Cryptographic Reductions: Classification and Applications to Ideal Models

Paul Baecher

Cryptographic Reductions: Classification and Applications to Ideal Models

Paul Baecher

Three Ways to Argue for Cryptographic Security

Cryptanalysis

Empirically evaluate real-world primitives

Information-theoretic arguments

Disregard any resource limitations

Provable security from assumptions

Efficient attackers only

Three Ways to Argue for Cryptographic Security

Provable security from assumptions

Efficient attackers only

Provable Security Follows a Common Structure

Construction

"To encrypt with ⟨construction⟩ , take the message and. . . "

Provable Security Follows a Common Structure

Construction	*"To encrypt with ⟨construction⟩ , take the message and. . . "*
Security proof	**Thm:** If ⟨assumption⟩ , then ⟨construction⟩ secure.

Provable Security Follows a Common Structure

Construction — *"To encrypt with ⟨construction⟩, take the message and..."*

Security proof — **Thm:** If ⟨assumption⟩, then ⟨construction⟩ secure in the ⟨ideal model⟩.

Provable Security Follows a Common Structure

Construction — *"To encrypt with ⟨construction⟩ , take the message and. . . "*

Security proof — **Thm:** If ⟨assumption⟩ , then ⟨construction⟩ secure in the ⟨ideal model⟩ .

Idealized primitive

Ideal Models Provide the "Best Possible" Primitive

Ideal model	Real life
Random oracle	MD5, SHA3, ...
Ideal cipher	DES, AES, ...

Ideal Models Provide the "Best Possible" Primitive

Ideal model	Real life
Random oracle	MD5, SHA3, ...
Ideal cipher	DES, AES, ...

Pick a random function from the set of all functions from k to n bits.

Comparing Two Constructions with Ideal-Model Proofs is Difficult

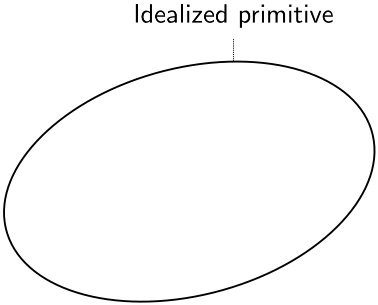
If $\langle$assump$\rangle$, then $\langle$constr$_1\rangle$ secure in the $\langle$ideal model$\rangle$.

If $\langle$assump$\rangle$, then $\langle$constr$_2\rangle$ secure in the $\langle$ideal model$\rangle$.

Comparing Two Constructions with Ideal-Model Proofs is Difficult

If $\langle$assump$\rangle$, then $\langle$constr$_1\rangle$ secure in the $\langle$ideal model$\rangle$.

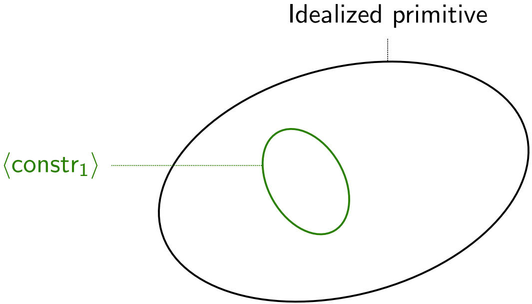
If $\langle$assump$\rangle$, then $\langle$constr$_2\rangle$ secure in the $\langle$ideal model$\rangle$.



Comparing Two Constructions with Ideal-Model Proofs is Difficult

If ⟨assump⟩, then ⟨constr_1⟩ secure in the ⟨ideal model⟩.

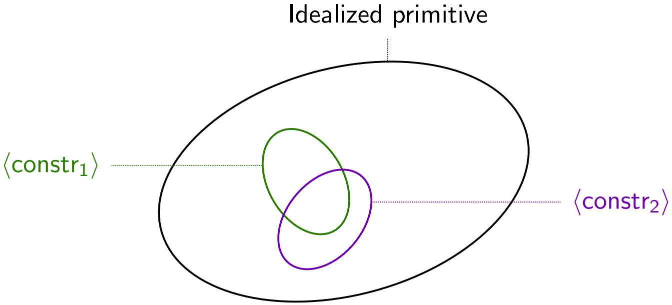
If ⟨assump⟩, then ⟨constr_2⟩ secure in the ⟨ideal model⟩.



Comparing Two Constructions with Ideal-Model Proofs is Difficult

If $\langle$assump$\rangle$, then $\langle$constr$_1\rangle$ secure in the $\langle$ideal model$\rangle$.

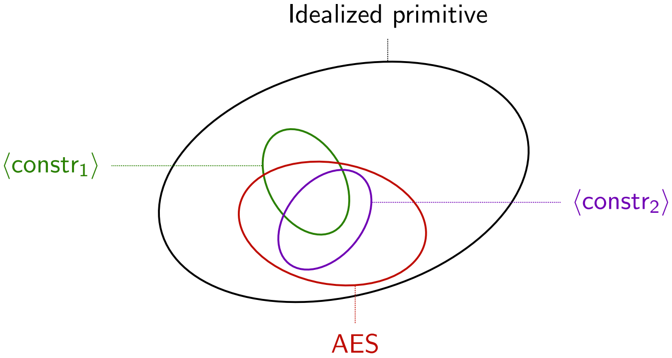
If $\langle$assump$\rangle$, then $\langle$constr$_2\rangle$ secure in the $\langle$ideal model$\rangle$.



Comparing Two Constructions with Ideal-Model Proofs is Difficult

If ⟨assump⟩, then ⟨constr$_1$⟩ secure in the ⟨ideal model⟩.

