A Secured Delegation of Remote Services on IPv6 Home Networks

Stere Preda, Laurent Toutain,
Nora Cuppens-Boulahia, Frédéric Cuppens
IT TELECOM Bretagne
CS 17607, 35576 Cesson-Sévigné, France
given_name.surname@telecom-bretagne.eu

Joaquin Garcia-Alfaro
School of Computer Science
Carleton University, K1S 5B6,
Ottawa, Ontario, Canada
joaquin.garcia-alfaro@acm.org

***Abstract*—IPv6 is an attractive technology for innovative services such as health care monitoring, alarm systems, peer to peer applications, virtual machine systems and so on. The generalization of end to end paradigm, possible due to the length of IPv6 addresses, eases the deployment of such services. Nevertheless end to end connection can be a threat since application can be easily accessible from outside and thus a compromised application may endanger others. In this paper, we study some of the advantages of using the IPv6 protocol in home networks but most particularly how to improve the security of home networks. We present an architecture allowing the definition of a partition between groups of applications and where communication between these groups is not permitted if there is no explicit delegation. We overview the key points of the current implementation and some initial results of our approach.**

Keywords—**Home Networking; IPv6; Network Security; Access Control.**

I. Introduction

Computer networks are challenging new domains of applications. Internet, after becoming the most popular interconnection protocol for office applications is entering more and more in the home network; nevertheless usages are very similar to an office (i.e., computer based environment). A wide sort of different devices — from traditional servers and personal computers to handhelds, cell phones, sensors, cameras — are currently getting interconnected through both wired and wireless networks. More electrical home appliances contain embedded minimal systems and network cards; their remote control over the Internet is becoming a current objective. The increase of broadband connections, moreover, makes easy the access to these devices from home to work networks or vice-versa. Remote services like health care, baby sitter's supervision, virtual machine systems, and so on, are today possible and very easy to deploy.

Before going further in this section, let us describe a home network as a local area network (LAN) mainly used for personal purposes and with the objective of connecting multiple devices (e.g., personal computers, laptops, cell phones, printers, web cameras, scanners, different sensors and other electrical appliances) within a domestic environment. This connectivity is typically used: (1) for the local exchange of information between these devices within the home network itself (e.g., for the exchange of documents between two or more computers, between a computer and a printer or for the indoor control of different electrical appliances); (2) for the exchange of information between these devices and the Internet.

To make possible this second option, a broadband connection supplied by a service provider is necessary, as well as a set of network devices, such as a broadband modem (if just a single device is supposed to be connected to the Internet) or a router (if multiple devices should get connected in an independent fashion) - generally called *home gateway* (HGTW).

The two most popular types of topologies we may find in home networking are both wired and wireless networks. In both of these types, it is normal to find a central device (i.e., a broadband router or modem connected to the phone line or to a cable Internet connection) which is in charge of directing the home network traffic between the connected devices and the Internet. Some other features, such as filtering of anomalous traffic or analysis of malicious messages, can also take place at this single point; or distributed within the home network as independent devices.

Regarding the network traffic, the common protocol used by most of these devices for the exchange of information is the IPv4 protocol. Even so, and although the use of IPv4 connections may still be dominating domestic connections for next years, it is clear that the promising features of the new version of the IPv6 protocol will finally impose the use of this protocol on home networking; though IPv6 indeed provides a simpler management of the home network (e.g., autoconfiguration) the existent IPv4 security threats will persist. With the generalization of end-to-end addresses, every equipment is visible from outside the home network. A filtering router blocking systematically incoming traffic will limit strongly the benefit of IPv6. Even if some flows are blocked, an attacker may degrade the service of an equipment visible from outside and then get down the home equipments. For example in a case of an alarm system connected to the home IP network, if a Personal Computer is attacked, the alarm system must not be compromised.

In this paper, we study some of the advantages of using the IPv6 protocol for home networking, but paying special attention to the security benefits that the use of IPv6 and

further auxiliary mechanisms (e.g., CGA - *Cryptographically Generated Addresses*) may bring about. We consider that the authentication is one of the main security requirements derived from current home network scenarios and thus we will insist on. We propose a mechanism to partition users and applications in home networks in order to enforce the service confinement aspect. We also overview the key points for the implementation of such a proposal for GNU/Linux systems, based on the manipulation of system calls and on the filtering of both IPv4 and IPv6 network traffic through a netfilter module.

