Service Discovery in Wireless Mesh Networks: a Survey on Candidate Solutions

Pedro Fernandes, Rui Rocha
Instituto de Telecomunicações, Av. Rovisco Pais 1, 1049-001 Lisboa, Portugal
pedro.figueiredo@lx.it.pt, rui.rocha@lx.it.pt

Abstract — **In self-organized mobile networks, the need to know identifiers in order to establish connections is a burden to the users. The focus can be shifted from the nodes to the services they provide. Service discovery enables simple browsing and location of services available in the network and can offer support to the creation and advertisement of user communities. Although solutions exists for ad-hoc networks, the combination of mobility, organization and configuration issues in wireless mesh networks render current solutions inappropriate. This paper compares two of the current main service discovery solutions and proposes enhancements for a better suitability for multi-tier wireless mesh networks.**

I. INTRODUCTION

The advances on information and communications technology (ICT) are providing the bases for a pervasive and ubiquitous access to information and services. In next generation wireless network architectures, such as the one envisioned in WIP [1], users can connect to the global internet through self-organized wireless access networks. These are in turn interconnected by a mesh infrastructure which ultimately connects to the fixed internet. Furthermore, throughout the whole network, users can associate themselves in user communities, according to their interests and needs. In this context, interconnection of computers and devices can no longer be an issue reserved for network administrators and computer experts. In order to raise the abstraction level, hiding network organization and configuration aspects from the user, the service paradigm is of vital importance.

Instead of perceiving a network as a group of nodes with different addresses and functionalities, users want to see a network as a collection of services they can easily browse and use. Likewise, users want to easily discover communities matching their interests and offer them their own services. All this should be possible independently of the network topology and with a minimal need for user-based configuration.

Service discovery protocols not only provide support to this view, but become mandatory in networks based on self-organization and mobility. When the network is composed of nodes which can join or leave the network at any time, addresses are automatically configured and cannot be practically used for identification. Even with dynamic naming mechanisms (such as mDNS[2]) users still have a proactive role, as they have to maintain knowledge about identifiers. The service paradigm allows the focus to be shifted from the nodes themselves to the services they provide, and service discovery offers the means necessary to browse, locate and use them. Basic as well as mandatory network and configuration services can thus be found and accessed, along with other available services available for use.

From a more technical point of view, a service discovery solution for wireless networks should offer the necessary scalability and robustness to operate in large dynamic environments. Mobility should be supported, not only by offering discovery mechanisms, but also by maintaining a coherent state and providing mechanisms for mobility awareness and response. The protocol should be flexible so that the requested service information can be obtained using minimal resources. In the same manner, each node should only need to run the set of protocol components it actually requires for its operation. Simple service provisioning and connection should be supported. Community awareness should be taken into account, allowing service discovery inside communities.

Even though the problem of Service Discovery in wired and wireless ad-hoc networks has been addressed by many projects (such as the SADHOC project [3]) the combination of mobility, organization and configuration issues in large scale dynamic multi-tier wireless mesh networks (WMNs) make service discovery challenging as current solutions cannot be directly applied in a satisfactory manner.

The remainder of this paper is organized as follows: section II presents two of the most well known service discovery protocols and discusses their applicability to complex multi-tier wireless mesh networks such as WIP's. Section III proposes enhancements to improve service discovery performance in these environments. The document ends with some brief conclusions.

II. SERVICE DISCOVERY SOLUTIONS

There are quite a few service discovery protocols proposals, many of them targeting ad-hoc networks. However, only few among them have met recognition and success from the users. Microsoft's Universal Plug & Play [4] (UPnP) and its Simple Service Discovery Protocol (SSDP) [5] are such an example. UPnP's main objective is to support the interconnection between stand-alone devices and computers. It provides node auto-configuration, service discovery mechanisms and service connection management. Integrated in the company's operating system, UPnP is well known for the easiness brought to the process of finding and configuring network devices, such as printers. UPnP's direct opponent, Apple's Bonjour [6], has also met considerable success. It also offers auto-configuration mechanisms and service discovery for small local networks. It is, however, less oriented to the connection of devices and more focused on enabling service

This work was supported by the European Commission project WIP under contract 27402.

and information sharing without user configuration. Also, instead of proposing a patented solution, Bonjour is based on open standards. More specifically, it is an IETF's Zeroconf [7] implementation and, in what concerns service discovery, uses IETF's DNS-SD [8], which on its turn is based on the DNS [9], [10] architecture. Despite its natural orientation towards small ad-hoc networks, Bonjour can also use DNS-SD and regular DNS servers to operate in the Internet, registering, browsing and locating services in DNS domains, in what is called Wide Area Bonjour. Finally, another rather successful service location protocol, also used by Apple Inc until recently and still used in linux distributions, is IETF's Service Location Protocol (SLP) [11]. Although not as simple and easily deployable as DNS-SD, SLP can also operate in both ad-hoc networks and larger structured networks, providing flexibility and functionality. Moreover, among some proposed protocol extensions, mesh-enhanced SLP (mSLP) [12] deserves special attention as it applies the principles of a mesh overlay to improve scalability in large networks.

