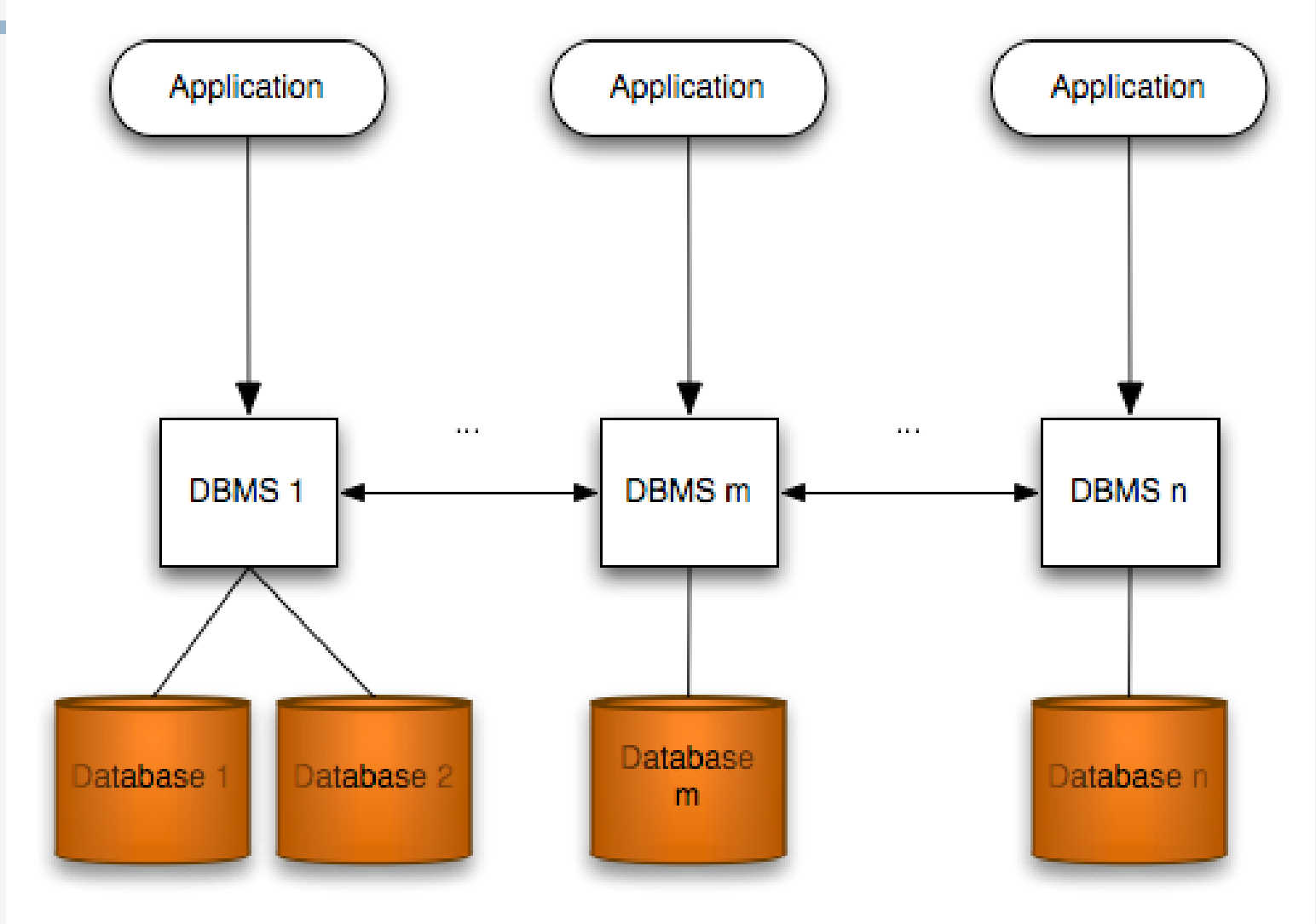
Introducing PATTY: Peer dATabase securiTY

Tony Young
M.Math Candidate
CS 856 - Fall 2004

Peer Database Systems

- Peer-to-peer (peer) architecture has recently been applied to database systems
  - Allows a node to act as a client for performing queries and a server to answer them
- Peer databases pose many implementation challenges

Peer Database Systems



Implementation Challenges

- Nodes may join and leave a cluster at any time and are not known to the cluster ahead of time
- The schema for a peer database is not global/homogeneous (i.e. several schemas might be used to represent the same data)

Implementation Challenges

- The data in a peer database system might not be complete (i.e. a group of peers might not have the complete set of information required to answer a query)
- Peer database systems must route queries to many nodes in order to receive a complete answer to their query Implementation Challenges

Application Areas

- Peer databases can be applied to
  - Development environment configuration management
  - Genomics
  - Healthcare
  - Contact information management
  - … Etc.

