Managing Wireless HotSpots: The Uni-Fy Approach

M. Brunato, D. Severina, R. Lo Cigno
Dipartimento di Informatica e Telecomunicazioni
Università di Trento — Italy

Abstract— **This paper presents and discusses user, account, and traffic management in a WLAN based HotSpot. The HotSpot is a large WLAN infrastructure at the Faculty of Science of the University of Trento, that uses a novel access management architecture named Uni-Fy developed at the University of Trento, and where roughly 20.000 users can have free access.**

The management scheme, named Uni-Fy, is based on the concept of Open Access Networks and manages the users access, privileges and traffic based on the users profiles and authentication provided by remote service providers, hence acting as an enabling factor rather than as an autonomous service itself.

Finally we provide some measurements and experience based on nearly one year of operation.

I. Introduction

Network management problems cover topics ranging from failure recovery, to service provisioning, to accounting, to security and privacy enforcing. The management of users and user access is definitely one of the topics that is drawing interest recently, specially with the widespread advent of wireless networks. In wired (telephone) networks the users were recognized, managed, accounted, etc. based on the physical location they were, unequivocally identified by the access wire. Cellular networks introduced the necessity of user authentication based on secret keys: as the GSM experience has shown, hardware identification is no longer enough to properly authenticate users and manage their services.

The recent explosion of 802.11 Wireless LANs (WLAN for short) is changing many of the game rules defined by cellular networks. WLANs are used in several different contexts. In private organizations they are used to provide nomadic access to employees, but they are also used to provide complimentary Internet access to visitors maintaining service differentiation with respect to employees. In public HotSpots (e.g., airports, stations, malls) WLANs are used to support Internet access to business people and to generic nomadic users that require connectivity. WLANs do not provide universal coverage, and WLANs providers are thousands around the world, contrasted to a few providers per country in cellular networks. Cellular networks define a limited number of communication services (often only one or two, e.g., telephone and packet based data access—GPRS) that require subscribing for their use. WLANs are not even a communication service *per-se*, they are rather an access means to the magmatic world of Internet services that users access on-demand based on needs.

This work has been supported by the Italian Ministry of Education and Research (MIUR) under the TWELVE PRIN project

The management of users accessing WLANS requires an architecture that takes into account the following features:

- The WLAN is an access network, normally used as a means to reach another communication system (the Internet, a Virtual Private Networks (VPN), etc.);
- There is no fixed binding between a user and the terminal it uses;
- The customer is normally not known in advance to the WLAN provider (e.g., a traveler in an airport may be using the local HotSpot for the first —and last— time);
- WLANS can be dynamic, ad-hoc networks setup for short time periods, their technology is evolving quickly and management solutions should be technology independent;
- Customers accessing communication services are known to some service provider somewhere (e.g., their GSM provider, the VPN manager of a global corporation);
- The customer privacy and the network security must be enforced actively to avoid improper use of resources and informations;
- The management system should require the least possible (ideally none) installation of ad-hoc code and applications on the clients.

This paper presents a tool, named Uni-Fy[1], which has been developed at the University of Trento as a means for managing WLAN HotSpots taking into account the features discussed above, and inspired to the principle of Open Access Networks (OAN) [4], [11]. We discuss the general requirements, describe a prototype implementation, and present the experience and feedback we got in nearly one year of its use.

II. Background and state of the art

Management of public HotSpots is a new topic, so there is little literature and experience on the field. Many HotSpots behave as ISPs or work on behalf of an ISP (e.g., Boingo HotSpots or Starbucks complimentary WLAN access) with little or no user management at all. However, some related work can be found.

Pioneering the field of OAN management has been the StockholmOpen project [18] carried out jointly by KTH and Stockholm University in Sweden. The StockholmOpen.net project consists of a WAN where wired and wireless access points are connected: the first ones are deployed in homes, while the second ones are in public places [11], [12]. The structure of this network allows the coexistence of different ISPs, but each provider must connect its own gateway to the OAN infrastructure to authenticate its users, and to provide the access to the global network. While this project was not focused on HotSpots, the OAN principle is indeed extremely interesting for HotSpot management. Other solutions based upon ideas of StockholmOpen.net are implemented also in other cities in Europe, North America and Oceania (see [22] for more information).

In general applying the ideas of OAN to a HotSpot does not mean free access or no-control access: it allows nomadic users to access to remote resources after an authentication. The philosophy is granting access to a new users without a prior contract, but with a strong authentication based on a remote authenticator. The most used authentication technique is the captive portal [6] solution: when users request a first web page, they are redirect to a portal, where they can choose the preferred Service Provider (SP) to be authenticated.

