Report from Dagstuhl Seminar 12281

Security and Dependability for Federated Cloud Platforms

Edited by

Rüdiger Kapitza[1], Matthias Schunter[2], Marc Shapiro[3], Paulo Verissimo[4], and Michael Waidner[5]

1 **TU Braunschweig, DE,** `kapitza@ibr.cs.tu-bs.de`
2 **INTEL CRI-SC, Darmstadt, DE,** `mts@schunter.org`
3 **INRIA & LIP6, Paris, FR,** `Marc.Shapiro@acm.org`
4 **University of Lisboa, PT,** `pjv@di.fc.ul.pt`
5 **Fraunhofer SIT - Darmstadt, DE,** `waidner@sit.fraunhofer.de`

Abstract

From July 8-13, 2012, the Dagstuhl Seminar "Security and Dependability for Federated Cloud Platforms" was held in Schloss Dagstuhl – Leibniz Center for Informatics. During this seminar, participants presented their current research and discussed open problems in the fields of security and dependability of infrastructure clouds and their federation. The executive summary and abstracts of the talks given during the seminar are put together in this paper.

Seminar 8.–13. July, 2012 – http://www.dagstuhl.de/12281
1998 ACM Subject Classification K.6.5 Security and Protection
Keywords and phrases Security, Cloud computing, Virtual systems, Dependability, Grid computing, Systems management
Digital Object Identifier 10.4230/DagRep.2.7.56

1 Executive Summary

Rüdiger Kapitza
Matthias Schunter
Marc Shapiro
Paulo Verissimo
Michael Waidner

Computing services are increasingly pooled within global utility computing infrastructures offered by providers such as Amazon, Google and IBM. Infrastructure clouds provide virtual machines and resources. These infrastructure clouds are used to enable "platforms as a service" that simplify implementation of arbitrary scalable services.

The seminar targeted the management and protection of individual clouds and addressed the trend towards cloud federation by bringing together researchers from systems management, security, and dependability. The idea was that only such an integrated approach is able to guarantee security and dependability while preserving the essential cost and efficiency benefits of today's emerging solutions.

The challenge to address was how to provide secure and dependable services on such federated cloud platforms. Selected research questions were: How can clouds securely interoperate, how can service availability be guaranteed despite failures or attacks by individual clouds, how can existing algorithms be adjusted to provide scalable eventual consistency, and finally whether cloud-of-cloud infrastructures can provide such benefits at costs that are competitive with single cloud solutions. While these questions where addressed during the seminar it got also clear that dependability and security of single clouds is by far not solved and therefore was also discussed in depth.

2 Table of Contents

Executive Summary

Overview of Talks

3 Overview of Talks

3.1 Capacity Planning for Clouds: Problems, Solutions, Consequences

Artur Andrzejak (Universität Heidelberg, DE)

Joint work of Andrzejak, Artur; Kondo, Derrick; Yi, Sangho; Anderson, David P.
Main reference Artur Andrzejak, Derrick Kondo, and Sangho Yi: Decision Model for Cloud Computing under SLA Constraints, 18th Annual Meeting of the IEEE/ACM International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS 2010), Miami Beach, Florida - August 17-19, 20

A major promise of cloud computing is that a customer can change resource capacity on-demand. Combined with multiplexing demands of (uncorrelated) customers, this seems to solve the capacity (planning) problem for cloud providers and their clients. We argue that despite of this obvious progress, there are still significant research challenges and as well as opportunities on both sides.

First we analyse the provider's strategies to handle demand spikes which are likely to be caused by correlated demand. Among different options, we take a closer look at the approach taken by Amazon's EC2, namely to provision for the worst case but sell unused capacity at a discount (via Spot Instances offered at a fluctuating market price). We study how this market approach offers to users opportunity to trade various QoS aspects (e.g. duration of a batch computation, number of interruptions) against the total cost. Our (quite surprising) conclusion is: it is not worth to bid low but it is significant to select the right instance type.

In the further part of the talk we discuss a technique for increasing user's computational capacity under cost constraints. It works by mixing resources of different availability levels (e.g. voluntarily resources, Spot Instances, and (highly available) dedicated resources according) in proportions which guarantee both a certain availability level and a cap on the total costs. An important aspect here is that we request the user to replace traditional notion of availability (of individual machines) by collective availability, which only guarantees that some fraction of resources within a set will be available in a time interval.