Important Aspects

- Availability
- Data Authenticity
- Performance
- Scalability
- Security
  - Authentication of Users
  - Data Encryption

Important Aspects

- My work focused on authentication and data encryption
- Referred to as “security services”
  - Not a complete set of “traditional” security services, but the phrase is used as an umbrella to cover both

Design Goals

- Authentication: PATTY seeks to provide a secure means of authentication of peers
  - After authentication, access rights can be granted to data, allowing private data to be shared in a peer manner
- Encryption: PATTY seeks to provide a secure means of communication and data storage
  - Encryption will allow PATTY to ensure that private data remains private

Design Goals

- Low Overhead: PATTY seeks to provide its security services with as little overhead as possible
  - Performance of user queries should not suffer because of the need to encrypt and decrypt data

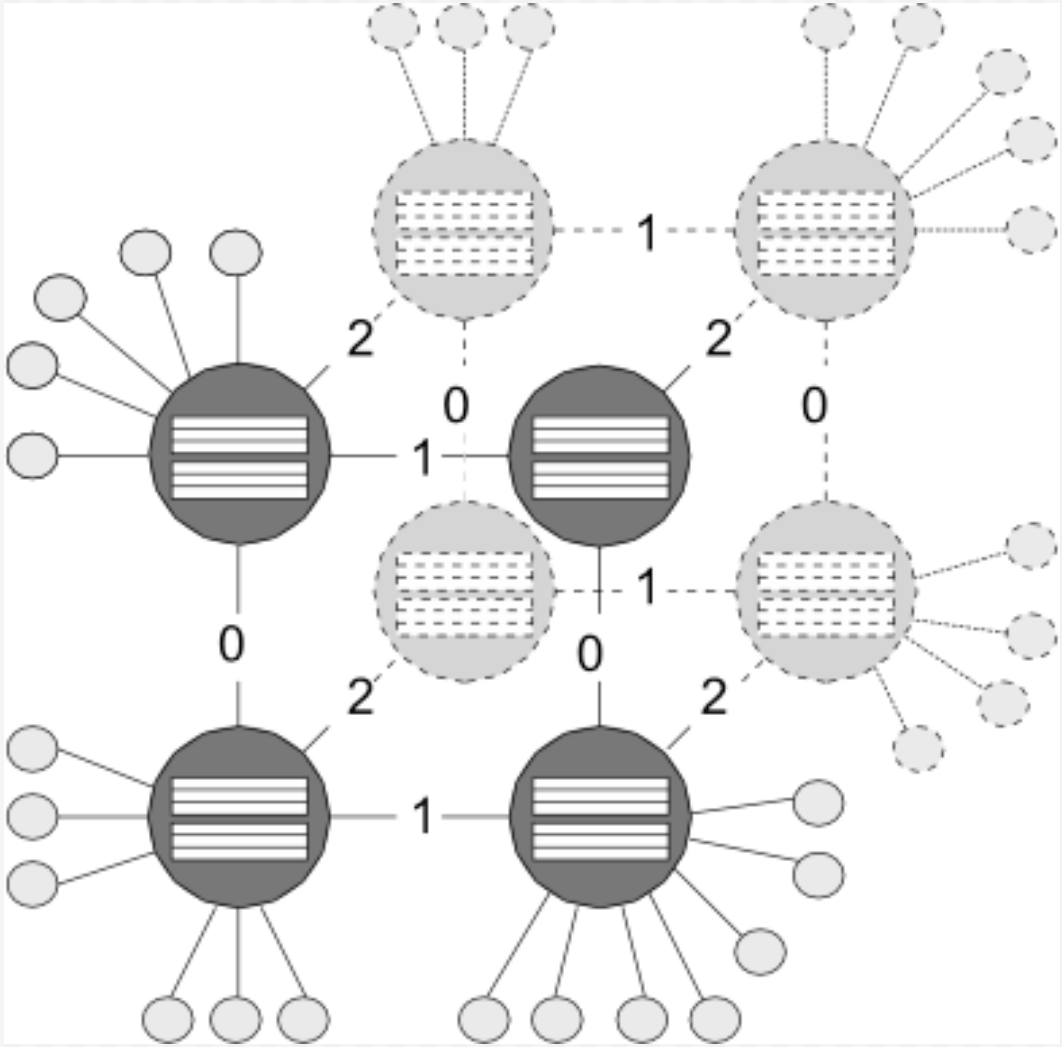
System Architecture

- Used the architecture from Edutella
  - It is most efficient in reducing the number of messages required to route queries
- Wolfgang Nejdl, Wolf Siberski, and Michael Sintek. Design Issues and Challenges for RDF and Schema-Based Peer-to-peer Systems. SIGMOD Rec., 32(3):41–46, 2003.

