K.Nagaraju *et al*, International Journal of Computer Science and Mobile Applications,
Vol.3 Issue. 12, December- 2015, pg. 1-7 **ISSN: 2321-8363**

A TRUSTED HARDWARE BASED DATA BASE WITH FILLED SECURITY

K.Nagaraju[1], A. Ravindra Kumar[2] M.Tech, MISTE, K.Prakash[3] M.Tech

1{M.Tech Student, C.S.E Department, Kuppam Engineering College, Andhra Pradesh, India}

2{Head Of the Department C.S.E, Kuppam Engineering College, Andhra Pradesh, India}

3{Asst. Professor, C.S.E, Kuppam Engineering College, Andhra Pradesh, India}

ABSTRACT:- The TrustedDB present an outsourced database model that permits end users to execute SQL inquiries with protection and under administrative consistence requirements by utilizing server-facilitated, sealed trusted equipment in basic inquiry execution restrictions of trusted equipment, Customarily then sent, for server-side inquiry preparing on the scrambled information, intrinsically farthest point question expressiveness. , when classification turns into a worry, information is scrambled before outsourcing to an administration supplier. Any product based cryptographic develops preparing stages, in this manner uprooting any confinements on the sort of bolstered inquiries. Here, we present Trusted DB, an outsourced database model that permits customers to execute SQL questions with security and under administrative consistence imperatives by utilizing server-facilitated, carefully designed trusted equipment in basic question we demonstrate that the expenses per question are requests of extent lower than any (current or) potential future programming just components. TrustedDB is based and keeps running on real equipment and its execution and expenses are assessed here.

1. INTRODUCTION.

Significant challenges lie in the path of large-scale adoption. Such [1] services often require their customers to inherently trust the provider with full access to the outsourced datasets. But numerous instances of illicit insider behavior or data leaks have left clients reluctant to place sensitive data under the control of a remote, third-party provider, without practical assurances of privacy and confidentiality – especially in business, healthcare and government. Most of the existing research efforts have addressed such outsourcing security aspects by encrypting the data before outsourcing. Once encrypted however, inherent limitations in the types of primitive operations that can be performed on encrypted data lead to fundamental expressiveness and practicality constraints. Recent theoretical cryptography results provide

hope by proving the existence of universal homomorphisms, i.e., encryption mechanisms that allow computation of arbitrary functions without decrypting the inputs [6]. Unfortunately actual instances of such mechanisms seem to be decades Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Articles from this volume were invited to present their results at The 37th International Conference on Very Large Data Bases, August 29th - September 3rd 2011, Seattle, Washington.

Traditionally, as soon as confidentiality becomes a concern, data is encrypted before outsourcing to a service provider. Any software-based cryptographic constructs then deployed, for server-side query processing on the encrypted data, inherently limit query expressiveness. Here, we introduce TrustedDB, an outsourced database prototype that allows clients to execute SQL queries with privacy and under regulatory compliance constraints by leveraging server-hosted, tamper-proof trusted hardware in critical query processing stages, thereby removing any limitations on the type of supported queries. Despite the cost overhead and performance limitations of trusted hardware, we show that the costs per query are orders of magnitude lower than any (existing or) potential future software-only mechanisms. TrustedDB is built and runs on actual hardware, and its performance and costs are evaluated here.

Recent theoretical cryptography results provide hope by proving the existence of universal homeomorphisms [2], i.e., encryption mechanisms that allow computation of arbitrary functions without decrypting the inputs. Unfortunately actual instances of such mechanisms seem to be decades away from being practical. Ideas have also been proposed to leverage tamper-proof hardware to privately process data server-side, ranging from smart-card deployment in healthcare, to more general database operations.

Yet, common wisdom so far has been that trusted hardware is generally impractical due to its performance limitations and higher acquisition costs. As a result, with very few exceptions, these efforts have stopped short of proposing or building full - fledged database processing engines.