Such "tweaking" of QoS notions is instrumental also in the third topic of the talk: cost-efficient backup storage via private clouds. The key idea is to use dedicated and non-dedicated institutional machines together, and lower the storage costs by trading these costs against QoS metrics such as time until data restore completes. Also this approach illustrate that a changed understanding of common QoS notions can provide substantial savings for users.

References

1 Artur Andrzejak, Derrick Kondo, and Sangho Yi, "Decision Model for Cloud Computing under SLA Constraints," in *18th Annual Meeting of the IEEE/ACM International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS 2010)*, Miami Beach, Florida - August 17-19, 2010.

2 S. Yi, D. Kondo, and A. Andrzejak, "Reducing costs of spot instances via checkpointing in the amazon elastic compute cloud," in *The 3rd International Conference on Cloud Computing (CLOUD'10)*, July 2010, pp. 236–243.

3 A. Andrzejak, D. Kondo, and D. P. Anderson, "Exploiting non-dedicated resources for cloud computing," in *12th IEEE/IFIP Network Operations & Management Symposium (NOMS 2010)*, Osaka, Japan, Apr 19–23 2010.

4 D. Kondo, A. Andrzejak, and D. P. Anderson, "On correlated availability in internet distributed systems," in *IEEE/ACM International Conference on Grid Computing (Grid)*, Tsukuba, Japan, 2008.

3.2 Towards a Cloud-of-Clouds File System

Alysson Neves Bessani (University of Lisboa, PT)

Joint work of Bessani, Alysson; Mendes, Ricardo; Oliveira, Tiago; Pasin, Marcelo

Recently it was shown that the use of multiple cloud providers for building cloud storage services can amend many concerns related to the lack of trust in storing critical data in the cloud. However, these services are still difficult to use when compared with local storage systems, e.g., a file system. In this talk we'll present the design of C2FS, a multi-client cloud-of-clouds file system that provides near- POSIX semantics (including strong consistency) even if a subset of the providers are subject to arbitrary faults. The design of C2FS leads to the development of new abstractions that may be useful for other cloud-backed files systems, or even distributed file systems in general.

3.3 Old and new security issues in the Cloud

Herbert Bos (VU - Amsterdam, NL)

The question we address is: to what extent are clouds different from other forms of computing?

We can loosely describe the cloud as a scalable solution where the more you pay, the more you get. Cycles, storage, bandwidth, availability – anything. It is interesting to note that the same description fits some of the modern botnets. We have compute botnets, DDoS botnets, botnets that steal information, and so on. The specialisation, flexibility and support provided by modern malware just about qualifies them to be called Malware as a Service (MaaS).

Besides noticing the remarkable similarities between clouds and botnets, the real research question in this presentation was whether or not (public) clouds are fundamentally different from local computation – from a security point of view. In other words, which security issues are fundamental to cloud computing and which are more or less similar? The talk proposed several such issues to serve as inputs for debate.

Positives

1. Clouds allow us to scale expert administration. On the one hand, the need for computation and storage keeps growing and so does the complexity of the systems that provides these services. On the other, we see no such growth in the number of expert administrators. There is a risk that expert administrators will become unaffordable in the near future for many organisations, with serious security implications. The cloud allows expert administrators to handle a much larger group of customers who can enjoy proper and secure administration of their systems.
2. Consolidation makes it easier to detect new attacks. The presence of many systems from many clients in the same data center makes it possible to detect anomalous behaviour earlier and more accurately.
3. Software monocultures *may* allow us to spread expensive security checks over multiple installations (collaborative security) and to spread cures quickly. A similar point was argued by Michael Locasto et al. in NDSS 2006 for the monoculture provided by MS Windows.

Negatives:

1. You make yourself a bigger target. For attackers, it may be more interesting to attack a system that hosts many applications from many different customers.
2. You become vulnerable to information loss/theft by the cloud provider (or perhaps a disgruntled employee) – neither of whom are under your control. Homomorphic encryption is too expensive to be a real solution in general in the foreseeable future. You may spread your trust over multiple clouds, but then cost equations change.
3. Software monocultures can also be a threat (and is frequently perceived as such). Exploits and malware spread faster.
4. You rely on a payment scheme to pay for the cloud resources. The payment scheme itself may be targeted by attackers. An example is the Zeus attack on the Canadian company Ceridian (where the attackers stole the credentials for the use of a cloud-based payroll system and created fake employees to have them paid).
5. Black hats may use legitimate clouds for criminal purposes. Clouds are interesting for attackers only. They offer a lot of temporary compute power to crack passwords, or launch DoS attacks, etc. Thus, we see that cybercriminals move to the cloud also.

