Prolog to Lecture 6
CS 236
On-Line MS Program
Networks and Systems Security
Peter Reiher

Authenticating for Small Devices

- Typical authentication mechanisms assume powerful devices
  - To run PK, for example
- What about embedded devices?
  - Small, limited hardware, poorly administered, limited battery
- Will PK authentication work for them?

Why Are They Authenticating?

- To receive commands
- To get data from other sensors
- To receive updates to their software
- To provide information to particular users
- They probably need to authenticate

Why Not PK?

- PK is expensive
  - 100X computation cost as symmetric crypto in software
  - 1000X in hardware
- Not only slow, but drains the battery
- Also issues of PK infrastructure

How About Symmetric Crypto?

- Faster and uses less power
- But poor authentication properties
  - Did I create it or did my partner?
- Also poor scaling properties
  - Which might or might not be an issue for embedded devices

Is There Another Choice?

- Maybe reverse hash chains
- Basic idea:
  - Authenticate end of hash chain via PK crypto
  - Use next link in hash chain to authenticate next message
  - And so on
  - Use PK to sign new hash chain, when needed

For Example

X''''' → X'''' → X''' → X'' → X' → X

Alice

$Y = E(X'''''', K_{EA})$

$Z = F(X'''', M)$

X'''', M

Bob

X''''''

Z

Is $H(X'''')$ = X''''''?

Is $Z = F(X'''', M)$?

Are There Problems With This?

- Ask yourself what Bob actually knows at each step
- Are there differences between what he actually knows and what we want him to know?
- Could attackers make use of those differences?