

Information Security for Sensors

Overwhelming Random Sequences and Permutations

Shlomi Dolev, Niv Gilboa, Marina
Kopeetsky, Giuseppe Persiano, and
Paul G. Spirakis

General Outline

- Introduction and Motivation
- Permutation Revealing Protocol PRP
- Permutation Encrypted Protocol PEP
- Conclusions

Bounded Storage Model

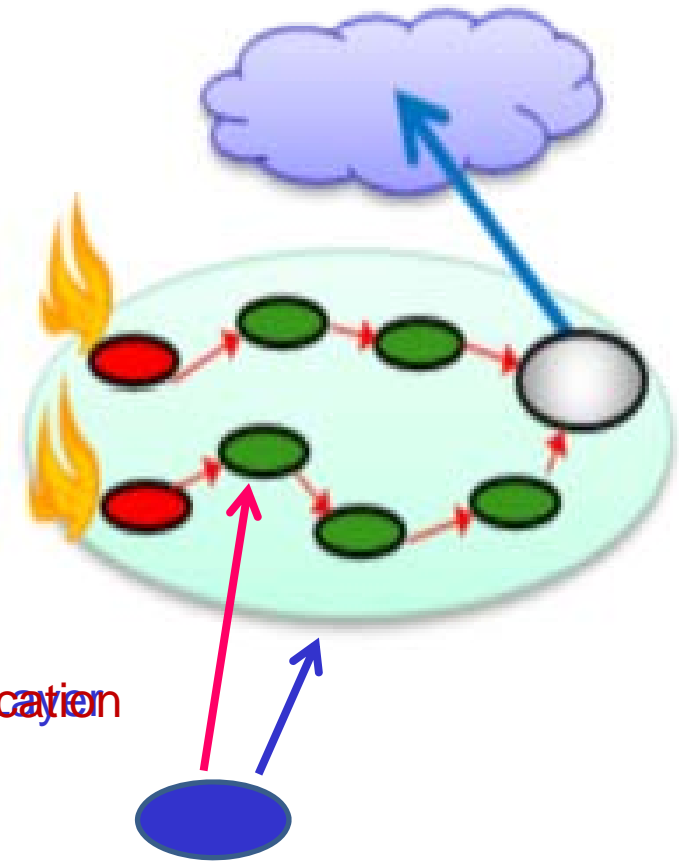
- Introduced by Maurer ('92)
- Adversary
 - Adversary has bounded storage capacity
 - A source of random bits broadcasts more traffic than adversary can store
- Task: Shared Key Establishment
- Information Theoretic secure



Generating a Shared Key in BSM

- Key sharing between Alice and Bob is possible without sharing bits before protocol begins (Maurer, 1992)
- Ratio between storage capacity of Alice, Bob, and Adversary (Dzeiembowski, Maurer, 2004)
It is impossible to overcome Birthday paradox
- Schemes for key expansion (Aumann, Rabin, Ding, 2002; Lu 2004; Vadhan 2004).
 - Our protocols are comparable to protocols of Aumann, Ding, Rabin.

Secure Physical communication authentication



Our Contribution

- Information Theoretic Secure key exchange schemes for BSM
 - Permutation Revealing Protocol PRP
 - Permutation Encrypted Protocol PEP
- Especially suitable for resource constrained & wireless environment
 - Simple - to implement and to run
 - Efficient
 - Uses typical physical implementation
 - Bits are sent in blocks/frames
 - Random source works in parallel over several channels

Setting and Notation

Wireless Network WN

0	0	1
---	---	-----	-----	---

encrypted message

0	0	1
---	---	-----	-----	---

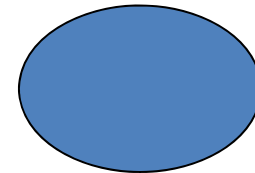
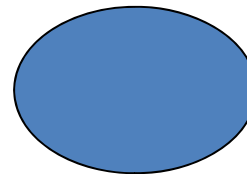
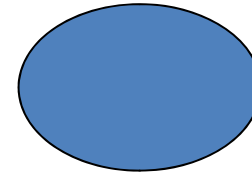
m bit OTP

