ROUND-OPTIMAL AUTHENTICATED KEY AGREEMENT FROM WEAK SECRETS

STOC 2009  Yevgeniy Dodis and Daniel Wichs (NYU)
Symmetric Key Cryptography

- Alice and Bob share a secret key $W$ and want to communicate securely over a public channel.
  - **Privacy:** Eve does not learn anything about the message.
  - **Authenticity:** Eve cannot modify or insert messages.

- This is a well-studied problem with many solutions:
  - **Information-theoretic security** (going back to Shannon in 1949).
  - **Computational security** (formally studied since the 1970s).
    - e.g. One Way Functions, Block Ciphers (AES).
Standard symmetric key primitives assume that Alice and Bob share a *uniformly random key* $W$. This is unreasonable/undesirable in many scenarios.

**Imperfect keys:**
- Human memorable passwords
- Biometrics

**Partially Compromised keys:**
- Side-channel attacks
- Malware attacks in the Bounded Retrieval Model
- Quantum Key Agreement, Wiretap Channel
General View of Weak Secrets

- We want to make *minimal* secrecy assumptions.
  - The secret $W$ comes from an arbitrary distribution which is \textit{“sufficiently hard to guess”}.
    - Formalized using conditional min-entropy.

- Two important domain-specific problems:
  - \textbf{Biometrics}: Successive scans of the same biometric are noisy.
  - \textbf{Bounded Retrieval Model}: Cannot read all of $W$ efficiently.

- \textbf{Goal}: Alice and Bob run a “key agreement protocol” to agree on a (nearly) uniform, random key $R$ by communicating over a public channel controlled by an active adversary Eve.
The secret $W$ is a random variable which is “sufficiently hard to guess” (conditioned on some side-information $Z$).

Formalized using conditional min-entropy. If entropy is $k$ then $W$ can’t be guessed with probability better than $2^{-k}$.

**Goal:** Base symmetric key cryptography on weak secrets.

**Authenticated Key Agreement.** Alice and Bob start out with a weak secret $W$ and agree on uniform key $K$, by running a protocol over a public channel.
Can be solved computationally using “Password Authenticated Key Exchange” [BMP00, BPR00, KOY01, GL01, CHK+05, GL06]

Alice and Bob can exchange arbitrarily many session keys using W.

Strong guarantees even if W comes from a very small dictionary.

Only achieves computational security using public key cryptography.

Efficient solutions require a common reference string or the random oracle model.

Interactive protocol: current best requires three flows.

This talk: focus on information theoretic security.

Only get a “one-time” key agreement protocol.

Need W to have “enough entropy”.

Minimalist approach – no assumptions!

Can do non-interactive with CRS or one-round without CRS.
This Talk vs.
“Password Authenticated Key Exchange”

“Password Authenticated Key Exchange”
[BMP00, BPR00, KOY01, GL01, CHK+05, GL06]

☑ Computational security using public key cryptography.

☑ Alice and Bob can exchange arbitrarily many session keys using W.

☑ Strong guarantees even if W comes from a very small dictionary.

☑ Efficient solutions require a common reference string (CRS) or the random oracle model.

☑ Interactive protocol: current best requires three rounds of communication.

This Talk:

☑ Information-Theoretic security. No assumptions.

☑ “One-time” key agreement protocol.

☑ Final key length is smaller than entropy of W.

☑ Two rounds without a CRS.
Alice and Bob apply some deterministic function $f$ to $W$ such that $K=f(W)$ is uniformly random.

No difference between active/passive adversary.

Impossible. There is a random variable $W$ distributed over $\{0,1\}^n$ with $n-1$ bits of entropy and the first bit of $f(W)$ is a constant!
Non-Interactive (One Round) Key Agreement?

- Alice computes a key $K$ and a “helper” $X$ which she sends to Bob.
- Bob uses $W$, $X$ to recover $K$.
- Security Guarantees:
  - Key $K$ looks random even if Eve sees $X$.
  - Eve cannot cause Bob to recover $K' \neq K$. 
An Alternative View of Non-Interactive Key Agreement.

- A protocol across time.
  - Helper P is stored on “public storage”
  - Alice can use it in the future to recover K from W.
- Future Alice cannot “interact” with past Alice.
Non-Interactive Key Agreement with Passive Attacker

Randomness Extractor. A randomized function $\text{Ext}$.
- **Input:** a weak secret $W$ and a random seed $X$.
- **Output:** extracted randomness $K = \text{Ext}(W;X)$.
- $K$ looks (almost) uniformly random even given the seed $X$.
- Can extract almost all of the entropy of $W$. 

