OCB Mode

Phillip Rogaway
Department of Computer Science
UC Davis + Chiang Mai Univ
rogaway@cs.ucdavis.edu
http://www.cs.ucdavis.edu/~rogaway
+66 1 530 7620   +1 530 753 0987

Mihir Bellare
UCSD
mihir@cs.ucsd.edu

John Black
UNR
jrb@cs.unr.edu

Ted Krovetz
~~Digital Fountain~~ Looking—contact Ted!
tdk@acm.org

NIST Modes of Operation Workshop 2 – Aug 24, 2001 - Santa Barbara, California

Two Cryptographic Goals

Privacy — What the Adversary sees tells her nothing of significance about the underlying message M that the Sender sent

Authenticity — The Receiver is sure that the string he receives was sent (in exactly this form) by the Sender

---

Authenticated Encryption — Achieves both privacy and authenticity

Why Authenticated Encryption?

- **Efficiency**
  By merging privacy and authenticity one can achieve efficiency difficult to achieve if handling them separately

- **Easier-to-correctly-use abstraction**
  By delivering strong security properties one may minimize encryption-scheme misuse

What does Encryption **Do**?

Strong

Idealized encryption — Security community's favored view

OCB — Authenticated encryption: IND under CPA + auth of ciphertexts

IND under CCA = NM under CCA

[Bellare, Namprenre], [Katz, Yung]

CTR, CBC$ — IND under CPA — Cryptographic community's favored view: sym encryption is for IND-CPA (and nothing more)

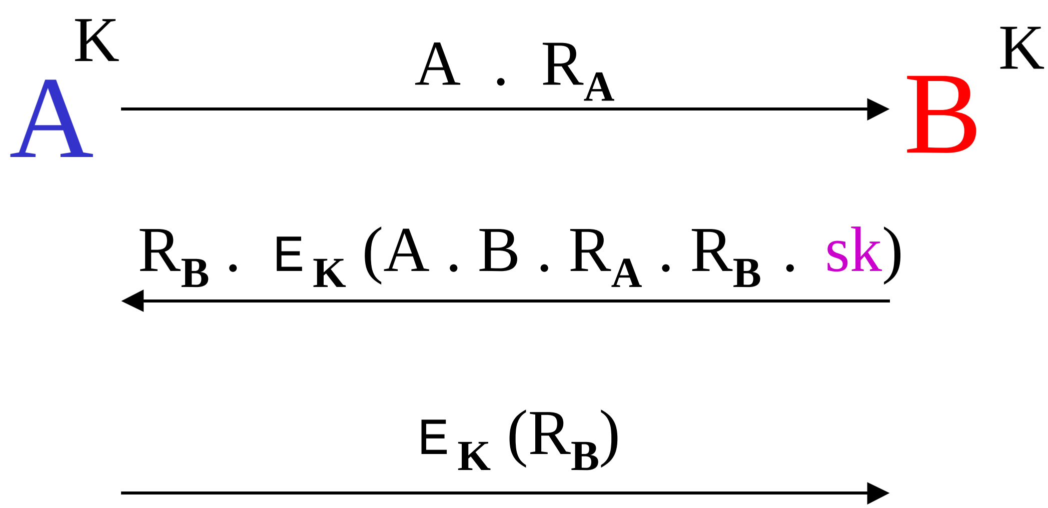
[Bellare, Desai, Jokipii, Rogaway]

ECB — No meaningful notion of privacy

Weak

Right or Wrong?

