International Journal of Computers & Technology
Volume 4 No. 2, March-April, 2013, ISSN 2277-3061

A REVIEW OF NEW ABOUT CLOUD COMPUTING SECURITY

Bohar Singh [(1)], Mandeep Kaur[(2),] Chetan Batra[(3)]
[(1)]Lecturer in SBSSTC (POLYWING), Ferozepur
bohar2@gmail.com
[(2)]Lecturer in SBSSTC (POLYWING), Ferozepur
menuk8@gmail.com
[(3)]Lecturer in SBSSTC (POLYWING), Ferozepur
chetan.batra55@gmail.com

ABSTRACT

Job scheduling system problem is a core and challenging issue in Cloud Computing. How to use Cloud computing resources efficiently and gain the maximum profits with job scheduling. Job scheduling system is one of the Cloud computing service providers' ultimate goals. In this paper, firstly, by analysis the differentiated QoS requirements of Cloud computing resources users' jobs, we build the corresponding non-pre-emptive priority M/G/1 queuing model for the jobs.Then, considering Cloud computing service providers' destination which is to gain the maximum profits by offering Cloud computing resources, we built the system cost function for this queuing model. After that, based on the queuing model and system cost function, considering the goals of both the Cloud Computing service users and providers, we gave the corresponding strategy and algorithm to get the approximate optimistic value of service for each job in the corresponding no-pre-emptive priority M/G/1 queuing model. Finally, we also provide corresponding simulations and numeral results.Analysis and number results show that our approach for job scheduling system can not only guarantee the QoS requirements of the users' jobs, but also can make the maximum profits for the Cloud computing service providers.While the economic case for cloud computing is compelling, the security challenges it poses are equally striking. In this work we strive to frame the full space of cloud-computing security issues, attempting to separate justified concerns from possible over-reactions. We examine contemporary and historical perspectives from industry, academia, government, and "black hats". We argue that few cloud computing security issues are fundamentally new or fundamentally intractable; often what appears "new" is so only relative to "traditional"computing of the past several years. Looking back further to the time-sharing era, many of these problems already received attention. On the other hand, we argue that two facets are to some degree new and fundamental to cloud computing: the complexities of multi-party trust considerations, and the ensuing need for mutual audit ability.

KEYWORDS

Cloud Computing; Job scheduling system;queuing system; QoS; Little's Law

INTRODUCTION

Job scheduling (JS) system is one of the core and challenging issues in a Cloud Computing system.Traditional job scheduling systems in Cloud (or Grid)computing only consider how to meet the QoS requirements for the resources users, they seldom consider how to make the maximum profits for the resource providers. Actually, a job scheduling system plays a very important role in how to meet Cloud computing users' job QoS requirements and use the Cloud resources efficiently in an economic way. Usually,from the Cloud computing resources users' sides (we use CCU stands for Cloud computing user), users always think which Cloud computing resource can meet their job QoS requirements for computing (such as the due time of job finishing, the computing capacity etc.), how much money they must pay for the Cloud Computing resources. While,from the Cloud Computing service providers (we use CCSP stand for Cloud Computing Service Provider) side, the CCSP always think how they can gain the maximum profits by offering Cloud Computing resources, apart from meeting the CCU's job QoS requirements. To make these two ends.The economic case for cloud computing has gained wide spread acceptance. Cloud computing providers can build large data entersat low cost due to their expertise in organizing and provisioning computational resources. The economies of scale increase revenue for cloud providers and lower costs for cloud users. The resulting on-demand model of computing allows providers to achieve better resource utilization through statistical multiplexing, and enables users to avoid the costs of resource over-provisioning through dynamic scaling [12, 2].At the same time, security has emerged as arguably the most significant barrier to faster and more widespread adoption of cloud computing. This view originates from perspectives as diverse as academia researchers [12], industry decision makers [35], and government organizations [29, 3]. For many business-critical computations,today's cloud computing appears inadvisable due to issues such as service availability, data confidentiality, reputation fate sharing,and others.To add to the confusion, some have citizen the term "cloud computing"as too broad [21]. Indeed, cloud computing does include established business models such as Software as a Service, and the underlying concept of on-demand computing utilities goes back as far as early time-sharing systems [17]. At the same time, the lack of consistent terminology for cloud computing has hampered discussions about cloud computing security. Thus, security criticisms of cloud computing have included a murky mix of ongoing and new issues. This context frames the genesis of our paper. We recognize that security poses major issues for the widespread adoption of cloud computing. However, secure or not, cloud computing appears here to stay. Thus, our ambition is to get past terminology issues (Section2) and attempt to sort out what are actually new security issues for cloud computing, versus broader and more general security challenges that inevitably arise in the Internet age. Our goal is to advance discussions of cloud computing security beyond confusion,and to some degree fear of the unknown, by providing acomprehensive high-level view of the problem space.We ground the development of our viewpoint in a survey of contemporaryliterature on cloud computing security, coupled with are view of historical work on early time-sharing systems and virtual

