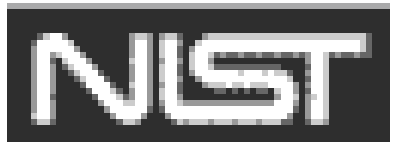
RPM: Lightweight Communication Security from Additive Stream Ciphers

Giovanni Di Crescenzo,
Applied Communication Sciences, NJ
gdicrescenzo@appcomsci.com

Glenn O. Veach,
Relevant Security Corp., CO, gveach@rscorp.com



Lightweight Cryptography Workshop 2015

Lightweight Cryptography: motivations and introduction

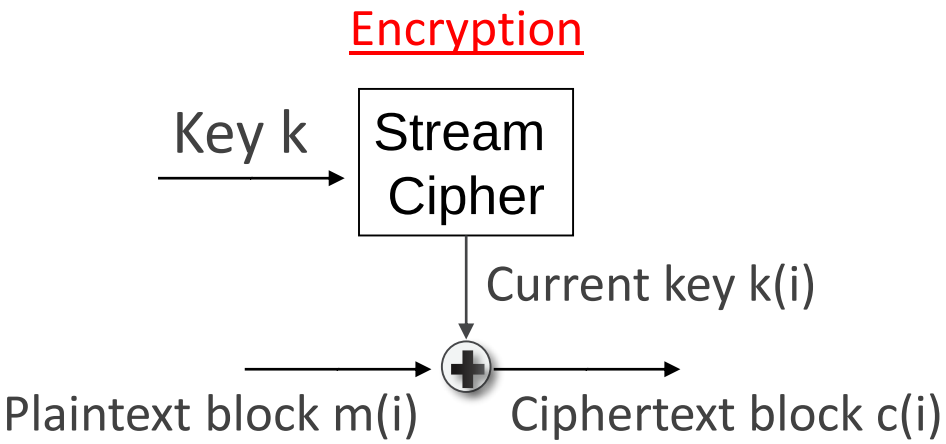
- Motivations and Introduction:
  - In today's interconnected world and the “Internet of Things”, certain sufficiently constrained environments exist such that the traditional NIST-approved cryptographic algorithms do not perform to the operating requirements
  - lightweight cryptographic primitives, although possibly not as secure as conventional cryptographic primitives (e.g., block ciphers, hash functions), may be efficient enough to enable applications over such constrained environment
  - given the current state of the art in lightweight cryptography, giving evidence that any level (even though smaller) of security is achievable, while guaranteeing the desired higher performance, is already a non-trivial challenge
- This paper:
  - analysis of lightweight cryptographic solutions used in RPM (a product from Relevant Security Corporation) to achieve communication security among interconnected devices working in Low Power Wide Area (LPWA) networks
  - these techniques were designed to take into consideration various constraining factors including lightweight computing of sensors, power requirements, data rates and bandwidth in conjunction with Medium Quality of Service (QoS) requirements such as in residential smart meters or HVAC, as well as High QoS requirements such as in heart monitors or power utility smart grid

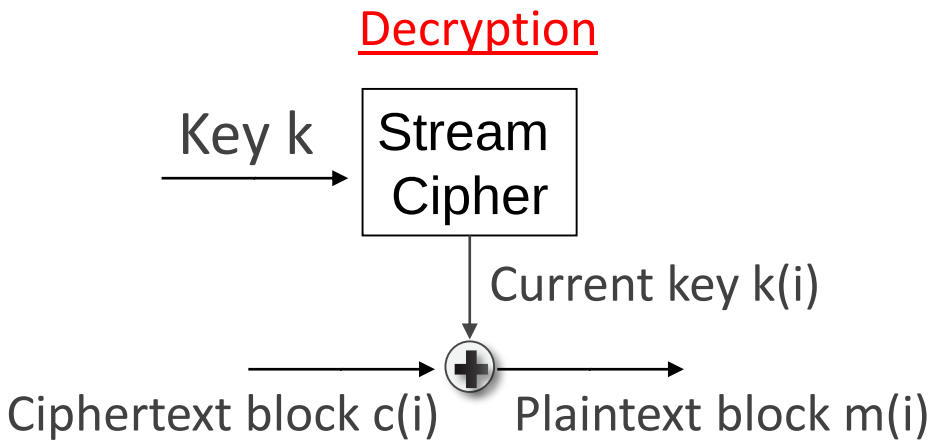
RPM lightweight cryptography solutions: our contributions

