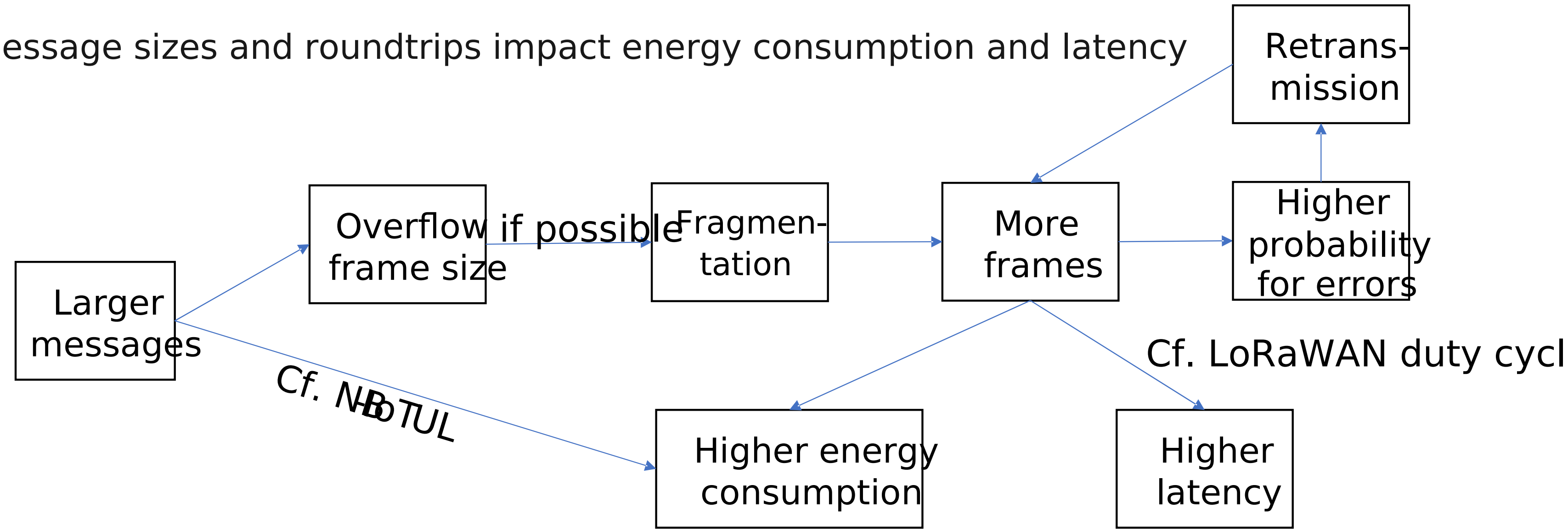
MTU size examples

MTU size (bytes)	Technology
12	Sigfox
16	CoAP Blockwise
32	CoAP Blockwise
47 (UL) / 49 (DL)	6TiSCH join protocol over proxy
51	LoRaWAN DR0-2 (excl. HC)
64	CoAP Blockwise
102	IEEE 802.15.4 (incl. frame overhead)
115	LoRaWAN DR3 (excl. HC)
128	CoAP Blockwise
140	SMS
. . .	. . .
222	LoRaWAN DR4- (excl. HC)

TLS 1.3 client hello: 207 bytes

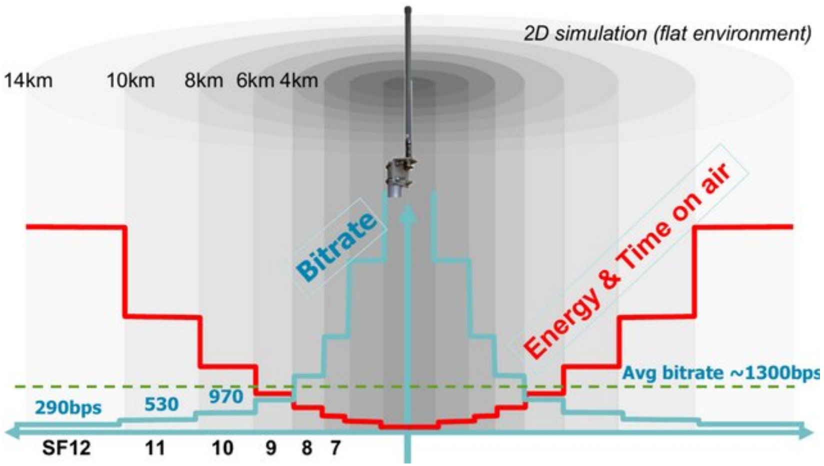
Constrained Characteristics

Message sizes and roundtrips impact energy consumption and latency



LoRaWAN (1)

- LoRaWAN employs unlicensed radio frequency bands
- Uses the 868 MHz ISM band in Europe regulated by ETSI EN 300 220
- Time-on-Air: The amount of time that the antenna is radiating power to transmit a packet
- After every transmission, there is a Back-off time period called Duty Cycle
    - Typical Duty Cycle in Europe is 1%
- Also, due to the regulations, the maximum payload size is limited for each LoRaWAN DataRate configuration