2. RELATED WORK.

The Caesar cipher is simple, but not secure. We believe that conventional public-key encryption schemes with modular exponentiations are secure, but modular exponentiation is not a very simple operation. If we were to forget our current schemes and start from scratch, perhaps something like the following scheme would be a good candidate for a simple symmetric encryption scheme.

Outsourcing has [3] finally arrived, due in no small part to the availability of cheap high speed networks, storage and CPUs. Clients can now minimize their management overheads and virtually eliminate infrastructure costs1. Virtually all major “cloud” providers today offer a database service of some

kind as part of their overall solution. Discussing the merits or faults of outsourcing and "clouds" is beyond the scope of this paper. Others have done and continue to do. Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Athens, Greece, ours startups also feature more targeted data management and/or database platforms. Yet, significant challenges lie in the path of large-scale adoption. Such services often require their customers to inherently trust the provider with full access to the outsourced datasets. But numerous instances of illicit insider behavior or data leaks have left clients reluctant to place sensitive data under the control of a remote, third-party provider, without practical assurances of privacy and confidentiality – especially in business, healthcare and government frameworks. And today's privacy guarantees of such services are at best declarative and subject customers to unreasonable fine-print clauses – e.g., allowing the server operator (or malicious attackers gaining access to its systems) to use customer behavior and content for commercial, profiling, or governmental surveillance purposes . Existing research addresses several such outsourcing security aspects, including access privacy, searches on encrypted data, range queries, and aggregate queries. To achieve privacy, in most of these efforts data is encrypted before outsourcing. Once encrypted however, inherent limitations in the types of primitive operations that can be performed on encrypted data lead to fundamental expressiveness and practicality constraints. Recent theoretical cryptography results provide hope by proving the existence of universal homomorphisms, i.e., encryption mechanisms that allow computation of arbitrary functions without decrypting the inputs. Unfortunately actual instances of such mechanisms seem to be decades away from being practical

3. SYSTETEM ANAALYSIS.

3.1 ARCHITECTURE

TrustedDB [3, 11]is built around a set of core components including a request handler, a processing agent and communication conduit, a query parser, a paging module, a query dispatch module, a cryptography library, and two database engines. While presenting a detailed architectural blueprint is not possible in this space, in the following we discuss some of the key elements and challenges faced in designing and building TrustedDB.

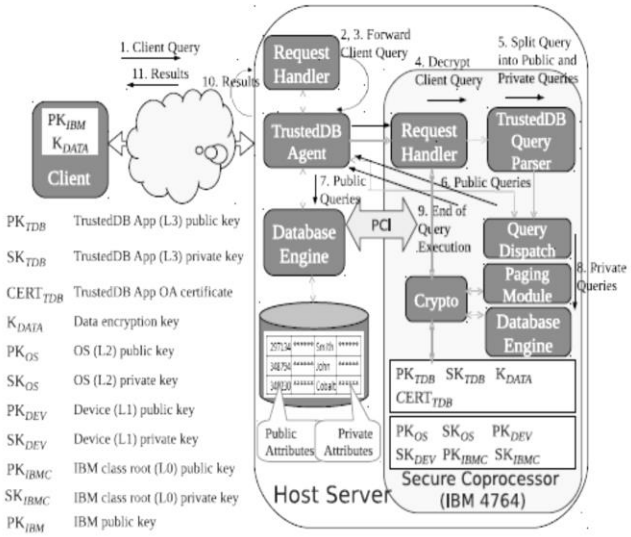

Fig 3.1 TRUSTED DB ARCHITECTURE.

