Security in the White-box Setting

Avik Chakraborti

Institute for Advancing Intelligence, TCG CREST

Background

How to Protect a Secret? In Secure Hardware

- Put it in a smart card

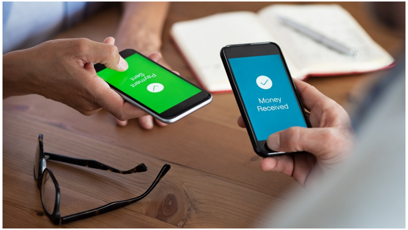
- Or in other secure hardware (say, HSM)

But What are the Disadvantages?

- Secure hardware is expensive
- Difficulty in upgrading. If a weakness is exposed, it is not easy to upgrade. Bugs, security flaws might occur.

Software Solutions are Better

- Cheaper
- Easy to update
- Easy to fix
- Application Example-
  - Digital rights management (DRM) (Adv: User)
  - Mobile payment (Adv: Malware)
  - Car Connectivity
  - and others......



Examples

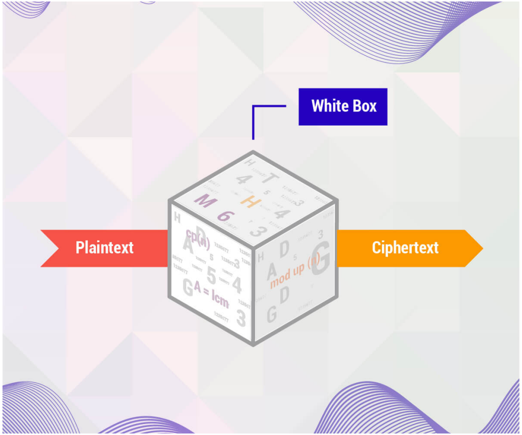
Overall

- Possible threats
    - Malware
    - Co-hosted apps
    - Users
- Adversary can
    - Fully control over the execution environment
    - Reverse-engineer
    - Access memory
    - Retrieve the secret

Intro to White-box Crypto

Breiefly, White-Box Cryptography (WBC) was

- Chow et al [CEJO01] in SAC 2001
- As a special-purpose obfuscation for AES
- Adversary has full access and control to the implementation and execution environment respectively
- Main goal is to make key extraction difficult
- Several other new goals were proposed later

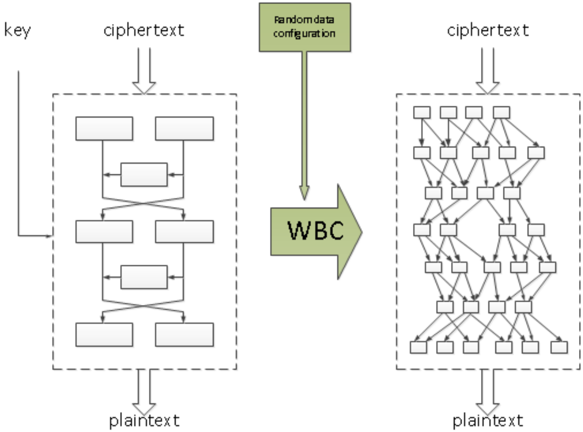


What is an Obfuscator?

- A word π in language L ($\pi \in$ L is some encoded string)
- Obfuscator O: A compiler that takes a program Π with an embedded secret S, denoted by Π_S, such that
  - $O(\Pi_S) \equiv \Pi_S$
  - No info on S is revealed given full access $O(\Pi_S)$

White-Box AES

- Here $\pi = AES$, $S = K$, and $\pi_S = AES_K$
- Target: Hide secret key in obfuscated key-embedded code for AES
- Simple but Inefficient solution: Huge table $O(AES_K) = T$, s.t $T[i] = AES_K(i)$
- Chow et al's work: Network of small tables masked with random non-linear encoding
- Broken in three years [BGE04]
- Several other dedicated designs: Broken

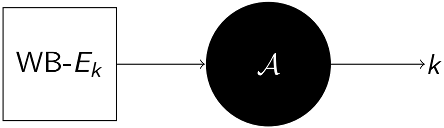


Formal Security Notions

First Attempt on Formal White-box Security Notions [DLPR13]

- Unbreakability
- One-Wayness
- **Incompressibility**
- ~~Traceability~~ (for public key solutions)

Unbreakability (Addresses Key-extraction)