3.4 On Dependability of Hadoop MapReduce in Cloud Environments

Sara Bouchenak (INRIA Rhône-Alpes, FR)

Main reference Amit Sangroya, Damian Serrano, Sara Bouchenak. Benchmarking Dependability of MapReduce Systems. In Proceedings of the 31st IEEE International Symposium on Reliable Distributed Systems (SRDS), Irvine, CA, October 2012.
URL http://sardes.inrialpes.fr/research/mrbs/

MapReduce is a popular programming model for distributed data processing. Extensive research has been conducted on the reliability of MapReduce, ranging from adaptive and on-demand fault-tolerance to new fault-tolerance models. However, realistic benchmarks are still missing to analyze and compare the effectiveness of these proposals. To date, most MapReduce fault-tolerance solutions have been evaluated using microbenchmarks in an ad-hoc and overly simplified setting, which may not be representative of real-world applications. This talk presents MRBS, a comprehensive benchmark suite for evaluating the dependability of MapReduce systems. MRBS includes five benchmarks covering several application domains and a wide range of execution scenarios such as data-intensive vs. compute-intensive applications, or batch applications vs. online interactive applications. MRBS allows to inject various types of faults at different rates. It also considers different application workloads and dataloads, and produces extensive reliability, availability and performance statistics. We illustrate the use of MRBS with Hadoop clusters running on Amazon EC2, and on a private cloud.

3.5 Decentralized Software Verification for Trusted Computing in Federated Clouds

Gregory Chockler (IBM - Haifa, IL)

One of the chief benefits of federated cloud environments is their potential to offer their users a rich ecosystem of diverse software services hosted within the confines of the individual federation members. To enable the interested parties to safely use these services in their applications, they should be able to gain assurance that the behavior of the services complies with their advertised specifications. We propose a new approach allowing the users to gain assurance in the service integrity in a scalable fashion without relying on either a centralized certification authority or access to the actual implementation code. Our approach relies on the existing techniques for black-box testing that generate test suites covering all interesting behaviors described by the specification. In our framework, the specification is described as a collection of safety and liveness properties, each of which is expressed as a temporal logic formula or a finite automaton. Coverage of the specification is achieved by ensuring that generated test cases exercise all paths through the automaton. Each individual test case is then executed large number of times with varying input parameters values thus reducing the provider's ability to forge the execution results without having access to correctly behaving software. Finally, our method gains efficiency by applying the techniques of property testing, which allow the verifier to estimate whether an execution satisfies the property with a high probability by sampling a constant number of bits from it. The selections are independent, hence the confidence in the result can be increased by increasing the size of the sample.

3.6 Byzantine Fault-Tolerant MapReduce in Clouds-of-Clouds

Miguel Pupo Correia (IST - TU of Lisbon, PT)

Main reference Pedro Costa, Marcelo Pasin, Alysson N. Bessani, and Miguel Correia. Byzantine Fault-Tolerant MapReduce: Faults are Not Just Crashes. In Proceedings of the 2011 IEEE Third International Conference on Cloud Computing Technology and Science (CLOUDCOM '11), 2011.

MapReduce is a popular framework for processing large data sets in cloud environments. Both the original MapReduce implementation and the open source Hadoop can tolerate some classes of faults: worker crashes and file corruptions. However, there is evidence that other, more pernicious, failure modes can affect MapReduce job executions. First, accidental arbitrary faults due to hardware faults may corrupt the results of a job execution. Second, malicious arbitrary faults, due to malicious insiders or intrusions, can affect the correctness and liveness of a job execution. Finally, many cloud outages have been reported, so it may be desirable to run jobs in a federation of clouds to tolerate such faults.

The talk is about a MapReduce runtime that tolerates arbitrary faults and runs in a set of clouds (a 'cloud-of-clouds') at a reasonable cost in terms of computation and execution time. The replication of MapReduce to tolerate accidental arbitrary faults has already been explored in a previous work (presented at CloudCom'11). The novel challenges were to deal with malicious faults and to avoid sending through the internet the huge amount of data that would normally be exchanged between map and reduce tasks.