Most of the solutions to grant access in HotSpots are based on this technique and often are open-source and free software:

- **WifiDog** [20] has optional centralized access control, full bandwidth accounting, node existence control and local content specific to each HotSpot;
- **Nocat** [7] is written in Perl and was the pioneer of captive portal solution; it started as a community-supported 802.11b WLAN in Sonoma County, CA.

Recently, commercial solutions based on the same philosophy appeared on the market, like FirstSpot by Pantronsoft (a Windows-based manager).

Usually, captive portal solutions run on a dedicated PC behind the physical access network. Recently, "HotSpot-in-a-box" solutions appeared on the market: the APs provide both physical connectivity to the backbone and the authentication of the users. In general the captive portal solutions are not fully fledged user management systems, and the HotSpot-in-a-box solutions are definitely not scalable, nor can they provide a transparent access to remote ISPs. A solution with modified firmware in the AP is the OpenWRT project [16]. With the open firmware for wireless router (Linksys devices), an implementation of access control in the AP can be managed, but binding users management to the APs is definitely a non-scalable, architecturally questionable solution.

To conclude this quick overview of HotSpot management techniques, it is necessary to mention standard (or de-facto standard) components that can be used in WLAN management, but are sometimes confused as if they were entire management systems.

Some APs offer the possibility of building over the same BSS (Basic Service Set) more than one logical network, separating users that set different SSID (Service Set Identifier) on their interfaces. Different SPs can use the same physical infrastructure by associating their users to different SSID. The solution is clearly non-scalable, and does not provide any means for the management of resources or users.

Furthermore there are some architecture systems that doesn't implement an authentication or authorization system, but are related to the user authentication to access web-based resources.

Shibboleth [17] is an architecture that enables organizations to manage a network that allows users to access web resources. The architecture of Shibboleth defines how the informations must be exchanged between an organization and a provider of digital resources. All the organizations that use this system must previously joint a federation.

Athens [14] is another access management system to control access to remote resources and services. This management system allow access to protected resources with authentication based on Shibboleth.

More interesting as management components are the 802.1x [1] standard and the work done in 802.11i [3] Task Group.

802.1x defines a number of techniques for user au-

[1]Uni-Fy has been developed with the support of the Autonomous Province of Trento under the WILMA project [21], [23], and is currently maintained and evolved under the TWELVE PRIN project [19]

thentication (based on EAP, PEAP, TLS, TTLS, etc.) and for the implementation of secure communications (based on tunneling: PPTP, PPoE, L2TP, ...). Many of these solutions can be embedded in a HotSpot management system, and the Uni-Fy we describe in this work can include some of them if required.

802.11i TG is working on the enhancement of security of the wireless medium by ameliorating the 802.11 MAC protocol, which is more than welcome in public HotSpots, where privacy and security are major concerns. It must be noted however, that privacy and security must be sought for, and enforced at the application level, where the knowledge of the data semantics allows taking the appropriate decisions and counter-measures, while trying to enforce security at the physical or MAC layer is an additional help, but not a definitive solution, as WEP [2] has shown in recent years [8].

III. System philosophy and architecture

The principle underlying the Uni-Fy system is the decoupling of the access network management from the service management, while preserving an access scheme and mechanism as easy as possible.

There are currently two basic alternatives to access to the network. The first one can be summarized as buy-it-all-in-one: the user gets all services (home connectivity, e-mail, roaming access, etc.) from the same provider, and normally logs to the network with a single procedure. The second one is far less widespread and consists in trying to buy the best possible service, which normally means buying different services from different providers. In this latter case, the log-in procedure can be annoying, since providers do not coordinate and require separate authentication and log-in procedures. For instance a user buying connectivity from one provider, but having the e-mail account with another one, and hosting his web home page with a third one, will be asked three different log-in procedures: a first one to have connectivity, a second one to read and send mails, and a third one to modify the web pages.

In case of HotSpots, there are two fundamental possibilities: either the HotSpot is an "extension" of a normal Internet provider, in which case the access can be managed similarly to home connectivity, or HotSpots are managed by third entities (normally the owner or keeper of the local facility), in which case the HotSpot represent an additional step in getting the service.

We deem the latter case the most interesting, because it is not conceivable that every Internet provider will cover every place where an HotSpot can be installed. In this case the access network in itself is not a complete "service," but rather an intermediary that allows a roaming user to be connected to his own service provider(s).

In this scenario, the goal of the Uni-Fy system is to grant access to users that can be authorized based on some remote service provider procedure without requiring any special or manual machine configuration, nor an explicit "commercial agreement" with the HotSpot provider. Instead, the Uni-Fy system will try to have commercial relationships with the service providers, so that the revenues generated by the service can be correctly split between the service and the HotSpot provider. Notice that we can expect to have hundreds of service providers and thousands (maybe tens of thousands) of HotSpot providers around the world, but hundreds of millions nomadic users, so that moving the burden of the mutual trust to build a commercial relationship from the user-HotSpot couple to the service provider-HotSpot couple represent a drastic simplification of the problem.

