Panel Discussion
Lessons Learned

Panel Moderator: John Kelsey, *NIST*

Panelists:

Lily Chen, *NIST*

Joan Daemen, *Radboud University*

Phillip Rogaway, *University of California, Davis*

Miles Smid, *Retired (NIST)*

Lessons Learned

—A journey of embracing challenges

Lily Chen
Computer Security Division, Information Technology Lab
National Institute of Standards and Technology (NIST)

Challenges: Past, Present, and Future

- Deal with extremes
  - Extremely powerful attack technologies, e.g. using quantum computers
  - Extremely constrained implementation environment, e.g. sensors
- Transition, forward secrecy, and backward compatibility
  - Increased key sizes, stronger hash functions, block ciphers
  - Post-quantum cryptography
- Extended security objectives and features
  - Deal with more sophisticated cryptanalysis methods, e.g. side-channel attacks, multiple-key/target attacks, etc.
  - Demand useability features, e.g. misuse resistance
- Special usage vs. general purpose standards
  - Some standards are developed for special usage, e.g. lightweight cryptography
- Synchronize with industry best practice and promote international adoption
  - Organizations tend to create standards divergent from existing ones

Perspectives on standardization

Joan Daemen

Radboud University

3rd NIST Workshop on Block Cipher Modes of Operation
Rockville, USA, October 3-4, 2023

Block cipher based crypto

Two-layer approach:

1. Build an n-bit block cipher B_K
   - goal: B_K behaves like a random n-bit permutation
   - (S)PRP distinguishing advantage $\epsilon_p(M, N)$ assumed to be small
   - assurance: based on public scrutiny by cryptanalysts
2. Build a mode of a random permutation
   - prove upper bound $\epsilon_m(M, N)$ for probability of breaking it (may be tricky)

Security of mode of concrete block cipher

Breaking probability $\leq \epsilon_m(M, N) + \epsilon_p(M, N)$

Block cipher based crypto

Two-layer approach:

1. Build an n-bit block cipher B_K
   - goal: B_K behaves like a random n-bit permutation
   - (S)PRP distinguishing advantage $\epsilon_p(M, N)$ assumed to be small
   - assurance: based on public scrutiny by cryptanalysts
2. Build a mode of a random permutation
   - prove upper bound $\epsilon_m(M, N)$ for probability of breaking it (may be tricky)

Security of mode of concrete block cipher

Breaking probability $\leq \epsilon_m(M, N) + \epsilon_p(M, N)$

Block cipher based crypto

Two-layer approach:

1. Build an n-bit block cipher B_K
   - goal: B_K behaves like a random n-bit permutation
   - (S)PRP distinguishing advantage $\epsilon_p(M, N)$ assumed to be small
   - assurance: based on public scrutiny by cryptanalysts
2. Build a mode of a random permutation
   - prove upper bound $\epsilon_m(M, N)$ for probability of breaking it (may be tricky)

Security of mode of concrete block cipher

Breaking probability $\leq \epsilon_m(M, N) + \epsilon_p(M, N)$

Block cipher based crypto

Two-layer approach:

1. Build an n-bit block cipher B_K
   - goal: B_K behaves like a random n-bit permutation
   - (S)PRP distinguishing advantage $\epsilon_p(M, N)$ assumed to be small
   - assurance: based on public scrutiny by cryptanalysts
2. Build a mode of a random permutation
   - prove upper bound $\epsilon_m(M, N)$ for probability of breaking it (may be tricky)

Security of mode of concrete block cipher

Breaking probability $\leq \epsilon_m(M, N) + \epsilon_p(M, N)$

Block cipher based crypto

Two-layer approach:

1. Build an n-bit block cipher B_K
   - goal: B_K behaves like a random n-bit permutation
   - (S)PRP distinguishing advantage $\epsilon_p(M, N)$ assumed to be small
   - assurance: based on public scrutiny by cryptanalysts
2. Build a mode of a random permutation
   - prove upper bound $\epsilon_m(M, N)$ for probability of breaking it (may be tricky)

Security of mode of concrete block cipher

Breaking probability $\leq \epsilon_m(M, N) + \epsilon_p(M, N)$

Block cipher based crypto

Two-layer approach:

1. Build an n-bit block cipher B_K
   - goal: B_K behaves like a random n-bit permutation
   - (S)PRP distinguishing advantage $\epsilon_p(M, N)$ assumed to be small
   - assurance: based on public scrutiny by cryptanalysts
2. Build a mode of a random permutation
   - prove upper bound $\epsilon_m(M, N)$ for probability of breaking it (may be tricky)

Security of mode of concrete block cipher

Breaking probability $\leq \epsilon_m(M, N) + \epsilon_p(M, N)$

Block cipher based crypto

Two-layer approach:

1. Build an n-bit block cipher B_K
   - goal: B_K behaves like a random n-bit permutation
   - (S)PRP distinguishing advantage $\epsilon_p(M, N)$ assumed to be small
   - assurance: based on public scrutiny by cryptanalysts
2. Build a mode of a random permutation
   - prove upper bound $\epsilon_m(M, N)$ for probability of breaking it (may be tricky)

Security of mode of concrete block cipher

Breaking probability $\leq \epsilon_m(M, N) + \epsilon_p(M, N)$

Block cipher based crypto

Two-layer approach:

1. Build an n-bit block cipher B_K
   - goal: B_K behaves like a random n-bit permutation
   - (S)PRP distinguishing advantage $\epsilon_p(M, N)$ assumed to be small
   - assurance: based on public scrutiny by cryptanalysts
2. Build a mode of a random permutation
   - prove upper bound $\epsilon_m(M, N)$ for probability of breaking it (may be tricky)

Security of mode of concrete block cipher

Breaking probability $\leq \epsilon_m(M, N) + \epsilon_p(M, N)$

Permutation-based crypto

Three-layer approach:

1. Build permutation f
2. Construct a deck function F_K or keyed duplex F_K on top of it
   - variable-length input and output and incrementality
   - F_K should have small $\epsilon_P(M, N)$ from random oracle $\mathcal{RO}$
   - assurance: based on public scrutiny by cryptanalysts
3. Build a mode of $\mathcal{RO}$: proving $\epsilon_m(M, N)$ is often simple

Security of mode of F_K with concrete permutation

Breaking probability $\leq \epsilon_m(M, N) + \epsilon_p(M, N)$

Permutation-based crypto

Three-layer approach:

1. Build permutation f
2. Construct a deck function F_K or keyed duplex F_K on top of it
   - variable-length input and output and incrementality
   - F_K should have small $\epsilon_P(M, N)$ from random oracle $\mathcal{RO}$
   - assurance: based on public scrutiny by cryptanalysts
3. Build a mode of $\mathcal{RO}$: proving $\epsilon_m(M, N)$ is often simple

Security of mode of F_K with concrete permutation

Breaking probability $\leq \epsilon_m(M, N) + \epsilon_p(M, N)$

Permutation-based crypto

Three-layer approach:

1. Build permutation f
2. Construct a deck function F_K or keyed duplex F_K on top of it
   - variable-length input and output and incrementality
   - F_K should have small $\epsilon_P(M, N)$ from random oracle $\mathcal{RO}$
   - assurance: based on public scrutiny by cryptanalysts
3. Build a mode of $\mathcal{RO}$: proving $\epsilon_m(M, N)$ is often simple

Security of mode of F_K with concrete permutation

Breaking probability $\leq \epsilon_m(M, N) + \epsilon_p(M, N)$

Permutation-based crypto

Three-layer approach:

1. Build permutation f
2. Construct a deck function F_K or keyed duplex F_K on top of it
   - variable-length input and output and incrementality
   - F_K should have small $\epsilon_P(M, N)$ from random oracle $\mathcal{RO}$
   - assurance: based on public scrutiny by cryptanalysts
3. Build a mode of $\mathcal{RO}$: proving $\epsilon_m(M, N)$ is often simple

Security of mode of F_K with concrete permutation

Breaking probability $\leq \epsilon_m(M, N) + \epsilon_p(M, N)$

Permutation-based crypto

Three-layer approach:

1. Build permutation f
2. Construct a deck function F_K or keyed duplex F_K on top of it
   - variable-length input and output and incrementality
   - F_K should have small $\epsilon_P(M, N)$ from random oracle $\mathcal{RO}$
   - assurance: based on public scrutiny by cryptanalysts
3. Build a mode of $\mathcal{RO}$: proving $\epsilon_m(M, N)$ is often simple

Security of mode of F_K with concrete permutation

Breaking probability $\leq \epsilon_m(M, N) + \epsilon_p(M, N)$

Permutation-based crypto

Three-layer approach:

1. Build permutation f
2. Construct a deck function F_K or keyed duplex F_K on top of it
   - variable-length input and output and incrementality
   - F_K should have small $\epsilon_P(M, N)$ from random oracle $\mathcal{RO}$
   - assurance: based on public scrutiny by cryptanalysts
3. Build a mode of $\mathcal{RO}$: proving $\epsilon_m(M, N)$ is often simple

Security of mode of F_K with concrete permutation

Breaking probability $\leq \epsilon_m(M, N) + \epsilon_p(M, N)$

Permutation-based crypto

Three-layer approach:

1. Build permutation f
2. Construct a deck function F_K or keyed duplex F_K on top of it
   - variable-length input and output and incrementality
   - F_K should have small $\epsilon_P(M, N)$ from random oracle $\mathcal{RO}$
   - assurance: based on public scrutiny by cryptanalysts
3. Build a mode of $\mathcal{RO}$: proving $\epsilon_m(M, N)$ is often simple

Security of mode of F_K with concrete permutation

Breaking probability $\leq \epsilon_m(M, N) + \epsilon_p(M, N)$

Permutation-based crypto

Three-layer approach:

1. Build permutation f
2. Construct a deck function F_K or keyed duplex F_K on top of it
   - variable-length input and output and incrementality
   - F_K should have small $\epsilon_P(M, N)$ from random oracle $\mathcal{RO}$
   - assurance: based on public scrutiny by cryptanalysts
3. Build a mode of $\mathcal{RO}$: proving $\epsilon_m(M, N)$ is often simple

Security of mode of F_K with concrete permutation

Breaking probability $\leq \epsilon_m(M, N) + \epsilon_p(M, N)$

Panel discussion:

A few comments

Phillip Rogaway
University of California, Davis, USA

1) What <u>is</u> a blockcipher mode of operation?

A cryptographic scheme Π = **MODE**$[E, p]$ that depends on an arbitrary blockcipher
E: $\{0,1\}^k \times \{0,1\}^n \rightarrow \{0,1\}^n$
and possibly other parameters p.

- Hardness should only come from E
- Best if p is absent or limited to a single number.

2) How many modes has NIST standardized in its 800-38 schemes?

1. ECB	5. CBC-CS1	9. CMAC[t]	13. XTS	17. FF3
2. CBC	6. CBC-CS2	10. CCM[Ctr, Fmt, t]	14. KW	
3. CFB[s]	7. CBC-CS3	11. GCM[t]	15. KWP	
4. OFB	8. CTR[Inc]	12. GMAC[t]	16 FF1	

3) Is it clear what each mode is supposed to do? Is that definition adequately strong?

1.	ECB	wide-block BC	IND$ if rand msg. Very weak
2.	CBC	IV-based sym enc	IND$ if rand IV. Weak
3.	CFB[*s*]	IV-based sym enc	IND$ if rand IV. Weak
4.	OFB	IV-based sym enc	IND$ if rand IV. Weak
5.	CBC-CS1	IV-based sym enc	IND$ if rand IV. Weak
6.	CBC-CS2	IV-based sym enc	IND$ if rand IV. Weak
7.	CBC-CS3	IV-based sym enc	IND$ if rand IV. Weak
8.	CTR[Inc]	nonce-based sym enc	IND$ if componentwise-nonce
9.	CMAC[t]	PRF (so a MAC)	
10.	CCM[Ctr, Fmt, t]	AEAD	Requirements on algorithm Ctr, For?
11.	GCM[t]	AEAD	
12.	GMAC[t]	MAC-like: conventional MACS have no nonce	
13.	XTS	wide-block, one-query/tweak TBC: XTS(K, i, X).	
14.	KW	pseudorandom injection (PRI): KW(K, P)	
15.	KWP	pseudorandom injection (PRI) : KWP(K, P)	
16.	FF1	format-preserving encryption (FPE)	
17.	FF3	format-preserving encryption (FPE)	

4) What do you think about this list?