Choose seed $X$. 

Bob $\quad W \quad K = \text{Ext}(W;X)$

Eve $\quad X$

Alice $\quad W \quad K = \text{Ext}(W;X)$
Non-Interactive Key Agreement with Active Attacker

Bob

W

K' = Ext(W; X')

X'

Eve

X

K = Ext(W; X)

Alice

W

Choose seed X.

What if Eve is active?

- Can modify the seed X to some other value X' and cause Bob to recover an incorrect key K' = Ext(W; X').
- Eve may even fully know K'!
Is there some other construction of non-interactive authenticated key agreement?

Our answer: Impossible when \( k \leq n/2 \) (\( k \) = entropy of \( W \), \( n \) = length of \( W \)).

Solutions exist for \( k > n/2 \) [MW97] [DKRS06] [KR09].

- Extracted key is short: \( k-n/2 \) bits. Communication is \( n-k \) bits.

For \( k \leq n/2 \) we need interaction.
A Simple Protocol in the CRS Model

Common Reference String:

Bob

W

Eve

X

Alice

Choose seed X.

K = Ext(W;X)

Make the seed X a common reference string.
- Chosen by some trusted party (Microsoft?) and hardcoded into hardware/software. Assumed to be public (seen by Eve).
- No communication required!
- Problem: Requires a trusted party.
- Problem: What if Eve can learn information about W adaptively.
  - e.g. Side-channel attacks, Bounded Retrieval Model.
  - Not a problem for biometrics.
Side note: biometrics are noisy…

**Common Reference String:** $X$

Solution: Alice sends some “sketch” of $W$ to Bob which allows him to “correct” differences and recover $W$ from $W'$ without revealing (much) about $W$ to Eve. [DORS04]

... but now we need to worry about active attacks again. What if Eve modifies the “sketch”?

Solution 1 (No CRS): Requires $k > n/2$ [DKRS06].

Solution 2 (CRS): Works for any $k$ [CDFPW08].
Interactive Key Agreement Protocols

- The only known interactive protocol is a construction by Renner and Wolf from 2003.
  - Requires many rounds of interaction.
    - Not constant - proportional to security parameter.
    - In practice 100s of rounds would be required.
- Question: What is the minimal number of rounds? Is a two round interactive protocol possible?
  - Yes - we show that two rounds is enough!
Interactive Key Agreement Protocols

- The hard part is *message authentication*.
  - Implies Key Agreement
  - Root of inefficiency in Renner-Wolf construction.
- We construct a **two round** message authentication protocol and then convert it into a **two round** key agreement protocol.
- Protocols have a challenge-response structure.
  - Bob sends a *random challenge* to Alice. Alice uses the challenge to authenticate a message to Bob.
I.T. MACs: Authentication using strong keys.

- Warm-up: what if Alice and Bob already share a strong (uniform) key?
- I.T. Message Authentication Code (MAC):
  - For any $m$, if adversary sees $\sigma = \text{MAC}_R(m)$, cannot forge $\sigma' = \text{MAC}_R(m')$ for $m' \neq m$.
  - Known constructions with excellent parameters.
Authentication with Weak Keys: Protocol Template

- **Idea:** If Eve is passive in round 1, then Alice shares a “good” key with Bob and can authenticate a message in round 2.

- **Problem:** What if Eve modifies $X$?

**Diagram:**

- **Bob:** $W$
  - $R = \text{Ext}(W;X)$
  - $\sigma = \text{MAC}_R(m)$

- **Alice:** $W$
  - $R = \text{Ext}(W;X)$
  - $\sigma = \text{MAC}_R(m)$

- **Eve:**
  - $X$
  - $m, \sigma$

**Message:**

- $m$

---

- **Idea:** If Eve is passive in round 1, then Alice shares a “good” key with Bob and can authenticate a message in round 2.

- **Problem:** What if Eve modifies $X$?
Authentication with Weak Keys: Protocol Template

Bob

$W$

$R = \text{Ext}(W;X)$

Eve

$X' \rightarrow R' = \text{Ext}(W;X')$

Alice

$W$

Message:

$m$
Authentication with Weak Keys: Protocol Template

\[ R = \text{Ext}(W;X) \]

\[ R' = \text{Ext}(W;X') \]

\[ \sigma = \text{MAC}_{R'}(m) \]

Message: \( m \)
Authentication with Weak Keys: Protocol Template

Bob

\[ R = \text{Ext}(W;X) \]

Alice

\[ R' = \text{Ext}(W;X') \]

\[ \sigma' = \text{MAC}_R(m') \]

\[ \sigma = \text{MAC}_{R'}(m) \]

- Eve gets to see \( \text{MAC}_{R'}(m) \) and must forge \( \text{MAC}_R(m') \).
- Non-standard security notion.
- If \( R \) and \( R' \) are related then Eve may succeed!
Goal: Construct special extractors and MACs for which the protocol is secure.

