# Federated Autonomic Management of HAN Services

Rob Brennan, Zohar Etzioni, John Keeney,
Kevin Feeney, Declan O'Sullivan
FAME & KDEG, School of Computer Science and Statistics
Trinity College Dublin
Ireland
{rob.brennan, etzioniz, john.keeney, kevin.feeney, declan.osullivan}@cs.tcd.ie

William Fitzgerald, Simon Foley
FAME & Department of Computer Science,
University College Cork,
Cork,
Ireland
{wfitzgerald, s.foley}@cs.ucc.ie

*Abstract*— Management of a heterogeneous "outer edge" network, where a Home Area Network (HAN) interoperates with access networks and service providers, is complex and error prone. Configuration of a HAN is typically performed by non-technical end-users. As a consequence, an effective HAN configuration may be hampered by a poor understanding and/or management of HAN service requirements. Mis-configuration, may result in the failure to adequately provide HAN services on an end-to-end basis, with consequent operational and support costs for network and/or service providers. Thus, the challenge becomes one of autonomically maintaining meaningful and error-free heterogeneous HAN configurations. This paper explores an integrated solution to address the following identified requirements: managed capability sharing, usability and security. A prototype HAN gateway architecture is outlined, which builds upon explicit user-centric semantics and enables autonomic management of shared UPnP services with appropriate access controls. A case study and performance evaluation of this work is also presented.

***Keywords- HAN, Semantics, Federation, Access Control***

## I. INTRODUCTION

An emerging characteristic of communications networks is the growing complexity and heterogeneity of the "outer edge" domain – the point where Home Area Networks (HANs) and other restricted private networks attach to commercial access networks [1]. Management of the components of these networks, outer edge devices, such as femto base stations, home gateways, set-top boxes, networked consumer electronic devices, is today provided on a piecemeal basis, with different devices having a wide range of management functionality, from none to proprietary or, at best, conforming to one of a range of competing standards [2]. Furthermore, management operations must be performed by end-users, so there is huge potential for mis-configuration that can impact significantly upon service delivery on an end-to-end basis with consequent operational and support costs for service providers [3].

This paper proposes new approaches for how services can be deployed and managed in a HAN context. Our primary research focus is on exploring methods through which HAN-based network management systems can assume autonomically the responsibility for appropriate and secure configuration of HAN devices as new services are deployed and delivered to end-users. A significant challenge in this regard is that as the diversity and capabilities of HAN devices increases, it becomes increasingly difficult to capture and exchange the knowledge required to facilitate delegation of management capabilities between management domains.

A second focus is on end-user enablement through the application of autonomic techniques and semantic policy-based control in the HAN. This includes compliance-driven network access control configuration synthesis that autonomically mitigates user-defined levels of threat. Future research will investigate autonomic network control, e.g. QoS enforcement, via refinement of user-defined service or network policies and end-user focused visualization tools that leverage semantically annotated management and monitoring data. This work supports the first focus above by providing HAN-centric policy-based control interfaces that are suitable for end-to-end integration with peer or operator management systems and also meaningful ways for end-users to manipulate and monitor the federated, managed, autonomic control loops of their devices.

These goals give rise to the following research questions:

*1:* What methodologies and techniques are appropriate to capture semantic models and their mappings that will enable the exchange of the management capabilities plus needs of HAN devices and service execution platforms in support of coordination of management activities on an end-to-end basis?

*2:* How can the flow of authority and knowledge in network management systems shape appropriate and secure configuration of devices and services deployed in HAN environments?

To address these questions an end-to-end service for sharing UPnP capabilities between federated HANs [4] is developed, furthering building upon previous research by using the Federal Relationship Manager (FRM) [5]. In this paper we discuss how that work has been extended with semantic descriptions of shared capabilities and autonomic access controls that build on both the semantic service descriptions and a security knowledge base. In addition we investigate the performance overheads of deploying our prototype capability sharing system in terms of processor load on a gateway. We present a revised gateway architecture based on [3] that links these components together.

This paper is organised as follows. Section 2 describes a use case, section 3 derives requirements for the system. In section 4 we provide an overview of existing solutions for HAN capability sharing. In section 5 we describe our approach in terms of the overall gateway architecture, UPnP capability sharing over XMPP, modelling semantic capability graphs and

This research is supported by the Science Foundation Ireland (Grant 08/SRC/I1403) as part of the Federated, Autonomic End to End Communications Services (FAME) Strategic Research Cluster (www.fame.ie).

automating HAN access control rule generation. Then in section 6 we present a performance test-bed developed with our prototype and describe an experiment to evaluate the CPU load to deploy the system. Finally section 7 describes our conclusions and plans for future work.

