30 April 2025	· · · · · ·)ne-Way	Functions	· · · ·
· · · · · · · · · · · · · · · · · ·	· · · · · ·	· · · · · · · · · ·	· · · · · · · ·	· · · · · · · · · · · ·	· · ·
$\frac{P}{e}$	· · · · · ·	· · · · · · · · · ·	· · · · · · · ·	· · · · · · · · · · ·	
* Cryptography	from	Computational) Handne	<u>-</u> 35	· · · ·
* Announceme					· · · ·
× Oire - Wary	Funct	γ · · · · · · · · · · · · · · · · · · ·			· · · ·
· · · · · · · · · · · · · · · · · · ·		· · · · · · · · · ·	· · · · · · · ·	· · · · · · · · · · ·	· · · ·
· · · · · · · · · · · · · · · · · ·		· · · · · · · · · · ·	· · · · · · · ·	· · · · · · · · · · · ·	
· · · · · · · · · · · · · · · · · ·		· · · · · · · · · · ·	· · · · · · · ·	· · · · · · · · · · · ·	
· · · · · · · · · · · · · · · · · · ·					

Motivation:	Secure	Communication	· · · · · · · · · · · ·
		· · · · · · · · · · · · · · · · · · ·	
· · · · · · · · · · · · ·	· · · · · · ·	· · · · · · · · · · · · · · · · · · ·	· · · · · · · · · · · ·
· · · · · · · · · · · ·	· · · · · · ·	· · · · · · · · · · · · · · · · · · ·	· · · · · · · · · · ·
		\mathbb{B}	· · · · · · · · · · · ·
· · · · · · · · · · · · · ·	· · · · · · · ·	· · · · · · · · · · · · · · · · · · ·	
· · · · · · · · · · · · ·	· · · · · · ·	Eve Eve to the total second	· · · · · · · · · · · ·
· · · · · · · · · · · · ·	· · · · · · ·		· · · · · · · · · · · ·
Goal:	· · · · · · ·	· · · · · · · · · · · · · · · · · · ·	
	sends	message to Bob	
× Eve	observes	transmission, but should 1	earn nothing.
· · · · · · · · · · · ·	· · · · · · ·		· · · · · · · · · · ·
		· · · · · · · · · · · · · · · · · · ·	· · · · · · · · · · · ·

Motivation: Secure Communication i li e K he K B = Dec(c;k)c = Enc(m; k)Even of a state () Functionality Encryption Scheme Dec(Enc(m;k);k) = m k ~ Gen() Enci (m. ; k) 2) Secrecy $m \leftarrow Dec(c \circ k)$ Eve shouldn't learn abort in given c

Information -	- Theoretic	Solution: (Due-Time Pa	\mathcal{J}
$\begin{array}{cccccccccccccccccccccccccccccccccccc$		· · · · · · · · · · · ·	· · · · · · · · · · ·	· · · · · · · · · · ·
$ \Rightarrow c = c + c + c + c + c + c + c + c + c +$	uniformly	random bit	string (for	random K)
\Rightarrow $ eaks$	No inform	nation about		
		· · · · · · · · · · ·		

Information - T	hearefic	Solution		e-Time	Pad	· · · · · · · · · ·
$\begin{array}{cccccccccccccccccccccccccccccccccccc$	$ \begin{array}{cccccccccccccccccccccccccccccccccccc$	· · · · · · · ·	· · · · · ·	· · · · · · ·	· · · · ·	· · · · · · · · ·
$ \Rightarrow C is w$						
$ \begin{array}{cccccccccccccccccccccccccccccccccccc$	Jo inform	ictor al	sout a m		• • • •	
$\mathbf{D}_{\mathbf{u}} = \{\mathbf{u}_{\mathbf{u}}, \mathbf{u}_{\mathbf{u}}, \mathbf{u}, \mathbf{u}$		· · · · · · · ·	· · · · · ·			· · · · · · · · ·
Problem: Infer						· · · · · · · · · ·
La Requires						· · · · · · · · ·
· · · · · · · · · · · · · ·	· · · · · · ·	· · · · · · · ·		· · · · · · ·	· · · ·	· · · · · · · · ·
· · · · · · · · · · · · ·	· · · · · · ·					· · · · · · · · ·
· · · · · · · · · · · · · ·	· · · · · · ·	· · · · · · · ·		· · · · · · ·	· · · · ·	· · · · · · · · ·
· · · · · · · · · · · · ·						
· · · · · · · · · · · · ·						

