Reconnecting the Internet with ariba: Self-Organizing Provisioning of End-to-End Connectivity in Heterogeneous Networks

Christian Hübsch, Christoph P. Mayer, Sebastian Mies,
Roland Bless, Oliver P. Waldhorst, and Martina Zitterbart
Institute of Telematics – Universität Karlsruhe (TH) – 76128 Karlsruhe, Germany
{huebsch,mayer,mies,bless,waldhorst,zit}@tm.uka.de

1. INTRODUCTION

The evolution of the Internet complicates provisioning of end-to-end connectivity through (1) growing heterogeneity induced by new protocols like IPv6, (2) middleboxes like NAT and Firewalls, and (3) parallel deployment of new Internet architectures and virtual networks. Several solutions are available to re-establish end-to-end connectivity, e. g., 6-to-4 tunnels, NAT traversal and port forwarding, or bridging of virtual networks. Such approaches have several disadvantages: First, they often require manual configuration of dedicated systems. Second, they may not only affect a specific application but also overall network behavior. Third, bridging is complex due to different semantics and functionality provided by the virtual networks.

In recent years, overlay-based applications have become popular since they can add functionality missing in the Internet without infrastructure support, e. g., multicast, partial QoS-support, or security. Furthermore, frameworks for simplifying network topologies or even entire network architectures depend partially or entirely on overlay approaches [4]. Such overlay-based applications rely on end-to-end connectivity between all participating nodes. To achieve this, current applications either provide custom solutions to, e. g., deal with NAT, or cannot be deployed in heterogeneous networks at all.

To ease creation of overlay-based applications we present *ariba*: a generic solution to provide consistent per-application end-to-end connectivity employing identifier-based addressing [1]. *ariba* is fully self-organizing, end-system-based and does not require infrastructure support. On the one hand, *ariba* provides a homogeneous, mobility-invariant network substrate to application developers [2] (cf. Figure 1). On the other hand, self-configuration relieves end-users from error-prone manual configuration. Furthermore, *ariba* is adaptive, i. e., it can handle network settings that are dynamic and may change over time. *ariba* itself uses an overlay to realize basic connectivity over heterogeneous networks. The self-stabilizing property of this overlay assures robustness against node failures. Applications can easily build new overlays on top of *ariba* without dealing with network difficulties like NAT, heterogeneity, and mobility. *ariba* is a part of the *Spontaneous Virtual Networks* (SpoVNet) architecture [4], enabling spontaneous and flexible creation of overlay-based applications and services on top of heterogeneous networks.

We propose a demonstration that illustrates *ariba*'s main feature: application-specific provision of end-to-end connectivity in the face of heterogeneous and dynamic networks, easing both development and configuration of overlay-based applications. The demonstrator shows that *ariba* (re-) establishes application-specific end-to-end connectivity despite heterogeneity and middleboxes, dynamic network changes, and movement of clients between networks.

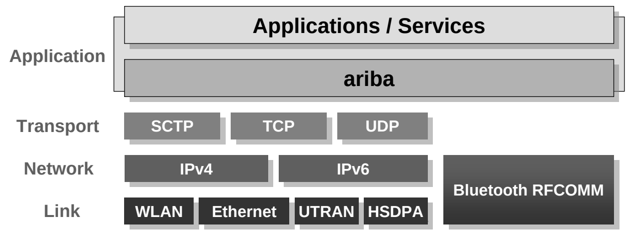

Figure 1: Integration of *ariba* into the network stack

2. MAIN IDEA OF THE DEMO

The main intention of the demonstration is to show how *ariba* eases application deployment upon heterogeneous networks. We consider an exemplary scenario (as shown in Figure 2) that consists of two LANs, one running IPv4 and one running IPv6, respectively. Furthermore, one WLAN attached to notebook N1 and a bluetooth device connected to notebook N3 are deployed. The WLAN uses NAT to multiplex the single IP address of the access point to multiple wireless devices. Furthermore, we employ native RF-COMM for communication between N3 and P2, using MAC addresses. Notebook N2 and N4 are dual-stacked and connected to both, the IPv4 and IPv6 LAN.

All end-systems in this scenario run an application that requires end-to-end connectivity. In the following we refer to the instance of the application running on an end-system as *node*. Two nodes are *directly connected*, if they can communicate through a common subset of protocols and bidirectional packet flow is not inhibited by middleboxes. In the exemplary scenario shown in Figure 2 nodes N1 and N4 are directly connected, whereas N1 and N3 are not. To illustrate the establishment of end-to-end connectivity, consider a communication path between P2 and P1. Using a conventional approach lots of additional mechanisms are required to achieve end-to-end connectivity: First, N2 and N3 need to configure a point-to-point tunnel or personal area network daemon (*pand*) to connect P2 via Bluetooth to the

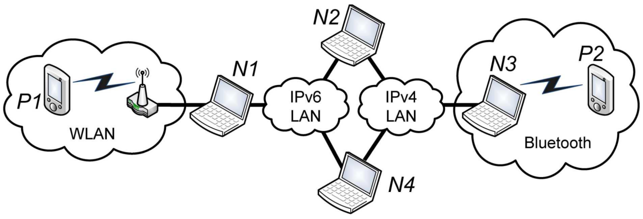

Figure 2: Example scenario consisting of IPv4 and IPv6 LANs, private WLAN, and a Bluetooth point-to-point serial connection.