3.7 Resource and service sharing in clouds of mobile devices

Alexandra Dmitrienko (Fraunhofer Inst. - Darmstadt, DE)

Mobile phones are personal devices that provide useful services to their users, such as telephony, SMS messages and the Internet connection. While traditionally these services are consumed by the phone owners, we consider application scenarios where these resources are shared among multiple users within a cloud of mobile devices. However, the ability to share services and resources with other potentially untrusted entities exposes mobile devices to attacks. We discuss one possible solution to this problem based on the adaptation of the trusted virtual domains (TVDs) concept known in the PC world. TVDs allow for isolating private and public workloads in different domains. However, distributed and ad-hoc nature of a mobile cloud makes the application of a typical TVD design very challenging and requires significant refinements of the TVD architecture. Thus, we are investigating possible TVD designs for ad-hoc TVDs in order to provide secure service sharing among mobile devices.

3.8 Cost-efficient Robust Atomic Storage

Dan Dobre (NEC Laboratories Europe - Heidelberg, DE)

We address the problem of robustly sharing data among a possibly unbounded number of clients by leveraging a set of untrusted cloud servers. Robustness here means providing wait-free and atomic read/write access to shared data in the face of asynchrony, concurrency and the largest possible number of malicious failures by servers and clients. In this work, we present a protocol that features optimal worst-case read/write latency of two and three communication round-trips respectively. In addition, our protocol exhibits communication efficient reads, rendering our solution particularly suitable for sharing large data objects. As far as we are aware, this is the first result showing that the optimal read latency of two round-trips can be attained without relying on expensive public-key cryptography. Furthermore, that in the Byzantine context, rather than writing back full values when reading, it is sufficient to store pointers to values already held in storage. For these purposes, we are using a novel technique enabling readers to provably determine the progress of a (possibly concurrent) write operation by inspecting only a fraction of the servers accessed by that write. We give two alternative implementations of our technique, relying on one- way functions and secret sharing respectively. While the former is cost-efficient and has direct practical applicability, the latter shows that the same properties can be attained even in the presence of a computationally unbounded adversary.

3.9 Dart - a new programming language for structured web programming

Nicolas Geoffray (Google - Aarhus, DK)

Dart is a new programming language for creating structured web applications. It has an unsurprising and familiar syntax and it has been designed from ground up with performance and ease-of-use in mind. The presentation targeted the story behind Dart, and an introduction to the language, with an emphasis on its security features.

3.10 On the insecurity of Cloud configurations

Gabriela Gheorghe (University of Luxembourg, LU)

Joint work of Gheorghe, Gabriela; Crispo, Bruno; Carbone, Roberto; Desmet, Lieven; Joosens, Wouter
Main reference Gheorghe, Gabriela and Crispo, Bruno and Carbone, Roberto and Desmet, Lieven and Joosen, Wouter; Deploy, adjust and readjust: supporting dynamic reconfiguration of policy enforcement; Proceedings of the 12th ACM/IFIP/USENIX international conference on Middleware; 2011; 350–369
URL http://dx.doi.org/10.1007/978-3-642-25821-3_18

For large distributed applications based on Cloud technologies, security and performance are two requirements often difficult to satisfy together. In the first part of this talk, I discuss the limits of Cloud abstractions when it comes to security: verification, cross-layer trust concerns, absence of semantics coupled with security policy languages, the need for security at the middleware layer. In the second part of the talk, I discuss the security implications of mishandling authorisation information, by subjecting it to performance-enhancing techniques like caching. I argue for the necessity of managing security data separately from application data, and in a way that adapts to the fluctuating tradeoff between the security and performance requirements of the Cloud application. This approach leaves open questions such as: what are the different constraints on handling application data? what are the ways to configure or misconfigure a Cloud from the point of view of management of security data? how to be sure that a distributed system is handling security data as it should?

