INTERNATIONAL JOURNAL OF RESEARCH IN COMPUTER APPLICATION & MANAGEMENT

A Monthly Double-Blind Peer Reviewed (Refereed/Juried) Open Access International e-Journal - Included in the International Serial Directories

Indexed & Listed at:

Ulrich's Periodicals Directory ©, ProQuest, U.S.A., EBSCO Publishing, U.S.A., Cabell's Directories of Publishing Opportunities, U.S.A.,
Open J-Gage, India [link of the same is duly available at Inflibnet of University Grants Commission (U.G.C.)],
Index Copernicus Publishers Panel, Poland with IC Value of 5.09 & number of libraries all around the world.
Circulated all over the world & Google has verified that scholars of more than 2980 Cities in 165 countries/territories are visiting our journal on regular basis.

Ground Floor, Building No. 1041-C-1, Devi Bhawan Bazar, JAGADHRI – 135 003, Yamunanagar, Haryana, INDIA

http://ijrcm.org.in/

CONTENTS

CALL FOR MANUSCRIPTS

We invite unpublished novel, original, empirical and high quality research work pertaining to recent developments & practices in the areas of Computer Science & Applications; Commerce; Business; Finance; Marketing; Human Resource Management; General Management; Banking; Economics; Tourism Administration & Management; Education; Law; Library & Information Science; Defence & Strategic Studies; Electronic Science; Corporate Governance; Industrial Relations; and emerging paradigms in allied subjects like Accounting; Accounting Information Systems; Accounting Theory & Practice; Auditing; Behavioral Accounting; Behavioral Economics; Corporate Finance; Cost Accounting; Econometrics; Economic Development; Economic History; Financial Institutions & Markets; Financial Services; Fiscal Policy; Government & Non Profit Accounting; Industrial Organization; International Economics & Trade; International Finance; Macro Economics; Micro Economics; Rural Economics; Co-operation; Demography: Development Planning; Development Studies; Applied Economics; Development Economics; Business Economics; Monetary Policy; Public Policy Economics; Real Estate; Regional Economics; Political Science; Continuing Education; Labour Welfare; Philosophy; Psychology; Sociology; Tax Accounting; Advertising & Promotion Management; Management Information Systems (MIS); Business Law; Public Responsibility & Ethics; Communication; Direct Marketing; E-Commerce; Global Business; Health Care Administration; Labour Relations & Human Resource Management; Marketing Research; Marketing Theory & Applications; Non-Profit Organizations; Office Administration/Management; Operations Research/Statistics; Organizational Behavior & Theory; Organizational Development; Production/Operations; International Relations; Human Rights & Duties; Public Administration; Population Studies; Purchasing/Materials Management; Retailing; Sales/Selling; Services; Small Business Entrepreneurship; Strategic Management Policy; Technology/Innovation; Tourism & Hospitality; Transportation Distribution; Algorithms; Artificial Intelligence; Compilers & Translation; Computer Aided Design (CAD); Computer Aided Manufacturing; Computer Graphics; Computer Organization & Architecture; Database Structures & Systems; Discrete Structures; Internet; Management Information Systems; Modeling & Simulation; Neural Systems/Neural Networks; Numerical Analysis/Scientific Computing; Object Oriented Programming; Operating Systems; Programming Languages; Robotics; Symbolic & Formal Logic; Web Design and emerging paradigms in allied subjects.

Anybody can submit the **soft copy** of unpublished novel; original; empirical and high quality **research work/manuscript** ***anytime*** in ***M.S. Word format*** after preparing the same as per our **GUIDELINES FOR SUBMISSION**; at our email address i.e. infoijrcm@gmail.com or online by clicking the link **online submission** as given on our website (***FOR ONLINE SUBMISSION, CLICK HERE***).

GUIDELINES FOR SUBMISSION OF MANUSCRIPT

1. **COVERING LETTER FOR SUBMISSION:**

DATED: _____________

THE EDITOR
IJRCM

Subject: **SUBMISSION OF MANUSCRIPT IN THE AREA OF .**

(e.g. Finance/Marketing/HRM/General Management/Economics/Psychology/Law/Computer/IT/Engineering/Mathematics/other, please specify)

DEAR SIR/MADAM

Please find my submission of manuscript entitled '____' for possible publication in your journals.

I hereby affirm that the contents of this manuscript are original. Furthermore, it has neither been published elsewhere in any language fully or partly, nor is it under review for publication elsewhere.

I affirm that all the author (s) have seen and agreed to the submitted version of the manuscript and their inclusion of name (s) as co-author (s).

Also, if my/our manuscript is accepted, I/We agree to comply with the formalities as given on the website of the journal & you are free to publish our contribution in any of your journals.

