Distributed Software-based Attestation for Node Compromise Detection in Sensor Networks

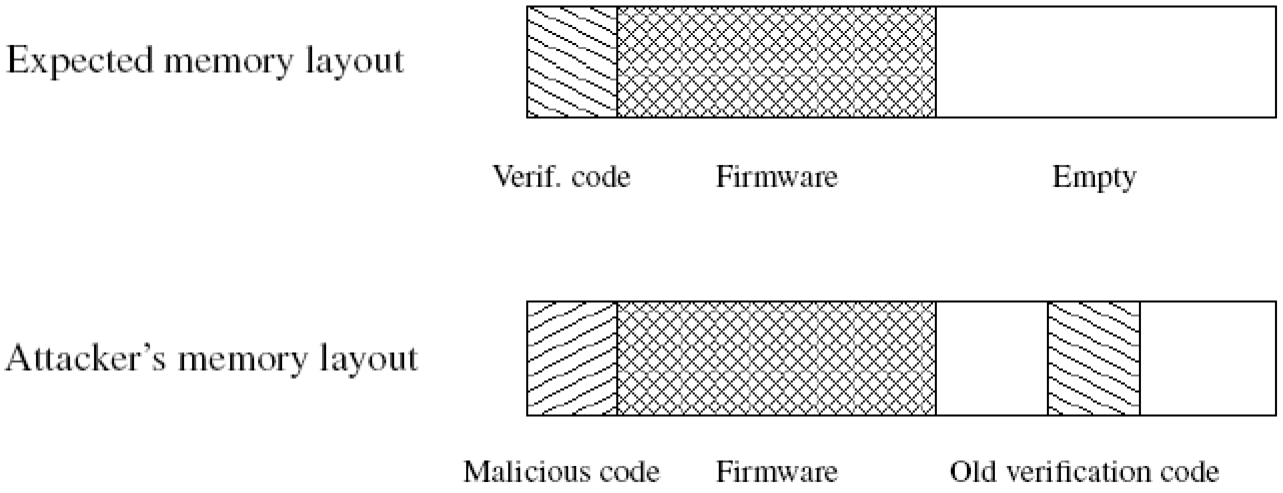
Outline

- Introduction
- System Model and Assumptions
- Preliminaries
- Proposed Schemes
- Simulation results
- Conclusion
- Comment

Introduction

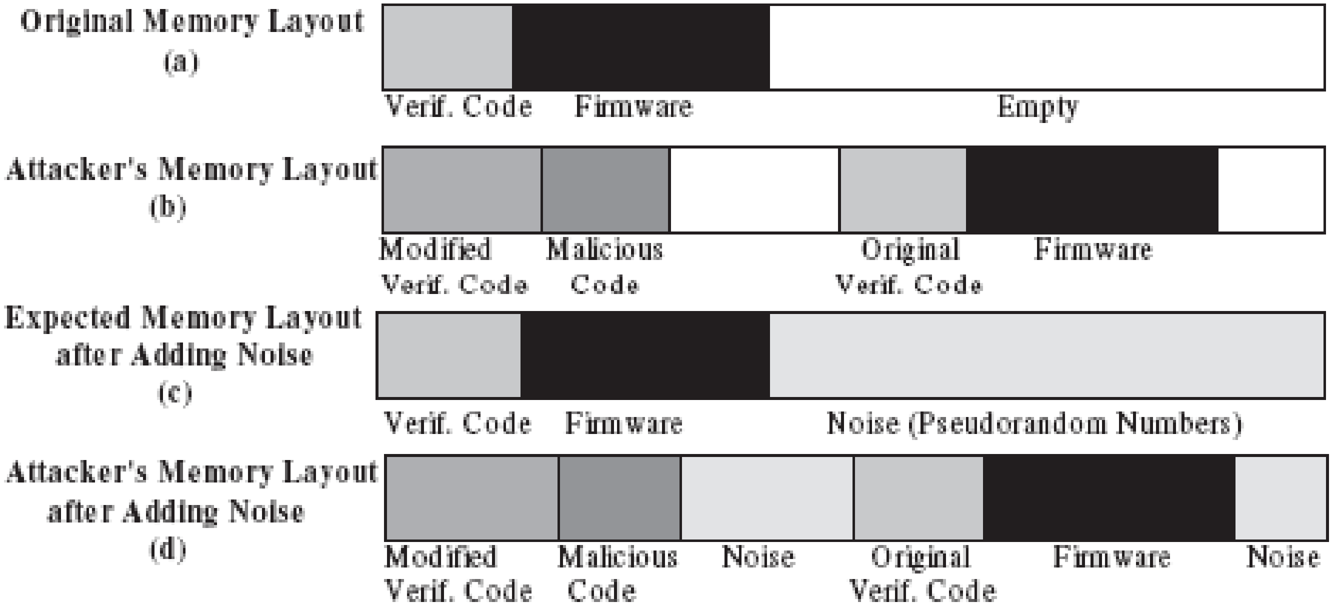
- SWATT(**SoftWare-based ATTestation**) *externally attests the code, static data, and con*figuration settings of an embedded device

