Securing random key predistribution against key capture

Mirosław Kutyłowski

Wrocław University of Technology

Shanghai, 20.10.2010, joint work with Prof. Jacek Cichoń, Jarosław Grząślewicz, Zbigniew Gołebiewski

Network Model

Ad hoc network
of tiny artefacts

Devices

- weak computationally
- no asymmetric cryptography

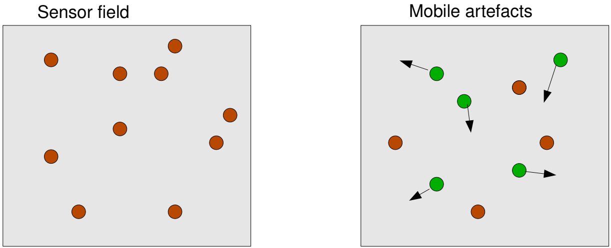
Communication

- wireless links
- no advance knowledge of network architecture
- mobility of nodes
- nodes join and leave the network
- unpredictable who will talk with whom and when

Ad hoc network
of tiny artefacts

Scenarios

- sensors fields
- mobile artefacts

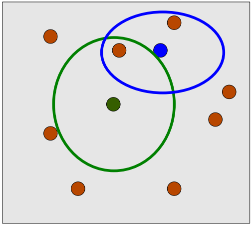


Ad hoc network
of tiny artefacts

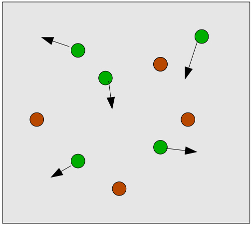
Scenarios

- sensors fields
- mobile artefacts

Sensor field



Mobile artefacts

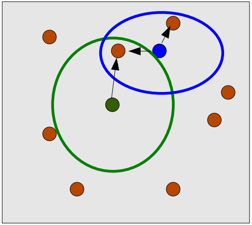


Ad hoc network
of tiny artefacts

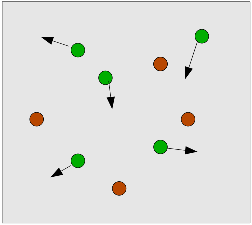
Scenarios

- sensors fields
- mobile artefacts

Sensor field



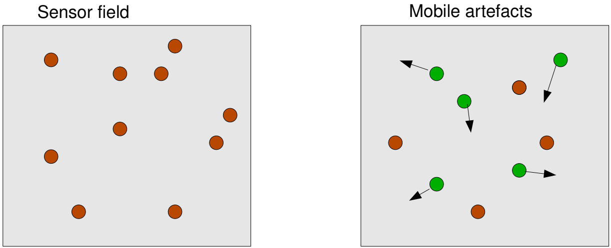
Mobile artefacts



Ad hoc network
of tiny artefacts

Scenarios

- sensors fields
- mobile artefacts

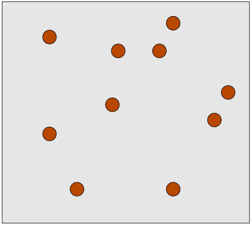


Ad hoc network
of tiny artefacts

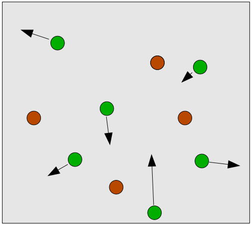
Scenarios

- sensors fields
- mobile artefacts

Sensor field



Mobile artefacts



Data protection
for tiny artefacts

Data

- sensitive information (e.g. personal data)
- safety critical data (e.g. monitoring industry)
- ...

Data protection

for tiny artefacts

Data

- sensitive information (e.g. personal data)
- safety critical data (e.g. monitoring industry)
- ...

Adversary

- capturing data
- impersonation
- intercepting nodes

Data protection
for tiny artefacts

Data

- sensitive information (e.g. personal data)
- safety critical data (e.g. monitoring industry)
- ...

