Privacy-Preserving Attribution and Provenance

Alex C. Snoeren, Stefan Savage, Amin Vahdat, Geoffrey M. Voelker, and Yoshi Kohno
University of California, San Diego and the University of Washington

UCSDCSE
Computer Science and Engineering

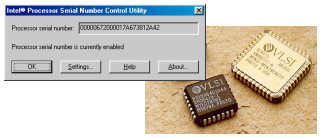
We want to be here…

The crisis of anonymity on the Internet

and simultaneously here

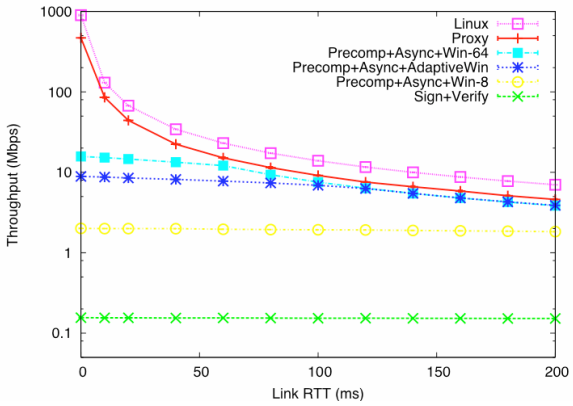
Complete anonymity — We are here — Fingerprints & DNA

- Prima Facie, digital objects like packets are not unique
  - Contrast to physical objects/actions which tend to carry/acquire distinguishing attributes
  - E.g. Location/Time, Weight, Size, Fingerprints/DNA…
- Thus, there is a heavier burden to attribute digital actions
  - Yet, users have an expectation or rights to privacy
  - Hence, we don't want to leak physical identity
- Must not repeat notable historic failures
  - Clipper/Capstone chip (1993) encrypted voice/data; keys escrowed with the government
  - Intel Pentium III PSN (1999) had unique per-processor serial number
  - Both abandoned in face of huge pressure from civil liberties groups

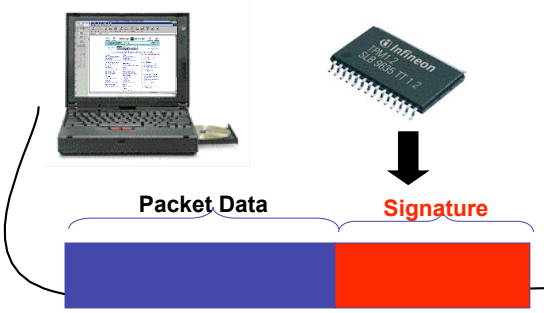


A first step towards accountability

- Provide **network support** to tie a digital action (packet) to a physical machine
  - Link different criminal actions
  - User attribution via ownership records/physical forensics
- **Any** network element can vet a packet as being "attributable," but can't identify physical machine (privacy preserving)
- **Authorized** actor can reveal origin for any attributable packet at any point in future
- Key mechanism: **short group signature**
  - PK cryptosystem; *n* secret keys, one public verification key, and one opener key
  - Verify(): signature was generated with *one* of the group's secret keys
  - Open(): reveals *which* secret key generated this signature
  - Standard symmetric-key crypto is *not* sufficient: no way to verify signatures



Our prototype implementation: CLUE



- Assume secure hardware (TPM) on each device (secure key & computation)
- Digitally sign each packet; signature identifies physical origin
- Expensive in naive implementation : 24ms sign, 28ms verify
- We have implemented several optimizations
  - **Precomputation**: Most of signature can be pre-computed in advance (during idle cycles); sign 35us
  - **Windowed verification**: Modify signature so you can verify this packet or window of the last *n* packets
  - **Asynchronous verification**: Overlap verification with communications

This work is supported by the National Science Foundation FIND program through grants CNS-0722031 & CNS-0722000