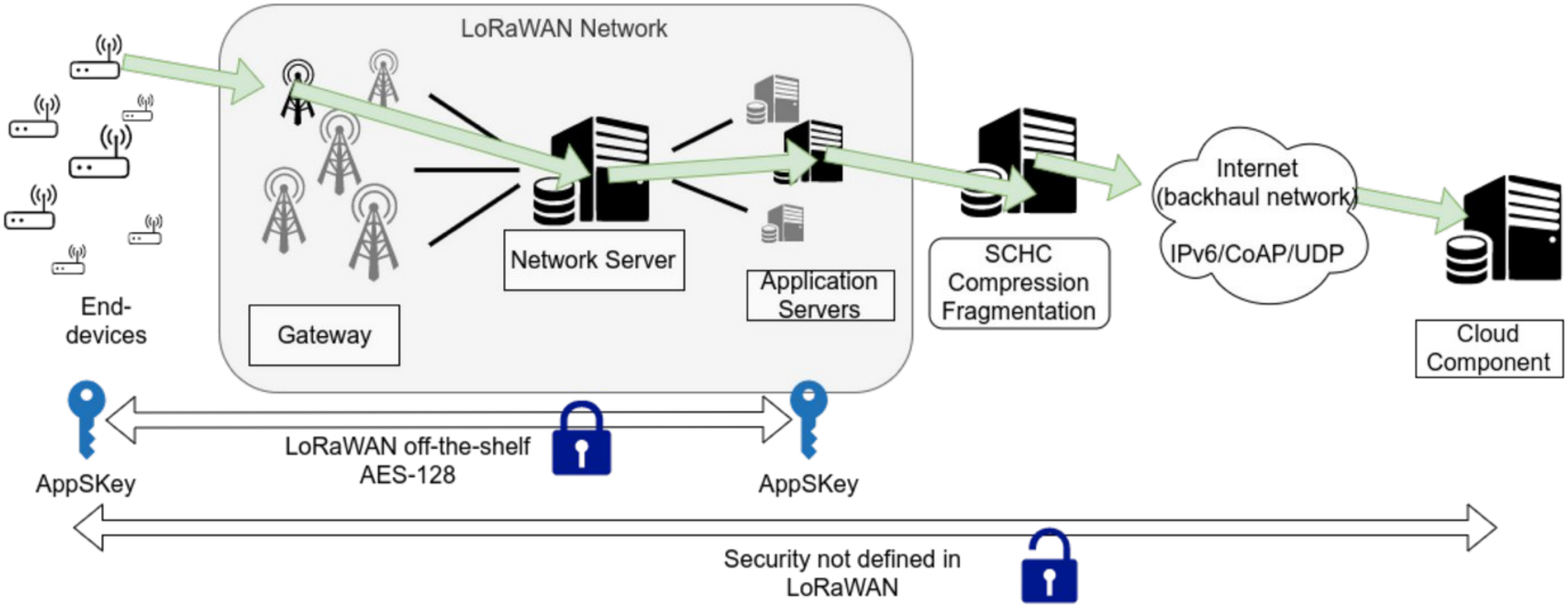


DataRate	M	N
0	59	51
1	59	51
2	59	51
3	123	115
4	230	222
5	230	222
6	230	222
7	230	222
8:15	Not defined	

Table 7: EU863-870 maximum payload size

LoRaWAN (2)

- LoRaWAN (v1.0) security employs a preprovided root key: AppKey. After deployment, a pair of session symmetric keys are derived: AppSKey and NwkSKey. These keys employ AES-128.
- Security outside of the LoRaWAN network is not defined in LoRaWAN specification.



Some more

LoRa/ LoRaWAN

specific aspects

Keys and key provisioning

Dumb role of gateways (a benefit!)

Physical layer: possibility of jamming

**Combined MAC/Physical layer:
battery depletion attacks**

Cloning of devices

Replay attacks

User interfaces of devices (often Bluetooth/Wi-Fi) / default credentials

Computing an RSA key

```
$ time ssh-keygen -N dummy -f dummy
Generating public/private rsa key pair.
Your identification has been saved in dummy.
Your public key has been saved in dummy.pub.
The key fingerprint is:
SHA256:nNTlC/O6inpKPaZ9CATsyYCHr2XIlnU0uJWEZTpWkNM albr@oersted
The key's randomart image is:
+---[RSA 2048]----+
|.o .XB.     .    |
|+ +*+E.  . o     |
|.Bo*=   . + .    |
|.+O.o  o . + .   |
|.+ .    S   o    |
|.   ..      .    |
|    ..+.  .      |
|   . +oo.  .     |
|    ++o....      |
+----[SHA256]-----+
ssh-keygen -N dummy -f dummy  0,23s user 0,00s system 97% cpu 0,240 total
```

Speed on my 3Ghz x86 processor, how fast can this guy do it?