Machine monitors. Contemporary discussions reveal security concerns that are indeed "new" relative to computing of the past decade (Section 3); however, looking back several decades, many contemporary challenges have quite similar historical counterparts(Section 4).We build the case that few of the security problems arising in cloud computing are in fact new, even though satisfactory solutions

International Journal of Computers & Technology
Volume 4 No. 2, March-April, 2013, ISSN 2277-3061

formany still will require significant development. The combined contemporaryand historical viewpoints allow us to identify a numberof research topics that deserve more attention (Section 5). On theother hand, we argue that two facets are to some degree new andfundamental to cloud computing: the complexities of multi-party trust considerations, and the ensuing need for mutual audit ability.

Fig-1: Clouding Computing

THE CLOUD ONTOLOGY

Cloud computing systems fall into one of five layers: applications, software environments, software infrastructure, software kernel, and hardware. Obviously, at the bottom of the cloud stack is the hardware layer which is the actual physical components of the system. Some cloud computing offerings have built their system on subleasing the hardware in this layer as a service, as we discuss in subsection IV-E. At the top of the stack is the cloud application layer, which is the interface of the cloud to the common computer users through web browsers and thin computing terminals. We closely examine the characteristics and limitations of each of the layers in the next five subsections.

Cloud Application Layer:

The cloud application layer is the most visible layer to the end-users of the cloud. Normally, the users access the services provided by this layer through web-portals, and are sometimes required to pay fees to use them. This model has recently proven to be attractive to many users, as it alleviates the burden of software maintenance and the ongoing operation and support costs. Furthermore, it exports the computational work from the users' terminal to data centres where the cloud applications are deployed. This in turn lessens the restrictions on the hardware requirements needed at the users' end, and allows them to obtain superb performance to some of their cpu-intensive and memory-intensive workloads without necessitating huge capital investments in their local machines. As for the providers of the cloud applications, this model even simplifies their work with respect to upgrading and testing the code, while protecting their intellectual property. Since a cloud application is deployed at the provider's computing infrastructure (rather than at the users' desktop machines), the developers of the application are able to roll smaller patches to the system and add new features without disturbing the users with requests to install major updates or service packs. Configuration and testing of the application in this model is arguably less complicated, since the deployment environment becomes restricted, i.e., the provider's data canter. Even with respect to the provider's margin of profit, this model supplies the software provider with a continuous flow of revenue, which might be even more profitable on the long run. This model conveys several favourable benefits for the users and providers of cloud applications, and is normally referred to as Software as a Service (SaaS). Sales force Customer Relationships Management (CRM) system [7] and Google Apps [8] are two examples of SaaS. As such, the body of research on SOA has numerous studies on compostable IT services which have direct application to providing and composing SaaS. Our proposed ontology illustrates that cloud applications can be developed on the cloud software environments or infrastructure components (as discussed in the next two subsections). In addition, cloud applications can be composed as a service from other cloud services offered by other cloud systems, using the concepts of SOA. For example, a payroll application might use another accounting SaaS to calculate the tax deductibles for each employee in its system without having to implement this service within the payroll software. In this respect, the cloud applications targeted for higher layers in the stack are simpler to develop and have a shorter time-to market. Furthermore, they become less error-prone since all their interactions with the cloud are through pretested APIs. Developed for a higher cloud-stack layer, the flexibility of the applications is however limited and this may restrict the developers' ability to optimize their applications' performance. Despite all the advantageous benefits of this model, several deployment issues hinder its wide adoption specifically, the security and availability of the cloud.

