Public Infrastructures for Internet Access in Metropolitan Areas

Elias C. Efstathiou, Fotios A. Elianos, Pantelis A. Frangoudis, Vasileios P. Kemerlis
Dimitrios C. Paraskevaidis, Eleftherios C. Stefanis and George C. Polyzos

Mobile Multimedia Laboratory
Department of Computer Science
Athens University of Economics and Business
Athens 113 62, Greece

efstath@aueb.gr, elianos@cs.aueb.gr, pfrag@aueb.gr, vpk@cs.aueb.gr
dcp@aueb.gr, leste@aueb.gr, polyzos@aueb.gr

***Abstract*—Wireless Community Networks (WCNs) are metropolitan-area networks whose nodes are owned and managed by volunteers. These networks can be used to build large scale public infrastructures for providing ubiquitous high-speed wireless broadband access through the private contributions of individual community members who use their hotspots to forward foreign traffic from and to nearby low-mobility clients. We have designed and developed a prototype aggregation scheme that (1) assumes that community members are selfish and do not trust each other and uses a secure incentive technique to encourage their contribution; (2) protects the real-world identities of community providers and clients by relying only on disposable opaque identifiers (public/private key pairs); (3) is fully distributed, open to all, and does not rely on any authority to resolve disputes or to control membership; (4) is automated, using standard hardware and software we developed for some of the main available platforms (Linux-based WLAN access points and Windows Mobile-based cell phones). Thus, it can easily complement 2G/3G cellular networks in metropolitan areas where some WCNs provide wide coverage.**

***Index Terms*—Community Networks, Peer-to-Peer, Incentives, Security, WiFi Networks, Secure VoIP.**

I. Introduction and Related Work

The *WiFi* standard for *Wireless LANs* (WLANs), is becoming increasingly popular worldwide for implementing hotspots that provide wireless Internet access in campuses and many other public areas. WiFi enabled network interface cards are now becoming standard equipment for mobile devices such as laptops, PDAs and advanced cell phones. Moreover, low-cost wireless access points (APs) are increasingly used even in households, providing wireless coverage for home networks. This popularity, along with its easy and inexpensive deployment, indicates WiFi as a technology that will be an integral part of a high-speed wireless broadband provision system with aspirations to appeal to the masses.

Wireless Community Networks (WCNs) are metropolitan wireless networks that can be viewed as the evolution of amateur radio communities and they are characterized by an altruistic and self-organizing spirit. Their nodes are owned and managed by volunteer WLAN enthusiasts and can be constructed by using many individual hotspots.

WCNs are an emerging mode for the provision of public broadband services. Point-to-point wireless links are used to create a citywide backbone network and operate in the unlicensed 2.4 and 5.8 GHz bands. High-gain directional antennas are employed so that long-distance links (of a few kilometers) can be achieved using networking equipment designed for short-range transmission. At the edges of the WCN there exist nodes that also operate omni-directional antennas, acting as Access Points to the WCN for those who wish to join the network without setting up a backbone node. Usually, such community networks have a mesh-like topology and provide a variety of services to their members, ranging from web and ftp to game servers and voice over IP within the boundaries of the WCN. However, there is limited or no roaming support within the WCN.

In recent years, with the proliferation of low-cost WLAN equipment, WCNs, as well as other privately-operated Wireless LANs offer nearly complete coverage in some densely-populated metropolitan areas. Successful WCNs in terms of coverage and number of participants include *Seattle Wireless* [1], *NYCwireless* [2], and the *Athens Wireless Metropolitan Network* [3].

We have been developing and advocating a practical incentive scheme which could be used to stimulate participation in wireless communities and make ubiquitous wireless Internet access become a reality through the private contributions of WLAN owners. Our prototype system for WLAN sharing is based on indirect service reciprocity – only WLAN owners who share their bandwidth with others may consume bandwidth when they themselves are mobile. We call the proposed scheme the *Peer-to-peer Wireless Network Confederation (P2PWNC)*.