6187.42 MIPS (per core) vs. 40 MIPS

Advanced Security Properties

Dissecting Confidentiality

Secrecy – the information is only known by a restricted set of authorized principals

(examples: keys, medical records, readings of a gas or heat meter)

Dissecting Confidentiality

Privacy – it is impossible to distinguish whether one or another piece of information was exchanged

Follow the flow

Data analysis (water)

00000100000006c7c0149b5037e000004ff070000000004ff0802000000041402004448407318010000b26d4448025d464804ff832c0bb8d3786b91aa6ced39295818d43bc85f1340de368b013e0d00
0000030000000000000000000000000000000000000000000000000000000000000000000000000000000000000b99200
00000111000000a2b88f40151e38572a961707d594662785894ba5e3ee1c9e5f101770f2ea97a724e9e2c8302f87dbd0ad50d43feffafe5d00
00001f0100000092ae1f2859d0c6a85805c3ab4350fe25380316990c3139f8a37ada0a512bafc4740fc977402cbc4a3c9d42000efa7db8c200
0000010000000ba1f000000108f00000000000000100000000000000000170000000000000000000000000000000000000000000000000364a00
000003000000099d40000006b6400000000000000000000000000000000000500000000000000000000000000000000000000000000000acb100
00000131000000815b19e5d1a80261212b5eb579ef6a8f496c93a247a39882c8e8ce600562201758f19977fa0319c0eca31c38629302221b00
00000701000000c9392582b3b7292c7416d6ac1841a1790d26e65c4a4ad539783ea1c6e15849488b33653123af365930b61b42041a22eb26a5
0000010000000001b90149b5479f000004ff070000000004ff0803000000951502004448407392010000b26d4448025d464804ff832c0295a51d7e1f2145784c28c5ac9d736bbf19c42bb2225a514c00
000003000000006e010000003123000000000000000000000000000000000350000000000000000000000000000000000000000000000290700
00000111000000c18e7ef60e4ec7de6db40c777dccc45a9a1c990e4d677a94012aaff0592ca3af554cec696dd195f2936fe5b84c54364db600
00000f01000000daefae2730777a8e98802bf3321249a3eda60284a249b41004ec0f810241615b0247b62b4988b56ab563f632cf4f02268a00
0000010000000ce40000000bdac0000000000000000000000000000000000f000000000000000000000000000000000000000000000902c00
000003f10100005a9ccb03405b6e0a0334f60397d2791c07c3cd640d615ab6aa1bbe7b466f91c05e0441312fd465a88be300920da2e97f1c00
00000101000000044d7e6aba2fb1f144f23aa07d89a2cb7b5a85d553781afbb0be6817695b77ed4e494dd1be14726f2c5671ee196848e726c00
00000700000009c9f0149b5ecd8000004ff070000000004ff08040000009a1504004448407398010000b26d4448025d464804ff832cc5f099
0000010000000c8f10149b5c054000004ff070000000004ff080400000078150400444840739801000b26d4448025d464804ff832cbeca99c973ce90b66f54c4ee9c976ad27bf5d982ada37879d000
00000311000000a58819edca7b023c6c9b5aa928f94e000e611a44eba2cf5b86e82bbbdc7b8e3b75ae02477768249afefa69748333f5c8a900
00000101000000d77abbc32d7bde73924cf03ef2986286d256caf4daacdcb007936fe75f527cfe93da5375633416e95e99951aa52b4abb3100
00003f00000000af69000000a9cb000000000000000000000000001000000e80000000000000000000000000000000000000000000000006a1b00
00000100000000546e0000002c8c0000000000000000000000000000000000e200000000000000000000000000000000000000000000005df500
000003310000009d86007c1289602cfd6b654109b9524bca5d30760c88e6581fcea7e9c165f9aa0b3c7a8815b5889d276614129091806 5eb00
000001010000003c9ab42f3c29ba8e18b16d1343da5061a3b0d3a487de295583ab5a3b85968603d2663fe75887d964d496c2d63adf6d716300
00000700000000bcf30149b5819f000004ff070000000004ff0805000000a61505004448407394010000b26d4448025d464804ff832cd9689d
00000210000000 6a56d1db1ebbb064c901122802cd47e2539cad80136610984ae134ecdddb4b25cce8e20d15e97b9de6130b5de2306147c69d1ecedeaf42c36dc9be33901eacd03fb3165415762bb300

Data analysis (no water)