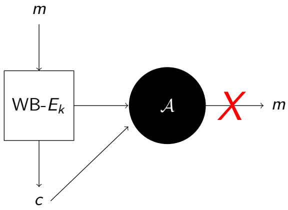


For simplicity use $Comp(S)$ instead of $Comp(\pi_S)$. Let $\mathcal{E}$ be an encryption scheme, $C_{\mathcal{E}}$ be a white-box compiler, and $\mathcal{A}$ be an adversary. For $atk \in \{\text{CPA, CCA, RCA}\}$, The success probability for *Unbreakability* is defined as

$$\text{Succ}_{\mathcal{A},C_{\mathcal{E}}}^{ubk-atk} = Pr[k \xleftarrow{\$} \mathcal{K}, r \xleftarrow{\$} \mathcal{R}, P = C_{\mathcal{E}}(k,r), \hat{k} \xleftarrow{\$} \mathcal{A}^{\mathcal{O}}(P) : \hat{k} = k], \ \ s.t$$

- $\mathcal{O}(.) = \epsilon$ when $atk = \text{CPA}$, $\mathcal{O}(.) = \mathcal{D}_k(.)$ when $atk = \text{CCA}$, and
- $\mathcal{O}(.) = C_{\mathcal{E}}(k, \mathcal{R})$ when $atk = \text{RCA}(recompilationattack)$

One-Wayness

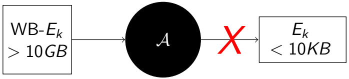


Let $\mathcal{E}$ be an encryption scheme, $C_{\mathcal{E}}$ be a white-box compiler, and $\mathcal{A}$ be an adversary. For $atk \in \{\text{CPA, CCA, RCA}\}$, The success probability for *One-wayness*, $\text{Succ}^{ow-atk}_{\mathcal{A},C_{\mathcal{E}}}$ is defined as

$$Pr[k \xleftarrow{\$} \mathcal{K}, r \xleftarrow{\$} \mathcal{R}, P = C_{\mathcal{E}}(k,r), m \xleftarrow{\$} \mathcal{M}, c = \mathcal{E}_k(m), \hat{m} \xleftarrow{\$} \mathcal{A}^{\mathcal{O}}(P,c) : \hat{m} = m], \; s.t$$

- $\mathcal{O}(.) = \epsilon$ when $atk =$ CPA, $\mathcal{O}(.) = \mathcal{D}_k(.)$ when $atk =$ CCA, and
- $\mathcal{O}(.) = C_{\mathcal{E}}(k, \mathcal{R})$ when $atk =$ RCA

(λ, δ)-Incompressibility (Addresses Code-Lifting)



Let $\mathcal{E}$ be an encryption scheme, $C_{\mathcal{E}}$ be a white-box compiler, and $\mathcal{A}$ be an adversary. For $atk \in \{\text{CPA, CCA, RCA}\}$, let $\text{Adv}^{(\lambda,\delta)-inc-atk}_{\mathcal{A},C_{\mathcal{E}}}$ is defined as

$$Pr[k \xleftarrow{\$} \mathcal{K}, r \xleftarrow{\$} \mathcal{R}, P = C_{\mathcal{E}}(k,r), P_{com} \xleftarrow{\$} \mathcal{A}^{\mathcal{O}}(P) : \Delta(P_{com}, \mathcal{E}_k(.)) \leq \delta \cap |P_{com}| < \lambda]\ s.t$$

- $\mathcal{O}(.) = \epsilon$ when $atk = \text{CPA}$, $\mathcal{O}(.) = \mathcal{D}_k(.)$ when $atk = \text{CCA}$, and
- $\mathcal{O}(.) = C_{\mathcal{E}}(k, \mathcal{R})$ when $atk = \text{RCA}$

Here, $\Delta(P, f) = Pr[a \xleftarrow{\$} A, b \leftarrow P(a) : b \neq f(a)]$. We say $C_{\mathcal{E}}$ is (t, ϵ) secure in the sense of (λ, δ)-*inc-atk*, if for any $\mathcal{A}$ with running time t, $\text{Adv}^{(\lambda,\delta)-inc-atk}_{\mathcal{A},C_{\mathcal{E}}} \leq \epsilon$.