Proposed	Solution:	Computational	lly - Bounded	Advensaries
· · · · · · · ·	· · · · · · · · ·	· · · · · · · · · ·	· · · · · · · · · · · · · ·	· · · · · · · · · · · · ·
. 	terre de la constante de la co		polynomial	Aime
	· · · · · · · · ·			

Proposed Solution: Computationally - Bounded Adversaries
E polynomial time
· · · · · · · · · · · · · · · · · · ·
Semantic Security (Goldwassen & Micali)
An Encryption Scheme is semantically secure
if every efficient earesdropper learns
(essentially) nothing about message m
given $c = Enc(m_j k)$

* Suppose problem TT & P. * Build Encryption Scheme	Computation	al Security	\sim	Reduction	
	* Suppose pr	roblem Tt & F		· ·	
1 1	* Build En	cryption Scheme	· · · · · ·	· · · · · · · · · · · ·	
1 1	· · · · · · · · · · · · · ·		· · · · · ·	· · · · · · · · · · · ·	
1 1	· · · · · ·
1 1	· · · · · · · · · · · · · ·	
1 1	· · · · · · · · · · · · · · ·	. .	· · · · · ·	· · · · · · · · · · · · ·	. .
· · · · · · · · · · · · · · · · · · ·					

Computational Security via Reduction
* Suppose problem TI & P
* Build Encryption Scheme
* Show Reduction (Security Proof)
(in poly-time) (Ever 140
⇒ Scheme is secure against poly-time adversaries!
$(else TT \in P)$
A A A A A A A A A A A A A A A A A A A

Announce	ments a constant			
× HWS	dne	· · · · · · · · · ·	· · · · · · · · · ·	· · · · · · · · · · · ·
X HW 9	. optional	for 4820	(graded	on completion)
· · · · · · · · · ·	required	for 5820	· · · · · · · · · ·	
X Final			· · · · · · · · · ·	
	info abort		review me	ter icly
	will be	posted t	E. J	· · · · · · · · · · · ·
. .	will be	posted. H		. .
. .		posted		. .
			. .	$\begin{array}{cccccccccccccccccccccccccccccccccccc$
. 	

Computational Security via Reduction
* Suppose problem TI & P
* Build Encryption Scheme
* Show Reduction (Security Proof)
(in poly-time) (Ever 140
⇒ Scheme is secure against poly-time adversaries!
$(else TT \in P)$
A A A A A A A A A A A A A A A A A A A

Computational Security via Reduction
* Suppose problem TF & P
* Build Encryption Scheme.
* Show Reduction (Security Proof)
(in poly-time) (Ever 140
⇒ Scheme is secure against poly-time adversaries!
$(else TT \in P)$

Computa	tional Handness for (L'uppography
× Need	a problem that	15 Hand
· · · · · · · · · ·	$\mathbb{T} = \left\{ \begin{array}{cccccccccccccccccccccccccccccccccccc$. .
* * B~+	NP-Hardness may v	lot work.
 	· · · · · · · · · · · · · · · · · · ·	. .
· · · · · · · · · ·	· · · · · · · · · · · · · · · · · · ·	· ·
· · · · · · · · · ·	· · · · · · · · · · · · · · · · · · ·	
 	· · · · · · · · · · · · · · · · · · ·	. .
		· ·