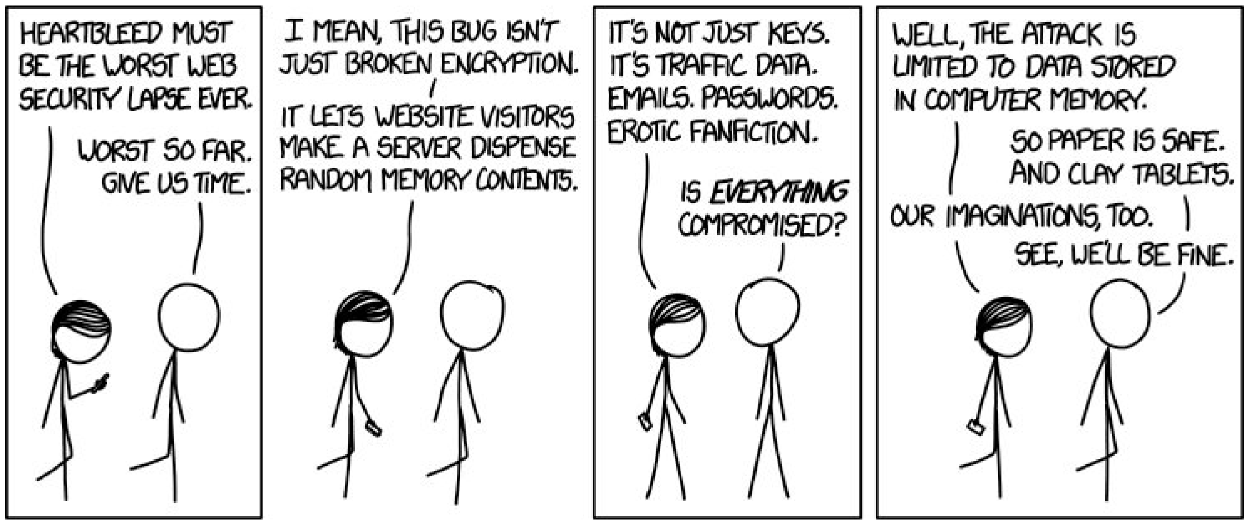
0000010100000095f7161c077ee652c111cb7978537e626e65b51c8df33c678c0d59e4b761c1ecee5cbfda31cbd0b1a9acac2ce08d85aeda251e0d19acdfcb5189da47f9d8e51e0a03439a59f4049f00
000003000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000b99200
000001110000007d0c606e293cacadbf08151db41ce8e0a6fa94e9fbbbe8ab287f8d427e158dba73301f546c1d08322d6cf774cbc277973a00
00000f0100000063 0999a8dc4f72555b69385683179d1ac41204d3424d6b67b9aa18faaf5e7acc8e1b1cdd9bce8baf78c0b35675ac3af64000
000001000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000688e00
000003000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000999200
000006700000005cc3b0ec83ca1af9d8d53827faeb95f93da71f2bda5801b7081dce9ecc63a0932e8090a32a4ec2f28b95546bdbdf1b32c6d3
00000100000000fa150149b55c7e000004ff070000000004ff080b00000004140b004448407318010000b26d4448025d464804ff832c6901d3786b91aa6ced39295818d43bc85f1340de368b016cbb00
00000211000000a2b88f40151e38572a961707d594662785894ba5e3ee1c9e5f101770f2ea97a724e9e2c8302f87dbd0ad50d43feffa47cf00
0000fe01000000ca4e1f2859275da85805c3ab4350ee25380316990c313973a27ada0a512bafc4750fc977402cbc4a3c9d42000efa7dcdb400
00000331000000033119e5d1712161212b5eb579ef6a8f496c93a247a398813a544031bbb7dd0f54557432e34045534e38669f5c26d15b2700
000001010000003f6a24cb06e1042c741229ab1841a17909d9ee5e4a4ad5a79f802fd317cdc7c2869788c657a822c8cf6b29e1c52cdd43a900
00000700000000fa150149b55c7e000004ff070000000004ff080b00000004140b004448407318010000b26d4448025d464804ff832c5d87a5
00000211000000b9f17fbfbbcb9ade6db0f3707dccc45a9ee391054d677a931521afb4116cd0b7544cecdb0095dde0ce29adbcb3d71ab14aa51d7e1f2145784c28c5ac9d736bbf19c42bb2225af48e00
00000101000000e37aae273090918e98802bf3321249a3eda60284a249b42f04ec0f810241615b0247b62b4988b57ab563f632cf4f02160000
00000f000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000251000
000001000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000688e00
000003f1010000aca9cb03402f6e0a0334f60397d2791c07c3cd640d615aadaa1bbe7b466f91c05e0441312fd465a88be300920da2e9e75c00
00000101000000a7d6e6aba2961f144f23aa07d89a2cb7b5a85d523781afa90be6817695b77ed4e494dd1be14726f2c5671ee196848ebbc500
00000700000000fa150149b55c7e00000cff070000000004ff080b00000004140b004448407318010000b26d4448025d464804ff832cae4f99
00000111000000de0d18a47f041b3c6c97a5ae28f94e000a9e124feba2cfb492e32bff943bfd2374ae02f51a2c6c98a3bc21707cb0d977e999c973ce90b66f54c4ee9c976ad27bf5d982ada378c7d700
00000301000000d77abbc32d7bde73924cf03ef2986286d252caf4daacdcb007936fe75f527cfe93da5375633416e95e99951aa52b4a6a2d00
000001000000000000000000000000000000000000000000040000000000000000000000000000000000000000000000000000000001688e00
00001f00000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000122ae00
00000131000000ca12007c125fa82cfd6b654109b9524bca5d30760c88e6521fcea7e9c165f9aa0b3c7a8815b5889d27661412909180597600
00000301000000529bb42f3c18998e18b16d1343da5061a3b0d3a487de296083ab5a3b85968603d2663fe75887d964d496c2d63adf6d30ea00
000001000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000688e00
00000700000000fa150149b55c7e000004ff070000000004ff080b00000004140b004448407318010000b26d4448025d464804ff832c61aa9d
00000111000000055c894a216943a658be8e5844d06a62923199de8ae5f0d9a66d46786634d1b2d60bf6ba52aeed20c3bcb9dc0da6ba724f9d1ecedeaf42c36dc9be33901eacd03fb31654157618c100

Dissecting Confidentiality

Unlinkability – it is impossible for an observer to link multiple sessions executed by the same entity [Note: there's conflict here again! ed.]