It's too long

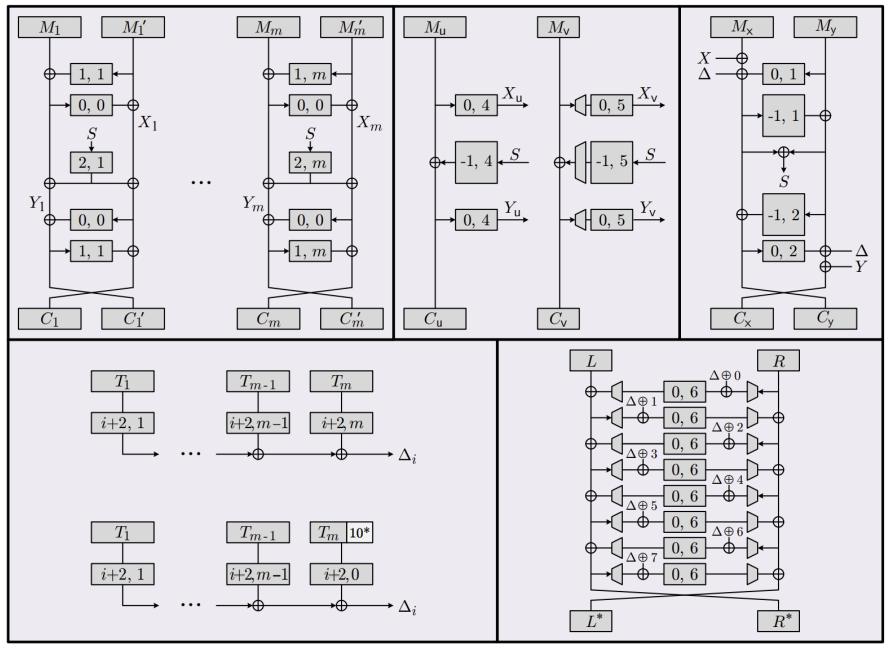
5) Is there any natural way to shorten it?

One mode to rule them all….

5, cont. The apparent goal of **every** NIST 800-38 scheme is subsumed by

ENC[t, radix]: Key × Tweak × Σ^* → Σ^*

where ENC(K,T, ·) is injective for all K, T
and | ENC(K,T, X) | = |X| + t
and Σ = {0, 1, …, radix-1};
with the natural & strong PRI-security definition (at a minimum). Can select Key as small as $\{0,1\}^k$ or as large as $\{0,1\}^*$, and Tweak as small as {unused} or as large as $\{0,1\}^{**}$.



6. Can such an object be practical?

Probably. AEZ 4.1 (radix=2) is too complex, but it was efficient, certainly practical in SW. It did limit to radix = 2.

Lessons Learned

Miles Smid

- Cryptographic security has made significant advances
- But so has cryptanalysis
  - Attacks on the base algorithm
  - Attacks on the mode of operation
  - Attacks exploiting weaknesses in the cryptographic module (FIPS 140-n)
  - Attacks on the application or system procedures
  - Attacks on the real life usage
  - Combination of above

Lessons Learned

- Developing a strong crypto standard is hard work. *(J. Kelsey)*
- *Good enough today may not be good enough in the future (security strength vs. security life)*
- *Crypto is not always used as intended*
- The number of different applications of a good crypto is always underestimated

Where Do We Go from Here?

- A standard crypto is best developed using a consensus process involving vendors, users, and crypto experts. *(NIST, ANSI,IEEE)*
- Our Knowledge Base keeps Growing so build for growth
- Need more work on crypto modules (FIPS 140-n), crypto applications and systems
- Exploit the Potential of AI?
  - New and better crypto algorithms (neural networks?)
  - Continual automated analysis of crypto algorithm security
  - Managing the usage and security of the crypto module
  - Monitoring the current knowledge base and providing estimates of current security strength and anticipated security life
  - Provide security warnings and shut down at critical conditions