Cloud Software Environment Layer:

The second layer in our proposed cloud ontology is the cloud software environment layer (also dubbed the software platform layer). The users of this layer are cloud applications' developers, implementing their applications for and deploying them on the cloud. The providers of the cloud software environments supply the developers with a programming-language-level environment with a set of well-defined APIs to facilitate the interaction between the environments and the cloud applications,as well as to accelerate the deployment and support the scalability needed of those cloud applications. The service provided by cloud systems in this layer is commonly referred to as Platform as a Service (PaaS). One example of systems in this category is Google's App Engine [5],which provides a python runtime environment and APIs for applications to interact with Google's cloud runtime environment. Another example is Sales Force Apex language [9] that allows the developers of the cloud applications to design, along with their applications'logic, their page layout, workflow, and customer reports.Developers reap several benefits from developing theircloud application for a cloud programming environment,including automatic scaling and load balancing, as well as integration with other services (e.g. authentication services, email services, user interface) provided to them through the PaaS-provider. In such a way, much

International Journal of Computers & Technology
Volume 4 No. 2, March-April, 2013, ISSN 2277-3061

of the overhead of developing cloud applications is alleviatedand is handled at the environment level. Furthermore,developers have the ability to integrate other services to their applications on-demand. This in turn makes the cloud application development a less complicated task, accelerates the deployment time and minimizes the logic faults in the application. In this respect, a Hadoop [10] deployment on the cloud would be considered a cloud software environment, as it provides its applications'developers with a programming environment, i.e. map reduce framework for the cloud. Similarly, Yahoo'sPig [11], a high-level language to enable processing of very large files on the hadoop environment may be viewed as an open-source implementation of the cloud platform layer. As such, cloud software environments facilitate the process of the development of cloud applications.4C. Cloud Software Infrastructure Layer. The cloud software infrastructure layer provides fundamental resources to other higher-level layers, whichin turn can be used to construct new cloud software environments or cloud applications. Our proposed ontology reflects the fact that the two highest levels in the cloud stack can bypass the cloud infrastructure layer in building their system. Although this bypass can enhance the efficiency of the system, it comes at the cost of simplicity and development efforts.

DISTRACTED BY DEFINITIONS

The lack of a clear and widely accepted definition has posed a barrier to talking about cloud computing in general. Clearly "cloud computing" is an evolving term, defined more by usage than by written documents. That said, overly broad use has lead to criticism that cloud computing "include[s] everything that we already do" [21]. Similarly, splitting hairs on the precise definitions distracts us from the core technology issues. In this section, we briefly frame the definition we use for the remainder of our discussion.An "early" (less than one year old!) effort at systematically framing cloud computing, "Above the Clouds: A Berkeley View of Cloud Computing," defined cloud computing to include application software delivered as services over the Internet, and the hardware and systems software in the data enters that facilitate the services [12]. Key characteristics of cloud computing include the illusion of infinite hardware resources, the elimination of up-front commitment, and the ability to pay for resources as needed.This whitepaper spurred a flurry of follow-on cloud computing definitions and reports. For our purposes, the most notable of these is that published by the U.S. National Institute of Standards and Technology (NIST) [30]. NIST frames a broader definition, one that includes nearly all common terms used in cloud computing discussions and forms the basis for the NIST guide on cloud computing security [29]. It appears that other efforts may converge on a similar framing; most visibly, the European mirror effort to [29], a report from the European Network and Information Security Agency(ENISA), defines cloud computing in the same spirit as the NIST definition [3].According to the NIST definition, key characteristics of cloud computing include on-demand self service, broad network access, resource pooling, rapid elasticity, and metered service similar to autility. There are also three main service models—software as a service (SaaS), in which the cloud user controls only application configurations; platform as a service (PaaS), in which the cloud user also controls the hosting environments; and infrastructure as a service (IaaS), in which the cloud user controls everything except the data center infrastructure. Further, there are four main deployment models: public clouds, accessible to the general public or a large industry group; community clouds, serving several organizations;private clouds, limited to a single organization; and hybridclouds, a mix of the others.In keeping with this evolution, and because we believe the broadscope of the NIST definition enables us to encompass the full setof issues of interest, for the rest of this paper, we will talk about"cloud computing" in the spirit of the NIST definition.

CONTEMPORARY ASSESSMENT

In this section, we assess what appears new to cloud computing and what does not, so that we can identify the most challenging aspects of the cloud computing security threat model.