## II. USE CASE

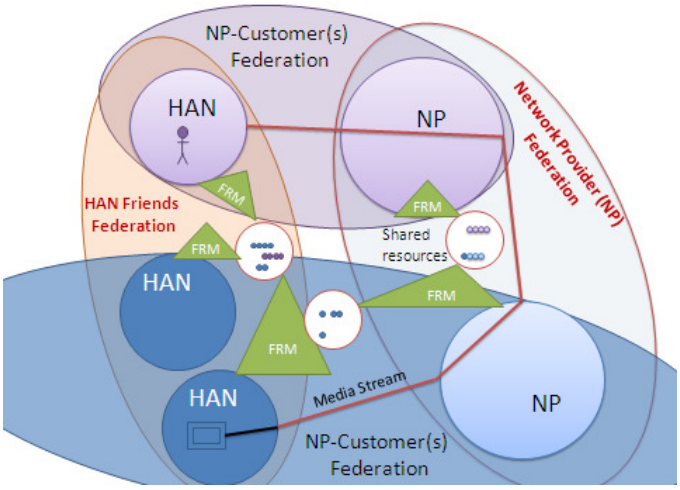


Figure 1: HAN and Network Provider (NP) Federations

Eric is playing a football game at his grandfather John's house. Eric's father, George, is at home on other business. However, George has recently installed a new autonomic gateway in his Home Area Network (HAN). This HAN gateway provides George with access to a number of new service providers and the ability to share HAN capabilities with them, other HANs or his network provider. George has subscribed to a service called *StreamToHAN* that allows him to receive real-time multimedia streams to his HAN media center from remote (trusted) users. John's mobile service provider, for its part, provides a service which allows their subscribers to utilise the *StreamToHAN* service – they can send real-time high-quality video and audio streams directly from their video devices to any other HAN that supports the service. There is a local gateway component which also acts as the UPnP domain controller. George has a wide variety of devices on his HAN and at least some of them are typical consumer devices that do not support remote management interfaces.

As the football match is about to end, Eric makes a run towards the goal, John invokes the service on his recorder and starts filming. Grandad George receives an invitation from John to view a live multimedia stream of his grandson's football match. He accepts the invitation and the football match is displayed on his TV via his HAN media center. In the background George already has several other active network streams downloading software updates, email and media files for later use.

As the game approaches its conclusion, a more serious error occurs in the HAN the media renderer encounters an unrecoverable stream processing error and that disables the television from receiving the live multimedia stream. George decides to watch the remainder of his son's football match on his laptop. However, the laptop is on a separate subnet protected by the gateway firewall and is not currently configured to receive communications from the HAN media center. The HAN gateway hosts a autonomic Home Area Network Access Control (HANAC) service. The HANAC service will re-configure network communications (multimedia) between the relevant HAN security zones in accordance with George's security requirements.

As George had previously authorised remote management by his network provider, this rich HAN capability information is also available to the network provider, to allow their engineers to analyse the error and to give them the ability to subsequently remotely fix the error in the HAN. In conjunction with the HAN forwarding semantic monitoring information to the network service provider, George has subscribed to a third party remote HAN management service provider who also receives the relevant information. Based on this information, the relevant external service providers shall try to remotely rectify the fault and George may be presented with the opportunity to view the final moments of the football match on the television once again.

## III. REQUIREMENTS

### A. Capability/Management Capability Sharing

We define a capability as an abstraction of one or more useful aspects of one or more resources or services. Capabilities can be local or remote and must be actively shared to grant remote access. Federation is defined as *a "persistent organisational agreement which enables multiple autonomous entities to share capabilities in a controlled way"* [3]..

There are multiple, overlapping reasons for network and service providers to engage in capability sharing (federation) in the context of the HAN environment. For example, they gain access to individual HAN capabilities in order to maximize their ability to deliver end-to-end services to HAN owners (customers). Note, HAN owners may place requirements on their networks in terms of capacity, services or resources offered. For example, some of the management capabilities of their own networks are shared. From a HAN user perspective, having the flexibility to deal with multiple providers, perhaps even on a per-service basis, may drive down costs and increase business agility. Finally, if there is an emergence of widespread prosumers (producer-consumers) in the marketplace then it is likely that every actor will have more dynamic business relationships in the future, perhaps structured as value networks. Even if such content prosumers are not motivated by profit, the flexibility engendered by pervasive social networking and other similar advances in media production democratization on the Web means that people wish to connect directly their digital infrastructure with that of their friends, on a peer to peer (HAN to HAN) basis.

### B. Usability (Manageability)

The shift in value towards products' ability to be used in concert with the rest of the digital ecosystem means that consumers must be able to manage (or delegate the management of) federated multi-device, multi-user, multi-network deployments, which was once only the remit of traditional operator's network control centers. Thus the key to

the success of these networking features will be the ease with which ordinary users can access them.

Empowering non-technical service consumers and managers depends on making complex systems comprehensible and in a manner that makes it much easier for the user to elicit knowledgeable conclusions from the information presented. This is also essential to empower non-technical service consumers or suppliers to make sense of complex tasks, where users need to be able to understand the information that informs the task and be able to abstract and contextualise this possibly unfamiliar information from a viewpoint that makes sense to them [6].

There are several sources of system complexity that must be tamed to create usable federated HANs. One possible portioning of system complexity is into four parts: complexity of the system model that users must understand to correctly use or manage the system, complexity of the system's interaction model, the complexity of the system's governance model, and the level of automation that the system provides. Semantic modelling approaches favour dynamic exploration of systems at multiple levels of abstraction, which can help bridge the semantic gap between users and the technical internals of systems. Self-configuring autonomic systems are one approach to increasing usability [7][8][9], while policy-based management has been shown to be an intuitive governance model for systems [10].

### C. Security

While HAN services may provide their own security, for example access control, it is considered best practice to rely on multiple layers of security, for example the deployment of firewalls [11]. The Home Area Network Access Control (HANAC), e.g., a gateway firewall, provides an important point of demarcation between networks of different levels of trust. The challenge is to generate a HANAC configuration that is aligned with the HAN service security requirements, i.e., it permits valid service traffic, no more and no less.

