Practical applications of Lightweight Block Ciphers to Secure EtherNet/IP Networks

Jordon Woods
Innovasic

October 14, 2015

Authors

Jordon Woods
Chief Technical Officer
Innovasic, Inc.

Patricia Muoio
Director of Research and Development
G2, Inc.

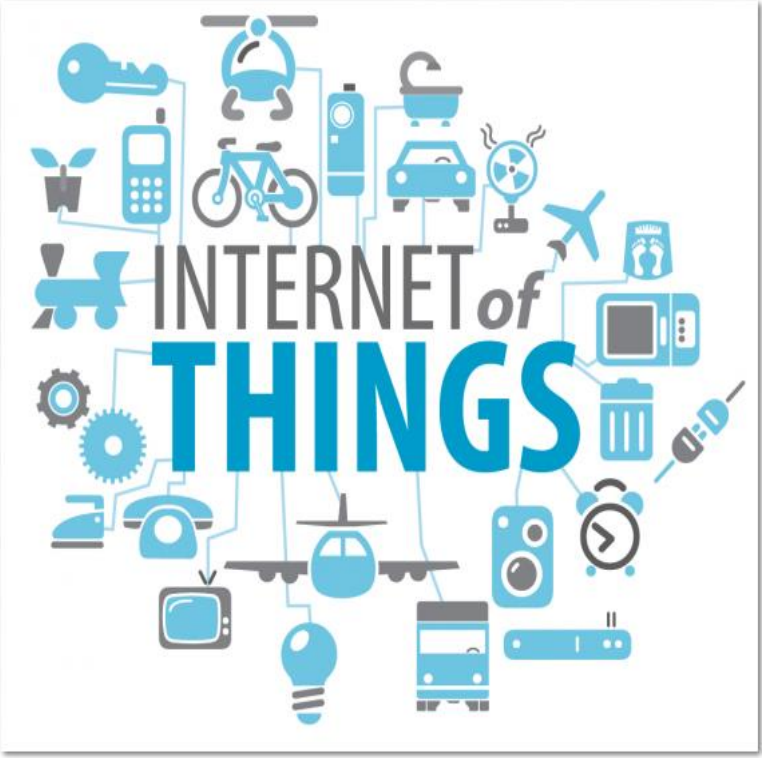
A New Era?

- 50B Internet connected devices by 2025 (IoT)
- Of that 50B, ~40% will be Industrial devices (Industry 4.0, IIoT)
  - (Source: IHS 2013, Internet Connected Devices)
- These devices are sensors, actuators, field devices…

- …used in Building Automation, Factory Automation, Process Automation, Water/Wastewater, Transportation, Smart Grid, etc.

Constraints of the IIOT

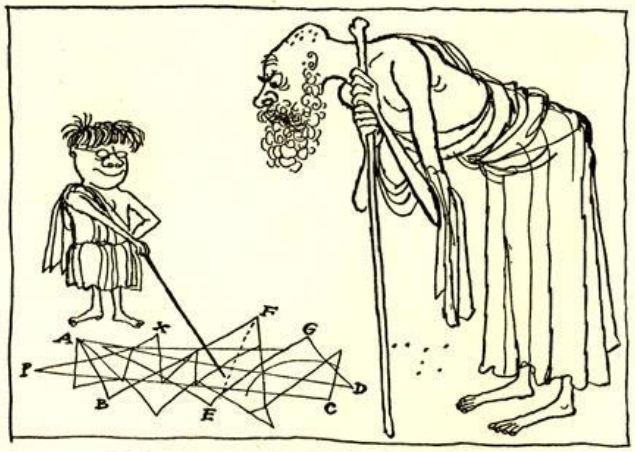
- Realities
  - Widely dispersed
  - Interconnected
  - Homogeneous networks
  - Not very secure
    - Enormous attack surface
    - Consistency across platforms
      - Hardware, SW, languages, devices, etc.
- Constraints
  - Limited processing resources
  - Limited memory
  - Cost-constrained
  - Area/footprint constrained



The Promise of Lightweight Cryptography

- *The small size and limited processing power of many connected devices could inhibit encryption and other robust security measures.*
  - Edith Ramirez, chair, US Federal Trade Commission