The high level architecture of the Uni-Fy system is shown in Fig. 1. The Gateway is a dynamic firewall that regulates what traffic can enter and exit the HotSpot. Packets generated within the HotSpot, but not belonging to already authorized users, are 'captured' and forwarded to the Gatekeeper, which implements the "intelligent" part of the system.

The operations the Gatekeeper does on captured packets can be classified (with some approximations for the sake of simplicity) in the following three categories:

- Packets that belong to DHCP requests are managed locally or remotely: if the packets belong to an initial request, a temporary IP address is assigned to the querying machine to allow authentication, all other DHCP packets are forwarded to the proper DHCP server;
- Packets that belong to an initial service phase are managed to allow prior authentication and commercial agreement (dashed channels in Fig. 1). To clarify the idea, if the service is "web browsing," the overall behavior is similar to a captive portal: the first HTTP request is re-directed to an authentication server that handles the authentication;
- All other packets are discarded on the assumption that they are attempts of illegal use; if it is required (e.g., by the amount of illegal traffic), the Gatekeeper can instruct the Gateway or the APs to take further steps (e.g., an AP can de-associate the machine generating illegal traffic).

The Gatekeeper also maintains secure connectivity with all the ISPs that want to use the HotSpot resources. Two features of the Uni-Fy are distinctive and interesting: i) the connectivity is at the logical level, and does not require the ISP to install any hardware and/or servers

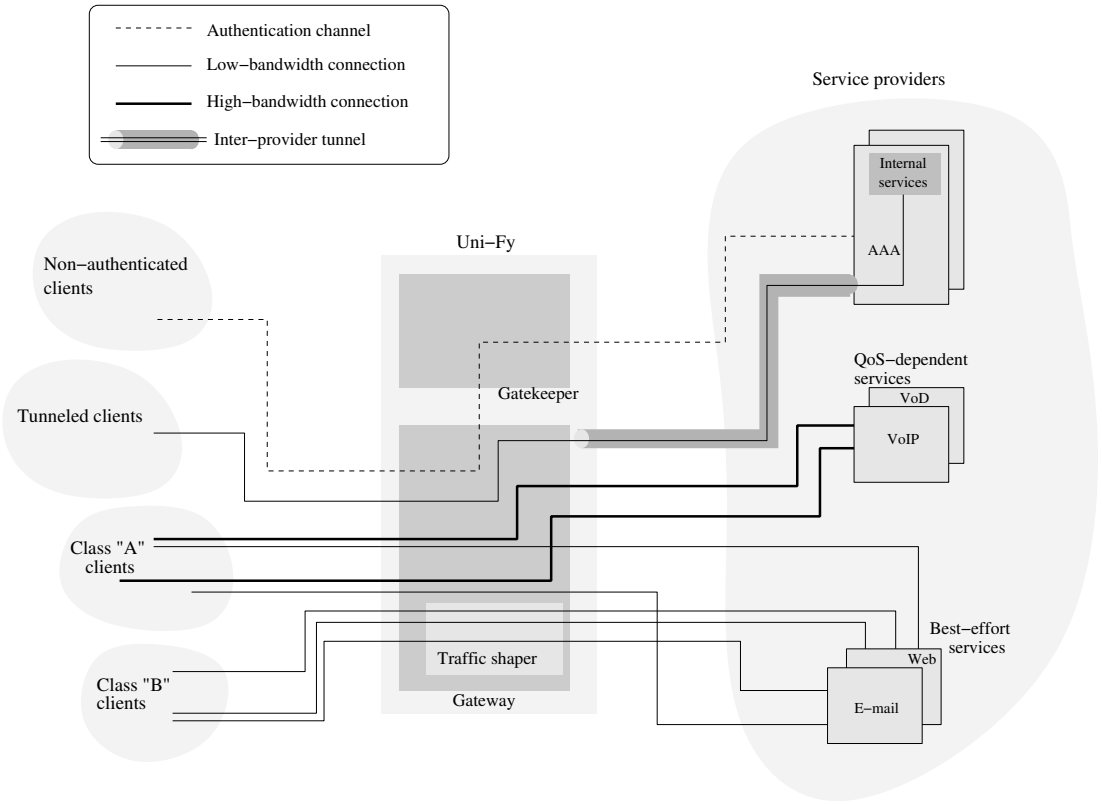

Fig. 1. High level architecture of the Uni-Fy system

within the HotSpot; and ii) the system can be entirely distributed allowing the HotSpot, e.g., via the intermediation of clearinghouses, to provide authenticated and secure service to users of ISPs that are not known *a-priori* to the HotSpot itself.