Perfect Forward Secrecy

After a compromise, all previous sessions are maintained secure. I.e. revealing of long-term keys will not give any information about the session keys to the attacker.



Post-compromise Security

- What to do after a compromise?
  - Throw away the device?
  - Reset new keys?
  - Keep it and hope for the best?
- Post-compromise Security:
  ”Assume the attacker has access to your device* and can sign, encrypt and decrypt all your data for a period of time, the device should be secure again after that period.”

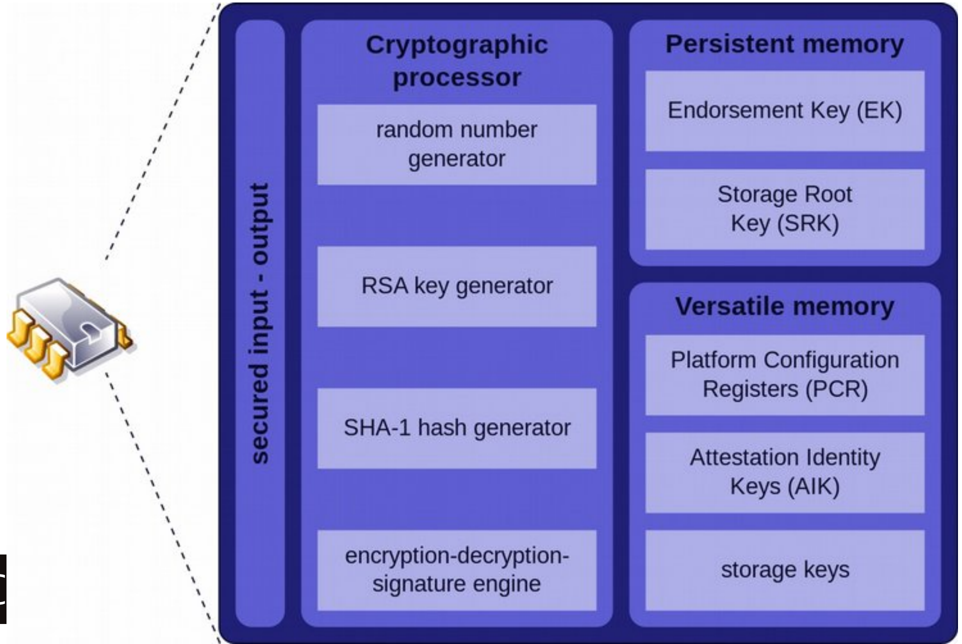
Notable mentions:
- Signal (WhatsApp, Telegram etc..)
- TLS 1.3 (Weak Post-compromise)

* not necessarily to the keys

Hardware Security Modules (HSMs)

A Hardware Security Module (HSM) is a hardware module that provides a secure storage and execution environment for cryptographic APIs.

Examples: TPM, ARM TrustZone, Intel SGX



IoT Security Protocols

A_S | A | S

A_S: c

A: k_A, g^{k_S}

S: c, g^{k_A}, k_S

$A_S \xrightarrow{g^c} A$

$A \xrightarrow{\left(g^{k_A}, (g^c)^{k_A}\right)} S$

sample r, m

$A \xleftarrow{\left(\overbrace{g^{c^r}}^{d_1}, \overbrace{m \cdot g^{k_A c^r}}^{d_2}\right)} S$

$$m = \frac{d_2}{(d_1)^{k_A}}$$

sample r'

$A \xrightarrow{\hspace{3em}} S$

$$\left(\underbrace{g^{r'}}_{d_3}, \underbrace{m \cdot (g^{k_S})^{r'}}_{d_4}\right)$$

check $m = \dfrac{d_4}{(d_3)^{k_S}}$

LoRaWan security

LoRaWAN networks are spreading but security researchers say beware

IOActive security researchers say LoRaWAN networks are vulnerable to cyber-attacks despite boastful claims about the protocol's security features.

https://www.zdnet.com/article/lorawan-networks-are-spreading-but-security-researchers-say-beware/