A. Service Discovery in Wireless Networks

When considering a service discovery solution suitable for both wireless ad-hoc and mesh networks, one has to consider a set of important aspects and functionalities. The ability to operate in both fully distributed manner and with a mesh overlay are naturally critical aspects to support both small and large mobile wireless networks. The way protocol messages propagate in the network and the use of caches are two important aspects since they affect the usage of the rather constrained wireless network bandwidth and the energy consumption. A main objective of service discovery is to raise the abstraction level and be as user-friendly as possible, making networking accessible to everyone. The protocol flexibility on service browsing, queries for service information, connection establishment and control over provided services cannot thus be ignored. Also related to this objective, the support of the emerging concept of communities of users should not be left out. Finally, in order to correctly support a dynamic environment, the meshed overlay needs efficient mechanisms to self-organize, exchange information, maintain coherent state and respond to changes in the network.

B. DNS-SD, mSLP and Wireless Mesh Networks

From the small selection of popular service discovery solutions, DNS-SD and mSLP are probably those which attract the most attention from the research community as they are recognized open standards in this area. Both have shown their suitability to small ad-hoc networks, where service requesters use multicast (mDNS extension in DNS-SD) to flood the networks with their queries for service information, to which service providers answer with matching information concerning the services they offer. However, when considering large scale networks, matters are different. mSLP proposes a meshed overlay to maintain and exchange information on different portions of the network, while DNS-SD suggests the use of hierarchies of DNS name servers to achieve the same purpose. Both, however, need improvements in order to appropriately support large wireless networks and the inherent mobility issues. This is an especially complex issue for DNS-SD, since the domain name server hierarchy is not known for its dynamicity and flexibility.

Scalability

Message propagation must be carefully controlled in order not to overwhelm the network, even though discoverability is sacrificed, as a query may not reach all instances of matching services. DNS-SD limits message forwarding to a maximum number of hops in ad-hoc networks, whereas mSLP does not impose any restriction. Responses in DNS-SD are also multicast, which has interesting advantages, like the fact that every node overhears all queries and respective responses and caches the information for their own usage, despite an obvious impact on protocol traffic. In wide area networks, DNS-SD messages only travel between DNS name servers associated with the DNS domains related to the query. In mSLP, even if the network is organized in scopes (logical groups of nodes), every service directory in the mesh overlay maintains a full list of available services on each scope it belongs to, which may lead to a considerable amount of traffic in larger networks and scopes.

Service Information Maintenance

When a node joins a network or moves, it has to make itself (and its services) known. In mSLP, unsolicited service announcements are inexistent which translates into a non-optimized mobility response. DNS-SD and mDNS propose announcements to be made when connectivity changes are detected, like joining or changing networks.

Even though service announcements could be used by the service directories (Directory Agents and DNS servers) to collect information about the services available in their area, in both solutions the providers are responsible for registering their services. In mSLP, nodes register their services in every directory they can hear announcements from, as long as scopes match. Furthermore, directories propagate these registrations to their peers in the same scope, resulting in duplicated messages. DNS-SD, on the other hand, takes a very different approach. Providers register their services in chosen DNS domains using Dynamic Updates [13] to contact the local DNS server, which is either manually configured, or configured with DHCP [14]. This is still clearly an approach which scares normal users and brings the need for administrators.

As nodes are mobile, the information stored in service directories needs to be kept fresh, which means that registrations are associated with lease times. To maintain the information updated, providers need to periodically notify their presence to the respective directories. While mSLP requires providers to reregister their services, DNS-SD takes advantage of Dynamic DNS Update Leases [15] extension to use smaller notification messages.

Flexibility

For better user experience and protocol efficiency, browsing should be as flexible as possible, allowing users to specify the kind of services they are looking for. In this aspect, mSLP offers great flexibility by allowing services to be browsed by abstract types, concrete types and service attribute conditions (e.g. all printers within a certain area). DNS-SD, on the other hand, allows only browsing by domain, service type and subtype, limiting the queries' expressive power.

In the same manner, a user should be able to obtain the information he desires about a given service, as supported by mSLP. DNS-SD, however, allows only the full list of service attributes to be requested, playing in favor of simplicity instead of efficiency.