- Preliminary results include quantitative aspects of (known) relationships between stream ciphers and secure symmetric encryption schemes:
  - stream ciphers generating a pseudo-random sequence of values can be turned into secure symmetric encryption schemes; and
  - the period of stream ciphers directly bounds the time window of security for the associated symmetric encryption scheme.
- In our main results, we isolate the underlying RPM lightweight cryptographic primitives as two additive and non-linear stream ciphers, for which we show:
  - Symmetric encryption based on the 2 lightweight RPM stream ciphers is ~ 1 order of magnitude faster than AES-based symmetric encryption,
  - On a more theoretical side, we show that idealized versions of the 2 lightweight RPM stream ciphers have a long period; and
  - In combination with conventional cryptographic modes of operation, the 2 lightweight RPM stream ciphers very efficiently provide communication security properties, including: confidentiality via encryption, implicit mutual and constant authentication, continuous key management by key re-freshing.

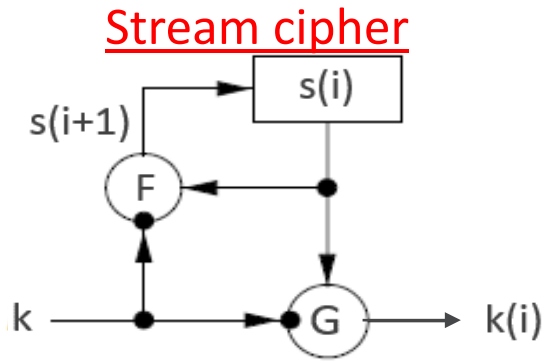
Stream ciphers and symmetric encryption

- Symmetric encryption based on additive stream ciphers:





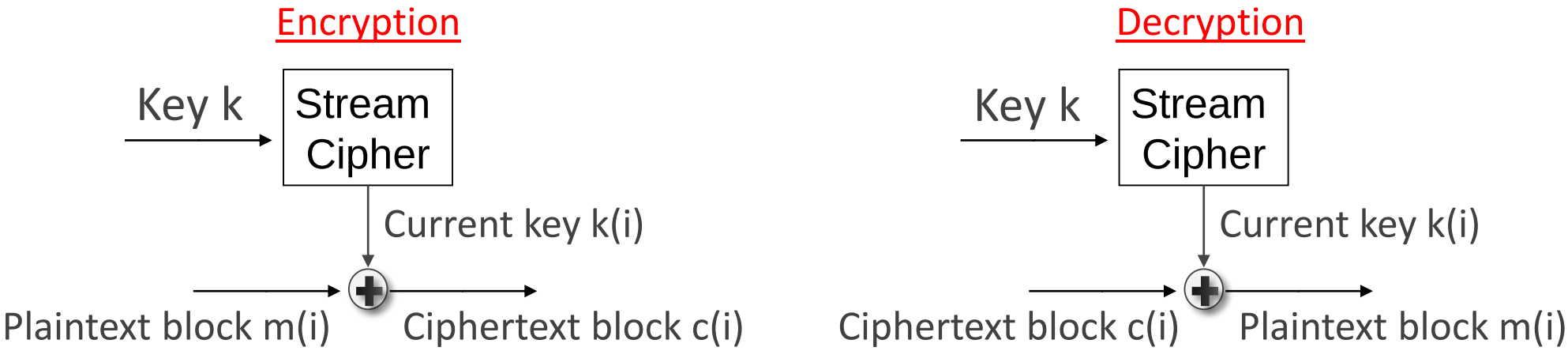
- A (synchronous, additive) stream cipher SC includes a next state function F and a current key function G:



- A (synchronous, additive) stream cipher SC has period t if s(i+t)=s(i) for some positive integer j and all i>=j

Symmetric encryption based on stream ciphers: (quantitative aspects of) known properties

- Symmetric encryption based on additive stream cipher SC:



- Proposition 1 (a security property): If SC returns a (t,ε)-pseudo-random sequence of q keys, then the symmetric encryption scheme based on SC satisfies decryption correctness and (t',q',ε')-real-or-random security, for t'=t-O(q) and ε'=ε.
  - E.g.: AES in Counter or OFB mode returns a pseudo-random sequence of keys, assuming AES is a pseudo-random function
- Proposition 2 (an insecurity property): If SC has period p, then the symmetric encryption scheme based on SC satisfies decryption correctness but does not satisfy (t,q,ε)-real-or-random security, for t=O(p), q>=p and any ε>0.

