Bridging the security drawbacks of virtualized network resource provisioning model

Ayush Sharma, Volker Fusenig and Ingmar Schoen
Fraunhofer Research Institution for Applied & Integrated Security
Parkring 4, 85748 Garching, Germany
Email: {firstname.lastname}@aisec.fraunhofer.de

Anand Kannan
School of Information and Communication Technology
KTH, Royal Institute of Technology
Stockholm, Sweden
E-mail: anandk@kth.se

ABSTRACT
Cloud networking receives a lot of attention from the research community, especially due to its ability to bridge the dependability gaps in the existing cloud service provisioning models by enabling provisioning of virtualized network resources and providing network guarantees to the end-user. In cloud networking, network resources shared between multiple tenants are virtualized, and provisioned to customers in an elastic fashion. However, the existing cloud networking systems have many drawbacks pertaining to security, management, and performance. Therefore, it is necessary to develop new security architectures and suitable algorithms to provide effective security to the virtualized network resources available in the cloud. In this paper, we propose a new architecture which focuses on providing a security mechanism for cloud network resource provisioning models. The central feature of this architecture is a hierarchical, multi-domain, and multi-level security goal translation function which promotes security of the virtualized network resources and trust management between the service providers.

Categories and Subject Descriptors
D.3.2, C.2.4, and D.4.0

General Terms
Algorithms, Design, Performance, Reliability, and Security

Keywords
Cloud networking, Security architecture, Dependability, Virtualization, Privacy

1. INTRODUCTION
Cloud computing has experienced an exponential growth and witnessed widespread industry acceptance in the previous decade. Cloud computing entails the virtualization of the underlying physical resource set, and provisions access to the cloud services for its consumers. Cloud computing involves a variety of service provisioning models, which include Software-as-a-Service (SaaS), Platform-as-a-Service (PaaS), and Infrastructure-as-a-Service (IaaS). Some common examples of the above service provisioning models include GoogleDocs [1] for SaaS, GooglesAppEngine[2] for PaaS, and Amazon's EC2 [3] for IaaS.

The most frequent concerns, while managing the risk involved in adopting cloud services in production environments, are dependability, latency, QoS, and Service level agreement (SLA) conformance on the underlying communications infrastructure. Networks are critically important to the overall cloud ecosystem, as the fundamental promise of cloud computing is to migrate the workload to the cloud, which can be then reliably accessed using the Internet. This makes network performance important both within the cloud environment and over the networks which are used to access the cloud resources and services. The European project SAIL [4] focuses on technologies that will enable provisioning of dynamic, virtualized, and elastic networking capabilities by utilizing the underlying network infrastructure. A cloud network architecture (CloNe) has been developed to provision virtualized network resources by utilizing the Network as a service (NaaS) provisioning model for the cloud ecosystem.

There is a strong demand for a well-defined security architecture, which is tightly integrated with the CloNe architecture, due to the considerable number of network-related security challenges and their serious impact to service delivery [5]. Moreover, the new technologies needed to deploy the NaaS model introduce additional security challenges. To address such security challenges, we propose a security framework for cloud networks which will be tightly integrated to the existing CloNe architecture. Moreover, a special emphasis has been made on the methodology behind translating the security-specific requests made by the cloud user in a high level language into concrete, low-level, and machine understandable language. The paper describes a security goal translation function, which accepts the security goals from the different entities in the CloNe infrastructure, and translates them into resource specifications which can be deployed on the underlying set of resources. The main contribution of this paper is the proposal of new security architecture and the security goal translation function for the cloud networking environment.

This paper is organized as follows. Section 2 provides a survey of related works in this area. Section 3 explains the CloNe architecture, and some use case scenarios. Section 4 describes the proposed cloud network security architecture, with special emphasis on the methodology behind translating the security-specific requests from the user into concrete, pareto-optimal resource specifications to be implemented on the underlying resource set. Section 5 describes the various security functions and their interactions. Section 6 provides the results and comparison, with respect to other cloud security architectures. Section 7 concludes the work and shows further working directions.