Variants of One-way ness

Strong White-box [BBK14]

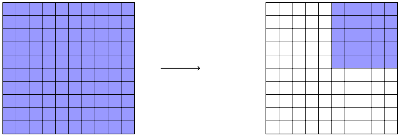
- Adversary should be unable to mimic decryption function, given white-box code of E_K.
- Resemblance with trapdoor perm
- CPA setting, goal is stronger
- Used multivariate crypto: lack reductions to established assumption
- Broken in [GPT15] (key recovery), [DDKL15] (decomposition), [MDFK15] (key recovery)

Challenger		$\mathcal{A}$
chooses $K \xleftarrow{\$} \{0,1\}^k$		
computes $P \xleftarrow{\$} Comp(K)$		
	$\xrightarrow{P}$	
	$\xleftarrow{\mathcal{D}'}$	
check $\mathcal{D}' \approx D_K$ or not		

Variants of Incompressibility: Most Desired Notion

Background: Code-Lifting Attack

- When key-extraction is not possible, the adversary lifts the code



- First addressed in [DLPR13]
- Large incompressible program
- Full code distribution hard for a attacker
- Incompressible design by [BBK14], [BI15], [FKKM16], [BIT16]
- Several other designs

Important point

- Unbreakability, One-way ness: Single adversary (say malware)
- Incomprerssibility: Two adversaries ($\mathcal{A}_{local}, \mathcal{A}_{remote}$)

Total Four Attempts of Formalizing Security Against Code-lifting

- Incompressibility by [DLPR13] (already discussed above)
- Weak Whitebox by [BBK14]
- Space-hardness by [BI15]
- Weak and Strong Incompressibility by [FKKM16]

Second Attempt: T-secure Weak White-box (w.r.t $\mathcal{A}_{local}$) [BBK14]

Challenger ; $\mathcal{A}_{local}$

chooses $K \xleftarrow{\$} \{0,1\}^k$

computes $P \xleftarrow{\$} Comp(K)$

$\xrightarrow{\quad P \quad}$

computes $\mathcal{E}'$ with $|\mathcal{E}'| \leq T$

$\xleftarrow{\quad \mathcal{E}' \quad}$

check $\mathcal{E}' \approx E_K$ or not

T-secure Weak White-box

- Simply, adversary who gets a secure weak white-box implementation is unable to find out any compact (shorter than T) equivalent representation of it

- ASASA structure based cipher (two secret non-linear layer + three secret affine layer)

- Uses a dedicated small block cipher for the keyed-sbox.

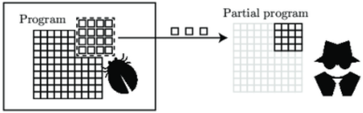
- Broken in a year [DDKL15] (Decomposition Attack), [MDFK15] (Key-Recovery attack)

Third Attempt: Informal Variant of Incompressibility [BI15]

The notion is called Space-hardness

- The difficulty of code lifting attack is measured by space-hardness
- Introduced by Bogdanov et al., in ACM, 2015 and proposed a space-hard cipher SPACE
- ASIACRYPT 2016 paper [BIT16] proposed white-box implementation of SPNbox cipher achieving better space-hardness than SPACE

Space Hardness



Attack setup: (1) Local adversary (leaks), (2) Remote adversary (receives leakage)

(M, z) Weak Space Hardness

An encryption scheme E_k is said to be weak (M, z) space hard if it is infeasible to encrypt (decrypt) any randomly chosen plaintext (ciphertext) with probability more than 2^{-z} given any code (table) of size less than M-bits

(M, Z) Strong space-hardness

An encryption scheme E_k is said to be strong (M, z) space-hard if it is infeasible to compute a plaintext-ciphertext pair with probability more than 2^{-z} given any code (table) of size less than M-bits

Weak and Strong Space-hardness

Weak Space-hardness

Challenger		$\mathcal{A}_{remote}$
chooses $k \xleftarrow{\$} \mathcal{K}$		
computes $P \xleftarrow{\$} Comp(k)$		
$\lvert P \rvert = T$	$\xrightarrow{L, \ni \lvert L \rvert \leq M}$	
$m^{ch} \xleftarrow{\$} \mathcal{M}$	$\xrightarrow{m^{ch}}$	
	$\xleftarrow{c}$	
check $c = P(m^{ch})$ or not		