Work with similar motivation to ours is presented in [4]. WISPs have multilateral roaming contracts and must register with a central authority that maintains reputation records, which are updated with QoS reports submitted by the roamers. There are also a few commercial solutions that deal with WLAN sharing [5][6][7], which usually involve central management and aim at WLAN sharing in a for-profit basis.

Among all services that WCNs provide to their users, we focus on the provision of wireless Internet access to pedestrian users through WCN controlled WLAN access points (APs). If such a service were commonplace, WCNs would be able to complement 2G/3G cellular networks in metropolitan areas. This has become more important with the advent of WLAN-enabled mobile phones [8]. WCNs could also rival in coverage similar centrally managed WLAN schemes, such as *Wireless Philadelphia* [9]. We expect that the most prevalent application over the proposed architecture will be free, anonymous and secure VoIP.

In order to generate a 2G/3G rival we can't rely on WNC participants' altruism. Instead, we use an indirect *reciprocity algorithm* whose input is the system's history of prior Internet service provisions and its role is to identify potential *free-riders*[1] (and exclude them from service), by providing users with the appropriate incentives to share their Internet connections using their home WLAN equipment. Extensive simulations presented in [10] show that the P2PWNC scheme can stimulate participation in WCNs and sustain cooperation between community members in the form of indirect reciprocity. The proposed mechanism is compatible with the distinctive nature of WCNs; it does not require registration with central authorities and relies only on free identities. The real world identities of the scheme's users are protected, and free, anonymous WLAN roaming can be achieved.

In the P2PWNC scheme, participants are identified by uncertified public/private key pairs. The P2PWNC accounting scheme is based on the exchange of "receipts" digitally signed by service consumers, as proofs of service provision. Communication between the entities of the proposed architecture is carried out using a simple ASCII-based protocol. Our protocol makes use of public key cryptography, and, in particular, the RSA and Elliptic Curve Digital Signature Algorithm. In order to protect the resources of the WLAN owners and to implicitly punish/reward users according to their contributions, we use different Quality-of-Service levels.

Our scheme was designed with resource-constrained, embedded devices in mind. Thus, we have implemented it on top of common, low-cost WCN equipment (such as the Linksys WRT54GS wireless router [11]) and WLAN-enabled smart-phones. What is more, we have conducted performance measurements to test the P2PWNC protocol's overhead and its effects on the behaviour of the embedded devices upon which it operates.

The remainder of this paper is organized as follows. Section II briefly presents the P2PWNC scheme. In Section III we propose Quality-of-Service extensions to our architecture and a performance evaluation follows in Section IV. In Section V we discuss deployment issues of our architecture in metropolitan areas, before we conclude in Section VI.

II. The P2PWNC Hotspot Roaming Architecture

A. Entities and Trust Model

In the P2PWNC scheme, users are organized into small teams. Teams manage and operate a number of WLAN access points connected to Cable/DSL links at locations throughout the city. Team A *consumes* when a member of Team A accesses the Internet through an AP of another Team, B, and *contributes* when a member of a team other than Team A uses an AP that belongs to Team A.

Teams are the *peer* entities of our scheme. A team is a service provider and a consumer at the same time; it provides service via its own access points, and it consumes when its members visit foreign WCN-controlled access points.

Each team is identified by an uncertified public/secret key pair. A member of the team acts as the *team leader* who is responsible for maintaining the team's secret key. The team leader can recruit team members by issuing a public/secret key pair and a *member certificate* for each one of them. A certificate denotes membership of an individual with the issuing team.

It should be stressed that since no central authority is required for certifying peer identities and since user identities are not bound with real-world ones, member privacy is enhanced. Peer pseudonyms in P2PWNC are *cheap* and disposable.

The P2PWNC is designed to operate in a fully decentralized manner. However, in some cases, a centralized setting with minimal demands on the center may be more preferable. To elaborate, there are cases when a central entity may enhance the operation of the P2PWNC scheme without compromising its peer-to-peer and autonomous nature. The responsibilities of such a central entity are limited to maintaining the system's *history of transactions*. In this case, it is assumed that all peers trust the central entity (meaning that they use the transactions stored there because they provide an overall view of the system).