2. RELATED WORK
There are many works in the literature which discuss about security in the cloud computing ecosystem. Tripati et al.

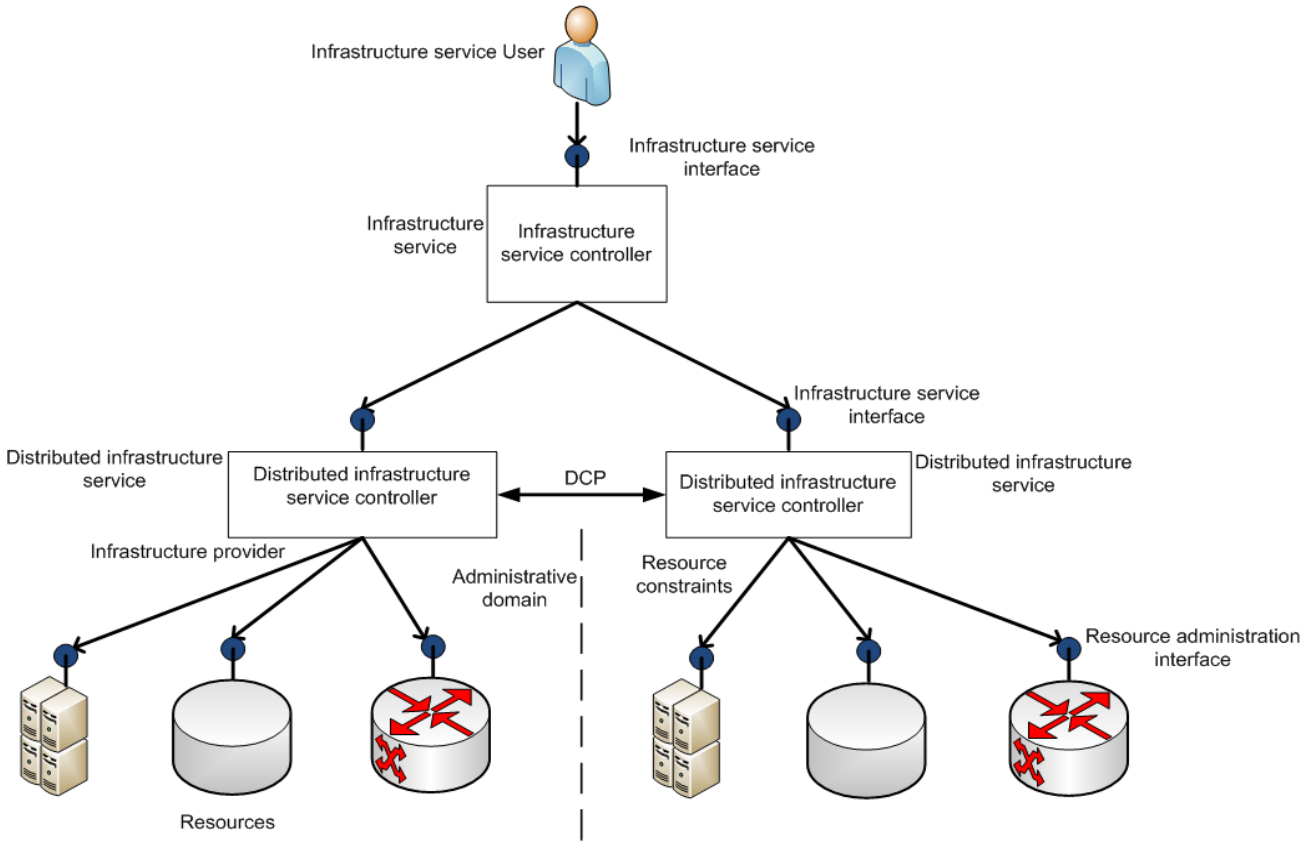

Figure 1: High-level CloNe architecture

[10] describes some of the key security issues that arise in a cloud computing environment. Virtualization is the key ingredient of the cloud ecosystem. Van Cleff et al. [11] performed a systematic literature review on the security effects of virtualization. They conclude with the fact that with virtualization, high availability and performance can be obtained. However, the effect on confidentiality and integrity is less positive. Yamuna devi et al. [12] discusses about the live migrations of VM in cloud environment. Their experimental observations conclude that new security challenges are introduced by these technologies. Srivastava et al. [13] analyzed the security landscape in detail and proposes a cloud security architecture. Their architecture uses role-based access control policies which addresses the security flaws introduced while accessing the cloud resources. However, other key security flaws such as virtualization level security challenges are not addressed by this architecture. Dayananda et al. [14] introduced IPsec VPNs to mitigate the security flaws of the underlying physical network. However, their architecture cannot be extended to multi domain and multi-level resource provisioning models.