The first RPM stream cipher: primitives and construction

- Primitives: Position Digit Algebra Function (PDAF) and One-Way Cut (OWC)

Parameters for PDAF: positive numbers n, r, where n is even.

Input to PDAF: lists $x = (x[0], \ldots, x[n-1])$ and $y = (y[0], \ldots, y[n-1])$, where $x[i], y[i] \in \{0, \ldots, r-1\}$, for $i = 0, \ldots, n-1$.

Instructions for PDAF:

1. For $i = 0, \ldots, n-1$,
   set $j(i) = (i + y(i)) \bmod n$
   set $z[i] = (x[i] + x[j(i)]) \bmod r$
2. Return: $z = (z[0], \ldots, z[n-1])$.

Parameters for OWC: positive numbers n, r, where n is even.

Input to OWC: list $x = (x[0], \ldots, x[n-1])$, where $x[i] \in \{0, \ldots, r-1\}$, for $i = 0, \ldots, n-1$.

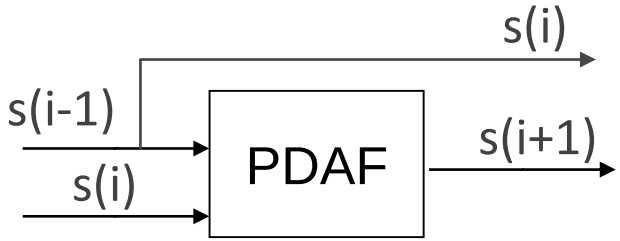
Instructions for OWC:

1. For $i = 0, \ldots, n/2-1$,
   set $z[i] = (x[2i] + x[2i+1]) \bmod r$
2. Return: $z = (z[0], \ldots, z[n/2-1])$.

- The (synchronous, additive) stream cipher rpmSC1 has the following next state function F and current key function G:

Function F

(s(0) random, s(-1)=k)



Function G

The first RPM stream cipher: primitives and an idealized construction

- Primitives: Position Digit Algebra Function (PDAF) and One-Way Cut (OWC)

Parameters for PDAF: positive numbers n, r, where n is even.

Input to PDAF: lists $x = (x[0], \ldots, x[n-1])$ and $y = (y[0], \ldots, y[n-1])$, where $x[i], y[i] \in \{0, \ldots, r-1\}$, for $i = 0, \ldots, n-1$.

Instructions for PDAF:

1. For $i = 0, \ldots, n-1$,
   set $j(i) = (i + y(i)) \bmod n$
   set $z[i] = (x[i] + x[j(i)]) \bmod r$
2. Return: $z = (z[0], \ldots, z[n-1])$.

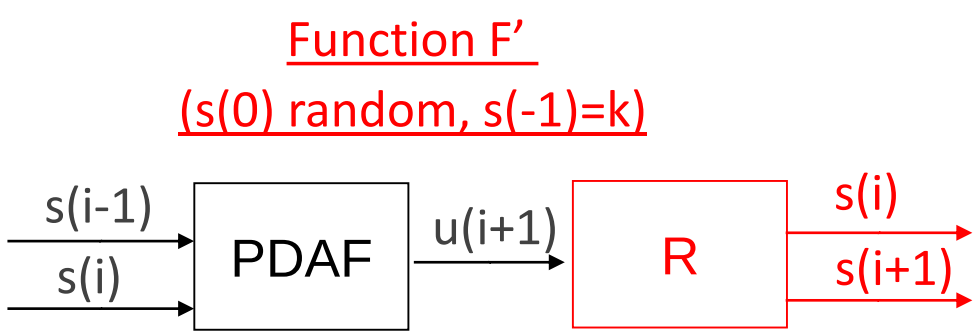
Parameters for OWC: positive numbers n, r, where n is even.

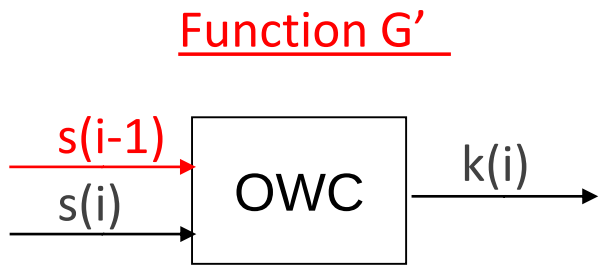
Input to OWC: list $x = (x[0], \ldots, x[n-1])$, where $x[i] \in \{0, \ldots, r-1\}$, for $i = 0, \ldots, n-1$.

