Towards User Centric Data Governance and Control in the Cloud

Stephan Groß and Alexander Schill

Technische Universität Dresden
Fakultät Informatik
D-01062 Dresden, Germany
{Stephan.Gross,Alexander.Schill}@tu-dresden.de

Abstract. Cloud Computing, i. e. providing on-demand access to virtualised computing resources over the Internet, is one of the current megatrends in IT. Today, there are already several providers offering cloud computing infrastructure (IaaS), platform (PaaS) and software (SaaS) services. Although the cloud computing paradigm promises both economical as well as technological advantages, many potential users still have reservations about using cloud services as this would mean to trust a cloud provider to correctly handle their data according to previously negotiated rules. Furthermore, the virtualisation causes a location independence of offered services which could interfere with domain specific legislative regulations. In this paper, we present an approach of putting the cloud user back into power when migrating data and services into and within the cloud. We outline our work in progress, that aims at providing a platform for developing flexible service architectures for cloud computing with special consideration of security and non-functional properties.

1 Motivation

The recent progress in virtualising storage and computing resources combined with service oriented architectures (SOA) and broadband Internet access has led to a renaissance of already known concepts developed in research fields like grid, utility and autonomic computing. Today, the term cloud computing describes different ways of providing on-demand and pay-per-use access to elastic virtualised computing resource pools [15]. These resources are abstracted to services so that cloud computing resources can be retrieved as infrastructure (IaaS), platform (PaaS) and software (SaaS) services respectively. The pay-per-use model of such service oriented architectures includes Service Level Agreements (SLA) negotiated between service provider and user to establish guarantees for required non-functional properties including mandatory security requirements. The (economical) advantages of this approach are fairly obvious: One saves costly investments for procuring and maintaining probably underused hardware and at the same time gains new flexibility to react on temporal higher demands.

Nevertheless, there are reasonable reservations about the deployment of cloud computing services, e.g. concerning data security and compliance. Most of these

J. Camenisch and D. Kesdogan (Eds.): iNetSec 2011, LNCS 7039, pp. 132–144, 2012.

concerns result from the fact, that cloud computing describes complex socio-technical systems with a high number of different kinds of stakeholders following different and possibly contradicting objectives. From a user's perspective, one has to hand over the control over his data and services when entering the cloud, i.e. the user has to trust that the cloud provider behaves in compliance with the established SLA. However, to actually agree on a specific SLA a user first has to assess his organizational risks related to security and resilience [5].

Current solutions that restrict the provision of sensible services to dedicated private, hybrid or so-called national clouds[1] do not go far enough as they reduce the user's flexibility when scaling in or out and still force him to trust the cloud provider. Furthermore, private clouds intensify the vendor lock-in problem. Last but not least, there is no support for deciding which services and data could be safely migrated to which cloud. Instead we demand new methods and technical support to put the user in a position to benefit from the advantages of cloud computing without giving up the sovereignty over his data and applications. In our current work, we follow a system oriented approach focussing on technical means to achieve this goal.

The remainder of this paper is structured as follows. We first refine our problem statement in section 2. Then, in section 3, we sketch our approach of developing a secure platform for easy and flexible cloud service architectures. Our solution is based on the idea of a personal secure cloud (Π-Cloud), i.e. the conglomerate of a user's resources and devices, that can be controlled by a specialized gateway, the so-called Π-Box. We elaborate on its basic components in section 3 and further exemplify how the Π-Box supports the controlled storage of a user's data in the cloud in section 4. We conclude with a discussion of our approach and compare it with related work in section 5. Finally, section 6 provides an outlook on future work.

2 Problem Statement

We identified security as a major obstacle that prevents someone to transfer his resources into the cloud. In order to make sound business decisions and to maintain or obtain security certifications, cloud customers need assurance that providers are following sound security practices and behave according to agreed SLAs [4]. Thus, our overall goal is the development of a flexible open source cloud platform that integrates all necessary components for the development of user-controlled and -monitored secure cloud environments. This platform should provide the following functionality:

[1] The mentioned cloud types define different deployment models of cloud computing systems. In contrast to public clouds that make services available to the general public, private clouds are operated solely for an organization although the resources used might be outsourced to some service company. Hybrid clouds describe a mixture of public and private cloud, i.e. when users complement their internal IT resources with public ones. The term national clouds describes a scenario, where the location of the cloud resource pool is restricted to one country or legislative eco-system like the EU.