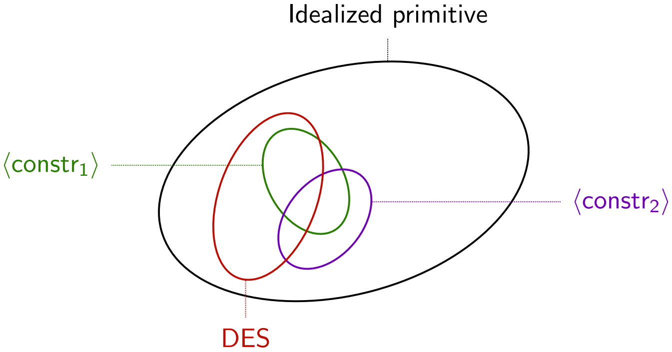
If ⟨assump⟩, then ⟨constr$_2$⟩ secure in the ⟨ideal model⟩.



Comparing Two Constructions with Ideal-Model Proofs is Difficult

If ⟨assump⟩, then ⟨constr_1⟩ secure in the ⟨ideal model⟩.

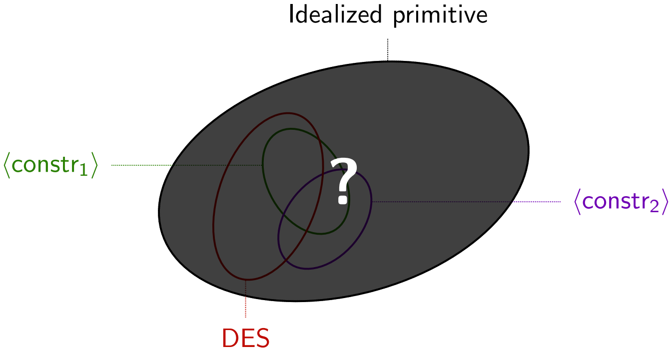
If ⟨assump⟩, then ⟨constr_2⟩ secure in the ⟨ideal model⟩.



Comparing Two Constructions with Ideal-Model Proofs is Difficult

If ⟨assump⟩, then ⟨constr_1⟩ secure in the ⟨ideal model⟩.

If ⟨assump⟩, then ⟨constr_2⟩ secure in the ⟨ideal model⟩.



Comparisons Might Still Be Possible Without Fully Understanding Ideal Primitives

Can we compare constructions relative to each other?

How do popular constructions compare?

Oracle reducibility enables sound comparisons of cryptographic constructions whose proofs are in ideal models.

Outline

[BF11,BFFS13]

Oracle reducibility
A versatile comparison paradigm

Ideal-cipher comparisons
Blockcipher-based compression functions

Random-oracle comparisons
ElGamal-type encryption schemes

Outline

[BF11,BFFS13]

Oracle reducibility

A versatile comparison paradigm

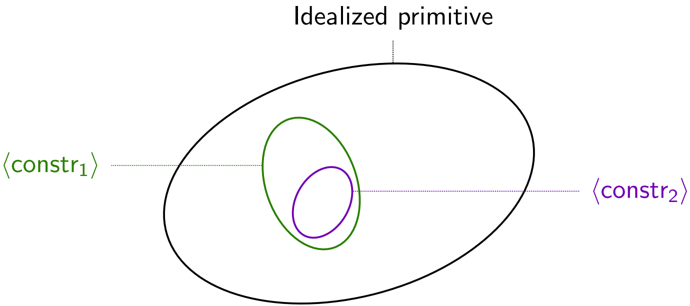
Ideal-cipher comparisons

Blockcipher-based compression functions

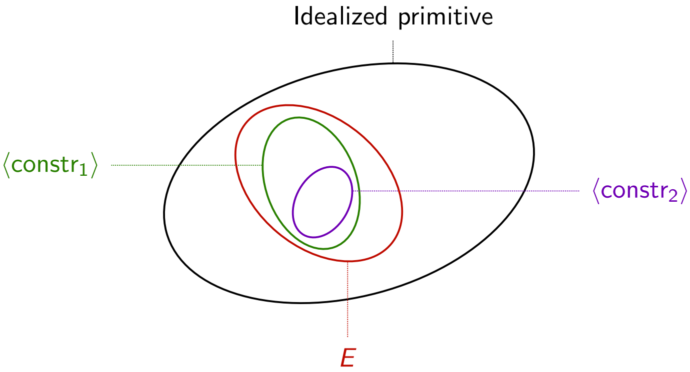
Random-oracle comparisons

ElGamal-type encryption schemes

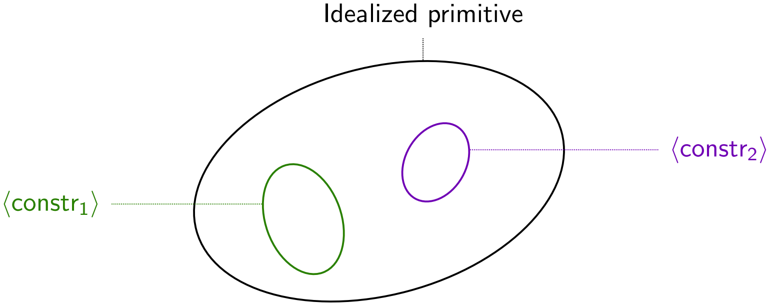
What Makes ⟨constr$_1$⟩ Secure Also Makes ⟨constr$_2$⟩ Secure



