# Scenarios for a Future Internet based on Cross-Layer Functional Composition

Christian Henke[1], Carsten Schmoll[2], Tanja Zseby[2], Abbas Ali Siddiqui[3], Paul Müller[3], Thomas Magedanz[1], Erwin Rathgeb[4]

[1]Technical University Berlin, Straße des 17. Juni 135, 10623 Berlin
{c.henke | tm} @tu-berlin.de

[2] Fraunhofer FOKUS, Kaiserin-Augusta-Allee 31, 10589 Berlin
{tanja.zseby | carsten.schmoll}@fokus.fraunhofer.de

[3] University of Kaiserslautern, Erwin-Schrödinger-Straße, 67663 Kaiserslautern
{siddiqui | pmueller}@informatik.uni-kl.de

[4] University of Duisburg-Essen, Ellernstr. 29, 45326 Essen
erwin.rathgeb@iem.uni-due.de

*Abstract*— **Functional composition is an approach of a future Internet architecture that decomposes the network stack in functional blocks that can be dynamically composed upon connection request. This decomposition resembles the idea of web and telecommunication service in the SOA world. In this paper we will present an architecture to integrate functional blocks from different domains and describe different communication scenarios that demonstrate the advantages and flexibility of such an approach.**

## 1 Introduction

The Service Oriented Architecture (SOA) paradigm has influenced the landscape of application development. Inspired by SOA, companies aligned their internal business processes in software components which are reusable and provide composition of heterogonous resources. Leveraged through the standardization of web services and their description, companies could also offer their capabilities to external customers who can enrich their applications with new functionalities. Within the ongoing process of network convergence to an IP-based Next Generation Network (NGN), Telecommunication providers realized the potential of web services, and started to expose their capabilities like SMS, accounting, authorization for third party service developers. Realizing the potential of offering resources and capabilities to the outside, providers now offer many things as a service – software (SaaS), infrastructure (IaaS), and platforms (PaaS).

At another site of research it has been realized that the current Internet architecture will face several challenges in the future because it has become too complex and ossified, hindering the deployment of new functionalities in the network. Nevertheless new functionalities in the network are needed to keep up with future application demands in terms of security, mobility, privacy, scalability, sustainability and QoS support. Besides this rigidness of the Internet there are also "cyberspace tussles" [12] between users, intellectual property rights holders, government, over-the top service providers (e.g. youtube) and Internet Service Providers, each of which look for potential but also contradictory benefits of the Internet for themselves.

Fueled through future Internet research initiatives in Europe (FIRE, G-LAB), US (GENI, FIND) and ASIA (AKARI) evolving and clean-slate approaches for improving the Internet have been investigated. One of these clean-slate approaches is functional composition, which decomposes the network stack into functional building blocks and reorganizes the functionalities in a composition framework. Functional composition has been and still is in the focus of multiple projects and papers – ANA [6], RBA [5], 4WARD [1], AutoI [7], Self-Net [3], Net-Silo [10], TinyXXL [9], RNA [11], Network Service Architecture [8], and NetServ [2]. Functional composition facilitates the management and integration of new functionalities and also enables the dynamic composition of network functions based on the application requirements. With functional composition the network services are exposed to the service layer.

In the G-Lab DEEP project we investigate means to integrate services from differing domains - telecommunication, web applications and network in a secure cross-layer composition architecture. In this short paper we present a first architecture and use cases for such cross-layer, cross-domain composition.

G-Lab DEEP is funded by the German Federal Ministry of Education and Research (BMBF).

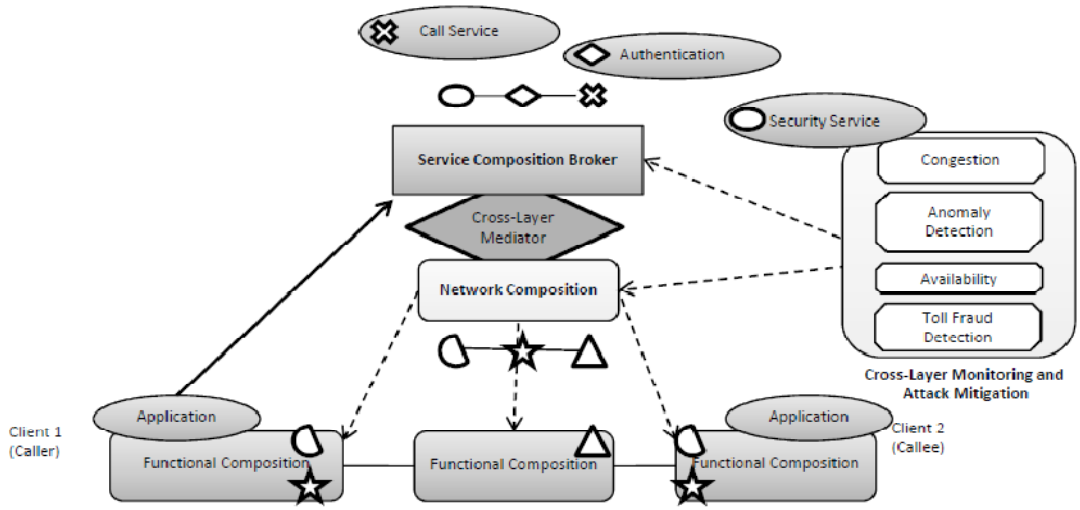


**Figure 1 - Proposed cross-layer composition architecture**

## 2 Proposed Architecture

Our proposed architecture is shown in Figure 1. On service startup the **service composition broker** will get an application request (e.g. make an emergency voice call) and compose application level services so that these fulfill the user request. The service broker is aware of the application specific requirements and will negotiate through the **cross-layer mediator** how the network will be composed. The mediator will resolve whether the network can fulfill the requirements and where functionalities that are available for both levels are executed (e.g. encryption and authentication on network or service layer). The **network composition** will compose functional blocks in the network based on the application requirements and signal the network service workflow to the functional composition framework of the involved nodes that execute the composition for the specific application data flow.
Orthogonal to the proposed cross-layer composition is a cross-layer monitoring service that collects and aggregates relevant service and network level measurements which indicate 1) the network state (e.g. availability, congestion) or 2) security state (anomaly detection, user specific evaluation like black lists). Through the use of this cross-layer monitoring information the service and network composition can adapt to the current situation of the network and applications.