- By “externally” we mean that the entity performing the attestation (verifier) is physically separated



System Model and Assumptions

- Network Model
  - Densely deployed
  - Multiple immediate neighbors
- Attack Model
  - Not enlarge the sensor’s memory
- Program memory layout before and after adding noise

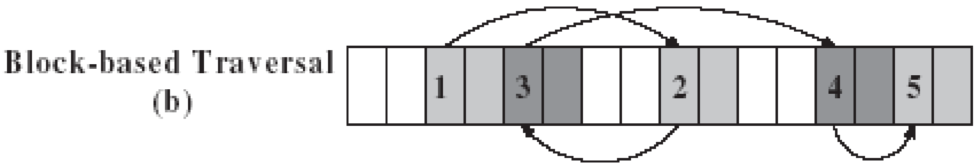


Preliminaries - Noise Generation

- The design principle is for the attestor's convenience in reconstructing the attested node's memory image
  - Pseudo Random Number Generator (PRNG)
- Each time we are outputted with an 8-byte noise, we only need *m'/8 counters to generate all the noise, if m' is* empty memory size (in byte) to be filled

Preliminaries - memory traversal

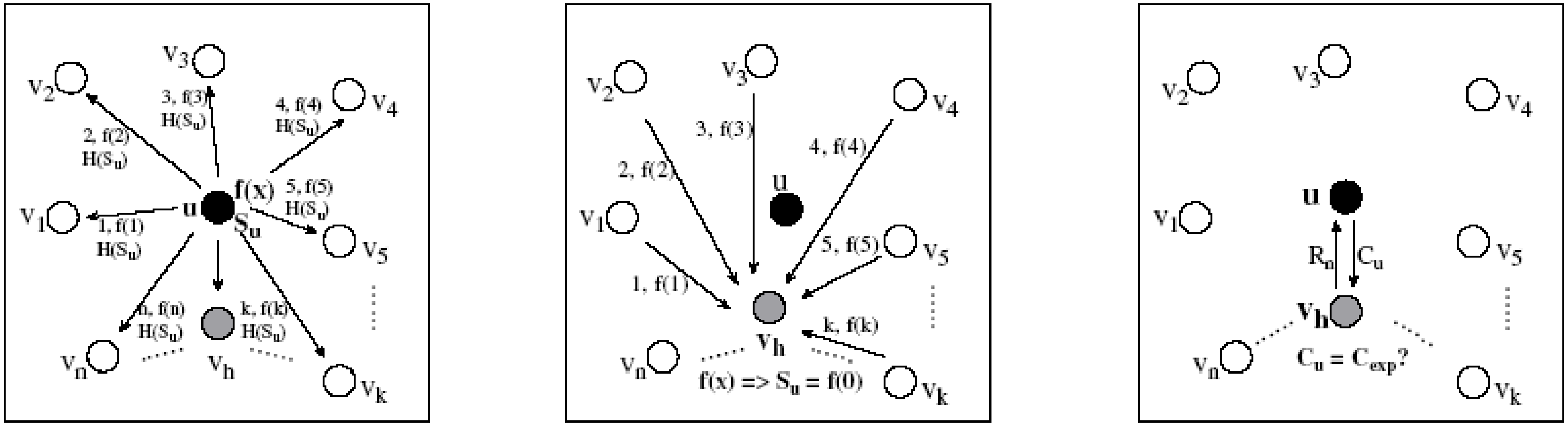
- *Block-based pseudorandom* memory traversal algorithm
  - Traverse memory and efficient 'XOR' operations are executed within blocks



  - Based on the Coupon Collector's problem, on average *O(m ln m) traversals are needed for the* memory size of *m, so the new one is O((m ln m)/b)*
    - Coupon Collector's problem describe the "collect all coupons and win" contests

Proposed Schemes

- **Scheme I: A Basic Threshold Secret Sharing Scheme**
  - The empty memory space of each node *is filled with pseudorandom numbers derived from* a unique noise-generation seed *Su*
  - Node *u distributes one share of its noise-generation seed to* each of its neighbors
  - When an attestation is triggered against node *u, neighbors collaborate to recover Su,* reconstruct the memory image of node *u*