B. Service Accounting Scheme

Each time a service provision takes place, a digital *receipt* is generated. The receipt is signed by the consuming roamer using his secret key as an acknowledgement of the provided service. It contains the certificate of the consumer, the public key of the provider, a timestamp and the receipt

[1] Free riders: Users who consume resources without contributing to the community.

weight. The timestamp represents the time a session between a mobile user and an AP started, while the weight field shows the amount of traffic forwarded by the AP on behalf of the consumer. The structure of a P2PWNC receipt is presented in Fig. 1.

Receipts represent the system's history of transactions. They are stored in team-local repositories and form a directed graph which is used as input to a *reciprocity algorithm*. The vertices of this graph are the peers of the system, while graph edges point from service consumers to service providers, encoding an "I owe you" relationship. We treat service dept transitively; if peer A owes service to peer B and peer B owes service to peer C, then there is an indirect dept from A to C.

The reciprocity algorithm is responsible for deciding whether a roamer deserves to be granted service by a potential provider based on the prior contributions of the two parties. The reciprocity algorithm makes use of maximum flow techniques [12][13] to calculate the amount of service the potential provider indirectly "owes" to the service requesting peer. Its role is to identify free riders and exclude them from service and its output can be used to decide on the QoS level that the roamer may enjoy.

Our reciprocity algorithm [10] encourages peers to match their consumption by at least an equal amount of contribution. Free-riders that wish to consume much more than they contribute will find it hard to obtain service. What is more, only short term history is important: old receipts get outdated, so peers must contribute continuously in order to be able to consume continuously.

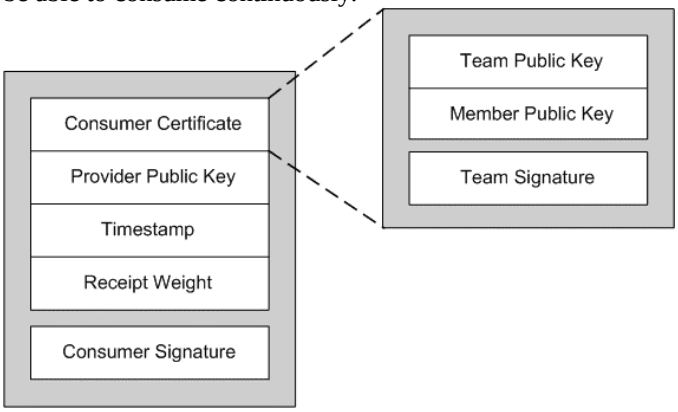

Fig. 1. P2PWNC digital receipt. A receipt is a proof of a service provision. It contains the certificate of the consumer, the public key of the providing team, a timestamp denoting the time the session started and the receipt weight. It is digitally signed using the consumer's private key.

C. The P2PWNC protocol

P2PWNC entities communicate using a simple ASCII-based protocol. Since some of the messages contain binary data, they are Base64-encoded to appear in readable form.

At the beginning of a P2PWNC session, the roamer issues a connection request. The visited AP queries the receipt repository and is notified whether the requesting party should be granted service, as well as the QoS level that he deserves. If the response to the query is positive, the P2PWNC session is initiated. During a session, the AP periodically requests that the client signs a fresh receipt, acknowledging the service he has consumed thus far. The session terminates as soon as the client fails to deliver a receipt in response to a receipt request (which normally happens when a client walks off the AP). There is no explicit session termination message.

Apart from the receipt generation protocol, there is also a gossiping protocol, which assists in receipt dissemination among the team-local receipt repositories (when operating in decentralized mode). Each time a client approaches a foreign AP, he can present the AP with a number of receipts that are then forwarded to the receipt repository. These receipts can help the AP have a broader view of the system's history. Thus, the client can inform the AP of his contributions and can improve the chances of acquiring service of better quality. Such a gossiping protocol is very important in the decentralized mode of operation, where peers have a limited view of the system's history of transactions.