- Build a special non-malleable extractor Ext so that $R = \text{Ext}(W;X)$ and $R' = \text{Ext}(W;X')$ are related in only a limited way.
- Build a special MAC which is resistant to the limited types of related key attacks that are allowed by the extractor.
  - Seeing $\text{MAC}_{R'}(m)$ does not allow the adversary to forge $\text{MAC}_R(m')$.

Two approaches:
- Approach 1: A very strong non-malleability property for Ext + standard MAC. (Non-Constructive)
- Approach 2: A weaker non-malleability property for Ext + special MAC. (Constructive)
Approach 1: Fully Non-Malleable Extractors

- Adversary sees a random seed $X$ and produces an arbitrarily related seed $X' \neq X$.
- Let $R = \text{nmExt}(W;X)$, $R' = \text{nmExt}(W;X')$.

**Non-malleable Extractor:** $R$ look uniformly random, even given $X$, $X'$, $R'$.

- Extremely strong property. No existing constructions achieve it.
  - Natural constructions susceptible to many possible malleability attacks.
  - Not immediately clear that it can be achieved at all!

- Surprising result: Non-malleable extractors exist.
  - Can extract almost $\frac{1}{2}$ of the entropy of $W$ (optimal).
  - Follows from a (non-standard) probabilistic method argument.
  - Does not give us an efficient candidate.
Approach 1: Fully Non-Malleable Extractors

If Eve does not modify $X$, then Alice and Bob share a uniformly random key $R' = R$.
- Standard MAC security suffices.

If Eve modifies $X$, then Bob’s key $R$ is random and independent of Alice’s $R'$.
- $\text{MAC}_{R'}(m)$ does not reveal anything about $R$. 

$\sigma' = \text{MAC}_R(m')$
Approach 1: Summary

- Strong extractor property: “fully non-malleable” extractor.
- Standard MACs.

- Parameters: To authenticate an $m$ bit message with security $2^{-\lambda}$ using an $n$-bit secret $W$ we need:
  - The entropy of $W$ is $k > O(\log(\log(n)) + \log(m) + \lambda)$.
  - Communication $m + O(\log(n) + \log(m) + \lambda)$.

- Unfortunately, we do not have an efficient construction of fully non-malleable extractors.
  - Great open problem!

  Solved for $k>n/2$ [DLWZ11, Li12, DY13]
Approach 2: “Look-Ahead” Extractors

- Much weaker non-malleability property. The extracted randomness consists of t blocks:

\[
\text{laExt}(W;X) = [R_1, R_2, R_3, R_4, R_5, \ldots, R_t]
\]

\[
\text{laExt}(W;X') = [R'_1, R'_2, R'_3, R'_4, \ldots, R_t]
\]

- Adversary sees a random seed X and modifies it to X'.

**Require:** Any suffix of \( \text{laExt}(W;X) \) looks random given a prefix of \( \text{laExt}(W; X') \).

- Cannot use modified sequence to “look-ahead” into the original sequence.
Approach 2: Constructing “look-ahead” extractors.

- Based on “alternating-extraction” from [DP07].
- Two party interactive protocol between Quentin and Wendy.
- In each round $i$:
  - Quentin sends $S_i$ to Wendy.
  - Wendy sends $R_i = \text{Ext}(W;S_i)$.
  - Quentin computes $S_{i+1} = \text{Ext}(Q;R_i)$
Approach 2: Alternating-Extraction Theorem

- **Alternating-Extraction Theorem**: No matter what strategy Quentin and Wendy employ in the first $i$ rounds, the values $[R_{i+1}, R_{i+2}, \ldots, R_t]$ look uniformly random to Quentin given $[R'_1, R'_2, \ldots, R'_i]$.