NAME OF CORRESPONDING AUTHOR:
Designation:
Affiliation with full address, contact numbers & Pin Code:
Residential address with Pin Code:
Mobile Number (s):
Landline Number (s):
E-mail Address:
Alternate E-mail Address:

NOTES:

a) The whole manuscript is required to be in ***ONE MS WORD FILE*** only (pdf. version is liable to be rejected without any consideration), which will start from the covering letter, inside the manuscript.
b) The sender is required to mentionthe following in the **SUBJECT COLUMN** of the mail:
New Manuscript for Review in the area of (Finance/Marketing/HRM/General Management/Economics/Psychology/Law/Computer/IT/Engineering/Mathematics/other, please specify)
c) There is no need to give any text in the body of mail, except the cases where the author wishes to give any specific message w.r.t. to the manuscript.
d) The total size of the file containing the manuscript is required to be below **500 KB**.
e) Abstract alone will not be considered for review, and the author is required to submit the complete manuscript in the first instance.
f) The journal gives acknowledgement w.r.t. the receipt of every email and in case of non-receipt of acknowledgment from the journal, w.r.t. the submission of manuscript, within two days of submission, the corresponding author is required to demand for the same by sending separate mail to the journal.

2. **MANUSCRIPT TITLE**: The title of the paper should be in a 12 point Calibri Font. It should be bold typed, centered and fully capitalised.

3. **AUTHOR NAME (S) & AFFILIATIONS**: The author (s) **full name**, **designation**, **affiliation** (s), **address**, **mobile/landline numbers**, and **email/alternate email address** should be in italic & 11-point Calibri Font. It must be centered underneath the title.

4. **ABSTRACT**: Abstract should be in fully italicized text, not exceeding 250 words. The abstract must be informative and explain the background, aims, methods, results & conclusion in a single para. Abbreviations must be mentioned in full.

5. **KEYWORDS**: Abstract must be followed by a list of keywords, subject to the maximum of five. These should be arranged in alphabetic order separated by commas and full stops at the end.

6. **MANUSCRIPT**: Manuscript must be in ***BRITISH ENGLISH*** prepared on a standard A4 size ***PORTRAIT SETTING PAPER***. It must be prepared on a single space and single column with 1" margin set for top, bottom, left and right. It should be typed in 8 point Calibri Font with page numbers at the bottom and centre of every page. It should be free from grammatical, spelling and punctuation errors and must be thoroughly edited.

7. **HEADINGS**: All the headings should be in a 10 point Calibri Font. These must be bold-faced, aligned left and fully capitalised. Leave a blank line before each heading.

8. **SUB-HEADINGS**: All the sub-headings should be in a 8 point Calibri Font. These must be bold-faced, aligned left and fully capitalised.

9. **MAIN TEXT**: The main text should follow the following sequence:

INTRODUCTION

REVIEW OF LITERATURE

NEED/IMPORTANCE OF THE STUDY

STATEMENT OF THE PROBLEM

OBJECTIVES

HYPOTHESES

RESEARCH METHODOLOGY

RESULTS & DISCUSSION

FINDINGS

RECOMMENDATIONS/SUGGESTIONS

CONCLUSIONS

SCOPE FOR FURTHER RESEARCH

ACKNOWLEDGMENTS

REFERENCES

APPENDIX/ANNEXURE

It should be in a 8 point Calibri Font, single spaced and justified. The manuscript should preferably not exceed ***5000 WORDS***.

10. **FIGURES &TABLES**: These should be simple, crystal clear, centered, separately numbered &self explained, and **titles must be above the table/figure**. **Sources of data should be mentioned below the table/figure**. It should be ensured that the tables/figures are referred to from the main text.

11. **EQUATIONS**:These should be consecutively numbered in parentheses, horizontally centered with equation number placed at the right.

12. **REFERENCES**: The list of all references should be alphabetically arranged. The author (s) should mention only the actually utilised references in the preparation of manuscript and they are supposed to follow **Harvard Style of Referencing**. The author (s) are supposed to follow the references as per the following:

- All works cited in the text (including sources for tables and figures) should be listed alphabetically.
- Use (**ed.**) for one editor, and (**ed.s)** for multiple editors.
- When listing two or more works by one author, use --- (20xx), such as after Kohl (1997), use --- (2001), etc, in chronologically ascending order.
- Indicate (opening and closing) page numbers for articles in journals and for chapters in books.
- The title of books and journals should be in italics. Double quotation marks are used for titles of journal articles, book chapters, dissertations, reports, working papers, unpublished material, etc.
- For titles in a language other than English, provide an English translation in parentheses.
- The location of endnotes within the text should be indicated by superscript numbers.