- **What is not new?**

With increased employment of cloud computing comes increasingly frequent cloud computing security incidents. Arguably many of the incidents described as "cloud security" in fact just reflect traditional web application and data-hosting problems. The underlying issues remain well-established challenges such as phishing [4],downtime [24], data loss [38], password weaknesses [31], and compromised hosts running botnets [20]. The Twitter phishing incident provides a typical example of a traditional web security issue nowmiscast as a cloud computing issue [4]. In contrast, we find the recent Amazon botnet incident note worthy because it reflects one of the first known compromises of a major cloud provider [20],highlighting that servers in cloud computing currently operate as(in)securely as servers in traditional enterprise data centres'. In academia, cloud computing security has begun seeing the development of dedicated forums such as the ACM Cloud Computing Security Workshop, as well as dedicated tracks at major security conferences such as the ACM Conference on Computer and Communications Security (CCS). To date, most papers published on cloud security reflect continuations of established lines of security research, such as web security [40, 13], data out sourcing and assurance [14, 18], and virtual machines [41, 34]. The field primarily manifests as a blend of existing topics, rather than a set of papers with an exclusive focus on cloud security, though there are exceptions, such as [32], which we discuss below.The "black hat" community has also discovered cloud computing exploits that reflect extensions of existing vulnerabilities, with adedicated cloud security track emerging at Black Hat USA 2009.For example, username brute forcers and DebianOpen SSL exploit tools run in the cloud as they do in botnets [28]. Social engineering attacks remain effective—one exploit tries to convince Amazon Elastic Compute Cloud (EC2) users to run malicious virtual machine images simply by giving the image an official-sounding name such as "fedora core" [28]. Virtual machine vulnerabilities also remain an issue [25], as does weak random number generation due to lack of sufficient entropy [37].

- **What is new?**

For black hats, cloud computing offers a potentially more trust worthy alternative to botnets. While the recent brute-force presentation[28] claimed that using the cloud is presently more expensive than using botnets, another Black Hats presentation asserted that the botnet market likely suffers from the "lemon market" problem,where the lack of trust and the inability to verify the quality of goods leads to a minimal volume of goods being exchanged [22].If this were the case, then attackers can find more reliable service in

International Journal of Computers & Technology
Volume 4 No. 2, March-April, 2013, ISSN 2277-3061

cloud computing at a premium price.1 That said, botnets in the cloud are easier to shut down than traditional botnets.Also, because cloud computing introduces a shared resource environment,unexpected side channels (passively observing information)and covert channels (actively sending data) can arise. One noteworthy paper [32] tackles precisely this problem. The exposed vulnerabilities include ways to place an attacker virtual machine (VM) on the same physical machine as a targeted VM, and then to construct a side channel between two VMs on the same physical machine, which enables the SSH keystroke timing attack outlinedin [36]. This work also provides an example of research targeted exclusively at cloud computing. Another new issue comes from reputation fate-sharing, which has mixed consequences. On the plus side, cloud users canpotentially benefit from a concentration of security expertise atmajor cloud providers, ensuring that the entire ecosystem employs security best practices. On the other hand, a single subverter can disrupt many users. For example, spammers subverted EC2 and caused Spamha us to blacklist a large fraction ofEC2's IP addresses, causing major service disruptions. There1Note that the prices can be quite low. For example, we estimatethat to reduce the brute force exploit in [36] to a single minute,rather than 1.3 PC-days, would require 200 extra-large EC2 instances,which at January 2010 pricing would total at about $2 per exploit. After, if someone wants to send email from EC2, they must fill out the form (http://aws.amazon.com/contact-us/ec2-email-limit-request/), provide a list of (static) EC2addresses to authorize for sending, and document their use-case.Upon approval, Amazon forwards the EC2 addresses to Spamhaus for whitelisting [8].A second note worthy fate-sharing incident occurred during an FBIraid on Texas data enters in April 2009, based on suspicions of the targeted data enters facilitating cyber crimes. The agents seized equipment, and many businesses co-located in the same data enterfaced business disruptions or even complete business closures. One affected customer applied for a temporary restraining order, and was denied because the equipment concerned may have been used for criminal activities without the customer's knowledge [6].