Strong Space-hardness

Challenger		$\mathcal{A}_{remote}$
chooses $k \xleftarrow{\$} \mathcal{K}$		
computes $P \xleftarrow{\$} Comp(k)$		
$\lvert P \rvert = T$	$\xrightarrow{L, \ni \lvert L \rvert \leq M}$	
	$\xleftarrow{(m^{ch}, c^{ch})}$	
check $c^{ch} = P(m^{ch})$ or not		

Power of Adversary

- Known Space (KS) Attack
- Chosen Space (CS) Attack
- Adaptive Chosen Space (ACS) Attack

Known Space (KS) Attack

Challenger | $\mathcal{A}_{remote}$

chooses $k \xleftarrow{\$} \mathcal{K}$

computes $P \xleftarrow{\$} Comp(k)$

$|P| = T$

computes $y_i = P_j^r(x_i)$

$\xrightarrow{(x_1, y_1), \ldots, (x_q, y_q)}$

$m^{ch} \xleftarrow{\$} \mathcal{M}$

$\xrightarrow{m^{ch}}$

$\xleftarrow{c}$

check $c = P(m^{ch})$ or not

Chosen Space (CS) Attack

Challenger		$\mathcal{A}_{remote}$
chooses $k \xleftarrow{\$} \mathcal{K}$		
computes $P \xleftarrow{\$} Comp(k)$		
$\lvert P \rvert = T$		
	$\xleftarrow{x_1, x_2, \ldots, x_q}$	
computes $y_i = P_j^r(x_i)$		
	$\xrightarrow{y_1, y_2, \ldots, y_q}$	
$m^{ch} \xleftarrow{\$} \mathcal{M}$		
	$\xrightarrow{m^{ch}}$	
	$\xleftarrow{c}$	
check $c = P(m^{ch})$ or not		

Adaptive Chosen Space (ACS) Attack

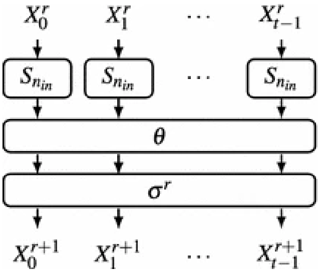
Challenger		$\mathcal{A}_{remote}$
chooses $k \xleftarrow{\$} \mathcal{K}$		
computes $P \xleftarrow{\$} Comp(k)$		
$\lvert P \rvert = T$	$\xleftarrow{\quad x_i \quad}$	
computes $y_i = P_j^r(x_i)$	$\xrightarrow{\quad y_i \quad}$	(q times adaptively)
$m^{ch} \xleftarrow{\$} \mathcal{M}$		
	$\xrightarrow{\quad m^{ch} \quad}$	
	$\xleftarrow{\quad c \quad}$	
check $c = P(m^{ch})$ or not		

Strong Space-hardness Under ACS

- Not possible to achieve
- Adversary chooses M, simply adaptively queries the table for the table invocations and compute C.
- Adversary outputs (M, C)

Potential open problem: Identify a Space-hardness notion between Weak Space-hardness and Strong Space-hardness, and design of a white-box cipher secure under the notion

An Efficient Space-hard Construction: SPNBox [BIT16]



- $S_{n_{in}}$ is (AES key addition + Sbox) 64-times (why?)
- $S_{n_{in}}$ is a block cipher with high key extraction security
- Key extraction security of SPNbox reduces to key extraction security of $S_{n_{in}}$
- Three choices of n_{in} : $8, 16$, and 32

Space-hardness of SPNBox

- Let $S_{n_{in}}$ is implemented by a table $T_{n_{in}}$
- $|T_{n_{in}}| = T$ in bits
- Assume $T/4$ table bits are leaked
- To compute C for an arbitrary P (say known and chosen space setting)
  - Total s Sboxes are invoked
  - Each Sbox can be computed with a probability $1/4$
  - Total Space-Hardness probability is $(1/4)^s$

SPNBox is not One way secure w.r.t local adversary

Fourth Attempt: Weak and Strong Incompressibility by [FKKM16]

- Fouque et al. proposed weak and strong incompressibility
- Provably secure (weak model) incompressible scheme: White-block (Invertible)
- Provably secure (strong model) incompressible scheme: White-key (Non-invertible)
- Table based construction (table is viewed as a PRF)
- **Weak incompressibility** Similar to space-hardness [BI15] and weak white-box [BBK14]
- **Strong incompressibility:** To distinguish the output of the encryption