Within this architecture, many classes of clients can be managed by the same architecture, and Fig. 1 shows three possible cases:

- client traffic can be "tunneled" to the Service Provider (usually, the same that performed authentication for that client) in order to let the user access private resources; the tunneling method can be either a VPN or an ad-hoc connection between the Uni-Fy and the AAA provider's gateway;
- some clients may be allowed full access to external services (see *Class "A" clients* in Fig. 1), some of which may have strict QoS requirements;
- other clients (*Class "B" clients*) are processed through additional modules for traffic shaping and firewalling in order to limit their bandwidth, so that only best effort services can be accessed.

IV. Uni-Fy implementation

The Uni-Fy system is currently implemented in C++ as applications running in user-space. While focused on Linux, coding has been designed in order to be portable on systems with a reasonably large subset of UNIX APIs. Of course, machine-specific optimizations, such as the implementation as kernel-space modules, are required in order to enhance performance of critical operations for carrier grade service. The present implementation is a proof-of-concept to demonstrate the feasibility of the architecture through extensive operation as described in Section V. Additional details on the Uni-Fy implementation and possible configuration can be found in [5].

As shown in Figure 2, the application is divided into two components, called "Gateway" and "Gatekeeper"; each component is composed of several modules (C++ objects). Performance and cost are the base to decide whether the two components can be embedded in the same machine or must be executed by two different computers.

All network-intensive functions (firewalling, routing, policy enforcement) are performed by the Gateway component, which is basically a router with some ad-hoc functionalities; any configurable router or a firewall with enough flexibility can be used. CPU-intensive functions (such as table lookups, DHCP, protocol redirections) are located within the Gatekeeper component, which routinely receives all unauthorized packets from the Gateway and, after processing them, launches the events that lead to an authentication procedure. The role of

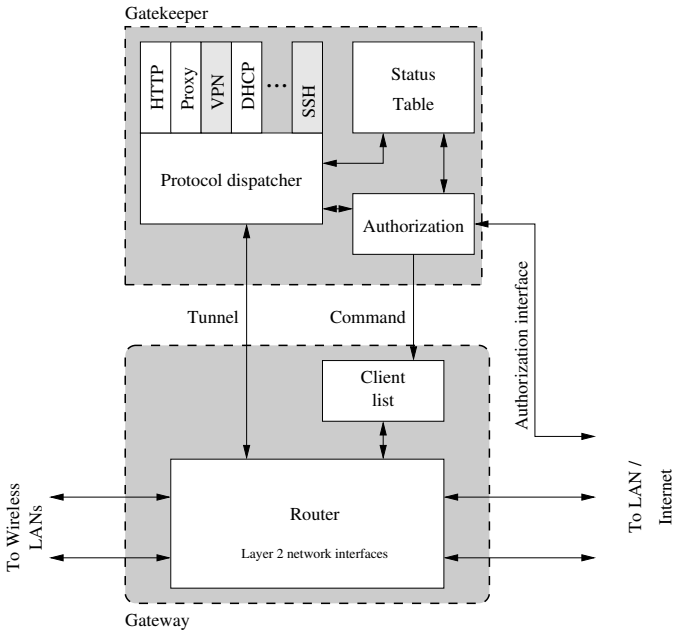

Fig. 2. Block diagram of the Uni-Fy system

components and their interactions depend on the authentication mechanism.

A. The Gateway component

The machine running the Gateway program contains at least two network interfaces and operates as a configurable Layer-3 switch.

Its core component is the *router* module, which manages the interchange of Ethernet frames among network interfaces and the Gatekeeper (which can be seen through an internal interface or be connected remotely through an SSL tunnel or an additional physical interface). The router module identifies packet sources and destinations according to their Ethernet MAC address and their IP address. Correspondence between MAC and IP addresses for authorized clients is stored in the client list, accessed by the router via a read-only interface.

Routing decisions are taken according to an internal firewalling rule table. Every rule mandates two possible actions, the first to be taken if the (IP,MAC) pair of the packet is authorized, the second in the opposite case. Authorization of a packet is checked against a client list, which is fundamentally a list of authorized (IP,MAC) address pairs completed with other relevant information such as time to live and, in the future, traffic class or other special permissions.

Three types of firewalling actions can be found: "drop the packet", "forward the packet", "send the packet to the Gatekeeper" for further processing. In principle, all packets coming from wireless clients whose status is unauthorized are sent to the Gatekeeper for better inspection, while authorized packets are forwarded. Some notable exceptions are DNS queries, which are always forwarded, and DHCP requests, which are always rerouted to the Gatekeeper which implements both a DHCP relay for usage with an external DHCP server and a simple DHCP server for standalone use.

