Robust Fuzzy Extractors
&
Authenticated Key Agreement from Close Secrets

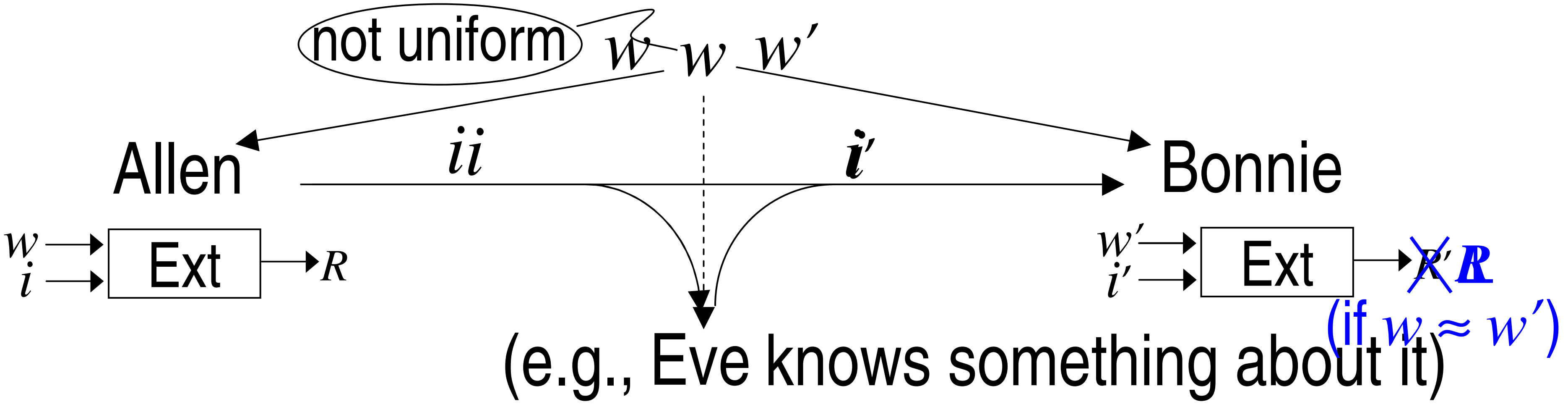
Yevgeniy Dodis
New York University

Jonathan Katz
University of Maryland

Leonid Reyzin
Boston University

Adam Smith
Weizmann → IPAM → Penn State

setting 1: info-theoretic key agreement



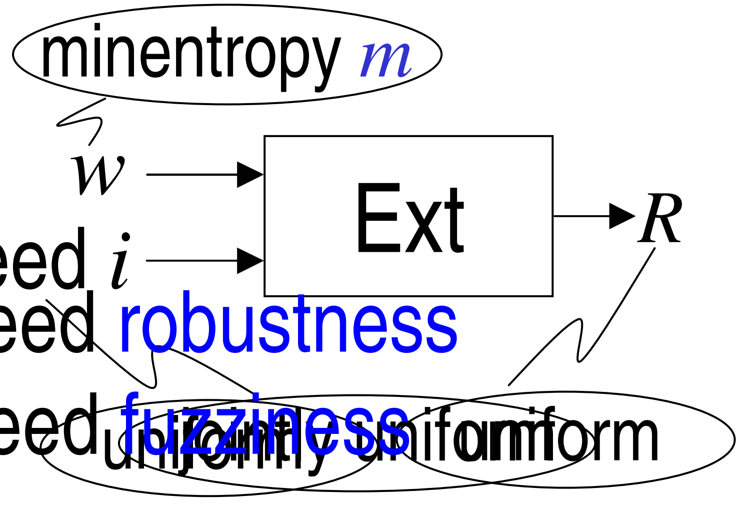
(e.g., Eve knows something about it)

Goal: from a nonuniform secret w

agree on a uniform secret R

No secure channel (else, trivial)

Simple solution: use an extractor

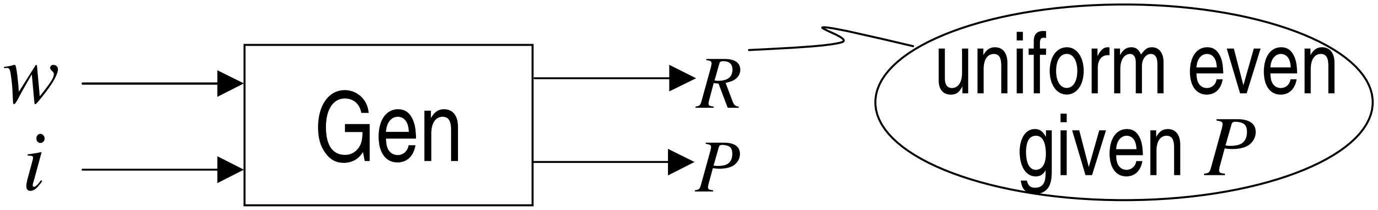


Problem 1: What if Eve is active? Need robustness

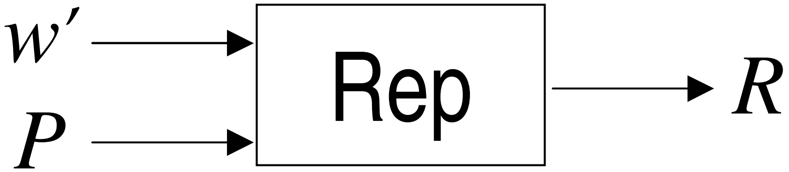
Problem 2: What if w is noisy? Need fuzziness

need: robust fuzzy extractor

- Extraction: generate uniform R from w (+ seed i)