The remaining of this paper has been organized as follows. Section II overviews the IPv6 basic principles. Section III discusses some security requirements derived from different scenarios in home networks. Section IV introduces our strategy and overviews the main aspects of our approach. Section V gives the current status of our implementation. Finally, Sections VI and VII give some related work and close the paper with some conclusions.

II. IPv6 Basic Principles

The new protocol designed by the IETF was a new version of the IP protocol, called IPv6 ([9]). IPv6 can be viewed as a simplification of the IPv4 protocol. IPv6 has been designed to transport information as fast as possible from one point in the network to another. Features that where not widely used in the IPv4 protocol have been removed. Address space has been considerably extended since the address sizes were increased by four. This removes almost all addressing constraints. Some network services are also being improved by IPv6 which offers support for Quality of Service (QoS), mobility and security.

An IPv6 address is divided into three parts. A Global Prefix GP (generally on 48 bits) is given by the access provider to the network. A Subnet ID is used to number internal topology, its size plus the GP size is now fixed to 64 bits. The remaining 64 bits used to number hosts on a link is called the Interface ID. The size of the IID is drastically over dimensioned compared to the numbering need on a link, which may generally contain about 50 nodes. The large size of the IID leads to good properties: (1) autoconfiguration can be done by using the Layer 2 address of the network card (48 bits for MAC addresses, and 64 for EUI-64), (2) it reduces the risk of collision to zero if random numbers are used to select the IID or if the IID is based on the hash of some well known values such as a public key. In this paper, we use this property to allocate several addresses to a same interface to create partitions among applications running on the network.

Moving from IPv4 to IPv6 is not straightforward since applications, operating systems, and routers, have to be modified to take into account the new IP protocol version. Different tunneling mechanisms can be used to gain IPv6 connectivity in IPv4 networks : 6to4 [11], Teredo [16] or Softwires [15]. New applications can start using these accesses. A simple transition model can be implemented: legacy applications continue to use IPv4 in conjunction with NAT and new applications are based on IPv6 to benefit of end to end capabilities.

Regarding security, IPsec is an extension of IPv6. Each network node implementing IPv6 must include IPsec. The IPsec technology is a set of mechanisms that guarantees a cryptography-based security to IP traffic. The proposed IPsec services offer access control, integrity, confidentiality, authentication of the data's origins and protection against the replay of the IP packets. IPsec uses two protocols to ensure traffic security: AH (Authentication Header) and ESP (Encapsulating Security Payload). Both support two function modes: (1) the transport mode (which is preferred in IPv6 [7]), offering protection only for those protocols above the IP protocol and (2) the tunnel mode which secures the IP and TCP protocols. The AH header adds an extra field to the IP datagram and ensures, depending on the IPsec mode, the integrity, the authentication of IP packet origins and possibly the anti-replay. The ESP header offers confidentiality services. The creation of IPsec tunnels may be done in an end-to-end manner (i.e., the two communicating parties represent the ends of the IPsec tunnels), gateway-to-gateway and/or combinations of the previous.

If IPsec tries to answer to the need of secured communications, the IPv6 autoconfiguration mechanisms may be a problem. For example, there are several threats discussed in [13] concerning the Neighbor Discovery protocol. As a solution, SEND protocol [14] provides non-IPsec mechanism for addressing these threats. The central mechanism in SEND is CGA [12]. CGA computes the 64-bit IID as a hash function on a set of parameters and mainly on: (1) the public key of the network device for which the CGA address is acquired; and (2) a "modifier": a random 128-bit value.

The main advantage of the CGA mechanism is the possibility to enhance authentication without the need of a PKI. It eliminates the address spoofing attacks of existent nodes. However it does not confirm that the node really exists; a pair of public/private keys may be obtained even with a cell-phone, thus anybody can fake the existence of a certain node.

We describe in the sequel the integration of IPv6 and the security requirements in Home Networking.

III. Security requirements in IPv6 Home Networks

A. Integrity, Confidentiality and Authentication

Home networking is a very good example of emerging applications for the IPv6 protocol. Currently Home Networks are very simple and are composed of a single link where Ethernet and Wi-Fi are bridged. IPv4 will still be accessible but generally through private addresses and NAT. IPv6 will provide full connectivity and end-to-end applications.

Even if the size of the network is limited to a house or an apartment, the network topology may be complex due to the different technologies used to transport information. For instance, a digital camera and a TV set can be connected through an IEEE 1394 bus. The introduction of the IPv6 protocol between the link layer and the video will be totally transparent for the user. If a router is connected between the IEEE 1394 and the home network, the video flow can be sent to any equipment in the Internet network.