It depends on what definition E satisfies



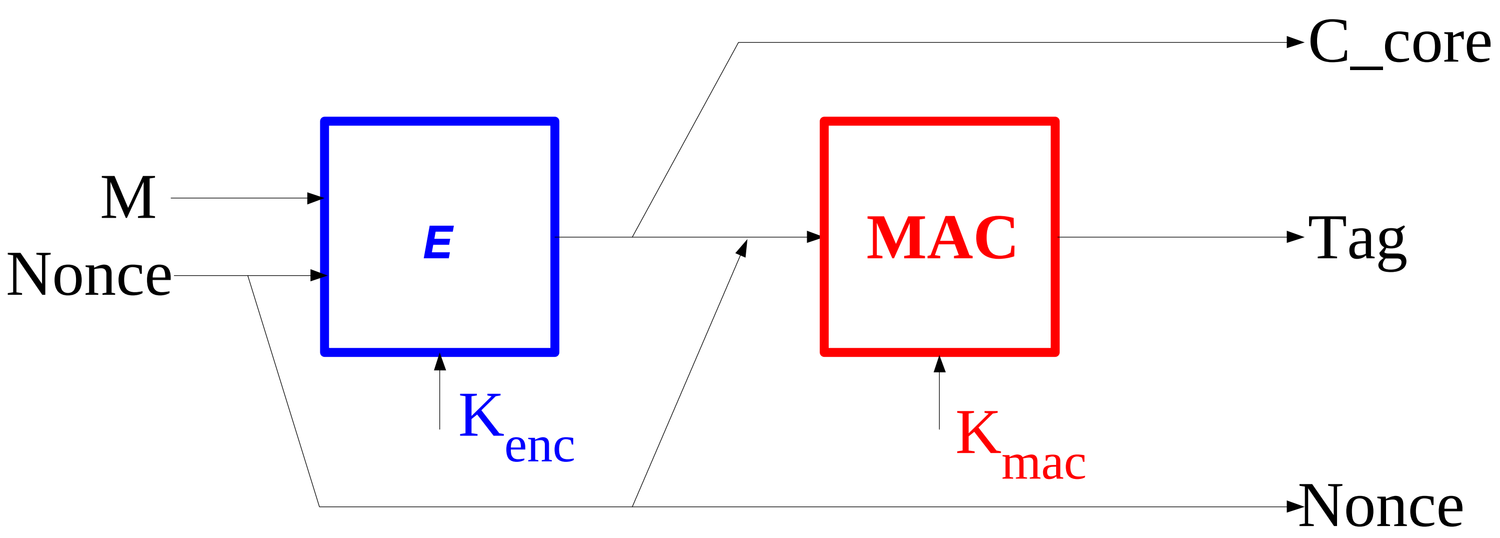
Generic Composition

Folklore approach. See [Bellare, Namprempre] and [Krawczyk] for analysis.

Traditional approach to authenticated encryption

Glue together an encryption scheme (E) and a Message Authentication Code (MAC)

Preferred way to do generic composition:



Generic Composition

+ Versatile, clean architecture
+ Reduces design work
+ Quick rejection of forged messages if use optimized MAC (eg., UMAC)
+ Inherits the characteristics of the modes one builds from