3.11 Building specialized BFT protocols using Abortable abstractions

Nikola Knezevic (IBM Research - Zürich, CH)

Joint work of Guerraoui, Rachid; Knezevic, Nikola; Quema, Vivien; Vukolic, Mark
Main reference Rachid Guerraoui, Nikola Knezevic, Vivien Quéma, and Marko Vukolic;. 2010. The next 700 BFT protocols. In Proceedings of the 5th European conference on Computer systems (EuroSys '10). ACM
URL http://doi.acm.org/10.1145/1755913.1755950

Modern Byzantine fault-tolerant state machine replication (BFT) protocols are notoriously difficult to develop, test and prove, as their implementations span over 20000 lines of involved

C++ code, related to synchronization, network and cryptography. Furthermore, one-size-fits-all does not easily apply to BFT protocols, which need to tolerate all possible, even malicious, situations well, making their implementations even more complex. In order to remedy this situation, we propose a new abstraction to simplify the development and the analysis of BFT protocols. We treat a BFT protocol as a composition of instances of our abstraction, where each instance can *abort* if working conditions differ from its specification. Each instance is developed and analyzed independently. To illustrate our approach, we show how abortable BFT could be used to develop a highly specialized protocol — namely, one that achieves the highest possible throughput when there is no faults (the most common situation). To cover worst-case situations, we developed another protocol, but our abstraction allows for using any other, existing BFT protocol. Typically, a good choice is a classical one like PBFT which has been proved correct and widely tested. Finally, our specialized protocol, named Ring, achieves up to 27other protocols, requiring only about 6000 lines of C++ code.

3.12 Performance Dependability for Complex Cloud Applications

Guillaume Pierre (VU - Amsterdam, NL)

Joint work of Pierre, Guillaume; Dejun, Jiang; Wei, Zhou

Online cloud applications often receive widely varying workloads which make it hard to guarantee performance and financial hosting costs. One solution for this problem is resource provisioning, which aims at maintaining the end-to-end response time of a web application within a pre-defined range (Service Level Objective, or SLO).

Resource provisioning is hard even if we assume that applications and resources are fully homogeneous. However, both of these hypotheses are usually untrue. This presentation discusses how one may handle these two challenges.

First, online applications are usually not homogeneous but are rather composed of multiple services calling each other to provide the desired service. When the SLO is violated, a difficult decision is to choose which service(s) should be re-provisioned for optimal effect. We propose to assign an SLO only to the front-end service. Other services are not given any particular response time objectives. Services are autonomously responsible for their own provisioning operations and collaboratively negotiate performance objectives with each other to decide the provisioning service(s).

Second, cloud resources are usually not homogeneous. Even when creating multiple instances with the the exact same VM type, the performance one gets out of these VMs is extremely heterogeneous. This has important implications for resource provisioning: one now needs to benchmark each VM instance individually, in order to determine how each VM can be put to use for the best effect.

3.13 Towards Secure Cloud Applications using Information Flow Control

Peter R. Pietzuch (Imperial College London, GB)

Joint work of Pietzuch, Peter R.; Ioannis Papagiannis; Matteo Migliavacca, David M. Eyers, Brian Shand, David Evans, Petr Hosek, Jean Bacon

Main reference Matteo Migliavacca, Ioannis Papagiannis, David M. Eyers, Brian Shand, Jean Bacon, and Peter Pietzuch, "DEFCon: High-Performance Event Processing with Information Security", USENIX Annual Technical Conference (USENIX ATC'10), Boston, MA, USA, USENIX, 06/2010.

URL http://www.smartflow.org

Ensuring the confidentiality and integrity of data in cloud-deployed healthcare or financial applications is challenging. Developers may introduce unintended or deliberate security flaws in different parts of an application, which may lead to the disclosure of sensitive data, or there may be vulnerabilities in the cloud platform itself. While access control mechanisms and source code auditing are used in practice to avoid security flaws, security violations are nevertheless a frequent occurrence.

Instead of attempting to avoid all security flaws, we propose to provide a "safety net" to cloud-deployed distributed applications that prevents sensitive data disclosure from happening. Our approach is to use information flow control (IFC) to track the flow of data through a complex, heterogeneous distributed application and constrain undesirable flows that could violate data protection policy. Our DEFCon middleware demonstrates how an IFC model can be applied to Java applications by adding support for strong isolation between objects to the Java runtime system.

3.14 Diagnostics and Forensics in Clouds of Clouds

Hans Peter Reiser (Universität Passau, DE)

Loss of control is a problem commonly attributed to cloud computing. Users no longer have direct physical control over the resources they use. This problem not only rises the question of trustworthiness of the cloud-based infrastructure, but also complicates diagnosis and forensics targeted at the user's applications. With growing application complexity, applications are likely to be confronted with more frequent failures. If a user's application no longer works as expected, failure and root-cause analysis is essential. Similarily, criminal forensics also requires approaches to acquire detailled information about a target user's applications in the cloud. In a cloud-of-cloud situation, in which applications are replicated and/or migrated over multiple, independent, competing clouds, a comprehensive analysis is even more challenging.