The P2PWNC scheme makes extensive use of public key cryptography for operations such as generation and verification of digital signatures. The RSA and ECDSA digital signature algorithms are both supported.

We have implemented out protocol to run on top of common WLAN equipment as well as desktop PCs and WLAN-enabled smart-phones. Our reference implementation is open-source and is available for download from the project's website [14]. The complete specification along with an initial evaluation of the P2PWNC protocol can be found in [15]

III. Quality-of-Service Extensions

A major issue during the development of the P2PWNC scheme was Quality-of-Service provision for both the P2PWNC-enabled access point owners and mobile clients.

Different QoS levels can be used to protect the resources of the WLAN owners and to reward/punish the P2PWNC roamers. Without such a mechanism in place, visitors might consume most of the available Internet bandwidth and the Internet link to which the AP is connected would be rendered useless for its owner. A peer that does not contribute much will find it hard to obtain service of acceptable quality.

For this task we have implemented a QoS module and extended the P2PWNC protocol in order to provide QoS services similar to the one described above. For the implementation of the module we used a popular bandwidth sharing algorithm [17] along with standard tools [16] that can be used to provide AP owners with the ability to control the bandwidth that mobile clients can consume while providing mobile clients with a guaranteed bandwidth (or class of service) depending on their contribution. This can stimulate mobile users to always contribute more in order to get a higher guaranteed bandwidth share when they visit foreign APs.

To protect both the Internet bandwidth and the wireless resources of an AP the QoS module can easily be applied. For example, if a P2PWNC participant owns a 2Mbps xDSL line connected to an 11Mbps WLAN AP, then he can limit the resources dedicated to P2PWNC to 512Kbps of Internet bandwidth and 2Mbps of wireless connectivity.

We define rb as the real bandwidth of a resource (xDSL line or WLAN connection) and pb the bandwidth of the resource that is available for P2PWNC peers (apparently, pb is a portion of rb). The bandwidth that is guaranteed for the AP owner is $rb - pb$ and the guaranteed bandwidth for each mobile client is:

$$cb_i = \frac{QValue_i}{\sum_{j=1}^{n} QValue_j} * pb .$$

$QValue_i$ is an eight (8) bit unsigned integer value that is returned by the reciprocity algorithm depending on the mobile user's contribution (values 0-255, higher values indicate greater contribution). $\sum_{j=1}^{n} QValue_j$ is the sum of all *QValue*s for the mobile clients that currently use the visited AP. Thus, cb_i is a value normalized to [0, 1] ensuring that users with high contribution have higher guaranteed bandwidth. Also, since each time a mobile user joins or leaves the visited access point all cb_i values are recomputed so that the guaranteed bandwidth is adaptive depending on the AP's load.

IV. Performance Evaluation

To evaluate the performance of our protocol we have conducted a set of experiments. Our aim was to test the behavior of the Linksys WRT54GS wireless router under heavy routing load and during the operation of the P2PWNC protocol. We wish to explore the performance overhead of the proposed scheme in real-world scenarios.

A. Testbed

In this section we present our experimental testbed. Our testbed was composed of fourteen (14) desktop PCs, two eight-port (8) 100BaseTX Ethernet switches and a Linksys WRT54GS wireless router with the OpenWRT [18] firmware, on top of which the P2PWNC software runs. Each switch was used to connect seven (7) PCs. The exact hardware and software specifications of the equipment used are presented in Table I.

