Challenges in Lightweight Crypto Standardization

Meltem Sönmez Turan

Fast Software Encryption 2015
March 9, 2015
about me

- Doing (symmetric) crypto research for 10+ years.
- Guest researcher at NIST for 5+ years
- Participated in the SHA-3 project, password-based KDFs project, stream cipher, RNG project (SP800 90B), etc.
Outline

• Lightweight crypto project at NIST
• Overview of the academic literature
• Overview of the standardization efforts
• Challenges in standardization
- Measurement science lab.
- Part of the US Department of Commerce
- Located at Gaithersburg, Maryland
- Founded in 1901, known as the National Bureau of Standards (NBS) prior to 1988
- Around 2700 employees, and 1,800 associates.

**NIST’s mission**

to develop and promote measurement, standards, and technology to enhance productivity, facilitate trade and improve the quality of life.
Laboratory programs

- Center for Nanoscale Science and Technology
  - Communications Technology
  - Engineering
  - Information Technology
- Advanced Network Technologies
  - Applied & Computational Mathematics
  - Information Access
  - Computer Security
  - Statistical Engineering
- Material Measurement
- Center for Neutron Research
- Physical Measurement
What do we do?

• **Algorithm specifications:**
  – Federal Information Processing Standards (FIPS) and Special Publications (SPs) specify a number of approved cryptographic algorithms.

• **General guidance on the use of cryptography:**
  – Covering selection, implementation, deployment and use of cryptography.

• **Guidelines in application-specific areas:**
  – Areas of particular need for the US government (e.g., PIV, TLS).

• **Testing:**
  – Providing assurance that crypto is implemented properly (e.g., FIPS 140 and CMVP)
Who do we work with?

• **Academic Researchers:**
  – Development of new algorithms/modes/schemes, to advance science of cryptography

• **Industry:**
  – On adoption of cryptographic algorithms, feedback mechanism on standards

• **Standards Developing Organizations:**
  – Adoption and development of new standards

• **Government:**
  – Core user community
How do we develop standards?

• **International Competitions**
  – Engage community through an open competition
  – *e.g.*, AES, SHA-3

• **Adoption of Existing Standards**
  – Collaboration with accredited standards organizations
  – *e.g.*, RSA, HMAC

• **Open call for proposals**
  – Ongoing open invitation
  – *e.g.* modes of operations (SP 800 38)

• **Development of New Algorithms**
  – Used if no suitable standard exists
  – *e.g.*, DRBGs

NIST IR 7977 [NIST Cryptographic Standards and Guidelines Development Process](#)
Example Research Projects

Post quantum crypto, Pairing-based crypto, Privacy enhancing crypto, Secure group communications, Circuit complexity, *Lightweight crypto*, etc.
Lightweight Crypto Project

Cryptographic solutions tailored to constrained environments.

- Focus: Symmetric-key crypto primitives.

Not meant to be weak

Not meant to replace general-purpose crypto primitives

Our initial questions:
- Is there truly a demand?
- Is the technology mature enough to be standardized?
- Internet of Things
- Pervasive computing
- Ubiquitous computing
- Ambient intelligence
- Calm computing

THE INTERNET OF THINGS
AN EXPLOSION OF CONNECTED POSSIBILITY

The demand

• Applications
  – Healthcare monitoring systems
  – Automated management of supply chain
  – Public transportation
  – Telephone cards, etc.

• Involve sensitive information

• Constrained devices with limited memory, power supply, etc.

• NIST-approved crypto algorithms may not be suitable.
Is the technology mature enough to be standardized?
Academic Research

• Significant academic interest
  – Around 1400 papers on *lightweight cryptography* in the last 10 years (according to Google Scholar)

• Dedicated academic workshops
  – e.g. Lightsec, RFIDsec, Lightweight Crypto Day, Four workshops sponsored by the ECRYPT project, etc.
What has been done? – Symmetric Crypto

Improved implementations of AES

- In HW, 2400 GEs (Moradi et al., Eurocrypt 11), 2090-gate design (Mathew et al, 2014)
- In SW, using 8-bit AVR microcontrollers, 124.6 and 181.3 cpb for encryption/decryption with a code size < 2 Kbyte (Osvik et al., FSE10).

AES should be used whenever possible!
What has been done?

