Offloading encryption/decryption of data/image to the cloud

Ashwin Shashidharan and Jitesh Shah
{ashashi3,jhshah}@ncsu.edu

September 28, 2011

Abstract

Cloud computing has emerged into a very popular [2] means of scaling up quickly without the associated infrastructure costs. The promise is complete freedom from maintaining own server infrastructures and moving to third-party infrastructure providers. For the cloud to be feasible, it is important for the clients to be able to verify that the image they are booting into isn't tampered with and that their data is safe in the cloud. If the cloud provider is not to be trusted with sensitive data, this means that the virtual image needs to be encrypted and the key stored with the client rather than the cloud provider. Also, the data in the cloud needs to be encrypted. We intend to address this problem using SSL to authenticate the server and then establishing a secure channel to exchange keys for decrypting the virtual image and a different key to secure any sensitive data. The final result is offloading encryption/decryption to the cloud rather than doing it on the client-side. As far as the implementation goes, our initial plan is to use Xen [8] to come up with a proof of concept.

Related work

VM images are more often than not stored and booted by the cloud provider. Thereby, arises one of the first security requirements in a cloud, customer ability to verify and trust their own image. Yet another major problem is that of security of sensitive data. It is issues like these and others that have mostly been grounds for distrust and slow adoption of cloud computing. Amazon, one of the biggest IaaS providers, themselves recommend customers to use client side encryption to secure their data [4] Also none of these providers have worked towards providing a strong guarantee of trusted boot. These and additional requirements of a trusted cloud platform have been well-summarized in [33]. The solution to the problem is two-fold: Trusted boot and Data encryption.

Trusted boot has been discussed in literature since long. One such article outlines a trusted boot method for standalone IBM machines using PKI that successively verifies each component before transferring control over to it [5]. There has also work been done exploring the use of a co-processor to verify each system block as it loads[39]. While the former approach requires significant firmware modifications, the latter involves re-designing hardware. These concepts though provide security; they are difficult to put in to practice in a virtualized environment.

Alternatives like Trusted Platform Module (TPM) [3] based solutions have explored widely in papers[30, 17, 20, 18]. There also exist implementations that use TPM to create a TVMM (Trusted VMM), used as the root of trust [30]. TPMs are separately added hardware modules and offer higher security. TVMM's have also been virtualised for use by guest operating systems and discussed earlier [9]. The Trusted Computing Group (TCG) has also come up with a secure alternative to BIOS called UEFI (Unified Extensible Firmware interface)[1]. The primary goal of UEFI is secure boot i.e., to allow booting only signed boot-loader/OS. Even with these approaches no one can really stop a cracker to boot a virtual image if he manages to break into the cloud provider's servers. TPM only assures us of an image which has not been tampered. Our work outlines a need for an encrypted kernel using a key provided by the customer on-demand. Furthermore, any process that enables this feature can include TPMs for it's verification.

Run-time Integrity verification is another aspect. All approaches outlined above only perform static verification. There has been related work [32, 29, 36, 34] that discuss the run-time requirements conforming to Clark-Wilson's integrity model [10]. Also a method [9] to use TVMMs to build run-time integrity mechanism has been discussed before. However, we leave out run-time integrity since it is beyond the scope of our project.

Policy based decisions to continue or terminate a connection is particularly useful to clients to establish integrity of the communication endpoint in the cloud. TLS[5] is the most widely used technology for securing a transmission channel to its endpoints. In conjunction with Trusted Computing technology, the TLS protocol in recent years has seen enhancements allowing peers on endpoints to decide on changes in connection [19].

Also, besides securing the channel, the user must be able to trust an application in the cloud. Owing to the trusted computing initiative there has been increasing research in attestation systems [11] and their use in verification of application integrity. With a combination of transport layer security, these schemes can be effectively used for appraisal of remote applications in the cloud. The applications may also continue to enforce standard OS implemented policies like SELinux for authorized use [24]. With closely coupled security architecture features and strict applications policies, systems may be configured to prevent unauthorized access [25]. Besides, Distributed Mandatory Access Control policies based on reference monitors [26] have adequately addressed policy issues pertaining to distributed machines applicable in a cloud.