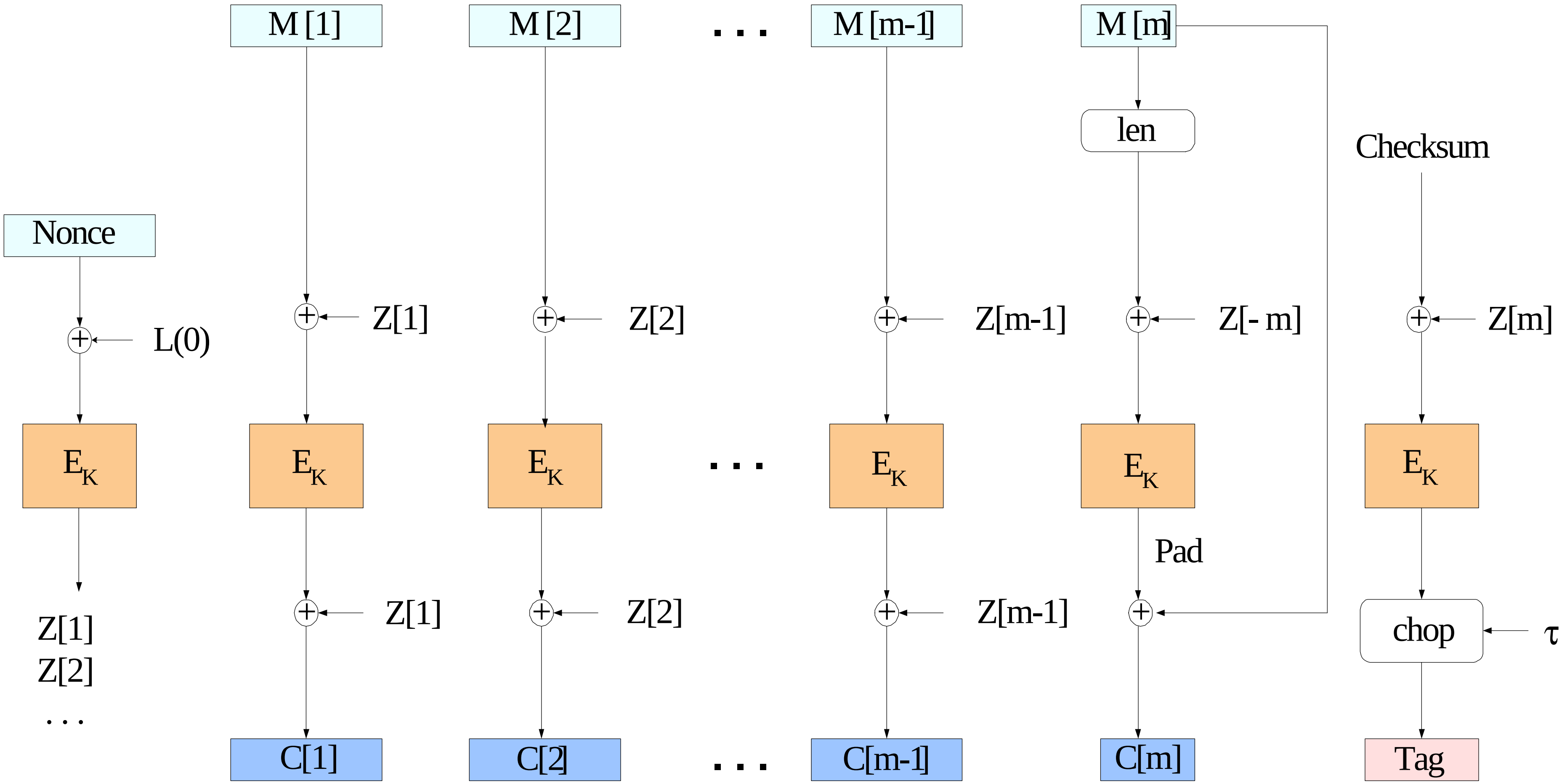
- Cost ≈ (cost to encrypt) + (cost to MAC)
  For CBC Enc + CBC MAC, cost ≈ 2 × (cost to CBC Enc)
- Often misused
- Two keys
- Inherits characteristics of the modes one builds from

Trying to do Better

- Numerous attempts to make privacy + authenticity cheaper
- One approach: stick with generic composition, but find cheaper privacy algorithm and cheaper authenticity algorithms
- Make authenticity an "incidental" adjunct to privacy within a conventional-looking mode
  - CBC-with-various-checksums (wrong)
  - PCBC in Kerberos (wrong)
  - PCBC of [Gligor, Donescu 99] (wrong)
  - [Jutla - Aug 00] First correct solution
- Jutla described two modes, IACBC and IAPM
- A lovely start, but many improvements possible
- OCB: inspired by IAPM, but many new characteristics

What is OCB?

- Authenticated-encryption scheme
- Uses any block cipher (eg. AES)
- Computational cost $\approx$ cost of CBC
- OCB-AES good in SW or HW
- Lots of nice characteristics designed in:
  - Uses $\lceil |M| / n \rceil + 2$ block-cipher calls
  - Uses any nonce (needn't be unpredictable)
  - Works on messages of any length
  - Creates minimum-length ciphertext
  - Uses a single block-cipher key, each block-cipher keyed with it
  - Quick key setup – suitable for single-message sessions
  - Essentially endian-neutral
  - Fully parallelizable
  - No n-bit additions
- Provably secure: if you break OCB-AES you've broken AES
- In IEEE 802.11 draft. Paper to appear at ACM CCS '01



Checksum = M[1] $\oplus$ M[2] $\oplus$ $\cdots$ $\oplus$ M[m-1] $\oplus$ C[m]0* $\oplus$ Pad

Z[i] = Z[i-1] $\oplus$ L(**ntz**(i))
L(0) = E_K(**0**) and each L(i) obtained from L(i-1) by a shift and conditional xor