3.1.1 Outline

Challenges. The IBM 4764 SCPU presents significant challenges in designing and deploying custom code to be run within its enclosure. For strong security, the underlying hardware code as well as the OS are embedded and no hooks are possible e.g., to augment virtual memory and paging mechanisms. We were faced with the choice of having to provide virtual memory and paging in user land, specifically inside the query processor as well as all the support soft-ware. The embedded Linux OS is a Motorola PowerPC 405 port with stripped down libraries required to support the IBM cryptography codebase and nothing else. This constituted a significant hurdle, as cross-compilation became a complex task of mixing native logic with custom-ported functionality. The SCPU communicates with the outside world synchronously through fixed sized messages exchanged over the PCI-X bus in exact sequences. Interfacing such a synchronous channel with the communication model of the query processors and associated paging components required the development of the TrustedDB Paging Module. The SCPU's cryptographic hardware engine features a set of latencies that effectively crippled the ability to run for highly interactive mechanisms manipulating small amounts of data (e.g., 32 bit integers). To handle this we ported several cryptographic primitives to be run on the SCPU's.

4. RESULTS.

4.1 Query Parsing

We propose a cost effective model [10] to design the more security filled hard ware based data base. Sensitive attributes can occur anywhere within a query (e.g., in SELECT, WHERE or GROUP-BY clauses, in aggregation operators, or within sub-queries). The Query Parser's job is then: • To ensure that any processing involving private at- tributes is done within the SCPU. All private attributes are encrypted using a shared data encryption keys between the client and the SCPU hence the host server cannot decipher these attributes. To optimize the rewrite of the client query such that most of the work is performed on the host server. This significantly increases performance. To exemplify how public and private queries are generated from the original client query we use examples from the TPC-H bench mark. TPC-H does not specify any classification of attributes based on security. Therefore, we define a attribute set classification into private (encrypted) and public (non-encrypted) [4,5]. The resultant schema is listed 6. In brief, all attributes that convey identifying information about Customers, Suppliers and Parts are considered private. The resulting query plans (including rewrites into main CPU and SCPU components) for TPC-H queries Q3, Q4, and Q6 are illustrated in Aggregation Example. For queries that have WHERE clause conditions on public attributes, the server can first SELECT all the tuples that meet the criteria. The private attributes' queries are then performed inside the SCPU on these intermediate results, to yield the final result. For e.g., query Q6 of the TPC-H benchmark is processed. The host server first executes a public query that filters all tuples which fall within the desired ship date and quantity range, both of these being public attributes. The result from this public query is then used by the SCPU to perform the aggregation on the private at-tributes extended price and discount. While performing the aggregation the private attributes are decrypted inside the SCPU. Since the aggregation operation results in a new attribute composing of private attributes it is re-encrypted before sending to the client. This encryption is also done within the SCPU. Note that the execution of private queries depends on the results from the execution of public queries and vice-a-versa even though they execute in separate database engines.

This is made possible by the TrustedDB [8] Query Dispatcher in con-junction with the Paging Module. Grouping Example. If the client query specifies a GROUP or ORDER BY on public attributes but the selection includes an aggregation of the private attributes, the grouping or sort operation is performed inside the SCPU. Illustrates this for the TPC-H query Q3. If the aggregation did not involve any private attributes then the host server performs all the GROUP BY and sorting operations. Nested Queries. The

case of nested queries is similar, yet additional care should be taken when computing execution plans to limit the amount of data transfer between the host server and the SCPU which may result in sub-optimal performance. One such example is query Q4 of the TPC-H [12] benchmark which includes a sub-query on a private attribute.

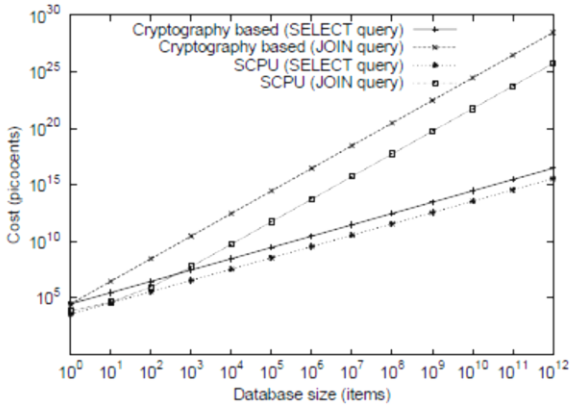

Fig 5. Cost effective Query Optimization.