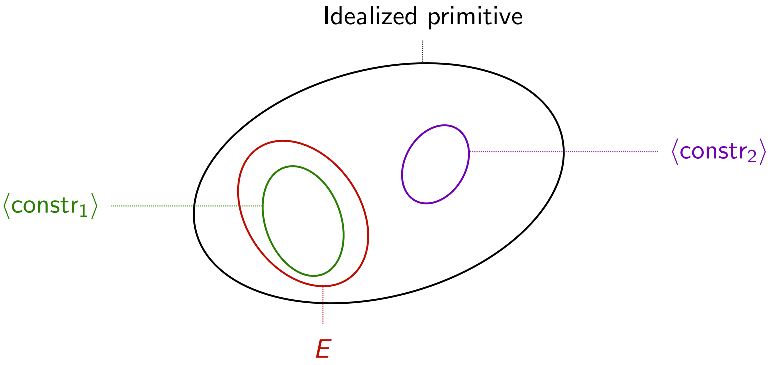
What Makes ⟨constr$_1$⟩ Secure Also Makes ⟨constr$_2$⟩ Secure



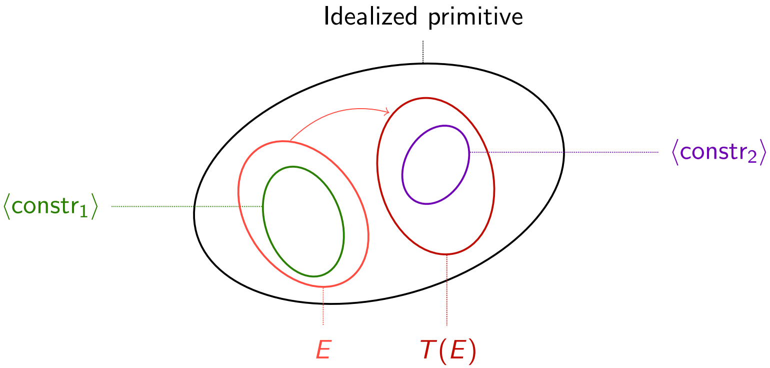
What Makes $\langle \mathsf{constr}_1 \rangle$ Secure Can be Adjusted to Make $\langle \mathsf{constr}_2 \rangle$ Secure



What Makes $\langle \text{constr}_1 \rangle$ Secure Can be Adjusted to Make $\langle \text{constr}_2 \rangle$ Secure



What Makes ⟨constr$_1$⟩ Secure Can be Adjusted to Make ⟨constr$_2$⟩ Secure



Formally Defining Oracle Reducibility

[**B**F11,**B**FFS13]

Direct reducibility

Any oracle O that makes C_1^O secure also makes C_2^O secure

Free reducibility

There exists T s.t. any oracle that makes C_1^O secure also makes $C_2^{T^O}$ secure

Formally Defining Oracle Reducibility

[**B**F11,**B**FFS13]

Direct reducibility

Any oracle O that makes C_1^O secure also makes C_2^O secure

$\Rightarrow$

Free reducibility

There exists T s.t. any oracle that makes C_1^O secure also makes $C_2^{T^O}$ secure

Outline

Oracle reducibility

A versatile comparison paradigm

[BFFS13]

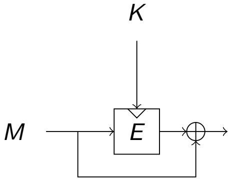
Ideal-cipher comparisons

Blockcipher-based compression functions

Random-oracle comparisons

ElGamal-type encryption schemes

Compression Functions Securely Shrink Their Input



$E(K, M) \oplus M$

Building block for hash functions

$2n$-to-n compression

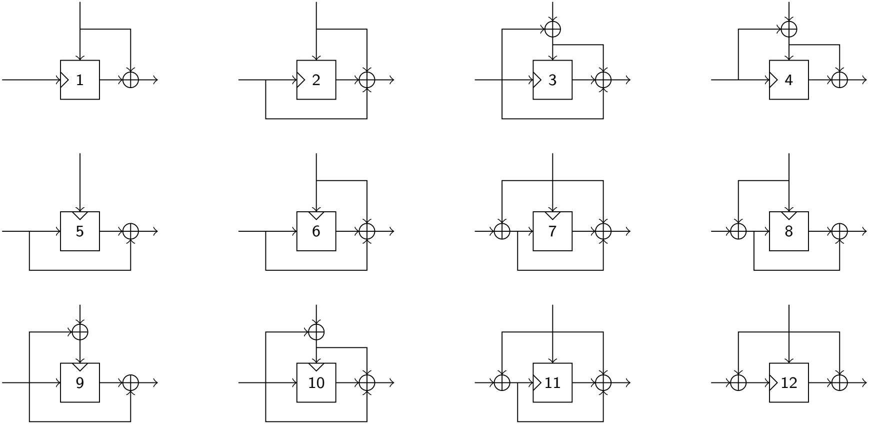
Built from a blockcipher

Design from [PGV93]

Collision resistant if E ideal

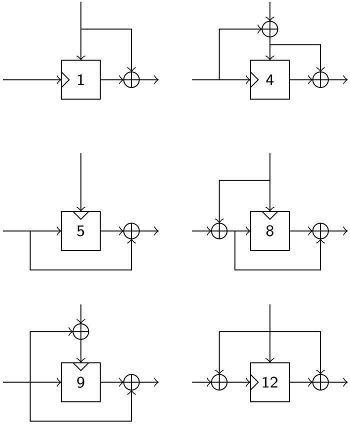
Proof due to [BRSS10]

