Illustrating a Publish-Subscribe Internet Architecture

TR10-002

Document Properties:

Title of Contract	Publish-Subscribe Internet Routing Paradigm
Acronym	PSIRP
Contract Number	FP7-INFSO-IST 216173
Start date of the project	1.1.2008
Duration	30 months, until 30.6.2010
Document Title:	Illustrating a Publish-Subscribe Internet Architecture
Date of preparation	01.06.2010
Author(s)	Nikos Fotiou (AUEB), Dirk Trossen (CAM), George C. Polyzos (AUEB)
Responsible of the deliverable	Nikos Fotiou Phone: +30 210 8203693 Fax: +30 210 8203686 Email: fotiou@aueb.gr
Target Dissemination Level:	PU
Status of the Document:	Final
Version	1.00
Document location	http://www.psirp.org
Project web site	http://www.psirp.org

Table of Contents

This document has been produced in the context of the PSIRP Project. The PSIRP Project is part of the European Community's Seventh Framework Program for research and is as such funded by the European Commission.

Illustrating a Publish-Subscribe Internet Architecture

Nikos Fotiou[1], Dirk Trossen[2], and George C. Polyzos[1]

[1]Athens University of Economics and Business, Athens, Greece,
{fotiou,polyzos}@aueb.gr
[2]Computer Laboratory, University of Cambridge,
dirk.trossen@cl.cam.ac.uk

Abstract. The Publish-Subscribe Internet Routing Paradigm (PSIRP) project aims at developing and evaluating an information-centric architecture for the future Internet. The ambition is to provide a new form of internetworking which will offer the desired functionality, flexibility, and performance, but will also support availability, security, and mobility, as well as innovative applications and new market opportunities. This paper illustrates the high level architecture developed in the PSIRP project, revealing its principles, core components, and basic operations through example usage scenarios. While the focus of this paper is specifically on the operations within the architecture, the revelation of the workings through our use cases can also be considered relevant more generally for publish-subscribe architectures.

Keywords: Future Internet, Clean Slate, Networking usage scenarios

1 Introduction

The current Internet architecture has remained relatively unchanged since its inception. The Internet was initially designed in a way that somewhat resembles a telephone network, where uniquely addressed endpoints trust each other and exchange data through an internetworking infrastructure. However, nowadays this design does not cope with current networking trends, neither with applications needs. Viruses and worms have led to a state where two arbitrary network nodes do not trust each other anymore. End-to-end communication does not seem to be the prevailing paradigm as data requests are likely to be served by an intermediate element–such as a content delivery network, or a proxy server–and popular applications (e.g., p2p file sharing applications) focus on the information itself rather than on the endpoints sotring the information or the location.

Furthermore, an imbalance of powers exists in the current Internet, where the network is designed to forward data from senders to receivers, whether the receivers want to receive the data or not, leading to problems such as denial of service attacks and spam email. Various mechanisms such as network address translation (NAT) and spam filters have been deployed in order to restore these issues; however they do not solve the problem completely. Moreover challenges

related to security, mobility, scalability, quality of service, and economics have increased the need for a clean slate approach to a new Internet architecture [8].

We underline the critical importance of information with the following realistic scenario of future Internet usage. We focus on the challenges that arise, manifesting this way a shift towards an information oriented future Internet.

During the summer Olympic games, various media providers offer live streaming video from the stadium. Wireless cameras deployed in the field, cameras on blimps and dozens of cameramen are used to offer a breathtaking experience to the viewers worldwide. Internet users are able to enjoy the streams in a variety of devices ranging from high definition TV to 3G mobile phones. Moreover users are able to choose among various coverage angles, as well as select the language of narration etc. Media providers also offer exclusive videos from the preparation room as well as interviews with the athletes and their coaches. All these videos are provided upon subscription. Meanwhile in the stadium, thousands of spectators shoot pictures, which they instantly upload to their photo blogs, record videos that they share in popular video sharing Web sites and make video calls in order to show to their friends famous athletes.

Various challenges can be identified in this scenario. Information needs to be accessed seamlessly by a variety of devices which can be fixed or mobile. Tussles arise as spectators and media providers compete for the network resources inside the stadium. Tussle may also arise between athlete sponsors, that may want to control the media regarding their athletes, and media providers, that want to reveal as much information as possible. The dissemination of information– especially concerning the exclusive videos and the video calls–needs to be limited to only the eligible users, assuring the respect of digital rights as well as the privacy of the users. Finally it needs to assure that end-users will receive the information which they are really interested in, excluding from the information delivery path malicious users, such as spammers.

We believe that an information-centric communication paradigm would address these challenges in a more successful way than the current Internet. Such a paradigm is the publish/subscribe (pub/sub). Pub/sub is an information-centric paradigm that shifts the power away from the data sender, i.e., data consumers express their interest in specific pieces of information explicitly, which are forwarded to them by the network when they become available. As a result, information is propagated only to those nodes which really want it.

The Publish/Subscribe Internet Routing Paradigm (PSIRP) project [18] is a multi-organization FP7 EU funded research effort aiming at creating a clean-slate architecture for the future Internet based on the pub/sub communication paradigm, taking nothing for granted. The contribution of this paper is to present the main architectural concepts that are underpinning the PSIRP approach. In particular, we outline the architectural concept equivalent to layering in today's networking, the so-called *bubble* concept. This concept is directly based on (a) the network functions being identified as central in PSIRP and (b) the information structures being defined within PSIRP. We expect this concept to have a direct impact on the way network nodes will be implemented as well as on how they will

function in a networked environment. This paper presents this concept as well as the general architectural thrust of PSIRP at conceptual as well as qualitative level.

We have implemented various parts of the presented architecture and installed early testbeds for upcoming performance results of meaningful scale. Furthermore, early performance results have been obtained for parts of the architecture with simulation and emulation. However, we left out the presentation of these results due to the architectural thrust of the paper, but we do recognize the need to extend the performance evaluation of various aspects of the architecture in order to fully appreciate and understand its benefits and in particular of the proposed bubbles concept.