System Architecture

- Super-peers are used to provide bootstrapping services to peers
- Forward all messages between peer groups
  - I.e. peers only communicate with their super-peer, even when returning query results
- Use an edge forwarding protocol
  - Each edge between peers organized in a hypercube network has a monotonically increasing label
  - A received message is only forwarded along the edges with labels greater than the label of the edge on which the message arrived

System Architecture



Authentication

- Modified the decentralized approach I presented earlier
  - Requires the least overhead
  - Least susceptible to attacks
  - Easily extendable to peer systems (from distributed systems)
- Michael Kaminsky, George Savvides, David Mazieres, and M. Frans Kaashoek. Decentralized User Authentication in a Global File System. Proc. of the 19th ACM symposium on OS principles, pp. 60–73. ACM Press, 2003.

Authentication

- Each node can specify access rights to tuples, tables, stored procedures, etc. as is possible in any fully functional database management system
- Peers submit authentication information with their query requests
  - The receiving node authenticates the requesting node based on its cached user and group records

Authentication

Query Statement	The text of the query that the sender is attempting to pose to peers in the network
Authentication Information	The user name and password of the requesting peer, encrypted with the requesting peer's private key
Public Key	The public key of the requesting peer
IP Address	The IP address of the requesting peer, encrypted with the requesting peer's private key

Authentication

- When a peer submits a query to its super peer, it is encrypted with the super peer's public key
  - The message can then be read only by the super peer when it is decrypted using the super peer's private key
- The included authentication information can be decrypted by anyone
  - To authenticate, the information must first be encrypted with the requesting node's private key
  - Even if a peer knows another's user name and password, they cannot use it to perform impersonation attacks
  - The requesting peer's IP address is encrypted as well to prevent redirection attacks.

Authentication

- The super peer determines where to forward the query next (i.e. to other peers and/or super peers)
  - Request is encrypted with each new receiver's public key and transmitted
- When a query request message is received, the receiving peer decrypts it and extracts the authentication information
  - The authentication information is decrypted using the included public key

Authentication

- The receiving peer then looks up the record for the user with the provided user name
  - The password is verified, and the receiver ensures that the public key used to decrypt the authentication information matches the public key stored in the user record
- The IP address provided is decrypted with the stored public key
- The query is performed at the remote site and the result is encrypted with the public key of the requester
- Results are transmitted directly to the requester

Encryption

- Makes use of two types of encryption
  - All data that is transferred between peers is encrypted and decrypted using the RSA algorithm
  - All data that is stored on a peer's disk is encrypted using the IDEA
- These algorithm have been proven secure, and are computationally inexpensive to perform using existing hardware
- Using these methods, we argue that our data will be secure

Encryption

- Rivest, Shamir, Adleman (RSA)

- R. L. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. Commun. ACM, 26(1):96–99, 1983.

Encryption

- RSA is a public key cryptosystem
- Very secure
- Peer P's Public key is distributed and can be used to encrypt for sending to P or for decrypting data from P
- Peer P's Private key is kept secret and is used to encrypt for sending from P or for decrypting data sent to P

Encryption

- International Data Encryption Algorithm (IDEA)
- M.P. Leong, O.Y.H. Cheung, K.H. Tsoi, and P.H.W. Leong. A bit-serial implementation of the international data encryption. Field-Programmable Custom Computing Machines, 2000 IEEE Symposium on, 2000.

Encryption

- IDEA is a strong version of the Data Encryption Standard (DES)
  - DES uses too few bits to be secure with modern hardware
  - IDEA increases bits used from 56 to 128
  - IDEA uses 16 rounds
    - XOR operations and transformation functions applied to half of the data during each round
    - Half is rotated each round
    - A 56 bit subset of the 128 bit key is used each round

Metrics

- Encryption Time: How long does it take to encrypt a query request or reply?
- Decryption Time: How long does it take to decrypt a query request or reply?
- Authentication Time: How long does it take to authenticate a peer once the authentication information has been decrypted?
- Overhead: How much additional overhead is required (on top of “traditional” processing time) to encrypt, decrypt and authenticate?
- Data Access: How long does it take to encrypt and decrypt data in base tables?