Service Management

It can be interesting for the user to have the solution manage the service startup and shutdown as well as connections to remote services. This, however, can imply restrictions to the services themselves and their development. The usual approach is to store as service attributes all necessary information for service connection and configuration and leave these tasks for the service clients themselves, as adopted by both DNS-SD and mSLP. If a standard for service discovery becomes popular enough, it might be interesting to support the two approaches, combining the advantages of both.

Self-organization

To be administrator-free, self-organization is of paramount importance. Unfortunately, DNS-SD does not support it in wide area networks. In fact, DNS server hierarchy is very rigid and still has to be mostly manually configured. The same happens with the association of a node with a server. mSLP, on the other hand, requires nodes to register with all directories from which they can hear their advertisements, as long as they belong to at least one of the nodes' scopes. Directories, in their turn connect to every directory they hear within the same scopes. This allows the nodes to self-organize without any user intervention, even though the processes could be improved by restricting the number of associations each node can perform.

Communication in Large Networks

Communication between directories must also be efficient, reducing traffic as possible to assure scalability, while assuring response to mobility. Directories in mSLP take a rather extreme approach, as they request from their peers either the full list of available services in a given scope, or a list of those registered since a given instant in time, or all services announced by a given peer. Although this allows changes to propagate on the network, it can obviously affect traffic and scalability. Based on DNS, DNS-SD uses the same mechanisms to obtain information: recursive DNS queries. Being on-demand assures the traffic is kept to a minimum, but can also have a negative impact on mobility response.

Event support can mitigate these issues by providing special services that publish changes in the network to which clients can subscribe (e.g. a user might want to be notified when a file server somewhere on the network becomes online). DNS-SD can use DNS Long Lived Queries [16] for that purpose.

User Communities

Finally, even if the ability for the user to be able to browse, join and create communities is more and more a desired feature, it is unsupported by both solutions. To announce, browse and discover communities is not much of a challenge, as they can be treated like regular services. To make the service discovery solution aware of communities is another matter. SLP scopes can be viewed as simplistic communities, as they are logical groups of nodes. They have, however, no information that can characterize them other than an identifier. Although the concept of SLP scopes can be enriched, DNS domains, on the other hand, are far more rigid. Their hierarchical nature and their relation to topological network organization make them a lot less flexible and probably unfit for dynamic environments such as those of wireless mesh networks.

III. ARCHITECTURE ENHANCEMENTS FOR WMNS

A suitable service discovery solution for wireless networks should be able to operate in both meshed and fully distributed manners, according to network topology. In the first case, depicted in fig. 1, the network topology is considered to have a multi-tier organization consisting of access networks interconnected by a mesh of access points. When access points are absent, the network has a flat topology and the solution has to operate in a fully distributed manner.

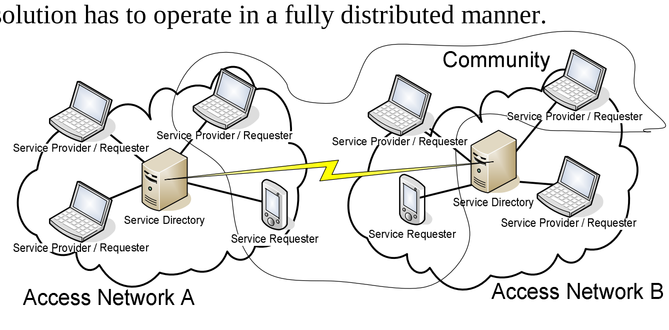

Fig. 1. Service Discovery architecture for mesh networks.

Taking into account the discussion in section II, three entities with distinct functionalities should be here considered: the service provider, the service requester and the service directory. Service providers are nodes which make services available to others. They announce their services to their vicinity in a spontaneous network, or to their access network. In the latter, they should also periodically notify any local service directories of their presence. The service requesters, on the other hand, may browse, discover and locate available services, either by broadcasting queries to its neighbors or contacting local service directories. Finally, service directories act as central service information repositories which can be contacted by local requesters and other service directories. They are located in, at least, some of the access points in the network mesh infrastructure and organize themselves in an overlay. They passively collect information about the services available in the respective access network and can request additional information from other directories. With the help of the providers, they maintain the stored information coherent. Periodically, they announce their presence so that both requesters and providers can learn about them. When requested, they may also notify requesters or peer service directories about changes on the local service panorama.

Support to the emerging concept of communities should also be provided, not only by offering mechanisms to discover services within communities, but by also offering means to announce and discover communities themselves.

A. Service Discovery and the Protocol Stack

Most service discovery protocols fit between the application layer and the network layer. This allows interaction with the applications (services), launching and establishing connections for them, as well as with the network layer, broadcasting and forwarding protocol messages.