3.15 Partial Replication in Multi-Clouds

Luis Rodrigues (Technical University - Lisboa, PT)

Joint work of Romano, Paolo; Carvalho, Nuno; Couceiro, Maria; Didona, Diego; Fernandes, Joao; Palmieri, Roberto; Peluso, Sebastiano; Quaglia, Francesco; Ruivo, Pedro; Bessani, Alysson

The first part of my talk makes an overview of the research that has been performed by the e Distributed Systems Group at INESC-ID on the topic Distributed Software Transactional Memories (DTM). Over the last 4 years we have published a number of papers based on prototypes that we have built to better understand the advantages and limitations of the DTM paradigm. These papers have been briefly introduced and put in the context of main research direction explored by our group in this domain.

The second part of the talk focused on one of these results, that we consider that can be potentially extended to run on multi-cloud environments. In particular, I have described GMU, a multiversion update-serializable protocol for genuine partial data replication. To the best of our knowledge, the first genuine, and hence highly scalable, multi-versioning protocol supporting invisible reads and wait-free read-only transactions, hence achieving excellent performance in read-dominated workloads, as typical of a wide range of real-world applications. Interestingly, achieving this result required introducing a slight relaxation of classic One Copy Serializability (1CS): GMU in fact guarantees update- serializability, a consistency criterion weaker than 1CS but still compliant with the ansi serializable isolation level.

Finally, the third part of the talk discussed how GMU may be extended to support partial replication in multi-clouds.

This work was partially supported by the FCT (INESC-ID multi annual funding through the PIDDAC Program fund grant, under project PEst-OE/ EEI/ LA0021/ 2011, and by the project PTDC/EIA-EIA/102212/2008) and also by project CloudTM (FP7-ICT 257784).

References

1 A. Adya. *Weak Consistency: A Generalized Theory and Optimistic Implementations for Distributed Transactions.* PhD thesis, MIT, 1999.

2 N. Carvalho, P. Romano, and L. Rodrigues. Asynchronous lease-based replication of software transactional memory. In *Middleware*, pages 376–396, 2010.

3 N. Carvalho, P. Romano, and L. Rodrigues. A generic framework for replicated software transactional memories. In *NCA*, 2011.

4 N. Carvalho, P. Romano, and L. Rodrigues. Scert: Speculative certification in replicated software transactional memories. In *SYSTOR*, 2011.

5 M. Couceiro, P. Romano, N. Carvalho, and L. Rodrigues. D2STM: dependable distributed software transactional memory. In *PRDC*, 2009.

6 M. Couceiro, P. Romano, and L. Rodrigues. A machine learning approach to performance prediction of total order broadcast protocols. In *SASO*, 2010.

7 M. Couceiro, P. Romano, and L. Rodrigues. Polycert: Polymorphic self-optimizing replication for in-memory transactional grids. In *Middleware*, 2011.

8 R. Palmieri, P. Romano, and F. Quaglia. Aggro: Boosting stm replication via aggressively optimistic transaction processing. In *NCA*, 2010.

9 A. Peluso, P. Ruivo, P. Romano, Quaglia F., and L. Rodrigues. When scalability meets consistency: Genuine multiversion update-serializable partial data replication. In *ICDCS*, 2012.

10 S. Peluso, J. Fernandes, P. Romano, F. Quaglia, and L. Rodrigues. SPECULA: speculative replication of software transactional memory. In *SRDS*, October 2012.

11 P. Romano, R. Palmieri, F. Quaglia, N. Carvalho, and L. Rodrigues. An optimal speculative transactional replication protocol. In *ISPA*, Taiwan, Taipei, 2010.

12 P. Romano and L. Rodrigues. An efficient weak mutual exclusion algorithm. In *ISPDC*, 2009.

13 P. Romano, L. Rodrigues, and N. Carvalho. The weak mutual exclusion problem. In *IPDPS*, 2009.

14 P. Ruivo, M. Couceiro, P. Romano, and L. Rodrigues. Exploiting total order multicast in weakly consistent transactional caches. In *PRDC*, 2011.