- Cryptographic solutions must be easy to implement and have high performance on a wide range of severely constrained devices. Cryptography should be an aid, not a hindrance, to achieving security



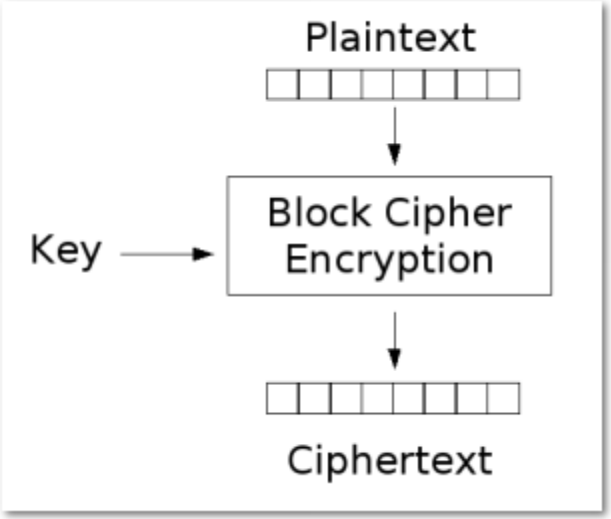
Pythagoras puzzled by one of my theorems.

Why Not AES?

- Existing cryptographic algorithms were, for the most part, designed to meet the needs of the desktop computing era
  - AES was designed specifically for environments that support a standard PC architecture
  - Power, memory and size essentially unconstrained
  - Substantial Latency/overhead
  - Over the last 15 years, a lot of effort has gone into reshaping the AES into a solution which will work in physically constrained systems
  - Still falls short for highly-constrained devices

The Promise of Lightweight Cryptography

- Lightweight cryptography lends itself to implementation as a block cipher
  - Small hardware footprint compared to comparable AES implementations
  - Scalable, pipelined architecture
  - In-line encryption/decryption
  - Comparatively low latency
  - Can be realized by small circuits with minimal power requirements
  - Provides comparable security to AES for a given key size



SIMON & SPECK

- Most Lightweight cryptography are designed for specific platforms (PRESENT, KATAN, Piccolo, etc.)
- Poor performance on other platforms can ruin overall performance

- SIMON & SPECK
  - Two families of highly flexible block ciphers.
  - High performance on ASICs, FPGAs, Microcontrollers and Microprocessors.
  - Flexible and secure
  - SIMON and SPECK are generalists

SIMON

- Versatile in hardware and software
- For pure hardware apps SIMON outperforms SPECK
- Small, fast, low energy and power. Record breaking performance on ASICs and FPGAs
- Excels on microcontrollers and microprocessors too

SPECK

- Versatile in software and hardware
- For pure software apps SPECK outperforms SIMON
- Small, fast, low energy and power. Record breaking performance on microcontrollers and microprocessors
- Excels on ASICs and FPGAs as well

SIMON

- SIMON & SPECK achieve robust encryption using repeated rounds of simple functions
- For SIMON, each round consists of a two-stage Feistel map. The Feistel map for the SIMON algorithm is given by:

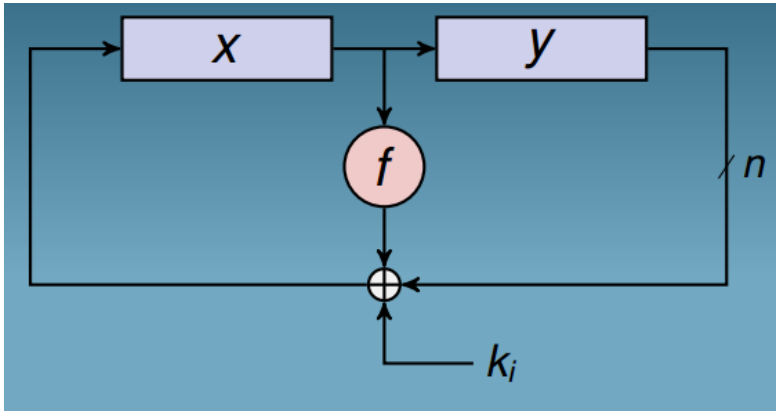
$$R_k(x, y) = (y \oplus f(x) \oplus k, x)$$