PGV Functions

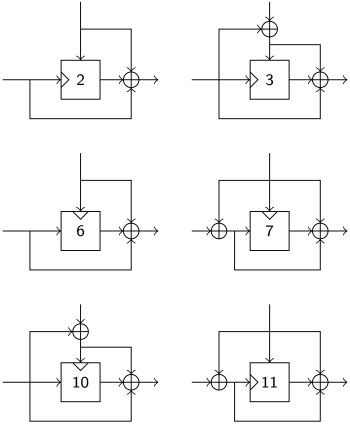


PGV Functions Fall Into Two Groups

[**B**FFS13]



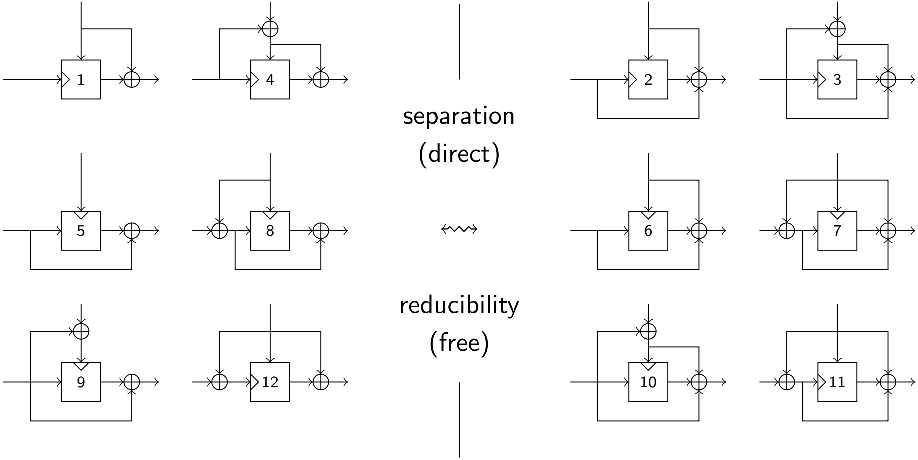
direct reducibility within



direct reducibility within

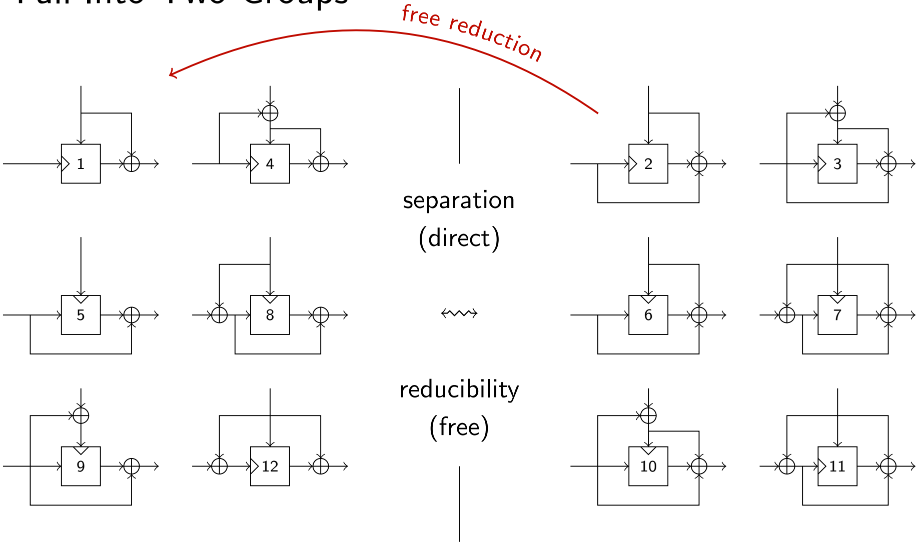
PGV Functions Fall Into Two Groups

[**B**FFS13]



PGV Functions Fall Into Two Groups

[BFFS13]



Free Reduction From PGV_2 to PGV_1

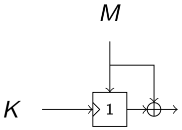
There exists T s.t. for any E: PGV_1^E secure $\Rightarrow$ $\mathrm{PGV}_2^{T^E}$ secure

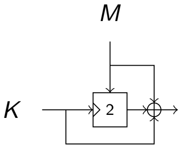
Free Reduction From PGV_2 to PGV_1

There exists T s.t. for any E: PGV_1^E secure $\Rightarrow$ $\mathrm{PGV}_2^{T^E}$ secure

$$T^E(K, M) := E(K, M) \oplus K$$

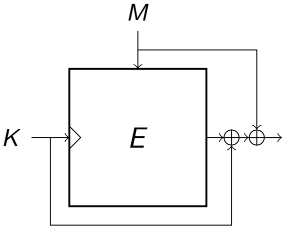
Free Reduction From PGV_2 to PGV_1



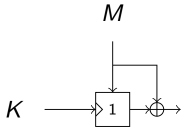


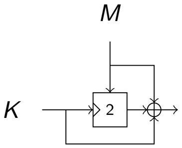
There exists T s.t. for any E: PGV_1^E secure $\Rightarrow$ $\mathrm{PGV}_2^{T^E}$ secure