There are, however, some interesting cross-layer interactions to be considered. Content-based routing is one of them, as the description of services and data can be used to route messages and can play an especially important role in routing and forwarding messages within user communities. Geographic naming schemes can enable browsing, locating and discovering services in specific physical locations. Service directories can also have a deeper relation with mesh routers, apart from simply residing in the same node. They can take advantage of their auto-configuration mechanisms so that when nodes are configured to access the mesh router, they can also be configured to use the respective service directory. Finally, mobility detection can greatly improve protocol performance. By having knowledge of when a node moves, the protocol can react by re-announcing the services the node provides, making its new neighborhood aware of its presence. This can reduce the number of exchanged protocol messages by avoiding the need for periodic service announcements and improves response to mobility.

B. A Starting Point for a Suitable Solution

We are currently working on service discovery solution tailored for wireless mesh and ad-hoc networks: WiNe-SD (Wireless Networks Service Discovery). Given that a clean board approach usually results in unnecessary work, known projects can be used as a starting point to the development. The work is being focused on improving and enhancing the functionalities which present limitations according to the discussion presented in this paper. Taking this into account, along with the arguments presented in section II, mSLP seems to be an excellent candidate to be extended and enhanced according to the desired functionalities.

IV. CONCLUSIONS

The service paradigm is of vital importance in order to hide network organization and configuration aspects from the user, becoming mandatory in networks based on self-organization and mobility. In networks where nodes can join or leave at any time, addresses are assigned to nodes by self-configuration and self-organization mechanisms and cannot be practically used for identification. Moreover, even with naming schemas, users still have to maintain knowledge of identifiers. Shifting the focus from the nodes themselves to the services they provide, service discovery offers the means necessary to browse locate and use them, enabling user-friendly ubiquitous networking environments.

Even though the problem of Service Discovery in wired and wireless ad-hoc networks has been studied, the combination of mobility, organization and configuration issues in large scale dynamic multi-tier wireless mesh networks make current service discovery solutions, such as SLP and DNS-SD, not directly applicable in a satisfactory way.

This paper discusses these two well known service discovery solutions in the context of wireless ad-hoc and mesh networks, highlighting their shortcomings and proposing some enhancements.

REFERENCES

[1] "IST WIP - An All-Wireless Mobile Network Architecture." [Online]. Available: http://www.ist-wip.org

[2] S. Cheshire, M. Krochmal, "Multicast DNS", *Internet Draft*, 10th August 2006.

[3] P. Fernandes, V. Aleluia, R. Rocha, F. Mira da Silva, "Mobile Self-organized Service-oriented Networks", in *Proceedings ICT 2006*, Funchal, Portugal, 9-12 May 2006.

[4] UPnP™ Forum, "UPnP™ Device Architecture", 8 June 2000.

[5] Y. Goland, T. Cai, P. Leach, Y. Gu, S. Albright, "Simple Service Discovery Protocol/1.0", *Internet Draft*, October 28, 1999.

[6] Apple Computer, Inc., "Bonjour Overview", 2006-05-23.

[7] E. Guttman, "Autoconfiguration for IP Networking: Enabling Local Communication", *IEEE Internet Computing*, June 2001.

[8] S. Cheshire, M. Krochmal, "DNS-Based Service Discovery", *Internet Draft*, 10th August 2006.

[9] P. Mockapetris, "Domain Names - Concepts and Facilities", *RFC 1034*, November 1987.

[10] P. Mockapetris, "Domain Names - Implementation and Specifications", *RFC 1035,* November 1987.

[11] E. Guttman, C. Perkins, J. Veizades, M. Day, "Service Location Protocol, Version 2", *RFC 2608,* June 1999.

[12] W. Zhao, H. Schulzrinne, E. Guttman, “Mesh-enhanced Service Location Protocol (mSLP)”, *RFC 3528*, April 2003.
[13] S. Cheshire, M Krochmal, “Multicast DNS”, *Internet Draft*, 10th August 2006.
[14] P. Vixie, S. Thomson, Y. Rekhter, J. Bound, “Dynamic Updates in the Domain Name System (DNS UPDATE)”, *RFC 2136*, April 1997.
[14] R. Droms, “Dynamic Host Configuration Protocol”, *RFC 2131*, March 1997.
[15] S. Cheshire, M. Krochmal, K. Sekar, “Dynamic DNS Update Leases”, *Internet Draft*, 10th August 2006.
[16] S. Cheshire, M. Krochmal, K. Sekar, “DNS Long-Lived Queries”, *Internet Draft*, 10th August 2006.