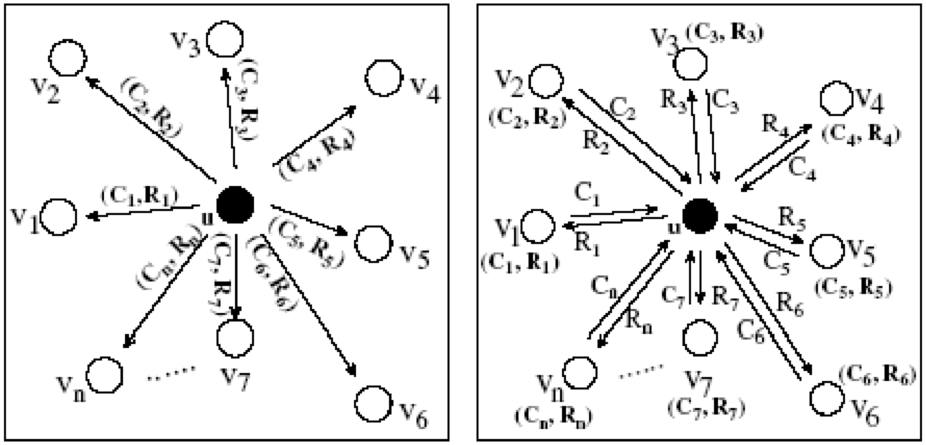
(a) Share distribution

(b) Seed recovery

(c) Attestation

Proposed Schemes

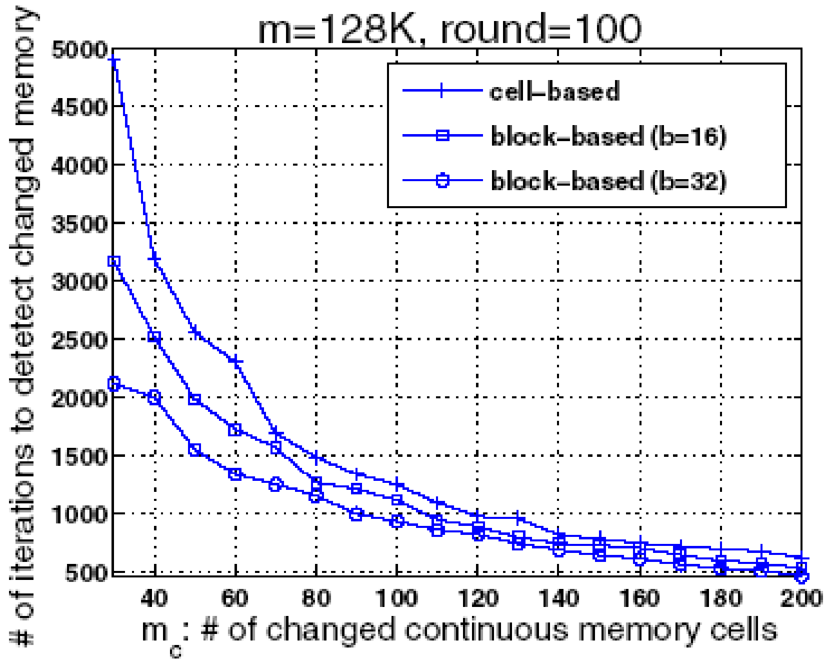
- **Scheme II: A Majority Voting Based Attestation Scheme**
  - Each neighbor is distributed with and keeps a challenge as well as the corresponding response
  - During an attestation, each neighbor sends the challenge and waits for the response from the attested node

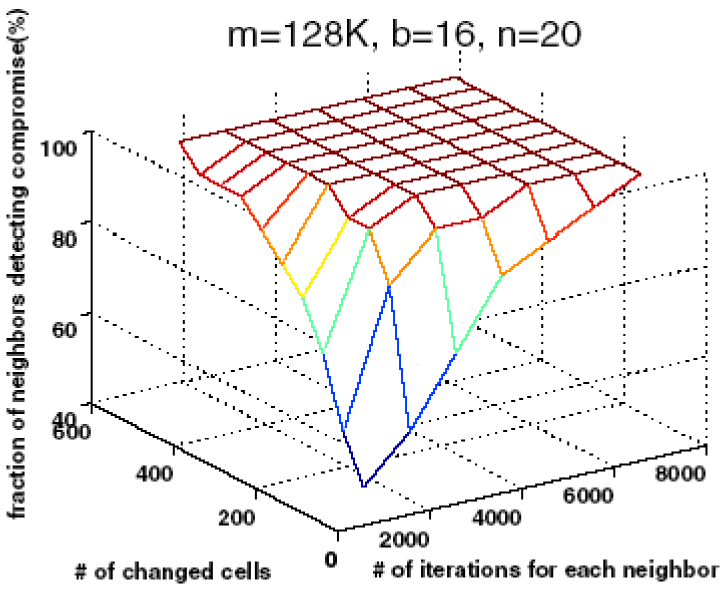


(a) Information distribution

(b) Attestation

Simulation results





Number of iterations for cluster head to detect changed cells in Scheme I.

Fraction of neighbors successfully detecting changed cells in Scheme II.

Conclusion

- Software-based code attestation identifying compromised nodes, they are not readily applied into regular sensor networks due to one or another limitations

- Their scheme do not depend on response time measurement by mobile verifiers or the base station

Comment

- In real networks, the actual number of neighbors may be different from what we have predicted.

- Nodes may be added and die during the network life time, it’s a problem that to solve the noise generation of new node.