<table>
<thead>
<tr>
<th>Quentin</th>
<th>Wendy</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Q, S_1$</td>
<td>$W$</td>
</tr>
</tbody>
</table>

- $S_1 \xrightarrow{R_1} R_1 = \text{Ext}(W;S_1)$
- $S_2 = \text{Ext}(Q;R_1)$
- $S_3 = \text{Ext}(Q;R_2)$
- $S_4 = \text{Ext}(Q;R_3)$
- $R_1 \xrightarrow{S_1} S_1$
- $R_2 \xrightarrow{S_2} S_2$
- $R_2 \xrightarrow{S_3} S_3$
- $R_3 \xrightarrow{S_4} S_4$

- **Assume that**:
  - $W$ is (weakly) secret for Quentin and $Q$ is secret for Wendy.
  - Wendy and Quentin can communicate only a few bits in each round.
  - Can they compute $R_i, S_i$ in fewer rounds?
Approach 2: Alternating-Extraction Theorem

- Intuition: Prior to round $i$, the values $S_i$, $R_i$ look random to Wendy and Quentin respectively.
- True for $i=1$ by extractor security.

<table>
<thead>
<tr>
<th>Quentin</th>
<th>Wendy</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Q, S_1$</td>
<td>$W$</td>
</tr>
</tbody>
</table>

\[
S_1 \xrightarrow{R_1} R_1 = \text{Ext}(W; S_1) \\
S_2 = \text{Ext}(Q; R_1) \xrightarrow{S_2} \ldots \\
S_3 = \text{Ext}(Q; R_2) \xrightarrow{S_3} \ldots \\
S_4 = \text{Ext}(Q; R_3) \\
\]

\[
\]
Approach 2: Alternating-Extraction Theorem

- **Intuition:** Prior to round $i$, the values $S_i, R_i$ look random to Wendy and Quentin respectively.
- **Induction:** Assume true for $i$, then for $i+1$...

\[
\begin{align*}
S_2 &= \text{Ext}(Q;R_1) \\
R_1 &= \text{Ext}(W;S_1) \\
S_3 &= \text{Ext}(Q;R_2) \\
R_2 &= \text{Ext}(W;S_2) \\
S_4 &= \text{Ext}(Q;R_3) \\
R_3 &= \text{Ext}(W;S_3)
\end{align*}
\]
Approach 2: Look-Ahead Extractor based on Alternating Extraction

Define: $\text{laExt}(W; X) = [R_1, R_2, R_3, \ldots, R_t]$ where the extractor seed is $X = (Q, S_1)$.

- $S_1 = \text{Ext}(Q; S_1)$
- $R_1 = \text{Ext}(W; S_1)$
- $S_2 = \text{Ext}(Q; R_1)$
- $R_2 = \text{Ext}(W; S_2)$
- $S_3 = \text{Ext}(Q; R_2)$
- $R_3 = \text{Ext}(W; S_3)$
- $S_4 = \text{Ext}(Q; R_3)$

Diagram:

- Quentin: $Q, S_1$
- Wendy: $W$
- Quentin: $Q, S_1$ (with $S_1$ and $R_1$)
- Wendy: $W$ (with $S_1$ and $R_1$)
- Quentin: $Q, S_1$ (with $S_2$ and $R_2$)
- Wendy: $W$ (with $S_2$ and $R_2$)
- Quentin: $Q, S_1$ (with $S_3$ and $R_3$)
- Wendy: $W$ (with $S_3$ and $R_3$)
- Quentin: $Q, S_1$ (with $S_4$ and $R_4$)
- Wendy: $W$ (with $S_4$ and $R_4$)
Approach 2: Look-Ahead Extractor based on Alternating Extraction

Define: \( \text{laExt}(W;X) = [R_1, R_2, R_3, \ldots, R_t] \)

where the extractor seed is \( X = (Q, S_1) \).

Alternating-Extraction in Bob’s head

Sample \( X = (Q, S_1) \)

\( X = (Q, S_1) \)

Alternating-Extraction in Alice’s head

\( X' = (Q', S'_1) \)

Eve

Bob

W

Alice

W
Approach 2: Look-Ahead Extractor based on Alternating Extraction

A modified seed $X'$ corresponds to a modified strategy by Quentin in Alice’s head.

$$\text{laExt}(W;X) = [R_1, R_2, R_3, \ldots, R_t]$$

$$\text{laExt}(W;X') = [R_1', \ldots]$$

Quentin Wendy

$Q, S_1$ $W$

$S_1$ $S_2$ $S_3$ $S_4$

$R_1$ $R_2$ $R_3$

$R_1 = \text{Ext}(W;S_1)$

$R_2 = \text{Ext}(W;S_2)$

$R_3 = \text{Ext}(W;S_3)$

Wendy

$Q'$ $S_1'$ $W$

$S_1'$ $S_2'$ $S_3'$

$R_1'$ $R_2'$

$R_1' = \text{Ext}(W;S_1')$
Approach 2: Look-Ahead Extractor based on Alternating Extraction