Adversary

- capturing data
- impersonation
- intercepting nodes

Possibilities:

- eavesdropping communication
- reverse engineering some devices
- cloning devices

Security requirements

Requirements

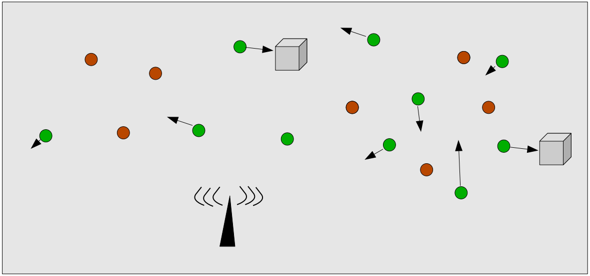
- **communication encrypted**
  (confidentiality)
- **data integrity**
  (data not manipulated when transmitted)
- **authentication of nodes**
  (impersonation impossible)

Ad hoc networks in a real world

Pure ad hoc networks versus reality

- research papers are focused on pure ad hoc networks - no infrastructure of any kind
- ... but even in emergency situations (typhoon, hurricane, earth quake,...) some kind of general infrastructure survives

A heterogenous network



Ad hoc network
of tiny artefacts

Goal

- design an ad hoc network keeping in mind that some service can be available from the network provider
- key replacement should be one of the main design goals
  – a time race with an adversary that tries to gather key material from captured devices

Random key predistribution

Initialization

- The system provider keeps a secret pool $\mathcal{K}$ of keys selected at random.
- Before being used a device receives k keys from $\mathcal{K}$ chosen at random.

Random Key Predistribution
simple devices, symmetric methods

Initialization

- The system provider keeps a secret pool $\mathcal{K}$ of keys selected at random.
- Before being used a device receives k keys from $\mathcal{K}$ chosen at random.

Setting up a connection between A and B

- A and B determine the keys they share, say $k_{i_1}, \ldots, k_{i_t}$,
- A and B compute the session key

$$\mathcal{K} = F(k_{i_1}, \ldots, k_{i_t}, A, B, \ldots)$$

based on the birthday paradox

- Probability that two subsets of size k of the pool of size n are disjoint equals

$$\left(1-\frac{k}{n}\right)\left(1-\frac{k}{n-1}\right)\cdots\left(1-\frac{k}{n-k+1}\right)\leq\left(1-\frac{k}{n}\right)^{k}$$

- For $k=\sqrt{n}$:

$$\left(1-\frac{k}{n}\right)^{k}=\left(1-\frac{1}{\sqrt{n}}\right)^{\sqrt{n}}\approx\frac{1}{e}$$

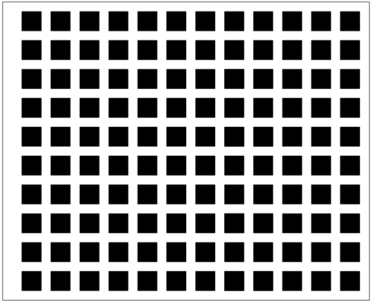
- For $k=2\sqrt{n}$:

$$\left(1-\frac{k}{n}\right)^{k}\approx\frac{1}{e^{4}}$$

Key predistribution

Pool of keys

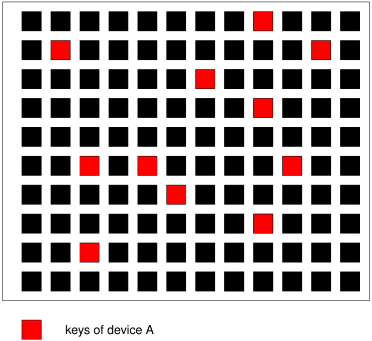
- the system provider generates a large pool of n keys
- each device receives a subset of keys of cardinality k



Key predistribution

Pool of keys

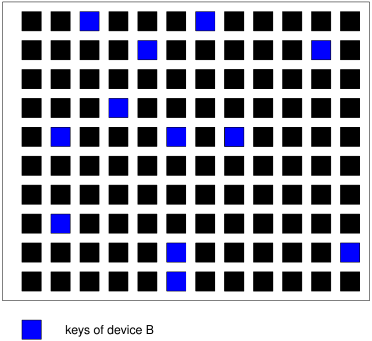
- the system provider generates a large pool of n keys
- each device receives a subset of keys of cardinality k