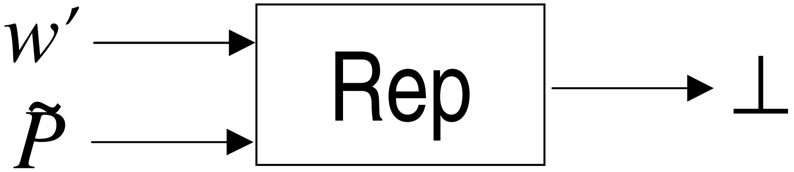


- Fuzziness: reproduce R from P and $w' \approx w$



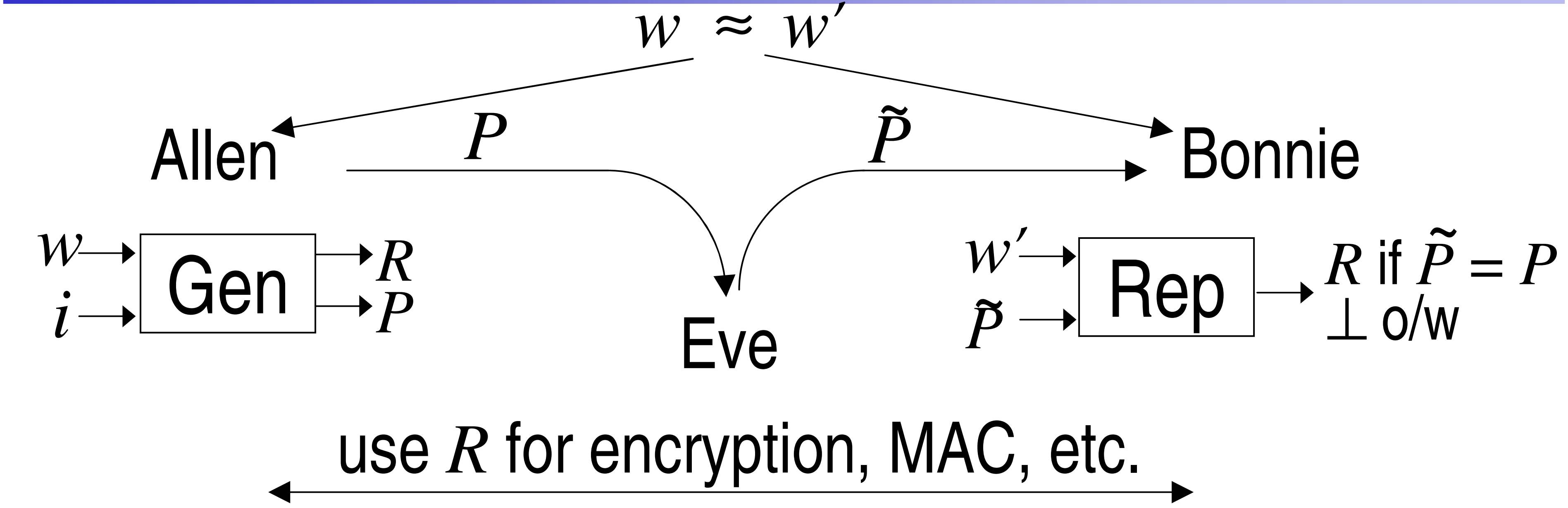
Fuzzy Extractor [Dodis, Ostrovsky, R., Smith]

- Robustness: as long as $w' \approx w$, if Eve(P) produces $\tilde{P} \neq P$



(with 1–negligible probability over w & coins of Rep, Eve)

setting 1: info-theoretic key agreement



$w \approx w'$

Allen

P

$\tilde{P}$

Bonnie

$w \rightarrow$ Gen $\rightarrow R$

$i \rightarrow$ Gen $\rightarrow P$

Eve

$w' \rightarrow$ Rep

$\tilde{P} \rightarrow$ Rep

Rep $\rightarrow$ R if $\tilde{P} = P$

$\perp$ o/w

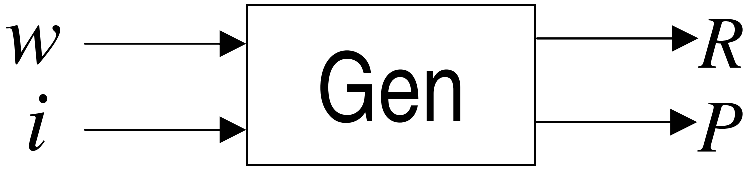
use R for encryption, MAC, etc.

Previously considered:

- If $w = w'$, or if w, w' and Eve's info come from repeated i.i.d. [Maurer, Renner, Wolf in several papers]
- Using random oracles [Boyen, Dodis, Katz, Ostrovsky, Smith]
- Interactive (more than one message): [MR,W,RW – limits on errors]
  [BDKOS – computational security, using PAK]

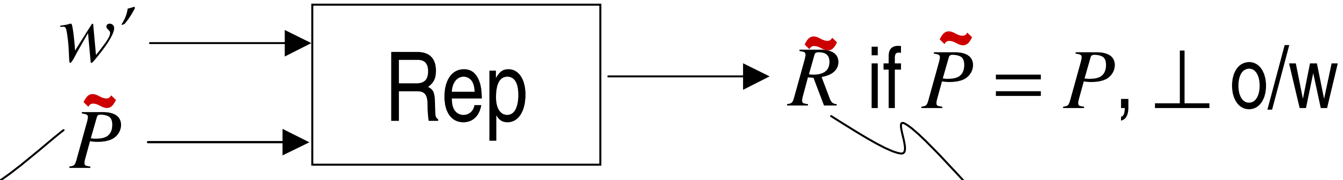
setting 2: noisy secret keys

- User has: noisy key w (e.g., biometric)



use to encrypt disk, derive (SK, PK), sign messages, ...

- Next time: needs same R (to decrypt disk, …)



$\tilde{R}$ if $\tilde{P} = P$, $\perp$ o/w

no trusted storage

various bad effects, depending on use of R

- Same problem as before, but noninteractivity essential!

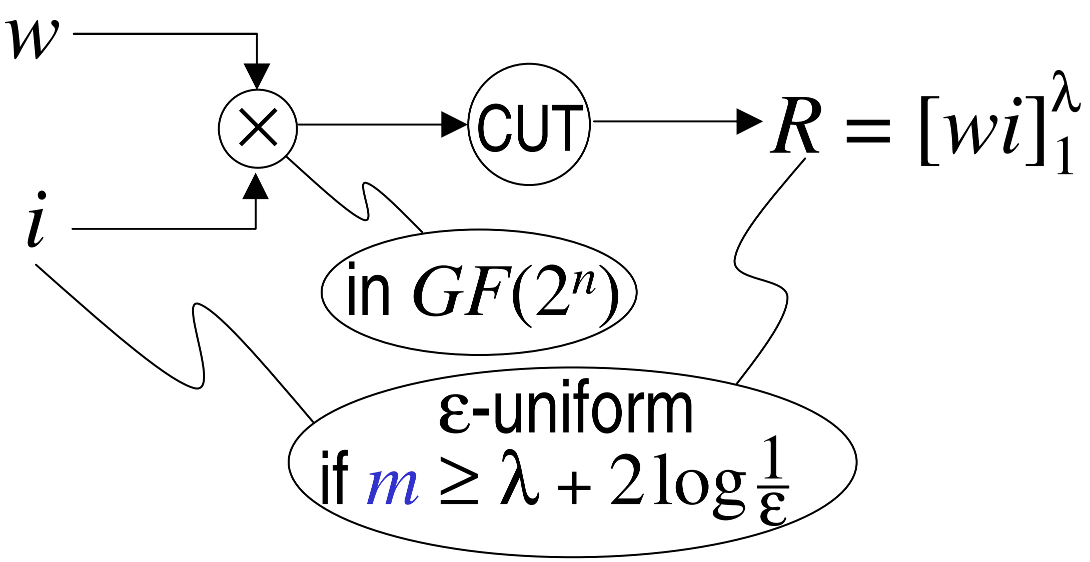
building extractors

Universal Hashing [Carter-Wegman]