Management of the HANAC configuration is complex and requires the HAN administrator (home user) to have a deep knowledge of the high-level security requirements of each HAN service and how those requirements may be upheld as low-level HANAC configurations. Effective configuration may be hampered by a poor understanding and/or management of each service's security requirements, which in turn, may unnecessarily expose the HAN to known threats. For example, consider the following high-level security requirement: *permit inbound IPTV traffic to IPTV capable HAN devices*. In practice, deploying a HANAC access-control rule that upholds this requirement is not simply about making the IPTV port accessible on an internal HAN device (for example, a set-top box or a UPnP server) for all traffic. One may wish to deny certain IPTV service providers (for example, IP addresses), permit IP address multicast over TCP and/or UDP, permit IGMP, permit RTSP streaming and, also deal with IPTV traffic that is tunnelled through proxies and VPN's available on other ports. Furthermore, it may be prudent to provide content sanitation at the IPTV application-layer. For example, IPTV content not 'Universal' or 'Parental Guidance' approved should not be displayed on a HAN TV within the children's play room but may be permitted to be displayed on a TV used by adults elsewhere in the HAN. Typical HAN administrators do not possess the expert knowledge of a security expert who draws upon best practice and standards in order to synthesise an effective HANAC configuration that is aligned with the high-level security requirements of HAN services.

## IV. Existing Solutions To HAN Capability Sharing

For the purposes of this discussion we split the prior work into the specific case of UPnP capability sharing and more general federation solutions.

### A. UPnP Capability Sharing

UPnP is a peer-to-peer architecture to allow network-enabled appliances to communicate with each other within a home area network. UPnP was not designed to run across multiple networks: its discovery is based on using local multicast addresses; its use of HTTP assumes seamless connectivity, which is unlikely across home networks where routers assign private internal IP addresses to devices.

Several researchers have proposed mechanisms for extending UPnP across multiple networks. Lee et al. [12] suggest an architecture for content sharing among UPnP devices, based on *HomeConnectors* communicating with remote *HomeConnectors* in other home area networks via a connection manager. A local SSDP manager listens to the local network and relays local SSDP announcements to remote HANs where they are repeated. However, this architecture does not traverse NAT or firewalls and assumes that all UPnP devices have public IP addresses. Chowdhury et al. [1] present a solution for connecting multiple UPnP networks based on a protocol for establishing trust groups of home networks. Once a group of home networks has been established, users can define which devices they wish to share with the group. Remote devices are represented as embedded devices in the home gateway device. This approach requires dynamic modifications to router and firewall configurations to enable sharing, which makes it less portable and resilient. Kang et al [13] present an architecture based on UPnP and OSGi that allows users to consume multimedia services from multimedia servers outside their home network. The home gateway acts as a proxy media server from multimedia providers reachable outside the HAN. However, the approach is specific to multimedia services and is not general to UPnP services. Kim et al [14] suggest using a SIP-UPnP bridge in the home gateway for allowing remote access to UPnP devices where secure VPN connections are established to support sharing between HANs.

### B. Management Federation

Historical approaches to federation or at least interoperability of telecommunications systems have always emphasized interoperability at the bearer and control planes, i.e. the minimum necessary components for service delivery. Unfortunately, as discussed above, this only gives very limited flexibility when offering services and provides virtually no support for managing the service lifecycle (although billing is always addressed) that is key to efficient leveraging of the network infrastructure. In fact it can reasonably be claimed that

this only really works when there is a very limited range of services based almost exclusively on voice circuits with well known properties and requirements. Although they are widely studied, service level agreements (SLAs) are most often a part of the legal framework for interworking as much as a technological issue. Nonetheless the ITU was standardising at least the abstract requirement for inter-domain management system interworking as far back as the early 1990s and the TeleManagement Forum has continued to work on this topic throughout the last decade. However it is unfortunate that progress has been slow and most probably the high cost of integrated OAM solutions within even a single domain make it prohibitively expensive to support inter-domain functions. A critical feature of most prior attempts to federation has been to assume or impose a single, unified management model for the network and services with consequent constraints on supported business models. It is the authors' contention that rather than imposing shared models, that management model heterogeneity is an axiomatic property of any realistic ecosystem supporting dynamic federation formation.

Proposing the use of semantic web technology for OAM is not new, see [15] for a recent survey. However new aspects of our approach are the emphasis on RDF rather than OWL (although see [16] for a recent "linked data" approach to OAM), the lack of unified, complete knowledge models of the network(s), the central role of a dynamic approach to semantic interoperability and the combination of organization-centric policy (rules) with semantics.

As business applications and processes that span organisations have become more prevalent, problems with the management of such applications and processes across multiple management domains using heterogeneous management technologies have become more apparent. Much of the research on cross-organisational management has focused on specifying contracts and agreements between organisations, which then must be monitored and enforced by both parties, in particular focusing on service levels agreements (SLAs), e.g., [17]. However they typically address much less dynamic environments than are discussed here.

## V. Our Approach

In the following sub-sections we describe the three main technical contributions of this paper: our HAN Gateway/Domain controller architecture, our novel XMPP-based approach to UPnP inter-domain capability sharing, semantic capability graphs for managing the shared capabilities and an autonomic access control configuration manager that provides robustness and usability for typical HAN users.