Memory $s < s_{AD} < n$

**Limited to store
sequence bits ...**

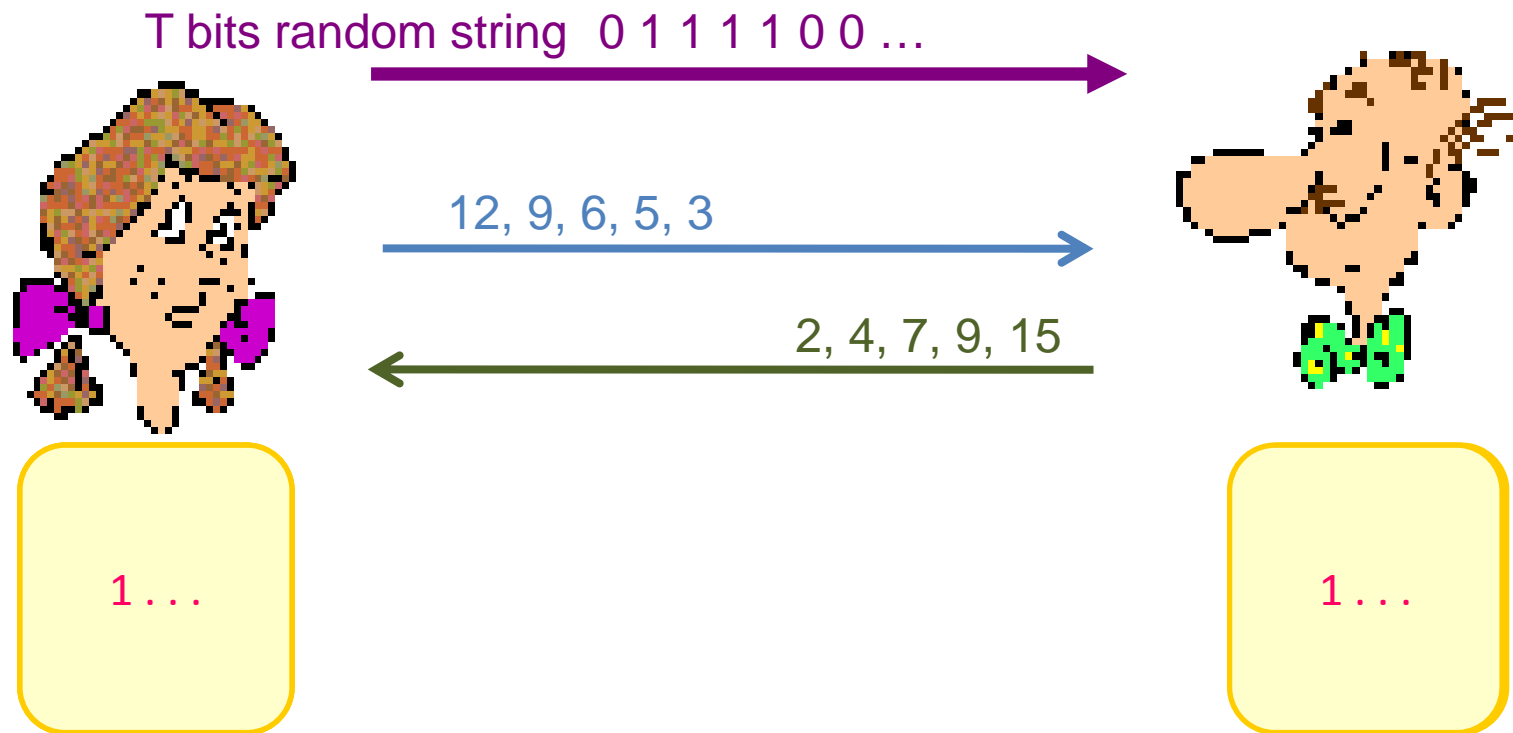


s



Setting and Notation

Process 1- Generating an Initial Key (Dziembowski, Maurer, 2004)



