

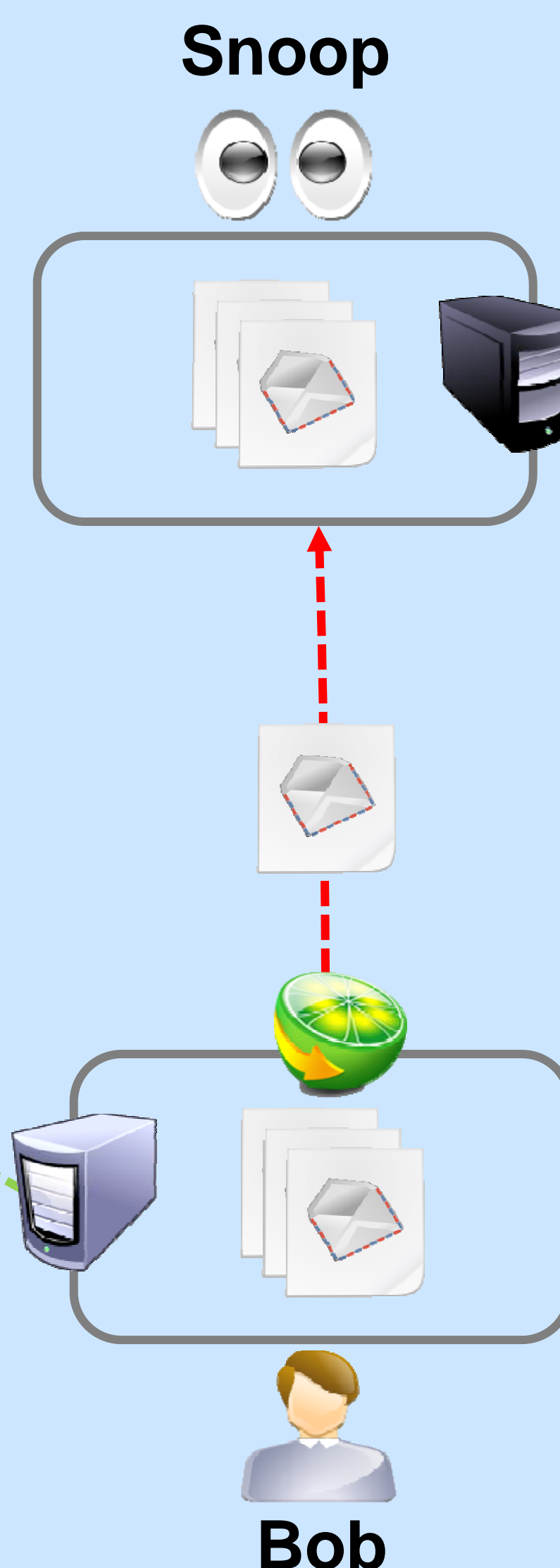
Practical Mandatory Access Control

Landon Cox, Duke University

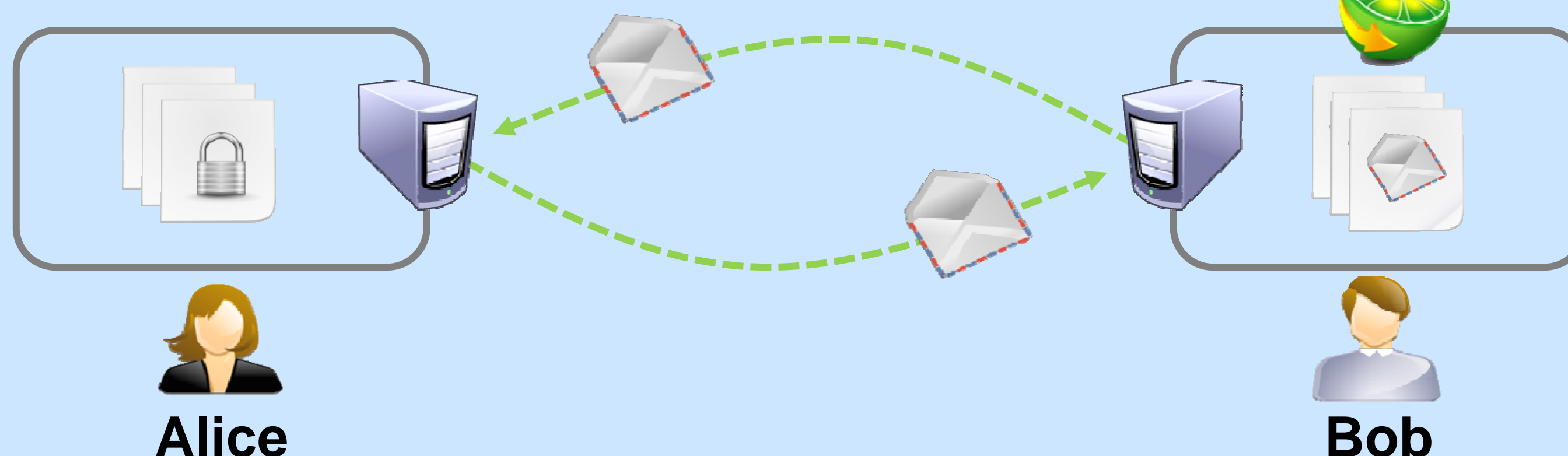


Fundamental tension between privacy, sharing

- ▶ Data sharing among users, organizations is widespread
 - ▶ Peer-to-peer file sharing, work outsourcing, etc
- ▶ Responsibility for sensitive state delegated to unreliable regimes
 - ▶ Poorly administered clients accumulate sensitive state
 - ▶ Client operators must properly specify access control policy
- ▶ **Problem: access control policy specification is hard**
 - ▶ Non-experts are unprepared to protect sensitive state
 - ▶ Has led to leaks in corporate, military, PC settings



Alice's secret is only as safe as her least competent confidante.



Three-phased approach to practical mandatory access control

1) Automatically flag sensitive files

Key insight: clients often *cache* sensitive data via *encrypted* channels

Approach: identify encrypted downloads by maintaining per-socket entropy scores, use information-flow analysis and process logging/replay to follow encrypted data into file system

2) Follow flow of sensitive data

Key insight: server-like applications often read a file, transform it, and send it to the network

Approach: when a process reads a sensitive file, create *copy doppelganger process* of that process; give the original process the sensitive data and the doppelganger a redacted copy of the data; compare the outputs of both processes transitively marking any differences as derived from the sensitive data

3) Automatically infer trusted endpoints

Key insight: secrets can be repeated back to their source without compromising privacy

Approach: maintain a *knows-of set* with each sensitive file containing the set of hosts who contributed sensitive data to it; any data derived from that file can be sent to a member of the "knows-of" set; otherwise, the operator must be notified of a potential leak