SMU | Bobby B. Lyle School of Engineering




USENIX Hotsec’11

Security Fusion: A New Security Architecture for Resource-Constrained Environments

Suku Nair, Subil Abraham, Omar Al Ibrahim
HACNet Labs, Southern Methodist University

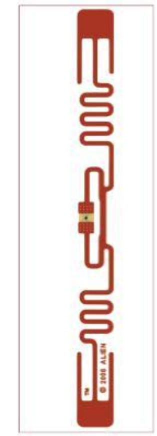
Resource-Constrained Devices

Alien Squiggle 1.1 (EPC C1G2)

Constraint	**Value**
Gate count	7500 GE
Memory	240 bits
Power consumption	25uW
Response time	15~30us
Bandwidth	860~960 MHz
Die space	0.4mm x 0.4mm
Physical size	97mm x 11mm

Iris Mote (IEEE 802.15.4)

Constraint	**Value**
Memory	Flash: 128 KB EEPROM: 4 KB RAM: 8 KB
Processor	16 MIPS @ 16 MHz
Power supply	2 AA Batteries
Radio communication	RF230 2.4 GHz IEEE 802.15.4



RFID

Sensors

References:
1) *Alien Squiggle family. http://www.alientechnology.com/docs/products/DS_ALN_9640.pdf*
2) *IRIS datasheet. http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/IRIS_Datasheet.pdf*

Encryption Algorithms

Algorithm	Key(bit)	Plaintext (bit)	Cycles	GE	Power	Technology (μm)
AES	128	128	1016	3595	8.15 μA	0.35
TEA	128	64	64	2355	12.34 μW	0.18
SHA-1	L	192(in) 160(out)	405	4276	26.73 (1.2V)	0.13
Stream-cipher (1 LFSR)	Max: 32	64	92	685	0.1582 μW	0.18
DES	56	64	144	2309	2.14 μW	0.18
ECC	Field = 113	L	195159	~ 10K	L	0.35
IDEA	128	64	320	4660	3 μW	0.18

Reference: R&D of Gen 2 with enhanced security mechanism, Auto-ID Lab at Fudan, March 2009

Challenges

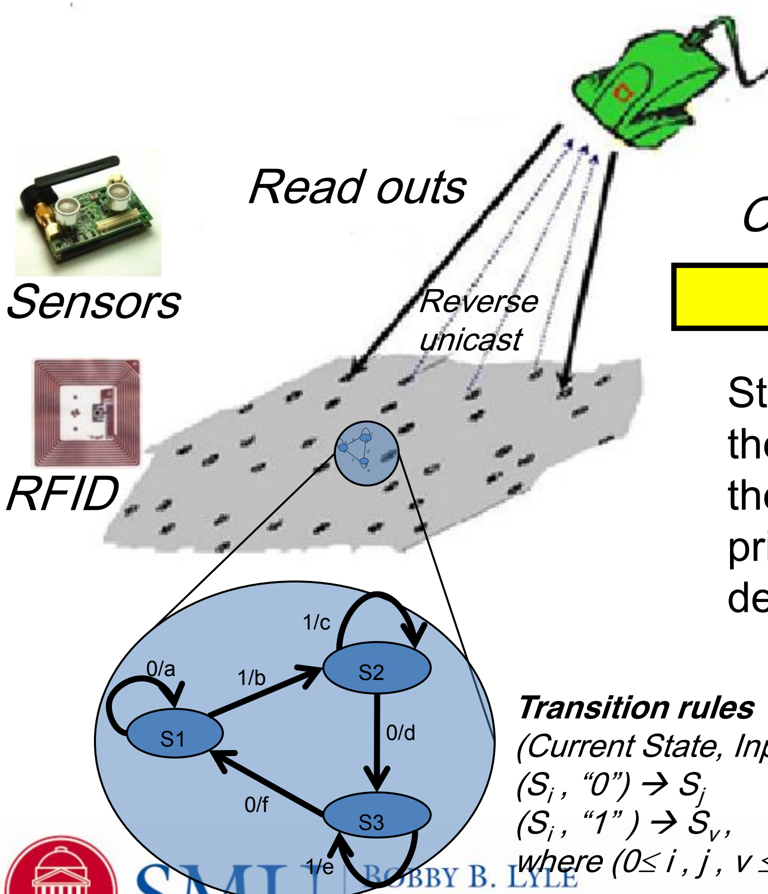
- Resource constraints
  - Crypto may not be available
  - AES/SHA-2 needs 20-30 thousand gates
  - Energy constraints
- Proliferated number of devices
- Untrusted environment
  - Nodes can be easily compromised
- Wireless medium – inherently broadcast
- Aggregation-based applications