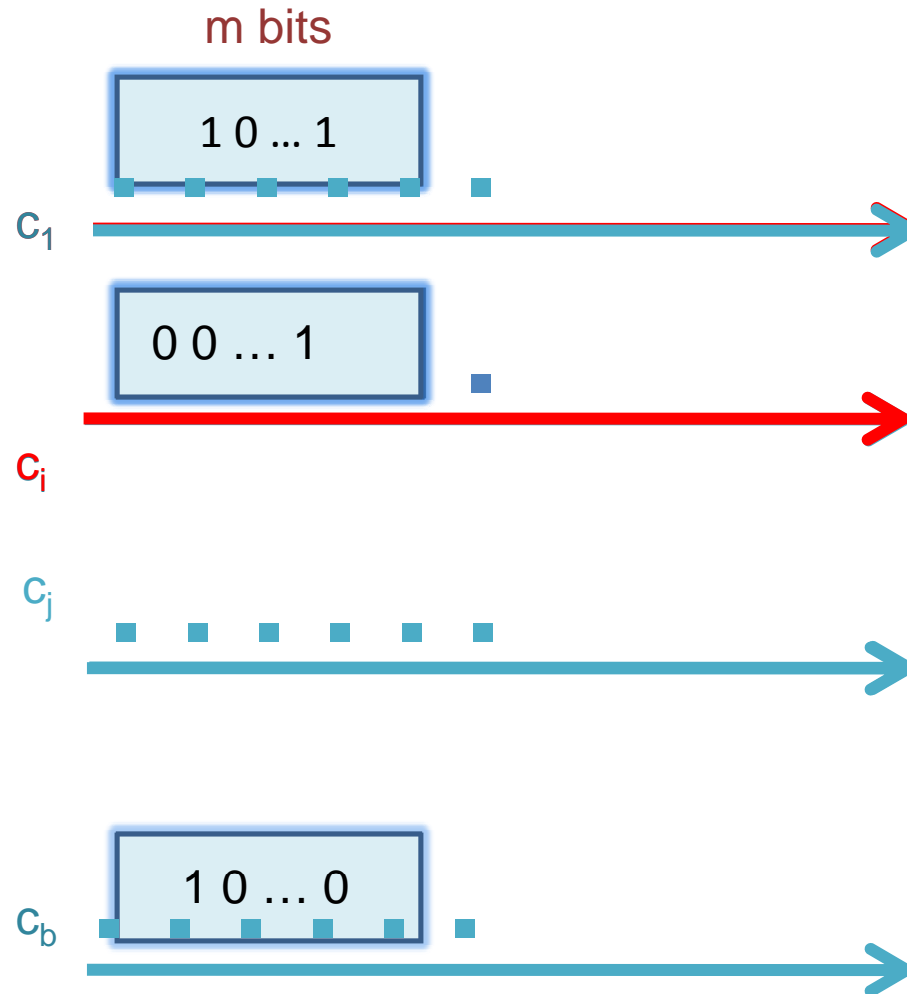
By **Birthday paradox**: for a single shared bit

Record $O(T^{1/2})$ random bits and their indexes

$O(T^{1/2} \log T)$ bits of memory

PRP Phase 1 - Sampling

Input $\log b$ ($\log m + k$) secret bits shared in Process 1
($\log m + k$) indexes of channels $b_1 \dots b_{\log m + k}$
Repeat $\log m + k$ times

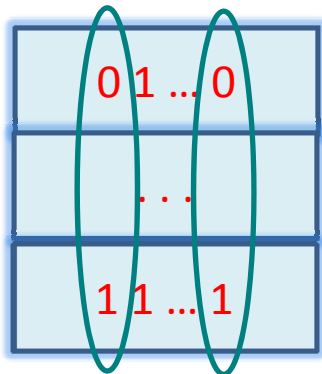
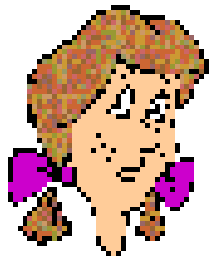
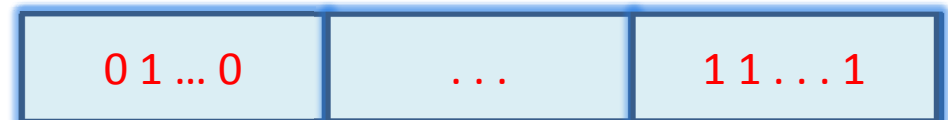


PRP Phase 2 - Extraction & Permutation Sharing

$m (\log m + k)$ shared bits

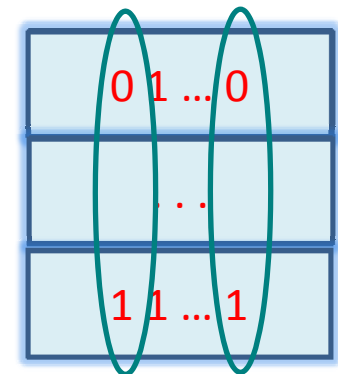


$m (\log m + k)$ shared bits



XOR

$\pi = \{12, 10, 1, 8, 6, \dots\}$



XOR

$OTP = (OTP_1 \dots OTP_m)$

Matrix size $(\log m + k) * m$

Why PRP works?