Firewalling rules do not apply to packets received from the Gatekeeper, which is always a trusted party.

The client list is kept as simple as possible in order to avoid computational bottlenecks, in particular when the software is executed on diskless dedicated hardware with a limited amount of computational power. The client list is updated by the Gatekeeper via a set of commands sent through a secure TCP connection.

B. The Gatekeeper component

The Gatekeeper component performs all tasks that require more processing than just looking at frame and packet headers; namely client status maintenance, DHCP management and client authorization based on information received from remote trusted authentication providers.

The Gatekeeper receives packets from the Gateway through the "tunnel" shown in Figure 2, implemented as a pair of UDP secure sockets. Packets are processed with the goal of authenticating the user. After receiving a packet from the Gateway, the Gatekeeper replies with a sequence of response packets that are forwarded by the Gateway to the desired destination. Packets of different protocols are managed by different "plug-in modules" in an easily extensible architecture.

In addition to the Gateway tunnel, Gatekeeper interacts with remote entities via the *Authorization* module, used to receive information by a trusted authentication server about authorization of users. Another module, not shown for simplicity, allows the Gatekeeper to report its status to trusted parties.

The Gatekeeper module has the ultimate responsibility of moving a client to the authorized status, granting full access to the network, and revoking it upon explicit logout or when the authorization period expires without renewal. The Gatekeeper's *status table* is a superset of the Gateway's client list, and contains information about a user's identity, DHCP status and other relevant data.

C. Client Authentication

When a client first connects to a Uni-Fy-served access point, after issuing a DHCP request it is assigned an IP address, and its status is recorded as "known but unauthorized." Moving to an authorized (and therefore

fully functional) status is the purpose of subsequent packet exchanges.

In order to obtain the authorization, the client contacts his own trusted authentication server and exchanges all relevant information about his identity. Depending on the authentication protocol, exchanged information includes login and password, cryptographic challenges, secure connection setup or other techniques. Because of the end-to-end character of such techniques, the Uni-Fy itself will never acquire any sensitive information during this phase and shall be transparent to authentication protocols. When the user is authenticated, the authentication provider sends the appropriate confirmation message to the Gatekeeper, which authorizes the client for the specified amount of time and the selected services with their QoS. Since users are recognized by their MAC-IP address pair, exposing the system to spoofing attacks, authentication must be renewed (transparently or with the user's collaboration) periodically, and users are requested, when possible, to perform an explicit logout from the system.

Two comments are in order on this authentication procedure.

1) During the unauthorized phase, techniques as segregation and bandwidth shaping cab be applied to avoid security attacks.
2) The privacy of users is preserved, since Uni-Fy does (and can) not collect personal information and access in granted on a pseudonym basis. Nevertheless, if public security requires it, the user identity can be retrieved by binding the local pseudonym to the remote authenticator user identity.

Renewal (or "keep-alive") operations can be based on any stateful protocol, either requiring a small piece of software to run in the client or based an standard applications as in the "captive portal" example described below.

1) The "captive portal" example: Currently, the implemented authentication technique mostly used in Uni-Fy installations is a variant of the common "captive portal" method via web pages. As it can be seen on Figure 3, after the initial DHCP phase the client's URL request, either direct or mediated by a system proxy server, is intercepted by the Gatekeeper, which redirects the user's browser to a local page which enables the user to choose his preferred authentication provider.

Interaction of clients with providers is carried out by a secure HTTP connection and ends, if successful, with an authorization notification to the Gatekeeper. The normal navigation session proceeds until an explicit logout by the user, or the expiration time is reached.

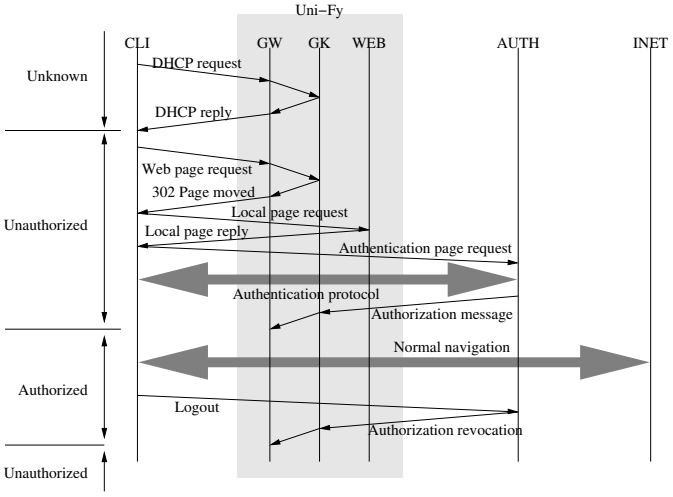

Fig. 3. Timeline of a typical user session. On the left, the status of the client as recorded in the Gatekeeper's status table. The client shall eventually fall back to the "unknown" status when the DHCP lease expires.