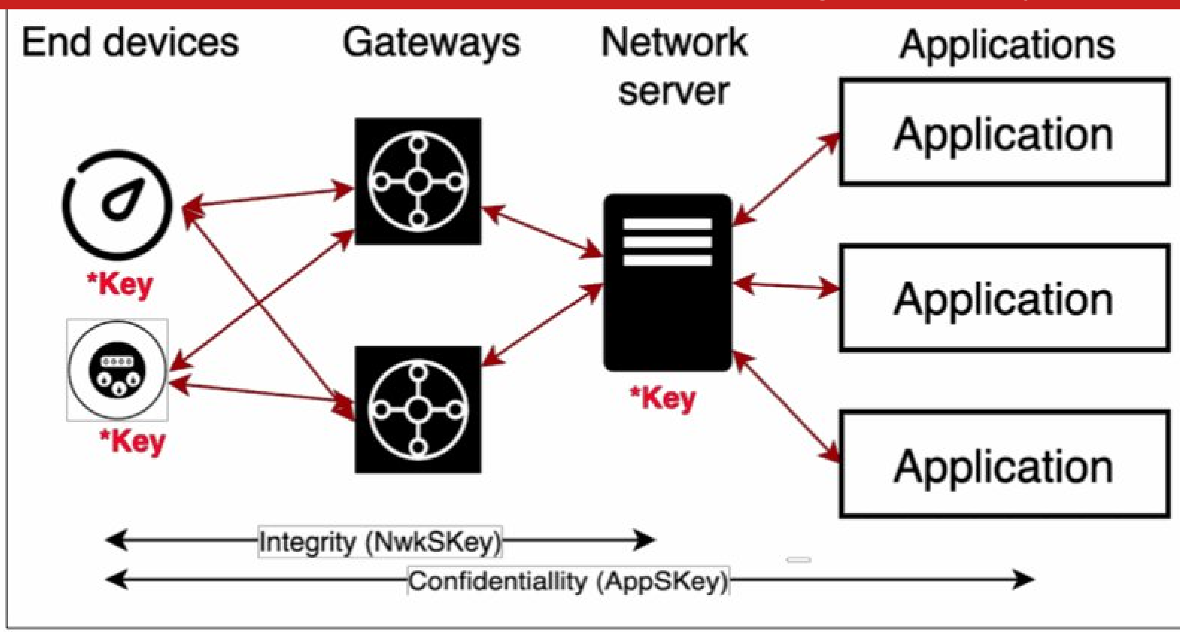


Figure 2. Session Keys and Functions in LoRaWAN v1.0.3

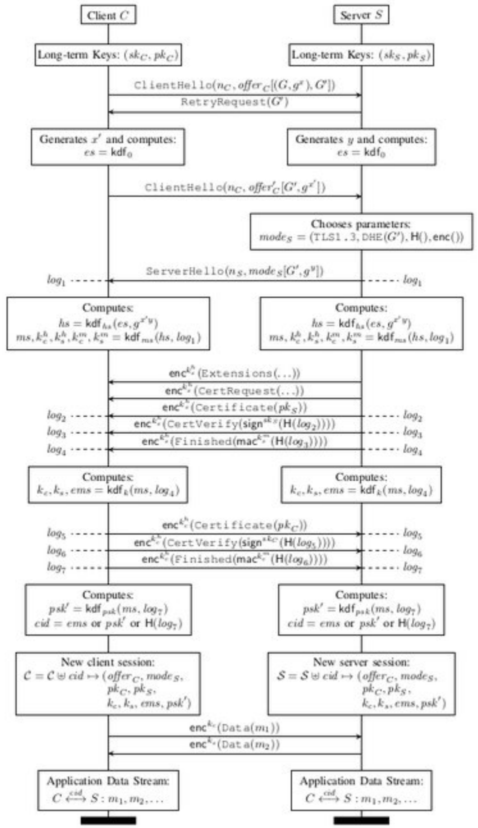
The Key Distribution Problem

In general, **connecting *n* nodes** in a network requires **O(n^2) keys**. With **public key crypto** solves the problem with only ***n* keys**.

Current IoT practice defies this approach because it's too expensive. So all symmetric keys are registered at the service provider.

Limitation: won't allow easy communication between devices.

TLS 1.3 Key exchange



Key Derivation Functions:

$\mathsf{hkdf\text{-}extract}(k, s) = \mathsf{HMAC\text{-}H}^k(s)$

$\mathsf{hkdf\text{-}expand\text{-}label}_1(s, l, h) =$

$\quad \mathsf{HMAC\text{-}H}^s(len_{\mathsf{H}()} \| \text{“TLS 1.3,”} \| l \| h \| \mathtt{0x01})$

$\mathsf{derive\text{-}secret}(s, l, m) = \mathsf{hkdf\text{-}expand\text{-}label}_1(s, l, \mathsf{H}(m))$

1-RTT Key Schedule:

$\mathsf{kdf}_0 = \mathsf{hkdf\text{-}extract}(0^{len_{\mathsf{H}()}}, 0^{len_{\mathsf{H}()}})$

$\mathsf{kdf}_{hs}(es, e) = \mathsf{hkdf\text{-}extract}(es, e)$

$\mathsf{kdf}_{ms}(hs, log_1) = ms, k_c^h, k_s^h, k_c^m, k_s^m$ where

$\quad ms = \mathsf{hkdf\text{-}extract}(hs, 0^{len_{\mathsf{H}()}})$

$\quad hts_c = \mathsf{derive\text{-}secret}(hs, \mathsf{hts}_c, log_1)$

$\quad hts_s = \mathsf{derive\text{-}secret}(hs, \mathsf{hts}_s, log_1)$

$\quad k_c^h = \mathsf{hkdf\text{-}expand\text{-}label}(hts_c, \mathsf{key}, \text{“”})$

$\quad k_c^m = \mathsf{hkdf\text{-}expand\text{-}label}(hts_c, \mathsf{finished}, \text{“”})$

$\quad k_s^h = \mathsf{hkdf\text{-}expand\text{-}label}(hts_s, \mathsf{key}, \text{“”})$

$\quad k_s^m = \mathsf{hkdf\text{-}expand\text{-}label}(hts_s, \mathsf{finished}, \text{“”})$