Key predistribution

Pool of keys

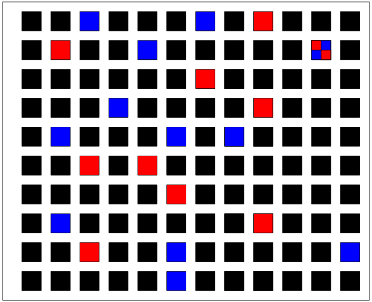
- the system provider generates a large pool of n keys
- each device receives a subset of keys of cardinality k



Key predistribution

Pool of keys

- the system provider generates a large pool of n keys
- each device receives a subset of keys of cardinality k



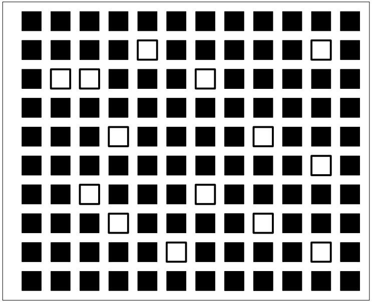
shared keys of devices A and B

Key predistribution

problems

Capturing keys

- an adversary can reverse engineer some devices
-

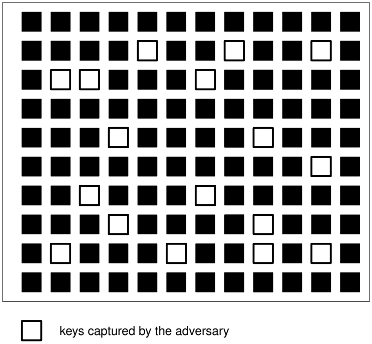


keys captured by the adversary

Key predistribution
problems

Capturing keys

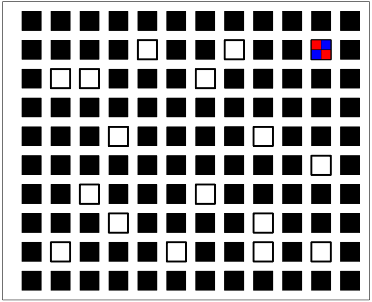
- an adversary can reverse engineer some devices
-



Key predistribution
problems

Capturing keys

- an adversary can reverse engineer some devices
- no more protection with the captured keys



keys captured by the adversary

Key predistribution
countermeasures

q-composite scheme

at least q shared keys are necessary for establishing a secure link,

- each device has to hold more keys
- attack effectiveness:
  - much harder for the adversary to have all q keys at once
  - much more keys are captured from each single device
- for a small number of captured nodes - improvement, for a larger number - vice versa

Multipath

devices A and B establish a session key from keys transported over the links:

$$A - C_1 - B,$$
$$A - C_2 - B,$$
$$\ldots,$$
$$A - C_q - B$$

- high density of devices necessary

Key Levels

Key Levels Technique

T Levels Scheme

1. each single key k from the basic method corresponds to an set of keys

$$K_1, K_2, \ldots, K_T$$

2. the keys related in a one-way fashion:

$$K_1 = K \quad \text{and} \quad K_{i+1} = G(K_i) \quad \text{for } i = 1, \ldots, T-1$$

where G is easy to compute but infeasible to invert

Establishing a Connection

T level scheme

Mechanism

if A holds K_i and B holds K_j, then $K_{\max(i,j)}$ used for establishing the shared key

computing K_s from K_t, for $s > t$, is easy,

it is infeasible for $s < t$

Gain

if an adversary holds

$$K_t \quad \text{for some } t > \max(i,j),$$

then the connection between A and B is secure against him

Problems

How to assign the levels

1. the uniform distribution is not optimal
2. example: the optimal pbb of choosing K_1, K_2, K_3, K_4:

   0.437055, 0.218527, 0.182106, 0.162312

Example: 2 levels

if level 1 is assigned with probability p, then pbb that Alice and Bob talk and Mallet cannot eavesdrop equals

$$f(p) = p^2(1-p)$$

Since the derivative $f'(p) = 2p - 3p^2$ is equal to 0 for $p = \frac{2}{3}$, and $f''(p) = 2 - 6p$ is negative for $\frac{2}{3}$, f reaches the maximum $\frac{4}{27}$ for $p = \frac{2}{3}$.