Instructions for OWC:

1. For $i = 0, \ldots, n/2-1$,
   set $z[i] = (x[2i] + x[2i+1]) \bmod r$
2. Return: $z = (z[0], \ldots, z[n/2-1])$.

- The idealized version rpmSC1' of the (synchronous, additive) stream cipher rpmSC1 has next state function F' (using random function R) and current key function G':





Analysis of (the idealized version of) rpmSC1: result and period properties

Theorem 3: The expected value of the period of rpmSC1' is >= $r^{n/2-0.85 \log_2 n-1}$

Proof ideas:

- By a geometric distribution argument, the period of rpmSC1' is computed from the probability that the sequence of keys returned by G' repeats
- z=PDAF(x,y) can be written as $z=M^{(y)}(x)$, where $M^{(y)}$ is a matrix with 1 or 2 nonzero elements on each row, depending on y
- The sequence of keys in rpmSC1' repeats when z repeats, which happens when a new x is in the preimage set of z, under $M^{(y)}$
- By linear independence, the average size of the preimage set of z, under $M^{(y)}$, is directly linked to the average rank of $M^{(y)}$
- By careful analysis of the structure of $M^{(y)}$ the average rank of $M^{(y)}$ depends on the average number of disjoint "cycle structures" in $M^{(y)}$
- We compute the average number of disjoint cycle structures in $M^{(y)}$ and derive a lower bound on the average period of rpmSC1'

The second RPM stream cipher: primitives and construction

- Primitives: Combine (CMBN) and Extract (EXT)

Parameters for CMBN: positive numbers n, r, where n is even.

Input to CMBN: lists $x = (x[0], \ldots, x[n-1])$ and $y = (y[0], \ldots, y[n-1])$, where $x[i], y[i] \in \{0, \ldots, r-1\}$, for $i = 0, \ldots, n-1$.

Instructions for CMBN:

1. Set $i_{-1} = j_{-1} = -1$
2. For $h = 0, \ldots, n-1$,
   set $i_h = (i_{h-1} + 1 + x[h]) \bmod n$
   set $j_h = (j_{h-1} + 1 + y[h]) \bmod n$
   set $z[h] = (x[j_h] + y[i_h]) \bmod r$
3. Return: $z = (z[0], \ldots, z[n-1])$.

Parameters for EXTC: positive numbers n, r, where n is even.

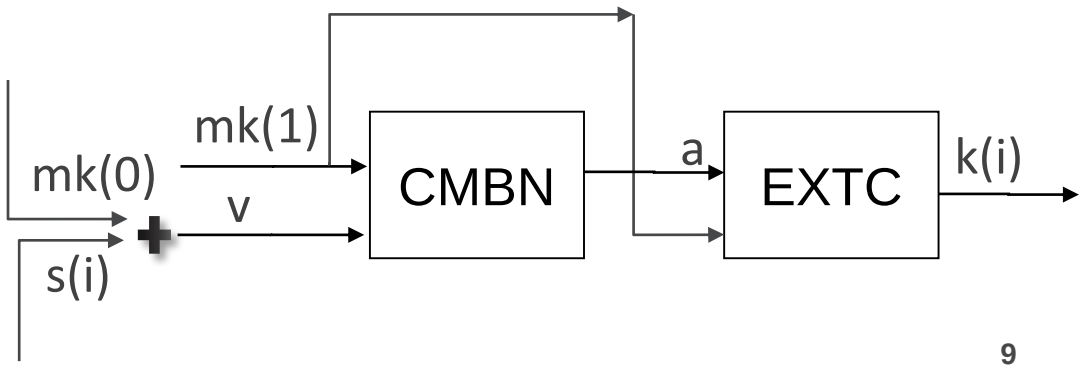
Input to EXTC: lists $x = (x[0], \ldots, x[n-1])$ and $y = (y[0], \ldots, y[n-1])$, where $x[i], y[i] \in \{0, \ldots, r-1\}$, for $i = 0, \ldots, n-1$.