Where k is the round key and

$$f(x) = (Sx \,\&\, S^8 x) \oplus S^2 x.$$

The inverse of the round function is used for decryption:

$$R_k^{-1}(x, y) = (y, x \oplus f(y) \oplus k)$$



SPECK

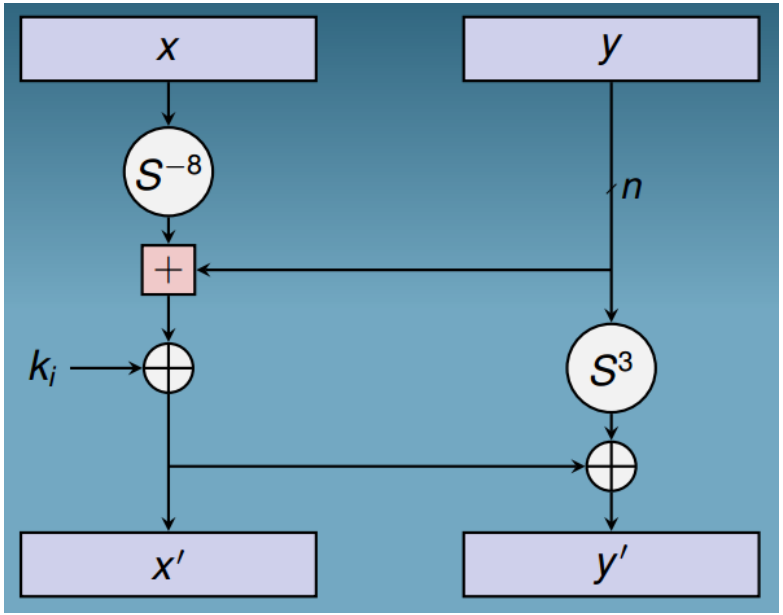
- The SPECK algorithm also utilizes a Feistel-based map:

$$R_k(x,y) = \big((S^{-\alpha}x + y) \oplus k,\ S^{\beta}y \oplus (S^{-\alpha}x + y) \oplus k\big),$$

with rotation amounts $\alpha = 7$ and $\beta = 2$, if n = 16 (block size = 32) and $\alpha = 8$ and $\beta = 3$ otherwise

The inverse of the round function uses modular subtraction for decryption:

$$R_k(x,y) = \big(S^{\alpha}\big((x \oplus k) - S^{-\beta}(x \oplus y)\big),\ S^{-\beta}(x \oplus y)\big).$$



SIMON & SPECK Parameters

- Each algorithm makes use of “rounds” or iterations operating on a given block sized and key size

block size $2n$	key size mn	word size n	key words m	const seq	rounds T
32	**64**	16	4	z_0	32
48	**72**	24	3	z_0	36
	96		4	z_1	36
64	**96**	32	3	z_2	42
	128		4	z_3	44
96	**96**	48	2	z_2	52
	144		3	z_3	54
128	**128**	64	2	z_2	68
	192		3	z_3	69
	256		4	z_4	72

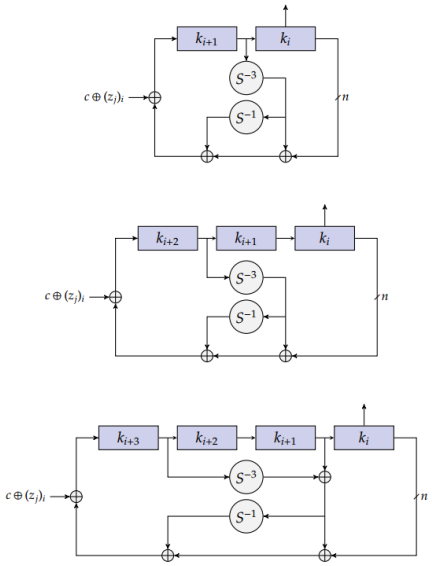
SIMON Parameters

block size $2n$	key size mn	word size n	key words m	rot α	rot β	rounds T
32	**64**	16	4	7	2	22
48	**72**	24	3	8	3	22
	96		4			23
64	**96**	32	3	8	3	26
	128		4			27
96	**96**	48	2	8	3	28
	144		3			29
128	**128**	64	2	8	3	32
	192		3			33
	256		4			34