$\Rightarrow$ Extractors [Bennet-Brassard-Robert, Impagliazzo-Levin-Luby]

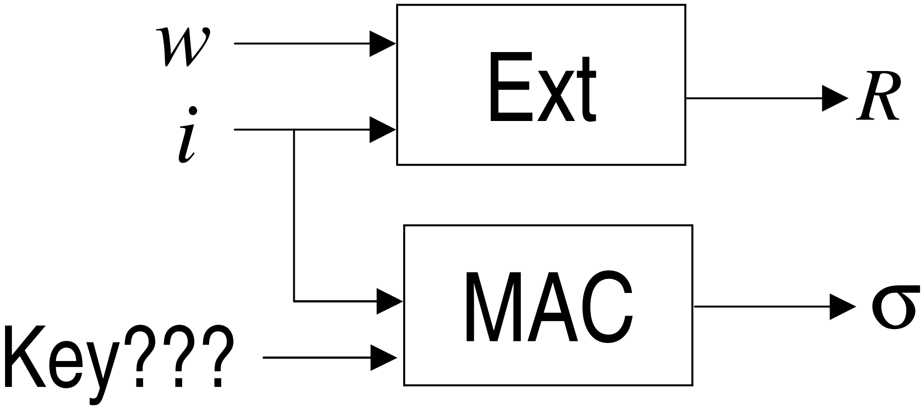
Simple Extractor: multiply-and-truncate

Let $|w| = n$, $H_\infty(w) = m$ Choose uniform i of length n



building extractors

Idea 0:



$P = (i, \sigma)$

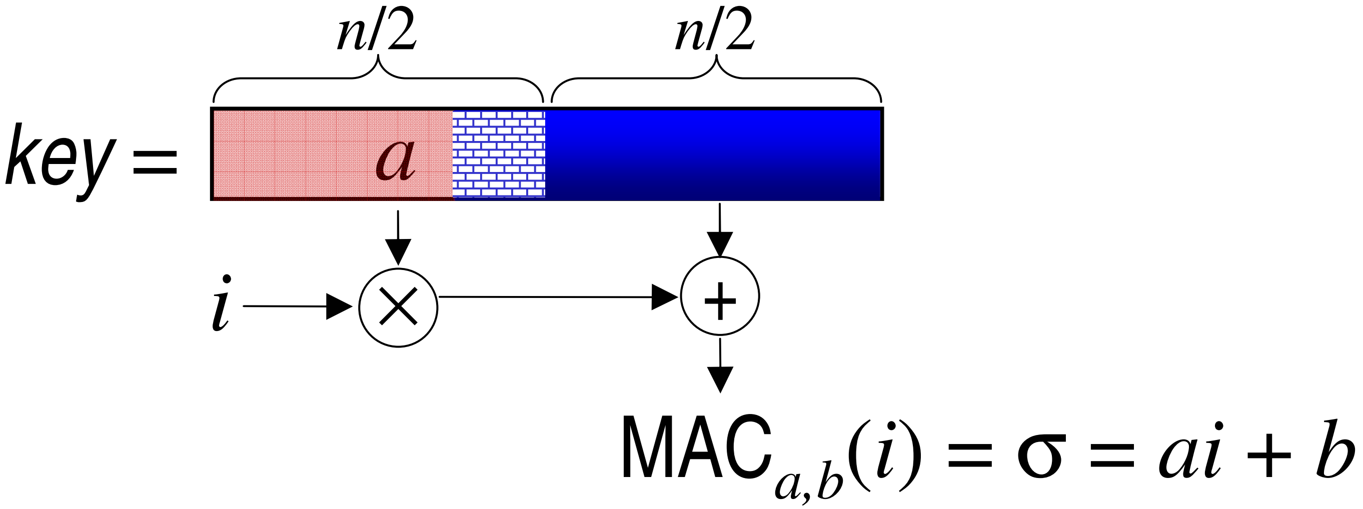
R? But if i changes ⇒ R changes

w! [Maurer-Wolf]

But w is not uniform ⇒

need MACs secure even with nonuniform keys

MACs with nonuniform keys



$\mathrm{MAC}_{a,b}(i) = \sigma = ai + b$

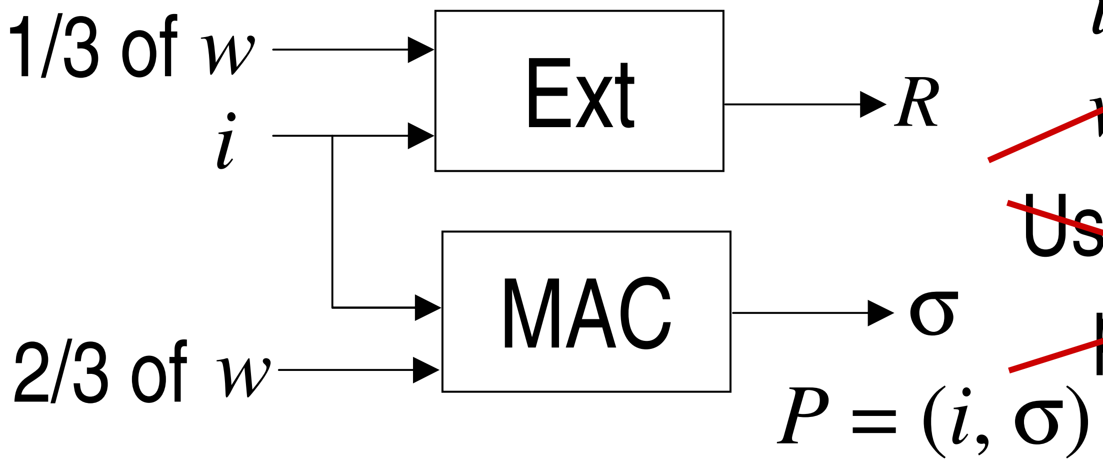
Let $|a,b| = n$, $H_\infty(a,b) = m$

Security: $m - n/2$

Let “entropy gap” $n - m = g$. Security: $n/2 - g$

building robust extractors

[Maurer-Wolf]:

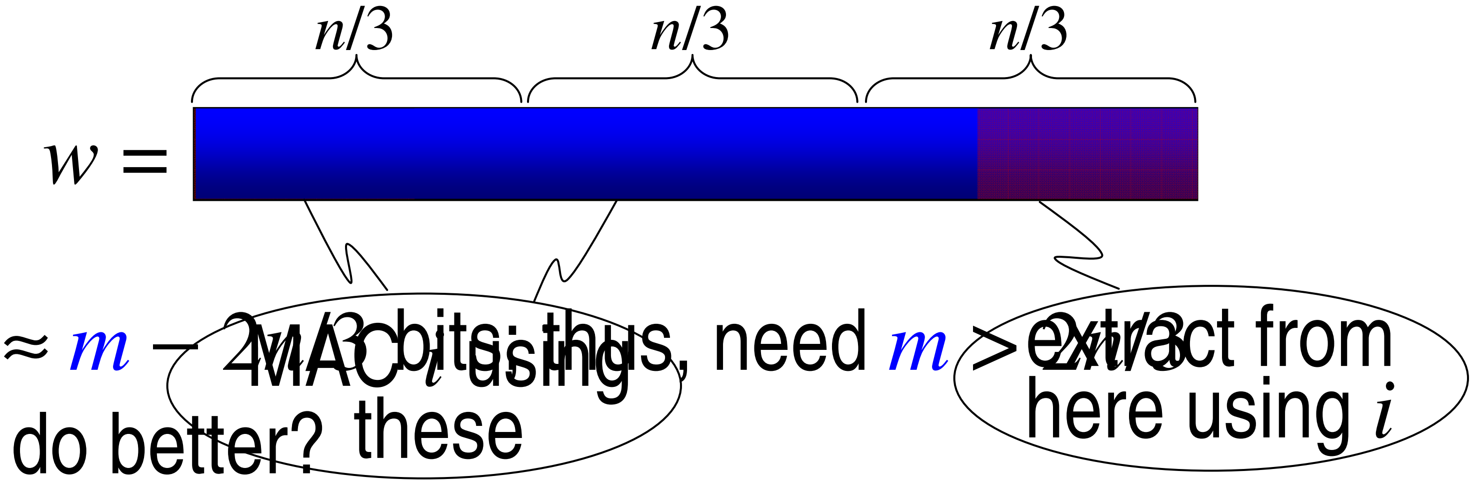


~~Circularity!~~

~~i extracts from w~~

~~w authenticates i~~

~~Use independent parts of w~~



Extract ≈ $m - 2n/3$ bits; thus, need $m > 2n/3$

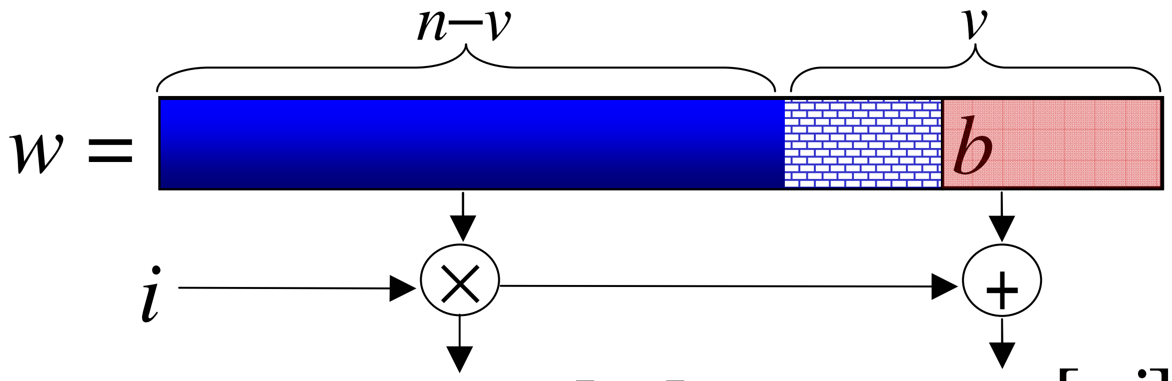
Can we do better?

Our idea: use circularity to our advantage!

building robust fuzzy extractors

Notation: $|w| = n$, $H_\infty(w) = m$, "entropy gap" $n - m = g$

Our construction: $w =$



$R = [ai]_{v+1}^{n-v}$ $\quad\sigma = [ai]_1^v + b$

Analysis:

jointly ε-uniform if $v > g + 2\log\frac{1}{\varepsilon}$

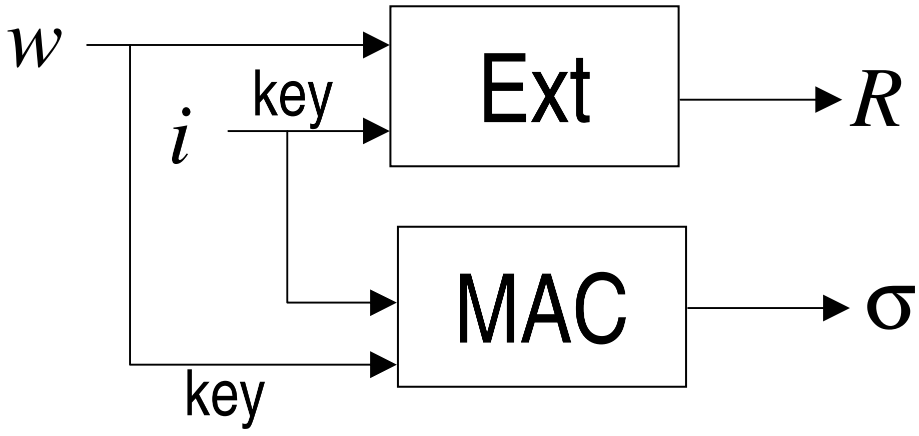
δ-secure if $v > g + \log\frac{1}{\delta}$

- Extraction: $(R, \sigma) = ai + b$ is a universal hash family (few collisions) (i is the key, $w = (a, b)$ is the input)
- Robustness: $\sigma = [ai]_1^v + b$ is strongly universal (2-wise indep.) ($w = (a, b)$ is the key, i is the input)

Extract $n - 2v \approx n - 2g = 2(m - n/2)$ bits (vs. $m - 2n/3$)

Note: $m > n/2$ is necessary [Dodis-Spencer]

building robust extractors ?