- ** Modifications of well-analyzed algorithms**
  - e.g. DESL, DESXL
- **Old interesting algorithms**
  - e.g. RC5, TEA, XTEA
- **New dedicated algorithms.**
  - e.g. CLEFIA, Fantomas, HIGHT, ICEBERG, KASUMI, LBlock, LED, KATAN/KTANTAN, Klein, mCrypton, MIBS, NOEKEON, Piccolo, PRESENT, PRINTcipher, PUFFIN, PUFFIN2, PRINCE, PRIDE, SEA, SIMON, SPECK, TWIS, TWINE ...
Characteristics of new designs

• Many iterations of simple rounds
• Simple operations like XORs, rotation, 4X4 Sboxes, bit permutations
• Smaller block sizes
• Smaller key sizes
• Simpler key schedules
• Small security margins by design
  – Many designs, but many were broken in a short time
Different threat models

Different capabilities of attackers

- Limited number of known plaintexts/ciphertexts
- Less concern on related key attacks. From ideal cipher to ideal permutation assumption.

Justifications:

- Limitations of the devices (e.g. battery life)
- Protection through the protocols
Different threat models

Different capabilities of attackers
- Limited number of known plaintexts/ciphertexts
- Less concern on related key attacks. From *ideal cipher* to *ideal permutation* assumption.

Justifications:
- Limitations of the devices (e.g. battery life)
- Protection through the protocols

*Example: Prince*

- Claims $126 - n$ bit security for an attacker with access to an $2^n$ input/output pairs.
- Decryption for free = encryption with a related key.
Side channel attacks

Serious threat for constrained devices
- Attacker may have physical access.
- Devices are cheaper.

With countermeasures, the area increases by a factor of 3 to 5 compared to the non-protected implementations (Fisher, Gammel, ’05)

New designs with side-channel resistance:
- Fides, LS family, PICARO
ISO/IEC - 29192

- **Part I:** General, First edition, 2012
- **Part II:** Block ciphers, 2012
  - 64-bit PRESENT (80, 128 bit key)
  - 128-bit CLEFIA (128, 192 or 256-bit key)
- **Part III:** Stream ciphers, 2012
  - Enocoro (80, 128 bits key)
  - Trivium (80 bit key)
- **Part IV:** Mechanisms using asymmetric techniques, 2013
  - Identification scheme cryptoGPS
  - Authentication and key exchange mechanism ALIKE
  - ID-based signature scheme IBS
- **Part V:** Hash functions - not published.
ISO/IEC - 29167

• A number of cryptographic suites designed for protecting application information transmitted across the RFID air interface, product authentication, and protecting access to resources on the tag.

• 10 Parts

• Algorithms :
  – PRESENT-80, ECC-DH, Grain-128A, AES OFB, Crypto suite XOR, ECDSA-ECDH, cryptoGPS, RAMON
Industry-specific standards

• Proprietary designs

• Examples:
  – A5/1 (in GSM), E0 (in Bluetooth), Crypto1 (in Mifare RFID tags), Cryptomeria (C2) (for digital rights managements), Dect (cordless phones), DST40 (TI), KeeLoq (authentication in car locks), Kindle stream cipher

• Most reversed engineered, practically broken.
ECRYPT eSTREAM Project

a 4-year network of excellence funded project started in 2004 by European Network of Excellence for Cryptology (ECRYPT)

Goal: To identify new stream ciphers that might be suitable for widespread adoption and to stimulate work in stream ciphers.

Profile I: for software applications with high throughput requirements with key size of 256 bits.

Profile II: for hardware applications with restricted resources with key size of 80 bits.
Finalists of Profile II

• Grain
  – Widely analyzed
  – Tweaked twice
  – A new version Grain128a, featuring authentication
  – Flexible

• Trivium
  – Widely analyzed
  – Not tweaked, simple and elegant,
  – Flexible

• Mickey
  – Lightly analyzed, security depends on the hardness of analysis.
  – Less implementation flexibility, due to irregular clocking
  – Susceptible to timing and power analysis attacks
Lightweight versions of KECCAK

• In 2012, KECCAK was selected as SHA-3.
  – Instantiation of a sponge function
  – Permutation based, with seven different sizes \{25, 50, 100, 200, 400, 800, 1600\}.
  – Design of permutations follows the Matryoshka principle.

• Lightweight instance:
  – 200-bit permutation with, \(r=40, c=160\), 12 rounds.
  – Security strength of 80 bits.
  – Offers tradeoffs
  – Reusing permutation for AE, hashing, etc.
  – Crunchy contest (practical attacks):
    • Preimage attacks up to 2 rounds, collision attacks up to 4 rounds.
Lightweight versions of KECCAK (cont.)