SPECK Parameters

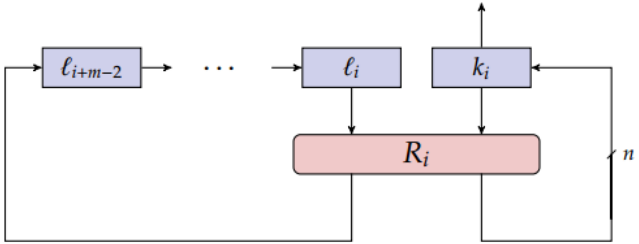
Key Schedule

- Likewise, each algorithm generates “sub-keys” for each round. Sub-keys depend only upon the block/key size and thus, may be pre-calculated



SIMON 2, 3, & 4 word Key Expansion

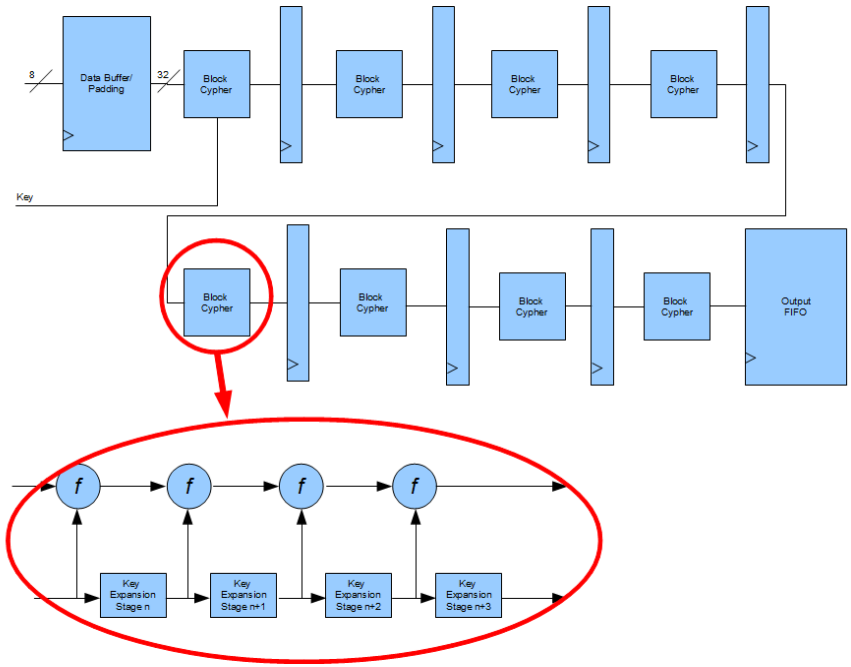
$\ell_{i+m-1} = (k_i + S^{-\alpha}\ell) \oplus i$ and
$k_{i+1} = S^{\beta}k_i \oplus \ell_{i+m-1}$.



SPECK key expansion,
where R_i is the SPECK round function with i acting as round key.