Definition of OCB[E, t]

algorithm OCB-Encrypt $_K$ (Nonce, M)
$L(0) = E_K(\mathbf{0})$
$L(-1) = \mathrm{lsb}(L(0))?\ (L(0) >> 1) \oplus \mathrm{Const43}\ :\ (L(0) >> 1)$
for i = 1, 2, … **do** $L(i) = \mathrm{msb}(L(i-1))?\ (L(i-1) << 1) \oplus \mathrm{Const87} : (L(i-1) << 1)$
Partition M into M[1] ··· M[m]   // each n bits, except M[m] may be shorter
$\mathrm{Offset} = E_K(\mathrm{Nonce} \oplus L(0))$
for i=1 **to** m-1 **do**
  $\mathrm{Offset} = \mathrm{Offset} \oplus L(\mathrm{ntz}(i))$
  $C[i] = E_K(M[i] \oplus \mathrm{Offset}) \oplus \mathrm{Offset}$
$\mathrm{Offset} = \mathrm{Offset} \oplus L(\mathrm{ntz}(m))$
$\mathrm{Pad} = E_K(\mathrm{len}(M[m]) \oplus \mathrm{Offset} \oplus L(-1))$
$C[m] = M[m] \oplus (\text{first } |M[m]| \text{ bits of Pad})$
$\mathrm{Checksum} = M[1] \oplus \cdots \oplus M[m-1] \oplus C[m]0^* \oplus \mathrm{Pad}$
$\mathrm{Tag} = \text{first } \tau \text{ bits of } E_K(\mathrm{Checksum} \oplus \mathrm{Offset})$
return C[1] ··· C[m] || Tag

Assembly Speed

Data from **Helger Lipmaa** www.tcs.hut.fi/~helger helger@tcs.hut.fi

// Best Pentium AES code known. Helger's code is for sale, btw.

OCB-AES	16.9 cpb	(271 cycles)	6.5 % slower
CBC-AES	15.9 cpb	(255 cycles)	
ECB-AES	14.9 cpb	(239 cycles)	
CBCMAC-AES	15.5 cpb	(248 cycles)	

The above data is for 1 Kbyte messages. Code is pure Pentium 3 assembly. The block cipher is AES128. Overhead so small that AES with a C-code CBC wrapper is slightly more expensive than AES with an assembly OCB wrapper.

C Speed

Data from **Ted Krovetz** . Compiler is MS VC++. Uses rijndael-alg-fst.c ref code.

OCB-AES	28.1 cpb	(449 cycles)	4.9 % slower
CBCMAC-AES	26.8 cpb	(428 cycles)	

Why I like OCB

- **Ease-of-correct-use**. Reasons: all-in-one approach; any type of nonce; parameterization limited to block cipher and tag length
- **Aggressively optimized**: ≈ optimal in many dimensions: key length, ciphertext length, key setup time, encryption time, decryption time, available parallelism; SW characteristics; HW characteristics; …
- **Simple but highly non-obvious**
- Ideal setting for **practice-oriented provable security**

What is Provable Security?

- Provable security begins with [Goldwasser, Micali 82]
- Despite the name, one doesn't really *prove* security
- Instead, one gives *reductions*: theorems of the form

  If a certain primitive is secure

  then the scheme based on it is secure

  Eg:

  If AES is a secure block cipher

  then OCB-AES is a secure authenticated-encryption scheme

  Equivalently:

  If some adversary **A** does a good job at breaking OCB-AES

  then some comparably efficient **B** does a good job to break AES
- Actual theorems quantitative: they measure how much security is "lost" across the reduction.

(

The Power of **Definitions**

- Let's you carry on an intelligent conversation
- Let's you investigate the "space" of goals and how they are related
- Often let's you easily see when protocols are **wrong**
- Let's you prove when things are right, to the extent that we know how to do this.