PLEASE USE THE FOLLOWING FOR STYLE AND PUNCTUATION IN REFERENCES:

BOOKS

- Bowersox, Donald J., Closs, David J., (1996), "Logistical Management." Tata McGraw, Hill, New Delhi.
- Hunker, H.L. and A.J. Wright (1963), "Factors of Industrial Location in Ohio" Ohio State University, Nigeria.

CONTRIBUTIONS TO BOOKS

- Sharma T., Kwatra, G. (2008) Effectiveness of Social Advertising: A Study of Selected Campaigns, Corporate Social Responsibility, Edited by David Crowther & Nicholas Capaldi, Ashgate Research Companion to Corporate Social Responsibility, Chapter 15, pp 287-303.

JOURNAL AND OTHER ARTICLES

- Schemenner, R.W., Huber, J.C. and Cook, R.L. (1987), "Geographic Differences and the Location of New Manufacturing Facilities," Journal of Urban Economics, Vol. 21, No. 1, pp. 83-104.

CONFERENCE PAPERS

- Garg, Sambhav (2011): "Business Ethics" Paper presented at the Annual International Conference for the All India Management Association, New Delhi, India, 19–22 June.

UNPUBLISHED DISSERTATIONS AND THESES

- Kumar S. (2011): "Customer Value: A Comparative Study of Rural and Urban Customers," Thesis, Kurukshetra University, Kurukshetra.
- **ONLINE RESOURCES**
- Always indicate the date that the source was accessed, as online resources are frequently updated or removed.

WEBSITES

- Garg, Bhavet (2011): Towards a New Natural Gas Policy, Political Weekly, Viewed on January 01, 2012 http://epw.in/user/viewabstract.jsp

DATA PROTECTION IN CLOUD COMPUTING

CHENNA LAKSHMI
ASST. PROFESSOR
DEPARTMENT COMPUTER SCIENCE & ENGINEERING
RGM COLLEGE OF ENGINEERING & TECHNOLOGY
NANDYAL

ABSTRACT

The data-protection-as-a-service cloud platform architecture dramatically reduces the per-application development effort required to offer data protection while still allowing rapid development and maintenance. Cloud computing enables highly scalable services to be easily consumed over the Internet on an as-needed basis. A major feature of the cloud services is that users' data are usually processed remotely in unknown machines that users do not own or operate. Offering strong data protection to cloud users while enabling rich applications is a challenging task. We explore a new cloud platform architecture called Data Protection as a Service, which dramatically reduces the per-application development effort required to offer data protection, while still allowing rapid development and maintenance. A recent Microsoft survey found that "58 percent of the public and 86 percent of business leaders are excited about the possibilities of cloud computing. But more than 90 percent of them are worried about security, availability, and privacy of their data as it rests in the cloud." Protecting user data while enabling rich computation requires both specialized expertise and resources, which might not be readily available to most application developers.

KEYWORDS

cloud computing, DPaaS, security.

INTRODUCTION

Offering strong data protection to cloud users while enabling rich applications is a challenging task. Researchers explore a new cloud platform architecture called Data Protection as a Service, which dramatically reduces the per-application development effort required to offer data protection, while still allowing rapid development and maintenance.

Although cloud computing promises lower costs, rapid scaling, easier maintenance, and service availability anywhere, anytime, a key challenge is how to ensure and build confidence that the cloud can handle user data securely. A recent Microsoft survey found that "58 percent of the public and 86 percent of business leaders are excited about the possibilities of cloud computing. But more than 90 percent of them are worried about security, availability, and privacy of their data as it rests in the cloud."1 This tension makes sense: users want to maintain control of their data, but they also want to benefit from the rich services that application developers can provide using that data. So far, the cloud offers little platform-level support or standardization for user data protection beyond data encryption at rest, most likely because doing so is nontrivial. Protecting user data while enabling rich computation requires both specialized expertise and resources that might not be readily available to most application developers. Building in data-protection solutions at the platform layer is an attractive option: the platform can achieve economies of scale by amortizing expertise costs and distributing sophisticated security solutions across different applications and their developers.

We propose a new cloud computing paradigm, data protection as a service (www.mydatacontrol.com). DPaaS is a suite of security primitives offered by a cloud platform, which enforces data security and privacy and offers evidence of privacy to data owners, even in the presence of potentially compromised or malicious applications.

SECURITY AND PRIVACY CHALLENGES

It's impossible to develop a single data-protection solution for the cloud because the term means too many different things. Any progress must first occur in a particular domain ccordingly, our work focuses on an important class of widely used applications that includes e-mail, personal financial management, social networks, and business tools such as word processors and spreadsheets. The following criteria define this class of applications:

- provide services to a large number of distinct end users, as opposed to bulk data processing or workflow management for a single entity;
- Use a data model consisting mostly of sharable units, where all data objects have access control lists (ACLs) with one or more users; and
- Developers could run the applications on a separate computing platform that encompasses the physical infrastructure, job scheduling, user authentication, and the base software environment, rather than implementing the platform themselves.