Weak and Strong Incompressibility

Weak Incompressibility

Challenger		$\mathcal{A}_{remotey}$
chooses $T \xleftarrow{\mathcal{P}} \mathcal{T}$		
	$\xleftarrow{q_i}$	
	$\xrightarrow{T(q_i)}$	s times
$P \xleftarrow{\$} \mathcal{P}$	$\xrightarrow{P}$	
	$\xleftarrow{C}$	
check $C = E_T(P)$ or not		

Strong Incompressibility

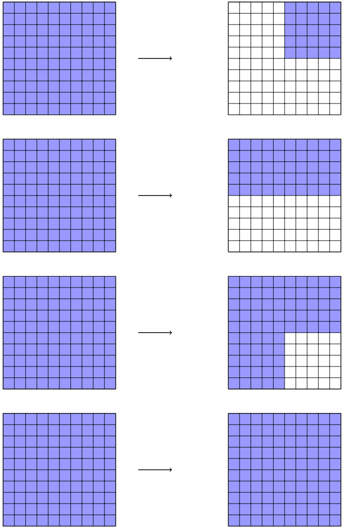
Challenger		$\mathcal{A}_{remote}$
		chooses set $\mathcal{S}$ and compression algorithm $f : \mathcal{T} \rightarrow \mathcal{S}$
chooses $T \xleftarrow{\mathcal{P}} \mathcal{T}$	$\xleftarrow{f}$	
	$\xrightarrow{f(T)}$	
	$\xleftarrow{m_i}$	
computes $c_i = E_T(m_i)$	$\xrightarrow{c_i}$	s times
	$\xleftarrow{m_0, m_1}$	
$b \xleftarrow{\$} \{0, 1\}$	$\xrightarrow{E_T(m_b)}$	
	$\xleftarrow{b'}$	
check $b' = b$ or not		

Weak Incompressibility is Similar to Weak Space-Hardness

- A scheme is (s, λ, δ)-weakly incompressible iff any adversary allowed to adaptively query up to s entries of the table T can only correctly encrypt up to a proportion δ of plaintexts (except with negligible probability $2^{-\lambda}$ over the choice of T)
- (s, λ, δ)-weak incompressibility matches exactly with $(s, -log(\delta))$-space-hardness

A New Stronger Notion of Incompressibility: Longevity [KI21]

Idea



Longevity

- Continuous leakage of the code
- Incompressibility under continuous leakage

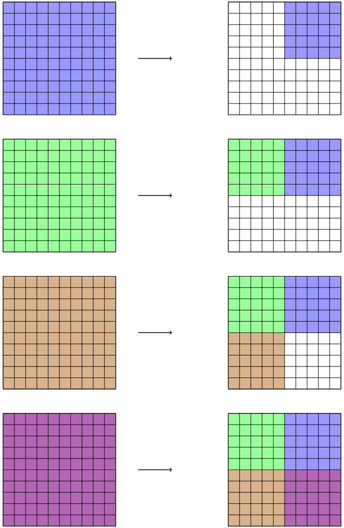
z-longevity

A cryptographic scheme has z-longevity if it is computationally difficult to encrypt (decrypt) any randomly chosen plaintext (ciphertext) with probability not more than 2^{-z} where the functionality remains constant, and code (table) is continuously leaked to the adversary

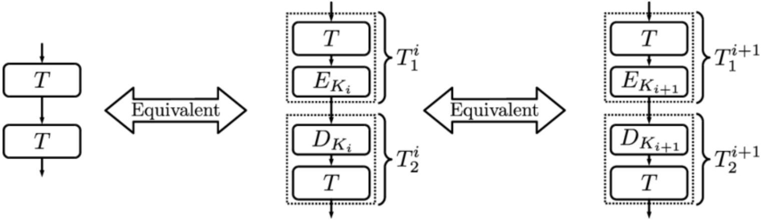
- Proposed a white-box secure construction Yoroi achieving longevity
- Used table update keeping same functionality