3. PROPOSED CloNe ARCHITECTURE

CloNe focuses on two distinct scenarios of application. The first scenario is termed as *Dynamic enterprise*, which entails provisioning of IT/IS solutions from the cloud network ecosystem to the enterprise market. The use case is applicable if the infrastructure of an enterprise is partially/wholly shifted into the cloud. The second scenario is termed *Distributed cloud: Elastic video delivery*, which allows the offering of real time video via a cloud to the consumers. Both scenarios depict real world situations, and suffer from dependability, security, and performance problems due to an absence of a secure network resource provisioning solution integrated into existing provisioning models of the cloud. The first scenario requires an enterprise centric cloud networking solution which provides full resource isolation between tenants, both in the WAN and in the datacenters. Furthermore it requires programmability of network resources in the datacenter and WAN, dynamic scaling of virtual resources, for example computing and storage, and dynamic provisioning and scaling of network resources like bandwidth. The second scenario requires cloud network capabilities for dynamic resource provisioning and scaling of distributed virtual resources which are spread over an operator network. Moreover, distributed load balancing and optimal placement of content servers in the distributed cloud is essential to meet the requirements, for e.g., quality of service requested by a real time service.

Flash network slice (FNS) is a virtual network resource which would provide dynamic network resource provisioning and distributed processing capabilities in operator controlled network environments. A FNS is a resource which provides a network service. It can have multiple access points and implements forwarding between those access points. An FNS can be linked to other resources through connections. A VM may be connected to one FNS, or two FNSs can be connected to each other. An FNS can be provisioned inside a single administrative domain (single operator controlled environment). It has measurable and acceptable QoS and setup times. Finally, any required behavior

