Oppportunistic Software Deployment in Disconnected Mobile Ad Hoc Networks

Frédéric Guidec, Valoria, Université Européenne de Bretagne, France
Nicolas Le Sommer, Valoria, Université Européenne de Bretagne, France
Yves Mahéo, Valoria, Université Européenne de Bretagne, France

ABSTRACT

This paper presents a middleware platform allowing the dissemination of software components on handheld devices forming a disconnected MANET. It is based on a model that exploits peer-to-peer and opportunistic interactions between neighboring devices to deploy component-based applications, without relying on any kind of infrastructure network. Each device runs a deployment manager, which strive to fill a local component repository with software components so as to be able to satisfy the deployment requests expressed by the user. To do so the deployment manager interacts with peer managers located on neighboring devices, providing its neighbors with copies of software components it owns locally, while obtaining itself from these neighbors copies of components it lacks. The platform also provides communication facilities adapted to disconnected MANETs that notably allow efficient multi-hop exchanges.

Keywords: Mobile Ad Hoc Networks, Opportunistic Networking, Pervasive Computing, Software Component Deployment, Ubiquitous Computing

INTRODUCTION

An approach to handle the complexity of modern software applications is to define these applications as assemblies of software components. Software components are independent, reusable and replaceable units of software that are meant to fulfill a well-defined function in an application (Szyperski, 1998). An important research topic about software components aims at defining how component-based applications can be deployed on a target platform. Some works about software deployment have contributed to identify interrelated activities needed for this deployment. For example in (Carzaniga et al., 1998; Lestideau et al., 2002) it has been proposed to distinguish between activities pertaining to the provision, the delivery, the installation, the configuration, the execution, the adaptation, and the removal of software components. In this paper we mostly focus on two of these activities, namely the provision and the delivery of software components.
The originality of our work lies in the fact that the target platforms we consider for component deployment are disconnected mobile ad hoc networks composed of lightweight mobile devices capable of wireless ad hoc communication (e.g., laptops, netbooks, mobile Internet devices, smartphones). A mobile ad hoc network (MANET) is a network that can appear and evolve spontaneously as mobile devices themselves appear, move and disappear dynamically (Perkins, 2001). Traditionally, MANETs are considered connected, allowing a device to communicate with any other in the network temporarily, thanks to routing by the other devices. However, in many realistic conditions, for example when devices are distributed sparsely or irregularly, a MANET can become disconnected, and get fragmented into communication islands.

For the users of laptops and handheld devices, the prospect of deploying software applications on these devices as and when needed obviously appears as an attractive one, no matter if these devices communicate in infrastructure or in ad hoc mode. Yet, the specificities of MANETs, and especially those of disconnected MANETs, lead us to reconsider the software deployment problem in this particular context.

In this paper we describe a model for software component deployment on disconnected MANETs, as well as a platform that implements this model. The paper is organized as follows. We first motivate our work by showing how infrastructure-based networks and disconnected MANETs constitute radically different environments as far as the problem of software deployment is concerned. We then present the main characteristics of a platform we designed, which provides a specific communication support for disconnected MANETs and a protocol for software deployment in such networks. Finally, the results we obtained by running our middleware platform on a mobile ad hoc network simulator are presented before discussing about related works and concluding the paper.

**RATIONALE**

In this section we show that deploying software components in an ad hoc network raises issues that usually do not appear in infrastructure networks. As a reminder, we first describe how software component provision and delivery are usually performed in an infrastructure-based environment. We then show that a disconnected MANET presents additional constraints that need to be addressed specifically.

**Software Deployment in an Infrastructure Network**

In an infrastructure network, some stable hosts can be in charge of storing components in so-called component repositories, and of implementing server programs capable of delivering these components on demand. Other hosts in the network can then behave as simple clients with respect to these servers. Whenever the owner—or the administrator—of one of the client hosts initiates the deployment of a new component-based software application on this device, the problem mostly comes down to locating at least one of the servers capable of providing the components required by this application, and downloading these components so they can be installed locally. A component may actually be provided by several servers, for example in order to balance the workload in the network, or to allow fault tolerance. In any case, once a client has identified a server that can provide a component, obtaining this component simply requires its download from the server to the client. Note that in such a context the deployment of a component on a given host can usually be considered as a “real time” operation: once a user has ordered the deployment middleware to locate and download a component, this operation can usually be performed immediately.

In the remainder of this section, we show that deploying components in an ad hoc environment can in contrast require a more lengthy process, which requires some middleware
Using the Wii Remote for Mobile Device Application Testing: A Proof-of-Concept
[www.igi-global.com/article/using-the-wii-remote-for-mobile-device-application-testing/138114?camid=4v1a](www.igi-global.com/article/using-the-wii-remote-for-mobile-device-application-testing/138114?camid=4v1a)

On Cryptographically Strong Bindings of SAML Assertions to Transport Layer Security
[www.igi-global.com/article/cryptographically-strong-bindings-saml-assertions/58903?camid=4v1a](www.igi-global.com/article/cryptographically-strong-bindings-saml-assertions/58903?camid=4v1a)