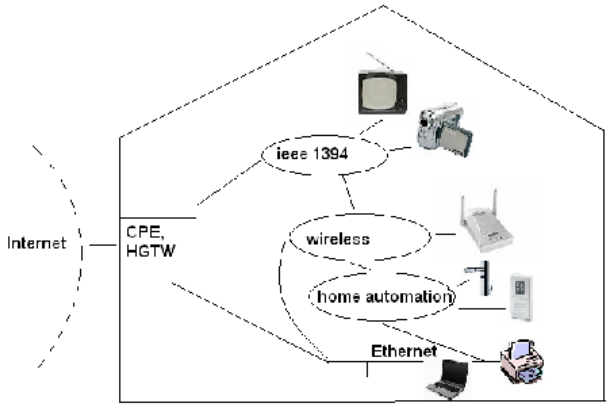

Fig. 1. A Home Network Architecture.

Home networks may have the following characteristics:

- no network knowledge from the users;
- complex topology due to the absence of knowledge from users;
- limited number of links, routers or equipments;
- intuitive results: for instance if a user adds a link or a router on his network, the network performances must be improved;
- include unicast and multicast routing;
- manage dynamicity of the network, partitioning and merging;
- require few CPU power of memory to be integrated in cheap equipments.

Auto-configuration in IPv6 networks is a major feature to deploy this protocol in new areas. Home networks have to take into account new functionalities if IPv6 protocol management has to be invisible from the user. However, if IPv6 becomes suitable from this point of view, there are security requirements that have to be ensured in home networks like in any other information system.

Integrity in home networking is the security feature that guarantees to those parties that are interacting that the information exchanged was not modified during the transmission; and, if such information was altered, the security mechanisms should ensure that the modifications are detectable. Thus, enforcing integrity means prevention and detection. Confidentiality is the security property that guarantees that the sensible information is accessible/revealed only to the trusted parties. Authentication is one of the main problems we effectively address in this paper (cf. Section IV). The authentication may be dealt with directly at the IP level and IPsec is consequently suitable. IPsec-based tunnels respond to these demands.

IPsec mechanisms become a requirement in home network scenarios such as the sharing of user-created content, and remote control of electrical appliances. However it is not always possible to set up a secure communication in terms of authentication, integrity, and confidentiality, because:

- the home network is not controlled by a home server involving advanced security features; a home server would be a perfect solution to secure these services: it could embed a CA (*Certificate Authority*), IPv4/IPv6 transition mechanisms, IPsec modules, or access control modules ensuring permission/authorization management [19];
- there is no PKI infrastructure;
- a user may decide to limit the access to certain appliances even inside his home network; different home users have different privileges upon home appliances;
- the configuration of these devices requires a non-negligible user-burden;
- the consumption of resources during the encryption computations is unacceptable.

We describe in the sequel two scenarios that effectively address the security problems we are dealing with.

B. Remote Services

Home network services, as defined in [20], may be classified as follows:

- indoor home appliances communication (i.e., inside the home network);
- Internet services to which the home user subscribes; this is currently related to the purchase of a special device functioning, for example, in home automation (e.g., special sensors, cameras for remote supervision, etc.) and which embeds a basic system and a network interface;
- remote control of home devices by an Internet user/application.

There are many home usage scenarios that require the set up of certain security mechanisms in order to ensure the well functioning of the entire home network. A malicious indoor device (e.g., a malicious PC) can compromise other home appliances involved in home automation. For example, a movement sensor can be installed on a stair for different purposes (1) to trigger an alarm in case of robbery if the alarm system is set (2) to switch on the light when the alarm system is off (3) to monitor on a PC the number of people using the stair. An attacker evading the security of the PC may access the sensor and then the alarm system.

There should be a mechanism to guarantee the partition of users and applications inside the home network; this is a strong requirement especially for current home networks that do not contain a security featured home server. In order to highlight this partitioning requirement and also the authentication issue, we present a scenario of service delegation. We consider the delegation of a service as granting access to an already deployed home network service. Figure 2 depicts an IPv6 architecture involving two home networks (HM) and two service providers (SP) which may act in partnership. HM1 subscribes to SP1 for a certain service (e.g., an alarm) after purchasing a dedicated sensor (dev1) which embeds a minimal IPv6-enabled system. The class of service is “silver” type: it will involve only SP1. If the service class had been “gold” (more expansive), SP1 would have contacted SP2 when certain parameters of alarm messages received from HM1 run over the

"silver" service characteristics. We would have called this a delegation between SP1 and SP2. Meanwhile, HM1 delegates this service to HM2: only one device (dev2) in HM2 will be able to connect the HM1 sensor dev1. The service demands the authentication of the two communicating parties.