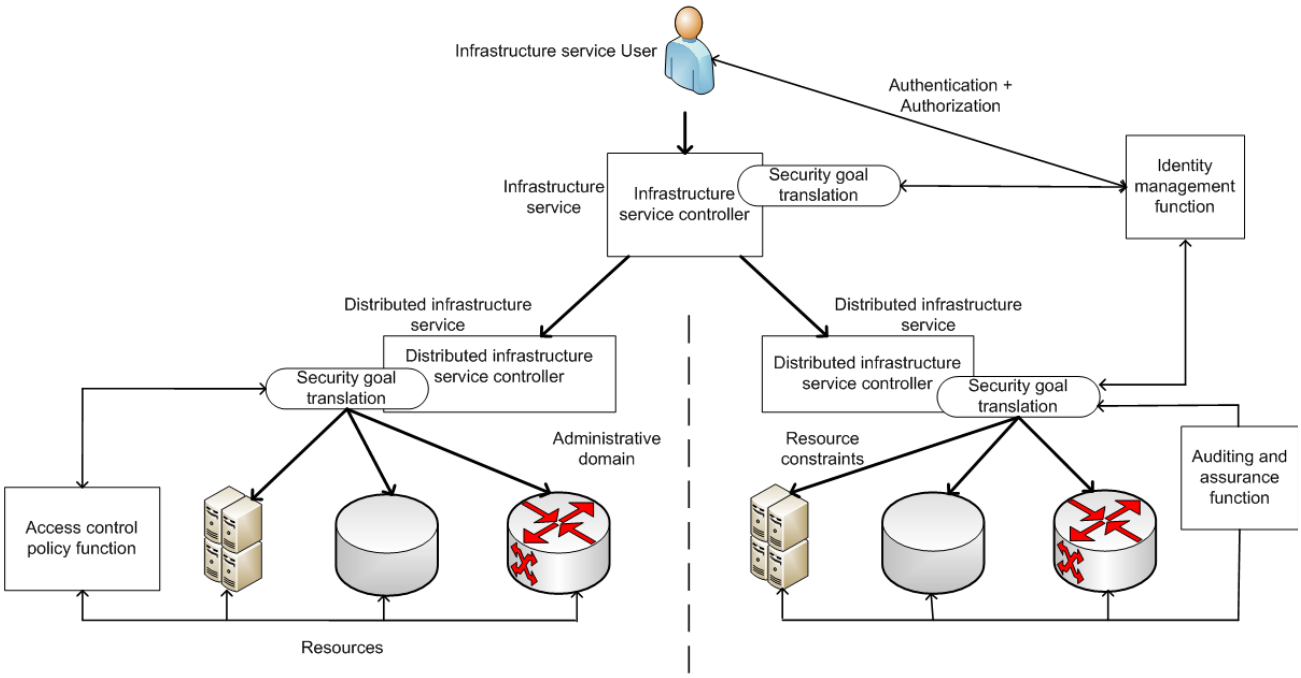

Figure 2: CloNe security architecture

from the underlying network would be expressed through the infrastructure service provider's interfaces, and require no network-specific implementation by the user.

The CloNe architecture shall try to fulfill the abstract requirements of the FNS, and allow dynamic, virtual, network resource provisioning to the users. The high level architecture of CloNe consists of four parts 1. Three layer Model, 2. Set of Roles, 3. A set of interfaces by which the roles interact, and 4. A set of management modules in which these roles participate. Figure 1 shows the high level architecture of CloNe. An administrative domain is a collection of physical or virtual equipment which is controlled by a single administrative authority but an infrastructure can span over more than one administrative domain. An administrative domain is controlled by the role *infrastructure provider*. An *infrastructure provider* could be an operator such as Deutsche Telekom, and could own and/or control a set of physical or virtual components. The second role is that of the *infrastructure service user*, who has been restricted as a cloud service tenant for the time being. The cloud service tenant may then provision the same service to the end users.

Management modules: The CloNe architecture requires a set of management APIs that shall be responsible for implementing the varied management tasks, namely goal translation, fault management, resource management, and security management. **Goal translation** module provides the central backbone behind all goal translations (the security goal translation is a security-specific extension to the central goal translation module) and is used to translate and optimize high-level objectives in to low-level objectives/resource configurations for a service request.

Security management module is the central module of this publication. Its main function is to translate security-specific requirements into resource configurations.

4. CloNe SECURITY ARCHITECTURE

The multi-level cloud network security architecture proposed in this paper is described in Figure 2. The current section describes the set of interfaces which allow the different roles to communicate with each other (especially while translating a security requirement) in detail.

The goal translation process is kicked off by the tenant, who shall request a service to the *infrastructure service*, using the *infrastructure service interface.* The *infrastructure service interface* allows the tenant to specify a set of high level (security) requests, which are encapsulated as abstract service level (security) objectives. The language chosen to depict these requests is VXDL [6], which satisfies the aim to allow the desired levels of abstraction while specifying requests, and ease of specifying the overall information security policy without getting into the specifics of the underlying hierarchy or architecture details.

VXDL acts as the core modeling language for applications which currently provide extensions to implement firewall rules on the underlying virtual infrastructure. Virtual networks allow three plausible placements for firewalls, viz. on the common service interface, network links between the VMs defined using VXDL and on the varied access points. Common service interfaces include the interfaces which are shown in Figure 1. Firewalls shall allow isolation between the services that are delivered by the different virtual machines in a specific domain. Isolation between the network links and demarcation of interaction between VMs inside a specific administrative domain, or between different domains are also controlled by firewalls. Finally, firewalls at the access points shall allow isolation of the service that the virtual machine is delivering, with inputs from regions exterior to its specific administrative domain.

The *infrastructure service* shall receive a (security) service request from the tenant (*infrastructure service user*). This request could be an entirely new request, or a delta of a previous request. Each request requires interaction with the model checker module to perform the above step (detect if request is new or delta). If the request is entirely new, all its composing entities, viz. the client name, VMs, network resources, KPIs employed by the user etc.