INTRODUCING DPAAS

Building in data-protection solutions at the platform layer is an attractive option. Data Protection as a Service (DPaaS):

- Can achieve economies of scale by amortizing expertise costs and distributing sophisticated security solutions across different applications and their developers
- Enforces fine-grained access control policies on data units through application confinement and information flow checking
- Employs cryptographic protections at rest and offers robust logging and auditing to provide accountability
- Addresses the issues of rapid development and maintenance.

DPaaS, if offered by cloud platform providers in addition to their existing hosting environment, could be especially beneficial for small companies who don't have much in-house security expertise.

ENCRYPTION: HOW DPAAS SCORES OVER OTHER TECHNIQUES

In terms of encryption, the two prominent techniques - full-disk encryption (FDE) and fully homomorphic encryption (FHE) – fail to provide a practical solution in a cloud computing setup.

PARAMETER	FDE	FHE
Key management	Ideal for physical attacks; does not prevent leakage of data on account of online attacks	Users own the FHE encryption keys; does not address the challenge of storing the keys securely
Sharing	Key granularity does not line up with access control granularity; sharing is, therefore, not foolproof	With users holding and managing the keys, access control is a challenge
Aggregation	Users fully trust the cloud; this makes aggregation easier	Does not readily allow computing on data encrypted under different keys; aggregation is, therefore, a challenge
Performance	When implemented on disk firmware, can avoid slowdown	Not yet efficient enough for deploying on scale
Ease of development	No impact on application development	Developers cannot look at the data, making debugging, testing and improvements difficult

The DPaaS approach is better suited for the target cloud applications because it falls between the two. It keeps the “natural” granularity of FHE by keying on units of sharable data and maintains the performance of FDE by using symmetric encryption. It moves key management and access control to a middle tier—the computing platform—to balance rapid development and easy maintenance with user-side verifiability.

MAINTAINING DATA INTEGRITY ON THE CLOUD

Access controls, authorization, and auditing capability are common challenges for application developers. Incorporating these features within the platform is a significant improvement in terms of ease of use. DPaas uses a combination of encryption at rest, application confinement, information flow checking and auditing to ensure the security and privacy of users’ data.

AUTHENTICATION AND AUTHORIZATION

DPaaS can guarantee the integrity of the data at rest via cryptographic authentication of the data in storage and by auditing the application code at runtime. DPaaS can accomplish user authentication either with a proprietary approach or using open standards such as OpenID and OAuth.

AUDITING

The DPaaS approach provides logging and auditing at the platform level, sharing the benefits with all applications running on top. Because the platform mediates all data access, authenticates users and runs binaries, it knows what data is accessed by what user and with which application. It can generate meaningful audit logs containing all these parameters and optionally incorporate additional information from the application layer.

Given its ability to perform different types of audit, DPaaS can also support third-party auditing services, thus helping users understand how their data has been accessed and manipulated, and which services to trust.

THE WAY FORWARD

As private data moves online, the need to secure it properly becomes increasingly urgent. The good news is that the same forces concentrating data in enormous datacenters will also aid in using collective security expertise more effectively. Adding protections to a single cloud platform can immediately benefit hundreds of thousands of applications and, by extension, hundreds of millions of users.

DATA PROTECTION AS A SERVICE

Currently, users must rely primarily on legal agreements and implied economic and reputational harm as a proxy for application trustworthiness. As an alternative, a cloud platform could help achieve a robust technical solution by

- Making it easy for developers to write maintainable applications that protect user data in the cloud, thereby providing the same economies of scale for security and privacy as for computation and storage; and
- Enabling independent verification both of the platform’s operation and the runtime state of applications on it, so users can gain confidence that their data is being handled properly. Much as an operating system provides isolation between processes but allows substantial freedom inside a process, cloud platforms could offer transparently verifiable partitions for applications that compute on data units, while still allowing broad computational latitude within those partitions.
- DPaaS enforces fine-grained access control policies on data units through application confinement and information flow checking. It employs cryptographic protections at rest and offers robust logging and auditing to provide accountability. Crucially, DPaaS also directly addresses the issues of rapid development and maintenance. To truly support this vision, cloud platform providers would have to offer DPaaS in addition to their existing hosting environment, which could be especially beneficial for small companies or developers who don’t have much in-house security expertise, helping them build user confidence much more quickly than they otherwise might.

MODULE DESCRIPTION

1. CLOUD COMPUTING

Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. This cloud model promotes availability and is composed of five essential characteristics, three service models, and four deployment models.