Computational Howdness for Cryptography
* Need a problem that is Hand.
$\mathcal{T} = \mathcal{T} = $
* But NP-Hardness may not work
A Dort MI Harvernesz Mad Mart
Le II her Le
Lo TT must be havd on typical instances
(3SAT hard in worst-case)
(3SAT hand in worst-case)

Computational Handness for Cryptography
* Need a problem that is Hand.
$\mathcal{T} \stackrel{\circ}{\leftarrow} \mathcal{P} \stackrel{\circ}{\leftarrow} \mathcal{P}$
* But NP-Hardness may not work
Lo TT must be hand on typical instances
(3SAT hard in worst-case)
L. Need to be able to generate
solved instances of TT easily.
(Given a SAT formula, how do we know if it's SAT or UNSAT ??)

One-Way	Functions	•
. .	fis easy to compute	•
· · · · · · · · · · · · · ·	(f(x))	•
f is HARD		
		•
	// Given f(x), havd to find x	•
. .		•
. .	(even for RANDOM x)	•
		•
	$(even for RANDOM \times)$	
	(even for RANDOM ×)	•

One-Way Functions	· · · ·	· · · · · · · · · ·	· · · · · · · · · · · · ·
· · · · · · · · · · · · · · · · · · ·	f is	s easy to ce	mpute
		$\sum_{i=1}^{n} \left(\frac{1}{2} \left(\frac{1}{2} \right) \right)^{n} = \left(\frac{1}{2} \left(\frac{1}{2} \right)^{n} = \left(\frac{1}{2} \left(\frac{1}{2} \right)^{n} = \left(\frac{1}{2} \left(\frac{1}{2} \right)^{n} \right)^{n} = \left(\frac{1}{2} \left(\frac{1}{2} \left(\frac{1}{2} \right)^{n} = \left(\frac{1}{2} \left(\frac{1}{2} \left(\frac{1}{2} \right)^{n} =$	· · · · · · · · · · · · · ·
	· · · · ·		· · · · · · · · · · · · ·
f is HARD to invert	· · · · ·	· · · · · · · · · ·	· · · · · · · · · · · · · · ·
Candidate OWFs	· · · · ·		
Multiplication	· · · · · ·	Factoring	
· · · · · · · · · · · · · · · · · · ·			· · · · · · · · · · · · · ·
Modular Exponentiation		Discrete	
x	· · · · · ·	· · · · · · · · · · · ·	. .

Def.	A one-way	function	f: Zo,13*	-> Z0,13*	satisfies
. .	Computation	EASY : t	There is c	en algoriti	m that
· · · · · ·	given x,	veturius		poly(1) (1) (1) (1)	the filme , a second
· · · · · ·	· · · · · · · · · ·	· · · · · · · · ·	· · · · · · · · · ·	· · · · · · · · ·	· · · · · · · · · · · ·
· · · · · ·					
· · · · · ·	· · · · · · · · · ·	· · · · · · · · ·	· · · · · · · · ·	· · · · · · · · ·	· · · · · · · · · · · ·
		· · · · · · · ·			
· · · · · ·	· · · · · · · · · ·	· · · · · · · · · ·	· · · · · · · · · ·	· · · · · · · · ·	· · · · · · · · · · · ·
					· · · · · · · · · · · ·

f: Zo, 13* -> Zo, 13* satisfies
there is an algorithm that $f(x)$ in poly (1x1) time.
For all polynomial -time
$f(x') = f(x) \int \int \frac{1}{2\pi} \int 1$
w/ "Negligible" probability,
cEN and sufficiently large w/ "Negligible" probability.