Instructions for EXTC:

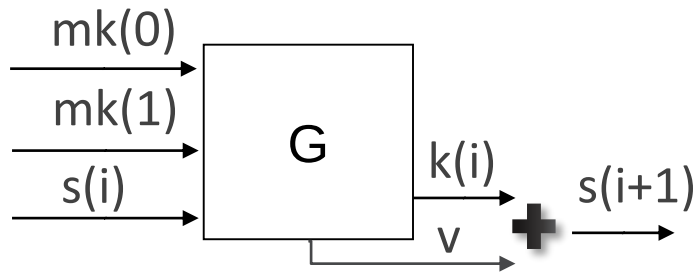
1. Set $i_{-1} = -1$
2. For $h = 0, \ldots, n-1$,
   set $i_h = (i_{h-1} + 1) + y[h] \bmod n$
   set $z[h] = x[i_h] \bmod r$ (variant : $z[h] = x[i_h] + y[i_h] \bmod r$)
3. Return: $z = (z[0], \ldots, z[n-1])$.

- The (synchronous, additive) stream cipher rpmSC2 has the following next state function F and current key function G:

Function G (with mk(0),mk(1) as master keys)



Function F



The second RPM stream cipher: primitives and an idealized construction

- Primitives: Combine (CMBN) and Extract (EXT)

Parameters for CMBN: positive numbers n, r, where n is even.

Input to CMBN: lists $x = (x[0], \ldots, x[n-1])$ and $y = (y[0], \ldots, y[n-1])$, where $x[i], y[i] \in \{0, \ldots, r-1\}$, for $i = 0, \ldots, n-1$.

Instructions for CMBN:

1. Set $i_{-1} = j_{-1} = -1$
2. For $h = 0, \ldots, n-1$,
   set $i_h = (i_{h-1} + 1 + x[h]) \bmod n$
   set $j_h = (j_{h-1} + 1 + y[h]) \bmod n$
   set $z[h] = (x[j_h] + y[i_h]) \bmod r$
3. Return: $z = (z[0], \ldots, z[n-1])$.

Parameters for EXTC: positive numbers n, r, where n is even.

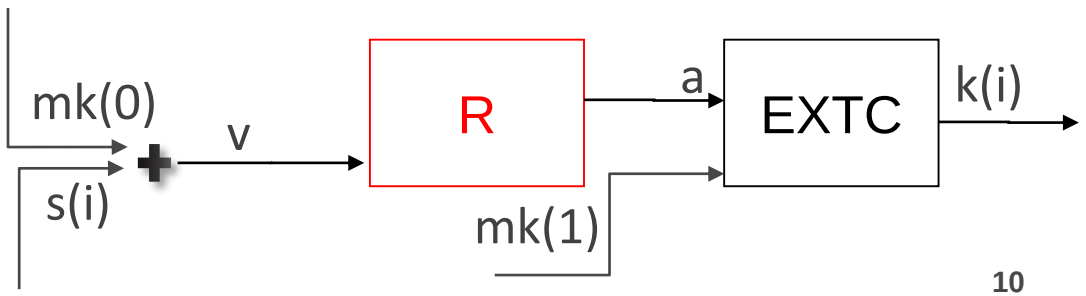
Input to EXTC: lists $x = (x[0], \ldots, x[n-1])$ and $y = (y[0], \ldots, y[n-1])$, where $x[i], y[i] \in \{0, \ldots, r-1\}$, for $i = 0, \ldots, n-1$.

Instructions for EXTC:

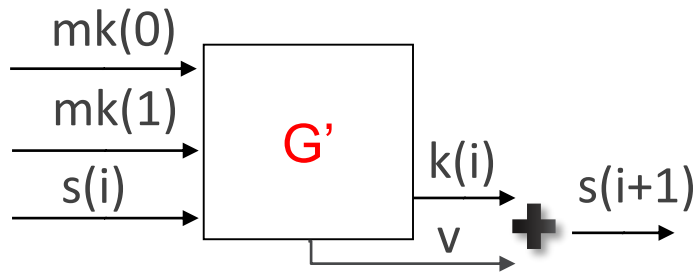
1. Set $i_{-1} = -1$
2. For $h = 0, \ldots, n-1$,
   set $i_h = (i_{h-1} + 1) + y[h] \bmod n$
   set $z[h] = x[i_h] \bmod r$ (variant : $z[h] = x[i_h] + y[i_h] \bmod r$)
3. Return: $z = (z[0], \ldots, z[n-1])$.