TABLE I
PLATFORM SPECIFICATIONS

Characteristic	PC Workstations	Linksys WRT54GS
CPU speed	3.00 GHz	200 MHz
CPU type	Intel Pentium 4	Broadcom MIPS32
RAM	512 MB	32 MB
Storage	2x70 GB HD	8 MB Flash 32 KB NVRAM
Network interfaces	SiS 100BaseTX Ethernet cards	Broadcom integrated 4-port 100BaseTX Ethernet switch 100BaseTX Ethernet WAN interface 802.11g wireless interface
Operating system	Linux kernel 2.6.10 (Knoppix 4)	Linux kernel 2.4.20 (OpenWRT)
Cryptographic Library	OpenSSL 0.9.8b	OpenSSL 0.9.8b

Fig. 2 depicts the experimental setting that has been developed. There are seven (7) transmitter-receiver pairs. Transmitting and receiving hosts are connected to two separate switches. Transmitting hosts are then connected (via the switch) to the LAN interface of the wireless router and are assigned private IP addresses in the 192.168.1.0/24 range. Receiving hosts are connected to the WAN interface of the router and their IP addresses are in the 192.168.0.0/24 range. The Linksys router performs Network Address Translation for the LAN hosts.

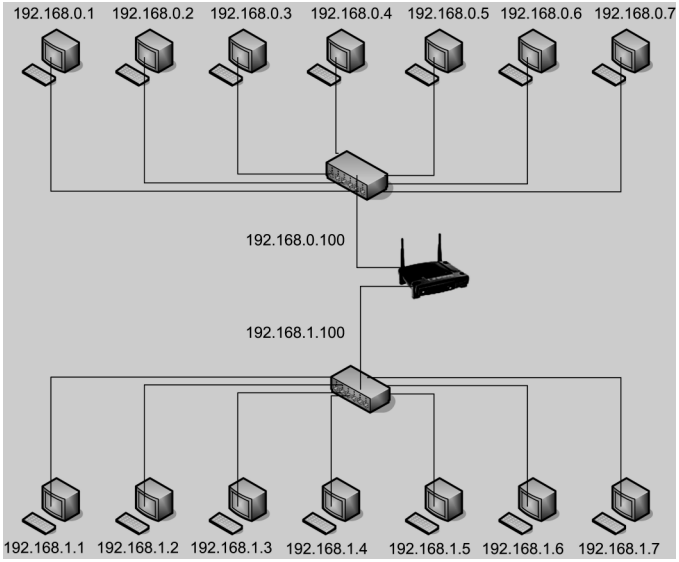

Fig. 2. Experimental Testbed. Transmitting nodes are in the 192.168.1.0 sunbent while receivers are in 192.168.0.0. Transmitters are in the Linksys router's private subnet, while receivers are reached via the router's WAN interface.

B. Methodology

Although our architecture aims at wireless communities, in the experiments presented in this section all communication is carried out over 100Mbps Ethernet. This decision was taken since we wish to measure the effects of our protocol on the routing performance of the Linksys WRT54GS box under more extreme traffic conditions than using the 54Mbps wireless interface.

In all our experiments, 100 Mbytes are transferred over TCP from transmitting to receiving hosts using the ttcp [19] utility. We measure the throughput achieved during the transmission from host 192.168.1.1 to host 192.168.0.1, in the presence of other simultaneously active transmitting-receiving pairs. The number of simultaneously transmitting hosts ranges from one (1) to seven (7).

Four sets of experiments have been carried out. The first set measures the pure throughput achieved without the intervention of the Linksys box and is used as a reference for the subsequent sets.

For the second set of experiments, the transmitting nodes are in the router's private subnet and all traffic is routed to the receiving hosts using Network Address Translation. We have carried out these measurements to observe the throughput degradation caused by the router's NAT module.

The last two sets of experiments involve the operation of the P2PWNC protocol. The transmitting hosts engage in P2PWNC sessions with the AP. After these sessions have been established, the data transmission occurs in the same manner as in the second experiment. During the transmission, the AP periodically asks clients for receipts, which are verified upon reception by the AP. Now, we wish to test the overhead of the P2PWNC protocol in the routing process. In these two final experiments we compare the application of the RSA and ECDSA algorithms respectively.

In each set of experiments a curve is generated. This curve consists of a number of data points which represent the throughput achieved by host 192.168.1.1 in the presence of 0, 2, 4 and 6 additional transmitting-receiving pairs. Each point is generated by repeating the experiment five (5) times so as to estimate the variance and averaging the five (5) throughput values.