Authorization renewal is automatically managed by a pop-up browser window which is refreshed with a specific periodicity. The window connects via HTTPS to the authentication provider and its aim is to prove that the address has not been spoofed in the meantime.

2) A WLAN/3G based example: Some partners of TWELVE Project developed an authentication procedure using WLAN/3G secure authentication based on SIP [9], [13]. The solution shows both the possible development of Uni-Fy system and the integration of 3G and WLAN to grant access to web services to users.

The authentication mechanism is based on authentication agents running on the client device and able to perform authentication procedures based on SIP. The authentication procedure can use user-id/password or stronger mechanism, e.g., based on secret keys stored in user USIM (the UMTS Subscriber Identity Module). After the association of the client to the access point, the client starts the SIP registration and the authentication procedure towards its SIP server. The Uni-Fy gate allows communications with authenticated SIP servers.

3) Small devices support: Small clients, such as PDAs, are not fully compatible with the authentication system described above; in particular, they may lack the ability of opening pop-up windows. Most operating systems for PDAs only support one window per program, so no automated renewal is possible and the user would be required to cooperate for renewal every few minutes.

While waiting for smarter PDA systems, the problem can be solved in different ways:

- bypassing the renewal procedure by semi-statically (e.g., 4 hours) inserting the (MAC, IP) address pair

in the Uni-Fy configuration parameters;
- creating an ad-hoc authentication procedure based on some resident software that takes care of renewals.

The first solution clearly reduces the system's security, because it is open to simple MAC-IP spoofing attack. The latter solution somewhat is contrary to the project philosophy as discussed in Section III, so pros and cons must be carefully weighed.

V. Experience and results

Network management proof-of-concepts must be supported by operations and measurements. Our implementation of the Uni-Fy system has been in operation for the management of the WLAN at the Faculty of Science of the University of Trento (FS-WLAN for short) for close to one year. More recently it has been installed for a few months management of a Public HotSpot in downtown Trento infrastructured by Alpikom S.p.A within the framework of the WILMA project [21], [23] (WILMA-HS for short), and it will be used as a common platform for the demonstration of QoS HotSpot management on a nationwide Italian experimentation form Palermo to Trento within the framework of the TWELVE PRIN project (see Sect. V-A).

The FS-WLAN comprises more than 30 Access Points whose traffic is collected on a 802.1q VLAN: the Uni-Fy bridges the VLAN with the rest of the University Campus LAN. Acting as authentication providers are the University Administration, so that all students and staff of the University (close to 20.000 people) are automatically recognized by the system, and a "WILMA project" authenticator, allowing people external to the University, but related to the WILMA project that originally spawned the work, to receive service.

WILMA-HS covered several streets and parks in the city, with 11 access points. It recognized as authentication providers the same two of the FS-WLAN plus Alpikom and the Trento Civic Library.

Alpikom has used the WILMA-HS to offer (free of charge) the WiFi connectivity in Trento at all the registered users to its free dial-up access service AkFree. Moreover when an user is connecting for the first time to the WILMA-HS, was offered the possibility to register directly by the WILMA-HS to the AkFree service filling a standard registration form. The user receives his new login and password in few minutes and he can start using the WILMA-HS immediately.

The Trento Civic Library can integrate in its own authentication data-base all the users of Trentino Public Library System, which potentially can include all the population of the Trentino Province (roughly half a million people). Unfortunately due to recent legislation changes on SPs in Italy the experiments on WILMA-HS HotSpot was stopped.

Fig. 4 shows the number of users and the number of accesses that the Uni-Fy system at FS-WLAN supported during its first life-time period. The initial growing of the number of users cannot be seen, because before this authentication system there was an access procedure based on RADIUS authentication with MAC address registration, and the WLAN HotSpot was already in use for more than one year. However, after the summer vacations the traffic (and user) volume increased by roughly 50%, testifying the popularity and acceptance of the system. The amount of traffic is related to weekday and holiday, besides it is correlated to the lesson-time and exam-time: most of the users are students. The Uni-Fy handled more than 40.000 accesses during a period of nine month, with a peak of 433 access and 136 different users in a single day.

In Fig. 5 we can see the use of the network considering the division of the users between student and other people that are related to University (professors, technical staffs, visitors, ...) The graph is made using the data collected by the authentication system from January 1^{st} to December 31 2005. The left part of the graph shows the percentage of the users per week and the right part shows the percentage of the accesses per week. The comparison of the two graph shows that all the users, student or not-student, use the network in the same way in terms of number access per week. We have not been able to collect the volume of traffic per access to measure potential different per-access behaviour of the two groups of users.