The advent of cloud storage also has seen increased research in non-standard cryptographic primitives and architectures. The driving force behind such research can be attributed to users craving for storage infrastructure without having to trust the provider [22]. Cryptographic techniques typically introduce additional computation which could render securing data in its lifetime as impractical. To search and retrieve encrypted data, techniques involving encrypted indexes have been devised [37]. For other instances where data integrity in the cloud is more significant than confidentiality, there has been work to verify stored data at the data storage itself [14, 7]. There has also been gaining popularity for work on disk encryption [16] like LUKS, Bit-Locker [15], TrueCrypt[11] but, primarily for operating systems which can encrypt an entire partition or storage device. Another approach to building the users trust extends conventional key management systems for data encryption on cloud storage [35].

It must be noted that a rigorous security policy for physically protecting computing elements in the cloud is likewise of no less importance. Hardware based attacks on disk encryption systems for analyzing memory content have also been discussed before and is not an entirely new area[21].

In conclusion, appropriate confidence building measures are necessary to gain a cloud user's trust without which cloud providers shall remain unsuccessful in alleviating user's fears about security in the cloud [38].

References

[1] Trusted platforms uefi, pi and tcg-based firmware. http://download.intel.com/technology/efi/SF09_EFIS001_UEFI_PI_TCG_White_Paper.pdf, September 2009.

[2] Cloud computing adoption survey results, 2010. Last accessed: 28 September, 2011. http://www.thehostingnews.com/cloud-computing-adoption-survey-results-released-12517.html.

[3] Trusted computing group, 2010. Last accessed: 28 Sept, 2011. http://www.trustedcomputinggroup.org/.

[4] Amazon web services: Overview of security processes, May 2011.

[5] W. A. Arbaugh, D. J. Farber, and J. M. Smith. A secure and reliable bootstrap architecture. In *Proceedings of the 1997 IEEE Symposium on Security and Privacy*, SP '97, pages 65–, Washington, DC, USA, 1997. IEEE Computer Society.

[6] Frederik Armknecht, Yacine Gasmi, Ahmad-Reza Sadeghi, Patrick Stewin, Martin Unger, Gianluca Ramunno, and Davide Vernizzi. An efficient implementation of trusted channels based on openssl. In *Proceedings of the 3rd ACM workshop on Scalable trusted computing*, STC '08, pages 41–50, New York, NY, USA, 2008. ACM.

[7] Giuseppe Ateniese, Randal Burns, Reza Curtmola, Joseph Herring, Lea Kissner, Zachary Peterson, and Dawn Song. Provable data possession at untrusted stores. In *Proceedings of the 14th ACM conference on Computer and communications security*, CCS '07, pages 598–609, New York, NY, USA, 2007. ACM.

[8] Paul Barham, Boris Dragovic, Keir Fraser, Steven Hand, Tim Harris, Alex Ho, Rolf Neugebauer, Ian Pratt, and Andrew Warfield. Xen and the art of virtualization. In *Proceedings of the nineteenth ACM symposium on Operating systems principles*, SOSP '03, pages 164–177, New York, NY, USA, 2003. ACM.

[9] Stefan Berger, Ramón Cáceres, Kenneth A. Goldman, Ronald Perez, Reiner Sailer, and Leendert van Doorn. vtpm: virtualizing the trusted platform module. In *Proceedings of the 15th conference on USENIX Security Symposium - Volume 15*, Berkeley, CA, USA, 2006. USENIX Association.

[10] David D. Clark and David R. Wilson. A comparison of commercial and military computer security policies. *Security and Privacy, IEEE Symposium on*, 0:184, 1987.

[11] George Coker, Joshua Guttman, Peter Loscocco, Amy Herzog, Jonathan Millen, Brian OHanlon, John Ramsdell, Ariel Segall, Justin Sheehy, and Brian Sniffen. Principles of remote attestation. *International Journal of Information Security*, 10:63–81, 2011. 10.1007/s10207-011-0124-7.