Successful execution of the experiments involves synchronization among participating hosts. To achieve this, we first synchronized the hosts using NTP and then scheduled simultaneous data transmissions using the Linux *crond* scheduler daemon on each host.

C. Setup Parameters

All the experiments were carried out in single user mode. We also minimized the number of running processes in the participating hosts and the Linksys box. Our setup was designed so that no other network traffic than the generated TCP flows and the P2PWNC protocol message exchange was present during the execution of our experiments.

As to the P2PWNC specific parameters, we fixed the RREQ interval (the time between two successive receipt requests by the AP to a client) to five (5) seconds, which is a reasonable choice considering our expectation that the primary application over P2PWNC will be VoIP; under such an assumption, we expect clients to place VoIP calls of a few seconds to few minutes, so requesting a fresh receipt every five (5) seconds ensures that most of the forwarded traffic will be acknowledged by the service consumer.

As far as the cryptographic parameters are concerned, 1024-bit RSA and 160-bit ECC keys have been used.

D. Results

The results of the four sets of experiments described in section IV.B are presented in Fig. 3. The "Reference" curve represents the pure bandwidth achieved without the use of the Linksys router and without any P2PWNC protocol operations.

As shown in the figure, NAT operation that has the most important impact on the achieved throughput. The additional overhead due to the P2PWNC protocol operations is minimal, given the selected RREQ interval and number of concurrent P2PWNC sessions. However, as the performance analysis in [15][10] has shown, ECDSA verifications are CPU intensive and the Linksys router is capable of performing approximately 8.7 160-bit ECDSA verifications per second when the router is idle. A large number of simultaneous P2PWNC sessions may result in more verification requests than the AP can handle in the given RREQ interval. Thus, reducing the RREQ interval limits the maximum number of sustainable P2PWNC sessions. As it seems, though, the value of five (5) seconds proves a conservative choice, assuming a typical WCN access point.

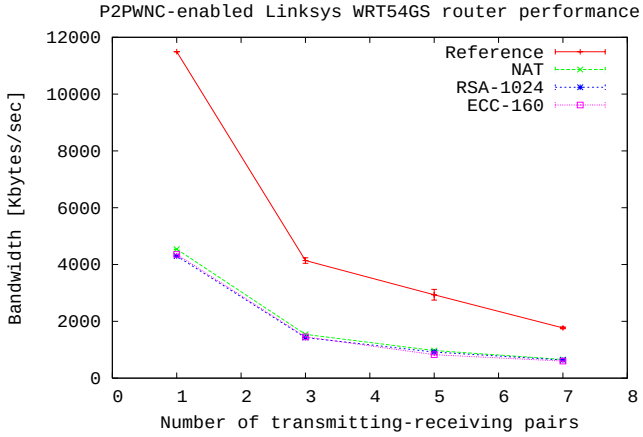

Fig. 3. Routing performance of the P2PWNC-enabled Linksys WRT54GS wireless router. Each of the curves represents one set of experiments. Each data point refers to the throughput achieved by a host in the presence of a number of simultaneous transmitting hosts.

V. DEPLOYMENT ISSUES

Having presented the design of the P2PWNC architecture, in this section we demonstrate cases for the deployment of the scheme in metropolitan areas. Our purpose is to show that the scheme we have designed is directly applicable to existing infrastructure and can

enhance the provision of services that, as of today, are limited or practically non-existent.

We begin by observing that in WCNs that exist in metropolitan areas mobility of their members is, in general, not considered. A WCN participant cannot access the WCN when mobile, even if this service would come at no cost to the providing WCN node (assuming that what is costly is access to the Internet other that to the WCN).

On the other hand residential WLAN owners that do not participate in WCNs do not share their Internet connections with passersby, raising security concerns and lack of the proper incentives.

We believe that the system that we present can provide an environment for unified access to the resources of WCNs and individual WLAN owners by mobile users. This environment also gives the opportunity for the emergence of differentially charged classes of services. We can distinguish between two basic such classes:

1) "Low-cost" access to the WCN.