$\mathsf{kdf}_k(ms, log_4) = k_c, k_s, ems$ where

$\quad ats_c = \mathsf{derive\text{-}secret}(ms, \mathsf{ats}_c, log_4)$

$\quad ats_s = \mathsf{derive\text{-}secret}(ms, \mathsf{ats}_s, log_4)$

$\quad ems = \mathsf{derive\text{-}secret}(ms, \mathsf{ems}, log_4)$

$\quad k_c = \mathsf{hkdf\text{-}expand\text{-}label}(ats_c, \mathsf{key}, \text{“”})$

$\quad k_s = \mathsf{hkdf\text{-}expand\text{-}label}(ats_s, \mathsf{key}, \text{“”})$

$\mathsf{kdf}_{psk}(ms, log_7) = psk'$ where

$\quad psk' = \mathsf{derive\text{-}secret}(ms, \mathsf{rms}, log_7)$

PSK-based Key Schedule:

$\mathsf{kdf}_{es}(psk) = es, k^b$ where

$\quad es = \mathsf{hkdf\text{-}extract}(0^{len_{\mathsf{H}()}}, psk)$

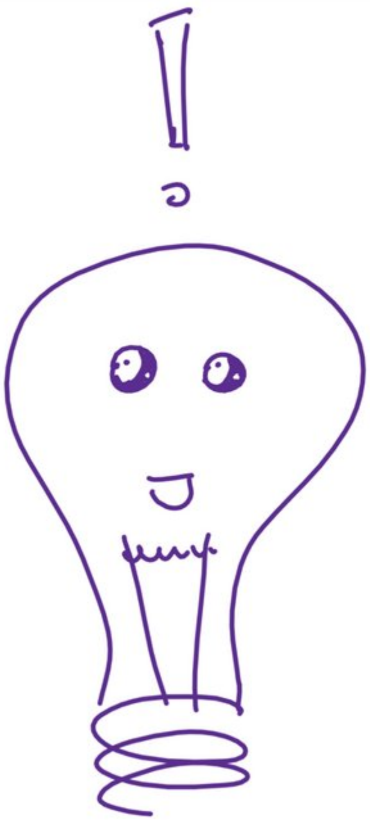
$\quad k^b = \mathsf{derive\text{-}secret}(es, \mathsf{pbk}, \text{“”})$

$\mathsf{kdf}_{0RTT}(es, log_1) = k_c$ where

$\quad ets_c = \mathsf{derive\text{-}secret}(es, \mathsf{ets}_c, log_1)$

$\quad k_c = \mathsf{hkdf\text{-}expand\text{-}label}(ets_c, \mathsf{key}, \text{“”})$

LAKE – Solving the Key Distribution Problem (Again)



The LAKE working group at IETF is in the process of establishing the next Lightweight Authenticated Key Establishment standard.

It should support:

- Secure key establishment for OSCORE
- A path for upgrade, from PSK, to raw public keys (RPK), to PKIs with certificates
- Identity protection
- Crypto agility
- Perfect forward secrecy
- Key compromise impersonation
- Mutual authentication

Lightweight

- I Compatible with 6TiSCH, LoRaWAN, NB-IoT
- I Metrics: bytes on the wire, round-trips, power, new code on top of OSCORE

Following a standard

Datatracker | Sign in | Document search

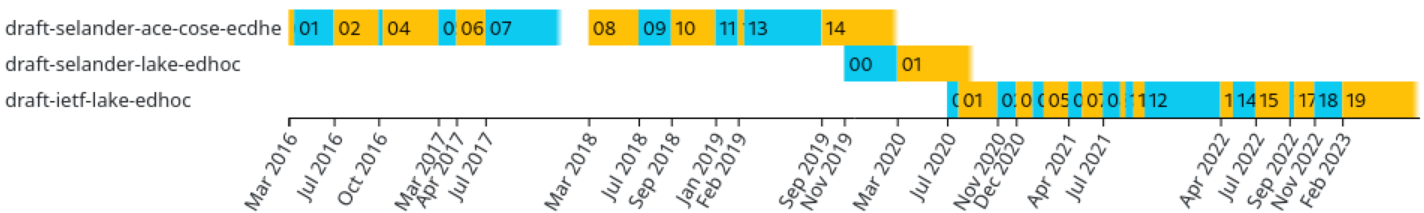
Ephemeral Diffie-Hellman Over COSE (EDHOC)

draft-ietf-lake-edhoc-19

Status | IESG evaluation record | IESG writeups | Email expansions | History

Versions:

00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19



Document	**Type**	Active Internet-Draft (lake WG)
	Authors	Göran Selander, John Preuß Mattsson, Francesca Palombini
	Last updated	2023-02-03
	Replaces	draft-selander-lake-edhoc
	RFC stream	Internet Engineering Task Force (IETF)
	Intended RFC status	Proposed Standard

Key takeaways

- IoT is changing the world we live in,
  we need to change our threat model too
- Constrained devices are challenging to
  secure:
  security comes at a cost
- Advocate open standards for IoT security
- Beyond secrecy:
  stronger security for deploying in an
  ”hostile” environment

Questions?