Cloud computing is the provision of dynamically scalable and often virtualized resources as a services over the internet Users need not have knowledge of, expertise in, or control over the technology infrastructure in the "cloud" that supports them. Cloud computing represents a major change in how we store information and run applications. Instead of hosting apps and data on an individual desktop computer, everything is hosted in the "cloud"—an assemblage of computer and servers accessed via the Internet.

Cloud computing exhibits the following key characteristics:

1. Agility improves with users' ability to re-provision technological infrastructure resources.

2. Multi tenancy enables sharing of resources and costs across a large pool of users thus allowing for:

3. Utilization and efficiency improvements for systems that are often only 10–20% utilized.

4. Reliability is improved if multiple redundant sites are used, which makes well-designed cloud computing suitable for business continuity and disaster recovery.

5. Performance is monitored and consistent and loosely coupled architectures are constructed using web services as the system interface.

6. Security could improve due to centralization of data, increased security-focused resources, etc., but concerns can persist about loss of control over certain sensitive data, and the lack of security for stored kernels. Security is often as good as or better than other traditional systems, in part because providers are able to devote resources to solving security issues that many customers cannot afford. However, the complexity of security is greatly increased when data is distributed over a wider area or greater number of devices and in multi-tenant systems that are being shared by unrelated users. In addition, user access to security audit logs may be difficult or impossible. Private cloud installations are in part motivated by users' desire to retain control over the infrastructure and avoid losing control of information security.

7. Maintenance of cloud computing applications is easier, because they do not need to be installed on each user's computer and can be accessed from different places.

2. TRUSTED PLATFORM MODULE

Trusted Platform Module (TPM) is both the name of a published specification detailing a secure crypto processor that can store cryptographic keys that protect information, as well as the general name of implementations of that specification, often called the "TPM chip" or "TPM Security Device". The TPM specification is the work of the Trusted Computing Group.

Disk encryption is a technology which protects information by converting it into unreadable code that cannot be deciphered easily by unauthorized people. ***Disk encryption*** uses disk encryption software or hardware to encrypt every bit of data that goes on a disk or disk volume. Disk encryption prevents unauthorized access to data storage. The term "full disk encryption"[5] (or ***whole disk encryption***) is often used to signify that everything on a disk is encrypted, including the programs that can encrypt bootable operating system partitions. But they must still leave the master boot record (MBR)[6], and thus part of the disk, unencrypted. There are, however, hardware-based full disk encryption systems that can truly encrypt the entire boot disk, including the MBR**.**

3. THIRD PARTY AUDITOR

In this module, Auditor views the all user data and verifying data and also changed data. Auditor directly views all user data without key. Admin provided the permission to Auditor. After auditing data, store to the cloud.

4. USER MODULE

User store large amount of data to clouds and access data using secure key. Secure key provided admin after encrypting data. Encrypt the data using TPM. User store data after auditor, view and verifying data and also changed data. User again views data at that time admin provided the message to user only changes data.

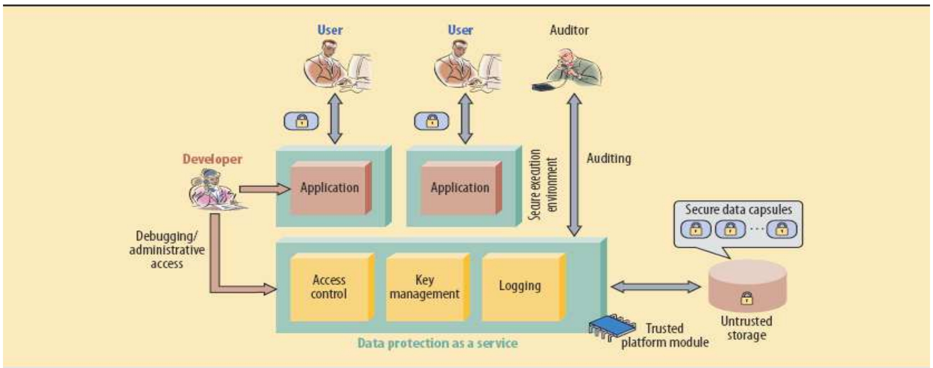


Figure 1. Sample architecture for data protection as a service illustrates how it's possible to integrate various technologies, such as application confinement, encryption, logging, code attestation, and information flow checking to realize DPaaS.

WAY FORWARD

In an OS, processes and files are the primary units of access control, and the OS provides suitable isolation for them. Applications can do what they like within these boundaries. In a cloud setting, the unit of access control is typically a sharable piece of user data—for example, a document in a collaborative editor. Ideally, the system offers some analogous confinement of that data, restricting its visibility only to authorized users and applications while allowing broad latitude for what operations are done on it. This can make writing secure systems easier for programmers because confinement makes it more difficult for buggy code to leak data or for compromised code to grant unauthorized access to data. A malicious program might find different ways to exfiltrate data, such as employing a side channel or covert channel, but the priority here is to support benign developers, while making all applications and their actions on users' sensitive data more easily auditable to catch improper usage.