$$\lambda = (\log m + k)$$

$$\text{Assume } s_{\text{Ad}} < bm\lambda/2$$

After π is revealed

Prob (Ad obtains correct bits) =

Prob (Ad stores all OTP_j in j -th column) =

Number of λ tuples
out of
 $bm\lambda/2$ stored bits

$$\frac{\binom{bm\lambda}{\lambda}}{b^\lambda \binom{m\lambda}{\lambda}} \leq 2^{-\lambda}$$

Number of possible channels

Number of λ tuples among $m\lambda$ bits

Prob (Ad obtains any of m bits of OTP) =

$$\sum_{i=1}^m 2^{-\lambda} = 2^{-k}$$

Permutation Encrypted Protocol PEP

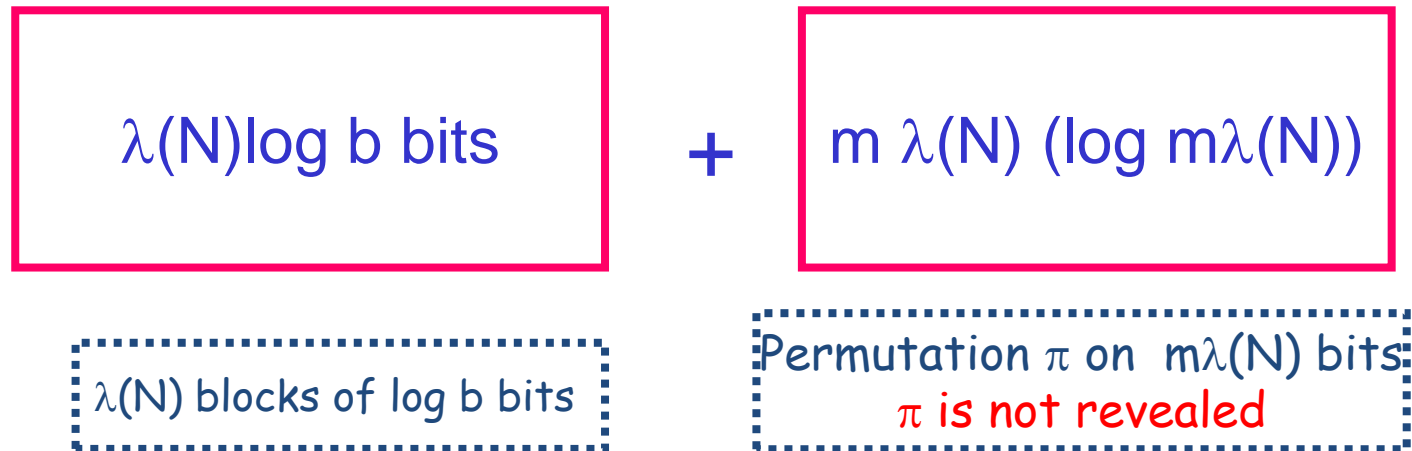
Motivation:

Shared bits that define the permutation are used by PRP **ONLY** once, and can be used multiple times...

Permutation Encrypted Protocol PEP

OTP in PEP is reusable for an exponential (in the security parameter k) number of encryptions.