The remainder of this paper is organized as follows. Section 2 presents related work in the area. Section 3 gives an overview of the PSIRP architecture, presenting its core elements and its basic operations as well as how information is organized and provided. In Section 4 PSIRP usage scenarios are given, illustrating at considerable level of detail the supported functionality and operations. Finally, Section 5 presents applications considerations for the PSIRP architecture and Section 6 presents our conclusions and plans for future work.

2 Related work

Publish/subscribe overlay systems have been widely studied, especially in cases of event notification architectures. Siena [6] and Hermes [17] are two notable examples of large scale pub/sub systems with intra-domain rendezvous functionality. Siena was particularly successful in demonstrating the applicability and effectiveness of the pub/sub paradigm in multi-domain environments, whereas Hermes provides middleware that accelerates the development of applications that operate in these environments. PSIRP extends the service model of these systems by applying the pub/sub paradigm at all levels of its architecture, targeting at the same time the provision of the necessary tools and APIs that will allow applications to harvest the full potential of this paradigm in a seamless way.

Various research efforts–such as the Internet Indirection Infrastructure (i3) [19] and the Host Identity Protocol (HIP) [2]–advocate indirection as the solution to the problems that point-to-point communication poses to mobility, multicast and multihoming. i3 implements an IP overlay network that replaces the point-to-point communication model with a rendezvous-based paradigm where senders send packets to a specific rendezvous-point while receivers issue triggers on specific packet identifiers. HIP introduces a new layer in the internetwork stack between the IP layer and the transport layer. This new layer decouples host identity from location identity. PSIRP uses similar concepts through the rendezvous and topology formation processes.

The problem of routing based on flat information identifiers rather than on hierarchical location-based identifiers has also been studied in the Data-Oriented Network Architecture (DONA) [15] and in the Routing on Flat Labels

(ROFL) [4] projects. DONA proposes a new identification scheme based on flat, self-certifying identifications as a replacement to the DNS naming resolution scheme that enables 'finding' and 'fetching' content. ROFL investigates the possibility of having an internetworking architecture solely based on flat identifiers, using DHTs and hierarchical DHTs. The evaluation results of ROFL show that this approach is feasible and it can incorporate all the internetworking structures that exist in the current Internet. PSIRP borrows the information identification concept of DONA, but chooses a separate inter-domain architecture with slow and fast paths. Moreover PSIRP extends ROLF towards flat identifiers within hierarchical *scopes*, which are expected to offer faster information scoping and dissemination.

PSIRP is not the only current research project aiming at redesigning the Internet with an information-centric or content-centric perspective. CCNx [7] is a research effort that proposes routing based on hierarchical naming. In CCNx consumers ask for content by broadcasting 'Interest' packets that contain the name of the content in request. Any 'Data' packet whose content name is a suffix of the name in the 'Interest' packet is conspired that it satisfies this interest. PSIRP, on the other hand, introduces flat label identifiers organized into scopes, allowing for a variety of naming approaches to be layered on top of the internetworking architecture. Moreover although overlaying PSIRP over the current Internet is possible, it is the declared goal of the PSIRP project to investigate a native solution that will replace current internetworking technology. This leads to a focus on inter-domain functions, which is not found in CCNx. 4WARD [1], another FP7 EU funded ongoing research project, also advocates an information-centric Internet which will enable network diversity, allowing various types of networks to co-exist and cooperate in a smooth and cost-efficient manner. It envisions an Internet where networks will be self-manageable and network paths will be an active networking component that it will be able to affect transport services. 4WARD borrows concepts from DONA in terms of labeling and intends to shed light on business aspects, similarly to the socio-economic work in PSIRP.

Other ongoing research investigates PSIRP performance and effectiveness regarding forwarding, caching, and mobility. PSIRP's forwarding is based on the formation of a bloom filter based data structure–called *zFilter*–that includes the identifiers of the links that a packet needs to traverse in order to reach its destinations. Jokela et al. implemented zFilter-based forwarding in NetFPGAs using temporary link identifiers achieving secure forwarding at **line speed** [9]. Katsaros et al. investigated content delivery in a PSIRP-like environment, where multicast is the primary delivery method and pub/sub based caches are used in edge routers; they showed that the approach has the potential to achieve substantial reduction in interdomain traffic and download time [13]. Finally it was demonstrated that even an overlay environment (which suffers from the inefficiency of stretch) abiding to PSIRP principles, i.e, pub/sub and multicast can be more effective than mobile IPv6 [11], [12].

3 A Publish-Subscribe Inter-Domain Architecture

The PSIRP architecture [18] is based on the premise of interconnecting information rather than endpoints, i.e., endpoint topologies are dynamically created based on the expressed availability (publication) and need (subscription) of information. With this, we envision a robust and scalable architecture where mobility will be the norm and data morphing will allow users to access information anywhere through any medium. Performance and efficiency will be achieved with the use of innovative multicasting and caching techniques, and security will be a native component of the architecture. Specific consideration is given to the ability to place functions in trustworthy points within our architecture, taking into account the Trust-to-Trust principle [3].

3.1 Information Concepts

Information is the core element in the PSIRP architecture; everything is information and information is everything [20]. Information is organized in a hierarchical way, so small 'meaningless' pieces of data, which can be arbitrary chunks of data, are combined into large complex information items–such as files, documents pictures and videos. An information item may be used as a reference to other items, providing information such as data size, information owner, permissions, composition elements. These items are referred to as the *metadata* and they can be used to group information based on some specific semantics. *Scoping mechanisms are used to limit the reachability* of the information to the parties having access to that particular *scope* [10]. Within a scope the architecture is neutral with regard to the semantics and structure of the data, although governance rules regarding the available information may be defined. Scopes can be regarded as the equivalent of IP topologies, i.e., as IP topology mechanisms allow the creation of a topological inter-network, scoping mechanisms allow for building information networks. Scopes have a hierarchical structure where parent-children and sibling relationships exist. In PSIRP, there can exist physical scopes, e.g., a University of X network, and logical scopes, e.g., a social network (hierarchy). Every piece of information is attached at least to one specific scope, which is represented by the scope identifier that publishers set when they publish information. Several mechanisms are used to control the scope of a piece of information. These mechanisms include access control, DRM, user authentication, and many others. Information items may be part of multiple scopes. For example, an information item (such as an image) may belong to a University and at the same time it may belong to a specific family scope.

