Low Cost Embedded Network Message Authentication

Chris Szilagyi, Prof. Phil Koopman

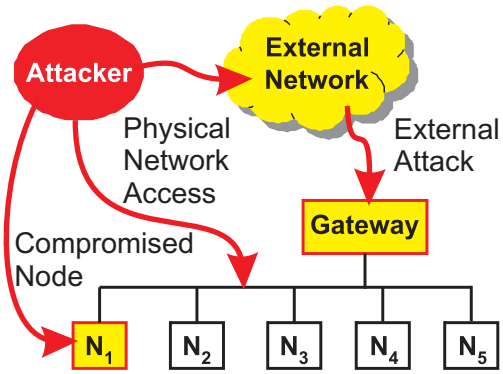
Carnegie Mellon University

Can you get multicast cryptographic authentication in only 3 or 4 bytes?

Motivation:

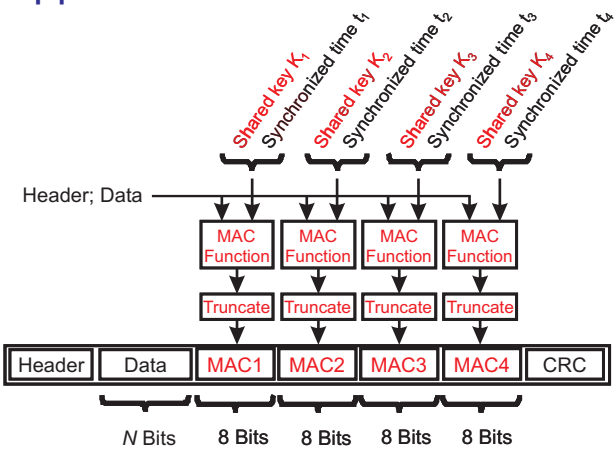
Most embedded network protocols provide no authentication, so you have no idea if the node that sent a message is the one that claims to have sent it. In other words, one node can masquerade as another node and send incorrect or malicious messages that disrupt system operation! This could make real time embedded networks such as CAN and FLEXRAY vulnerable to malicious attacks. It can also result in non-malicous faults resulting in incorrect system operation.

Nobody wants to put a 128-bit (or higher!) overhead on a message with an 8-bit payload. So, how can we provide cryptographically secure message authentication with about the same overhead as networks already pay for error checking? And, how can we do this without changing network protocol hardware?



Example: N_1 pretends to be N_2

Approach:



Use a few bits of overhead per message:

- Symmetric keys (one key per receiver) to generate secure hashes (MACs)
- Base each MAC on message contents, receiver key, and message round counter
- Add 1 or more bits per receiver and send truncated MAC bits

For state-changing or irrevocable actions:

- Accumulate several messages before committing to a change

For continous control messages:

- Take action immediately after receiving message
- Count on system inertia to smooth out bumps from successful attacks

For both cases, base MAC size on criticality and time constants.

Status:

Current Results:

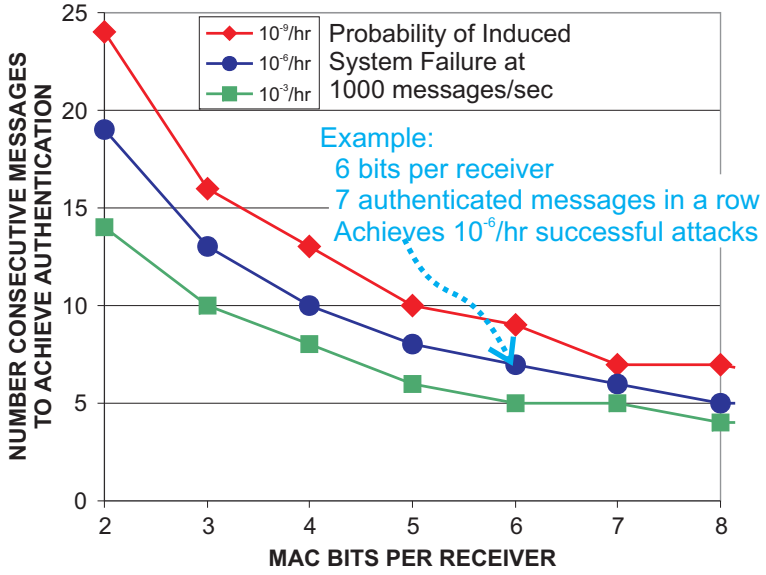
- Approach enables engineering tradeoff among:
  Per-message overhead *vs.* **Attack resistance** *vs.* **Detection Latency**
- More bits per message gives less likely attack success
- More bits per message reduces detection latency
- Approach will improve attack resistance even with only a few bits per message

Future Work:

- Further reduce overhead using group membership
- Combine with other techniques (e.g., TESLA)

Timeline:

- Expected Ph.D. thesis completion in August 2011



Electrical & Computer ENGINEERING

May 2010 – koopman@cmu.edu – http://www.ece.cmu.edu/~koopman