• •	.1	P.	•	VS.	• •	N	\sum	P	•	•			•	. (D~	ve ا	· -	<u>ل</u>)	Ŧ			c +	e e e e	1	ل	•	•	•	•	•			•		•	••••	•
		÷									•		0				*						÷	•				•				*	• •					• •	
		1	Fo	zc	÷			Ľ.	f	•	0~	re	-~	Ja	_مر				ς Ν	. fr	'n	<u> </u>			2 ×	: : :	s+))		•		•	• •				•	• •	•
• •	0	÷		•	• •	•		•	•		•	• •	٠	•	٠	•	•	•	• •	• •	•	•	٠	•	• •	•	•	•	•	٠	e	•	• •		•	•		• •	•
• •	•	•	•	•	• •	•	•	•	•	•	•	••••		P		Ŧ		N	19		r	•	•	•	• •	•	•	•	•	•	•	•	• •		•	•	•	• •	•
• •	•	•	•	•	• •	•	•	•	•	•	•	• •	•	•	•	•	•	•	• •		•	•	•	•	• •	•	•	• •	•	•	•	•	• •		•	•	•	• •	•
	•	•	•	•			•	•		•	•	• •	•	•		•	•	•			•	•	•	•		•	•	• •	•	•	•				•	•	•	• •	•
• •	•	•	•	•	• •		•	•	•	٠	•	• •	0	•	•	0	•	•			•	•	0	•	• •	•	•	• •	•	•	•	•	• •		•	•			•
			٠					٠		٠					•		٠					٠		·			٠	•		٠									٠
• •	•	•	•	•		•	•	•	•	•	•	• •	•	•	•	•	•	•	• •		•	•	•	•	• •	•	•	• •	•	•	•	•	• •		•	•	•		÷
• •	•	*	•		• •		•	•	•	•	0	• •		•	•		•		• •		•	•		•	• •	•	•	• •	•	•	0 0	•	• •	9 9 9 9		•		••••	•
• •	•	•	•	•	• •	•	•	•	•	•	•	• •	•	•	•	•	•	•	• •		•	•	•	•	• •	•	•	•	•	•	•	•	• •		•	•		• •	•
	•	•	•	•		•	•	•	•	•	•	• •	•	•	•	•	•	•			•	•	•	•	• •	•	•	• •	•	•	•	•			•	•	•	• •	•
• •	•	•	•	•	• •		•	•	•	•	•	• •	0	•	•		•	•	• •		•	•	0	•	• •	•	•	• •	•	•	•	•	• •			•			•
• •	•	•	•	•	• •		•	•	•	•	•	• •	•	•	•	•	•	•	• •	•	•	•	•	•	• •	•	•	• •		•	•	•		• •	•	•	•		•
• •	٠	٠	٠	٠	• •		٠	٠		٠	•	• •	٠	٠	٠	0	٠	0	• •	• •	٠	٠		٠	• •	٠	٠	e		٠	٠		• •		٠	٥	÷	• •	٠
																																						• •	
																																						• •	
																																						••••	
	٠							٠		•						٠	•												•		٠								

$\frac{1}{2} \frac{1}{2} \frac{1}$	NP 2	One - Nau) Functions	· · · · · · · ·	· · · · · · · · ·
Fact	If one-	-vaz fune	tions exist	· · · · · · · · · · · · · · · · · · ·
· · · · · · · · ·	· · · · · · · · · ·	$P \neq NP$	· · · · · · · · · · · ·	 	· · · · · · · · ·
<u>S</u>	P = NP =	⇒ No Cry	ptography =		
· · · · · · · · · ·	· · · · · · · · · ·	· · · · · · · · ·	· · · · · · · · · · · ·
· · · · · · · · · ·	· · · · · · · · · · ·	· · · · · · · · ·		
· · · · · · · · ·		· · · · · · · · ·		 	
			· · · · · · · · · · ·		

P. vs. NP 2 One-Way Function
Fact. If one-way functions exist,
$P \neq NP$
S_{o} $P = NP \implies No Cryptogrouphy \frac{1}{2}$
Open Question. Does $P \neq NP$ imply the existence of $OWFs^{?}_{0}$
· · · · · · · · · · · · · · · · · · ·
(See recent works of Prof. Rafael Pass & collaborators)
· · · · · · · · · · · · · · · · · · ·
· · · · · · · · · · · · · · · · · · ·