- The (synchronous, additive) stream cipher rpmSC2' has next state function F' and current key function G', where R is a random function:

Function G' (with mk(0),mk(1) as master keys)



Function F'



Analysis of (the idealized version of) rpmSC2: result and period properties

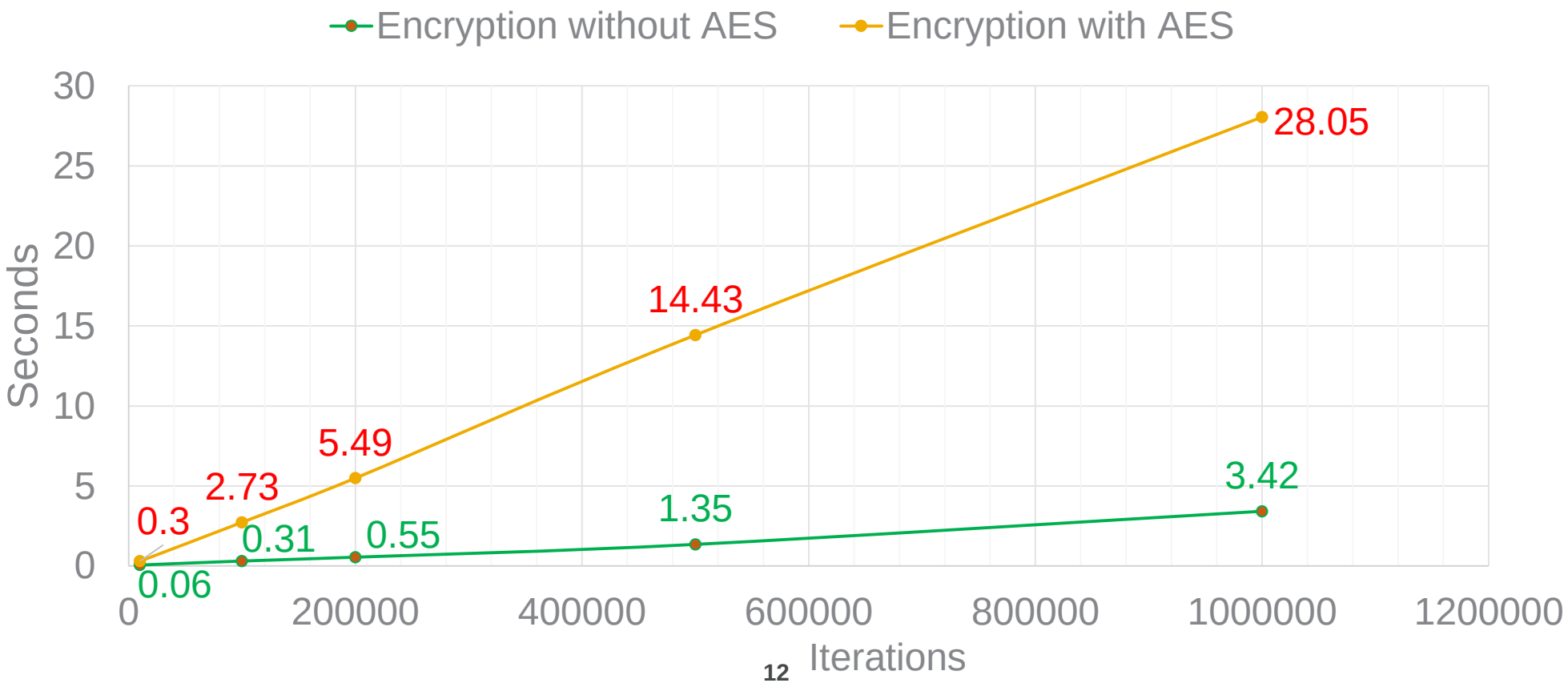
Theorem 4: The expected value of the period of rpmSC2' is >= $r^{0.316\,n}$

Proof ideas:

- By a geometric distribution argument, the period of rpmSC2' is computed from the probability that the sequence of keys returned by G' repeats
- The output of G' can be written as z=EXTC(x,k), for some master key k, and some variable random value x
- The sequence of keys returned by G' repeats when z repeats, which happens when a new x is in the preimage set of z, under EXTC(.,k)
- We observe that z[h]=x[i(h)], for h=0,…,n-1, where the vector {i(h)} can be written as = c + T k, where c is a constant vector, T is a triangular matrix of full rank, and k is the vector of random master keys
- By linear independence, the vector {i(h)} is uniformly distributed and may miss some of the position indices, thus resulting in an increase of the preimage set
- Then the log of the average size of the preimage set of z can be computed as the average number of empty bins obtained when throwing n balls into n bins
- The latter is computed as n/e, e=2.87, from which the bound $r^{(n/2-n/2e)}$ directly follows

Performance analysis

- Measurements done in 2010, with machine specifications: Intel Core i3-2330M 2.2GHz, 4GB RAM
- Tests: Measurements were performed for symmetric encryption based on the additive stream cipher rpmSC2, in two cases: (1) encryption is performed using modular sum between the stream cipher's next key and the message block; and (2) encryption is performed using the AES block cipher, whose input key was the stream cipher's next key
- Results: Performance of encryption without AES is almost one order or magnitude faster. Empirically observed that rpmSC1 has similar performance as rpmSC2, even though a bit slower.