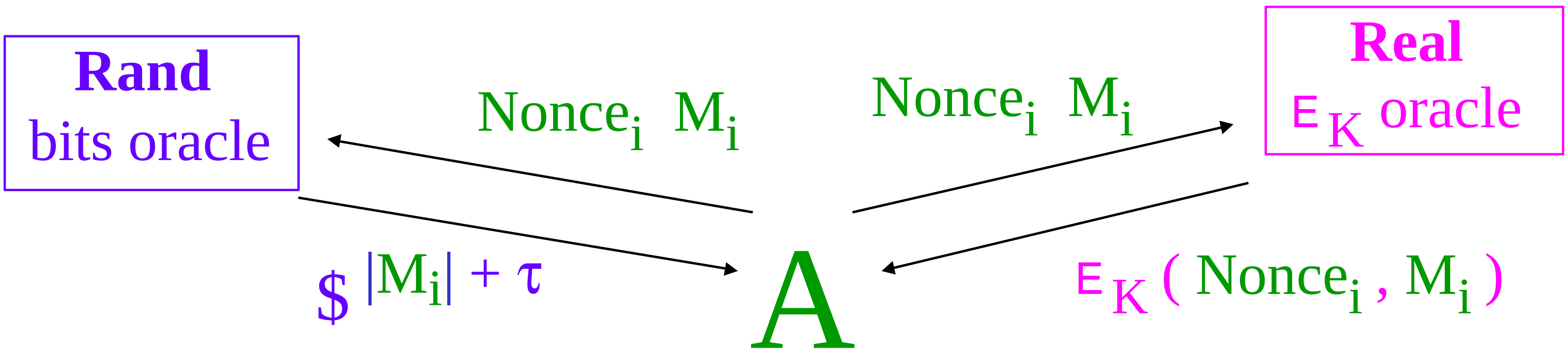
It took about an hour to break the NSA's "Dual Counter Mode". What did I have that the NSA authors didn't? Just an understanding of a good **definition** for the goal.

)

Privacy

Indistinguishability from Random Bits

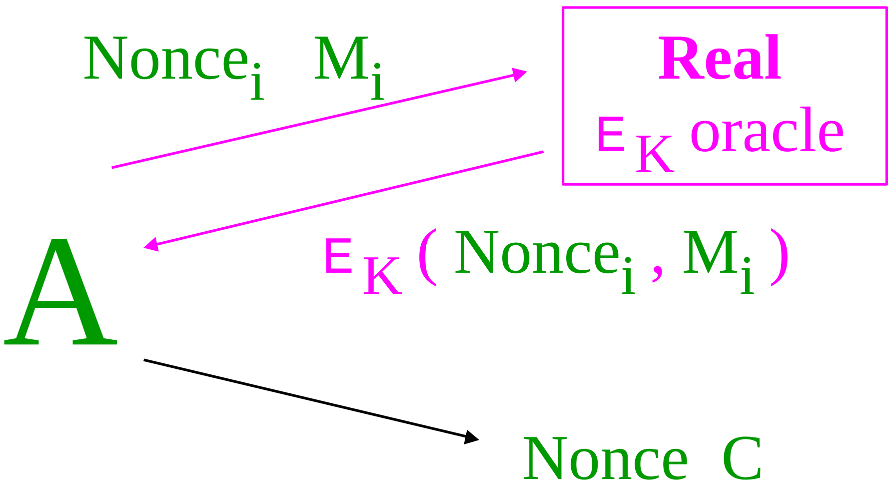
[Goldwasser, Micali]
[Bellare, Desai, Jokipii, Rogaway]



$$\mathbf{Adv}^{\mathbf{priv}}(\mathbf{A}) = \Pr[\mathbf{A}^{\mathbf{Real}} = 1] - \Pr[\mathbf{A}^{\mathbf{Rand}} = 1]$$

Authenticity:
Authenticty of Ciphertexts

[Bellare, Rogaway]
[Katz, Yung]
this paper



A forges if she outputs forgery attempt Nonce C s.t.

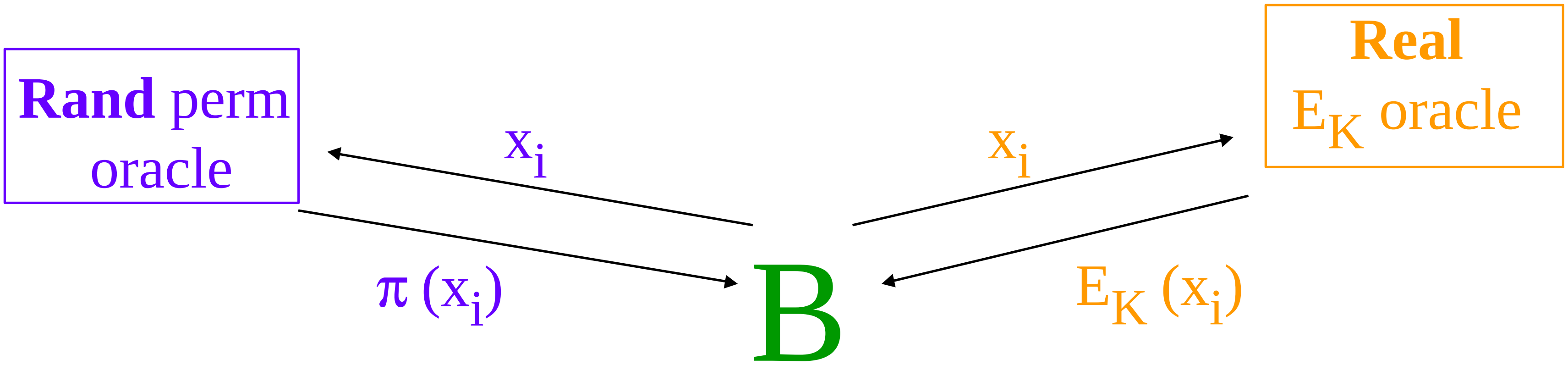
- C is **valid** (it decrypts to a message, not to **invalid**)
- there was no E$_K$ query Nonce M$_i$ that returned C