As for identification, every piece of information is identified with a (statistically) unique label. This label is used in order for subscriber interests to be matched with published information. The function of matching subscribers interests with published information is known as the rendezvous function and for this reason this label is referred to as the *rendezvous identifier* (RId). A subclass of rendezvous identifiers is the *scope identifier* (SId). SId denotes the specific scope within which the information is reachable. Rids and Sids are independent from

the endpoints producing and consuming the associated information items. Flat and endpoint independent labels seem to be a natural choice for information oriented architectures as they clearly separate location from identity allowing for properly incorporating mobility, multicasting, and multihoming into the architecture. as well as a more comprehensive notion of identity [4]. The PSIRP architecture includes publishers, subscribers and rendezvous points. Publishers are information providers that feed information elements into the pub/sub network by virtue of publications. Subscribers are consumers that explicitly express their interest in a specific publication by issuing subscription messages. These messages contain the criteria that a publication should fulfill in order to be forwarded to a subscriber. Publications may have different versions, and whenever a new version of a publication is created, all subscribers are being informed.

3.2 Bubbles: A Layer Concept for an Information-Centric World

As outlined above, information in the PSIRP architecture is organized using scopes with an identification structure that allows for hierarchically organizing information. In the following, we outline the so-called *bubble* concept which is akin to the layering model in our current Internet model, i.e., it defines the concepts and methods which are used to provide information items within a set of scopes within particular usage scenarios. In other words, bubbles complement the scope concept with a notion of provisioning information within a particular scope.

Each bubble implements scope-specific *Rendezvous*, *Topology* and *Forwarding* (RTF) functions to enable the provisioning of the information within the scope. The Rendezvous function is responsible for matching subscribers' interests with publications. The node where the matching of a publisher's content with a subscriber's interest takes place is referred to as the *rendezvous point* (RP). These elements initiate routing, forwarding, and distribution decisions, eventually leading to the delivery of the content from publishers to subscribers. Hence, RPs ensure a balance of power between sender and receiver of information, i.e., no information is delivered without explicit signaling of availability (publish) and interest (subscribe). Publication and subscription operations are decoupled in time and space as they do not have to be synchronized; they do not block each other and publishers do not have to be aware of the subscribers–and vice versa.

Whenever publishers wish to issue a new publication they have to use two identifiers: RId and SId. A publication's RId can be derived by an application specific function. A publication's SId should denote to which extent the publisher wishes the publication to be available. Prior to publishing an information element, publishers have to locate the nodes that are responsible for managing the desired scope. One of these nodes will be the RP for the publication. The nodes that are responsible for managing a scope and are eligible for becoming a RP are referred to as the *rendezvous nodes*. What is actually published to the RP is the publication's metadata, which contain information specific to the actual publication; this can be for instance the author of the publication, its size and

perhaps a small description of it. In order for a subscriber to access a publication she must be aware of its RId and SId. She expresses her interest about a specific publication by issuing a subscription message towards the publication's RP, identified by the SId. The RP is responsible for matching publications with subscriptions and for initializing the forwarding of a publication from the publisher towards the subscriber.

The topology function monitors the network topology and detects changes using various techniques, depending on the bubble it is implemented in. Moreover, the topology function is responsible for creating information delivery paths at different levels of the inter-domain system.

Finally, the forwarding function implements information forwarding throughout the delivery paths using MPLS-like label-based forwarding mechanisms. The forwarding is implemented through assigning a stack of path specific identifiers to the publication, named *forwarding identifiers* (FId). These FIds are used by intermediary forwarding nodes to forward the publication to the desired destination. Publications may be cached along the path. In case that more than one subscriber subscribe to a specific RId, a multicast tree is created in order to deliver the publication.

The particular implementation fro RTF functions depends on the specific context in which the bubble has been created. For instance, methods applying locality could be utilized for the Access Point (AP) bubble, having simple forms of rendezvous, largely driven by the local attachment and by virtue of the local link discovery (i.e., literally the L2 discovery of the channel). The topology function running on the Operating System (OS) bubble is responsible for maintaining connectivity and for predefining (forwarding) labels that will be used by the forwarding function in order to forward information items through the various interfaces. Larger bubbles, such as the global one, need to solve more complex 'matching,' but also topology problems, which leads to more complex solutions in these areas.

As for the organization of bubbles, they can be included in each other or can just touch each other (implementing a sequence of information traversals). Information within each bubble traverses through the bubble from points on its membrane–the traversal implemented through the proper RTF functions. The points on the membrane constitute publishers and/or subscribers of information within the enclosed but also the enclosing or touching bubble.

The bubble concept bears similarities with the Recursive Network Architecture (RNA) [21], as well as with Netlets [22]. RNA uses a single, tunable protocol for different layers of the protocol stack, reusing basic protocol operations across different protocol layers whilst Netlets, encapsulate protocol stacks, with the principle of hiding protocol details but yet providing a number of properties via its interfaces. Our model of bubbles differs from the model of recursive layering because a particular layering is not assumed. Bubbles describe the conveyance of information within a particular region in which this bubble makes sense. Information at the edge of the bubble can be conveyed further (within another bubble) at the same 'layer' but using different methods for RTF. In other words,

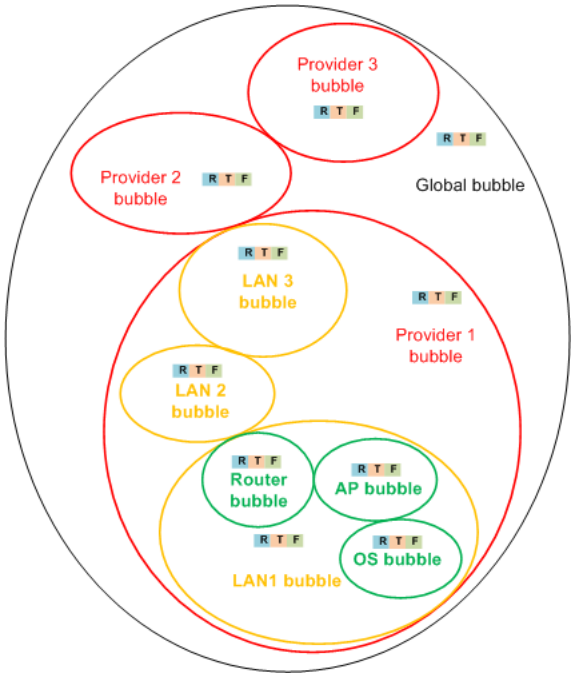

Fig. 1. An instance of the bubble model

bubbles can include other bubbles (akin to recursive layering), but also touch each other, i.e, similar layers with different functionality for RTF. Moreover, in contrast to Netlets, PSIRP bubbles predefine the functionality that all bubbles shall offer, i.e, the RTF functions.