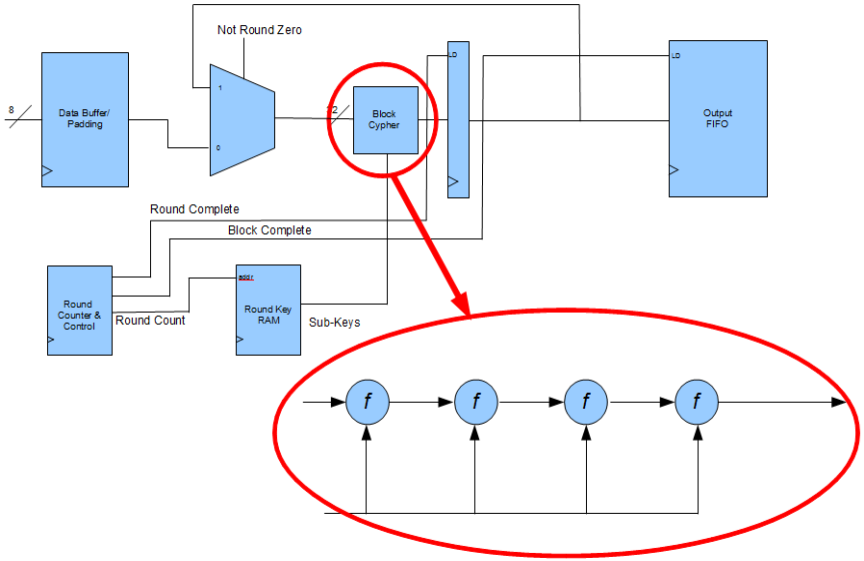
SIMON Example

- A simple SIMON 32/64 implementation
  - 3 Pipeline stages to buffer 8-bit data to a 32-bit block
  - Pad bytes are needed for messages not comprised of an even number of block
  - Eight additional pipeline stages
  - Four SIMON rounds per pipeline stage
  - A total of 11 pipeline stages for a total of 88 nS latency at 125 MHz



SIMON Example

- BUT:
  - Sub-keys can be pre-calculated and stored to reduce hardware footprint
  - Intermediate round results can aren't needed, so the cypher block can be re-used on subsequent rounds
  - Need to add only a small amount of control logic
- Further optimizations are readily feasible.
  - All 44 rounds of SIMON 64/128 can be performed in a single pipeline stage
  - Can be clocked at 300 MHz for a 130nm process node
  - 8 pipeline stages (7 data buffering, 1 for SIMON) total less than 27 nS of total latency at 300 MHz



SIMON & SPECK Peformance

- For most platforms and constraints SIMON, SPECK or both outperform existing block ciphers
  - ASIC/FPGA area
  - ASIC/FPGA efficiency (throughput/area)
  - Latency
  - Ease of side-channel protection
  - Power and energy efficiency
  - Software performance (size, speed, energy) on 8-, 16-, 32- and 64-bit processors

SIMON & SPECK vs. AES

- For a given block and key size, SIMON & SPECK provide comparable or superior security to AES

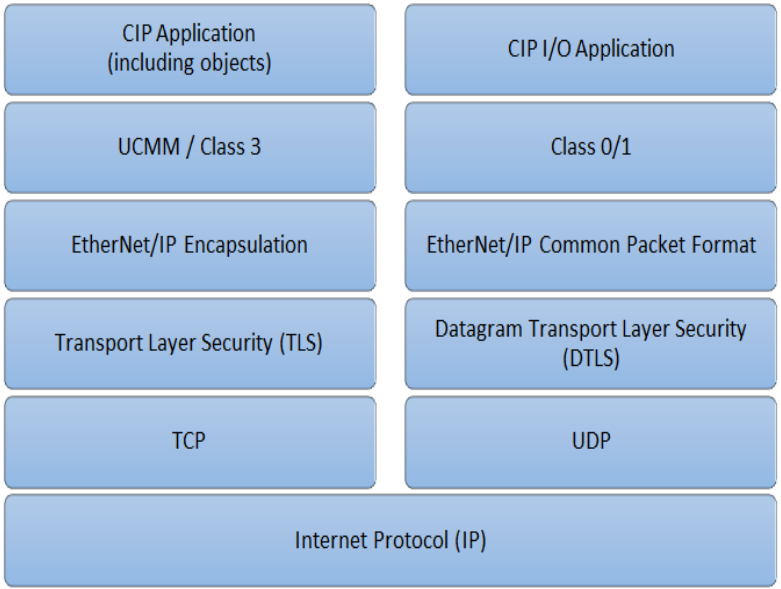
size		SIMON rounds		SPECK rounds	
block	key	total	attacked	total	attacked
48	96	36	24 (67%)	23	15 (65%)
64	96	42	28 (67%)	26	18 (69%)
64	128	44	29 (66%)	27	19 (70%)
96	96	52	37 (71%)	28	16 (57%)
96	144	54	37 (69%)	29	17 (59%)
128	128	68	49 (72%)	32	17 (53%)
128	192	69	49 (71%)	33	18 (55%)
128	256	72	50 (69%)	34	19 (56%)
AES-128		10	7 (70%)	10	7 (70%)
PRESENT		31	26 (84%)	31	26 (84%)