[12] Alexei Czeskis, David J. St. Hilaire, Karl Koscher, Steven D. Gribble, Tadayoshi Kohno, and Bruce Schneier. Defeating encrypted and deniable file systems: Truecrypt v5.1a and the case of the tattling os and applications. In *Proceedings of the 3rd conference on Hot topics in security*, pages 7:1–7:7, Berkeley, CA, USA, 2008. USENIX Association.

[13] Joan Daemen and Vincent Rijmen. The block cipher rijndael. In *Proceedings of the The International Conference on Smart Card Research and Applications*, pages 277–284, London, UK, 2000. Springer-Verlag.

[14] Chris Erway, Alptekin Küpçü, Charalampos Papamanthou, and Roberto Tamassia. Dynamic provable data possession. In *Proceedings of the 16th ACM conference on Computer and communications security*, CCS '09, pages 213–222, New York, NY, USA, 2009. ACM.

[15] Niels Ferguson. Aes-cbc + elephant diffuser: A disk encryption algorithm for windows vista. http://www.microsoft.com/downloads/details.aspx?familyid, 2006.

[16] Clemens Fruhwirth. New methods in hard disk encryption. Technical report, 2005.

[17] Tal Garfinkel, Ben Pfaff, Jim Chow, Mendel Rosenblum, and Dan Boneh. Terra: a virtual machine-based platform for trusted computing. pages 193–206. ACM Press, 2003.

[18] Tal Garfinkel, Mendel Rosenblum, and Dan Boneh. Flexible os support and applications for trusted computing. In *IN 9TH HOT TOPICS IN OPERATING SYSTEMS (HOTOS-IX)*, pages 145–150, 2003.

[19] Yacine Gasmi, Ahmad-Reza Sadeghi, Patrick Stewin, Martin Unger, and N. Asokan. Beyond secure channels. In *Proceedings of the 2007 ACM workshop on Scalable trusted computing*, STC '07, pages 30–40, New York, NY, USA, 2007. ACM.

[20] Vivek Haldar, Deepak Chandra, and Michael Franz. Semantic remote attestation - a virtual machine directed approach to trusted computing. In *USENIX Virtual Machine Research and Technology Symposium*, pages 29–41, 2004.

[21] J. Alex Halderman, Seth D. Schoen, Nadia Heninger, William Clarkson, William Paul, Joseph A. Calandrino, Ariel J. Feldman, Jacob Appelbaum, and Edward W. Felten. Lest we remember: cold-boot attacks on encryption keys. *Commun. ACM*, 52:91–98, May 2009.

[22] Seny Kamara and Kristin Lauter. Cryptographic cloud storage. In *Proceedings of the 14th international conference on Financial cryptograpy and data security*, FC'10, pages 136–149, Berlin, Heidelberg, 2010. Springer-Verlag.

[23] Chongkyung Kil, Emre C. Sezer, Ahmed M. Azab, Peng Ning, and Xiaolan Zhang. In *Proceedings of the 39th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN 2009)*.

[24] Peter Loscocco and Stephen Smalley. Integrating flexible support for security policies into the linux operating system. In *Proceedings of the FREENIX Track: 2001 USENIX Annual Technical Conference*, pages 29–42, Berkeley, CA, USA, 2001. USENIX Association.

[25] Peter A. Loscocco and Stephen D. Smalley. Meeting critical security objectives with Security-Enhanced linux. In *Proceedings of the 2001 Ottawa Linux Symposium*, 2001.

[26] Jonathan M. McCune, Trent Jaeger, Stefan Berger, Ramon Caceres, and Reiner Sailer. Shamon: A system for distributed mandatory access control. In *Computer Security Applications Conference, 2006. ACSAC '06. 22nd Annual*, pages 23 –32, dec. 2006.