PRP requires more shared bits in the beginning



N number of rounds to use the same shared permutation

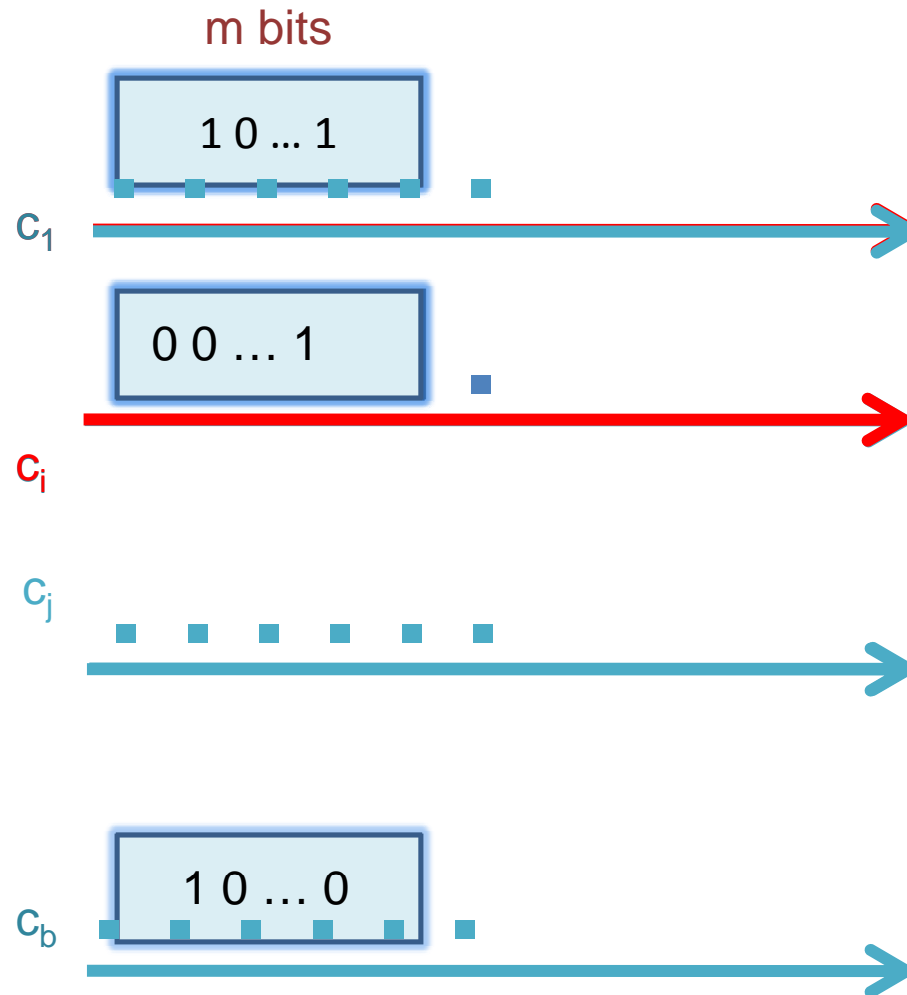
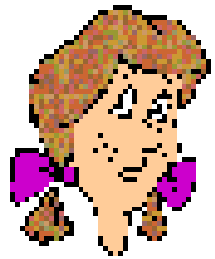
$\lambda(N) = \log(mN+k)$