In Fig. 6 we can see the distribution of the flow length measured in the period january–august 2005. Very short flows of less than 500 bytes dominate the lot, specially for outgoing traffic. Flows with data length greater than 500 byte show a linear trend, typical of heavy tailed distributions. The peak at the end of the graph considers all the flow with a length greater than 500 kByte. This simple graph shows that HotSpot traffic is not different from a standard wired LAN (see [10] for a comparison), a feature that must be taken into account for HotSpot design management.

A. TWELVE activities

The TWELVE PRIN Project partners work both in fundamental research and in extensive campaigns of experiments and demonstrations with the aim of disseminating mature results with major practical impact.

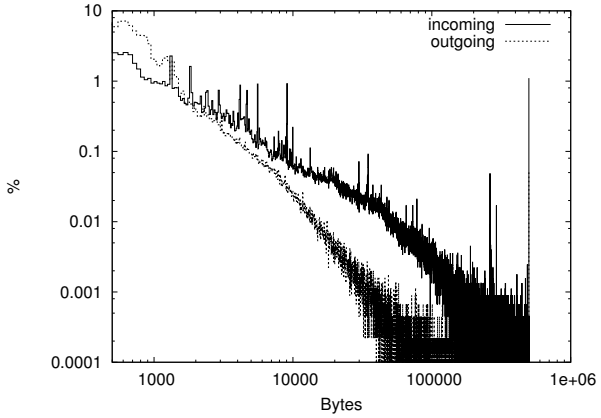

Fig. 6. Incoming and Outgoing TCP flow length measured at the Gateway

The overall organization of such experimental activities relies on a nationwide testbed of HotSpots, unified by a framework for users and service management that we call UniWireless.

UniWireless is a collection of coordinated HotSpots, all managed by the Uni-Fy gateway provided by the University of Trento. Such HotSpots are coordinated in order to enable presentations and demonstrations of innovative services, algorithms, protocols and management techniques developed within the project. Further information about TWELVE testbed can be found in [9].

VI. Conclusions

In this paper we have described the structure of an authentication system to manage wireless HotSpots. This system, called Uni-Fy, is based on the Open Access Network philosophy. It is used for access management and accounting in WLANs, but it can also be used for LANs where the users access with wired connections. Uni-Fy manages the authentication by interacting with remote authentication servers that may use different authentication protocols (LDAP, RADIUS, ...).

Furthermore, we have presented results about our running implementation of Uni-Fy in our Faculty buildings, whose network consists of more than 20 access points, where the authentication system has been running for more than one year.

Acknowledgments

As authors of this paper we wish to thank prof. Roberto Battiti, coordinator of the WILMA project, Dott. Alessandro Villani, for the technical support in the development of the authentication system and infrastructure network, and all the people that are working in the TWELVE project to extend and improve the features of the HotSpot management system.

References

[1] IEEE Std 802.1X: Standard for Local and metropolitan area network – Port-Based Network Access Control. IEEE, 2001 (Revision 2004).

[2] IEEE 802.11: Standard for local and metropolitan area networks - Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. IEEE, 1999 (Last revision 2003).

[3] IEEE Std 802.11i: Standard for Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks. IEEE, 2004.

[4] R. Battiti, R. Lo Cigno, M. Sabel, F. Orava, and B. Pehrson. Wireless LANs: from warchalking to open access networks. *Mobile Networks and Applications*, Vol. 10:275–287, 2005.

[5] M. Brunato, D. Severina. WilmaGate: a New Open Access Gateway for Hotspot Management. *Proceedings of WMASH2005*, 56–64, 2005.

[6] Rob Flickenger. Wireless Hack. O'Reilly, 2003, Chapter 7.

[7] Rob Flickenger. Building Wireless Community Networks. O'Reilly, 2003.

[8] S. Fluhrer, I. Mantin, A. Shamir. Weaknesses in the Key Scheduling Algorithm of RC4. *8th Annual International Workshop on Selected Areas in Cryptography, SAC 2001*, Toronto, Ontario, Canada, August 16-17, 2001.

[9] R. Lo Cigno, V. Ammirata, M. Brunato, D. Di Sorte, M. Femminella, R.G. Garroppo, D. Giustiniano, A. Ordine, G. Reali, S. Salsano, D. Severina, I. Tinnirello, and L. Veltri. TWELVE Test Bed and Demonstration Planning. *Network Workshop 2006*, Courmayeur, 11-13 January, 2006.