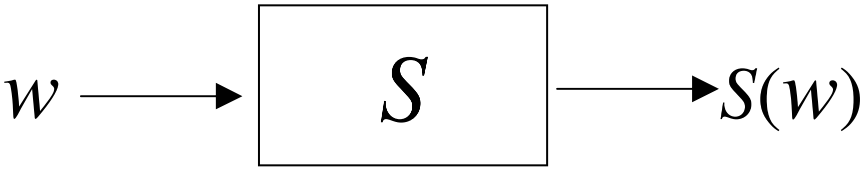
$$R = [ai]_{v+1}^{n-v}$$

$$\sigma = [ai]_1^v + b$$

$$P = (i, \sigma)$$

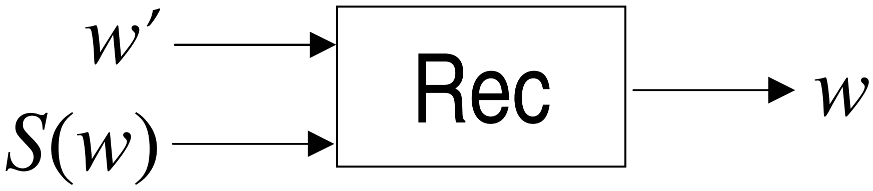
tool: secure sketch [DORS]

- Compute k-bit sketch $S(w)$



w → S → $S(w)$

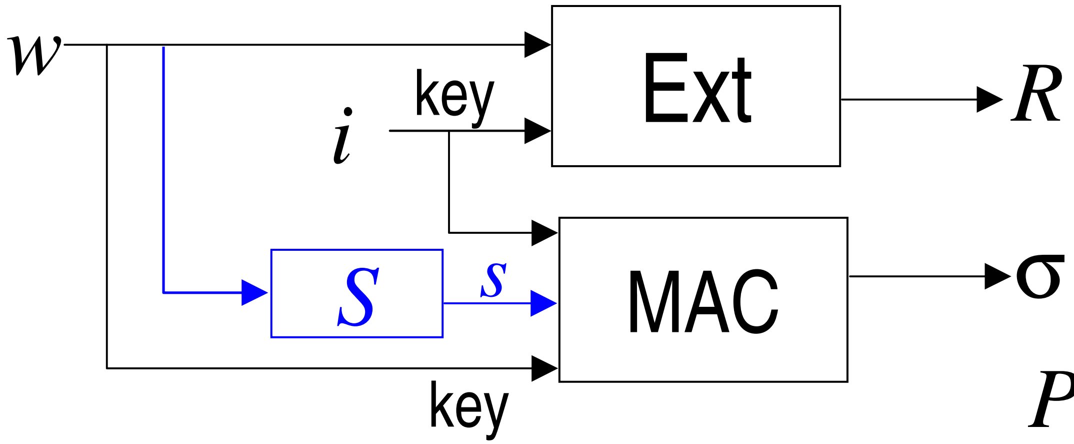
- Recover w from $S(w)$ and $w' \approx w$



w', $S(w)$ → Rec → w

- For Hamming metric, $S(w)$ can be a linear function (simply $\mathrm{syndrome}(w)$ in an $[n, n\text{–}k, 2t\text{+}1]_2$ code)

building robust fuzzy extractors



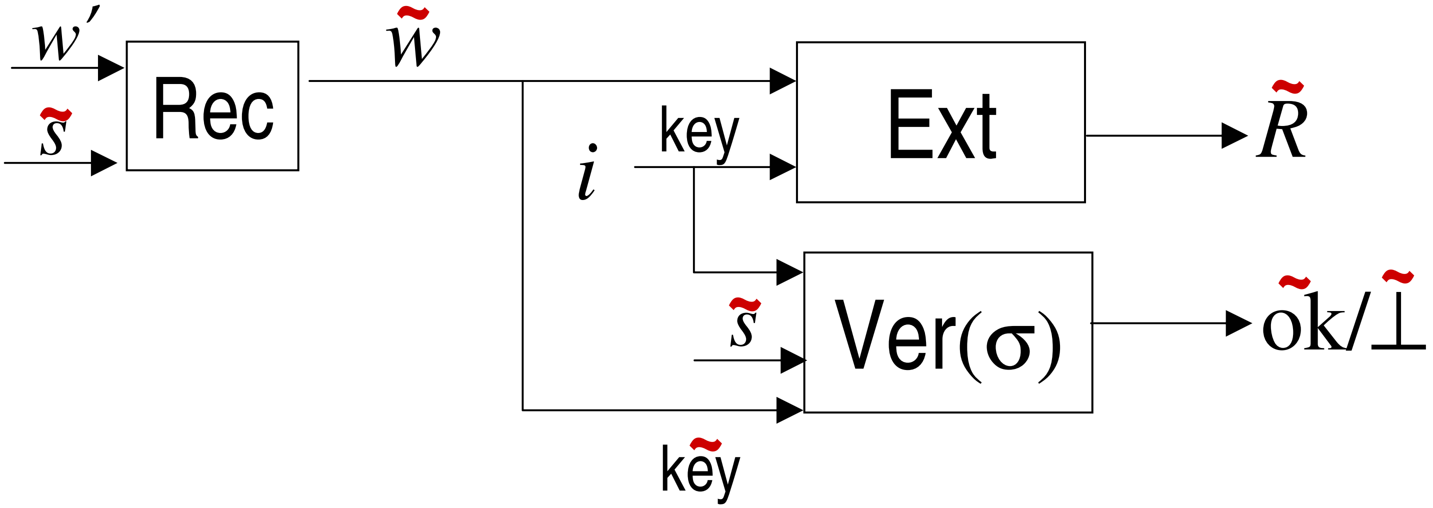
$\sigma = \text{MAC}_w(i, s)$

$P = (i, s, \sigma)$

How to MAC long messages? $\sigma = [a^2 s + ai]_1^v + b$

(recall $w = a|b$)

How to Rep



oops…

the MAC problem

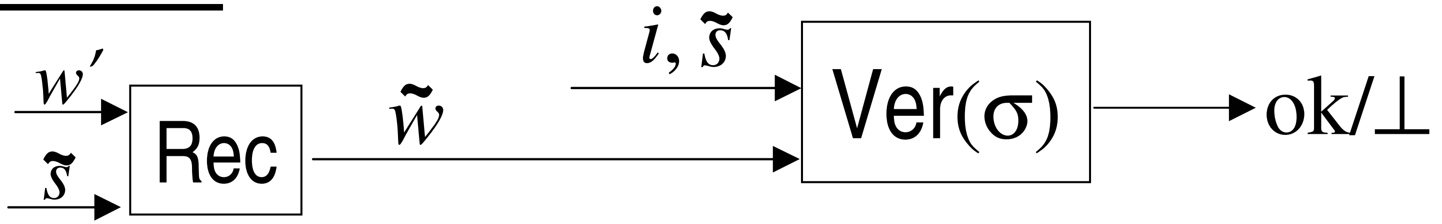
<u>Authentication:</u>

$\sigma = \text{MAC}_w(i, s) = [a^2 s + ai]_1^v + b$

(recall $w = a|b$)