k security parameter

PEP- Sampling

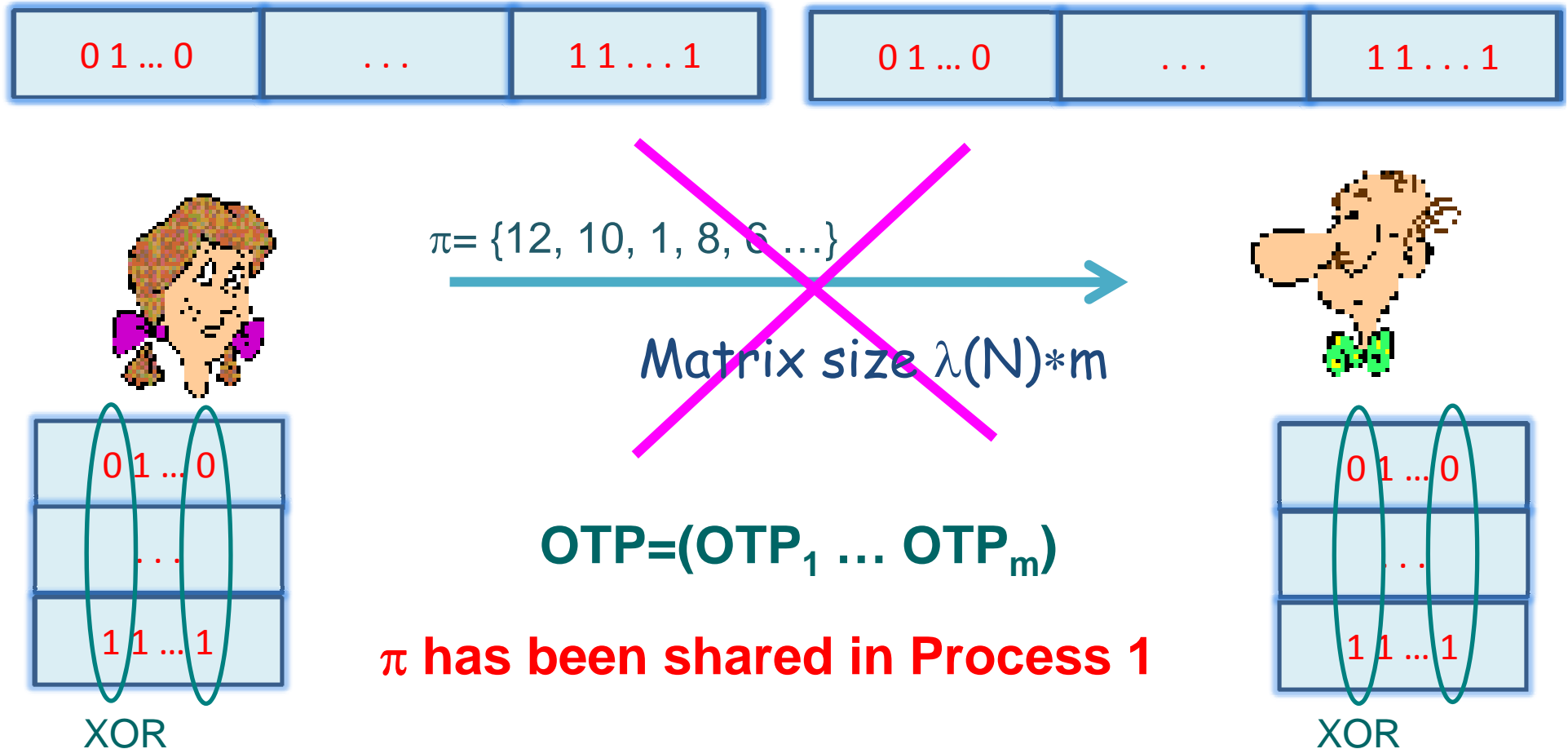
Repeat $\lambda(N)$ times

N number of rounds while OTP is reusable



PEP Phase 2 - Extraction

1. Alice and Bob repeat this process N times
2. Each time Alice sends new random bits and Alice and Bob use the same π



Why PEP works?

Prob (Ad obtains correct bit of OTP) =

$$\left(\frac{\frac{bm\lambda(N)}{2}}{\lambda(N)} \right) \leq 2^{-\lambda(N)}$$

Number of $\lambda(N)$ tuples
out of $bm\lambda/2$ stored bits

Number of
possible
channels

Number of λ tuples among $m\lambda$
bits

Prob (Ad obtains any of m bits of OTP) =

$$\sum_{i=1}^{mN} 2^{-\lambda(N)} = 2^{-k}$$

Comparison With Previous Schemes

- Efficiency measure- expansion factor
 - Ratio between length of OTP and length of initial key
- Unified "Sample and Extract Approach" (Lu and Vadhan)
 - Expander graph based schemes, high computational complexity

Comparison With Previous Schemes

Paper Initial secret
for producing m
shared bits

Ding-Rabin [1]

$$k \log n$$

$k > \log b$: our expansion factor is better [1], [2]

Dziembowski
-Maurer [2]

$$k \log n$$

Lu [3]

$$(k + \log n)^2 / \log n$$

$k > \log b \log n$: our expansion factor is better than [3]

Vadhan [4]

$$k + \log n$$

Our work

$$\log b(k + \log m)$$

$b=2$: our expansion factor is better by constant factor than [4]

Comparison With Previous Schemes

- Efficiency measure-number of bits read from random source
 - Wireless traffic is sent in blocks/frames of α bits

- As α grows our scheme becomes more efficient
 - $\alpha = m$ is optimal

n = random string length

Paper	Number of bits read
Ding-Rabin [1]	$m\alpha$
Dziembowski-Maurer [2]	$m\alpha$
Lu [3]	$m\alpha$
Vadhan [4]	$(k + \log n)\alpha$
Our work	$\lceil m/\alpha \rceil \alpha(k + \log m)$

Our scheme reads less bits than any other scheme when:

$$\left\lceil \frac{m}{\alpha} \right\rceil \leq \frac{\log n + k}{\log m + k}$$

Conclusions

- **A new technique**
 - Defining blocks/sections of random sequences, rather than bits
 - Random permutation of the bits among the concatenation of the chosen sections
- **Improving Expansion Factor of PEP**
 - Run PRP to expand the initial key for PEP
 - Perform PEP with the expanded shared key



Conclusions

- Improving Key Exchange Algorithms
 - Sample blocks of bits instead of distinct bits
 - Apply random permutation on random bits
- PRP and PEP fit multi-frequency wireless communication
 - Choice subset of frequencies implies exponentially growing security parameter
- Establishing short **secret from scratch.**

Thank you