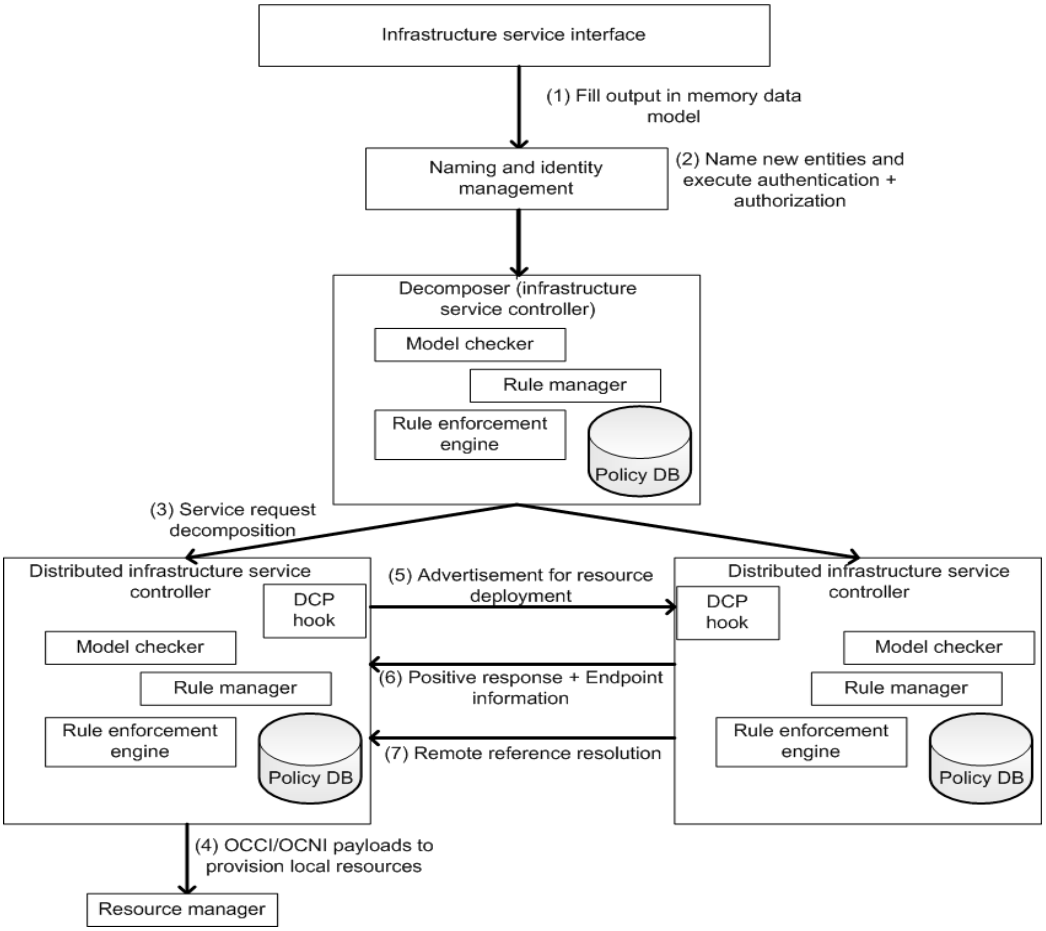

Figure 3: Security goal translation function

will have to be given a uuid by the naming module, and successively the named objects would be added in the in-memory data model. In-memory data models store models for each service request.

However, if the tenant request is a delta, the model checker module shall detect the modifications from the original request (with respect to which the current request is a delta) and implement the necessary changes. Once the request has been decoded, the request has to be cross-verified from the identity management function by utilizing its authentication and authorization functions. The tenant's request will be compared with its respective access control policy, and a decision will be taken to drop the request, or comply with it based on the usage rights of the tenant. Moreover, the user's identity also has to be authenticated.