Problems

procedure for computing optimal probabilities

from k to $k+1$:

- choose $p_1, \ldots, p_L$ such that the expression

$$\sum_{i=2}^{L}(p_1 + \ldots + p_{i-1})^2 \cdot p_i$$

  is maximized

- Let q denote the probability of choosing the first L levels. The probability of adversary's failure equals

$$P(q,p) = q^2 \cdot (1-q) + q^3 \cdot p$$

  where p is the probability of adversary's failure conditioned on the event that the level of the shared key is within the first L levels for all.

- The optimal p known by induction.

How many levels?

lower bound

Theorem

For any L and any probability distribution $\mathcal{P}$, probability that Mallet can eavesdrop Bob and Alice (denoted $S_{1,L,\mathcal{P}}$) is $\leq \frac{1}{3}$.

Let A, B, M be independent random variables denoting the level of Alice, Bob and Mallet. according to pbb distribution $\mathcal{P} = [p_1, \ldots p_k]$. Then

$$\Pr[M > \max\{A,B\}] = \sum_{i=1}^{L} \Pr[M > \max\{A,B\} | M = i] \cdot \Pr[M = i] =$$

$$\sum_{i=1}^{L} \Pr[i > \max\{A,B\}] \cdot p_i = \sum_{i=2}^{L} (p_1 + \ldots + p_{i-1})^2 \cdot p_i \quad .$$

Let $q_0 = 0$ and $q_i = p_1 + \ldots + p_i$ for $i = 1, \ldots, L$. Let us split interval $[0,1]$ into subintervals $I_i = [q_{i-1}, q_i)$. Then

$$\frac{1}{3} = \int_0^1 x^2 dx \geq \sum_{i=1}^{L} \inf_{x \in I_i} (x^2) \cdot |I_i| = \sum_{i=2}^{L} (p_1 + \ldots + p_{i-1})^2 \cdot p_i = S_{1,L,\mathcal{P}} \; .$$

Attack Cost

the expected number of devices corrupted until a connection becomes insecure

Theorem (2 level case, p is the probability to choose level 1)

Let $L_{m,p}$ denote the number of steps after which adversary collects all keys for compromising connection based on m shared keys. Then

$$E[L_{m,p}] = \int_0^\infty \left(1 - \frac{H(t)}{e^t}\right) dt \, , \tag{1}$$

where $H(z) = \left(e^{z/m} - 1 - p^2(e^{qz/m} - 1)\right)^m$ and $q = 1 - p$.

Attack Cost

the expected number of devices corrupted until a connection becomes insecure

Corollary

- For $m = 1$ the optimal value of p is 0.5; then $E[L_m] \approx 1.25$.
- If $m = 10$, then the optimal value of p is 0.32164; in this case we get $E[L_m] = 40.9724$, so $E[L_m] = 1.39887 \cdot m \cdot H_m$, where $H_m =$ the mth harmonic number. So the actual cost of breaking the transmission is increased by $\approx 40\%$

Very large number of levels

From factor 1 improve to 1.5 as a limit value.

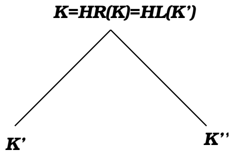
Trees

an extension with no *weak keys*

Idea

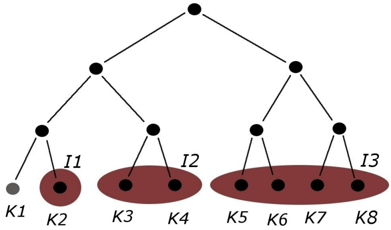
Instead of a single key K or a chain of keys $K_0, K_1 \ldots$, we can construct the following tree $T_{\hat{K}}$ of keys:

- each node of the tree is labeled with a key, the root is labeled with $\hat{K}$,
- if a node is labeled with key K, then its parent is labeled with $H_i(K)$, where $i = L, R$



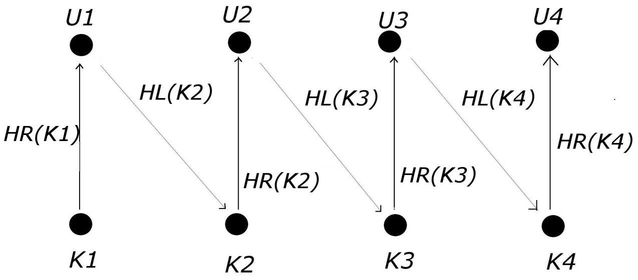
Trees

an extension with no *weak keys*



a tree containing keys $K_1, \ldots K_8$, if adversary is holding the key K_1, then the communication between A and B is not broken if they both hold keys from $I1 = \{K_2\}$ or from $I2 = \{K_3, K_4\}$ or from $I3 = \{K_5, K_6, K_7, K_8\}$

Reducing the number of keys in a device
keeping connectivity



1. special choice of keys in the pool
2. the devices do not have to share a key, subsequent keys can be used as well

Refreshing

with key levels

Infinitely many levels

- The system provider has a one-way function with a trapdoor.
- For each key from the pool there are infinitely many levels.
- The provider uses the trapdoor to compute keys of lower indexes.

Evolving keys

- from time to time each device visits a kiosk run by the system provider
- during the visit an independent verification and ... getting the key level of the current epoch
- the system loads the new keys of the epoch to each kiosk.

Random Key Redistribution

Key redistribution scheme

General framework

- **predistribution keys** used only for encryption of temporal keys
- **temporal keys** used for communication between devices
- new temporal keys **broadcasted** periodically, every key from the pool used to encrypt one temporal key

Key redistribution scheme

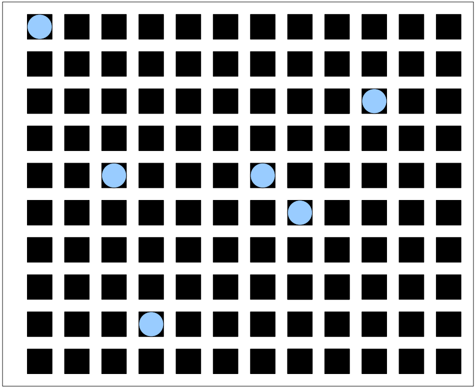
General framework

- **predistribution keys** used only for encryption of temporal keys
- **temporal keys** used for communication between devices
- new temporal keys **broadcasted** periodically, every key from the pool used to encrypt one temporal key

Main trick

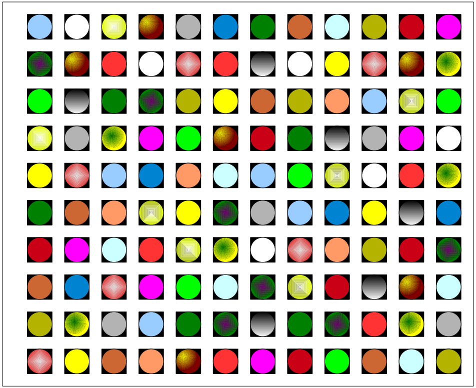
each temporal key encrypted **by m randomly chosen predistribution keys**

Key redistribution scheme
how does it work?



predistribution keys used to encrypt temporal key

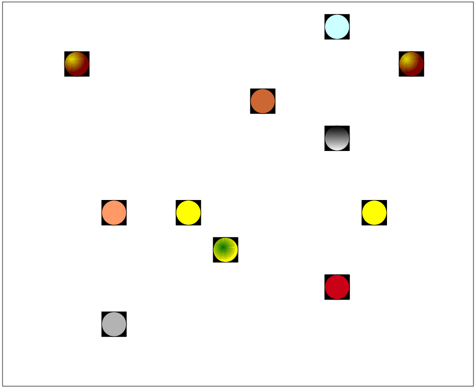
Key redistribution scheme
how does it work?