1. Mechanisms to enable a *user-controlled migration of resources and data into the cloud*. These mechanisms should support (semi-)automatic configuration of cryptographic algorithms to simplify the enforcement of a user's security requirements as well as the dynamic selection of cloud providers that best fit the user's requirements and trust assumptions. Thus, we need a formalised way to acquire a user's requirements. Furthermore, we need to integrate the user's private resources and different cloud providers in our cloud platform, e.g. by using wrapper mechanisms or standardised interfaces.
2. A sound and *trustworthy monitoring system for cloud services* that is able to gather all relevant information to detect or even predict SLA violations without manipulations by the cloud provider under control. To support the configuration of the monitoring system, there should be some mechanism that derives relevant monitoring objectives from negotiated SLAs. Thus, we need a formalised language for machine-readable SLA focussing on the technical details of a cloud computing environment.
3. *Adaptation mechanisms optimizing the cloud* utilization according to user-defined constraints like cost, energy consumption as well as to react on SLA violations detected by the monitoring system in order to mitigate the resulting negative effects. This includes migration support to transparently transfer resources between different cloud providers as well as adaptation tools that leaves the resources at the chosen provider but transforms them to further meet the user's non-functional and security requirements.

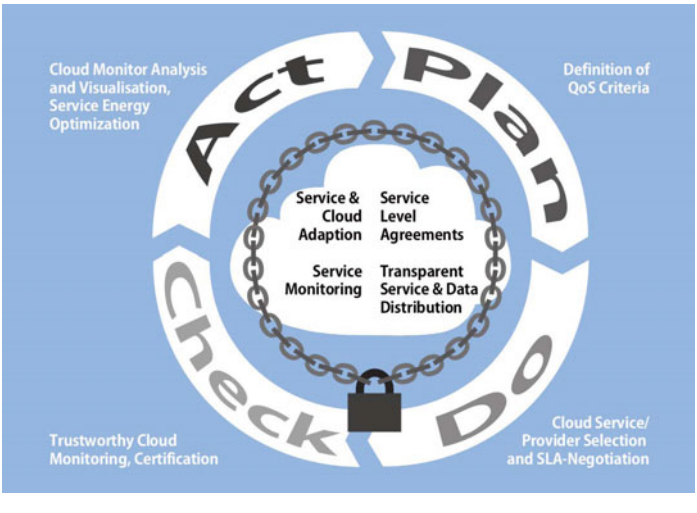

Fig. 1. Support for major cloud computing quality management objectives

In other words, we demand the implementation of an iterative quality management process to establish a secure cloud computing lifecyle that enables the

user to constantly supervise and control his cloud computing services. By applying the well-established PDCA model [13] the major objectives of such a cloud computing management process can be summarized as depicted in figure 1.

3 Introducing FlexCloud

Within the FlexCloud project we aim at developing methods and mechanisms to support the development of flexible and secure service architectures for cloud computing. Our major objective is to put the user in a position to externalize his IT infrastructure without losing control. For this, we have first refined the definition of cloud deployment models by introducing the concept of a personal secure cloud.

We define a *personal secure cloud* or Π-*Cloud* as a hybrid cloud that covers all ressources, services and data under complete control of a user. The user is able to dynamically adjust the Π-Cloud's shape according to his actual demands, i.e. to securely include foreign services and ressources as well as to securely share parts of his Π-Cloud with others.

Thus, we need to control the data-flow as well as the service distribution and execution. The technical means to control the Π-Cloud when sharing ressources or exchanging data are provided by the so-called Π-*Gateway*. The Π-Gateway provides all mechanisms to manage and optimize a user's policies concerning security and other non-functional properties, e.g. performance, energy-efficiency or costs. Furthermore, it provides the necessary means to enforce these policies such as adaptation and migration mechanisms for services and data.