$$T^E(K, M) := E(K, M) \oplus K$$



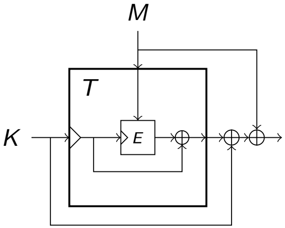
Free Reduction From PGV_2 to PGV_1



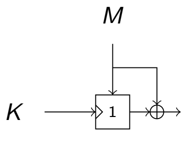


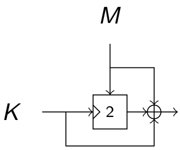
There exists T s.t. for any E: PGV_1^E secure $\Rightarrow$ $\mathrm{PGV}_2^{T^E}$ secure

$$T^E(K, M) := E(K, M) \oplus K$$



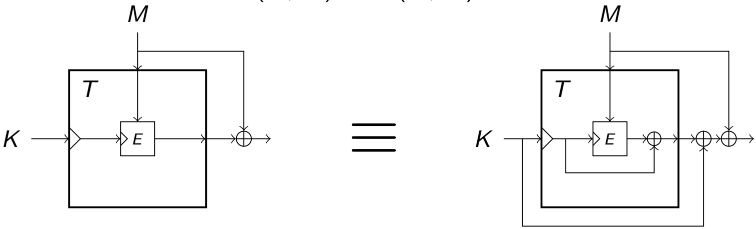
Free Reduction From PGV_2 to PGV_1





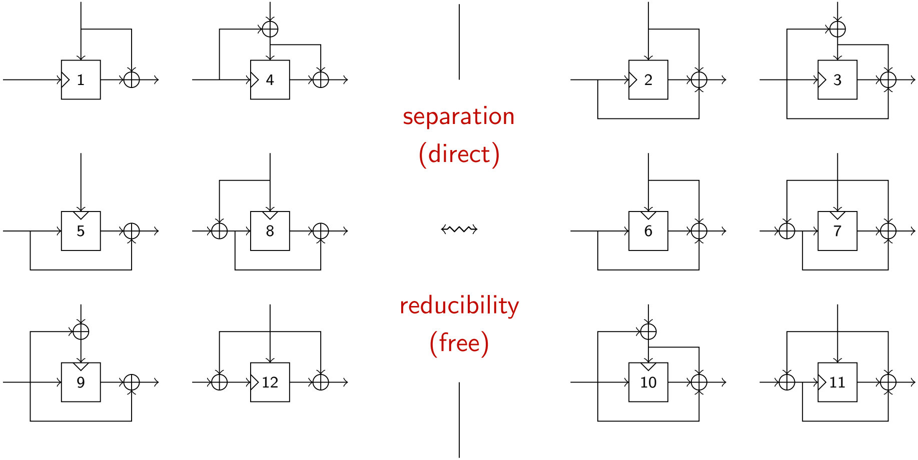
There exists T s.t. for any E: PGV_1^E secure $\Rightarrow$ $\mathsf{PGV}_2^{T^E}$ secure

$$T^E(K, M) := E(K, M) \oplus K$$



PGV Functions Fall Into Two Groups

[**B**FFS13]



Groups are Incomparable, No Clear Winner

No direct reducibility from #1 to #2

Or vice versa

Free reducibility “switches” group

But no simultaneous security for both

Groups are Incomparable, No Clear Winner

$$E \text{ s.t. } \begin{cases} \#1 \text{ secure} \\ \#2 \text{ ???} \end{cases}$$

No direct reducibility from #1 to #2

Or vice versa

Free reducibility "switches" group

But no simultaneous security for both

Groups are Incomparable, No Clear Winner

$$E \text{ s.t. } \begin{cases} \#1 \text{ secure} \\ \#2 \text{ ???} \end{cases}$$

$\downarrow T$

$$T(E) \text{ s.t. } \begin{cases} \#1 \text{ ???} \\ \#2 \text{ secure} \end{cases}$$

No direct reducibility from #1 to #2

Or vice versa

Free reducibility "switches" group

But no simultaneous security for both

Groups are Incomparable, No Clear Winner

$$T(T(E)) \text{ s.t. } \begin{cases} \#1 \text{ secure} \\ \#2 \text{ ???} \end{cases}$$

$$\uparrow T$$

$$T(E) \text{ s.t. } \begin{cases} \#1 \text{ ???} \\ \#2 \text{ secure} \end{cases}$$

No direct reducibility from $\#1$ to $\#2$

Or vice versa

Free reducibility “switches” group

But no simultaneous security for both

Outline

Oracle reducibility

A versatile comparison paradigm

Ideal-cipher comparisons

Blockcipher-based compression functions

[BF11] Random-oracle comparisons

ElGamal-type encryption schemes

Cryptographic Constructions Often Undergo Iterative Improvements

Feasibility result

Not practical, but it works

Practical result

Simpler, tighter, faster, . . .

Further improvements

Milder or fewer assumptions

Cryptographic Constructions Often Undergo Iterative Improvements

Further improvements

Milder or fewer assumptions

An "Improved" Construction May be Worse in Other Ways

If a_1 and a_2 hold, then C is secure in $\langle$ideal model$\rangle$.

$\overset{?}{<}$

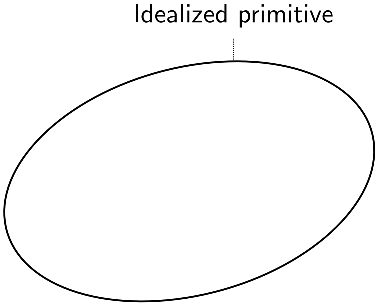
If a_1 holds, then C' is secure in $\langle$ideal model$\rangle$.

An "Improved" Construction May be Worse in Other Ways

If a_1 and a_2 hold, then C is secure in ⟨ideal model⟩.