2) "Expensive" access to the Internet.

To access the WCN, a roamer needs to visit one of the WCN-controlled WLAN APs. In such a case, the user may, for example, access services provided internally by the WCN, such as free VoIP, without being routed to the Internet.

In the case of a visited AP that has a connection to the Internet (either through the WCN or via its own broadband connection) the mobile user has access to more expensive Internet services.

Making only minor modifications to the existent P2PWNC accounting scheme, the above charging model can be easily implemented. Namely, the same protocols, receipt-based accounting and reciprocity algorithms can be maintained. However, the APs will be equipped with additional charging modules to distinguish between the available service classes and manipulate the "weight" field of the P2PWNC receipts.

Deploying the P2PWNC architecture in this environment is rather straightforward. As far as WCN nodes are concerned, apart from the backbone point-to-point links, they need to operate public APs to act as points of attachment to the WCN for mobile users. On the other hand, residential WLAN owners that do not belong to WCNs can simply run the P2PWNC software on their private APs.

VI. Conclusion

In this paper, we presented the design of a WLAN aggregation scheme suitable for Wireless Communities, guided by the principles of reciprocity and self-organization. We believe that it is straightforward for the scheme to be deployed over existing infrastructure to offer WLAN roaming capabilities in citywide areas, where wireless coverage is abundant. The P2PWNC scheme is designed to provide participating micro-operators incentives for cooperation so that free and ubiquitous Internet access can be achieved in metropolitan areas. If such a service were commonplace, a low-cost substitute to 2G/3G cellular services would be a reality. Finally, we evaluated a user's cost in terms of resources that have to be provided for participation in a P2PWNC enabled WCN using low cost WiFi equipment.

References

[1] Seattle Wireless, http://www.seattlewireless.net

[2] NYCwireless, http://www.nycwireless.net

[3] Athens Wireless Metropolitan Network, http://www.awmn.net

[4] N. Ben Salem, J.-P. Hubaux, and M. Jakobsson, "Reputation-based Wi-Fi deployment," *Mobile Computing and Communications Review* (MC2R), July 2005.

[5] Speakeasy WiFi NetShare Service, http://www.speakeasy.net/netshare/

[6] FON, http://en.fon.com

[7] Linspot, http://www.linspot.com/businessmodel.html

[8] QTEK 9100 Pocket PC Phone ed., WLAN-enabled, http://www.qtek.nu/europe/products/9100.aspx

[9] Wireless Philadelphia Executive Committee, http://www.phila.gov/wireless

[10] E.C. Efstathiou, P.A. Frangoudis, and G.C. Polyzos, "Stimulating Participation in Wireless Community Networks," In Proc. of 25th IEEE Conference on Computer Communications (INFOCOM 2006), Barcelona, Spain, April 23-29, 2006.

[11] Linksys Wireless-G broadband router, http://www.linksys.com

[12] A.V. Goldberg and R.E. Tarjan, "A new approach to the maximum-flow problem," *Journal of the ACM*, vol. 35, no. 4, pp. 921-940, 1988.

[13] É. Tardos and K.D. Wayne, "Simple generalized maximum flow algorithms," In Proc. 6th International Conference on Integer Programming and Combinatorial Optimization, pp. 310-324, 1998.

[14] The P2PWNC project website, http://mm.aueb.gr/research/P2PWNC/

[15] P.A. Frangoudis, "The Peer-to-Peer Wireless Network Confederation Protocol: Design Specification and Performance Analysis," Master's Thesis, Athens University of Economics and Business, 2005, http://mm.aueb.gr/technicalreports/2005-MMLAB-TR-02.pdf.

[16] Hierarchical Token Bucket packet scheduler, http://luxik.cdi.cz/~devik/qos/htb/

[17] Iproute2, http://linux-net.osdl.org/index.php/Iproute2

[18] OpenWRT Linux Distribution, http://openwrt.org

[19] The ttcp tool, http://ftp.arl.mil/ftp/pub/ttcp/