### *A. Gateway/Domain Controller Architecture*

Home gateway devices such as set-top boxes, cable/DSL modems, energy management gateways or networked games consoles are an obvious candidate for situating domain management software within the HAN. Gateways provide natural locations for mediation between the HAN and external actors such as service providers. They often support multiple service plane interactions and thus are more likely to be powered on and available for longer time-frames than many HAN devices. They often assume super-peer or controller roles in their associations with other local devices. Finally they tend to be built on more general purpose computing platforms with more extensive computational resources.

Our current prototyping work focuses on a Java-based gateway implementation with UPnP device connectivity. However the gateway architecture presented here (fig. 2) is itself more general. Conceptually the gateway architecture is based on three layers; the application, data interchange and instrumentation layers. The lowest layer is the instrumentation layer that mediates between different device or network technologies and local gateway functions or authorized remote users of local devices and services. The data interchange layer acts as a generalized repository for federation, gateway, HAN, device and application layer management data. The use of RDF/XML for our knowledge and information models simplifies selective re-use and merging of repositories traditionally kept separate in management systems. It also enables parallel development of our work at the application layer since most of the knowledge is kept in self-describing ontologies. The application layer hosts a set of management applications and remote service interfaces, e.g. for multimedia capability sharing or federated management functions.

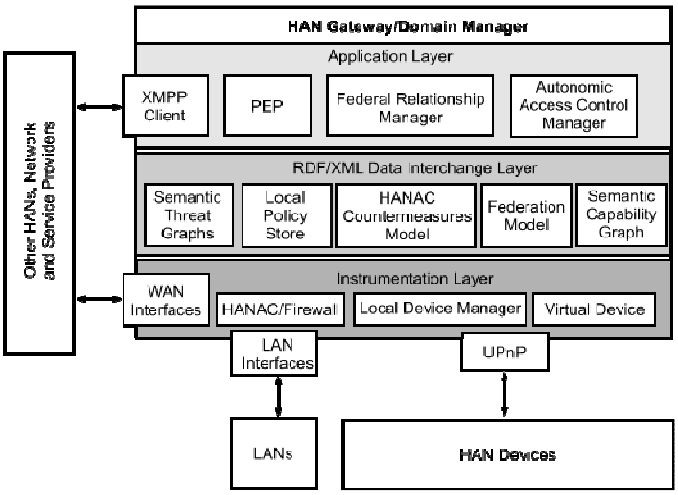


Figure 2: Gateway Architecture

### *B. XMPP-based UPnP Capability Sharing*

In order to enable secure and simple sharing of UPnP devices (and their capabilities) we have extended the UPnP protocol to work over the eXtensible Messaging and Presence Protocol (XMPP) [18] messaging infrastructure. XMPP is an open, XML-based protocol for near real-time messaging and presence. Using XMPP as an infrastructure for connecting multiple networks provides secure and standard communication, simple user roster-management and a powerful presence mechanism, which is useful for the dynamic nature of HANs. The home gateway runs an XMPP client that connects to an XMPP server in order to communicate with the user's defined friends (and shared capabilities/UPnP devices on their networks). Once a friend becomes available (online), capability sharing processes can be initiated.

This service architecture depends on two custom components: a local UPnP network manager and virtual remote devices (proxies for shared devices). The local UPnP network

manager acts as a UPnP control point for local devices and acts as an endpoint for remote invocations of UPnP's simple service discovery protocol (SSDP). This enables permitted remote networks to discover local UPnP devices. Each remote device has a local virtual device instance which is visible on the local UPnP network; this allows devices and managers to act on remote devices as if they were local. All UPnP SOAP requests or responses for remote devices are subjected to local access controls and filtering and then tunnelled through XMPP to the remote network. For a full description of this architecture see [4]. It acts as a flexible local capability definition and sharing infrastructure supporting services such as file sharing, playing media streams generated on one network on media renderers (e.g. HD TVs) on another network.

### C. Semantic Capability Graphs

Our Federal Relationship Manager (FRM) provides a means for domains to manage capability sharing, e.g. through establishing shared semantics, secure capability delegation and negotiating the operational rules for sharing, and so on. The overall FRM architecture is described elsewhere [5]. A key feature of the system is the distribution of self-describing capability authorities across federated domains. Here we present for the first time the modelling approach employed to define a semantic capability graph to support shared capability models in federated systems.

#### 1) Capability Authorities

Shared capabilities must map onto some local resources or services in a consistent way and there must be a mechanism to verify which local resources have been shared and to whom. Capability sharing is enabled by delegating capability authorities between federated domains. A Capability Authority is both a well defined capability and an associated set of permissions and non-functional restrictions.

Any party that wishes to make capabilities available to third parties must construct a capability authority model to express how the capabilities that it is offering are bundled together into capability authorities – basic aggregations of sets of capabilities with the permissions to use them. The capability authority model is instantiated as a service that compares two capability authorities and answers questions as to whether one capability authority encapsulates a second according to the model. This allows capability authorities that represent arbitrary aggregations of specific permissions to be distributed between federal participants. Whenever a third-party wishes to invoke a capability of a federal partner, the federal partner merely needs to establish whether the capability being invoked is encapsulated by a capability authority that has been issued to that third party. Capability authorities are abstractions that may map to specific resources, services or functions, but they may also map to sets of services with restrictions on parameters. So, e.g. a capability authority named *AccessMediaStreamer* may map directly to a service of the same name, or perhaps a set of services (e.g. *GetMediaInfo*, *SetPlayMode*,…). By extension, the capability authority *AccessLoungeMediaStreamer* may map to these same services but with their parameters restricted to only allow the services to be invoked on a specific device. Capability authority models thus serve to aggregate resources and services into bundles that are useful for distribution and are abstracted away from the underlying implementations.