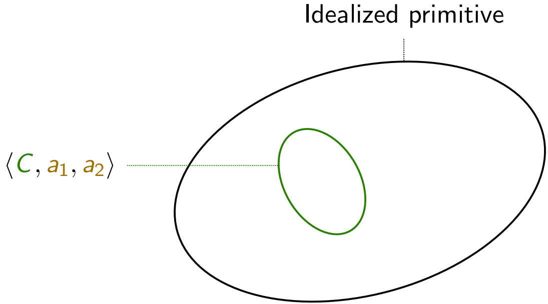
$\stackrel{?}{<}$

If a_1 holds, then C' is secure in ⟨ideal model⟩.



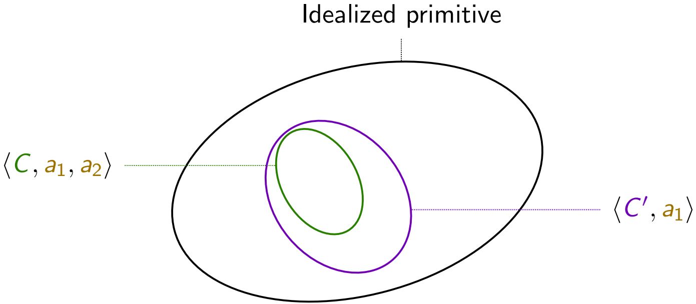
An "Improved" Construction May be Worse in Other Ways

If a_1 and a_2 hold, then C is secure in ⟨ideal model⟩. $\overset{?}{<}$ If a_1 holds, then C' is secure in ⟨ideal model⟩.



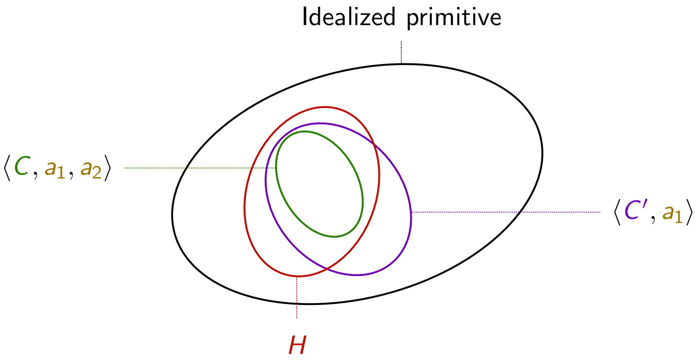
An "Improved" Construction May be Worse in Other Ways

If a_1 and a_2 hold, then C is secure in ⟨ideal model⟩. $\overset{?}{<}$ If a_1 holds, then C' is secure in ⟨ideal model⟩.



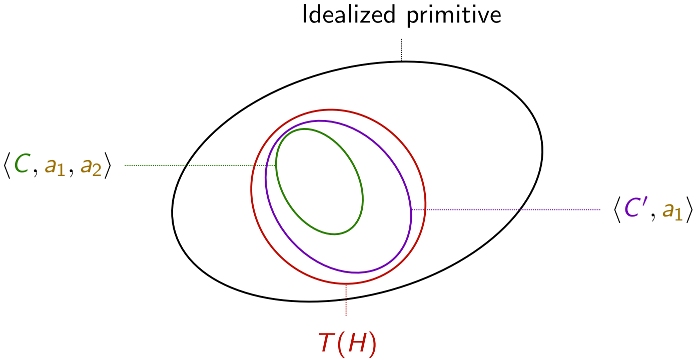
An "Improved" Construction May be Worse in Other Ways

If a_1 and a_2 hold, then C is secure in ⟨ideal model⟩. $\overset{?}{<}$ If a_1 holds, then C' is secure in ⟨ideal model⟩.



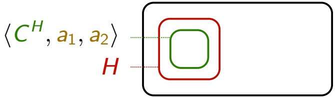
An "Improved" Construction May be Worse in Other Ways

If a_1 and a_2 hold, then C is secure in ⟨ideal model⟩. $\stackrel{?}{<}$ If a_1 holds, then C' is secure in ⟨ideal model⟩.



For Assumptions a_1, a_2 Three Notions Emerge

[**B**F11]

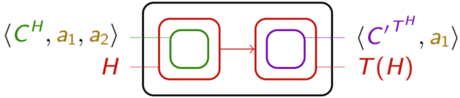


Strict reducibility

Definitely better

For Assumptions a_1, a_2 Three Notions Emerge

[**B**F11]



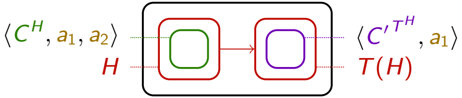
Strict reducibility

Definitely better

For Assumptions a_1, a_2

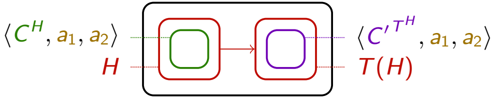
Three Notions Emerge

[**B**F11]



Strict reducibility

Definitely better

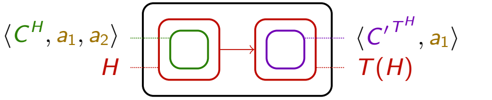


Weak reducibility

As good as

For Assumptions a_1, a_2 Three Notions Emerge

[**B**F11]