Types of Attacks

- Eavesdropping
- Malicious reads
- Replay attacks
- Cloning
- Brute-force search
- Denial-of-service

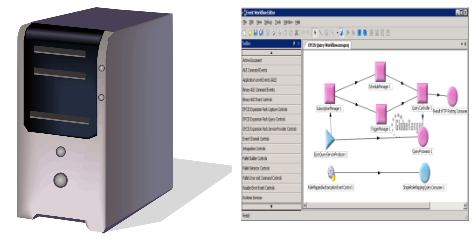
Security Fusion: The Concept

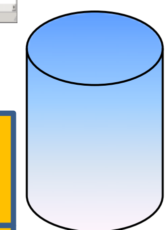
A new paradigm in security for resource-constrained environments



Collect responses

Strong security properties at the infrastructure level through the synergy of inherently weak primitives from multiple devices





Application

Integration

Networking

Transition rules

(Current State, Input) → Next State

$(S_i, \text{"0"}) \rightarrow S_j$

$(S_i, \text{"1"}) \rightarrow S_v$,

where $(0 \leq i, j, v \leq n)$

Output rules

(Current State, Input) → Output

$(S_i, \text{"0"}) \rightarrow a_i$

$(S_i, \text{"1"}) \rightarrow b_i$,

where $a_i \neq b_i$

State Machine Model

State machine description (Mealy machine):

Transition rules

(Current State, Input) → Next State

$(S_i\,,\ input_A\,) \rightarrow S_j$

$(S_i\,,\ input_B\,) \rightarrow S_v\,,$

where $(0 \leq i\,,\ j\,,\ v \leq n)$ *and* $input_A \neq input_B$

Output rules

(Current State, Input) → Output

$(S_i\,,\ input_A\,) \rightarrow a_i$

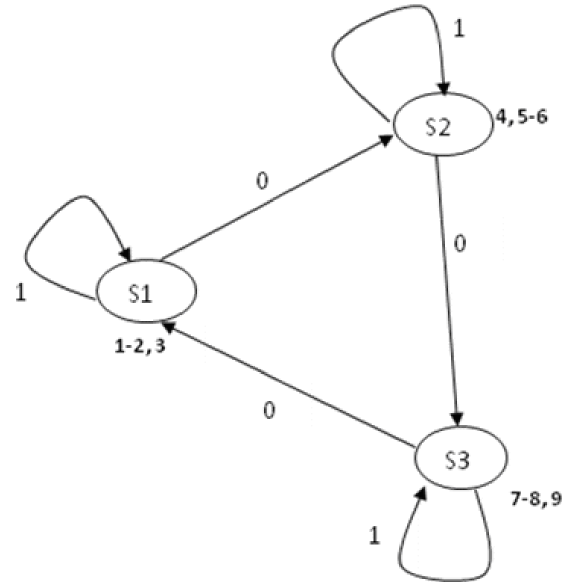
$(S_i\,,\ input_B\,) \rightarrow b_i\,,$

where $a_i \neq b_i$ *when* $input_A \neq input_B$

Example

Consider a 3-state Finite State Machine (FSM)

- **n=3** $\{s_1, s_2, s_3\}$
- **k=3** [Each state is assigned a set of *3* pseudonyms of which *p (1<= p < k)* pseudonyms may be used to represent (*0*) and *q = k-p* pseudonyms may be used to represent a (*1*).]
- The total set of pseudonyms available for the 3- finite state machine = **nk** = 9
- Each state (s_1, s_2, s_3) will have *k* pseudonyms assigned to it.