#### 2) RDF-based Capability Models

Delegation of capability authorities is a flexible and expressive means of applying access control to capability sharing in federal relationships without requiring all of the parties to a priori support common policy or information models. There are already a wide variety of RDF-based formats for describing service invocation, e.g. see [19]. Thus, we adopt an agnostic attitude to semantic service description languages and adopt the simple assumption that the various services that constitute our capabilities may be described by arbitrary sets of RDF triples. This parallels the approach of the Linked Data community [20] to encourage the publication of structured information that is interlinked into a wider web of data to give it context and the opportunity to leverage these other information sources. Note that this does not preclude using any particular formalism such as OWL-S within a particular domain, it just does not make it a perquisite for deploying the system. The objective of this structured capability description is to hold sufficient information to assist human intervention in the likely case that completely automated approaches to universal interoperability fail [21].

Based upon these assumptions, we construct our hierarchical capability authority models as RDF graphs themselves by adding a set of triples to whatever RDF triples are available to describe the services that we wish to share as capabilities. We do this, firstly by defining the *hasAuthority* relation as a transitive relation in OWL.

| | | |
|---|---|---|
| *frm:hasAuthority* | *rdf:type* | *owl:TransitiveProperty* |
| | *rdfs:range* | *frm:CapabilityAuthority* |

Then, we can define OWL or RDF classes to represent whatever collection of services that represent the most convenient aggregations of services for our sharing requirements. For example, we could define the aggregate *UserInfoService* Instance to contain (*hasAuthority*) the capability authorities of a list of services that provide different types of information about users. Then, delegating the capability authority for all of these services can be expressed as the following RDF statement.

*frm:Recipient frm:hasAuthority frm:UserInfoService*

Thus, we can use an OWL reasoner to perform authentication on all requests to the individual constituent local capabilities – if the third party that attempts to invoke any given service has authority for that service then the request is permitted. Hence, we can inject our hierarchical capability authority models into any set of RDF triples simply be adding a small number of triples that represent the particular authority hierarchy that is most convenient for distributing the capabilities that we wish to make available to third parties.

As can be seen RDF Capability Authority definition (expressed in Turtle syntax omitting standard RDF/RDFS prefixes) that follows, this description mixes a number of vocabularies to describe the CA, some are highly structured such as assigning it to the class *frm:UpnpOverXmppService* which will define its service endpoint and protocol parameters and requirements to consume the service, others such as

defining the *frm:serviceType* as the concept *Video_Mixing_Renderer* from the *DBpedia* vocabulary have been generated through the user-based or automated tagging of the device's capabilities and these more general terms are more likely to help with high-level semantic interoperability decisions between disparate domains.

```
@prefix    frm:    <http://fame.ie/federalrelationshipmanager> .
@prefix    dc:     <http://purl.org/dc/elements/1.1>/ .
@prefix    dbpedia:<http://dbpedia.org/resource/> .
@prefix    ex:     <http://example.org#> .

ex:MyBigTV      frm:hasAuthority    ex:MySharedUpnpServices ;
      rdf:type      frm:UpnpOverXmppService ;
      dc:creator    ex:TheHanOwner ;
      dc:date       2010-01-14;
      rdfs:comment  "Capability descfor HD TV in front room";
      frm:generatesEvent    frm:ConfigurationError ;
      frm:generatesEvent  http://sw.opencyc.org/concept/Mx4rwJN-
                                        YpwpEbGdrcN5Y29ycA ;
      frm:serviceName       "Display on my big TV" ;
      frm:serviceType       dbpedia:Video_Mixing_Renderer ;
      frm:hasInput          dbpedia:Streaming_media .
```

Thus capability models provide a basic means of access control across a federal relationship. Capability authority models can also be associated with policy rules that are defined within the management system of the party that owns or controls the underlying resources that offer the capabilities, for example the rules automatically generated by the Autonomic HANAC Configuration Manager described next.

### D. *Autonomic Access Control Configuration*

We argue that a framework is required in which one can uniformly represent and reason about the knowledge associated with HANAC security configuration to simplify user involvement in this complex area. We take an ontology engineering approach to modelling this HANAC security configuration knowledge. In [22][23] the research focused on using ontologies to model network access control configuration for iptables [24] and TCP-Wrapper [25].

An ontology provides a conceptual model of a domain of interest by providing a formal vocabulary describing various aspects of the domain of interest and provides a rich set of constructs to build a more meaningful level of knowledge. In the case of HANAC configuration management, an ontology provides the ability to make logical assertions and inferences with which to structure, share and infer new knowledge about the HAN service security requirement and the HANAC configuration domains.

A threat-based approach is proposed as a means of structuring the knowledge about the management of access control configuration. *Semantic Threat Graphs* [26], a variation of the traditional threat tree, are encoded within the ontology-based framework in order to relate knowledge about high-level HAN security requirements, best practice recommendations and HANAC access-control rules in terms of assets, threats, vulnerabilities and countermeasures. Threats are organized into a hierarchical structure such as a Microsoft STRIDE based [27] hierarchy. Identifying threats in this way, for example Denial of Service attacks, facilitates the generation of appropriate access-control rules (countermeasures) such as, automatic whitelisting of permitted IPTV service providers (IP addresses) and connection-throttling. The semantic threat graph approach takes advantage of an ontology's ability to share and integrate knowledge within other ontologies. Thus, the iptables and TCP-Wrapper ontologies are reused to describe detailed HANAC countermeasure configurations.