TrustedDB on the other hand utilizes secure, tamper resistant hardware such as the IBM 4764/5 [4,5] cryptographic coprocessors deployed on the service provider's side to implement a complete SQL database processing engine. The TrustedDB design provides strong data confidentiality assurances. Moreover, it does not limit query expressiveness.

5. CONCLUSION.

This work's inherent that, at scale, in outsourced contexts, computation inside secure hardware processors is orders of magnitude cheaper than equivalent cryptography performed on provider's unsecured server hardware, despite the overall greater acquisition cost of secure hardware. We thus propose to make trusted hardware a first-class citizen in the secure data management arena. Moreover, we hope that cost-centric insights and architectural paradigms will fundamentally change the way systems and algorithms are designed. We propose a cost effective model here to design the more security filled hard ware based data base. Sensitive attributes can occur anywhere within a query.

REFERENCES.

[1] Sumeet Bajaj and Radu Sion. TrustedDB: A Trusted Hardware based Outsourced Database Engine. VLDB, DEMO, 2011.
[2] Marten van Dijk, Craig Gentry, Shai Halevi, and Vinod Vaikuntanathan. Fully homomorphic encryption over the integers. In Henri Gilbert, editor, EUROCRYPT, volume 6110 of Lecture Notes in Computer Science, pages 24–43. Springer, 2010.
[3] TrustedDB: A Trusted Hardware based Database with Privacy and Data Confidentiality
[4]IBM 4764 PCI-X Cryptographic oprocessor. Online at http://www-03.ibm.com/security/cryptocards/pcixcc/overview.shtml, 2007.
[5] IBM 4765 PCIe Cryptographic Coprocessor. Onlineathttp://www-03.ibm.com/security/cryptocards/pciecc/overview.shtml, 2010.
[6] Mihir Bellare. New proofs for nmac and hmac: Security without collision-resistance. pages 602–619. Springer-Verlag, 2006.
[7] FIPS PUB 140-2, Ssecurity Requirements for Cryptographic Modules. Online at http://csrc.nist.gov/groups/STM/cmvp/ standards.html#02.
[8]TPC-H Benchmark. Online at http://www.tpc.org/tpch/.
[9] IBM 4764 PCI-X Cryptographic Coprocessor. Online at http://www-03.ibm.com/security/cryptocards/pcixcc/overview.shtml, 2007.
[10] Gagan Aggarwal, Mayank Bawa, Prasanna Ganesan, Hector Garcia-Molina, Krishnaram Kenthapadi, Rajeev Motwani, Utkarsh Srivastava, Dilys Thomas, and Ying Xu 0002. Two can keep a secret: A distributed architecture for secure database services. In CIDR, pages 186–199, 2005.
[11] Alexander Iliev and Sean WSmith. Protecting Client Privacy with Trusted Computing at the Server. IEEE, Security and Privacy, 3(2), Apr 2005.
[12] Mihir Bellare. New proofs for nmac and hmac: Security without collision-resistance. pages 602–619. Springer-Verlag, 2006.

ABOUT THE AUTHORS

1. K.Nagaraju M.Tech student in the Department of Computer Science and Engineering, Kuppam Engineering College, Kuppam, Andhra Pradesh India. *E-mail:*smile2saynaga@gmail.com

2. A.Ravindra Kumar **M.Tech, MISTE.** Head Of the Department in Computer Science and Engineering, Associate Professor, Kuppam Engineering College, Kuppam, Andhra Pradesh India.
E-mail: avula.ravindra1981@gmail.com

3. K.Prakash **M.Tech, IAENG.** Assistant Professor in Computer Science and Engineering Department, Associate Professor, Kuppam Engineering College, Kuppam, Andhra Pradesh India.
E-mail: prakashkrmkp@gmail.com