One of the main concerns people and organizations have about putting data in the cloud is that they don't know what happens to it. Having a clear audit trail of when data is accessed—and by whom or what—bolsters confidence that data is being handled appropriately. Confinement can be effective for most normal user accesses, but administrative access that's outside the normal flow of user access and involves human administrators (for example, for debugging and analysis) can especially benefit from auditing.

VERIFIABLE PLATFORM SUPPORT

Bugs need to be fixed. Data needs to be updated and migrated as schemas change. Offline computation is valuable for data aggregation across users or for pre computation of expensive functions. To reduce the risk of unaudited backdoor access, all these functions should be subject to the same authorization flows and platformlevel checks as normal requests, albeit with a separate, appropriate policy. Platform providers should build support for confinement
and auditing into the platform in a verifiable way. This approval has many advantages:

- Application developers don't have to reinvent the wheel;
- Application code is independent of ACL enforcement;
- Third-party auditing and standards compliance are easier; and
- The verifiable platform extends to virtualized environments built atop it.

Finally, the cost of examining the platform is amortized across all its users, which means significant economies of scale for a large-scale platform provider.

DESIGN SPACE AND A SAMPLE ARCHITECTURE

Figure 1 illustrates an example architecture for exploring the DPaaS design space.5 Here, each server contains a *trusted platform module* (TPM) to provide secure and verifiable boot and dynamic root of trust. This example architecture demonstrates at a high level how it's potentially possible to combine various technologies such as application confinement, encryption, logging, code attestation, and information flow checking to realize DPaaS.

A *secure data capsule* (SDC) is an encrypted data unit packaged with its security policy. For example, an SDC might encompass a sharable document or a photo album along with its ACL. The platform can use confinement and information-flow controls to enforce capsules' ACLs. To avoid unauthorized leakage of user data in the presence of potentially buggy or compromised applications, DPaaS confines the execution of applications to mutually isolated *secure execution environments* (SEEs). Inter-SEE isolation has different levels, but stronger isolation generally exacts a greater performance cost due to context switching and data marshaling. At one end, a SEE could be a virtual machine with an output channel back to the requesting user. For performance reasons, it's possible to have a pool of VMs or containers in which the data state is reset before being loaded with a new data unit—
similar to how a thread pool works in a traditional server.

A more lightweight approach would be to use OS process isolation; an even lighter-weight approach would be to use language-based features such as information-flow controls or capabilities.7 We can use mechanisms such as Caja for JavaScript to confine user data on the client side as well, although we don't include that option as part of the platform. In some cases, applications need to call outside services or APIs provided by third-party websites—for example, the Google Maps API. An application might need to export users' data to outside services in this process. Users can explicitly define privacy policies to allow or disallow exporting SDCs to such third-party services, and DPaaS can enforce these policies. Additionally, DPaaS can log all instances where data is exported, and an auditor can later inspect these logs and detect any misuse a posteriori. Because our target applications have a basic requirement of sharable data units, DPaaS supports ACLs on SDCs. The key to enforcing those ACLs is to control the I/O channels available to the SEEs. To confine data, the platform decrypts the SDC's data only in a SEE in compliance with the SDC's security policy.

A SEE can funnel the output either directly to the user or to another SEE that provides a service; in either case, the platform mediates the channel. A buggy SEE only exposes a single SDC, an improvement over systems in which malicious input triggers a bug that allows access to all data. The platform also mediates ACL modifications, otherwise known as sharing or unsharing. A simple policy that the platform can enforce without having to know too much about the application is transitive: only currently authorized users can modify the ACL. For example, the creator is the first owner of a data unit, and at any time, any user with the owner status can add or revoke other authorized users. The support of anonymous sharing, in which possession of, say, a secret URL grants access to data, is also straightforward.

The platform itself doesn't need to understand granular, application-specific permissions; a simple, binary access-versus-no-access distinction goes a long way. The application can, of course, enforce any additional restrictions it requires on top of those the platform provides. There are no particular requirements for the data unit's underlying storage service.

The DPaaS approach places two additional requirements on the platform:

- It must be able to perform user authentication, or at least have a trusted way to know who's logged in and accessing the service; and
- It must rely on encryption and authenticated data store techniques to remove the need to trust the storage service.

DPaaS can accomplish user authentication either with a proprietary approach or using open standards such as OpenID and OAuth. Because the platform mediates all interactions, symmetric encryption suffices. With AES hardware units in commodity CPUs exceeding throughput of 1 Gbyte/second/core, performance is unlikely to be a bottleneck for all but the most I/O-intensive applications. Once the system loads the data into the SEE, it doesn't need to be encrypted or decrypted again until storage. In this model, the application can offload much of the basic work for identity and ACL enforcement to the platform and get certain user-level guarantees for free. This alone makes it much easier for developers to reason about system security because, by default (without any authorized user present), the data is simply unavailable.