A knowledge-base of best practice standards provide a basis for the generation and analysis of network access control configuration. Ontologies for best practice standards (e.g. [28]) for firewalls, Email servers, Web servers and XMPP servers [29][30][31] are developed [32][33]. Future research will consider additional best practice standards applicable for the HAN environment.

The advantage of taking a ontological approach to representing the semantic threat graph is that it provides a basis for extendibility, interoperability and complex composition of other security domains of interest based on the principles of Open World Assumption (OWA) [34]. For example, by including a model of an intrusion detection system within the semantic threat graph, one can more effectively reason about the HANAC recommendations been made based on both a top-down approach (best-practice standards) and a bottom-up approach (IDS rule signatures). In [33] an ontology engineering approach to the management of heterogeneous security access control configuration is considered.

#### *1)* Automated HANAC Configuration Synthesis

Synthesis of an appropriate access control configuration relies on the existence of a knowledge-base of candidate HANAC access-control rules that are consistent with the high-level security requirements of each HAN service. These could, e.g., represent considered best practice for the HANAC (e.g., firewall best practice [28]) that protect HAN services, e.g. IPTV-based services.

The following is a generic SWRL rule that examines the threats (*?threat*) and vulnerabilities (*?vul*) that each HAN service (*?srvc*) has, and searches for suitable countermeasures (*?rule*) that may be implemented by the HANAC gateway(*hanacGW*).

*HANService(?srvc)* ^ *HANSecService(*`hanacGW`*)*
*^Threat(?threat)* ^ *Vulnerability(?vul)* ^
*Countermeasure(?rule)* ^ *hasWeakness(?srvc, ?vul)*^
*threatens(?threat, ?srvc)* ^ *exploits(?threat, ?vul)* ^
*mitigates(?rule, ?vul)* ^ *protects(*`hanacGW`*, ?srvc)*
→ *implements(*`hanacGW`*, ?rule)*

As knowledge about assets, threats and vulnerabilities become known, it becomes possible to consider automatic synthesis of HANAC access-control rules as a basis for the previous SWRL rule. The following SWRL rule fragment will automatically populate the knowledge-base with a set of iptables firewall rules (using built-in *swrlx:makeOWLIndividual*), which considers a IPTV service provider HANAC whitelist. Knowledge about an IPTV service's IP address (variable *?iptvip*) and the source IP addresses in which the threat of not providing intended IPTV service provider access (*Threat(?noIPT V Access)*) is used to synthesise specific firewall rules (*?iptr*) from a template iptables rule (`iptrtemp`).

*HANService(?iptv) ^ Threat(?noIPTVAccess)^*
*Vulnerability(?noIPTVAllowRule)^*
*TemplateIPTRule(*`iptrtemp`*)^*
*hasWeakness(?iptv, ?noIPTVAllowRule)^*
*exploits(?noIPTVAccess, ?noIPTVAllowRule)^*
*mitigates(*`iptrtemp`*, ?noIPTVAllowRule)^*
*hasThreatSource(?noIPTVAccess, ?tip)^*
*hasIPAddress(?iptv, ?iptvip) ^ hasPort(?iptv, ?iptvp)*
*swrlx:makeOWLIndividual(?iptr,* `iptrtemp`*,*
*?noIPTVAccess, ?iptv)*
*→ IPTRule(?iptr)^*
*hasChain(?iptr, forward)^*
*hasSrcIPAddress(?iptr, ?tip)^*
*hasDstIPAddress(?iptr, ?iptvip)^*
*hasDstPort(?iptr, ?iptvp)^*
*hasAction(?iptr, accept)^*
*mitigates(?iptr, ?noIPTVAllowRule)*

## VI. An Initial UPnP Capability Sharing Performance Evaluation

In order to evaluate performance and scalability of the proposed approach an instrumented emulation testbed based on our prototype was implemented. This approach was used to create an isolated test environment with control over the parameters that affect the system's behavior such as the number of UPnP devices deployed in the HANs, the number of services per device and their duration in the network, the mix of mobile and stationary devices and the frequency of search requests. The emulated environment is symmetrical in the sense that all HAN gateways have the same distribution of load in terms of numbers of devices, search requests, federation partners and so on; however each HAN differs in the hardware they execute on. For this initial study only a single evaluation parameter was considered: the load on gateway CPU – i.e., the processing overhead caused by deploying the federated capability sharing system.

### *1) Device Emulation*

A device emulator is an entity that implements the UPnP specification, which can be configured to load a number of services and embedded devices. When a search request is received by the device emulator, it responds with search response packets for each supported service corresponding to the service type in the request. Based on its configured duration the device emulator sends presence announcements. The device emulator listens to http description requests and responds with an XML description document. It is expected that two types of devices will connect to the home UPnP network: stationary devices and mobile devices. Stationary devices do not regularly leave and rejoin the network e.g. UPnP enabled refrigerator, UPnP TV, residential gateway, etc. The UPnP device architecture document recommends1800 seconds as the minimal duration and recommends stationary devices to have much longer duration such as a day. In the emulation stationary devices are assigned duration from {900, 1800} seconds depending on the iteration, in order to increase the stress on the system due to more packets being announced on a shorter time. Mobile devices such as mobile phone, laptops, etc. are expected to be less reliable; they can join and leave the network frequently and their duration is much shorter. Mobile devices can potentially overload the system much more than stationary devices, as they require much more frequent updates and communication between remote home networks. The emulated UPnP network is created with a mix between stationary and mobile devices with various durations in order to evaluate the behavior of the system under different types of stress. Stationary devices are constructed in the beginning of the experiment and announce their presence in regular intervals. Mobile devices are constructed in the beginning of the experiment but are short lived and once their duration expires they are immediately replaced by another mobile device with random short duration in order to maintain a fixed number of mobile devices in the network.