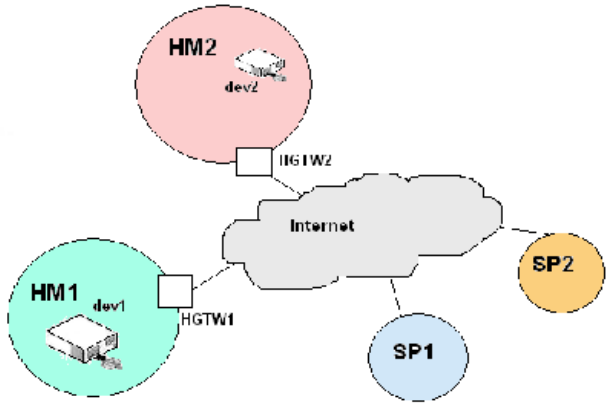

Fig. 2. Envisioned Sample Scenario.

A very current application is the distribution and/or recording of a video flow inside the home network. A satellite receiver may broadcast a video stream on the home network and some TV displayers can be used to visualize the stream. A DVD recorder may also be used to generate another video stream on the network; in order to protect some video contents, the DVD recorder is not allowed to record the video stream produced by the satellite receiver. A security mechanism should ensure not only a correct authentication of each communicating parties but also a confinement flow issue.

We try to answer to these security requirements by proposing a partitioning mechanism of groups of applications and users. The mechanism is actually defined as a set of two sub-mechanisms: (1) the first ensures the authentication inside each group of applications (we will call it a domain) and (2) the second deals with the user management.

IV. Proposed Mechanism

The global connectivity of IPv6 nodes and the lack of a centralized security control in a home network determine the main idea of our approach: enforcing a local access control (at the OS level) by imposing the use of a single CGA address for each service. Even if CGA is appropriate for IP layer authentication, we try to extend this mechanism to an upper layer (i.e., the application layer) in order to ensure the (non-IPsec) authentication of our home devices. Of course, security may be enforced if an IPsec context is related to such CGA address. However, we believe not all devices are able to sustain IPsec capabilities. We make use of the CGA "modifier" and we consider it as an out-of-band pre-shared key. In [12], several CGA addresses may be computed with the same public key (by choosing different "modifier" values); we also consider using the same "modifier" to obtain different CGA addresses (each implying a different public key).

A. The Authentication Mechanism

To illustrate our approach, let us consider the same architecture as in Figure 2. The home devices (e.g., the dedicated sensor) embed a small module to their system that deals with authentication. During the subscription phase to the "silver" service, the home user of HM1 will acquire (out-of-band) the "modifier" *modif_silver* from SP1. A modifier will serve as an identifier for a *domain*. A domain is defined as a set of devices/applications involved in the same service(s). In Figure 2, one domain corresponding to the "silver" service includes only dev1 and SP1 at the very beginning. All devices of a domain share the same "modifier" (i.e., the secret). A device may belong to different domains, but in each domain it uses a different IPv6 CGA address. A user may add new devices to a domain; this corresponds to the delegation of a service as exemplified in Sec. III-B.[1] Let us consider the following notations:

- hash (A): a one-way hash function on the A bit-value;
- key_X: the public key of the entity X;
- modif_Y: a "modifier" value of the domain Y (e.g., modif_silver);
- prefix_Z: the IPv6 prefix of the Z home network;
- |: the concatenation operator;
- ID_Y_Z: the IPv6 interface identifier of the device Z in the domain Y;
- @HMi_j_serv_z : the CGA IPv6 address of the device j for the service z in home network i.

Given the same topology as in Figure 2, dev1 will acquire its CGA address (which will be used in the "silver" service with SP1) as following:

ID_silver_dev1 = hash (key_dev1 | modif_silver), @HM1_dev1_silver= (prefix_1 | ID_silver_dev1).

We considered a simplified CGA mechanism, with no auxiliary CGA parameters. For any other communication which does not involve the "silver" service, dev1 will use a different IPv6 source address. In order to delegate the "silver" service as described in Sec. III-B, dev2 will receive the *modif_silver* (in an out-of-band manner) and will acquire a CGA address.

