Outline

- Introduction
- Background
- Distributed DBMS Architecture
- Distributed Database Design
- Distributed Query Processing
- Distributed Transaction Management
- Building Distributed Database Systems (RAID)
- Mobile Database Systems
- Privacy, Trust, and Authentication
- Peer to Peer Systems

Useful References

- B. Bhargava and L. Lilien, *Private and Trusted Collaborations*, in Proceedings of Secure Knowledge Management (SKM), Amherst, NY, Sep. 2004.
- W. Wang, Y. Lu, and B. Bhargava, *On Security Study of Two Distance Vector Routing Protocols for Mobile Ad Hoc Networks*, in Proc. of IEEE Intl. Conf. on Pervasive Computing and Communications (PerCom), Dallas-Fort Worth, TX, March 2003.
- B. Bhargava, Y. Zhong, and Y. Lu, *Fraud Formalization and Detection*, in Proc. of 5th Intl. Conf. on Data Warehousing and Knowledge Discovery (DaWaK), Prague, Czech Republic, September 2003.
- B. Bhargava, C. Farkas, L. Lilien, and F. Makedon, *Trust, Privacy, and Security*, Summary of a Workshop Breakout Session at the National Science Foundation Information and Data Management (IDM) Workshop held in Seattle, Washington, September 14 - 16, 2003, CERIAS Tech Report 2003-34, CERIAS, Purdue University, November 2003.
- P. Ruth, D. Xu, B. Bhargava, and F. Regnier, *E-Notebook Middleware for Accountability and Reputation Based Trust in Distributed Data Sharing Communities*, in Proc. of the Second International Conference on Trust Management (iTrust), Oxford, UK, March 2004.

Motivation

- Sensitivity of personal data
  - 82% willing to reveal their favorite TV show
  - Only 1% willing to reveal their SSN
- Business losses due to privacy violations
  - Online consumers worry about revealing personal data
  - This fear held back $15 billion in online revenue in 2001
- Federal Privacy Acts to protect privacy
  - E.g., Privacy Act of 1974 for federal agencies
    - Still many examples of privacy violations even by federal agencies
      - JetBlue Airways revealed travellers' data to federal gov't
  - E.g., Health Insurance Portability and Accountability Act of 1996 (HIPAA)

Privacy and Trust

- Privacy Problem
  - Consider computer-based interactions
    - From a simple transaction to a complex collaboration
  - Interactions involve *dissemination of private data*
    - It is voluntary, "pseudo-voluntary," or required by law
  - Threats of privacy violations result in lower trust
  - Lower trust leads to isolation and lack of collaboration
- Trust must be established
  - Data – provide quality an integrity
  - End-to-end communication – sender authentication, message integrity
  - Network routing algorithms – deal with malicious peers, intruders, security attacks

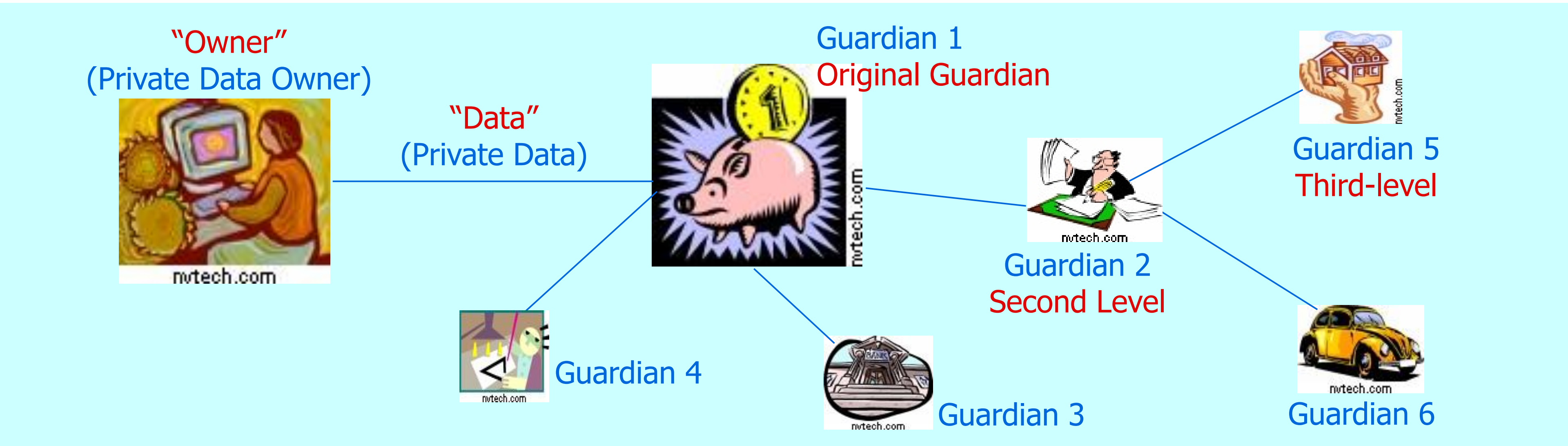
Fundamental Contributions

- Provide measures of privacy and trust
- Empower users (peers, nodes) to control privacy in ad hoc environments
  - Privacy of user identification
  - Privacy of user movement
- Provide privacy in data dissemination
  - Collaboration
  - Data warehousing
  - Location-based services
- Tradeoff between privacy and trust
  - *Minimal* privacy disclosures
    - Disclose private data absolutely necessary to gain a level of trust required by the partner system