### *2) Control Point Emulation*

A control point emulator is a UPnP client application that sends periodic search requests. The simulator selects randomly a service type to search from a predefined set. Every predefined interval the control point sends 3 search requests to the local UPnP network with a short wait between them. For all received search responses that represent a root device, a description request is initiated. The purpose of the control point simulation is to increase the stress on UPnP network and evaluate it under relatively extreme conditions. Each search request requires all devices in the network that support the service type to respond with a search response. In a multi home UPnP network this requires the virtual device to respond on behalf of all remote devices that support the requested service type.

### *3) Experimental Setup*

Figure 3 shows the experimental setup. Each gateway is represented by a desktop machine running an instance of the system as well as an instance of the UPnP device and control point emulations. The three home networks are connected to an XMPP server that runs on a remote host.

TABLE I. Hardware Used in Experiment

| No. | Gateways (Desktop machines with Linux Ubuntu 10.04) | | |
|---|---|---|---|
| | ***Processor(s)*** | ***Cache*** | ***Memory*** |
| 1 | Intel Pentium 4, 2GHz | 512KB | 1GB |
| 2 | Intel Pentium 4, 2x3GHz | 2MB | 2GB |
| 3 | Intel Pentium 4, 2x3.2GHz | 2MB | 2GB |
| | **XMPP Server (Macbook Pro with OS X 10.5)** | | |
| 1 | Intel Core 2 Duo,2.6GHz | 6MB | 4GB |

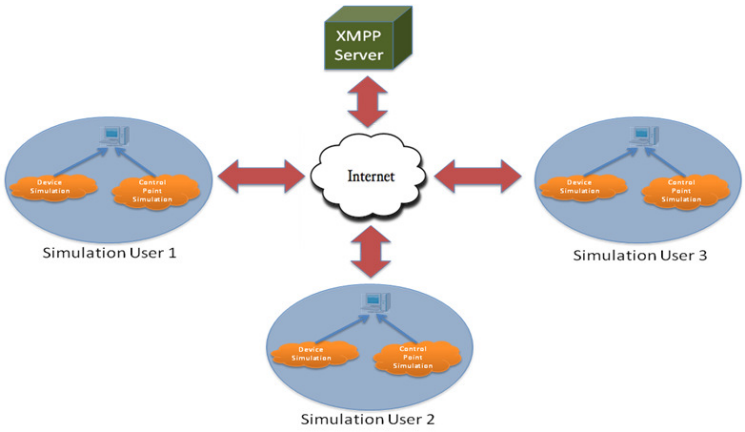


Figure 3: Experimental setup

*4) Experiment*

The experiment runs three home network instances on three desktops described above communicating with each other through a remote XMPP server over the Internet.

The experiment involved 11 iterations such that each iteration evaluates 4 different emulated network settings with a fixed number of devices. The variation between the 4 configurations is in the mix between stationary and mobile devices and in the duration of stationary devices. The number of devices grows from 5, 10, …, 50 and increased by 5 in each iteration with an additional iteration with 100 devices per network. Each of the devices is shared with all other users therefore if the number of devices in the local network is 5, the number of shared remote devices is 10, totalling to 15 devices visible to the local network user. Each iteration runs 4x60 minutes scenarios such that:

(i) Scenario I: Device emulation with 80% stationary devices with duration=900 seconds and 20% mobile devices. Mobile device emulators are created with duration randomly selected from the set (in seconds) {60, 120, 300, 450, 600}. Once the duration expires the device announces *byebye* and is replaced with a new dynamic device emulator.

(ii) Scenario II: Similar to scenario I with stationary device emulators created with duration=1800 seconds.

(iii) Scenario III: Device emulation with 60% stationary devices with duration=900 seconds and 40% mobile devices.

(iv) Scenario IV: Similar to scenario III with stationary device emulators created with duration=1800 seconds.

The Control point emulator is configured to send 3 search requests every 30 seconds such that the service type for the request is selected randomly from a set of 10 service types with probability of 0.5 to select *ssdp:all*. The rationale behind that is to increase the system stress as this request must be responded with all UPnP services.

*5) Performance Analysis*

The results shown in this section are based on averaging the results of the 4 scenarios described above with a fixed number of devices across scenarios and variable mix between stationary and mobile devices and variable duration for stationary devices.

The purpose of analyzing the CPU behavior is to verify that enabling the user to share devices with remote buddies does not come at too high cost in terms of processing overhead. In addition, since the system is targeted for home networks, it should be shown that it does not require high-end processing power. Figure 4 shows the average CPU behavior across iterations. Desktop 1 is a much inferior machine than Desktop 2 and 3 therefore it is not surprising that its CPU grows much higher than the others. It can be seen however that even with an extreme number of devices per network such as 100 (which means 100 local devices + 200 remote devices announced locally) the average CPU remains fairly reasonable. With 25 devices in the local network, Desktop 2 and Desktop 3 are still ~2% CPU while Desktop 3 is ~8% and with 100 devices Desktop 2 and Desktop 3 are up to 6.4% and 11% respectively while Desktop 1 is on 25.5% which is still relatively low considering the extreme stress.