The infrastructure service controller, which is the central controller responsible for the request translation/decomposition, shall then decompose the incoming request, and identify the resources which need to be provisioned to the tenant, and through which domain. Each distributed infrastructure service controller operating in each individual administrative domain receives a part of the request to be provisioned. If the local domain can't provision the entire service itself, it delegates the request (or part of it) to any other domain which is capable of meeting the request by advertising the request using the DCP (DCP, or distributed control plane is a cross-domain plane which allows different domains to communicate with each other in a RESTful manner) hook.

The advertisement creates a DCP topic with the uuid of the service, and the local domain subscribes to listen to all responses which fulfill the DCP request. A number of remote domains could answer via the DCP and provide their endpoint information. This initiates the remote reference resolution process, if the local domain decides to collaborate with the chosen remote domain through the DCP. The local domain requests the OCCI/OCNI (OCNI is the planned extension as a part of the SAIL project, which provides a suite of protocols and provide API support for dynamic network resource provisioning) server of the recently coupled remote domain to perform the deployment of the resources. During the deployment, each domain could communicate with the respective security functions for the deployment. Example interactions include utilizing the access control function, to deploy the resources compliant with the tenant-relevant access control policies. Successively, the identity management function can be used to authenticate the different resources/domains.

The request translation/decomposition is executed by the security goal translation function with the help of supporting management and security functions. The decomposition process also utilizes additional supporting modules, which include a rule manager, rule enforcement engine and a monitoring collector. The rule manager interacts with a policy database, which manages the storage of policies set by the different participating entities. The Rule manager shall be responsible for naming the rules, by interacting with the naming module, and performs update (CRUD) actions on the rules. Rules are then implemented and

enforced by the rule enforcement engine. Overall, the rule manager is responsible for validating the rule's syntax and organizes CRUD of the rules, with respect to a provisioned resource.

A Monitoring collector acts as a collection module for procuring the metrics which measure the service activity and performance. It is implemented as a pub/sub system with per-service topics, and future research includes adding add-on features to the module. The remote domain publishes the resolved references of the recently deployed resources, which the local domain receives by virtue of their subscription to the respective DCP topic. The actual deployment of the virtual resources is done by the resource manager function. The end user receives an ACK when all the requested service elements have been deployed and all the references have been resolved. The auditing and assurance function shall be invoked in the background by the security goal translation function, as soon as it received a service request. The auditing function shall audit all the actions carried out by the management and security functions. Whereas, the assurance function shall be useful to assure the tenant about the veracity of the properties of the provisioned resources and the participating entities.

Figure 3 depicts the entire security goal translation function diagrammatically.

5. SECURITY FUNCTION AND INTERACTIONS

The respective security functions and their interactions with the security goal translation function are depicted in Figure 2. The access control policy function aids the different entities in the CloNe environment to set and implement access control policies on the underlying resources, with respect to each *infrastructure service user*. The access control policies may either be directly specified by entities with plausible roles (viz. the tenant or infrastructure service user, infrastructure service or the infrastructure provider) or could be indirectly derived from the security goals specified by any of the entities described above.

The auditing and assurance function checks whether the parameter constraints, which have been defined by the goal translation function and need to be realized on the underlying hardware resources, have indeed been fulfilled or not, and under which capacity. The auditing mechanism is executed after periodic intervals, but could also be invoked upon request and/or need. The participating entities shall want to verify whether all the security mechanisms functioned properly during a specific interval of time, especially in the case of a security breach. The assurance function is responsible for assuring the infrastructure service user, besides other entities, regarding the properties of entities/resources communicating with it.

The identity management solution provides a total of five functionalities to support the overall security goal translation function, which include identity provisioning, authentication, federated identity management, authorization and user profile management and compliance. Identity provisioning promotes the secure and efficient management of provisioning and deprovisioning user identities, while authentication allows credential management, strong authentication and the option to choose the desired strength of authentication on the fly, delegated authentication and managed trust across all entities involved in the architecture.