The ACL governs ordinary user access, but administrative access requires its own separate policy, which in turn can be audited to hold developers and administrators accountable. Because each specific invocation of the administrative policy might entail human access to data, it should be logged and made available for auditing. The same kind of mechanism could handle batch access, perhaps with different logging granularity. To prevent misuse, the platform can restrict batch processes to only an approved set of programs, for example, requiring the programs to have controlled or quantifiable information release, such as differential privacy8 or quantitative information flow.9

PLATFORM VERIFIABILITY

The DPaaS approach provides logging and auditing at the platform level, sharing the benefits with all applications running on top. Offline, the auditor can verify that the platform implements each data protection feature as promised. At runtime, the platform provider can use *trusted computing* (TC) technologies to attest to the particular software that's running. TC uses the tamperproof TPM as well as the virtualization and isolation features of modern processors, such as Intel VT or AMDV. TC also allows for a dynamic root of trust—while the system runs, the CPU can enter a clean state, and the TPM can verify, load, and execute a *trusted computing base* (TCB), which is responsible for security-critical functionalities such as isolation enforcement, key management, access control, and logging.

Moreover, a third-party auditor can verify the code of the TCB that has been loaded onto the cloud platform. In this way, users and developers can gain confidence that the applications are indeed running on the correct TCB, and consequently trust the security guarantees and the audit logs the TCB provides. One challenge in code attestation is how to establish a set of acceptable binaries in the presence of rapid software updates such as bug fixes and new features. One potential way is to log the history of software updates and perform verification a posteriori. For the application itself, getting from verifiable to verified isn't easy; in a system with a lot of users, doing allpairs verification is prohibitively expensive.

ACHIEVING DATA PROTECTION GOALS

We assume in the analysis that the platform behaves correctly with respect to code loading, authorization, and key management, and that the TPM facilitates a runtime attestation to this effect.

DPaas uses a combination of encryption at rest, application confinement, information flow checking, and auditing to ensure the security and privacy of users' data. Application confinement isolates faults and compromises within each SEE, while information flow checking ensures that any information flowing among SEEs, data capsules, and users satisfies access-control policies. Controlling and auditing administrative accesses to data provides accountability.

DPaaS can guarantee the integrity of the data at rest via cryptographic authentication of the data in storage and by auditing the application code at runtime. Access controls, authorization, and auditing capability are common challenges for application developers. Incorporating these features within the platform is a significant improvement in terms of ease of use, and it doesn't constrain the types of computation that can be performed within a SEE. The platform logs common maintenance and batch processing tasks to provide accountability. These tasks too often require one-off work in the development process and can benefit from standardization.

As private data moves online, the need to secure it properly becomes increasingly urgent. The good news is that the same forces concentrating data in enormous datacenters will also aid in using collective security expertise more effectively. Adding protections to a single cloud platform can immediately benefit hundreds of thousands of applications and, by extension, hundreds of millions of users. While we have focused here on a particular, albeit popular and privacy-sensitive, class of applications, many other applications also need solutions, and many practical questions still remain open:

- Can we standardize technology across platforms to facilitate switching among providers?
- How can we make migration to the DPaaS cloud as easy as possible for existing applications?
- How can we minimize the cost of application audits?
- What kinds of audits are most important for building user confidence?
- Can technologies such as TC and code attestation be made scalable in the presence of constantly evolving software?
- How can we generalize the ideas presented here to other classes of applications?

In posing these questions, we hope to provoke thought and inspire future research and development in this important direction.

CONCLUSION

As private data moves online, the need to secure it properly becomes increasingly urgent. The good news is that the same forces concentrating data in enormous data enters will also aid in using collective security expertise more effectively. Adding protections to a single cloud platform can immediately benefit hundreds of thousands of applications and, by extension, hundreds of millions of users. While we have focused here on a particular, albeit popular and privacy-sensitive, class of applications, many other applications also needs solutions.

FUTURE ENHANCEMENT

In our system we are uploading the data files and protecting the files by encrypting the data. In future not only uploading the data file but also we can enhance the number of users by providing download option also.