- For a given block and key size, SIMON is more efficient than AES

Size	Algorithm	Area (GE)
128/128	SIMON	1234
	SPECK	1280
	AES	2400

Application to CIP Security

- CIP Security uses proven, open security technologies:
  - X.509v3 Digital Certificates used to provide cryptographically secure identities to users and devices
  - TLS (Transport Layer Security) and DTLS (Datagram Transport Layer Security) cryptographic protocols used to provide secure transport of EtherNet/IP traffic
  - Hashes or HMAC (keyed-Hash Message Authentication Code) as a cryptographic method of providing data integrity and message authentication to EtherNet/IP traffic
  - Encryption as a means of encoding messages or information in such a way as to prevent reading or viewing of EtherNet/IP data by unauthorized parties

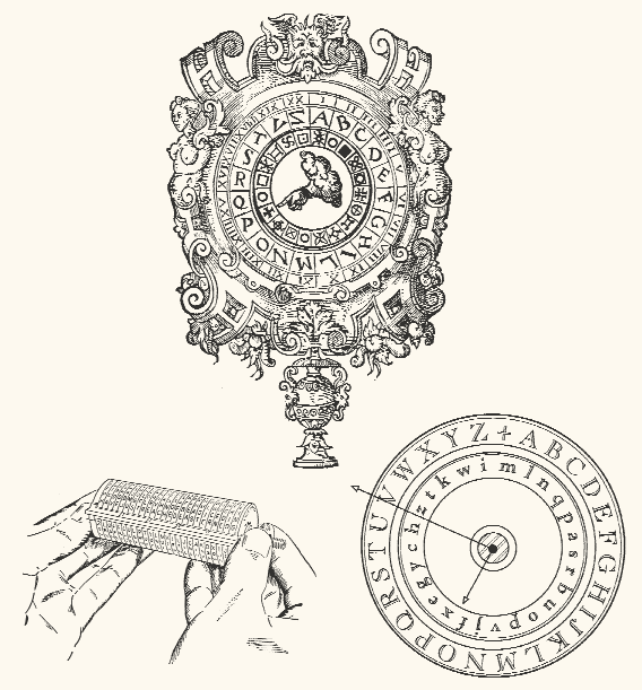


Application to CIP Security

- Obvious applications of SIMON & SPECK to CIP Security is message encryption
- However, CIP Security is also strongly focused on authentication
  - Digital Certificates for identity establishment
  - HMAC for message authentication
- Hash algorithms and RSA based certificate exchange also require significant resources
- An alternative for highly-constrained devices may be attractive

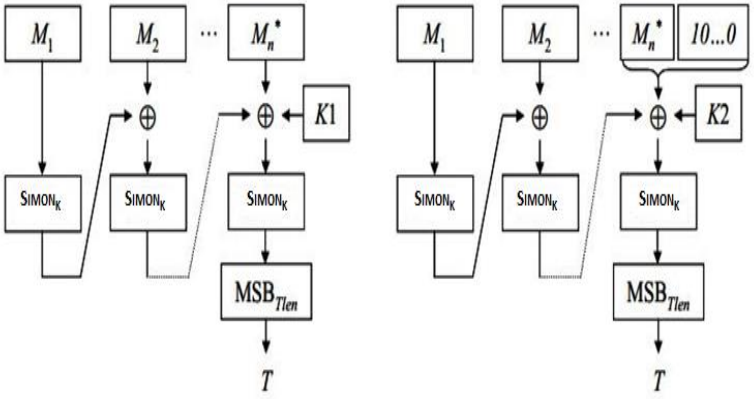
Alternative for Authentication

- Cipher-based message authentication code (CMAC) provides an alternative means of message authentication based on a symmetric key block cipher
  - NIST special publication 800-38B
  - Proven implementations with AES (IPSEC)
  - Suitable for SIMON & SPECK
  - May be more appropriate for highly-constrained devices
  - In extremely constrained environments, the symmetric keys can be pre-shared