To bridge the gap between a Π-Cloud's raw ressources, i.e. a user's devices, and the actually used services, we rely on a *service platform*. Its primary task is to dynamically allocate a user's software service to the available infrastructure services.

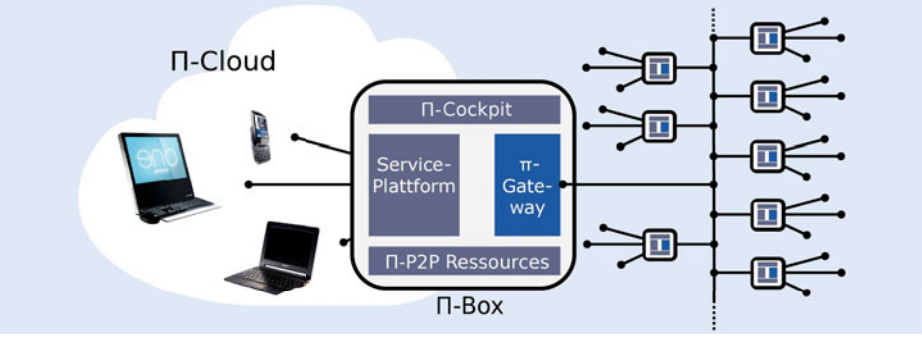

Fig. 2. Controlling the cloud with the Π-Box

Figure 2 shows our vision of a secure cloud computing setup. On the left hand side we have a user's personal devices building his Π-Cloud. It is controlled by his Π-Box (rectangle in the middle) that combines service platform and Π-Gateway. Depending on the Π-Cloud's size, the Π-Box can be realized physically (either as a separate hardware appliance or as a part of an existing device such as a router). It can also be virtualized, thus it can be migrated within the Π-Cloud or to some trustworthy cloud provider.

The following subsections give more details on the differnt parts of our approach.

3.1 Service Platform

A major foundation of our work is represented by our service platform SPACE. SPACE is an open source platform for the Internet of services which provides basic tools for contract-bound adaptive service execution and acts as a hosting and brokering environment for services. It already integrates techniques for trading Internet services and their surveillance during execution by the user as well as the service provider.

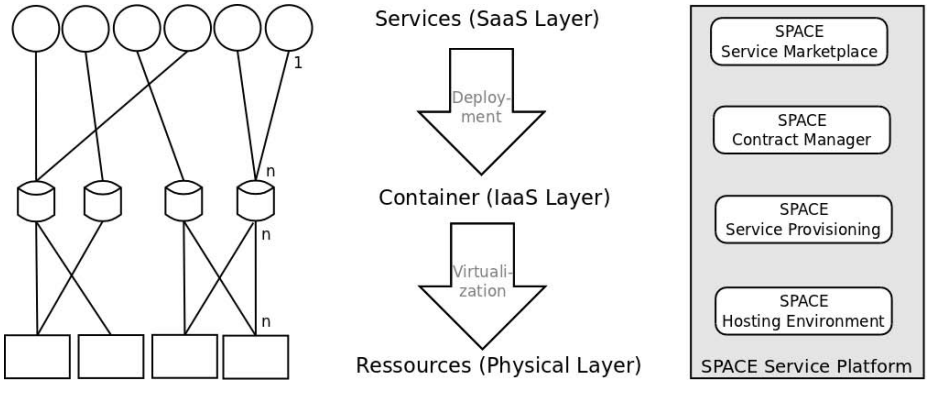

Fig. 3. Functionality of the SPACE service platform

Figure 3 summarizes the functionality implemented by SPACE. In general, SPACE provides a platform for building marketplaces for contract-bound adaptive service execution. The service marketplace and contract manager components comprise mechanisms for trading services, i.e. offering, searching, configuring, using and rating them. The service hosting enviroment binds heterogenous implementation technologies to a unified interface for service deployment, execution and monitoring. Overall, SPACE provides all necessary functions for the deployment of arbitrary software services on virtualized physical ressources provided as IaaS containers.