Outline

1. Assuring privacy in data dissemination
2. Privacy-trust tradeoff
3. Privacy metrics

1. Privacy in Data Dissemination



- "Guardian:"

  Entity entrusted by private data owners with collection, storage, or transfer of their data
  - owner can be a guardian for its own private data
  - owner can be an institution or a system
- Guardians allowed or required by law to share private data
  - With owner's explicit consent
  - Without the consent as required by law
    - research, court order, etc.

Problem of Privacy Preservation

- Guardian passes private data to another guardian in a data dissemination chain
  - Chain within a graph (possibly cyclic)
- Owner privacy preferences *not* transmitted due to neglect or failure
  - Risk grows with chain length and milieu fallibility and hostility
- If preferences lost, receiving guardian unable to honor them

Challenges

- Ensuring that owner's metadata are never decoupled from his data
  - Metadata include owner's privacy preferences
- Efficient protection in a hostile milieu
  - Threats - examples
    - Uncontrolled data dissemination
    - Intentional or accidental data corruption, substitution, or disclosure
  - Detection of data or metadata loss
  - Efficient data and metadata recovery
    - Recovery by retransmission from the original guardian is most trustworthy

Proposed Approach

A. Design self-descriptive private objects

B. Construct a mechanism for apoptosis of private objects

   apoptosis = clean self-destruction

C. Develop proximity-based evaporation of private objects

A. Self-descriptive Private Objects

- Comprehensive metadata include:
  - owner's privacy preferences — How to read and write private data
  - guardian privacy policies — For the original and/or subsequent data guardians
  - metadata access conditions — How to verify and modify metadata
  - enforcement specifications — How to enforce preferences and policies
  - data provenance — Who created, read, modified, or destroyed any portion of data
  - context-dependent and other components — Application-dependent elements; Customer trust levels for different contexts; Other metadata elements

Notification in Self-descriptive Objects

- Self-descriptive objects simplify notifying owners or requesting their permissions
  - Contact information available in the *data provenance* component
- Notifications and requests sent to owners immediately, periodically, or on demand
  - Via pagers, SMSs, email, mail, etc.

Optimization of Object Transmission

- Transmitting *complete* objects between guardians is inefficient
  - They describe all foreseeable aspects of data privacy
    - For any application and environment
- Solution: prune transmitted metadata
  - Use application and environment semantics along the data dissemination chain

B. Apoptosis of Private Objects

- Assuring privacy in data dissemination
  - In benevolent settings:

    use *atomic* self-descriptive object with retransmission recovery
  - In malevolent settings:

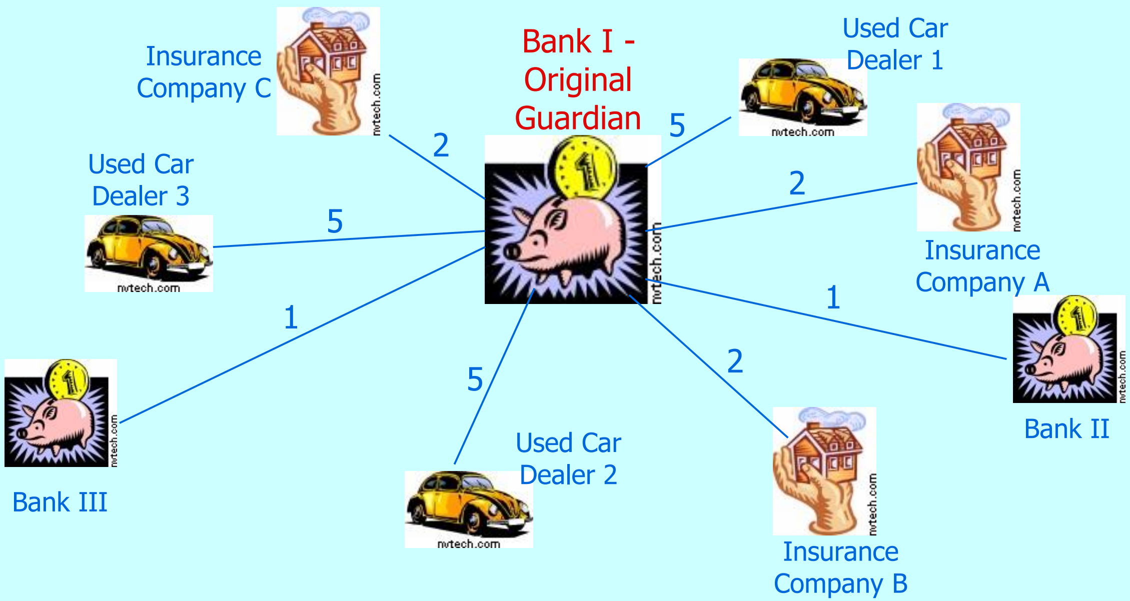
    when attacked object threatened with disclosure, use *apoptosis* (clean self-destruction)
- Implementation
  - Detectors, triggers, code
  - False positive
    - Dealt with by retransmission recovery
    - Limit repetitions to prevent denial-of-service attacks
  - False negatives