This notion needs to be redefined

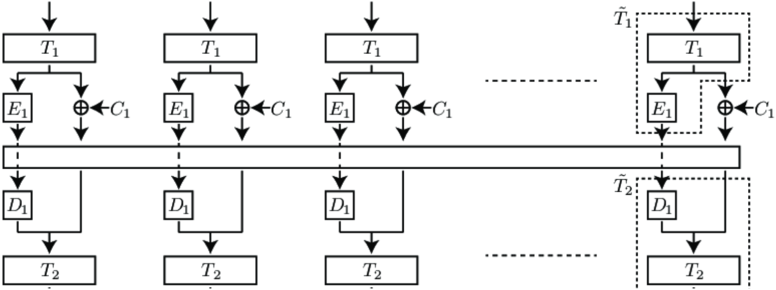
Design Idea



YOROI


T
T
Equivalent
T
E_{K_i}
T_1^i
D_{K_i}
T
T_2^i
Equivalent
T
$E_{K_{i+1}}$
T_1^{i+1}
$D_{K_{i+1}}$
T
T_2^{i+1}

YOROI



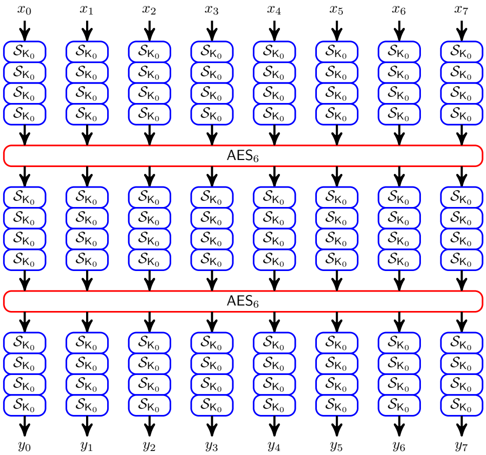
Unfortunately

- Yoroi was broken in a year [TI23]

- Tables from different updates are not independent: Leakage from one table leaks information about leakage from the updated table

- Our very recent work [CGIK25] on designing EWEMrI (with Shibam Ghosh, Takanori Isobe, and Sajani Kundu) achieves Lonegvity but assuming adversary can only leak

EWEMrl



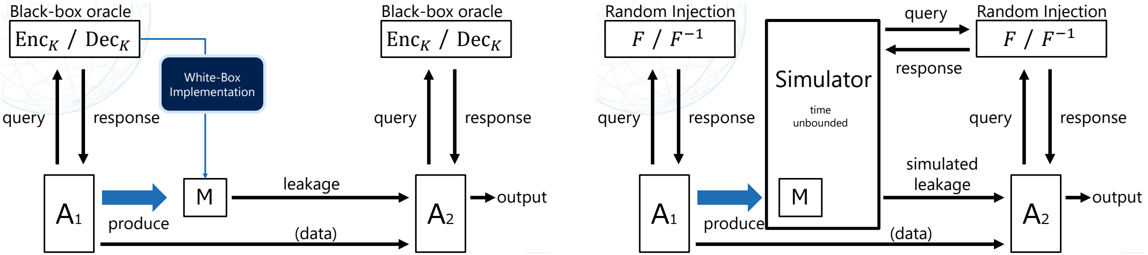
x_0 x_1 x_2 x_3 x_4 x_5 x_6 x_7
$\mathcal{S}_{K_0}$
AES$_6$
AES$_6$
y_0 y_1 y_2 y_3 y_4 y_5 y_6 y_7

ASIACRYPT 2022 Paper [HIT22]

White-Box Security Formalization

- First WB security notions considering two stage adversaries,
- First notion addressing WB security of modes
- A weak variant of public indifferentiability implies reduction,(informally, primitive is white-box secure + the idealized mode weak-public indifferentiable implies the mode is white-box secure)
- White-box security analysis of SIV-CTR AEAD

Real and Ideal World



wh-PRP

Ch.

chooses $K \xleftarrow{\$} \mathcal{K}$
computes $P \xleftarrow{\$} Comp(K)$
chooses $b \xleftarrow{\$} \{0,1\}$
If $b = 1$, $P = E_K, P^{-1} = D_K$
Else, choose a random permutation P

$\mathcal{A}(\mathcal{A}_{create}, \mathcal{A}_{dist})$

(optional)

$\xleftarrow{M_i}$

$\xrightarrow{C_i = P(M_i)}$

$\mathcal{L}, St \leftarrow \mathcal{A}_{create}^{P(\cdot)}(\cdot)$

Lifter $\mathcal{L}(P)$/ simulator $\mathcal{S}^{P,P^{-1}}(.)$ leaks data L

$\xrightarrow{L}$

(optional)

$\xleftarrow{M'_i}$

$\xrightarrow{C'_i = P(M'_i)}$

$\xleftarrow{b'}$

$b' \leftarrow \mathcal{A}_{dist}^{P(\cdot)}(L, St)$

Check $b' = b$ or not

Why not Wh-AEAD?