Figure 1 shows a particular instance of the `bubble` model, demonstrating the inclusive and touching nature of bubbles. An information item from the shonw OS in LAN1 can traverse the following bubbles in order to be delivered/accessed by a user in the Provider 3 bubble: OS-AP-Router-(LAN1)-LAN2-LAN3-(Provider1)-Provider2-Provider3. Additional steps may be required within Provider2–and at different points–but they are abstracted out here. In particular, points where internal bubbles touch external bubbles at the same point as that of a bubble containing them, the traversal to the exterior bubble need not to be explicit. This case is shown with the parenthesis for LAN1 and Provider1 in this example.

3.3 Mobility

User mobility is regarded as a two-dimensional problem. The first dimension of the problem concerns the scale of the mobility which can be local or global. The second dimension of the problem reflects how the mobility is handled by the architecture. Mobility can be handled either in a static or in a dynamic way. When

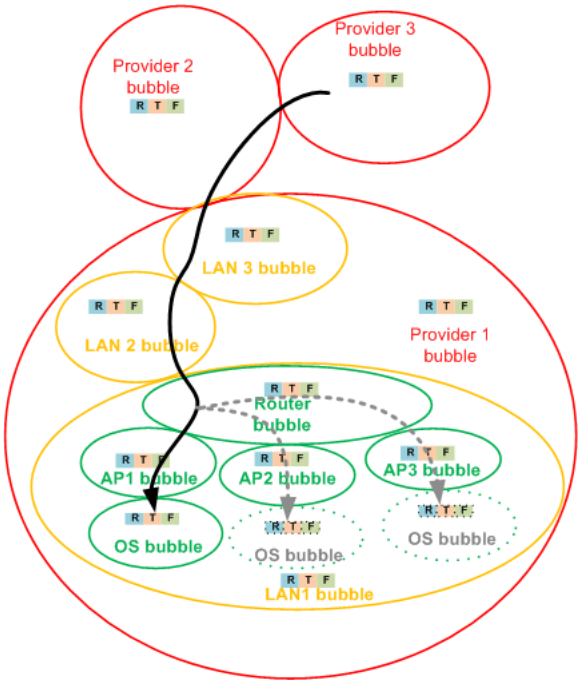

Fig. 2. Static/Local mobility handled through isolation

static mobility is involved, bubbles simply 'fly around' within the perimeter of their containing bubble detaching and attaching to other bubbles. In the case of dynamic mobility, temporary bubbles are created that allow for information transition between different environments–which can be different wireless technologies or even different administrative domains. Table 1 gives an overview of mobility categorization in PSIRP. A user moving around with his laptop inside a campus covered by a (multi-AP) WLAN is an example of static-local mobility, whereas a Vehicular Network which involves sensing from various sensors deployed along the road directly or receiving the same information indirectly through communication with other cars is a typical example of dynamic/local mobility. On the other hand handover of a roaming, user to another provider, i.e., a new administrative domain, similar to today's roaming is a static-global case of mobility, while vertical handover without roaming is a case of dynamic-global mobility.[1]

Figure 2 demonstrates a static/local mobility scenario. A mobile terminal (MT) is moving around within the premises of *LAN1*. Its initial AP is *AP1*, and it moves to *AP3* through *AP2*. This MT is receiving an information flow from *Provider 3* through *Provider 2* and *LANs 3 and 2* of *Provider 1*. When the

[1] In this case the user is a client of both providers separately and no roaming is initiated

MT changes AP the information delivery tree is simply re-built by the *Router bubble* of *LAN1*, which forwards information to the appropriate AP. In order to achieve information delivery tree reconstruction, the MT needs to issue a new subscription message from the new location. This subscription message will ultimately trigger the *Topology* and *Forwarding* functions of the *Router bubble* as well as that of the new *AP bubble*, leading to information flow redirection. Outside *LAN1's* bubble the mobility of the MT is transparent as it is 'isolated' inside LAN1's bubble, causing no change to any other bubble in the network. A dynamic-global mobility scenario is depicted in Figure 3. This scenario in-

	Local	**Global**
Static	Handover in managed WLAN environment	Nets between cars
Dynamic	Handover with roaming	Handover without roaming

Table 1. Mobility categorization in PSIRP through examples; static refers to no change in bubble; dynamic refers to the case where a new bubble is created

volves a MT that has a 3G and a WLAN interface. Initially the MT receives an information flow from *Provider 3* through its 3G interface. It then detects an available WLAN and it decides to perform a *vertical handover*. Although this scenario involves little–or no–physical MT movement, it may cause global information-flow shift, as the 3G operator and WLAN provider can be different. The new location of the MT needs to be informed about the upcoming arrival and state needs to be transfered from the *CELL2 bubble* of *Provider V* to the *LAN1 bubble* of *Provider 1*. In order for this state transfer to occurre the MT needs to inform CELL2 about its movement. CELL2 in return creates a dynamic bubble between its bubble and the *AP1 bubble*. This 'dynamically created bubble' enables state transfer from the *3G bubble* to the AP to which the MT is going to be attached. The bubble of the AP in return is going to perform all the necessary actions in order to redirect the MT's information-flow to the new location and when it is ready, it will inform the CELL2, using the dynamic bubble. The information flow redirection requires the activation of the *RTF* functions of all bubbles between the *AP1 bubble* and the *Provider1 bubble* as new subscription messages need to be sent from the new location. The dynamically created bubble not only allows two different media to communicate, but it enables information transfer between two differently managed environments, which involves trust relationships and business agreements. PSIRP security mechanisms will handle the security issues that will be possibly raised by the bubble creation.