Being designed as a stand-alone server in the beginning, we are now working on bringing SPACE into the cloud, making it a fully distributed platform. Its latest extension already integrates Amazon EC2-compatible cloud environments as target environment for complex services delivered as virtual machines [20]. However, adding and removing new ressources and infrastructure services is still a rather static process.

3.2 Π-P2P Ressources

Π-P2P Ressources The Π-P2P Ressource component aims at improving the functionality to dynamically adjust the ressource and infrastructure pool available within a Π-Cloud. This includes the different devices in the Π-Cloud under the user's control as well as external cloud ressources (right part in figure 2). The on-going work on this topic are twofold.

On the one hand, we are further extending the SPACE by specific protocols to organize the physical ressource pool in a peer-to-peer network. This includes adaption algorithms to reorganize the container setup at the IaaS layer.

On the other hand, the Π-P2P Ressources component gets integrated with the Π-Gateway to implement a control flow that guarantees the combination of services and ressources according to a user's given policies.

For further details the interested reader is refered to [16].

3.3 Π-Gateway

We illustrate the Π-Gateway's functionality by an example: confidential cloud storage. Although there already exist first cloud storage solutions providing client-side encryption (e.g. [10]), it is cumbersome to integrate these systems into a company's existing IT infrastructure. For example, they introduce new potential for human errors as they ignore available access control systems and must be manually configured, e.g. by (re)defining the encryption keys for authorised users. It is also difficult to control if they comply with given regulations.

Figure 4 sketches how the Π-Gateway incorporates different tools, such as existing access control and user management systems, face recognition or information retrieval tools to determine the users authorised to access a specific file. For example, it could search a text file for a specific confidentiality note, analyse the people displayed on a foto, or simply check the file system's access rights in combination with the operating systems user database to determine the user identities to be granted access. By using these identities it is able to retrieve the necessary public keys from a public key infrastructure and to encrypt the data accordingly before storing it into the cloud.

Thus, the Π-Gateway consolidates all functionality to control and coordinate a user's cloud. This includes

- the management and enforcement of user policies concerning security and other non-functional properties;

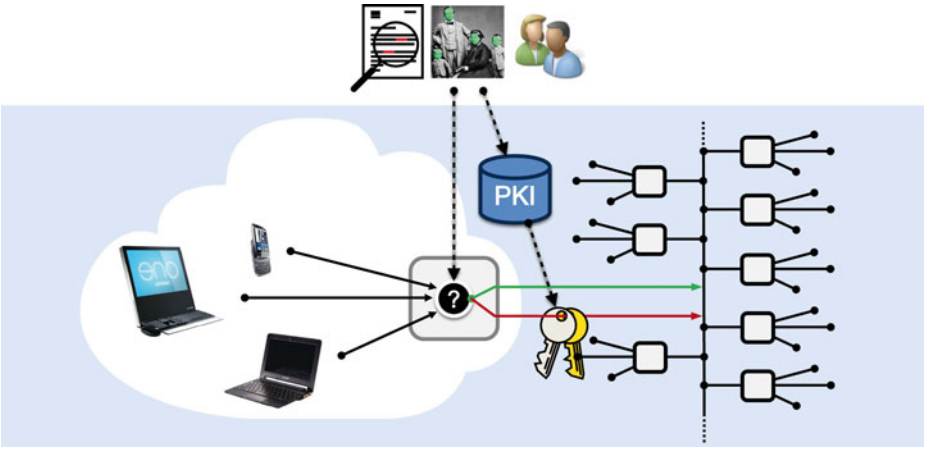

Fig. 4. Automized confidentiality for cloud storage

- the aggregation and analyzation of monitoring data retrieved by sensors implemented in the service platform as well as in the Π-P2P Ressources component;
- mechanisms to react on monitoring events and to optimize the ressource utilization of the Π-Cloud;
- technical foundations to improve the Π-Box's joy of use.

In other words, whereas the Π-Gateway represents the Π-Box's brain the service platform and the Π-P2P Ressources forms its body and extremeties respectively.