3.16 Policy-Sealed Data: A New Abstraction for Building Trusted Cloud Services

Nuno Santos (MPI für Softwaresysteme - Saarbrücken, DE)

Joint work of Santos, Nuno; Rodrigues, Rodrigo; Gummadi, Krishna; Saroiu, Stefan
Main reference Policy-Sealed Data: A New Abstraction for Building Trusted Cloud Services; Nuno Santos, Rodrigo Rodrigues, Krishna P. Gummadi, and Stefan Saroiu; Proceedings of USENIX Security, Bellevue, WS, USA, August, 2012
URL https://www.usenix.org/conference/usenixsecurity12/tech-schedule/technical-sessions

Accidental or intentional mismanagement of cloud software by administrators poses a serious threat to the integrity and confidentiality of customer data hosted by cloud services. Trusted computing provides an important foundation for designing cloud services that are more resilient to these threats. However, current trusted computing technology is ill-suited to the cloud as it exposes too many internal details of the cloud infrastructure, hinders fault tolerance and load-balancing flexibility, and performs poorly. We present Excalibur, a system that addresses these limitations by enabling the design of trusted cloud services. Excalibur provides a new trusted computing abstraction, called policy-sealed data, that lets data be sealed (i.e., encrypted to a customer-defined policy) and then unsealed (i.e., decrypted) only by nodes whose configurations match the policy. To provide this abstraction, Excalibur uses attribute-based encryption, which reduces the overhead of key management and improves the performance of the distributed protocols employed. To demonstrate that Excalibur is practical, we incorporated it in the Eucalyptus open-source cloud platform. Policy-sealed data can provide greater confidence to Eucalyptus customers that their data is not being mismanaged.

3.17 Swiftcloud: deploying conflict-free objects at large scale

Marc Shapiro (INRIA & LIP6, Paris, FR)

Joint work of Shapiro, Marc; Preguiça, Nuno; Baquero, Carlos; Zawirski, Marek
Main reference Conflict-free Replicated Data Types. 13th Int. Symp. on Stabilization, Safety, and Security of Distributed Systems (SSS). Grenoble, France, 10-12 October 2011
URL http://dx.doi.org/10.1007/978-3-642-24550-3_29

Conflict-Free Replicated Data Types (CRDTs) support Strong Eventual Consistency (SEC). They can be replicated at extremely large scale, while remaining extremely responsive, available, and fault tolerant.

In the first part of this talk, we first define SEC and CRDTs. Then we give some simple sufficient conditions for conflict-freedom in both the state-based (aka data-shipping) and the operation-based (aka function-shipping). Then we show how to design a CRDT, focusing on the example of a set, to ensure that it has intuitive semantics and that it uses memory efficiently.

In the second part, we describe the Swiftcloud CRDT store and its support for conflict-free transactions. Swiftcloud aims to deploy shared CRDT objects at extreme scale, very close to clients at the network edge. To make it easier to program with CRDTs, a conflict-free transaction presents the application with a consistent snapshot of the database and ensures that its results are transmitted atomically. We have implemented a social-network application; experiments show several orders of magnitude performance improvement over a more classical synchronisation-based approach.

3.18 Dynamic Reconfiguration of Primary/Backup Clusters (with application to Apache ZooKeeper)

Alexander Shraer (Yahoo! Research - Santa Clara, US)

Joint work of Shraer, Alexander; Reed, Benjamin; Malkhi, Dahlia; Junqueira, Flavio
Main reference Alexander Shraer, Benjamin Reed, Dahlia Malkhi and Flavio Junqueira, USENIX Annual Technical Conference 2012 and Hadoop Summit 2012
URL http://www.cs.technion.ac.il/ shralex/zkreconfig.pdf