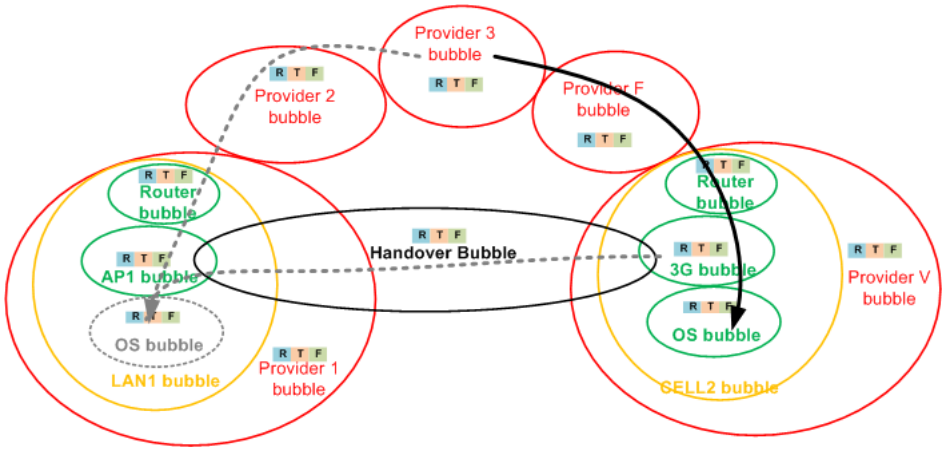

Fig. 3. Dynamic/Global mobility, involving vertical handover, handled through temporary bubble creation

3.4 Security

The Trust-to-Trust principle [3] is one of the core principles of PSIRP. Rendezvous, topology management and forwarding functions take place in trustworthy points. The pub/sub nature of PSIRP, its decentralized nature of address space management, as well as its rendezvous-driven operation offers some significant security advantages. The network balance is shifted towards the information receiver, relieving the information provider from the burden of constant information requests. This balance shift is expected to attenuate DDoS attacks, that plague the current Internet as well as to constrain unwanted traffic such as spam since no information[2] is delivered without explicit signaling of availability (publish) and interest (subscribe). Moreover some level anonymity is inherited from the pub/sub paradigm as interacting parties do not need to know each other or even come to direct contact. The publication and subscription operations are also decoupled in location and time, since they do not have to be synchronized, and, thus receiver and sender unlinkability is also achieved. Furthermore, by using multicast as the preferred delivery method, it is expected to have better resource utilization and information availability.

PSIRP considers security even at the packet level. Packet Level Authentication (PLA) [5] is a novel technique, for protecting the networking architecture. PLA is based on the assumption that per packet public key cryptographic operations are possible at wire speed in high speed networks due to new cryptographic algorithms and advances in semiconductor technology. Moreover PSIRP's forwarding mechanism is based on dynamically generated forwarding identifiers [9] making it almost impossible for an attacker to forward malicious data packets.

[2] except to RPs, but those are few, key points of the architecture that can be adequately protected

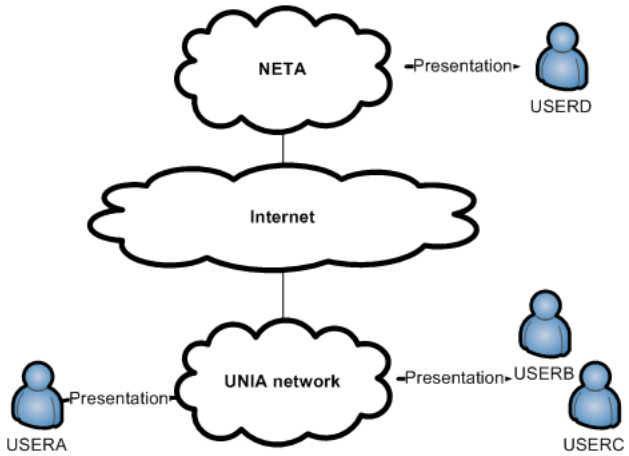

Fig. 4. A PSIRP usage scenario setup

When it comes to the network level, attachment procedure in PSIRP [14] assures the proper user authentication, protecting both users from improper configuration, as well as network from DoS attacks. Finally at the application level, it has been found that, although networking protocols will be redisgned, current cryptographic protocol analysis can be applied to a certain extent, with only minor modifications mostly in notation [16].

4 PSIRP usage scenario

After we outlined the general architecture in the previous section, we now illustrate the workings of the architecture in some typical usage scenarios. With this, we attempt to further clarify the relations and operations within the architecture, but also shed some light onto the development of typical applications.

4.1 Scenario setup

A user, USERA, works in a university UNIA, in the department DEPTA. USERA has prepared a presentation and he wants to make it available to his colleagues in his department. USERA has three colleagues; USERB, USERC and USERD. USERB and USERC want to access USERA's presentation through UNIA's local network and USERD wants to access it from his home network, NETA. Figure 4 depicts this scenario setup.

4.2 Publication

The UNIA network consists of four rendezvous nodes RN001, RN002, RN003 and RN004. DEPTA has its own scope with SId 00A1. Scope 00A1 is managed

by RN003 and RN004, and every potential publication to scope 00A1 will be forwarded to either RN003 or RN004. Scope 00A1 implements the following access policy: Only members of DEPTA are allowed to publish information and to subscribe for publications in that specific scope. USERA wants his presentation to be accessed only by members of DEPTA, so he decides to publish it in scope 00A1. A publication is created and an application specific function generates an RId identification number for this publication, which is AA12. USERA's application running on SERVER01–which is located inside the UNIA network–issues a publish message with SId 00A1 and RId AA12, and this message is forwarded to RN003. The publish message contains, along with the identifiers, USERA's presentation metadata. The publication operation ends with RN003 updating its internal RIds database by adding AA12 and becoming the rendezvous point for this RId. The publication operation is depicted in Figure 5. The successful completion of the publication operation presumes that USERA has properly authenticated himself to the scope 00A1.