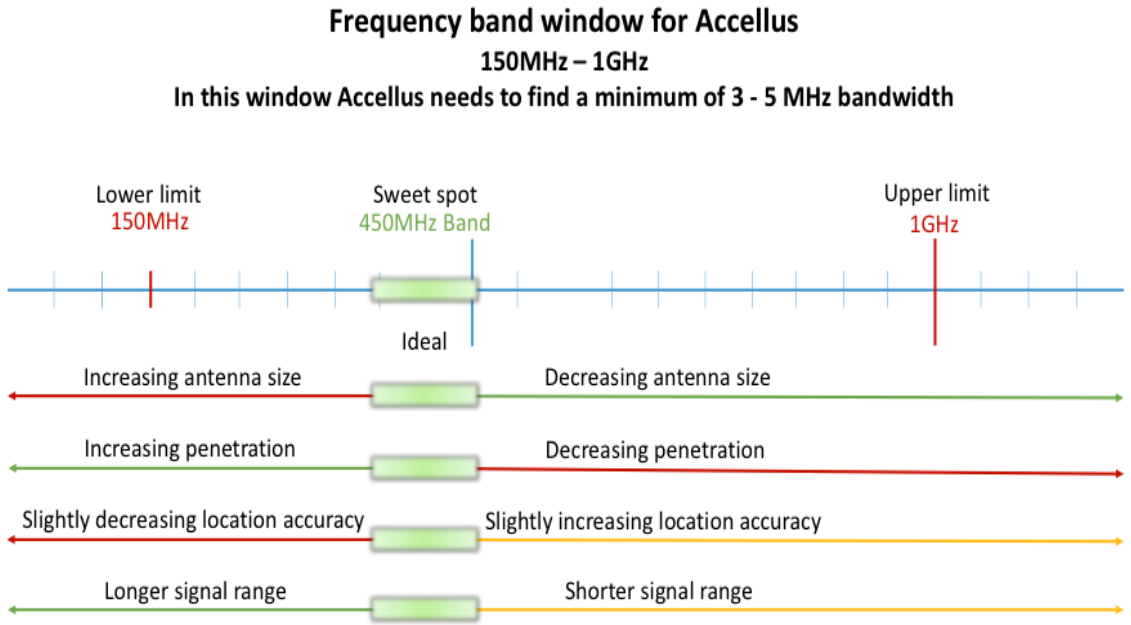


Application use cases (1)

- An example application use case for the RPM stream ciphers and secure communication solutions is for securing Low Power Wide Area (LPWA) Networks with the following features:
  - utilize lightweight computing devices,
  - have minimal power requirements,
  - operate at low data rates; e.g., 1Kbps,
  - use 150 MHz to 1 GHz transmission power with unlimited duty cycles,
  - can deliver Medium, High quality of service,
  - have mission critical performance capability, and
  - can provide packet level security.