Dynamically changing (reconfiguring) the membership of a replicated distributed system while preserving data consistency and system availability is a challenging problem. In this talk I will discuss this problem in the context of Primary/Backup clusters and Apache Zookeeper. Zookeeper is an open source system which enables highly reliable distributed coordination. It is widely used in industry, for example in Yahoo!, Facebook,Twitter, VMWare, Box, Cloudera, Mapr, UBS, Goldman Sachs, Nicira, Netflix and many others. A common use-case of Zookeeper is to dynamically maintain membership and other configuration metadata for its users. Zookeeper itself is a replicated distributed system. Unfortunately, the membership and all other configuration parameters of Zookeeper are static - they're loaded during boot and cannot be altered. Operators resort to "rolling restart" - a manually intensive and error-prone method of changing the configuration that has caused data loss and inconsistency in production. Automatic reconfiguration functionality has been requested by operators since 2008. Several previous proposals were found incorrect and rejected. We designed and implemented a new reconfiguration protocol in Zookeeper and are currently integrating it into the codebase. It fully automates configuration changes: the set of Zookeeper servers, their roles, addresses, etc. can be changed dynamically, without service interruption and while maintaining data consistency. By leveraging the properties already provided by Zookeeper our protocol is considerably simpler than state of the art in reconfiguration protocols. Our protocol also encompasses the clients – clients are rebalanced across servers in the new configuration, while keeping the extent of migration proportional to the change in membership.

3.19 Storing data to the Intercloud

Marko Vukolic (Eurecom, FR)

Joint work of Basescu, Cristina; Cachin, Christian; Eyal, Ittay; Haas, Robert; Sorniotti, Alessandro; Vukolic, Marko; Zachevsky, Ido

Main reference Cristina Basescu, Christian Cachin, Ittay Eyal, Robert Haas, Alessandro Sorniotti, Marko Vukolic, Ido Zachevsky: Robust data sharing with key-value stores. DSN 2012: 1-12

A key-value store (KVS) have become the most popular way to access Internet-scale 'cloud' storage systems. In short, KVSs offer simple functions for storing and retrieving values associated with unique keys. Precisely because of the limited interface of a KVS, textbook-style solutions for reliable storage either do not work or incur a prohibitively large storage overhead.

We present an efficient wait-free algorithm that emulates multi-reader multi- writer storage from a set of potentially faulty KVS replicas in an asynchronous environment. Our implementation serves an unbounded number of clients that use the storage concurrently. It tolerates crashes of a minority of the KVSs and crashes of any number of clients. Our algorithm minimizes the space overhead at the KVSs and comes in two variants providing regular and atomic semantics, respectively.

Compared with prior solutions, our algorithm is inherently scalable and allows clients to write concurrently. It is hence a desirable solution for improving data availability in the Intercloud setting, i.e., beyond the availability of a single cloud provider.

Participants

- Artur Andrzejak
Universität Heidelberg, DE
- Alysson Neves Bessani
University of Lisboa, PT
- Herbert Bos
VU - Amsterdam, NL
- Sara Bouchenak
INRIA Rhône-Alpes, FR
- Gregory Chockler
IBM - Haifa, IL
- Miguel Pupo Correia
IST - TU of Lisbon, PT
- Maria Couceiro
INESC-ID - Lisboa, PT
- Alexandra Dmitrienko
Fraunhofer Inst. - Darmstadt, DE
- Dan Dobre
NEC Laboratories Europe - Heidelberg, DE
- Kurt Geihs
Universität Kassel, DE
- Nicolas Geoffray
Google - Aarhus, DK
- Gabriela Gheorghe
University of Luxembourg, LU
- Stephan Groß
TU Dresden, DE
- Flavio Paiva Junqueira
Yahoo Research - Barcelona, ES
- Rüdiger Kapitza
TU Braunschweig, DE
- Nikola Knezevic
IBM Research - Zürich, CH
- Guillaume Pierre
VU - Amsterdam, NL
- Peter R. Pietzuch
Imperial College London, GB
- Hans Peter Reiser
Universität Passau, DE
- Luis Rodrigues
Technical University - Lisboa, PT
- Ahmad-Reza Sadeghi
TU Darmstadt, DE
- Nuno Santos
MPI für Softwaresysteme - Saarbrücken, DE
- Matthias Schunter
INTEL ICRI - Darmstadt, DE
- Marc Shapiro
INRIA & LIP6, Paris, France
- Alexander Shraer
Yahoo! Research - Santa Clara, US
- Radu Sion
Stony Brook University, US
- Jan Stoess
KIT - Karlsruhe Institute of Technology, DE
- Paulo Jorge Verissimo
University of Lisboa, PT
- Marko Vukolic
Symantec Research Labs - Biot, FR
- Michael Waidner
TU Darmstadt, DE
- Alexander Wiesmaier
AGT Group (R&D) GmbH - Darmstadt, DE