Measurement

- Sample Queries: A batch of sample queries would be run on actual data
  - The queries would be run with encryption turned on, then run again with encryption turned off
  - In this manner, the overhead of the security services could be measured and recorded in a table such as follows
  - Comparisons can then be made as to the amount of extra overhead the security services impose

Measurement

Time (ms)	*Encryption ON*	*Encryption OFF*	Δ
Minimum			
Mean			
Maximum			

Measurement

- Encryption Test: A batch of queries and result sets would be encrypted and decrypted
  - In this manner, it would be possible to measure the overhead of encrypting and decrypting query results and query request messages.
  - Results would be recorded in a table such as follows
  - The amount of overhead PATTY imposes on communication can then be determined.

Measurement

Time (ms)	*Encryption*	*Decryption*
Minimum		
Mean		
Maximum		

Measurement

- Access Test: A batch of accesses to data tables would be made and recorded
  - In this manner, it would be possible to measure the overhead associated with decryption of table data, and encryption of inserted data
  - Results would be recorded in a table such as follows
  - The amount of overhead PATTY imposes on data accesses can then be determined

Measurement

Time (ms)	*Encryption*	*Decryption*
Minimum		
Mean		
Maximum		

Measurement

Time (ms)	*Encryption*	*Decryption*
Minimum		
Mean		
Maximum		

Attacks

- Interception: PATTY is designed to be impervious to interception attacks with the use of RSA encryption for communications
  - Since messages are encrypted with the receiver's public key, they can only be decrypted and read by the receiver (who holds the private key)

Attacks

- Impersonation: PATTY is impervious to impersonation attacks due to the robustness of the authentication protocol
  - Any peer may decrypt and read a peer's authentication information using their private key
  - The public key associated with a user name must be included in the query request message
  - The authentication information must be decrypted using that public key
  - The public key must match the one associated with the user in the peer's authentication database
  - Since it is not possible for a malicious peer to encrypt the authentication information using the requesting peer's private key, authentication will fail if any of the information is tampered with

Attacks

- Wear and Tear : PATTY may suffer from wear and tear attacks on encryption keys
  - Directly due to the number of messages that will be encrypted using these keys
  - Use of public-key cryptography reduces the number of messages that can be collected and analyzed for each key
- Replay: PATTY can fall victim to replay attacks by malicious peers
  - Query request messages can be resubmitted by a malicious peer
  - This type of attack is possible in traditional and distributed database systems as well

Attacks

- Redirection: PATTY is impervious to redirection attacks by malicious peers
  - Since the IP address of the requesting node is encrypted using their private key, the IP address cannot be modified and still properly decrypt
- Data Theft : Although it is possible to steal base table data from a PATTY node, the data will not be readable as it is encrypted
  - If the encryption key is also stolen, data would be readable by malicious peers

Attacks

- Denial of Service: It is possible to perform denial of service attacks in a PATTY network
  - Many requests could be submitted to a super peer at once in order to refuse messages from legitimate peers
  - Since there are many super peers in the network, the network will remain usable and connected if a super peer fails.

Milestones

- Things I will achieve
  - Complete a thorough literature survey of peer database systems
  - Complete a thorough literature survey of authentication and encryption protocols
  - Complete a critical analysis of some selected peer database systems
  - Complete a critical analysis of some selected authentication and encryption protocols
- Complete!

Milestones

- Things I plan to achieve
  - Complete a discussion of how selected authentication and encryption protocols might be applied, and how well they might perform with, selected peer database systems
  - Propose an experiment to implement some authentication and encryption protocols on top of some peer database systems as well as metrics to measure how well the systems perform under attack
- Complete!

Milestones

- Things I might achieve
  - Complete a test implementation of some authentication and encryption services in a peer database system and determine if the security they provide is adequate to safeguard critical documents
- Ran out of time :(

Conclusions

- Much work still needs to be done in order to determine if the third design goal, low overhead, has been met
  - Performance testing with benchmark data can be performed to determine this
- Presented a secure protocol for authentication of peers in a peer database system
  - Combines and extends several approaches to obtain a good set of security services

Conclusions

- Protocol is impervious to several attacks, but still permits some attacks to affect availability
- System scales well in number of messages due directly to the efficiency of the routing algorithm

Questions?