- $\mathcal{A}(\mathcal{A}_{create}, \mathcal{A}_{dist})$ never queries (N, A, M)
- $\mathcal{A}_{create}$ creates $\mathcal{L}$ that leaks (C, T) for (N, A, M)
- $\mathcal{A}_{dist}$ makes a decryption query (N, A, C, T)
- In the Ideal world, always *Reject*
- Hence define *Wh-PRI*

wh-PRI

Ch. | $\mathcal{A}(\mathcal{A}_{create}, \mathcal{A}_{dist})$

chooses $K \xleftarrow{\$} \mathcal{K}$
computes $\mathcal{P} \xleftarrow{\$} Comp(K)$
chooses $b \xleftarrow{\$} \{0,1\}$
If $b = 1$, $f = E_K, f^{-1} = D_K$
Else, choose a random injection f

(optional)

$\xleftarrow{M_i}$

$\xrightarrow{C_i = f(M_i)}$

$\mathcal{L}, St \leftarrow \mathcal{A}_{create}^{f(\cdot), f^{-1}(\cdot)}(\cdot)$

Lifter $\mathcal{L}(\mathcal{P})$/ simulator $\mathcal{S}^{f,f^{-1}}(.)$ leaks data L

$\xrightarrow{L}$

(optional)

$\xleftarrow{M_i'}$

$\xrightarrow{C_i' = f(M_i')}$

$b' \leftarrow \mathcal{A}_{dist}^{f(\cdot), f^{-1}(\cdot)}(L, St)$

Check $b' = b$ or not $\xleftarrow{b'}$

References

- [*CEJO*01] Stanley Chow, Philip A. Eisen, Harold Johnson, Paul C. van Oorschot: White-Box Cryptography and an AES Implementation. Selected Areas in Cryptography 2002: 250-270
- [*BGE*04] Olivier Billet, Henri Gilbert, Charaf Ech-Chatbi: Cryptanalysis of a White Box AES Implementation. Selected Areas in Cryptography 2004
- [*DLPR*13]: Cecile Delerablee, Tancrede Lepoint, Pascal Paillier, Matthieu Rivain: White-Box Security Notions for Symmetric Encryption Schemes. Selected Areas in Cryptography 2013: 247-264
- [*BBK*14] Alex Biryukov, Charles Bouillaguet, Dmitry Khovratovich: Cryptographic Schemes Based on the ASASA Structure: Black-box, White-box, and Public-key. IACR Cryptol. ePrint Arch. 2014: 474
- [*BI*15] Andrey Bogdanov, Takanori Isobe: White-Box Cryptography Revisited: Space-Hard Ciphers. CCS 2015: 1058-1069

References

- [*FKKM*16] Pierre-Alain Fouque, Pierre Karpman, Paul Kirchner, Brice Minaud: Efficient and Provable White-Box Primitives. ASIACRYPT (1) 2016: 159-188
- [*BIT*16] Andrey Bogdanov, Takanori Isobe, Elmar Tischhauser: Towards Practical Whitebox Cryptography: Optimizing Efficiency and Space Hardness. ASIACRYPT (1) 2016: 126-158
- [*KI*21] Yuji Koike, Takanori Isobe: Yoroi: Updatable Whitebox Cryptography. IACR Trans. Cryptogr. Hardw. Embed. Syst. 2021(4): 587-617
- [*TI*23] Yosuke Todo, Takanori Isobe: Hybrid Code Lifting on Space-Hard Block Ciphers Application to Yoroi and SPNbox. IACR Trans. Symmetric Cryptol. 2022(3): 368-402
- [*HIT*22] Akinori Hosoyamada, Takanori Isobe, Yosuke Todo, Kan Yasuda: A Modular Approach to the Incompressibility of Block-Cipher-Based AEADs. ASIACRYPT (2) 2022
- [*CGIK*25] Avik Chakraborti, Shibam Ghosh, Takanori Isobe, Sajani Kundu: EWEMrl: A White-Box Secure Cipher with Longevity. IACR Cryptol. ePrint Arch. 2025: 1221

thank you!