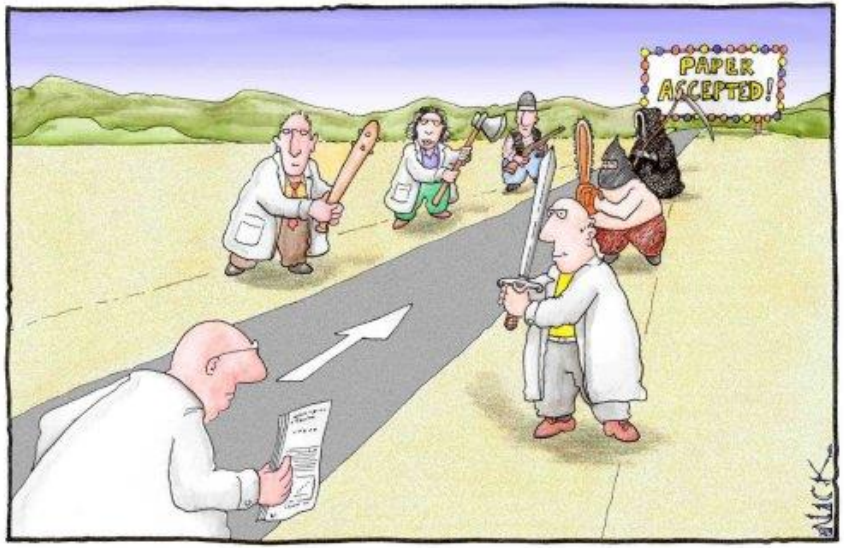
CMAC Overview

- The CMAC algorithm:
  - Takes a message (*M*), of bit length *Mlen*
  - Chains the block cypher by acting on a single block (*Mi*) and using a bitwise exclusive-or to sum the output of each stage in the chain
  - Produces a MAC, a.k.a message authentication code, (*T*), of bit length *Tlen*
  - *T* is appended to the outgoing message
  - On ingress, the process is repeated and the resulting MAC is compared to the MAC appended to the message
    - The CMAC algorithm also makes use of sub-keys
    - Distinct from the sub-keys generated for each SIMON/SPECK round



Setting a High Bar

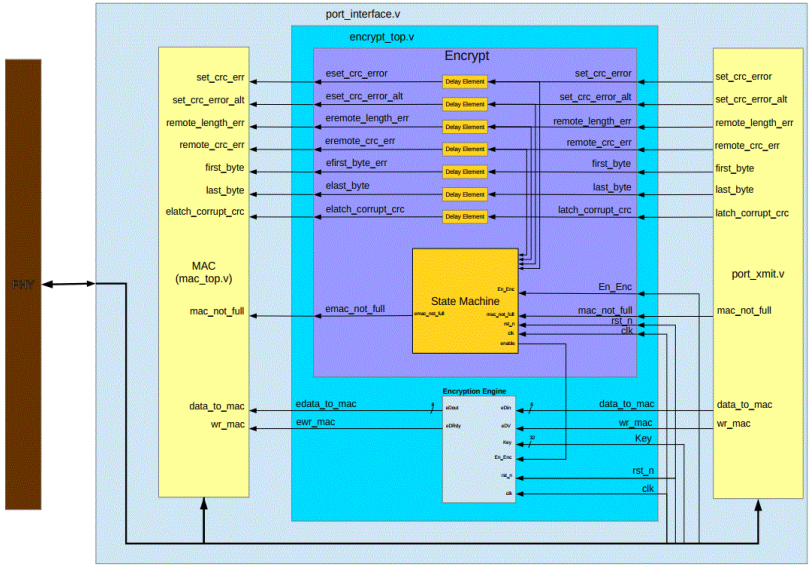
- ODVA makes consistent use of proven technologies
- Clearly, to be of use, SIMON & SPECK must be standardized and pass a high-level of scrutiny with the security community
  - SIMON and SPECK have been submitted for inclusion in ISO 29192-2, the standard for lightweight block ciphers. This proposal is currently in review
  - Significant analysis of this technology has already been performed and shows great promise for robust security in constrained applications
  - Open technology. The algorithm and associated research are public domain



Most scientists regarded the new streamlined peer-review process as 'quite an improvement.'

A Practical Example

- Goals:
  - Based upon an existing Ethernet/IP DLR demonstration
  - Integrated SIMON IP with Switch IP.
  - No changes to stack or application SW.
  - Completely transparent to Demo operation.