REFERENCES

1. C. Dwork, "The Differential Privacy Frontier Extended Abstract," Proc. 6th Theory of Cryptography Conf. (TCC 09), LNCS 5444, Springer, 2009, pp. 496-502.
2. C. Gentry, "Fully Homomorphic Encryption Using Ideal Lattices," Proc. 41st Ann. ACM Symp. Theory Computing (STOC 09), ACM, 2009, pp. 169-178.
3. E. Naone, "The Slow-Motion Internet," Technology Rev., Mar./Apr. 2011; www.technologyreview.com/files/54902/ GoogleSpeed_charts.pdf.
4. A. Greenberg, "IBM's Blindfolded Calculator," Forbes, 13 July 2009; www.forbes.com/forbes/2009/0713/ breakthroughs-privacy-super-secret-encryption.html.
5. P. Maniatis et al., "Do You Know Where Your Data Are? Secure Data Capsules for Deployable Data Protection," Proc. 13th Usenix Conf. Hot Topics in Operating Systems (HotOS 11), Usenix, 2011; www.usenix.org/events/hotos11/ tech/final_files/ManiatisAkhawe.pdf.
6. S. McCamant and M.D. Ernst, "Quantitative Information Flow as Network Flow Capacity," Proc. 2008 ACM SIGPLAN Conf. Programming Language Design and Implementation (PLDI 08), ACM, 2008, pp. 193-205.
7. M.S. Miller, "Robust Composition: Towards a Unified Approach to Access Control and Concurrency Control," PhD dissertation, Dept. of Philosophy, Johns Hopkins Univ., 2006.
8. A. Sabelfeld and A.C. Myers, "Language-Based Information- Flow Security," IEEE J. Selected Areas Comm., Jan. 2003, pp. 5-19.
9. L. Whitney, "Microsoft Urges Laws to Boost Trust in the Cloud," CNET News, 20 Jan. 2010; http://news.cnet. com/8301-1009_3-10437844-83.html.

REQUEST FOR FEEDBACK

Dear Readers

At the very outset, International Journal of Research in Computer Application & Management (IJRCM) acknowledges & appreciates your efforts in showing interest in our present issue under your kind perusal.

I would like to request you tosupply your critical comments and suggestions about the material published in this issue as well as on the journal as a whole, on our E-mail**infoijrcm@gmail.com** for further improvements in the interest of research.

If youhave any queries please feel free to contact us on our E-mail **infoijrcm@gmail.com**.

I am sure that your feedback and deliberations would make future issues better – a result of our joint effort.

Looking forward an appropriate consideration.

With sincere regards

Thanking you profoundly

Academically yours

Sd/-
Co-ordinator

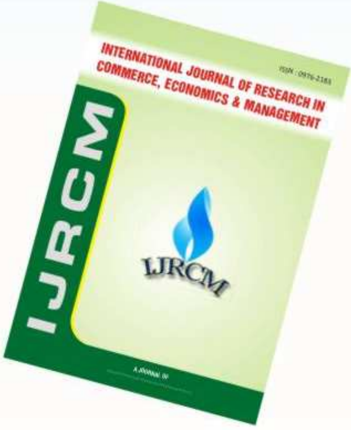
DISCLAIMER

The information and opinions presented in the Journal reflect the views of the authors and not of the Journal or its Editorial Board or the Publishers/Editors. Publication does not constitute endorsement by the journal. Neither the Journal nor its publishers/Editors/Editorial Board nor anyone else involved in creating, producing or delivering the journal or the materials contained therein, assumes any liability or responsibility for the accuracy, completeness, or usefulness of any information provided in the journal, nor shall they be liable for any direct, indirect, incidental, special, consequential or punitive damages arising out of the use of information/material contained in the journal. The journal, nor its publishers/Editors/ Editorial Board, nor any other party involved in the preparation of material contained in the journal represents or warrants that the information contained herein is in every respect accurate or complete, and they are not responsible for any errors or omissions or for the results obtained from the use of such material. Readers are encouraged to confirm the information contained herein with other sources. The responsibility of the contents and the opinions expressed in this journal is exclusively of the author (s) concerned.

ABOUT THE JOURNAL

In this age of Commerce, Economics, Computer, I.T. & Management and cut throat competition, a group of intellectuals felt the need to have some platform, where young and budding managers and academicians could express their views and discuss the problems among their peers. This journal was conceived with this noble intention in view. This journal has been introduced to give an opportunity for expressing refined and innovative ideas in this field. It is our humble endeavour to provide a springboard to the upcoming specialists and give a chance to know about the latest in the sphere of research and knowledge. We have taken a small step and we hope that with the active co-operation of like-minded scholars, we shall be able to serve the society with our humble efforts.

Our Other Journals



ISSN : 0976-2183

INTERNATIONAL JOURNAL OF RESEARCH IN COMMERCE, ECONOMICS & MANAGEMENT

IJRCM

A JOURNAL OF

ISSN : 0976-2183

INTERNATIONAL JOURNAL OF RESEARCH IN COMMERCE, IT & MANAGEMENT

IJRCM

A JOURNAL OF

ISSN : 0976-2183

INTERNATIONAL JOURNAL OF RESEARCH IN COMMERCE & MANAGEMENT

IJRCM

A JOURNAL OF