3.4 Π-Cockpit

Although we are aiming to include as much intelligence in our Π-Box to disburden the user from cumbersome administration tasks, it would be impudent to claim that our Π-Cloud is able to maintain itself. As we call for user-control, we need the necessary means to put him into this position even if he is not an expert. In short, we need a Π-Cockpit, i.e. adequate user interfaces to supervise and adjust the Π-Box. These user interfaces must be able to adjust to the user's skills and preferences as well as to the device's capabilities it is currently used on. The foundations for our work in this area are two-fold:

On the one hand, we aim at following up recent research in the area of usable security and privacy technologies that proposes an interdisciplinary approach to support a user in meeting his security demands independent of his skills and expertise. For an overview on related work in this field we refer to [6, 8, 9].

On the other hand, we are in line with recent developments in the human-computer interaction community that started to discuss the impact of cloud computing on the design of the user experience [7].

4 A First Use Case: Enterprise Cloud Storage

As a first evaluation scenario for our approach we have chosen the cloud storage use case already mentioned in the previous section. Thus, our pool of physical ressources consists of disk storage space that is provided by different cloud storage providers as an infrastructure service.

4.1 Problems of Current Cloud Storage Solutions

Current cloud storage offers suffer from different issues. First of all, off-site data storage raises several security and privacy concerns. Due to the nature of cloud computing the user can usually be not be sure about the geographical placement of his data. This leads to a possible mismatch of legislative rules in the cloud user's and provider's country respectively. Recent media reports document that even geographic restrictions for used ressources and services as assured by some cloud providers cannot guarantee a user's security compliance [22].

As a solution to ensure confidentiality, several cloud storage providers therefor rely on cryptography. However, for sake of simplicity and usability, most of them retain full control of the key management. Thus, the user's data stored at a trustworthy provider might be safe from intruders but can still subject to internal attacks or governmental desires. Even more, this increases the user's dependability on a specific storage provider leading to the point of a complete vendor-lockin and a possible loss of availability.

With respect to the functionality stated in section 2 we claim the following requirements for an ideal cloud storage solution:

User-controlled migration: The user should always be in the position to decide which data shall be migrated to which cloud storage provider. Furthermore, he should be assisted in applying cryptographic tools to enforce his security policies. To ensure best possible trustworthiness these security tools must be applied at the user's premises or at least by a fully trusted third party.

Trustworthy monitoring: After having transfered his data to the cloud the user should be able to control the reliability and trustworthiness of the chosen cloud storage providers. This includes audit mechanisms for the preservation of evidence to support subsequent legal enforcement.

Adaptation mechanisms: Finally, the user should be supported when recovering from detected malfunctions or inadequateness, e.g. securely restoring data stored at a provider or migrating it to another more trusted one.

4.2 Proposed Solution

We have developed a first prototype of a cloud storage integrator that aims at providing the stated functionality. Our prototype called SecCSIE (Secure Storage Integrator for Enterprises) implements a Linux based proxy server to be placed

in a company's intranet. It mediates the data flow to arbitrary cloud storage providers and provides a SMB/CIFS file based access to them for the average users. SecCSIE consists of five major components (for more details please refer to [19]):

Cloud Storage Protocol Adapter: To integrate and homogenize multiple cloud storage services we have implemented several protocol adapters. This includes adapters for common protocols like NFS, SMB/CIFS, WebDav and (S)FTP to access files over an IP network. Furthermore we provide access to Amazon S3, Dropbox and GMail storage by using existing FUSE (Filesystem in Userspace) models.

Data Dispersion Unit: Besides the cloud storage protocol adapter our data dispersion unit contributes to overcoming the vendor-lockin. By utilizing recent information dispersal algorithms [17] it distributes the user's data over different storage providers with higher efficiency than simple redundant copies. This also increases the overall availability and performance.

Data Encryption Unit: The encryption unit encapsulates different encryption algorithms to ensure confidentiality of the data stored. It also takes measures to preserver the stored data's integrity, e.g. by using AES-CMAC. The necessary key management can be handled by SecCSIE itself or delegated to an existing public key infrastructure.