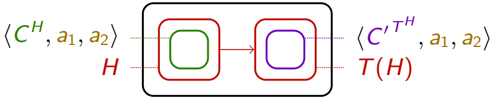


Strict reducibility

Definitely better

Strong reducibility

As good as, possibly better

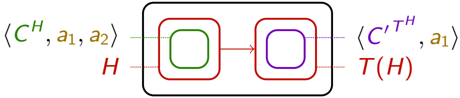


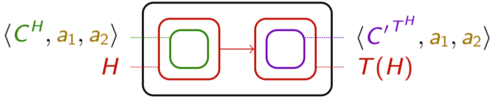
Weak reducibility

As good as

For Assumptions a_1, a_2 Three Notions Emerge

[BF11]





Strict reducibility

Definitely better

$\Downarrow \quad \not\Uparrow$

Strong reducibility

As good as, possibly better

$\Downarrow \quad \not\Uparrow$

Weak reducibility

As good as

An Example Where the Improved Construction is Indeed Better

[**B**F11]

Hashed ElGamal encryption scheme

Improved scheme from [CKS09]

Milder assumption

[Strong] Diffie–Hellmann assumption

An Example Where the Improved Construction is Indeed Better

[**B**F11]

Hashed ElGamal encryption scheme
Improved scheme from [CKS09]

Milder assumption
[Strong] Diffie–Hellmann assumption

Strong reducibility
Possibly better, but not worse

Review and Conclusions

Comparison technique

Relative security regarding primitives

Review and Conclusions

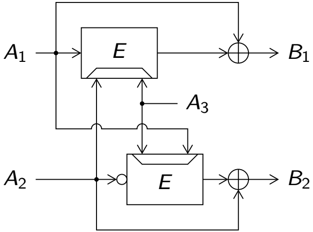
Comparison technique

Relative security regarding primitives

Various compression-function designs

Two groups, incomparable, superior one*

Review and Conclusions



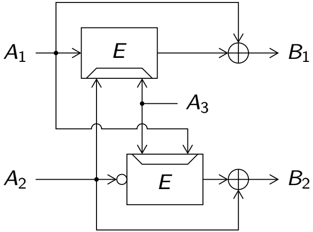
Comparison technique

Relative security regarding primitives

Various compression-function designs

Two groups, incomparable, superior one*

Review and Conclusions



Comparison technique

Relative security regarding primitives

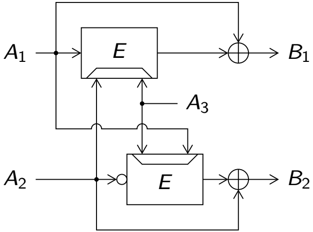
Various compression-function designs

Two groups, incomparable, superior one*

ElGamal-type encryption schemes

Construction in [CKS09] is possibly better

Review and Conclusions



Comparison technique

Relative security regarding primitives

Various compression-function designs

Two groups, incomparable, superior one*

ElGamal-type encryption schemes

Construction in [CKS09] is possibly better

Results enable sound comparison

Guidance for implementors facing choices

List of Publications

[BBF13] Notions of Black-Box Reductions, Revisited. Paul Baecher, Christina Brzuska, Marc Fischlin. ASIACRYPT 2013.

[BBM13] Reset Indifferentiability and its Consequences. Paul Baecher, Christina Brzuska, Arno Mittelbach. ASIACRYPT 2013.

[BFFS13] Ideal-Cipher (Ir)reducibility for Blockcipher-Based Hash Functions. Paul Baecher, Pooya Farshim, Marc Fischlin, Martijn Stam. EUROCRYPT 2013.

[BF11] Random Oracle Reducibility. Paul Baecher, Marc Fischlin. CRYPTO 2011.

[BFS11] Expedient Non-Malleability Notions for Hash Functions. Paul Baecher, Marc Fischlin, Dominique Schröder. CT-RSA 2011.

[BBFM11] Breaking reCAPTCHA: A Holistic Approach via Shape Recognition. Paul Baecher, Niklas Büscher, Marc Fischlin, Benjamin Milde. IFIP SEC 2011.

[BFGLLS10] CAPTCHAs: The Good, the Bad, and the Ugly. Paul Baecher, Marc Fischlin, Lior Gordon, Robert Langenberg, Michael Luetzow, Dominique Schröder. LNI 2010.

[BKB09] PUF-Based Authentication Protocols – Revisited. Heike Busch, Stefan Katzenbeisser, Paul Baecher. WISA 2009.

[ABFGH09] Massively-Parallel Simulation of Biochemical Systems. Jens Ackermann, Paul Baecher, Thorsten Franzel, Michael Goesele, Kay Hamacher. LNI 2009.

[BKHDF06] The Nepenthes Platform: An Efficient Approach to Collect Malware. Paul Baecher, Markus Koetter, Thorsten Holz, Maximillian Dornseif, Felix C. Freiling. RAID 2006.

List of Publications

[BBF13] Notions of Black-Box Reductions, Revisited. Paul Baecher, Christina Brzuska, Marc Fischlin. ASIACRYPT 2013.

[BBM13] Reset Indifferentiability and its Consequences. Paul Baecher, Christina Brzuska, Arno Mittelbach. ASIACRYPT 2013.

[BFFS13] Ideal-Cipher (Ir)reducibility for Blockcipher-Based Hash Functions. Paul Baecher, Pooya Farshim, Marc Fischlin, Martijn Stam. EUROCRYPT 2013.

[BF11] Random Oracle Reducibility. Paul Baecher, Marc Fischlin. CRYPTO 2011.

[BFS11] Expedient Non-Malleability Notions for Hash Functions. Paul Baecher, Marc Fischlin, Dominique Schröder. CT-RSA 2011.