Candid	cte	OWF:	Multipli	cation	· · · · · ·	· · · · · · · · ·	•
$ \begin{array}{cccccccccccccccccccccccccccccccccccc$		$0,13 \times 70$		z z z z z z z z z z		· · · · · · · · ·	•
			· · · · · · · · · · · · ·		· · · · · ·	· · · · · · · · ·	
Factoring	· · · · · · <td< th=""><th>. </th><th>· · · · · · · · · · ·</th><th>· · · · · ·</th><th>· · · · · · · · ·</th><th>•</th></td<>	· · · · · · · · · · ·	· · · · · ·	· · · · · · · · ·	•
Given			find			G = N	•
				$\mathcal{T} \otimes \mathcal{C} = \mathcal{T} \otimes $			•
				$\sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{i=1}^{n} \sum_{i=1}^{n} \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{i=1}^{n} \sum_{i=1}^{n} \sum_{i=1}^{n} \sum_{i=1}^{n} \sum_{i$			•
	<td> <td>. .</td><td>. </td><td>. .</td><td>· · · · · · · · · · ·</td></td>	<td>. .</td> <td>. </td> <td>. .</td> <td>· · · · · · · · · · ·</td>	· · · · · · · · · · ·
. <td>. .</td> <td>•</td>	. .	•

Candidate OWF: Multiplication
$f_{\text{unit}} : \{0, 1\} \times \{0, 1\} \longrightarrow \{0, 1\}^2$
$f_{mn} l + \left(\begin{array}{c} P_{n} \\ P_{n} \end{array} \right) = \begin{array}{c} P_{n} \\ P_{n} \end{array} = \begin{array}{c} N_{n} \\ N_{n} \end{array}$
Factoring Given $N \in [0,1]^{2n}$, find factors s.t. $p \cdot q = N$
Conjecture Factoring & P.
Current Best Classical Attach: $2\tilde{O}(n^{1/3})$
· ·

Candidate OWF: Multiplication
$f_{m,t} : \frac{10,13}{2} \times \frac{10,13}{2} \longrightarrow \frac{10,13}{2}$
$\int m H \left($
Factoring Given $N \in [0,1]^{2n}$, find factors s.t. $p \cdot q = N$
Conjecture FACTORING & P.
Current Best Classical Attach: $2^{\tilde{O}(n^{1/3})}$
Shor's Quantum algorithm: poly-time!
FACTORING E BQP,

Is Mult	-iplication a OW		· · · · · · · · · · · · · · ·	· · · · · · · · · ·
	algorithus A	· · · · · · · · · ·		
	$\left[\begin{array}{c} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 $	$= p', q' \wedge h$	$P'' q'' = P'' q'' + \int \frac{1}{2} \frac{1}{2$	heglig:ble
· · · · · · · · ·		· · · · · · · · · ·		
· · · · · · · · · ·	· · · · · · · · · · · · · · ·		· · · · · · · · · · · · · · ·	· · · · · · · · · ·
· · · · · · · · ·		· · · · · · · · · ·		
· · · · · · · · ·	· · · · · · · · · · · · · · ·	· · · · · · · · · · · ·	· · · · · · · · · · · · · · ·	· · · · · · · · · ·
	· · · · · · · · · · · · · · ·			
	· · · · · · · · · · · · · ·			

Is Multiplication ~ OWF?
Fefficient algorithms A
$P_{rq} \sim E_{0,1} Z^{n} \left[A \left(1^{n}; P \cdot q \right) = p', q' \wedge P \cdot q' = P \cdot q \right] \leq heglig: ble$
L'with probability at least 3/4. Por q is even.
Por quiss even.
· · · · · · · · · · · · · · · · · · ·

Is Multiplication ~ OWF?	· · · · · · · · · · · · ·
Fefficient algorithms &	
$P_{iq} \sim E_{0,1} Z^{*} \left[A \left(1^{n}; Pq \right) = p', q' \land P'q' = P'q' \right]$	L hegl; j: ble
L with probability at least 3/4. Por q is even.	. .
$ = \frac{1}{2} \int \frac$	
Succeedy W.P.	