an assigment of all temporal keys to predistribution keys

Key redistribution scheme

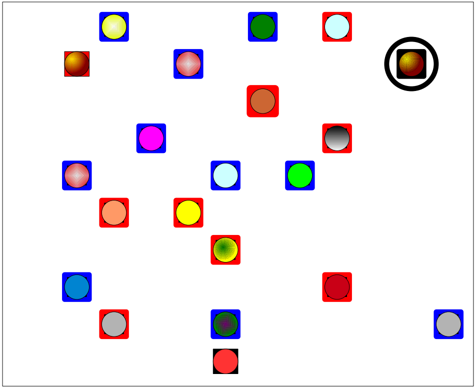
how does it work?



Temporal keys received by device A

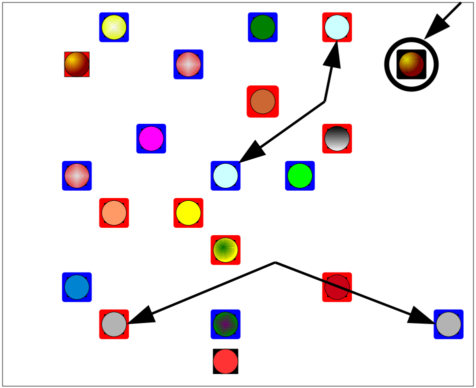
Key redistribution scheme

how does it work?



Temporal keys received by devices A and B

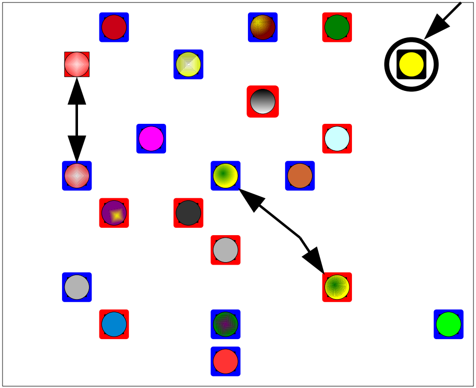
Key redistribution scheme
how does it work?



Temporal keys received by devices A and B

Key redistribution scheme

how does it work?



Temporal keys received by devices A and B for another session

Key redistribution scheme

how does it work?

Summary

- devices A and B may share a temporal key K_i' because:
  - K_i' was broadcasted as $E_{K_u}(K_i')$ and A knows K_u
  - K_i' was broadcasted as $E_{K_v}(K_i')$ and B knows K_v

  while A does not know K_v and B does not know K_u.

Key redistribution scheme
how does it work?

Summary

- devices A and B may share a temporal key K_i' because:
  - K_i' was broadcasted as $E_{K_u}(K_i')$ and A knows K_u
  - K_i' was broadcasted as $E_{K_v}(K_i')$ and B knows K_v

  while A does not know K_v and B does not know K_u.
- after broadcasting new temporal keys K_u and K_v does not help to share a key, since this time they encrypt different keys, say

$$E_{K_u}(K_r''), \quad E_{K_v}(K_z'')$$

Key redistribution scheme
properties

while A talking with B:

- after redistribution of temporal keys they share different keys
- an adversary impersonating B has to hold appropriate predistribution keys possessed by B

It does not suffice to hold some key of B in order to impersonate B or eavesdrop the whole communication of B. Now it is necessary to hold all or most keys of B!

Analytic results

Method used

combinatorial classes ...

Results

exact values for the expected number of shared:

- predistribution keys
- temporal keys

Analytic results

Expected number of shared temporal keys χ

Suppose that each predistribution key is broadcasted m times, and each device holds $k = \Theta(\sqrt{n})$ out of n predistribution keys. Then

$$E(\widetilde{\chi}) = \frac{m}{n}k^2 + O\left(\frac{1}{\sqrt{n}}\right) .$$

Precise values for any n, m, k are given in the paper

Corollary

so for $m = 2$ devices A and B should have 2 shared temporal keys!
From a random pair of predistribution keys!