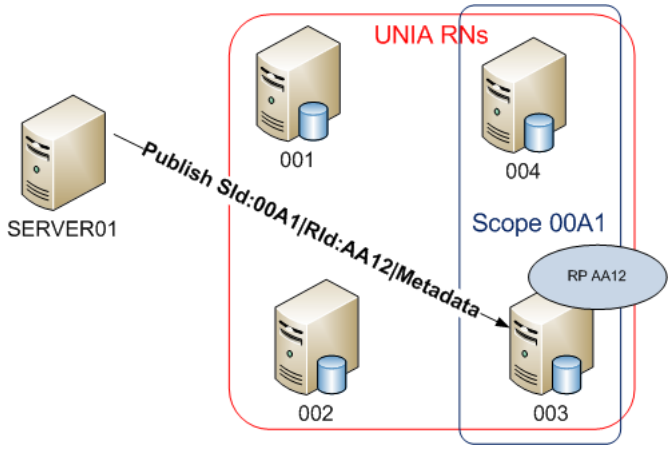

Fig. 5. PSIRP's publication operation

4.3 Subscription from the Internal Network

USERB learns about USERA's publication through some form of discovery, e.g., an internal listing of presentation or some (new form) of search engine. In order for USERB to subscribe to USERA's publication, he has to authenticate himself to scope 00A1. He achieves that by logging in to SERVER02 with credentials that allow him to subscribe for publications in scope 00A1. USERB's application running on SERVER02 issues a subscribe message towards scope 00A1, denoting that he is interested in the publication with RId AA12. RN003 receives USERB's subscription message and it initiates the creation of a forwarding path between publication's AA12 location, i.e., SERVER01, and SERVER02.

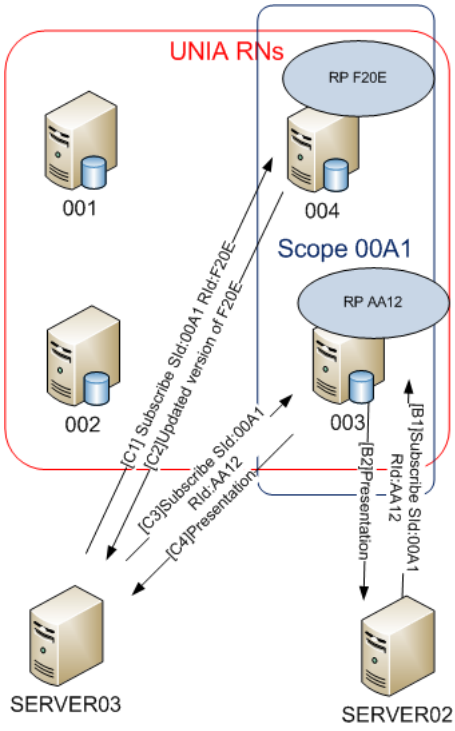

Fig. 6. PSIRP's subscription operation. The numbers inside brackets denote originator and the sequence of each message, e.g.,[1] is the first message that USERD sends.

In scope 00A1, a publication with Rid F20E exists, that contains the RIds of every presentation available in this scope. This publication is provided by a presentation announcement service. USERC is interested in every presentation in scope 00A1 so she has subscribed for publication F20E. USERC uses SERVER03 with credentials that allow her to subscribe for publications in scope 00A1. When USERA publishes his presentation, the presentation announcement service creates a new version of F20E, which contains USERA's publication RId. USERC receives the new version of F20E and she becomes aware of USERA's publication so, she subscribes to it. At this point a procedure similar to the one described for USERB's subscription is followed in order for the publication to be forwarded to SERVER03. Figure 6. depicts PSIRP's subscription operation.

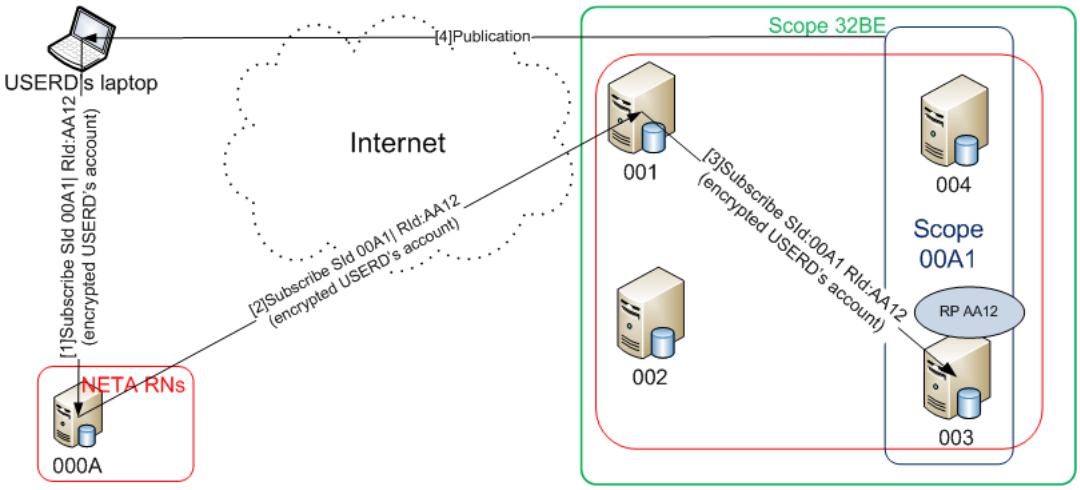

Fig. 7. PSIRP's subscription operation from an external network. The numbers inside brackets denote the sequence of message.

4.4 Subscription from an External Network

USERD wants to access USERA's presentation from his home network NETA. He creates a subscription message with destination SId being 00A1 and he encrypts his authentication data in the message payload. The subscription message is forwarded to NETA's default RP which is 000A. 000A forwards the subscription message to RN001. RN001 determines that the destination scope is 00A1 and it forwards the subscription message towards this scope. RN003, which is the RP for this specific RId, finally receives the subscription message. However, instead of creating a forwarding path towards USERD it initially checks for the validity of USERD's credentials, sent encrypted along with the subscription message. If USERD credentials are valid, a forwarding path from SERVER03 to

USERD is created and the desired publication is forwarded using this path. The overall operation is depicted in Figure 7.