Metadata Database: Within the metadata database all relevant information are collected to reconstruct and access the data stored in the cloud. Overall, this includes configuration parameters for data dispersion and encryption unit. Thus, the metadata database is absolutely irreplaceable for the correct functioning of our storage integrator.

Management Console: The management console implements a very straightforward web-based user interface to control SecCSIE's main functions. It provides rudimentary methods with which an enterprise's system administrator can check and restore the vitality of his storage cloud. Figure 5 gives an overview of the management console. Specific monitoring or configuration tasks can be accessed via the menu bar at the top or by clicking the respective component in the architecture overview.

The cloud storage protocol adapter together with the data dispersion and encryption units contribute to our overall objective of user-controlled migration. The trustworthy monitoring is accomplished by the storage proxy itself using the integrity checks provided by the data encryption unit as well as frequent checks of the network accessability of each storage provider. The management console provides an easy to use interface for this process so that one can estimate the reliability of the configured storage providers. Adaptation can be manually triggered in the management console. Furthermore, if an integrity check fails

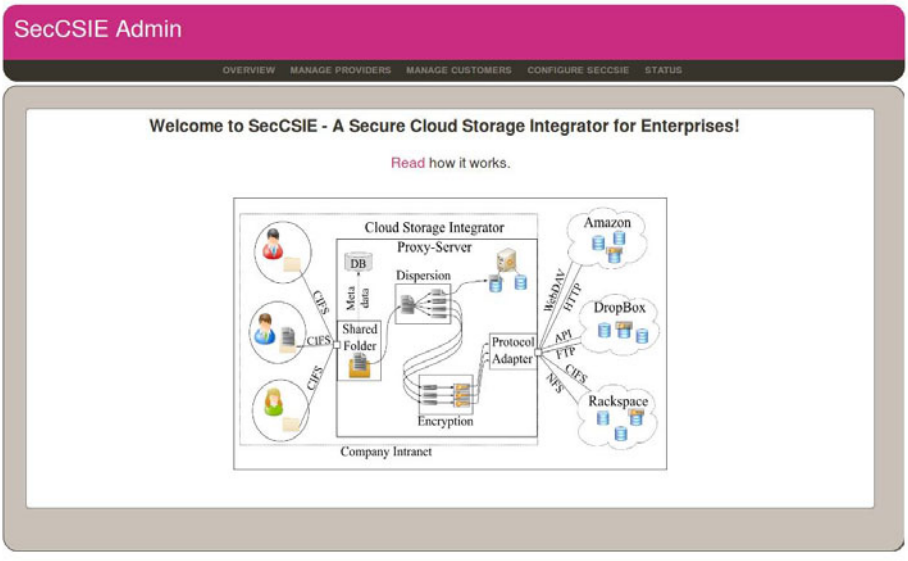

Fig. 5. Preliminary management console showing system architecture and data flow

when accessing a file, it can be automatically restored by switching to another storage provider and data chunk. The number of tolerable faults depends on the configuration of the dispersion unit, thus, it can be adjusted to the user's preferences.

5 Discussion and Related Work

In contrast to a common hybrid cloud, a Π-Cloud provides the following advantages:

- The user of a Π-Cloud retains full control over his data and services respectively.
- The user gains improved scalability as the Π-Gateway provides dedicated mechanisms to securely externalize data and services according to his security policies.
- The user no longer suffers from a vendor-lockin as the Π-Gateway integrates arbitrary service provider into a homogeneous view.

Thus, we achieve our goal of user-controlled migration into the cloud. The Π-Gateway together with the Π-Cockpit also provide a framework for implementing a sound monitoring system. Together with the Π-P2P Ressources framework it provides broad support for adaption and optimization scenarios.

Being more or less a topic only for industry in the beginning, cloud computing has seen more and more interest by academia over the last 3 years. Thus, there

exist several work with similar approaches to FlexCloud. The recently initiated Cloud@Home project [2] for example also aims at clients sharing their ressources with the cloud. Although the project also addresses SLAs and QoS it only sets a minor focus on security. The same applies for Intel's recent cloud initiative [12].