State Diagram

States	Transition on "0"	Transition on "1"
S_1	1, or 2	3
S_2	4	5, or 6
S_3	7,or 8	9

Pseudonyms Assignment

Security Protocol

Denote ***N*** : *Node*, ***R*** : *Reader*

R → *N:* Send read query

N: Obtain *<transition bit> (0/1)*

N→ *R:* *N* moves to the next state based on *<transition bit>* and outputs an pseudonym

R resolves *N*'s output and syncs

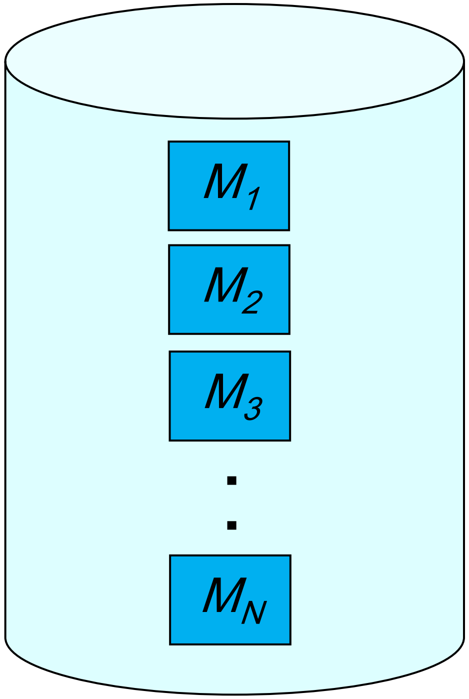
Machine Indexing

Machine input

Current execution

Node ID	Flag	Current State	Next State / Output	
			i=0	i=1
M_1	0	s_1	s_4/{14,7,39}	s_3/{17,4,23}
	1	s_2	s_2/{10,13,8}	s_2/{12,19,1}
	0	s_3	s_4/{6,11,26}	s_1/{32,5,18}
	0	s_4	s_3/{8,21,43}	s_2/{2,45,9}
M_2	…	…		
…	…	…		

Pseudonym set



k: pseudonyms/state
n: no of states
N: no of machines
$\Theta(k*n*N)$ entries

Fusion Logic

1. Consensus of the response pattern into one secure metric
2. With *N* nodes, an intruder needs to derive at least *N/2* state machines to influence system behaviour
3. Used to reach a global decision
4. Security complexity is non-linear

Machine Selection Criteria

1. State reachability
   – Every state should be reachable to every other state through a sequence of transitions
2. Machine complexity
   – *NFA-DFA conversion* should be non-linear
3. Pseudonym randomness
   – Values assigned to states are random and unpredictable.
4. Pattern randomness
   – The execution pattern should be random as well

Analysis: Large-Scale Attacks

NFA-DFA State Blowup

Given a natural number m, there exists an m-state NFA whose minimal equivalent DFA has ≥ 2^m-1 states