4.5 Forwarding

In PSIRP forwarding solely depends on information identifiers, i.e. RIds and SIds, thus MPLS-like label switching protocols are used. In order for an information item to be forwarded a stack of forwarding identifiers is determined for the links along the forwarding path. Each forwarding node along the path maintains a forwarding table which contains the incoming forwarding identifier and its corresponding outgoing interface and identifier. If we consider again the subscription from the internal network case, upon the successful completion of the subscription operation, a forwarding path is created from SERVER01 towards SERVER02 and SERVER03. The forwarding path can be seen in Figure 8. The box next to the dashed lines represents a data packet with its FId. The table below each forwarding node shows its forwarding table. The forwarding table contains the incoming Fid, the outgoing FId and the outgoing interface. SERVER01 sends a packet with FId 12. The first forwarding node checks its forwarding table and finds that it has to forward this packet to interface 2 with FId 14. In a similar way the second forwarding node duplicates and forwards the packet to its interface 2 with FId 19 and to its interface 3 with FId 20. This way the publication will reach USERB.

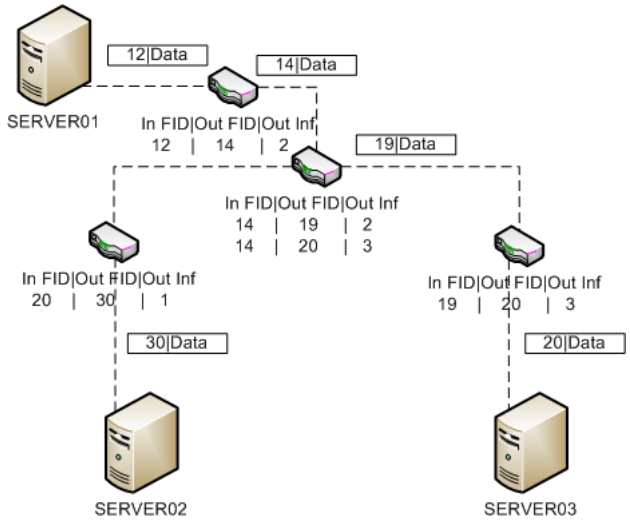

Fig. 8. PSIRP forwarding

4.6 RTF function execution within bubbles

During the whole operation, sequences of RTF functions are executed in a recursive way. In order to demonstrate this, we slightly modify the publication use case. We consider that the publishing application of USERA is running in bubble LAN1 of Figure 1 and the publication's RP is located in bubble Provider 3. The application needs to get to the global bubble Rendezvous (R) function and in order to do so it has to go through some of the relevant bubbles in-between. Initially the application makes the publication available to the OS R function. The OS R function determines that the OS bubble is not the destination bubble and asks the OS Topology (T) function to create a forwarding path towards the Provider 3 bubble. The OS T function determines, e.g., using static routing tables, that the publication message should be forwarded through the wireless interface and it informs the R function about the appropriate forwarding label that should be used. The R function instructs the application to forward the publication using the forwarding label of the wireless interfaces. The Forwarding (F) function makes all the necessary steps in order for the publication to reach the AP Bubble. Using a similar sequence of RTF executions the publication message will reach the Router bubble. The Router bubble is responsible for advancing the publication message to a higher layer bubble, thus, it has to make use of the more advanced RTF functions of the LAN 1 bubble. The R function of the LAN 1 bubble, identifies that the RP of the publication message is located in the Provider 3 bubble and it asks the T function for a path towards that bubble. The T function, that implements an intra-domain routing protocol, recognizes that the publication message needs to be forwarded through the LAN 3 to another network and it creates a forwarding path from LAN 1 to LAN 3 through LAN 2. Edge routers in these LANs update their routing tables with the new path and the F functions running in these routers make all the necessary forwarding decisions in order for the publication message to reach the LAN 3 edge router that interconnects the Provider 1 bubble with the Provider 2 bubble. At this point the publication message will be advanced to an even higher level bubble, so more complex RTF functions need to be executed and these are the RTF functions of the Provider's 1 bubble. The R function identifies the exact location of the Provider 3 RP and asks the T function, which implements an inter-domain routing protocol, to create a forwarding path towards this location. The T function creates a path from the Provider 1 bubble towards Provider 3 bubble, through the Provider 3 bubble.

5 An application developer's view

PSIRP's use cases are meant to also shed some light on application perspectives towards this architecture. The identifiers used in PSIRP are not application oriented and, their structure although very useful for the overall system effectiveness, it may pose a burden on application developers, as e.g., they should assure that the (publication) identifiers their applications generate are unique

within a scope. However, even at these early steps of PSIRP, an in-kernel module that generates publication identifiers is provided. Scopes in PSIRP are used for describing the extent of dissemination of information, and not the structure underneath, therefore they are a more general notion than what could be considered their first order approximation in the current Internet, i.e., networks. Scopes are managed by lower level mechanisms and their operation is completely transparent to applications. Even more transparent will be the operation of the forwarding function; forwarding identifiers will be completely hidden from applications. Applications will be able to smoothly operate without being aware of user mobility, information multicasting and multihoming, or any other specific Internet access mechanism. On the other hand, applications will be offered the possibility to define policies and requirements that will affect routing.

Another fundamental concept that may trouble developers is the use of trust with every transaction. Preserving trust is a core principle of the PSIRP architecture and all functions should take place at trusted points in the internetwork. PSIRP will provide to applications all the means for determining if another node in the network behaves in a trustful way, however applications are also expected to implement their own, application specific, trust evaluation algorithms, since at the application layer there are various trust expectations (and high-level information), which cannot be predicted by the lower layers of the architecture.

6 Conclusions, Ongoing and Future Work

PSIRP set out with an ambitious goal to define a new and information-centric inter-domain architecture. Progress toward this goal has been made with first results available and work still progressing at a good pace.

Many lessons have been learned from this effort. Some of these lessons relate to the design of such architecture. Not only did PSIRP apply a method of combining top-down (requirements-driven) as well as bottom-up (learning from code) approaches, but it also included socio-economic viewpoints very early in the design phase of the architecture. This has led to a deep understanding of what the core proposition of the architecture ought to be, namely its foundations in an identifier configuration that provides structures akin to complex application-layer information structures while being simple enough to scale to the desired size of the Future Internet.