• Performance on constrained environments.
  – 9.3kGE on a 130 nm CMOS process technology, by designers
  – Kavun & Yalcin implemented 200, 400, 800 and 1600 versions with
    2.52kGE, 5.09kGE, 13kGE and 20.79kGE, respectively.
  – Pessle & Hutter showed that 1600-bit version can be implemented
    with less than 5.5kGEs.
    • Low, but acceptable, throughput
    • 800-bit with 4.6kGE. (900GE less than full permutation and
      twice as fast.)
    • Don’t include side channel resistance.

More research is needed for lightweight uses of KECCAK.
Challenges
Bridging the Gap

- Industry needs vs. Academic solutions
  - Various applications with different requirements, use cases, constraints, target devices, etc.

- Communicating with industry to bridge the gap.
  - Workshop, and other meetings.
**Enforcing the threat model**

- Less flexible, less misuse resistant, more constraints, assumptions about attackers.

- Challenge to enforce the limitations
  - # of known/chosen plaintext/ciphertext blocks
  - Uniqueness of the IVs (e.g. AES GCM)

- Development of the protocols is important.
  - Non-cryptographic protocols, Message formats
### Selection of Key Size

Tradeoffs – Smaller key sizes to reduce cost

According to NIST SP 800-57:

<table>
<thead>
<tr>
<th>Security Strength</th>
<th>2011 through 2013</th>
<th>2014 through 2030</th>
<th>2031 and Beyond</th>
</tr>
</thead>
<tbody>
<tr>
<td>80</td>
<td>Applying</td>
<td>Deprecated</td>
<td>Disallowed</td>
</tr>
<tr>
<td></td>
<td>Processing</td>
<td></td>
<td>Legacy use</td>
</tr>
<tr>
<td>112</td>
<td>Applying</td>
<td>Acceptable</td>
<td>Acceptable</td>
</tr>
<tr>
<td></td>
<td>Processing</td>
<td></td>
<td></td>
</tr>
<tr>
<td>128</td>
<td></td>
<td>Acceptable</td>
<td>Acceptable</td>
</tr>
<tr>
<td>192</td>
<td>Applying/Processing</td>
<td>Acceptable</td>
<td>Acceptable</td>
</tr>
<tr>
<td>256</td>
<td></td>
<td>Acceptable</td>
<td>Acceptable</td>
</tr>
</tbody>
</table>
Selecting a Primitive

- Due to the variability of applications/requirements,
  - Hard to select a one-size-fits-all algorithm

- Tradeoff between performance, security, cost are highly important.
  - Depends on the target technology
  - HW/SW optimized algorithms
  - Optimized for both

Figure: A. Poschmann, *Lightweight Cryptography: Cryptographic engineering for a pervasive world*
Performance Comparisons


Our Tentative Plan

• An algorithm or a portfolio of algorithms

• Possibilities
  – Adoption of existing standards
  – Open ongoing call for proposals
## Tentative Schedule

<table>
<thead>
<tr>
<th>Phase</th>
<th>Objectives</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phase I</td>
<td>- Identify and evaluate the need</td>
<td>Late 2014 to June 2015</td>
</tr>
<tr>
<td></td>
<td>- Survey latest developments</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Announce intent</td>
<td></td>
</tr>
<tr>
<td>Phase II</td>
<td>- Workshop @NIST on July 20-21, 2015</td>
<td>July - December 2015</td>
</tr>
<tr>
<td></td>
<td>- Consider requirements and solutions</td>
<td></td>
</tr>
<tr>
<td>Phase III</td>
<td>- Define specific plan</td>
<td>2016 -</td>
</tr>
<tr>
<td></td>
<td>- Develop SP (if applicable)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Maintenance</td>
<td></td>
</tr>
</tbody>
</table>
Lightweight Crypto Workshop

Dates
Location: NIST Gaithersburg, MD
Date: July 20-21, 2015
Submission: April 1, 2015
Notification: May 15, 2015

Topics
• Requirements and characteristics of real-world applications
• RFID, SCADA, cyber-physical systems, and the Internet of Things
• Case studies of deployed systems
• Evaluation of threats, attacks and risks
• Restrictions and protections to reduce the risk of using lightweight primitives
• Design, analysis and implementation
• Lightweight public key cryptography
• Benchmarking of lightweight cryptographic algorithms in software and hardware
• Side channel attacks and countermeasures for constrained devices
Thanks!

meltem.turan@nist.gov