Novelties in the cloud threat model

Putting together these discussions, we argue that the cloud computing threat model includes several novel elements.First, data and software are not the only assets worth protecting.Activity patterns also need to be protected. Sharing of resources means that the activity of one cloud user might appear visible to other cloud users using the same resources, potentially leading to the construction of covert and side channels. Activity patterns may also themselves constitute confidential business information, if divulging them could lead to reverse-engineering of customer base, revenue size, and the like.Business reputation also merit protection. When using shared resources to do business-critical computations, it becomes harder to attribute malicious or unethical activity. Even if there are ways to clearly identify the culprits and attribute blame, bad publicity still creates uncertainty that can tarnish a long-established reputation.In addition, one must often accommodate a longer trust chain. For example, the application end-user could potentially use an application built by an SaaS provider, with the application running ona platform offered by a PaaS provider, which in turn runs on the infrastructure of an IaaS provider. While to our knowledge this extreme example cannot occur in practice today due to a lack of sufficient APIs, it illustrates that with any model of cloud computing,stakeholders' can find themselves with relationships considerably more complicated than simply a provider-user relationship.Some participants could be subverters, who maintain the appearance of a regular cloud user or cloud provider, but in fact perpetrate cybercrime or other cyber attacks. Examples include cloud users who run brute forcers, botnets, or spam campaigns from the cloud; or cloud providers who scan cloud users' data and sell confidential information to the highest bidder.Furthermore, competitive businesses can operate within the same cloud computing ecosystem: using the same cloud or ending upin a provider-user relationship. This can lead to strong conflicts of Interest, and creates additional motives to access the confidential information of a competitor. These complications point to the need for audit ability in cloud computing—already a requirement for health care, banking, and similar systems. What is new to cloud computing is mutual audit ability.Because the system includes stakeholders with potentially conflicting interests, cloud users and providers both need reassurance that the other in a fashion that is both benign and correct(from a billing standpoint).Mutual audit ability can also significantly assist with incident response and recovery, since both the cloud provider and the cloud user could be either the source or the target of an attack. Audit ability also enables the attribution of blame in search and seizure incidents, which can prove vital so that law enforcement agencies donot overreach in carrying out their duties.Finally, a subtle difficulty with understanding cloud computing threats arises from potentially inaccurate mental models of cloud computing as an always-available service. This viewpoint—which arises from the general paradigm of drawing upon a commodity service with much the flavour of a utility—can create a false sense of security, leading to inadequate security good practices, such as regular data backups across multiple cloud providers. As such, we could find that while cloud computing fails at the same rate as other types of systems, the impact of those failures manifest more severely.

FINAL THOUGHTS

Given the stakes, it strikes us as inevitable that security will become a significant cloud computing business differentiator. Furthermore,in addition to revisiting approaches for specific issues insecuring shared computing, history teaches us that developing security architectures early in the process can pay off greatly as systems evolve and accrue more disparate functionality. On the otherhand, the history of commercial Internet offerings repeatedly showsthat time-to-market and undercutting prices can greatly sway customer seven in the absence of sound security under pinnings. The situation may be somewhat different this time around, however,given that much of cloud computing targets customers who haveextensive business reasons (and scars from the past) leading themto treat security as an elevated priority.We close our discussion with what we find to be an interesting analogy.Companies such as National CSS began by offering affordable computation for businesses. Time-sharing eventually gave way to personal computers, which brought affordable computation to the general public. In a similar fashion, cloud computing currently offers affordable, large-scale computation for businesses. If the economic case prevails, then we may find that nothing—not even security concerns—will prevent cloud computing from becoming a consumer commodity. Just as the commodity PC and the Internet brought about the Information Revolution, and made information universally accessible, affordable, and useful, so too does cloudcomputing have the potential to bring about the Computation Revolution,in which large-scale computations become universally accessible,affordable, and useful. Let's hope we can add to this outcome"and be reasonably safe".

REFERENCES

[1] Amazon virtual private cloud.http://aws.amazon.com/vpc/.

International Journal of Computers & Technology
Volume 4 No. 2, March-April, 2013, ISSN 2277-3061