- A modified seed $X'$ corresponds to a modified strategy by Quentin.

$$\text{laExt}(W;X) = [R_1, R_2, R_3, \ldots, R_t], \quad \text{laExt}(W;X') = [R'_1, R'_2, R'_3, \ldots, R'_t]$$

\[\begin{align*}
\text{Quentin} & \quad \text{Wendy} \\
Q, S_1 & \quad W \\
S_1 & \quad R_1 = \text{Ext}(W;S_1) \\
S_2 = \text{Ext}(Q;R_1) & \quad S_2 \\
S_3 = \text{Ext}(Q;R_2) & \quad S_3 \\
S_4 = \text{Ext}(Q;R_3) & \\
\text{Quentin} & \quad \text{Wendy} \\
Q', S'_1 & \quad W \\
S'_1 & \quad R'_1 = \text{Ext}(W;S'_1) \\
S'_2 = \text{Ext}(Q';R'_1) & \quad S'_2 \\
S'_3 = \text{Ext}(Q';R'_2) & \quad S'_3 \\
S'_4 = \text{Ext}(Q';R'_3) & \\
\end{align*}\]
Approach 2: “Look-Ahead” Extractors

\[ R = \text{laExt}(W;X) \]

\[ R' = \text{laExt}(W;X') \]

\[ \sigma = \text{laMAC}_R(m) \]

\[ \sigma' = \text{laMAC}_R'(m') \]

\[ m', \sigma' \quad m, \sigma \]

- \( \text{laExt} \) ensures that “look-ahead” property holds between \( R, R' \).
- Need: \( \text{laMAC} \) which ensures that Eve cannot predict \( \text{laMAC}_R(m') \) given \( \text{laMAC}_R'(m) \).
Approach 2: Authentication using Look-Ahead

- Ensure that given $\text{laMAC}_R(m)$ it is hard to predict $\text{laMAC}_R(m')$ where $R = [R_1, R_2, \ldots, R_t]$, $R' = [R'_1, R'_2, \ldots, R'_t]$ have “look-ahead” property.
- No guarantees from standard MACs.
- Idea for 1 bit ($t=4$): $R = [R_1, R_2, R_3, R_4]$.
  - $\text{laMAC}_R(0) = [R_1, R_4]$  
  - $\text{laMAC}_R(1) = [R_2, R_3]$
Approach 2: Authentication using Look-Ahead

- Ensure that given \( \text{laMAC}_R(m) \) it is hard to predict \( \text{laMAC}_R(m') \) where \( R = [R_1, R_2, \ldots, R_t] \), \( R' = [R'_1, R'_2, \ldots, R'_t] \) have “look-ahead” property.

- No guarantees from standard MACs.

- Idea for 1 bit (\( t=4 \)): \( R = [R_1, R_2, R_3, R_4] \).
  - \( \text{laMAC}_R(0) = [R_1, R_4] \) \( \text{laMAC}_R(1) = [R_2, R_3] \)
  - \( \text{laMAC}_{R'}(1) = [R'_2, R'_3] \) \( \text{laMAC}_{R'}(0) = [R'_1, R'_4] \)
  - \( R_4 \) looks random given \( R'_2, R'_3 \)
  - \( R_2, R_3 \) look random given \( R'_1 \). \( R'_4 \) isn’t long enough to “reveal” both of them.
  - Easy to generalize to \( m \) bits with \( t=4m \).
In general: Find a collection $\Psi = \{S_1, \ldots S_M\}$ of subsets $S \subseteq \{1, \ldots, t\}$ which are “pairwise top-heavy”.

$S_1 = \{1, 4\}$

$S_2 = \{2, 3\}$

$\text{LaMAC}_R(m) = [R_i : i \in S_m]$ for $m \in \{1, \ldots, M\}$.

Construction with $M = 2^{t/4}$.

Choose orange/blue in each tuple:

$\{(1, 2, 3, 4) (5, 6, 7, 8) (9, 10, 11, 12) \ldots (t-3, t-2, t-1, t)\}$

$S_i = \{(2, 3) (5, 8) \ldots (a+1, a+2) \ldots (t-2, t-1)\}$
Approach 2: Authentication using Look-Ahead

- In general: Find a collection $\Psi = \{S_1, \ldots, S_M\}$ of subsets $S \subseteq \{1, \ldots, t\}$ which are “pairwise top-heavy”.
  
  \[
  \begin{align*}
  S_1 &= \{1, 4\} \\
  S_2 &= \{2, 3\}
  \end{align*}
  \]

- $\text{laMAC}_R(m) = [R_i : i \in S_m]$ for $m \in \{1, \ldots, M\}$.