[27] Jonathan M. McCune, Adrian Perrig, and Michael K. Reiter. Safe passage for passwords and other sensitive data. In *NDSS*. The Internet Society, 2009.

[28] Ahmad-Reza Sadeghi, Thomas Schneider, and Marcel Winandy. Token-based cloud computing: secure outsourcing of data and arbitrary computations with lower latency. In *Proceedings of the 3rd international conference on Trust and trustworthy computing*, TRUST'10, pages 417–429, Berlin, Heidelberg, 2010. Springer-Verlag.

[29] Reiner Sailer, Xiaolan Zhang, Trent Jaeger, and Leendert van Doorn. Design and implementation of a tcg-based integrity measurement architecture. In *Proceedings of the 13th conference on USENIX Security Symposium - Volume 13*, SSYM'04, pages 16–16, Berkeley, CA, USA, 2004. USENIX Association.

[30] Nuno Santos, Krishna P. Gummadi, and Rodrigo Rodrigues. Towards trusted cloud computing. In *HOTCLOUD*. USENIX, 2009.

[31] J. Schiffman, T. Moyer, T. Jaeger, and P. McDaniel. Network-based root of trust for installation. *Security Privacy, IEEE*, 9(1):40 –48, jan.-feb. 2011.

[32] Joshua Schiffman, Thomas Moyer, Christopher Shal, Trent Jaeger, and Patrick McDaniel. Justifying integrity using a virtual machine verifier. *Computer Security Applications Conference, Annual*, 0:83–92, 2009.

[33] Joshua Schiffman, Thomas Moyer, Hayawardh Vijayakumar, Trent Jaeger, and Patrick McDaniel. Seeding clouds with trust anchors. In *Proceedings of the 2010 ACM workshop on Cloud computing security workshop*, CCSW '10, pages 43–46, New York, NY, USA, 2010. ACM.

[34] Umesh Shankar. Toward automated information-flow integrity verification for security-critical applications. In *In Proceedings of the 2006 ISOC Networked and Distributed Systems Security Symposium (NDSS06)*, 2006.

[35] S H Shin and Kazukuni Kobara. Towards secure cloud storage. *Demo for CloudCom2010*, 2010.

[36] Santosh Shrivastava. Satem: Trusted service code execution across transactions. *Reliable Distributed Systems, IEEE Symposium on*, 0:337–338, 2006.

[37] Dawn Xiaoding Song, D. Wagner, and A. Perrig. Practical techniques for searches on encrypted data. In *Security and Privacy, 2000. S P 2000. Proceedings. 2000 IEEE Symposium on*, pages 44 –55, 2000.

[38] A. Tolnai and S.H. von Solms. Securing the cloud's core virtual infrastructure. In *Broadband, Wireless Computing, Communication and Applications (BWCCA), 2010 International Conference on*, pages 447 –452, nov. 2010.

[39] J. D. Tygar and Bennet Yee. Dyad: A system for using physically secure coprocessors. Technical report, Proceedings of the Joint Harvard-MIT Workshop on Technological Strategies for the Protection of Intellectual Property in the Network Multimedia Environment, 1991.

[40] David Wagner and Bruce Schneier. Analysis of the ssl 3.0 protocol. In *Proceedings of the 2nd conference on Proceedings of the Second USENIX Workshop on Electronic Commerce - Volume 2*, pages 4–4, Berkeley, CA, USA, 1996. USENIX Association.

[41] Shucheng Yu, Cong Wang, Kui Ren, and Wenjing Lou. Achieving secure, scalable, and fine-grained data access control in cloud computing. In *Proceedings of the 29th conference on Information communications*, INFOCOM'10, pages 534–542, Piscataway, NJ, USA, 2010. IEEE Press.

[42] W.Z.A. Zakaria, M.A. Parman, Z. Hamdan, M.S. Rohmad, and M.A.M. Isa. Secure virtual application distribution. In *Computer Technology and Development, 2009. ICCTD '09. International Conference on*, volume 1, pages 81 –85, nov. 2009.