Specific take-aways

(exam)

- C-I-A view
- Conflicts and trade-offs
- LoRaWAN specifics
  - keys and where they reach
  - physical layer attacks

Additional remarks and observations

The mainstream idea of IoT Security is often looking the wrong way, or, assuming the vulnerabilities to be in places where they are not.

The most vulnerable attack surfaces are NOT

- **The network**
  (and neither can systems be protected there)
- **The devices/gadgets**

Observed vulnerabilities

Humans & the way they handle credentials
(the keys are strong enough, key provisioning is not)

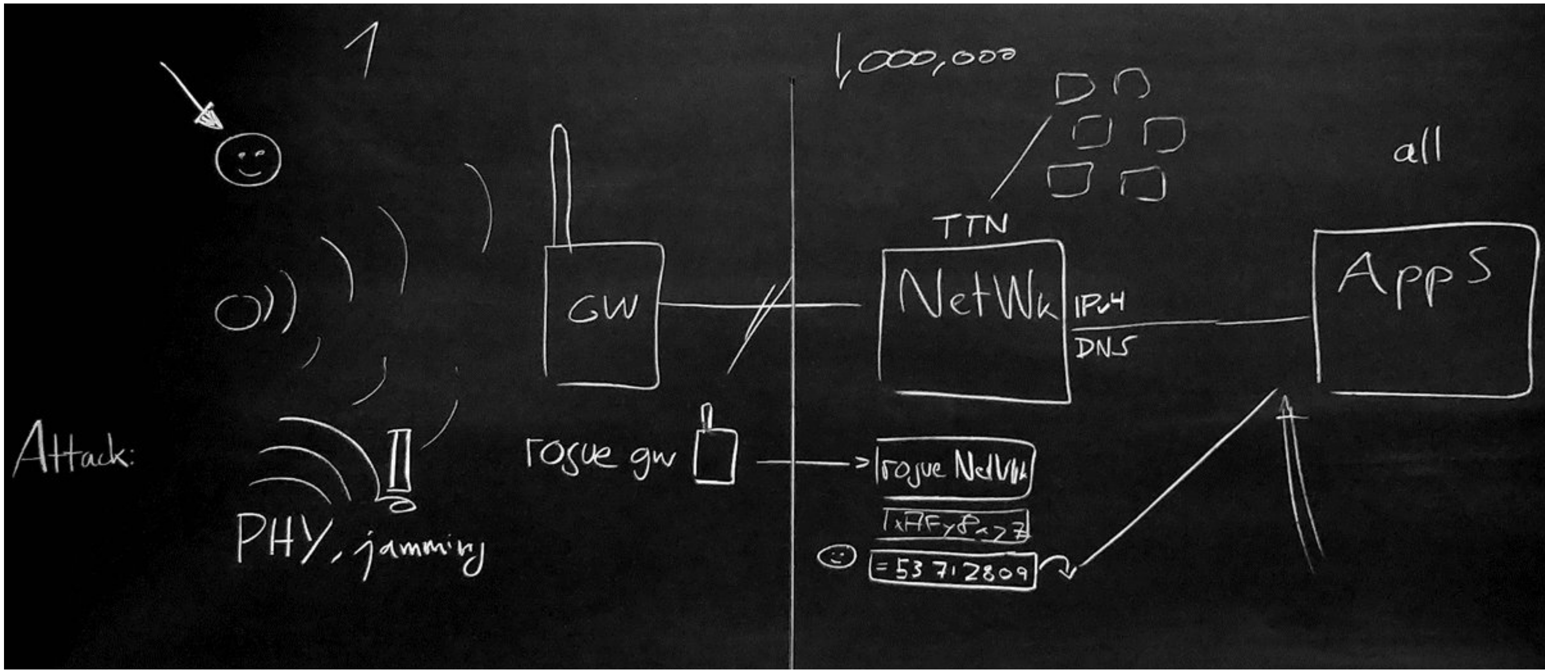
Social engineering
(”AI” and deep fakes as additional factor)

Inside attacks
(statistics say >50% of all attacks are from inside)
(update sources!)

Backend systems, platforms
(consider the numbers of targets exposed)

The client OS

Hypothetical Scenario



The OWASP Top-10 Bingo

Source: https://www.owasp.org/images/1/1c/OWASP-IoT-Top-10-2018-final.pdf

Exercise 1

The initial version of our MQTT broker allowed

a. anonymous publishing
b. http connections (not https)

We got contacted by DK-CERT and asked to change our configuration,
to have either one of these, but not both.
Please analyze this vulnerability by applying a Confidentiality-Integrity-Availability approach.
Which aspects are affected by these two measures, and how?
What are possible exploits?

Exercise 2.1

Threat analysis of your own sensor system (the CO_2 sensor)

Let us assume that our sensor was a little more important than it is now.
For example, if a facility management depended on the data,
and could be forced to take action, even evacuate rooms.
Draw a system diagram of your system (simplified) and identify attack surfaces -
how could your system be attacked?
what aspects (C-I-A) are affected?
how could you protect your system against each of these?

Exercise 2.2

Let us do this in teams.

An *evil team* -

Explain your motivation, goals and strategy

A *good team* -

How would you mitigate such attacks?
Preemptively, after attack, …

What are the LoRaWAN specifics here?