Federated identity management empowers the cloud tenant to authenticate themselves using their desired identity provider. Therefore, an exchange of identity attributes takes place between identity providers and service providers. Authorization and user profile management is useful for setting up access control policies and trusted user profiles. Information regarding access control policies has to be decided between the *infrastructure service*, *identity provider* (someone who manages the identities of *infrastructure service users* and authenticates them as and when needed) and sometimes the infrastructure service user. The *identity provider* maintains user profiles in tandem with the *infrastructure service user* himself, and the policy information is then decided upon between the service provider and tenant. Finally, compliance shall ensure that the CloNe architecture is compliant to the regulations specified by different organizations/regions and satisfies the enterprise and/or country audit and compliance reporting requirements.

6. RESULTS AND COMPARISON

Currently, there are multiple (security) architectures/toolkits besides CloNe that provide and strengthen the backbone infrastructure of the cloud delivery models. The most competent include the Open Security Architecture [7], IBM Cloud Computing Architecture [9], and the GRC Stack [8] developed by the Cloud Security Alliance. This section covers a comparison between these architectures, and the security architecture of CloNe described in this publication. The comparison is based on well accepted parameters in the cloud service provisioning ecosystem, and aims to reflect the overall dependability and performance characteristics of the underlying infrastructure. These parameters include *access control, on-demand secure virtual storage provisioning, on-demand virtual compute provisioning, on-demand virtual network provisioning, secure multi-domain communication, secure VM migration between domains, identity management solution, support for hybrid cloud computing, multi-objective security goal translation, on-demand secure network scalability*, and *multi-level security*.

The Open Security Architecture has been released by the OSA, which is a not for profit organization. Its main aim is to release best practices, security patterns, and architectures to help strengthen widely used (security) systems. Their architecture supports both *on-demand secure virtual storage* and *compute provisioning*. However, due to an absence of the virtual network resource provisioning ability in their architecture, their architecture fails to securely provision network resources. Similar to other architectures in the cloud ecosystem, their architecture supports the introduction of *identity management solutions*, although it is not as fine grained, or detailed as CloNe's security architecture. Both *secure multi-domain communication* and *secure VM migration between domains* are omitted from their architecture. Future provisioning infrastructures will need to integrate interactions between different administrative domains securely inside their existing delivery models, especially if the user's service demands can't be fulfilled completely by the currently serving infrastructure provider. The architecture supports *hybrid cloud computing*, which allows the users to pick and choose their final delivery models. Moreover, the architecture supports *multi-level security*, which provides a second (and sometimes third) line of defense. To conclude, the architecture has no support for *multi-objective security goal translation* and *on-demand secure network scalability*.

IBM cloud computing architecture enables the provisioning of virtualized resources to the end user. Moreover, the GRC stack by the Cloud Security Alliance provides an exhaustive toolkit to instrument and assess both private and public clouds against industry established best practices, standards, and critical compliance requirements. Unfortunately, both these

service models contain the same shortcomings as the Open Security Architecture, discussed above. The IBM cloud computing architecture allows the *on-demand virtual network provisioning*, but only within the serving infrastructure provider's administrative domain, thus ruling out chances for multi-domain network resource provisioning. This would render all these three service models ineffective for a multi-domain, multi-level service provisioning model.

In comparison, CloNe, and especially its security architecture supports secure interaction and trust management between different cloud service providers. The CloNe security architecture has a well-defined and multi-grained *access control policy function*, which can accept the access control policies from the different entities participating in the architecture, and deploys the same on the underlying resource set with minimum overhead. The architecture supports all three on-demand secure virtualized service provisioning models, viz. *on-demand secure virtual storage provisioning, on-demand virtual compute provisioning* and *on-demand virtual network provisioning*, thus improving the overall dependability levels of the offered service by involving network guarantees into the SLAs of the provisioned services. As covered earlier, the architecture encourages inter-operator communication and multi-operator service delivery models by supporting both *secure multi-domain communication* and *secure VM migration between domains*. Additionally, the architecture supports *hybrid computing* which enables the user to choose its preferred delivery model based on its KPIs. A customized, *multi-objective security goal translation* process, based on an efficient and accurate goal translation function, further highlights the overall superiority of the architecture over its competitors. The architecture is bolstered by its support for *on-demand secure network scalability*, but is hampered due to the absence of *multi-level security*.