C. Proximity-based Evaporation of Private Data

- Perfect data dissemination not always desirable
  - Example: Confidential business data shared within an office but *not outside*
- Idea: Private data *evaporate* in proportion to their "distance" from their owner
  - "Closer" guardians trusted more than "distant" ones
  - Illegitimate disclosures more probable at less trusted "distant" guardians
  - Different distance metrics
    - Context-dependent

Examples of Metrics

- Examples of one-dimensional distance metrics
  - Distance ~ business type



If a bank is the original guardian, then:
-- any other *bank* is "closer" than any *insurance company*
-- any *insurance company* is "closer" than any *used car dealer*

  - Security/reliability as one of dimensions

Evaporation Implemented as Controlled Data Distortion

- Distorted data reveal less, protecting privacy
- Examples:

accurate | more and more distorted

accurate		more and more distorted		
250 N. Salisbury Street West Lafayette, IN	→	Salisbury Street West Lafayette, IN	→	somewhere in West Lafayette, IN
250 N. Salisbury Street West Lafayette, IN [home address]	→	250 N. University Street West Lafayette, IN [office address]	→	P.O. Box 1234 West Lafayette, IN [P.O. box]
765-123-4567 [home phone]	→	765-987-6543 [office phone]	→	765-987-4321 [office fax]


nvtech.com


nvtech.com

Evaporation as Apoptosis Generalization

- Context-dependent apoptosis for implementing evaporation
  - Apoptosis detectors, triggers, and code enable context exploitation
- Conventional apoptosis as a simple case of data evaporation
  - Evaporation follows a step function
    - Data self-destructs when proximity metric exceeds predefined threshold value

Outline

1. Assuring privacy in data dissemination
2. Privacy-trust tradeoff
3. Privacy metrics

2. Privacy-trust Tradeoff

- Problem
  - To build trust in open environments, users provide digital credentials that contain private information
  - How to gain a certain *level of trust* with the least *loss of privacy?*
- Challenges
  - Privacy and trust are fuzzy and multi-faceted concepts
  - The amount of privacy lost by disclosing a piece of information is affected by:
    - Who will get this information
    - Possible uses of this information
    - Information disclosed in the past

Proposed Approach

A. Formulate the privacy-trust tradeoff problem
B. Estimate privacy loss due to disclosing a set of credentials
C. Estimate trust gain due to disclosing a set of credentials
D. Develop algorithms that minimize privacy loss for required trust gain

A. Formulate Tradeoff Problem

- Set of private attributes that user wants to conceal
- Set of credentials
  - Subset of *revealed* credentials R
  - Subset of *unrevealed* credentials U
- Choose a subset of credentials NC from U such that:
  - NC satisfies the requirements for trust building
  - PrivacyLoss(NC+R) – PrivacyLoss(R) is minimized

Formulate Tradeoff Problem - cont.1

- If multiple private attributes are considered:
  - Weight vector $\{w_1, w_2, \ldots, w_m\}$ for private attributes
  - Privacy loss can be evaluated using:
    - The weighted sum of privacy loss for all attributes
    - The privacy loss for the attribute with the highest weight

B. Estimate Privacy Loss

- Query-independent privacy loss
  - Provided credentials reveal the value of a private attribute
  - User determines her private attributes
- Query-dependent privacy loss
  - Provided credentials help in answering a specific query
  - User determines a set of potential queries that she is reluctant to answer

Privacy Loss Estimation Methods

- Probability method
  - Query-independent privacy loss
    - Privacy loss is measured as the difference between entropy values
  - Query-dependent privacy loss
    - Privacy loss for a query is measured as difference between entropy values
    - Total privacy loss is determined by the weighted average
  - Conditional probability is needed for entropy evaluation
    - Bayes networks and kernel density estimation will be adopted
- Lattice method
  - Estimate query-independent loss
  - Each credential is associated with a tag indicating its privacy level with respect to an attribute a_j
  - Tag set is organized as a lattice
  - Privacy loss measured as the *least upper bound* of the privacy levels for candidate credentials

C. Estimate Trust Gain

- Increasing trust level
  - Adopt research on trust establishment and management
- Benefit function *B(trust_level)*
  - Provided by service provider or derived from user's utility function
- Trust gain
  - $B(trust\_level_{new})$ - $B(tust\_level_{prev})$

D. Minimize Privacy Loss for Required Trust Gain

- Can measure privacy loss (B) and can estimate trust gain (C)
- Develop algorithms that minimize privacy loss for required trust gain
  - User releases more private information
  - System's trust in user increases
  - How much to disclose to achieve a target trust level?