## 3 Scenarios for cross-layer composition

In this section we will present some example scenarios for cross-layer composition. These shall demonstrate the flexibility and the dynamic adoption of the service composition on the service and network level based on the application requirements and the network state. Since we like to focus on applications that can make use of cross-layer information exchange, we have chosen to use Internet telephony applications.

### 3.1 Normal Phone Call in not-loaded network

In the most basic case user A initiates a (VoIP) phone call to user B. In this case the service broker composes a call service and an authentication service on the service layer. The call service will have requirements like unreliable transport, maximum delay < x, acceptable loss rate < y. The network will be composed based on the cross-layer network state and these requirements – an unreliable transport functional block will be chosen and composed with an extra FEC block if the loss in the network is larger than required or if FEC is not available in the network the mediator can also ask the application to use a more loss tolerant voice codec.

### 3.2 Emergency Call in congested network

In this scenario two factors have changed: User A wants to make an urgent call to some emergency centre and the network is congested, either because the emergency affects many people like, e.g. a big fire or there may be a lot of emergency and normal traffic events like at New Years Eve. The service broker composition

framework will adjust the selection of modules then as follows: Authentication is not needed anymore (i.e. to allow emergency call even without user ID or credit), but a locator service is added. The application will now require high QoS in the network and due to the combination of emergency and network congestion the network will add a prioritization module.

### 3.3 Compromised host dialing out for emergency call

In this scenario we assume that the end system from where user A wants to make a voice call (e.g. his own infected PC, or a public PC in an Internet café) shows some anomalous behavior to the network. For example it may generate lots of IP connections and packets due to a worm infection or still pending P2P downloads. For the user who wants to make a phone call this should still be possible. We assume to have monitoring capabilities activated in the network so that anomaly detection and load information per host is available to the cross-layer composition framework.
Knowing that this compromised host generates unwanted and regular traffic the composition framework will now include a deep packet inspection and filtering module in the workflow chain for all traffic from that host to differentiate between normal and malicious traffic. Further the service broker composition will include an authentication service for the emergency call for this compromised host to make sure a legitimate user and not a bot is trying to call the emergency centre.

### 3.4 Emergency Call from host under external attack

In this scenario a user wants to make a voice phone call but his machine or network is subjected to additional unwanted data traffic (e.g. DoS attack). The monitoring components in the network will note this state, and the functional composition framework will be informed and as an effect build a different functional components chain (compared to an unloaded network). Here, similarly to the previous scenario, a filter module is added to the functional composition chain in order to protect the legitimate user that wants to start a phone call. As a general network protection mechanism the functional composition framework may use the information about the network situation also to configure filter elements at the network borders. In that case it would activate a protection for all users of its network at once.

## 4 Conclusion

In this paper we presented a cross-layer composition architecture and communication scenarios that show the flexibility and advantages of such an architecture. Based on cross-layer monitoring information and application requirements, a cross-layer composition architecture can instantiate different functionalities on service and network layer to improve the overall communication effectiveness.

## 5 References


1. 4WARD EU Project. http://www.4ward-project.eu/
2. NetServ Project. http://www.cs.columbia.edu/irt/project/netserv/.
3. SelfNet Project. http://www.ict-selfnet.eu.
4. D. Clark, J. Wroclawski, K. R. Sollins, and R. Braden, Tussle in cyberspace: defining tomorrow's internet," IEEE/ACM Trans. Netw., vol. 13, no. 3, pp.462{475, 2005.
5. Robert Braden, Ted Faber, and Mark Handley. From protocol stack to protocol heap: Role-based architecture. SIGCOMM Comput. Commun. Rev., 33(1), 2003.
6. Autonomic Network Architecture (ANA). http://www.ana-project.org
7. Autonomic Internet. http://ist-autoi.eu/autoi/index.php.
8. Sivakumar Ganapathy and Tilman Wolf. Design of a network service architecture. In Proc. of Sixteenth IEEE International Conference on Computer Communications and Networks (ICCCN), Honolulu, HI, August 2007.
9. Andreas Lachenmann et. al. TinyXXL: Language and runtime support for cross-layer interactions. Proc. of the Third Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON) , 2006.
10. R. Dutta, G.N. Rouskas, I. Baldine, A. Bragg, and D. Stevenson. The Silo architecture for services integration, control, and optimization for the future internet. In Communications, 2007. ICC '07. IEEE International Conference on, pages 1899{1904, June 2007.
11. Venkata Pingali, Joseph D. Touch, Yu-Shun Wang. A recursive network architecture. 2006.
12. Clark, D. D., Wroclawski, J., Sollins, K. R., and Braden, R. 2005. Tussle in cyberspace: defining tomorrow's internet. *IEEE/ACM Trans. Netw.* 13, 3 (Jun. 2005), 462-475.