[BBFM11] Breaking reCAPTCHA: A Holistic Approach via Shape Recognition. Paul Baecher, Niklas Büscher, Marc Fischlin, Benjamin Milde. IFIP SEC 2011.

[BFGLLS10] CAPTCHAs: The Good, the Bad, and the Ugly. Paul Baecher, Marc Fischlin, Lior Gordon, Robert Langenberg, Michael Luetzow, Dominique Schröder. LNI 2010.

[BKB09] PUF-Based Authentication Protocols – Revisited. Heike Busch, Stefan Katzenbeisser, Paul Baecher. WISA 2009.

[ABFGH09] Massively-Parallel Simulation of Biochemical Systems. Jens Ackermann, Paul Baecher, Thorsten Franzel, Michael Goesele, Kay Hamacher. LNI 2009.

[BKHDF06] The Nepenthes Platform: An Efficient Approach to Collect Malware. Paul Baecher, Markus Koetter, Thorsten Holz, Maximillian Dornseif, Felix C. Freiling. RAID 2006.

List of Publications

[BBF13] Notions of Black-Box Reductions, Revisited. Paul Baecher, Christina Brzuska, Marc Fischlin. ASIACRYPT 2013.

[BBM13] Reset Indifferentiability and its Consequences. Paul Baecher, Christina Brzuska, Arno Mittelbach. ASIACRYPT 2013.

[BFFS13] Ideal-Cipher (Ir)reducibility for Blockcipher-Based Hash Functions. Paul Baecher, Pooya Farshim, Marc Fischlin, Martijn Stam. EUROCRYPT 2013.

[BF11] Random Oracle Reducibility. Paul Baecher, Marc Fischlin. CRYPTO 2011.

[BFS11] Expedient Non-Malleability Notions for Hash Functions. Paul Baecher, Marc Fischlin, Dominique Schröder. CT-RSA 2011.

[BBFM11] Breaking reCAPTCHA: A Holistic Approach via Shape Recognition. Paul Baecher, Niklas Büscher, Marc Fischlin, Benjamin Milde. IFIP SEC 2011.

[BFGLLS10] CAPTCHAs: The Good, the Bad, and the Ugly. Paul Baecher, Marc Fischlin, Lior Gordon, Robert Langenberg, Michael Luetzow, Dominique Schröder. LNI 2010.

[BKB09] PUF-Based Authentication Protocols – Revisited. Heike Busch, Stefan Katzenbeisser, Paul Baecher. WISA 2009.

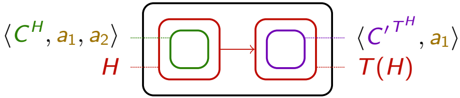
[ABFGH09] Massively-Parallel Simulation of Biochemical Systems. Jens Ackermann, Paul Baecher, Thorsten Franzel, Michael Goesele, Kay Hamacher. LNI 2009.

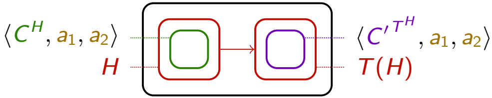
[BKHDF06] The Nepenthes Platform: An Efficient Approach to Collect Malware. Paul Baecher, Markus Koetter, Thorsten Holz, Maximillian Dornseif, Felix C. Freiling. RAID 2006.

Thank you!

For Assumptions a_1, a_2 Three Notions Emerge

[BF11]





Strict reducibility

Definitely better

$\Downarrow \quad \not\Uparrow$

Strong reducibility

As good as, possibly better

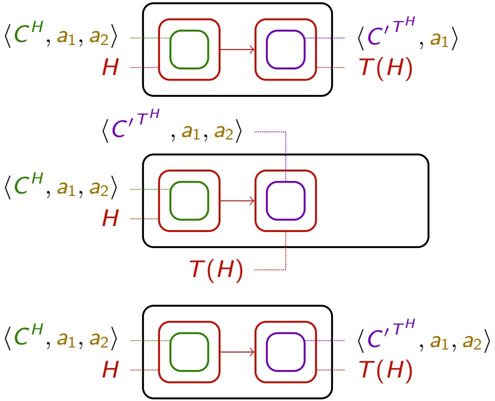
$\Downarrow \quad \not\Uparrow$

Weak reducibility

As good as

For Assumptions a_1, a_2 Three Notions Emerge

[**B**F11]



Strict reducibility

Definitely better

$\Downarrow \quad \not\Uparrow$

Strong reducibility

As good as, possibly better

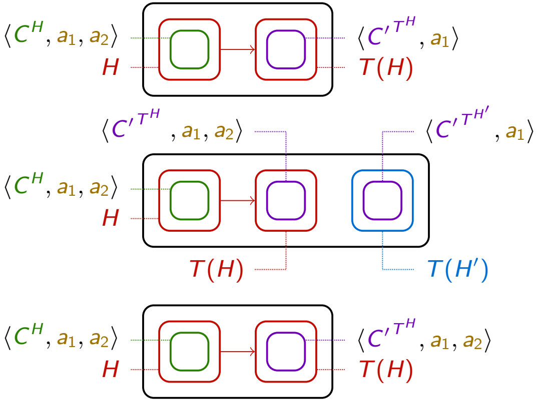
$\Downarrow \quad \not\Uparrow$

Weak reducibility

As good as

For Assumptions a_1, a_2 Three Notions Emerge

[**B**F11]



Strict reducibility
Definitely better

$\Downarrow \quad \not\Uparrow$

Strong reducibility
As good as, possibly better

$\Downarrow \quad \not\Uparrow$

Weak reducibility
As good as