The authentication of each home device occurs in a similar manner as described in [12]. Each node should store locally only the "modifiers" corresponding to each of the subscribed services. They will identify domains and represent the pre-shared out-of-band secrets. In addition, at the beginning of each communication, a simple application-level routine sends a message (e.g., CGA parameters data structure) that enables the other party to perform the CGA verification.

B. User Management

However, this approach cannot be satisfactory if there is no user management. As already mentioned, some users can have

[1]This approach resembles DTE (Domain Type Enforcement) [1]. In our case, passing from a domain to another supposes acquiring and using a different "modifier", as for the execution of the so-called entry-points in DTE.

access to all home appliances (e.g., the parents) and other users, a restraint access (e.g., children). Thus each class of users may have a different view on the home network resources and services. Furthermore, the manner in which each user launches a connection may be subject to an access control: for example, only certain home users have the permission to initiate secured communications (e.g., IPsec tunnels). The lack of a centralized home server does not facilitate user management. We propose to enforce the user management with a local access control. The access control will be basically a filtering mechanism on the source addresses each user can employ.

Every home console (e.g., a PC) has normally at least a network interface and several IPv6 addresses corresponding to each home user authorized to log in. These addresses may be simple ones or CGA addresses (as a consequence of service subscription). Some of these addresses are related to IPsec contexts and thus only certain users will be able to set up secured communications with the Internet, for example. There should be a user with privileges for system administration (e.g., a parent) that has a global view of the entire home network. On the other hand, none of the other users should be able to reveal (or to modify) somebody else's sensible information by the means of basic commands such as "ifconfig" and "cat".

Every user will get an IPv6 address for each network interface; we can envision here a way to obtain the IPv6 interface identifier similarly to the EUI-64 IPv6 address format (used by default by cisco routers in link-local addresses).

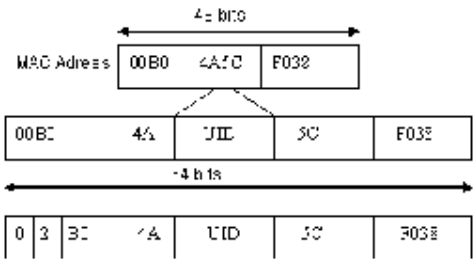

Fig. 3. IPv6 ID from MAC Address and UID.

Instead of the 16-bit FFFE separating the 48 bit MAC address (as in the current EUI-64 IPv6 Interface Identifier), we consider the UID (user ID) to simplify and to ease the separation of IPv6 local addresses. After the first login, each user will be free to create as many addresses as he/she wants and/or CGA addresses for possible services he/she subscribes. DAD (duplicate address detection) mechanisms will be enabled and slightly modified in order to prevent the conflict with other users' addresses.

V. State of the Implementation

Although the implementation of our proposal is not yet finished, we have already identified and developed its main functionality in the form of a set of modules for the series 2.6 of the linux kernel. We overview in the sequel, two of the main components of our implementation: a kernel module for the filtering of network traffic; and a kernel module for the manipulation and handling of system calls. By using these modules we can guarantee the regulation of proper source addresses, the recycling of already existing administration tools, at the same time that the system continues dealing with the user management and CGA authentication/computations.

A. Filtering of Network traffic

To enforce our user management scheme, we propose a local access control mechanism that regulates the source addresses that a given user may apply. With the objective of validating such an approach, we implemented a first prototype of our proposed local access control mechanism by using the netfilter framework of the linux kernel series 2.6 [29]. The netfilter framework offered us all these programming tools that we needed in order to manipulate network traffic, such as packet filtering, network address translation and many other packet alterations.

The architecture of netfilter is based on a set of well defined points (*hooks*) for the treatment or monitoring of network packets: `NF_IP_PREROUTING`, `NF_IP_LOCAL_IN`, `NF_IP_FORWARD`, `NF_IP_LOCAL_OUT` and `NF_IP_POST_ROUTING`. Therefore, when a packet enters into one of these hooks it can be monitored and manipulated by a chain of functions previously registered into the kernel of the operating system, as a set of kernel modules. Every function is free to manipulate the packet in many different ways: dropping it, passing it to the following function, and so on. In our case, we deployed the local regulation of source addresses by implementing a set of filtering functions in the `NF_IP_LOCAL_OUT` hook of netfilter with a higher priority than *iptables*. By analyzing these local IP datagrams that the system generates in order to contact a remote system, and given a set of filtering rules previously defined, we can detect when a specific address is not allowed to be used by a given user and, consequently, to drop such datagrams.