[10] M. Mellia, R. Lo Cigno, F. Neri, "Measuring IP and TCP behavior on edge nodes with Tstat," *Computer Networks*, Vol. 47, No. 1, pp. 1–21, Jan. 2005, Elsevier Science

[11] B. Pehrson, K. Lundgren, and L. Ramfelt. Open.Net - open operator neutral access network. In *12-th IEEE workshop on Local and Metropolitan Area Networks*, Stockholm, SE, 2002.

[12] E. Pelletta, F. Lilieblad, M. Hedenfalk, and B. Pehrson. The design and implementation of an operator neutral open wireless access network at the Kista it-university. *12-th IEEE Workshop on Local and Metropolitan Area Networks*, 2002.

[13] S. Salsano, G. Martinello, and L. Veltri. WLAN/3G secure authentication based on SIP. *Network Workshop 2006*, Courmayeur, 11-13 January, 2006.

[14] Athens. `http://www.athensams.net/`

[15] NoCat Network. `http://nocat.net/`

[16] OpenWRT. `http://nocat.net/`

[17] Shibboleth. `http://shibboleth.internet2.edu/`

[18] StockholmOpen Project. `http://www.stockholmopen.net/`

[19] TWELVE Project. `http://twelve.unitn.it/`

[20] Wifidog. `http://dev.wifidog.org/`

[21] WILMA Project. `http://www.wilmaproject.org/`

[22] Wigle Homepage. `http://wigle.net/`

[23] WilmaGate download page. `http://netmob.unitn.it/wilmagate.html`

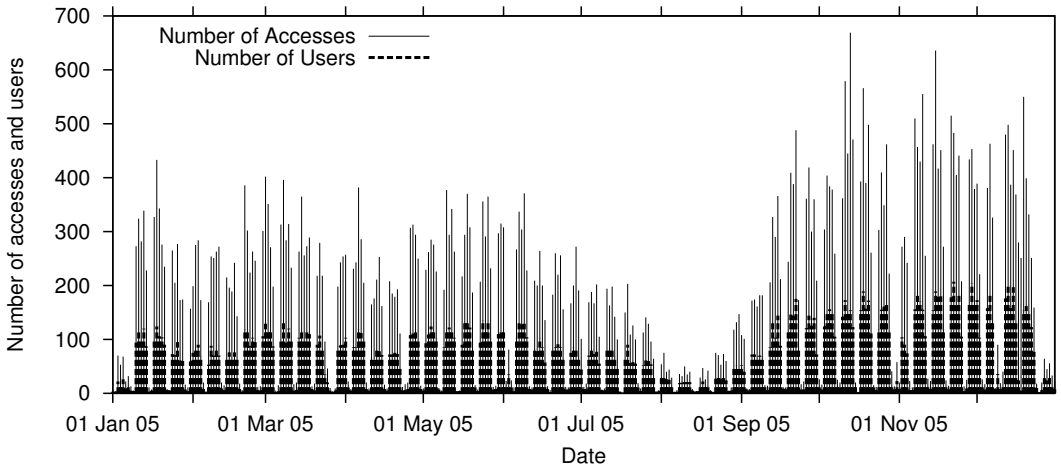

Fig. 4. Number of users and accesses during the nearly one year of operation at the Faculty of Science of the University of Trento

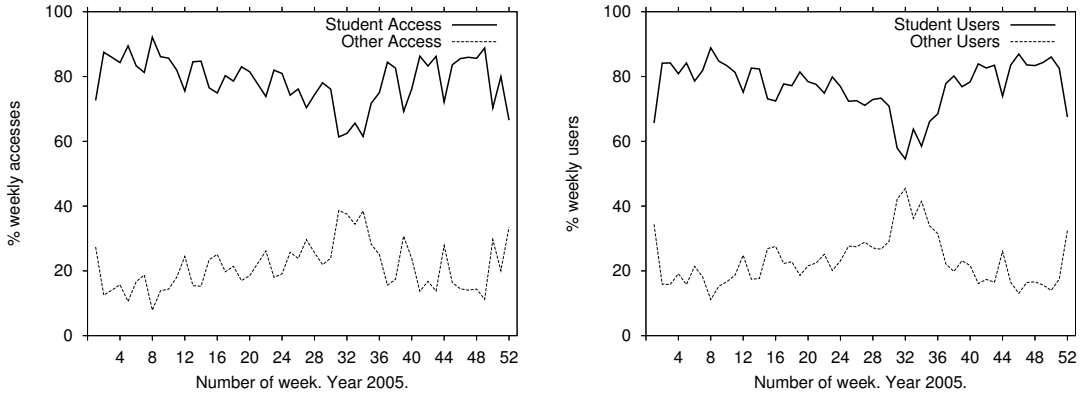

Fig. 5. Number of users and accesses during the nearly one year of operation at the Faculty of Science of the University of Trento