Hard to forge for any fixed Δw

<u>Verification:</u>

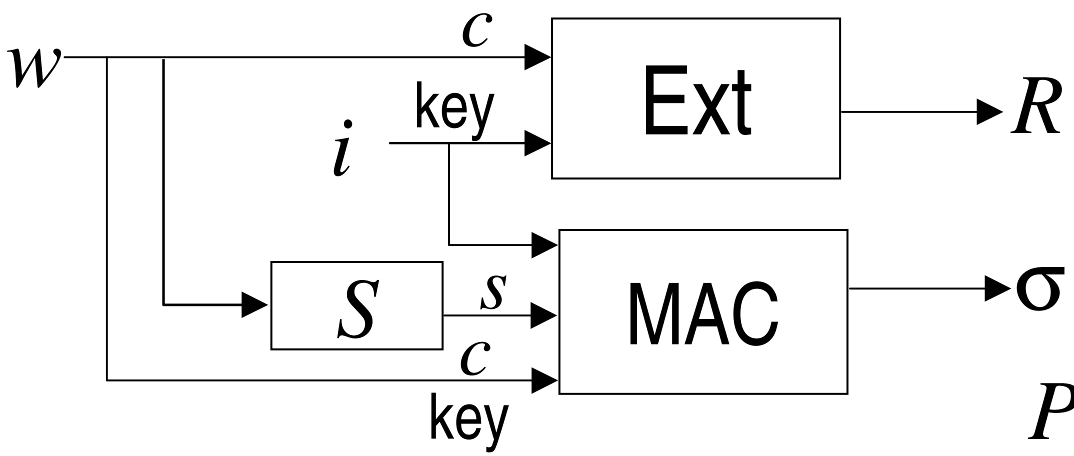


Problem: circularity (MAC key depends on s, which is being authenticated by the MAC)

Observe: knowing ($w' - w$ and $\tilde{s} - s$) $\Rightarrow \tilde{w} - w = \Delta w$

Need: $\forall\ \Delta w$, given $\text{MAC}_w(i, s)$, hard to forge $\text{MAC}_{w+\Delta w}(\tilde{i}, \tilde{s})$

building robust fuzzy extractors



$\sigma = \mathrm{MAC}_w(i, s)$

$P = (i, s, \sigma)$

Recall: without errors, extract $n-2g = m - g$

Problem: s reveals k bits about $w \Rightarrow$

m decreases, g increases $\Rightarrow$

lose $2k$

Can’t avoid decreasing m, but can avoid increasing g

$s = S(w)$ is linear. Let $c = S^{\perp}(w)$.

$|c|=|w|-k$, but c has same entropy as $w|s$. Use c instead of w.

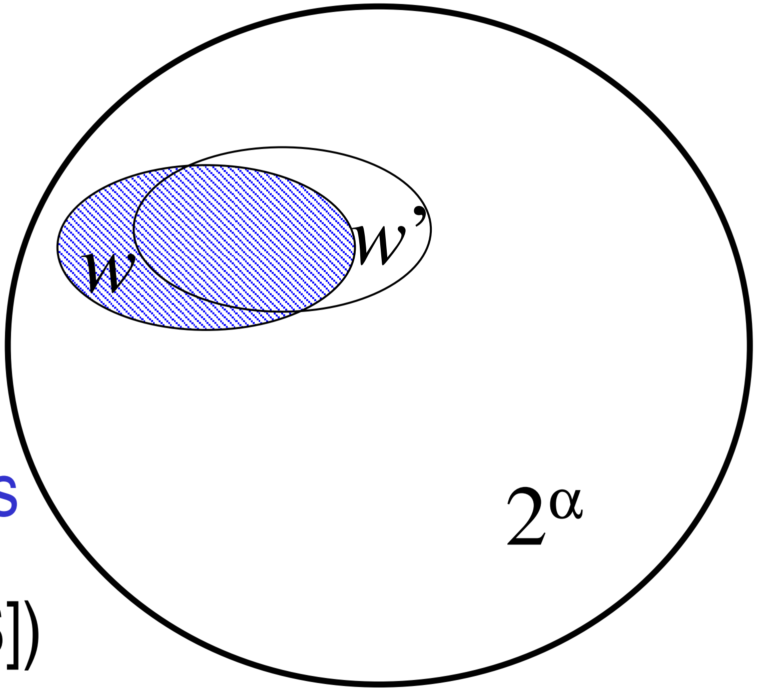
the bottom line

Result for with t Hamming errors:

given $[n, n{-}k, 2t{+}1]_2$ linear code,

extract $2(m - n/2){-}k{-}2b$ bits

$(b = \log \mathrm{Vol}(\mathrm{Ball}(t)) < t \log n)$



Result for with t set difference errors:

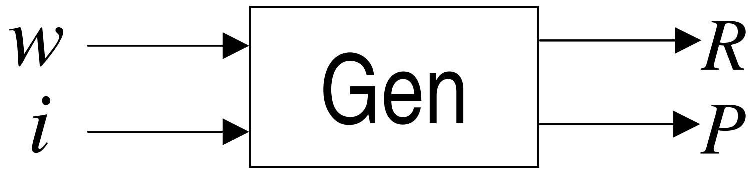
(w is a subset of a universe of size 2^α)

extract $2(m - n/2){-}3t\alpha$ bits

(uses BCH-based PinSketch of [DORS])

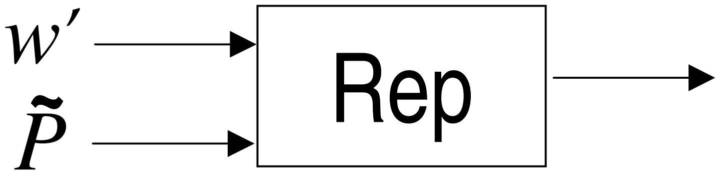
single user setting, revisited

- User has: noisy key w (e.g., biometric)



use to encrypt disk, derive (SK, PK), sign messages, ...

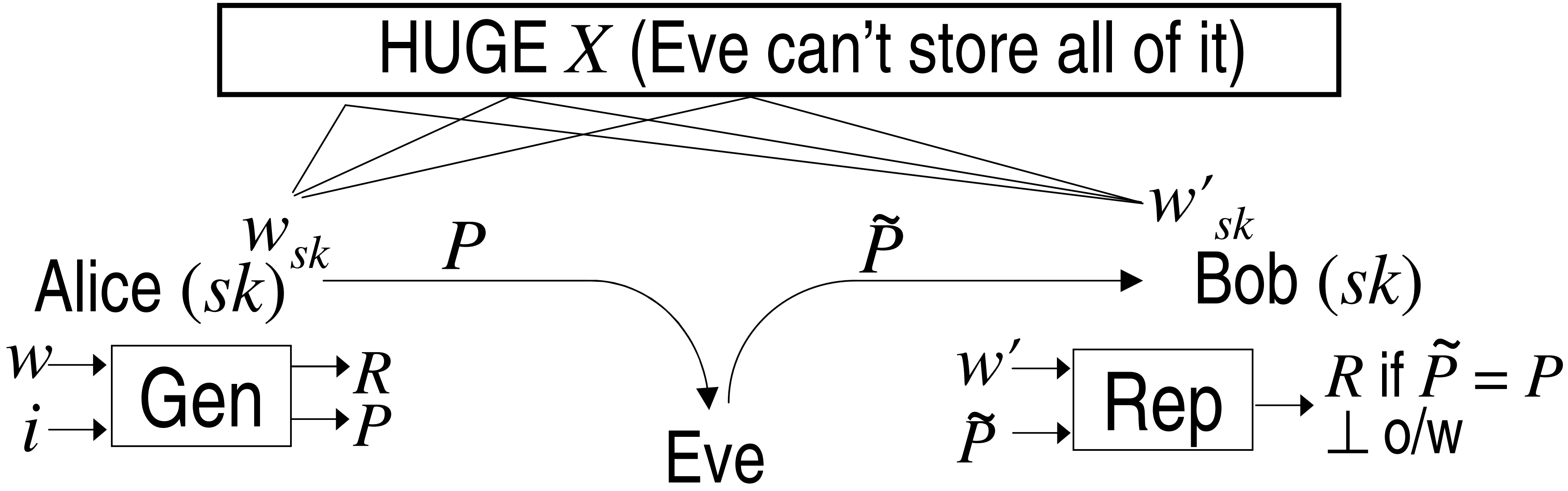
- Next time: needs same R (to decrypt disk, …)