To define which addresses (or services) may be used for each user, our set of filtering modules are actually configured through a set of security rules. Each rule defines an *action* in $\{deny, accept\}$ that applies over a set of *condition* attributes, such as user_id, process_id, source address, etc. We can also define, through these security rules, either open or closed default policies. The complete set of rules are stored in a set of configuration files that are loaded at boot time through the *proc file system*. The *proc file system* (procfs) is a special virtual file system in the linux kernel which allows user space programs to access kernel data structures.

Up to now, the specific policy of each module should be loaded at boot time. We are planning to extend this feature to allow the reload of policies at runtime. On the other hand, a third party tool that we presented in [24] allows us the automatic transformation of high level language policies into the specific set of configuration rules of each module.

B. Manipulation of System calls

With the objective of recycling already existing administrative tools, and as a complement of the filtering modules discussed above, we implemented in our prototype a mechanism for the manipulation of system calls — in order to alter their outcoming information. When a process calls a given routine of the kernel, to obtain for example which IP addresses are associated with a specific network interface, such a request is translated to a system call through the standard glibc library. Our proposed mechanism regulates the access to these system calls, by implementing a further control to the information supplied by the system call to the process, in order to be coherent with the scheme proposed in this section. The development of this mechanism relies on the use of the *Linux Security Modules* (LSM) framework for the series 2.6 of the kernel linux [30]. The LSM framework can accommodate several approaches rather than a single specific access control mechanism. It offers several interception points across the kernel that can be used to implement multiple access control strategies.

To deploy the strategy presented in this section, we implemented a set of LSM modules that extend the normal behavior of the traditional system calls that manage file system and network processes, such as the *read*, *write*, and *bind* system calls. The configuration of these LSM modules relies on a set of security rules loaded into each module through the proc file system, and that may define an *action* in $\{deny, accept\}$ that applies over a set of *condition* attributes, such as user_id and source address. In addition, since the access to some network administration tools like, for example, *ifconfig*, must be done through the *proc file system* associated to the kernel, we complemented the LSM modules with an intra-kernel routine that controls and guarantees the proper management of the data returned from the *proc file system*. This complementary routine intercepts the operations of this special file system, for example, the *proc_dir_entry_point* that targets the file /proc/net/if_inet6.

The overhead introduced by our proposed interception routines can be considered as the impact into the performance of the system introduced by the use of both the netfilter and the LSM framework, as well as the specific implementation of our filtering and manipulation modules. On the one hand, and as pointed out in [30] and [29], the overhead of these two frameworks into a normal GNU/Linux system is minimal compared to the standard kernel series 2.6 of linux. In the same scope, the inclusion of our modules does not introduce complex tasks that may require a lot of extra computation. We performed several tests over the implementation of our proposed modules and intra-kernel routines, by using an incremental number of security rules for each module. We tested them by using the benchmark tools of LMbench [28]. These initial experiments reported that the impact into the performance of the system was minimal (about 5%-15%). Similar experiments and results had been previously reported in [5], [6].

VI. Related Work

Several works take into account the home devices registration and authentication. The authors in [25] propose a smartcard-based authentication in the home environment. Because of the special requirements of the home environment, such a solution is recommended for indoors services but unfortunately not always satisfactory in, for example, remote control services. In [10], once a home device (e.g., an electrical appliance) is introduced in the home network, it can be remotely controlled whether indoors (i.e., from inside the home network) or outdoors (i.e., from the Internet) if the authentication phase is successful. The proposal in [10] takes into account a double-layered PKI infrastructure where the CPE/HGTW (which is also a home server) acts as a CA (Certificate Authority) root for home devices (first PKI layer). Thus the home network relies on this central home server that registers every new device which in turn receive certificates. Negotiations between different home domains are based on the exchange of certificates between their correspondent home servers which in turn must register the same CA root (second PKI layer - the "global" one).

MAGNET [21] was a R&D project that aimed to bring new technologies to the needs of a user in order to provide secure user-centric applications. Their main concepts are PN Personal Network and PN-Federations which are set of nomadic and interconnected devices belonging to different PNs. Related to the MAGNET project, the approach in [2] describes a very interesting solution to provide instant security to interconnected devices which cross different access domains. However, for such a problem, there must be a security server for each access domain and thus, the proposed solution is not totally adapted to the home network requirements.