· · ·		ak	e 0			, , ,	· ·	• •	•	•	· · ·	•	•	•	•	· ·	•	•	•	· · ·		•	· · ·	•		· ·	• •	•	•	· ·	•	•	· ·	•	•	•
· ·	•	× •	D	ί2 Ι	, קעיי ע	b	∼†.	ù1			F.	•		Sf		 	_و.< 	r / 1	•			י כי ע י	, , , , , ,	i ti c	۰ مـ		່ (ຍ	•	•	• •	•	•	• •	•	•	•
• •	Pr	с Ь.(محرف	· · ·	•	P	9		ر	Z	, Ç	2		•	ho	- S .		+			er	,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,	· · ·	•	fa	c t	₽~ {	.]	•	• •	•	•	• •	• • •	•	•
••••	•	• •	•	• •	•	•	• •	• •	•	E	A A	łS	Y	•	•	• •	•	•	•	• •	•	•	••••••••••••••••••••••••••••••••••••••	•	•	• •		•	•	• •	•	•	• •	• • •	•	•
• •	•	• •	•	• •	•	•	• •	• •	•	•	• •	•	•	•	•	• •	•	•	•	• •	•	•	• •	•	•	• •		•	•	• •	•	•	• •	• •	•	•
••••	•	••••	•	••••	•	•	• •	• •	•	•	· ·		•	•	•	••••	•	•	•	••••	•		••••	•		• •		•	•	••••	•	•	• •		•	•
	•	• •	•	• •	•				•	•	• •		•	•	•	• •	•	•	•	• •	•	•	• •	•	•			•	•		•	•			•	•
· ·	•	· ·	•	• •	•	•	• •	• •	•	•	• •	•	•	•	•	• •	•	•	•	· ·	•	•	• •	•	•			•	•	· ·	•	•	• •	•	•	•
• •			•	• •	•	•	• •	o o	0	•	• •	•	•	0	•	o o	•	•	÷	• •	•	•	o o	•	•	• •		•	•	• •	0	•	• •	•	•	•
••••	•	• •	•	••••	•	•	• •	• •	•	•	· ·	•	•	•	•	· ·	•	•	•	· ·	•	•	• •	•	•	• •		•	•	· ·	•	•	• •	• • •	•	•
• •	•	• •	٠	• •	٠	٠	• •	• •	٠	٠	• •	•	٠	0	٠	• •	٠	٠	٠	• •	٠	÷		٠	٠		•	٠	0	• •	÷	٠		• •	٠	٠
• •	•	• •	٠	• •	٠	٠	• •	• •		٠	• •	•	٠	٠	•	• •	•	•	٠		•	٠		٠	٠		•	٠	٠	• •	0	•		• •	•	•
• •	•	• •	٠	• •	٠	٠	• •	• •	٠	٠	• •	•	٠	٠	•	• •	•	•	•	• •	•	÷		٠	٠		•	٠	0	• •	÷	•		• •	•	·
	•																																			

Takeaway
* Distribution of instances is critical!
Problem P, 9 ~ Zo, 13 ^h has too many factors? EASY
Fix. Sample P.9 ~ Random n-bit primes
=> only prime factors of N=P.9
Cirre P

· ·

Takeaway
* Distribution of instances is critical!
Problem p, q ~ Zo, 13 ^h has too many factors? EASY
Fix. Sample p.g. ~ Random n-bit primes
\Rightarrow only prime factors of $N = P \cdot q$ oure $P \cdot \frac{1}{2} \cdot q$
Assumption. Factoring the product of two n-bit primes is infeasible. => DWF.