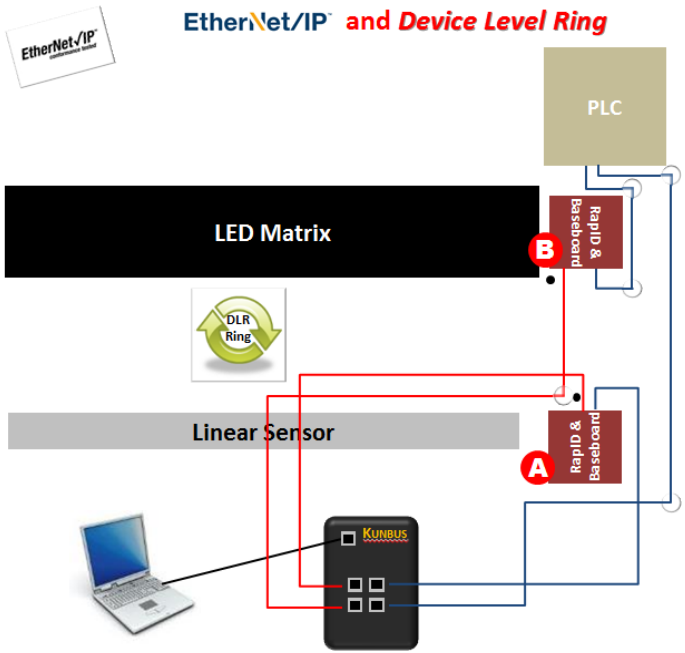
Demonstration Parameters

- SIMON 32/64
  - 32 rounds; Key 64 bits
  - Message encryption only
    - Layer 3 and above
- Latency
  - 11 stages @ 125 mhz; 88 ns
- Input data
  - 8 bit, 1-3 pad bytes added
- Key change process
  - Key expansion pre-computed to minimize latency
  - Requires pipeline flush and stall

block size $2n$	key size mn	word size n	key words m	const seq	rounds T
32	64	16	4	z_0	32
48	72	24	3	z_0	36
	96		4	z_1	36
64	96	32	3	z_2	42
	128		4	z_3	44
96	96	48	2	z_2	52
	144		3	z_3	54
128	128	64	2	z_2	68
	192		3	z_3	69
	256		4	z_4	72

Demonstration Operation

- Linear Sensor provides position information.
  - Control module routes data to PLC
- PLC provides simple ladder logic to route linear sensor data to the display Rapid Platform
  - Control module renders position on the display
- The encrypted link is shown in red between the REM-based modules labeled a and b with the security algorithm enabled.
- A network tap installed on the encrypted link and on the unencrypted link.
- A network analyzer demonstrates secure link is operational and transparent to operation.



A Practical Example

- Goals:
  - ✓ Based upon an existing Ethernet/IP DLR demonstration
  - ✓ Integrated SIMON IP with Switch IP.
  - ✓ No changes to stack or application SW.
  - ✓ Completely transparent to Demo operation.
    - 200 uS DLR beacon traffic unaffected by encryption
    - SIMON effectively looks like a slow wire (88 nS on ingress and egress)

Conclusions

- Ethernet has enjoyed unprecedented success as a communication medium
  - The promised explosions of IoT, iIoT and Industry 4.0 threaten to dwarf this success
- Don't believe in the IOT explosion? Consider this:
  - How many MAC Addresses did you use in 1998? Typically less than 5:
    - Work computer, home computer, a laptop. . .
  - Move to 2014. Now how many MAC Addresses do you use? Typically 10 to 15:
    - Cell phone, IP phone, laptop (2 – 1 for wired, 1 for wireless), laser printer (2 – same reason), set top box (2), TV, BluRay player, tablet, computer at home (2), wireless AP, . . .

Conclusions

- Ethernet's continued success will give rise to a host of new applications with extremely limited resources
- SIMON & SPECK potentially address such applications and should be considered as CIP security technologies evolve.
- Lightweight block cyphers offer:
  - A small hardware footprint (SIMON)
  - Small software footprint (SPECK)
  - Scalability
  - In-line encryption/decryption;
  - Low latency; Low jitter
  - Comparable security to AES

"We were ahead of the curve, and then the curve ran right over us."

THANK YOU