A major part of current research work on cloud computing is about cloud storage. Most theoretical publications in this area like [11, 21] apply existing algorithms from cryptography, peer-to-peer networking and coding theory to improve the integrity and availability of cloud storage. More sophisticated approaches lik [14] argue for a virtual private storage service that provides confidentiality, integrity and non-repudiation while retaining the main benefits of public cloud storage, i.e. availability, reliability, efficient retrieval and data sharing. However, although promising these approaches are still impractical to use. Other work tries to predict the required storage space to optimize the ressource allocation [3] or aims at the better integration with existing IT infrastructures [23]. However, to our best knowledge none of these work has presented a usable prototype implementation. On the practical side of the spectrum there are works like [1] or [18]. Both provide a similar approach to ours but only provide webservice based access to the storage gateway that complicates an integration with existing environments.

6 Conclusion

We have presented the overall objectives and first results of the FlexCloud project. In general, we are trying to keep the cloud user in control when using cloud services. We aim at providing a platform, the Π-Box, that provides all functionality to span a so-called Π-Cloud for flexible and secure cloud computing applications. As a first evaluation scenario we have chosen the use case of enterprise cloud storage for which we have implemented an initial prototype of the Π-Box called SecCSIE.

Concerning future work, our short-term objectives aim at consolidating the results achieved with SecCSIE. This includes further testing and optimization especially with respect to performance evaluation. We also plan to improve our monitoring and optimize our data dispersion mechanisms with respect to the user's requirements. Long-term objectives include the generalization of the storage scenario to other service types. We are especially interested in dynamic ressource allocation, e.g. by means of peer-to-peer mechanisms. Furthermore we plan to investige the implementation and evaluation of user interfaces with respect to his role and skills to improve the surveillance and management of the Π-Cloud.

Acknowledgement. The authors would like to express their gratitude to all members of the FlexCloud research group, especially its former member Gerald Hübsch, for many fruitful discussions that contributed to the development of the ideas presented in this paper.

This work has received funding under project number 080949277 by means of the European Regional Development Fund (ERDF), the European Social Fund

(ESF) and the German Free State of Saxony. The information in this document is provided as is, and no guarantee or warranty is given that the information is for any particular purpose.