$\tilde{R}$ if $\tilde{P} = P$, $\perp$ o/w

- But Eve sees effects of R (e.g., disk encrypted with R) before coming up with $\tilde{P}$
- New, stronger robustness notion: allow Eve to see (P, R)
- “post-application” (vs. “pre-application”) robustness
- Our constructions work, but only extract 1/3 the bits

application to bounded storage model



HUGE X (Eve can't store all of it)

w_{sk}

w'_{sk}

Alice (sk)

P

$\tilde{P}$

Bob (sk)

$w \rightarrow$ Gen $\rightarrow R$

$i \rightarrow$ Gen $\rightarrow P$

Eve

$w' \rightarrow$ Rep

$\tilde{P} \rightarrow$ Rep

$\rightarrow R$ if $\tilde{P} = P$

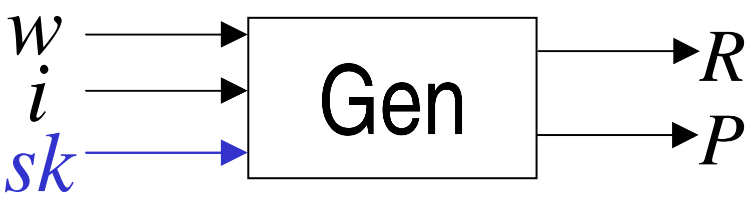
$\perp$ o/w

doesn't know sk, hence w has entropy

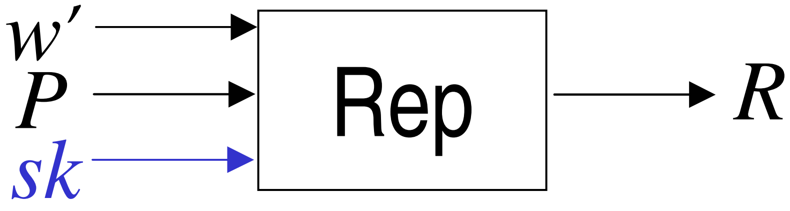
- Lots of prior work [Maurer,Cachin,Dziembowski,Aumann,Ding,Rabin,Lu,Vadhan,…]
- Noisy case: [Ding, Dodis-Smith]—stateful A&B, or passive Eve
- Use robust fuzzy extractors: stateless A&B, active Eve
- But parameters not great—better solution?
- Yes: in this special case, A&B have sk

need: keyed robust fuzzy extractor

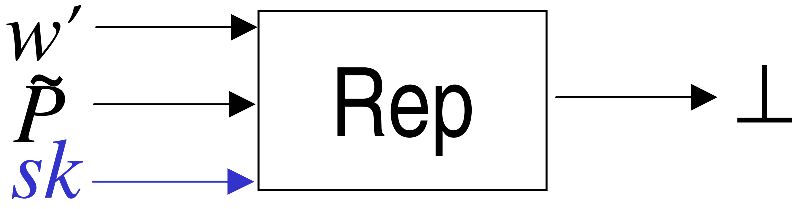
- Extraction: generate uniform R from w (+ seed i)