There are several works that propose solutions to secure home networks and most of them consider the existence of a fully security featured home server, including security modules. The proposal in [17] depicts a "star-topology" Home Network with a central home server. Their security requirements (e.g., confidentiality) is addressed with a new protocol designed to construct a "secured-channel" from home devices to Internet devices and using the home server as an intermediary. Implying a similar fully security featured IPv4 home server, the authors in [19] try to solve the confidentiality, integrity and authentication threats by attaching an IPsec module to every home device not having IPsec functionalities. The home server acts as a supervisor in the establishment of IPsec tunnels and also includes a DAC module (Discretionary Access Control) to mitigate the tunnel establishment attempts and user authentication via a home web server. Unfortunately, this approach lacks concrete implementation. We consider that current architectures do not incorporate a security featured home server yet (in fact, our architectures do not include a home server and centralized control) and the home gateway or the CPE is totally transparent to IPsec tunnels.

The approach in [8] proposes a "partial" user authentication: one can obtain privileges based on the trustworthiness of

the user within the system. They provide tools to "quantify" the authentication of a given user which presents a set of credentials in the system (for example, if the system trusts the user in a proportion of only 70%, the user will have less privileges). Their vision is similar to the following: for example, a user can be authenticated as root only when he/she is inside the home network and as a simple user ("other") when connected from the Internet; in both cases, this user uses the same credentials. Such a case is rapidly dealt with when there is a home server incorporating an access control module. The approach in [17] manages user authentication based on RBAC [26] modules placed on a central home server.

VII. Conclusions

The use of a wide sort of pervasive devices at both home and work networks is getting very popular, and a new brand of powerful remote services, such as health care monitoring or virtual machine systems, are today very easy to implement. Although the deployment of these services increases the productivity and flexibility of its users, unfortunately, it also increases the odds of being attacked and/or risking their privacy. In this paper, we studied some of the advantages of using the IPv6 protocol in home networks and how to improve the security in home network environments. We focused our work on defining a mechanism that partitions groups of applications and users. We based our approach on an IPv6 proposal: CGA - (Cryptographically Generated Addresses) and we proposed a scheme of non-IPsec authentication for home devices. We also addressed the problem of user management: users are allowed to use only their set of IPv6 addresses; some of them are related to IPsec contexts and thus we separate privileges based on an IPv6 addressing scheme. We believe that our approach can be successfully applied on current home networks with no central home server and only with a "transparent" home gateway.

Part of the work presented in this paper made possible the submission of a French INPI 08/05661 patent with the central idea of a mechanism allowing the attribution of several IPv6 addresses to a system interface and of a single IPv6 address per service (application) and/or of a different IPv6 address per user in order to ensure different views of the system based on the IPv6 address used to contact the system.

Acknowledgments— The work was supported by a grant from the Brittany region of France and by funding from the SEC6 Project.

References

[1] Badger, L., Sterne, D.F., Sherman, D.L., Walker, K.M., and Haghighat, S.A. Practical Domain and Type Enforcement for UNIX. In *Proceedings, IEEE Symposium on Security and Privacy*, pages 66–77, Oakland, California, 1995.

[2] Calin, D., McGee, A. R., Chandrashekhar, U., and Prasad, R. MAGNET: An approach for secure personal networking in beyond 3g wireless networks. In *Bell Labs Technical Journal*, 1(1):79–98, 2006.

[3] Chelius, G., Fleury, E., and Toutain, L. No administration protocol (nap) for ipv6 router autoconfiguration. *Int. J. Internet Protocol Technology*, 1(2):101–108, 2005.

[4] Chelius, G., Fleury, E., and Toutain, L. Using ospfv3 for IPv6 Router Autoconfiguration. Internet draft, IETF, June 2002.

[5] Garcia-Alfaro, J., Castillo, S., Castella-Roca, J., Navarro, G., and Borrell, J. SMARTCOP - A Smart Card Based Access Control for the Protection of Network Security Components. In International Workshop on Information Security (IS'06), Montpellier, France, 2006.

[6] Garcia-Alfaro, J., Castillo, S., Castella-Roca, J., Navarro, G., and Borrell, J. Protection of Components based on a Smart-card Enhanced Security Module. In International Workshop on Critical Information Infrastructures Security (CRITIS'06), Samos, Greece, 2006.