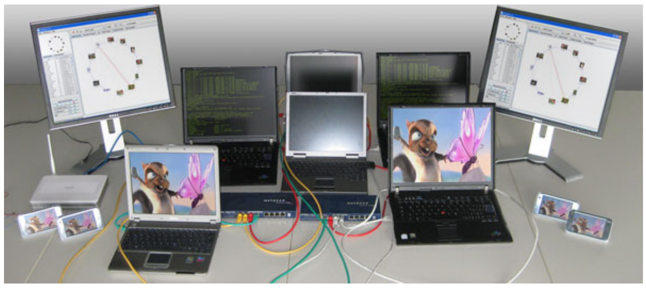

Figure 3: Demonstration setup

IPv4 network. Second, N2 *or* N3 need to be configured to forward packets from the IPv4 to the IPv6 network—this is only possible when using IPv4-mapped addresses. Finally N1 needs to forward packets for P1, and port forwarding must be configured on the NAT device. Note, that if the network setting is changed manual re-configuration is necessary to re-establish connectivity. During this time-consuming re-configuration process—which is usually error-prone and highly complex—end-to-end connectivity is unavailable.

ariba eases this process using a generic self-organizing approach: First, it does not rely on homogeneous addressing or protocols, in fact, *ariba* exploits different protocols to construct an application-layer path—looking homogeneous to the application—upon heterogeneous networks. This path is built hop-by-hop whereas each hop can run different transport- and network-layer protocols. Furthermore, it considers that network settings are dynamic and may change over time. For example, notebook N1 may get connected directly to notebook N3 and updated to support 6-to-4. In this case *ariba* adapts and incrementally optimizes connectivity. For this purpose *ariba* uses an overlay with a consistent *identifier*-based addressing scheme to overcome network heterogeneity: Nodes using the same application are connected by a logical overlay structure that allows forwarding packets using node identifiers (e. g., using one-hop or Chord key-based routing protocols).

The overlay is constructed incrementally: First, a joining node contacts another node—running the same *ariba*-based application—it has direct connectivity with. For example, N1 may use N2 to join the overlay. The joining node must establish connections to its logical neighbors in the overlay. Neighbors are discovered by issuing queries inside the overlay network using key-based routing. For example, if P2 is logical neighbor of N1, the query reaches P2. P2 has two options: P2 might try to establish a direct connection to N1—which is not possible due to heterogeneity—or use the overlay path the query originated to establish a connection. In the latter case, N2 (or N4) and N3 would be used to construct a relay path between N1 and P2. Relay paths may fail if the network setting is changed. In this case the node can re-establish relay paths by partially repeating the join phase for overlay stabilization.

For an instant decision whether two nodes can communicate directly and to optimize the length of relay paths *ariba* implements an unintrusive extension: *Connectivity Domain* management. The extension monitors overlay connections and relay paths to identify regions with direct connectivity—so called Connectivity Domains—and assigns a *Connectivity Domain Identifier* (CDID) to each Connectivity Domain. Using a gossip mechanism nodes inform each other about changes in connectivity characteristics (i. e., a Connectivity Domain split or merge) and resolve conflicting CDIDs. All nodes include CDIDs in their routing information. Thus other nodes can immediately decide whether they can communicate directly with a certain node by comparing CDIDs, which can also be used to discover shorter relay paths [3].

For demonstration purpose, the network settings shown in Figure 2 can be modified by connecting and removing relaying nodes, as well as connecting nodes to different networks interactively. *ariba* will automatically sustain connectivity between nodes. To visualize internal protocol functionality the application additionally shows its local view of the network: relay paths traversing the node and logical neighbors.

In its current form our approach has the following open issues: First, the overlay re-join mechanisms may suffer from overlay partitioning, and second, relay paths may degrade in case of network setting reconfiguration. However, our approach is feasible in a practical setting and allows easy deployment, as shown in the demonstration.

Acknowledgement

This work was partially funded as part of the *Spontaneous Virtual Networks* (*SpoVNet*) project by the Landesstiftung Baden-Württemberg within the BW-FIT program and as part of the Young Investigator Group *Controlling Heterogeneous and Dynamic Mobile Grid and Peer-to-Peer Systems* (*CoMoGriP*) by the *Concept for the Future* of Karlsruhe Institute of Technology within the framework of the German Excellence Initiative.

3. REFERENCES

[1] Ariba. http://www.ariba-underlay.org.

[2] R. Bless, C. Hübsch, S. Mies, and O. Waldhorst. The Underlay Abstraction in the Spontaneous Virtual Networks (SpoVNet) Architecture. In *Proc. 4th EuroNGI Conf. on Next Generation Internet Networks (NGI 2008)*, Apr. 2008. CD-ROM.

[3] S. Mies, O. Waldhorst, and H. Wippel. Towards End-to-End Connectivity for Overlays across Heterogeneous Networks. In *Proc. Int. Workshop on the Network of the Future (Future-Net 2009), co-located with IEEE ICC 2009*, Dresden, Germany, June 2009.

[4] O. Waldhorst, C. Blankenhorn, D. Haage, R. Holz, G. Koch, B. Koldehofe, F. Lampi, C. Mayer, and S. Mies. Spontaneous Virtual Networks: On the Road towards the Internet's Next Generation. *it — Information Technology Special Issue on Next Generation Internet*, 50(6):367–375, Dec. 2008. http://www.spovnet.de.