Securing key predistribution

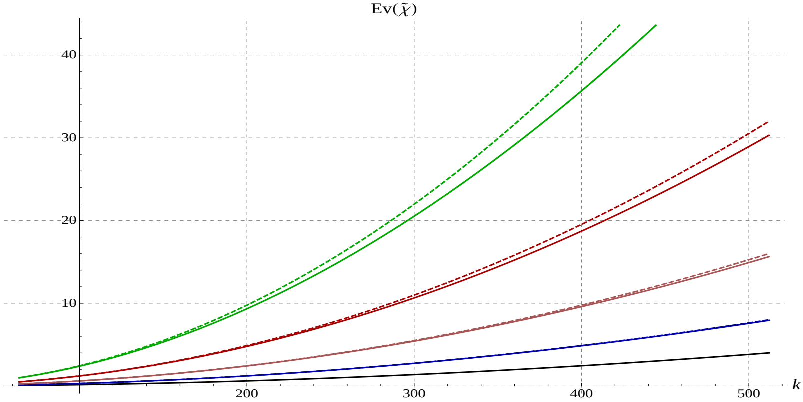
Analytic results

Number of shared keys

Assume that the key pool $\widetilde{\mathcal{K}}$ contains n/m keys, each encrypted with m different keys from $\mathcal{K}$ ($|\mathcal{K}| = n$) during the key update. Assume that each device holds exactly k keys each from the pool $\mathcal{K}$. Then :

1. the expected number of keys from $\mathcal{K}$ shared by devices A and B chosen at random equals
$$k^2/n,$$
2. the expected number of keys from $\widetilde{\mathcal{K}}$ shared by A and B equals
$$\frac{n\left(\binom{n}{k} - \binom{n-m}{k}\right)^2}{m\binom{n}{k}^2}$$

Analytic results



Rysunek: The expected number of temporal keys shared by A and B for $n = 2^{16}$, $2^6 \leq k \leq 2^9$ and $m = 1$ (black plot), $m = 2$ (blue plot), $m = 4$ (pink plot), $m = 8$ (red plot), $m = 16$ (green plot) (dashed plots present approximations from the previous slide).

Attack cost

Attack effectiveness

Let n be the pool size, k number of keys for each device, $m =$ number of copies of each temporal key.
Let $T_{A,B}$ be a set of temporal keys shared by the devices A and B. Let Ad denote the set of the temporal keys held by an adversary.
Then

1. If $|Ad| = \sqrt{n}$, then $\Pr[T_{A,B} \subseteq Ad] \leq (\frac{m}{\sqrt{n}})^m$.
2. If $|Ad| < \frac{n}{m2^{1/m}} \approx \frac{n}{m}(1 - \frac{\ln 2}{m})$, then $\Pr[T_{A,B} \subseteq Ad] < \frac{1}{2}$.

Full key update

Work in progress

We are working on the scheme such that the keys change fully at the transmission.
While the adversary cannot get an advantage and collect more keys as he had.
Based on key predistribution with projection spaces.

Conclusions

surprising advance that make predistribution effective and reliable without a substantial cost

Publications

Bibliography

- **“Securing Random Key Predistribution against Compromise via Node Captures”**, J. Cichoń, J. Grząślewicz, M. Kutyłowski ALGOSENSORS’2009, Rhodos, Greece, LNCS 5304, 64-75,
- **“From Key Predistribution to Key Redistribution ”**, J. Cichoń, Z. Gołębiewski, M. Kutyłowski, ALGOSENSORS 2010, Bordeaux, France, LNCS - in print
  invited to Special Issue of Theoretical Computer Science

Acknowledgment

This work has been partially supported by EU within the 7th Framework Programme, contract 215270 (FRONTS)

Thanks for your attention!

Contact data

1. Miroslaw.Kutylowski@pwr.wroc.pl
2. http://kutylowski.im.pwr.wroc.pl
3. +48 71 3202109, fax: +48 71 320 2105