References

1. Abu-Libdeh, H., Princehouse, L., Weatherspoon, H.: RACS: a case for cloud storage diversity. In: Proceedings of the 1st ACM Symposium on Cloud Computing, SoCC 2010, pp. 229–240. ACM, New York (2010), http://doi.acm.org/10.1145/1807128.1807165
2. Aversa, R., Avvenuti, M., Cuomo, A., Di Martino, B., Di Modica, G., Distefano, S., Puliafito, A., Rak, M., Tomarchio, O., Vecchio, A., Venticinque, S., Villano, U.: The Cloud@Home Project: Towards a New Enhanced Computing Paradigm. In: Guarracino, M.R., Vivien, F., Träff, J.L., Cannatoro, M., Danelutto, M., Hast, A., Perla, F., Knüpfer, A., Di Martino, B., Alexander, M. (eds.) Euro-Par-Workshop 2010. LNCS, vol. 6586, pp. 555–562. Springer, Heidelberg (2011)
3. Bonvin, N., Papaioannou, T.G., Aberer, K.: A self-organized, fault-tolerant and scalable replication scheme for cloud storage. In: Proceedings of the 1st ACM Symposium on Cloud computing (SoCC 2010), pp. 205–216. ACM, New York (2010)
4. Catteddu, D.: Cloud Computing – Benefits, risks and recommendations for information security. ENISA Report, ENISA (November 2009)
5. Catteddu, D.: Security & Resilience in Governmental Clouds – Making an informed decision. ENISA Report, ENISA (January 2011)
6. Cranor, L.F., Garfinkel, S.L.: Designing Secure Systems That People Can Use. O'Reilly (September 2005) ISBN 978-0-596-00827-7
7. England, D., Randles, M., Taleb-Bendiab, A.: Designing interaction for the cloud. In: Proceedings of the 2011 Annual Conference Extended Abstracts on Human Factors in Computing Systems, CHI EA 2011, pp. 2453–2456. ACM, New York (2011), http://doi.acm.org/10.1145/1979742.1979582
8. Fischer-Hübner, S., Iacono, L.L., Möller, S.: Usable Security und Privacy. Datenschutz und Datensicherheit (DuD) (11), 773 (2010)
9. Garfinkel, S.L.: Design principles and patterns for computer systems that are simultaneously secure and usable. Ph.D. thesis, Massachusetts Institute of Technology (2005), http://simson.net/thesis/
10. Grolimund, D., Meisser, L., Schmid, S., Wattenhofer, R.: Cryptree: A folder tree structure for cryptographic file systems. Technical report, Purdue University, Department of Computer Science, West Lafayette, IN, USA (2006)
11. He, Q., Li, Z., Zhang, X.: Study on Cloud Storage System Based on Distributed Storage Systems. In: 2010 International Conference on Computational and Information Sciences, ICCIS (December 2010)
12. Intel Corporation: Benefits of a Client-aware Cloud. White Paper Client-aware Cloud Computing (2011), http://partnerzones.i41.nbsp.de/_misc/download.cfm?filepath=/4/0/0/4/Benefitsofclientawarecloud.pdf&filename=Benefitsofclientawarecloud&filetype=pdf&filename=Benefitsofclientawarecloud&filetype=pdf&fid=624
13. Information technology – Security techniques – Information security management systems – Requirements. No. 27001 in ISO/IEC Standard, International Organization for Standardization (2005)

14. Kamara, S., Lauter, K.: Cryptographic Cloud Storage. Tech. rep., Microsoft Research Cryptography Group (2011)
15. Mell, P., Grance, T.: The NIST Definition of Cloud Computing. Recommendations of the National Institute of Standards and Technology (NIST), Special Publication 800–145 (January 2011),
http://csrc.nist.gov/publications/drafts/800-145/Draft-SP-800-145_cloud-definition.pdf
16. Mosch, M.: User-controlled data sovereignty in the Cloud. In: Proceedings of the PhD Symposium at the 9th IEEE European Conference on Web Services (ECOWS 2011), Lugano, Switzerland (September 2011)
17. Resch, J.K., Plank, J.S.: AONT-RS: blending security and performance in dispersed storage systems. In: 9th Usenix Conference on File and Storage Technologies FAST 2011 (February 2011)
18. Schnjakin, M., Meinel, C.: Plattform zur Bereitstellung sicherer und hochverfügbarer Speicherressourcen in der Cloud. In: Sicher in die digitale Welt von morgen – 12. Deutscher IT-Sicherheitskongress des BSI. SecuMedia Verlag, Bonn (2011)
19. Seiger, R., Groß, S., Schill, A.: SecCSIE: A Secure Cloud Storage Integrator for Enterprises. In: International Workshop on Clouds for Enterprises (C4E). Luxemburg (September 2011)
20. Spillner, J.: Spaceflight – A versatile live demonstrator and teaching system for advanced service-oriented technologies. In: Crimean Conference on Microwave and Telecommunication Technology (CriMiCo), Sewastopol, Ukraine (September 2011) (accepted for publication)
21. Wang, C., Wang, Q., Ren, K., Lou, W.: Ensuring data storage security in Cloud Computing. In: Proceedings of the 17th International Workshop on Quality of Service, Charleston, SC, USA (2009)
22. Whittaker, Z.: Microsoft admits patriot act can access EU-based cloud data. ZDNet iGeneration Blog (June 2011),
http://www.zdnet.com/blog/igeneration/microsoft-admits-patriot-act-canb-access-eu-based-cloud-data/11225
23. Xu, P., Zheng, W., Wu, Y., Huang, X., Xu, C.: Enabling Cloud Storage to Support Traditional Applications. In: 5th Annual ChinaGrid Conference (2010)