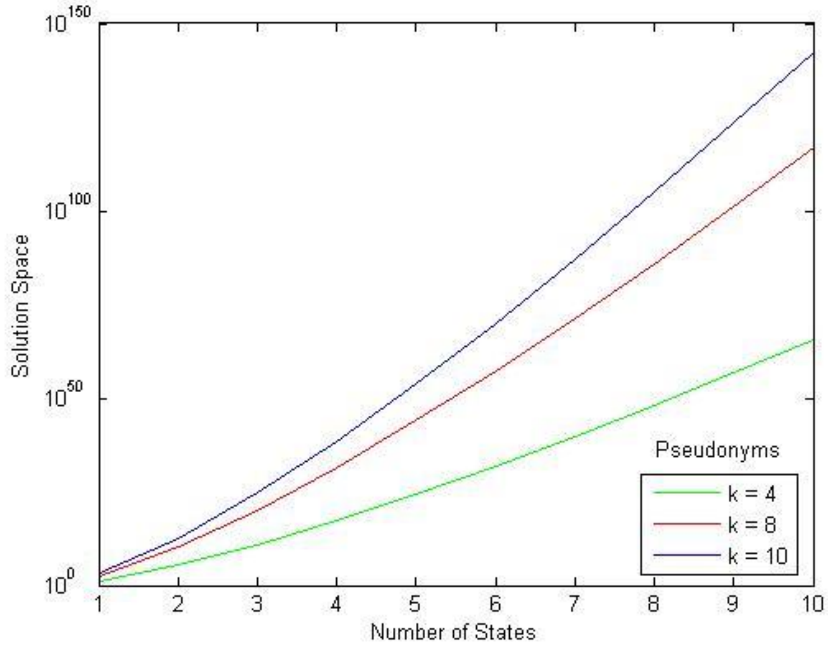
- *n: number of states, k: pseudonyms per state, and m=nk*
- Attacker builds an NFA with *nk* states *nk^2* edges
- Hopcroft's Algorithm : *m* log (m) for DFA*
- *NFA→ DFA conversion lead to exponential blowup in states for some machines*

Analysis: Solution Space

Observation

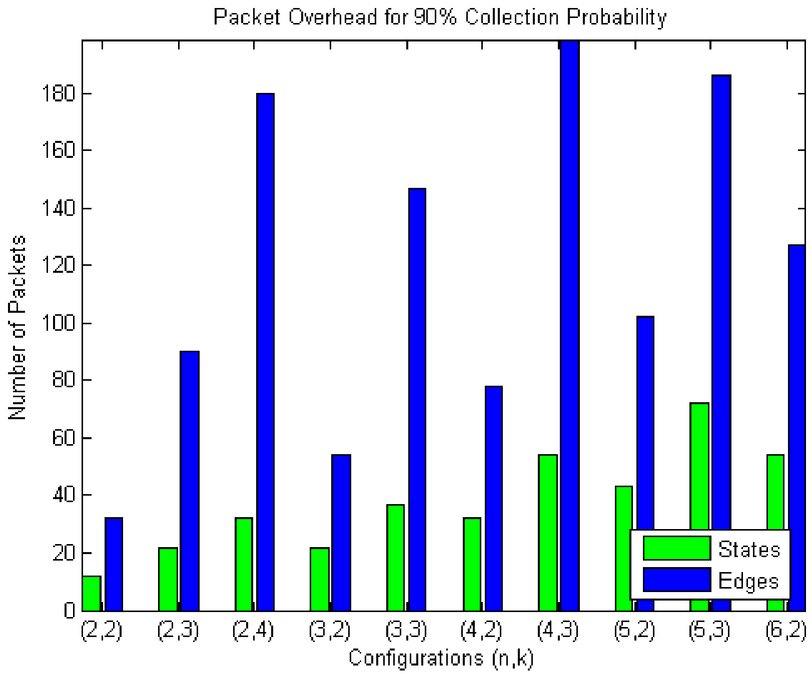
- *With n states, each of which may move to any state depending on two input values, and with nk numbers to be assigned into n states with k elements in each state, of which p (1≤ p < k) numbers may be used to represent a transition on 0, and q (q=k-p) numbers may be used to transition on 1, the total number of possible state machines that can be generated is*:

$$= (n)^{2n} \left[\sum_{p=1}^{k-1} \frac{k!}{p!(k-p)!} \right]^n \left[\frac{nk!}{(k!)^n} \right]$$



Analysis: Malicious Reads

- Estimate the number of packets to determine state values and transitions
- Randomness assumption based on Pascal's equations



Conclusion/Future Work

- New paradigm, namely “security fusion” has been introduced
- Explore finite automata concepts to realize security fusion
- Viable, state-machine based implementation of “security fusion”
- Investigate other models for security fusion to provide strong overall security guarantees for resource-constrained environments

Questions ?