[2] Amazon web services economics center.http://aws.amazon.com/economics/.
[3] Cloud computing risk assessment. European Network andInformation Security Agency.November 20, 2009.
[4] Gone phishing. Twitter Blog. January 03, 2009.
[5] Linux kernel. Wikipedia.
[6] Liquid Motors, Inc. v. Allyn Lynd and United States ofAmerica. U.S. District Court for the Northern District ofTexas, Dallas Division. April 2009.
[7] Summary of Linux 2.6.32. h-online.com.
[8] Thread 37650: Email changes. Amazon Web ServicesDiscussion Forums.
[9] Trusted Computer System Evaluation Criteria (OrangeBook). Department of Defense Standard.DoD 5200.28-STD.December 1985.
[10] VMware Workstation. http://www.vmware.com/products/workstation/.
[11] Xen hypervisor.http://xen.org/products/xenhyp.html.
[12] Michael Armbrust et al. Above the Clouds: A Berkeley Viewof Cloud Computing. Technical report EECS-2009-28, UCBerkeley, http://www.eecs.berkeley.edu/Pubs/ TechRpts/2009/EECS-2009-28.html, Feb 2009.
[13] Robert Biddle, P. C. van Oorschot, Andrew S. Patrick,Jennifer Sobey, and Tara Whalen. Browser interfaces andextended validation ssl certificates: an empirical study. InCCSW '09: Proceedings of the ACM workshop on Cloudcomputing security.
[14] Kevin D. Bowers, Ari Juels, and AlinaOprea. Hail: ahigh-availability and integrity layer for cloud storage. In CCS'09: Proceedings of the 16th ACM conference on Computer
and communications security.
[15] P. Ceruzzi. An Interview with Robert E. Weissman. Charles Babbage Institute. May 3, 2002.
[16] Anton Chuvakin and Gunnar Peterson.Logging in the age ofweb services.IEEE Security and Privacy, 7(3):82–85, 2009.
[17] Fernando J. Corbató and V. A. Vyssotsky.Introduction andoverview of the multics system.IEEE Ann. Hist. Comput.,14(2):12–13, 1992.
[18] Chris Erway, AlptekinKüpçü, CharalamposPapamanthou,and Roberto Tamassia.Dynamic provable data possession. InCCS '09: Proceedings of the 16th ACM conference on
Computer and communications security.
[19] H. Feinleib. A Technical History of National CSS.Computer History Museum. April 2005.
[20] M. C. Ferrer. Zeus in-the-cloud.CA Community Blog.December 9, 2009.
[21] G. Fowler and B. Worthen. The internet industry is on acloud—whatever that may mean. Wall Street Journal. March26, 2009.
[22] C. Herley. Economics and the underground economy.BlackHat USA 2009.

[23] P. Karger. Securing virtual machine monitors—what is needed. Keynote address, ASIACSS 2009.
[24] E. Knorr. Gmail follies and Google's enterprise pitch.InfoWorld. September 8, 2009.
[25] K. Kortchinsky. Cloudburst—a VMware guest to host escapestory. Black Hat USA 2009.
[26] Stuart E. Madnick and John J. Donovan. Application andanalysis of the virtual machine approach to informationsystem security and isolation. In Proceedings of theworkshop on virtual computer systems.ACM, 1973.
[27] V. McLellan. Case of the purloined password.New YorkTimes. July 26, 1981.http://www.nytimes.com/1981/07/26/business/case-of-the-purloined-password.html.
[28] H. Meer, N. Arvanitis, and M. Slaviero.Clobbering thecloud.Black Hat USA 2009.
[29] P. Mell and T. Grance.Effectively and securely using thecloud computing paradigm.National Institute of Standardsand Technology. October 7, 2009.
[30] P. Mell and T. Grance. NIST definition of cloud computing.National Institute of Standards and Technology. October 7,2009.
[31] D. Raywood. The twitter hacking incident last week shouldbe a call to better security awareness and not about cloudstorage. SC Magazine. July 20, 2009.
[32] Thomas Ristenpart, EranTromer, HovavShacham, andStefan Savage. Hey, you, get off of my cloud: exploringinformation leakage in third-party compute clouds. In CCS'09: Proceedings of the 16th ACM conference on Computerand communications security.
[33] Jerome H. Saltzer. Protection and the control of informationsharing in multics.Commun. ACM, 17(7):388–402, 1974.
[34] N. Santos, K. P. Gummadi, and R. Rodrigues. Towardstrusted cloud computing. Hot Cloud 2009.http://www.usenix.org/event/hotcloud09/tech/full_papers/santos.pdf.
[35] S. Shankland. HP's Hurd dings cloud computing, IBM.CNET News. October 20, 2009.
[36] Dawn Xiaodong Song, David Wagner, and XuqingTian.
Timing analysis of keystrokes and timing attacks on SSH. InSSYM'01: Proceedings of the 10th conference on USENIXSecurity Symposium.
[37] A. Stamos, A. Becherer, and N. Wilcox.Cloud computingsecurity—raining on the trendy new parade.Black Hat USA2009.
[38] J. Stokes. T-Mobile and Microsoft/Danger data loss is badfor the cloud. Arstechnica. October 2009
.[39] T. van Vleck. How the Air Force cracked multics Security.multicians.org. May 21, 1993.
[40] K. Vikram, AbhishekPrateek, and Benjamin Livshits.Ripley: automatically securing web 2.0 applications throughreplicated execution. In CCS '09: Proceedings of the 16th
ACM conference on Computer and communications security.