The PSIRP architecture effectively handles information provisioning using the concept of bubbles. Bubbles not only enable layering, but they also provide a mechanism for seamlessly integrating new functionality within the PSIRP architecture. By defining *what* should be implemented within a bubble–and not *how*, in conjunction with a common inter-bubble communication interface, the PSIRP architecture is expected to be extensible and flexible. The bubbles concept will allow the easy introduction of new technologies and protocols as well as for large scale architectural modifications, leveraging this way the architecture's composability.

Apart from the obvious technological advantages it offers, the concept of bubbles is also expected to have a significant socio-economic impact. PSIRP envisions a 'bubble-market', where providers will be able to sell a great variety of services ranging from access technologies to elaborate information retrieval and manipulation applications. Moreover the bubble concept is envisioned to play a key role for the easy integration and connection to the network of even more and diverse devices (as information appliances).

Our socio-economic work furthermore sheds light on the various tussles that are likely to occur in such architecture, explaining some of the design choices around topology formation and rendezvous from the angle of future markets in which such architecture would live in. And last but not least, we consider a solid thinking on potential migration crucial for the success of PSIRP. Work on overlay implementations of the architecture have been part of the efforts from the start, realizing that a 'native' deployment of the architecture would be nothing short of unrealistic. That gives us the required options to gradually deploy the architecture presented here, while still remaining 'clean-slate' in its design through questioning the necessary fundamentals of today's Internet.

However, the most important lesson learned throughout the efforts of PSIRP is certainly to better formulate the necessary and crucial questions we need to continue asking when envisioning an information-centric Internet.

PSIRP is an ongoing research effort. During this phase of development, part of PSIRP's functionality has been implemented (on top of the FreeBSD operating system) and it is being tested (at this point mostly in the local area but a wide-area testbed is in existence and being exploited for experimentation). The implementation is available under open source license terms [18] with the potential to create a growing developer community for this work.

Future work includes defining an inter-domain topology formation protocol, exploring trust related issues in PSIRP networks, implementing a fast forwarding mechanism, porting PSIRP functionality to other operating systems and incorporating existing applications to the PSIRP environment.

References

1. 4WARD: Web site (2010). `http://www.4ward-project.eu`
2. Al-Shraideh, F.: Host Identity Protocol. In: Networking, International Conference on Systems and International Conference on Mobile Communications and Learning Technologies, 2006. ICN/ICONS/MCL 2006. International Conference on, pp. 203–203 (2006)
3. Blumenthal, M., Clark, D.: Rethinking the design of the Internet: the end-to-end arguments vs. the brave new world. ACM Transactions on Internet Technology (TOIT) **1**(1), 109 (2001)
4. Caesar, M., Condie, T., Kannan, J., Lakshminarayanan, K., Stoica, I.: ROFL: routing on flat labels. ACM SIGCOMM Computer Communication Review **36**(4), 374 (2006)
5. Candolin, C.: Securing military decision making in a network-centric environment. Doctoral Dissertation, Helsinki Univarstiy of Technolgy

6. Carzaniga, A., Rosenblum, D., Wolf, A.: Design and evaluation of a wide-area event notification service. ACM Transactions on Computer Systems (TOCS) **19**(3), 332–383 (2001)
7. CCNx: Web site (2010). `http://www.ccnx.org`
8. Feldmann, A.: Internet clean-slate design: what and why? ACM SIGCOMM Computer Communication Review **37**(3), 64 (2007)
9. Jokela, P., Zahemszky, A., Esteve Rothenberg, C., Arianfar, S., Nikander, P.: LIPSIN: Line speed publish/subscribe inter-networking. ACM SIGCOMM Computer Communication Review **39**(4), 195–206 (2009)
10. Jokela, P., ed.: PSIRP deliverable 2.2, conceptual architecture definition, component descriptions, and requirements (d2.2) (2010). `http://www.psirp.org/`
11. Katsaros, K., Fotiou, N., Polyzos, G., Xylomenos, G.: Overlay Multicast Assisted Mobility for Future Publish/Subscribe Networks. In: Proceedings of the ICT Mobile Summit. Santander, Spain (2009)
12. Katsaros, K., Fotiou, N., Polyzos, G., Xylomenos, G., Athens, G.: Supporting mobile streaming services in future publish/subscribe networks. In: Proceedings of the 2009 Wireless Telecommunications Symposium, pp. 337–343. IEEE (2009)
13. Katsaros, K., Xylomenos, G., Polyzos, G.C.: A hybrid overlay multicast and caching scheme for information-centric networking. In: Proceedings of the 13th IEEE Global Internet Symposium. San Diego, CA, USA (2010)
14. Kjallman, J.: Attachment to a Native Publish/Subscribe Network. In: ICC Workshop on the Network of the Future (2009)
15. Koponen, T., Chawla, M., Chun, B., Ermolinskiy, A., Kim, K., Shenker, S., Stoica, I.: A data-oriented (and beyond) network architecture. ACM SIGCOMM Computer Communication Review **37**(4), 192 (2007)
16. Nikander, P., Marias, G.: Towards Understanding Pure Publish/Subscribe Cryptographic Protocols. In: Sixteenth International Workshop on Security Protocols, Cambridge, England (2008)
17. Pietzuch, P., Bacon, J.: Hermes: A Distributed Event-Based Middleware Architecture. In: In Proc. of the 1st Intl. Workshop on Distributed Event-Based Systems (2002)
18. PSIRP: Web site (2010). `http://www.psirp.org`
19. Stoica, I., Adkins, D., Ratnasamy, S., Shenker, S., Surana, S., Zhuang, S.: Internet indirection infrastructure. Peer-to-Peer Systems **2429**, 191–202 (2002)
20. Tarkoma, S., ed.: PSIRP deliverable 2.3, architecture definition, component descriptions, and requirements (d2.3) (2010). `http://www.psirp.org/`
21. Touch, J., Wang, Y., Pingali, V.: A recursive network architecture. ISI, Tech. Rep. pp. 2006–626 (2006)
22. Volker, L., Martin, D., El Khayat, I., Werle, C., Zitterbart, M.: A Node Architecture for 1000 Future Networks. In: Proceedings of the International Workshop on the Network of the Future (2009)