7. CONCLUSION AND FUTURE WORK

In this paper, a new security architecture and security goal translation function for the cloud networking environment have been proposed. This security architecture has been proposed as an integrated extension to the CloNe architecture developed by the SAIL project. The salient features of this security architecture are the provisioning of secure multi-operator resources and on-demand security goal translation.

This paper includes the design and deployment of a security goal translation function, which is integrated into the proposed CloNe security architecture. As a future work, an identity management system and supporting key management techniques will be proposed for further strengthening the security architecture.

8. ACKNOWLEDGEMENTS

The authors would like to express their gratitude to the European Commission for its funding through the "Scalable and Adaptive Internet Solutions", SAIL Project (FP7-ICT-2009-5-257448).

9. REFERENCES

[1] Google Docs. July 2011. [Online]. Available: http://docs.google.com

[2] Google App Engine. July 2011. [Online]. Available: http://code.google.com/appengine/

[3] Amazon Virtual Private Cloud. July 2011. [Online].Available:http://aws.amazon.com/ec2/

[4] SAIL project website. January 2011. [Online]. Available: http://www.sail-project.eu/

[5] SCHOO, P., FUSENIG, V., SOUZA, V., MELO, M., MURRAY, P., DEBAR, H., MEDHIOUB, H., AND ZEGHLACHE, D. In *Challenges for cloud networking security: in Mobile Networks and Management, ser. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*. Springer Berlin Heidelberg, 2010.

[6] KOSLOVSKI, G., PRIMET VICAT-BLANC, P., AND Charão, A. S. VXDL: Virtual Resources and Interconnection Networks Description Language. In *GridNets 2008*, Oct. 2008.

[7] Open Security Architecture Cloud Computing Pattern. February 2011. [Online]. Available: http://www.opensecurityarchitecture.org/cms/library/patternlandscape/251-pattern-cloud-computing

[8] GRC Stack. June 2011. [Online]. Available: https://cloudsecurityalliance.org/research/grc-stack/

[9] SCHMIDT-WESCHE, B., SNITZER, B., BREITER, G., WIDMAYER, G., WHITMORE, J., VILLAREAL, J., BEHRENDT, M., CAPONIGRO, R., CHANG, PAPPE, R., AND Et Al. IBM Cloud Computing & Common Cloud Management Platform Reference Architecture (CC & CCMP RA) 1.0, 2010.

[10] TRIPATHI, A. AND MISHRA, A. Cloud computing security considerations. *Signal Processing, 2011 IEEE International Conference on Communications and Computing (ICSPCC)*, vol., no., pp.1-5, 14-16 Sept. 2011 doi:10.1109/ICSPCC.2011.6061557

[11] VAN CLEEFF, A., PIETERS, W., AND WIERINGA, R.J. Security Implications of Virtualization: A Literature Study. *International Conference on Computational Science and Engineering, 2009. CSE '09*, vol.3, no., pp.353-358, 29-31 Aug. 2009 doi: 10.1109/CSE.2009.267

[12] YAMUNA DEVI, L., ARUNA, P., SUDHA, D. D., AND PRIYA, N. Security in Virtual Machine Live Migration for KVM. *International Conference on Process Automation, Control and Computing (PACC), 2011*, vol., no., pp.1-6, 20-22 July 2011 doi: 10.1109/PACC.2011.5979008

[13] SRIVASTAVA, P., SINGH, S., PINTO, A.A., VERMA, S., CHAURASIYA, V.K., AND GUPTA, R. An architecture based on proactive model for security in cloud computing. *International Conference on Recent Trends in Information Technology (ICRTIT), 2011*, vol., no., pp.661-666, 3-5 June 2011 doi: 10.1109/ICRTIT.2011.5972392

[14] DAYANANDA, M.S. AND Kumar, A. Architecture for Inter-cloud Services Using IPsec VPN. *Second International Conference on Advanced Computing & Communication Technologies (ACCT), 2012*, vol., no., pp.463-467, 7-8 Jan. 2012 doi: 10.1109/ACCT.2012.32