Adv$^{\textbf{auth}}$ (**A**) = Pr[**A** forges]

Block-Cipher Security

PRP and Strong PRP

[Goldreich, Goldwasser, Micali]
[Luby, Rackoff]
[Bellare, Kilian, Rogaway]



$$\textbf{Adv}^{\textbf{prp}}\,(\textbf{B}) = \Pr[\textbf{B}^{E_K} = 1] - \Pr[\textbf{B}^{\pi} = 1]$$

$$\textbf{Adv}^{\textbf{sprp}}\,(\textbf{B}) = \Pr[\textbf{B}^{E_K\ E_K^{-1}} = 1] - \Pr[\textbf{B}^{\pi\ \pi^{-1}} = 1]$$

OCB Theorems

Privacy theorem:

Suppose ∃ an adversary **A**	Then ∃ an adversary **B**
that breaks **OCB-E** with:	that breaks block cipher **E** with:
time = t	*time* ≈ t
total-num-of-blocks = σ	*num-of-queries* ≈ σ
adv = $\mathbf{Adv}^{\mathbf{priv}}$ **(A)**	$\mathbf{Adv}^{\mathbf{prp}}$ **(B)** ≈ $\mathbf{Adv}^{\mathbf{priv}}$**(A)** – $1.5\ \sigma^2 / 2^n$

Authenticity theorem:

Suppose ∃ an adversary **A**	Then ∃ an adversary **B**
that breaks **OCB-E** with:	that breaks block cipher **E** with:
time = t	*time* ≈ t
total-num-of-blocks = σ	*num-of-queries* ≈ σ
adv = $\mathbf{Adv}^{\mathbf{auth}}$ **(A)**	$\mathbf{Adv}^{\mathbf{sprp}}$ **(B)** ≈ $\mathbf{Adv}^{\mathbf{priv}}$**(A)** – $1.5\ \sigma^2 / 2^n$

What Provable Security Does, and Doesn't, Buy You

+ Strong evidence that scheme does what was intended
+ Best assurance cryptographers know how to deliver
+ Quantitative usage guidance

- An absolute guarantee
- Protection from issues not captured by our abstractions
- Protection from usage errors
- Protection from implementation errors

	Domain	| Ciphertext |	IV rqmt	Calls / msg	Calls / keysetup	Key length (#E-keys)	/ blk overhead	E circuit depth
IAPM **(lazy mod p)** [Jutla 00,01]	$(\{0,1\}^n)^+$	$\|M\| + \tau$	nonce (Jutla's presentation gave **rand** version)	$\|M\| /n + 2$	0	2k (2)	1 xor 2 add 1 addp	2
XECB-XOR [GD 01]	$\{0,1\}^*$	$\lceil \|M\| / n \rceil + n$	ctr	$\lceil \|M\| /n \rceil +1$	0	k+2n (1)	1 xor 3 add	1
OCB [R+ 00,01]	$\{0,1\}^*$	$\|M\| + \tau$	nonce	$\lceil \|M\| /n \rceil + 2$	1	k (1)	4 xor	3

Parallelizable Authenticated-Encryption Schemes

For More Information

- OCB web page → www.cs.ucdavis.edu/~rogaway
  Contains FAQ, papers, reference code, licensing info...
- Feel free to call or send email
- Upcoming talks: MIT (Oct 26), ACM CCS (Nov 5-8), Stanford (TBA)
- Or grab me now!

Anything Else ??