- Fuzziness: reproduce R from P and $w' \approx w$

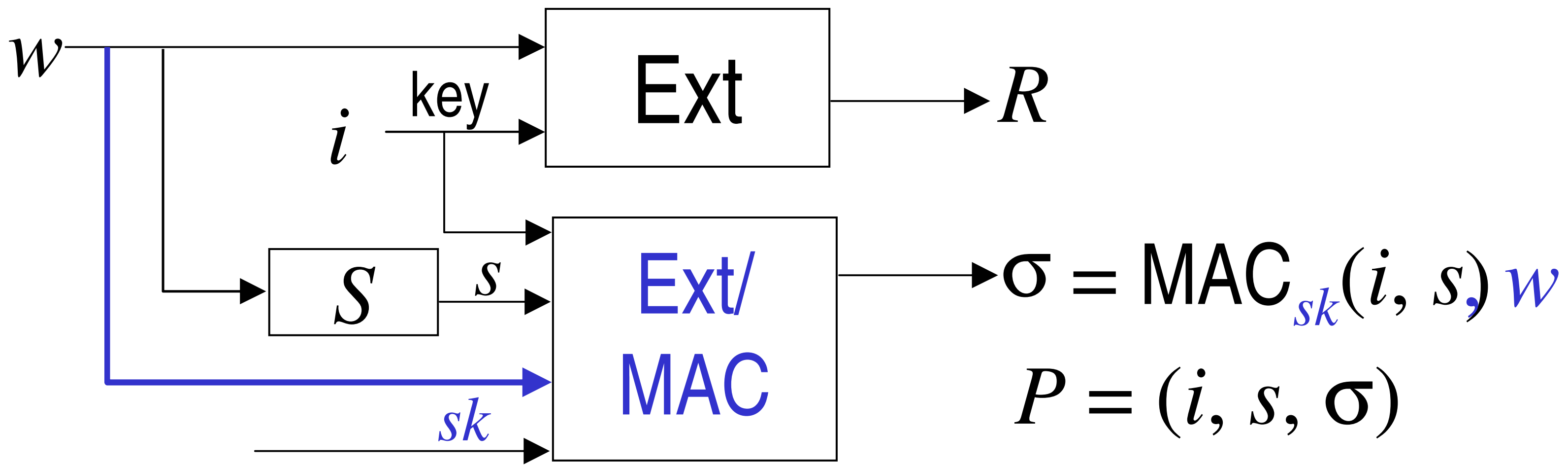


- Robustness: as long as $w' \approx w$, if Eve(P) produces $\tilde{P} \neq P$



- Crucial: sk must be reusable

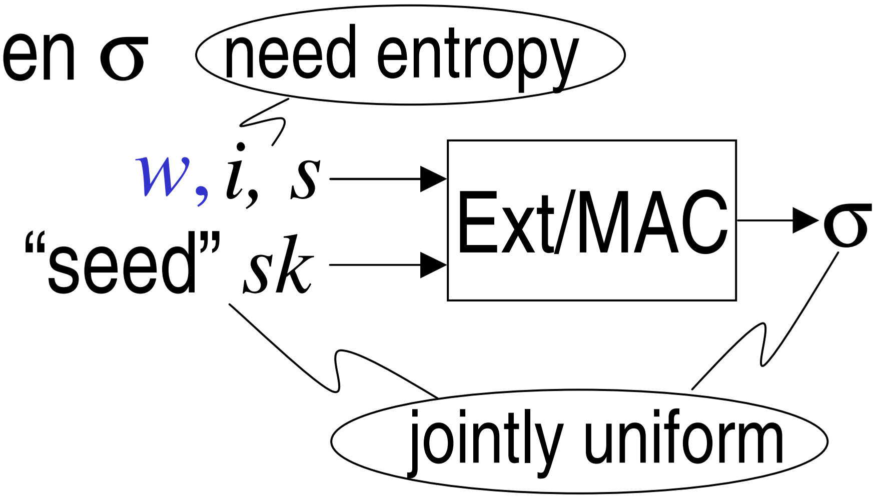
building keyed robust fuzzy extractors



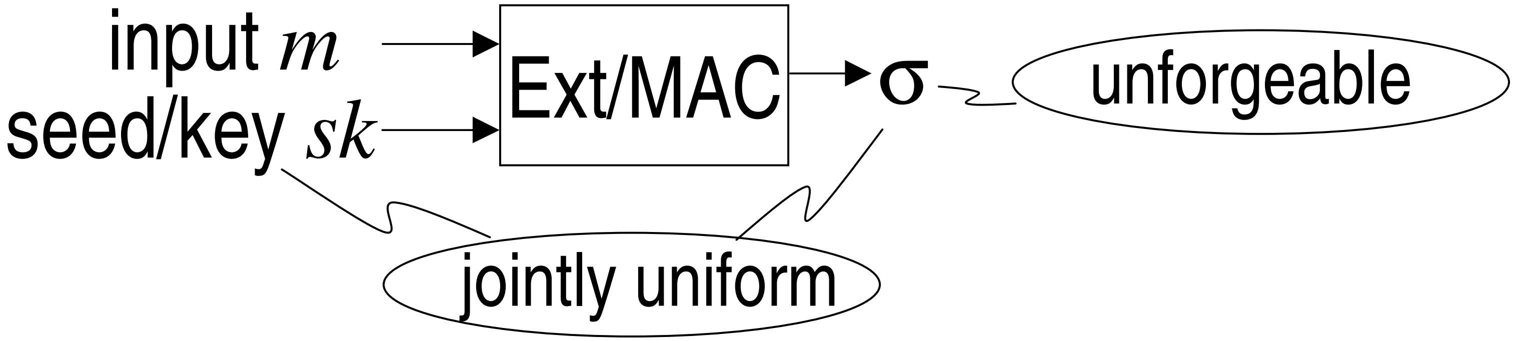
$\sigma = \text{MAC}_{sk}(i, s,) w$

$P = (i, s, \sigma)$

- Problem: sk is not reusable
- Need: sk is random even given σ
- Idea: use a MAC that is also an extractor



building extractor MACs



input m

seed/key sk

Ext/MAC

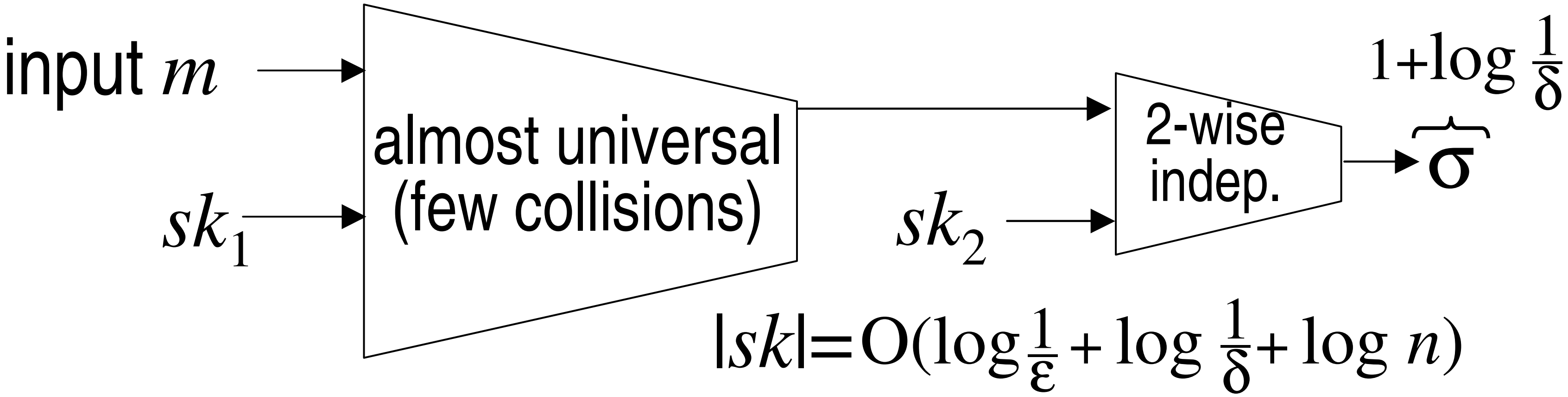
σ

unforgeable

jointly uniform

(note: unlike extractors, want short outputs σ)

- Idea 1: use pairwise-independent hashing
  - Both good MAC and good extractor, but long sk
- Idea 2 (modifying Srinivasan-Zuckerman):



input m

sk_1

almost universal (few collisions)

sk_2

2-wise indep.

$\overbrace{\sigma}^{1+\log\frac{1}{\delta}}$

$$|sk| = O(\log\tfrac{1}{\varepsilon} + \log\tfrac{1}{\delta} + \log n)$$

conclusions

- Keyless robust fuzzy extractors
  - errorless case: previously $|R| = m - 2n/3$, we $|R| = 2(m - n/2)$ ($m > n/2$ is minimum possible)
  - case with errors: previously only with random oracles, we solve Hamming distance and set difference without r.o.
  - new definition: post-application robustness, constructions that satisfy it

- Keyed case
  - Useful new notion: extractor-MAC
  - Application to stateless, active-attack-resistant, BSM with errors (previously stateful or passive attack only)

Thank you!