- Construction with $M = 2^{t/4}$.

- Choose orange/blue in each tuple:

\[
\{(1, 2, 3, 4) (5, 6, 7, 8) (9, 10, 11, 12) \ldots (t-3, t-2, t-1, t)\}
\]

- $S_i = \{(2, 3) (5, 8)\ldots (a+1, a+2)\ldots (t-2, t-1)\}$

- $S_k = \{(1, 4) (5, 8)\ldots (a, a+3)\ldots (t-3, t)\}$
Approach 2: “Look-Ahead” Extractors

- \( R = \text{laExt}(W;X) \)
- \( R' = \text{laExt}(W;X') \)
- \( \sigma = \text{laMAC}_R(m) \)
- \( \sigma' = \text{laMAC}_{R'}(m') \)

- \( \text{laExt} \) ensures that “look-ahead” property holds between \( R, R' \).
- \( \text{laMAC} \) ensures that Eve cannot predict \( \text{laMAC}_R(m') \) given \( \text{laMAC}_{R'}(m) \).
Approach 2: Summary of “look-ahead”

- Constructed a “look-ahead” extractor based on the idea of alternating-extraction.
- Constructed a MAC which is secure against “look-ahead” related-key attacks.

- To authenticate an $m$ bit message with security $2^{-\lambda}$, with an $n$-bit weak secret $W$ we need:
  - The entropy of $W$ is $k > O(m(m + \log(n) + \lambda))$.
  - Communication is $O(m(m + \log(n) + \lambda))$.

- Only efficient for short messages (small $m$).
- Next: show how to construct key agreement by authenticating a very short message!
Key Agreement from Authentication

- Idea: Alice authenticates a seed $Y$ to Bob using an authentication protocol. Shared key is $K = \text{Ext}(W;Y)$.
  - Standard extractor suffices here.
- Problem: May not be secure in general. Authentication protocol may reveal something about $K=\text{Ext}(W;Y)$.
  - This problem occurs in Renner-Wolf construction. Require even more rounds to get key agreement.
- Does not occur in our authentication protocols!
Eve sees $\sigma$ which depends on $W,Y$...

... but information in $\sigma$ is subsumed by $R'$ which is independent of $Y$!

Therefore $K$ looks uniformly random, even given Eve’s view of the authentication protocol (during an active attack).
Final Parameters

- **Efficient construction:** If secret $W$ has length $n$ and entropy $k$ and security parameter is $\lambda$ then the exchanged key is of length: $k - O(\log^2(n) + \lambda^2)$
  - Communication complexity: $O(\log^2(n) + \lambda^2)$.

- **Existential Result:** If secret $W$ has length $n$ and entropy $k$ and security parameter is $\lambda$ then the exchanged key is of length: $k - O(\log(n) + \lambda)$
  - Communication complexity: $O(\log(n) + \lambda)$. 
Properties of Key Agreement Protocol

- Alice derives a key $K$ which stays private no matter what the adversary does.
- Bob confirms that the response is valid. If so then Bob’s key matches Alice’s key.
- Alice can use the key in the second round.
  - Can encrypt and authenticate a message to Bob (I.T. or comp)!
Summary

- Show how to base symmetric key cryptography (information theoretic, computational) on weak secrets.
- Build a round-optimal “authenticated key agreement protocol”.
  - Extends to “Fuzzy” setting, Bounded Retrieval Model
- Interesting new tool: “non-malleable” randomness extractors: (1) fully non-malleable (2) “look-ahead”.
  - Other applications?
  - Open Problem: Efficient construction of fully non-malleable extractors.

Thank You!!!
Idea: Alice sends some “sketch” of $W$ to Bob which allows him to “correct” differences and recover $W$ from $W'$ without revealing (much) about $W$ to Eve. \cite{DORS04}

… but now we need to worry about active attacks again. What if Eve modifies the “sketch”?

Solution 1 (No CRS, 1 round): Requires $k > n/2$ \cite{DKRS06}.
Solution 2 (CRS, 1 round): Works for any $k$ \cite{CDFPW08}.

This paper (No CRS, 2 rounds): Works for any $k$.