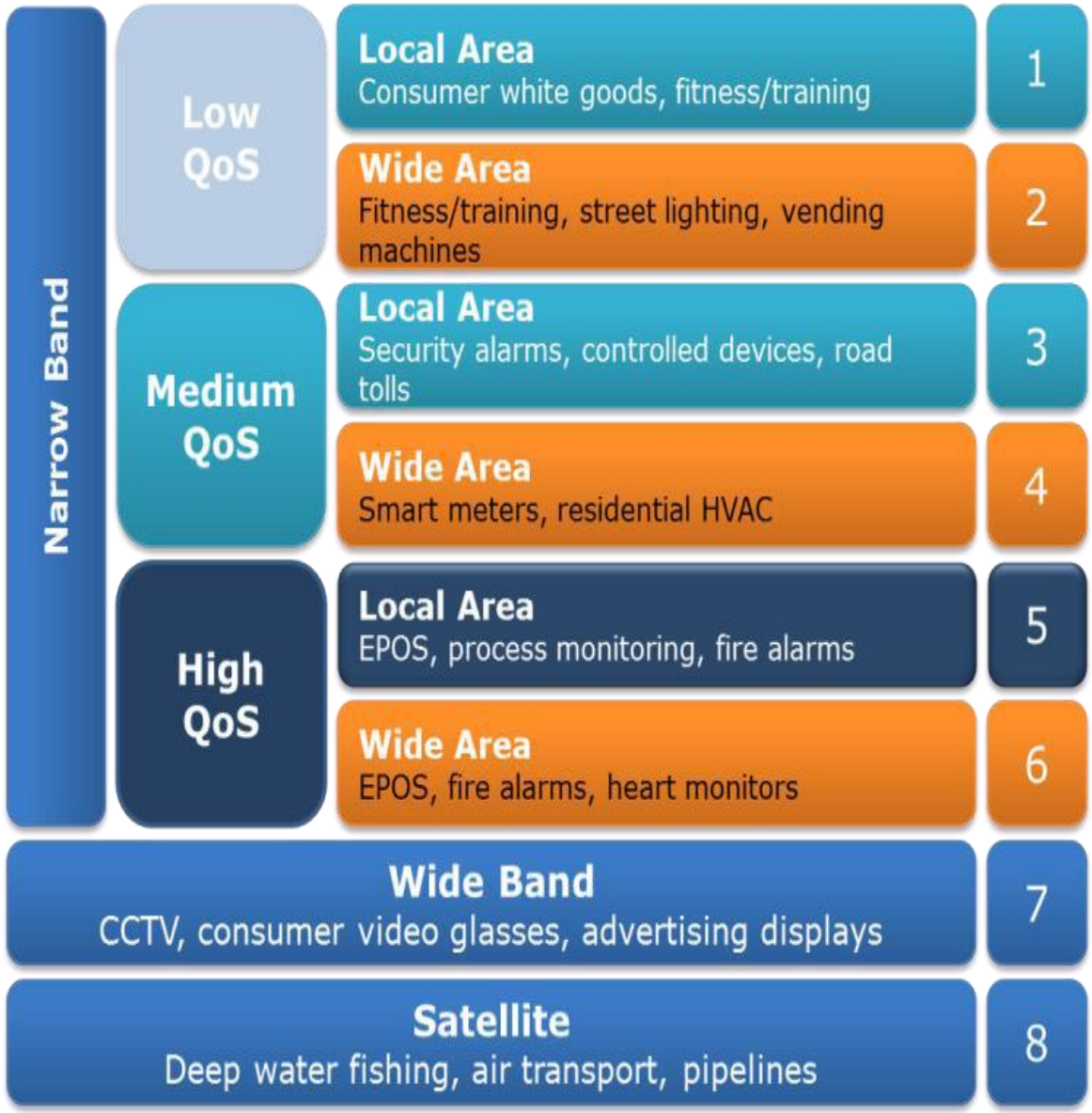
- The figure shows the optimal bandwidth factors of Accellus LPWA [Spl14]

ACCELLUS
COMMUNICATION NETWORKS



Application use cases (2)

- To identify the market for LPWA networks, AEgis Systems Limited and Machina Research [Aeg14,Mor13] have defined a set of 8 application categories, based on range, bandwidth and quality of service (QoS) but reflecting the predominantly narrow band nature of most Market-to-Market (M2M) applications.
- A sizeable proportion of the narrow band connections fall into the medium QoS category. This is largely a reflection of the M2M market itself, which is mainly accounted for by applications in sectors such as automotive, manufacturing, smart metering and building automation, which (whilst not being mission critical in life or death terms) nevertheless may have significant financial or public policy implications, if they should not perform in the required way.
- By design, the Accellus LPWA Network technology fits in groups 2, 4 and 6, (shown in orange) with the emphasis on groups 4 and 6.



Conclusions

- For cryptography primitives in conventional desktop and/or server environments, several years of modern cryptography research were needed to find the right formal definitions so to rigorously state their security properties under carefully formulated models and assumptions.
- For lightweight cryptography, such a process has barely started.
- In this paper, we proposed an approach to rule out certain natural attacks to lightweight cryptography primitives based on stream ciphers.
- In particular, we showed that (idealized versions of) the RPM stream ciphers have long period and thus lead to efficient and secure communication.
- Results need to be interpreted with caution, just like with conventional results in the “random oracle model”.