[7] Gisele Cizault (ouvrage collectif du G6). IPv6, théorie et pratique. O'Reilly - Paris, 11/2005.

[8] Gomez, L. Towards User Authentication Flexibility. In *International Conference on Security and Cryptography*, Barcelone, Spain, 2007.

[9] Huitema, C. IPv6: The New Internet Protocol. Prentice-Hall, ISBN 0138505055, 1997.

[10] Hwang, Jin-B., Kim, Do-W., Lee, Yun-K., and Han, Jong-W. Two Layered PKI Model for Device Authentication in Multi-Domain Home Networks. In *IEEE Tenth International Symposium on Consumer Electronics ISCE '06*, St. Petersburg, Russia, 2006.

[11] Internet Engineering Task Force. An Anycast Prefix for 6to4 Relay Routers. RFC 3972, June, 2001.

[12] Internet Engineering Task Force. Cryptographically Generated Addresses (CGA). RFC 3972, March, 2005.

[13] Internet Engineering Task Force. IPv6 Neighbor Discovery (ND) Trust Models and Threats. RFC 3756, May, 2004.

[14] Internet Engineering Task Force. SEcure Neighbor Discovery (SEND). RFC 3971, March, 2005.

[15] Internet Engineering Task Force. Softwire Problem Statement. RFC 4925, July, 2007.

[16] Internet Engineering Task Force. Teredo: Tunneling IPv6 over UDP through Network Address Translations (NATs) . RFC 4380, February, 2006.

[17] Jiang, Z., Kim, S., Lee, K., Bae, H., and Kim, S. Security service framework for home network. In *Fourth Annual ACIS International Conference on Computer and Information Science*, pages 233–238, Jeju Island, South Korea, 2005.

[18] Kim, Geon-W., Kim, Do-W., Lee, Jun-H., Hwang, Jin-B., and Han, Jong-W. Considerations on security model of home network. In *8th International Conference in Advanced Communication Technology ICACT 2006*, vol. 1, 2006.

[19] Larab, A., Martineau, P., and Gaucher, P. Un module de sécurité pour sécuriser les communications domestiques. In *Third International Conference: Sciences of Electronic, Technologies of Information and Telecommunications SETIT '05*, Tunisia, 2005.

[20] Lee, Yun-K., Ju, Hong-I., Kim, Do-W., and Han, Jong-W. Home Network Modelling and Home Network User Authentication Mechanism using Biometric Information. In *IEEE Tenth International Symposium on Consumer Electronics ISCE '06*, St. Petersburg, Russia, 2006.

[21] MAGNET Beyond project. `http://www.ist-magnet.org/`.

[22] Mangues-Bafalluy, J., Martinez-Perez, G., and Chelius, G. Evaluation of router autoconfiguration time during network initialization for centralized and distributed schemes. In *Globecom 2005*, IEEE, Saint Louis, USA, November 2005.

[23] Park, M. H., Kim, J. T., Paik, E. H., and Park, Kwang R. Deployment Strategy and Performance Evaluation of the IPv6 Home Network using the Home Server. In *IEEE Transactions on Consumer Electronics*, 53(1):114–119, 2007.

[24] Preda, S., Cuppens, F., Cuppens-Boulahia, N., Alfaro, J. G., Toutain, L., and Elrakaiby, Y. A Semantic Context Aware Security Policy Deployment. In *ACM Symposium on Information, Computer and Communication Security (ASIACCS 2009)*, Sydney, Australia, 2009.

[25] Pujolle, G., Urien, P., and Loutrel, M. A Smartcard for Authentication in WLANs. In *2003 IFIP/ACM Latin America conference on Towards a Latin American agenda for network research*, pages 125–130, La Paz, Bolivia, 2003.

[26] Role Based Access Control. `http://csrc.nist.gov/rbac`.

[27] Toutain, L., Bruno, S., Dupoin, F. and Binet, D. The Point6Box Approach. Internet Draft, IETF, January 2006.

[28] Voy, L. and Staelin, C. LMbench – Tools for Performance Analysis. `http://www.bitmover.com/lmbench/`, 1998.

[29] Welte, H., Kadlecsik, J., Josefsson, M., McHardy, P., and et al. The netfilter project. [Online]: `http://www.netfilter.org/`

[30] Wright, C., Cowan, C., Smalley, S., Morris, J., and Kroah, G. Linux Security Modules: General Security Support for the Linux Kernel. In *11th USENIX Security Symposium*, USA, 2002.