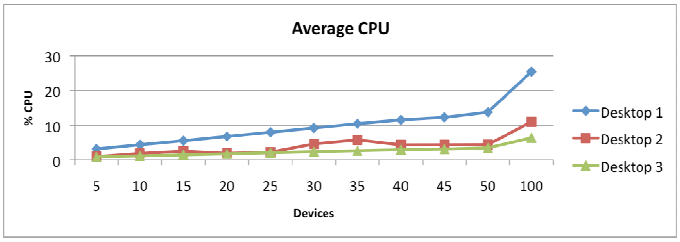


Figure 4: Average Load on HAN Gateway CPU

It must be noted that these initial results only pertain to global sharing/non-sharing decisions on the basis of whether or not another HAN is a "friend" i.e. federated with the current HAN. However the basic granularity of sharing configuration supported by the XMPP-UPnP testbed is on a device/service/action basis. If a device is shared, the remote network will be able to request the device description, however if no services are shared for that device with the remote network then the description will not contain information about services. The same logic applies to sharing configuration of actions, such that unless an action is shared, it will not appear in the service description delivered to the remote network. The sharing policy is checked with each discovery event, such as device advertisement received, service advertisement receive, description requested by remote network. Policy is also checked when an action execution request is received.

The additional overhead related to enforcement of access control policy is twofold: 1) the amount of time taken by the access control system to reason whether a capability should or should not be shared with a given remote network. 2) the processing overhead for filtering control data and content that should not be accessible to remote network. In future work we plan to investigate the relationship between reasoning complexity and processing/enforcement complexity for typical system scenarios.

## VII. Conclusions & Future Work

The work described here has progressed our understanding of the research questions laid out at the start of this paper as follows:

We have investigated the approach of defining new looser, semantic models of HAN capabilities and threat-graph based models for automatically generating access control rules on both those capabilities and any HANAC-defined network resource, e.g. a specific protocol or port. These ideas have been tested by describing the specific capabilities flexibly shared by our UPnP over XMPP system and deploying them for an access control scenario.

We have outlined the approach to leverage capability models, threat modes and known best practice guidelines in HANAC configuration to automatically generate policy rules to best protect a HAN with minimum security expertise required on behalf of the user.

Capability models themselves have previously been shown to provide a flexible and expressive means of applying access control to capability sharing in federal relationships without requiring all of the parties to support pre-agreed common policy or information models. It is hoped that further test-bed

experiments will evaluate the extent to which the semantic capability graphs enable semantic interoperability by allowing a range of service definition approaches to be combined and the relative costs of providing policy-based access controls at different levels of granularity of UPnP sharing.

In addition we hope to leverage our semantic HAN models to build dynamic visualizations of HAN activities at multiple levels of abstraction that will aid in HAN behavior comprehensibility for end-users. Future research will also consider self-optimisation and self-healing in conjunction with self-configuration.

## REFERENCES


[1] Chowdhury, R., Arjona, A., Lindqvist, J., Ylä-Jääski, A.: "Interconnecting multiple home networks services," International Conference on Telecommunications (ICT 2008), pp. 1-7, June 2008.

[2] Bottaro, A., Gérodolle, A., Lalanda, P.:"Pervasive Service Composition in the Home Network", 21st International IEEE Conference on Advanced Information Networking and Applications (AINA-07), Niagara Falls, Canada, May 2007.

[3] Brennan, R., Lewis, D., Keeney, J., Etzioni, Z., Feeney, K., O'Sullivan, D., Lozano, J. A. and Jennings, B.: Policy-based Integration of Multi-Provider Digital Home Services. IEEE Network, Nov/Dec, 2009

[4] Etzioni, Z., Feeney, K., Keeney, J., O'Sullivan, D.: "Federated Homes: Secure Sharing of Home Services", to appear in Proc. IEEE Consumer Communications and Networking Conference (CCNC11), 9-11 January 2011, Las Vegas, Nevada, USA

[5] Feeney, K., Brennan, R., Keeney, J., Thomas, H., Lewis, D., Boran, A., O'Sullivan, D.: "Enabling Decentralised Management through Federation", to appear in Elsiver Computer Networks 2010

[6] Novak, J.: "Helping Knowledge Cross Boundaries: Using Knowledge Visualization to Support Cross-Community Sensemaking", in Proc. of the Conference on System Sciences, HICSS-40, Hawaii, January 2007

[7] Russell, D.M, Maglio, P.P., Dordick, R., Neti, C.: "Dealing with ghosts: Managing the user experience of autonomic computing", IBM Systems Journal, Vol 42, No.1, 2003.

[8] Barrett, R., Maglio, P.P., Kandogan, E., Bailey, J.: "Usable autonomic computing systems: The system administrators' perspective", Advanced Engineering Informatics Vol 19, No, 3, July 2005

[9] Salehie, M. and Tahvildari, L. 2009. Self-adaptive software: Landscape and research challenges. ACM Trans. Auton. Adapt. Syst. Vol 4, No 2, May. 2009

[10] Barrett, R.: "People and Policies: Transforming the Human-Computer Partnership", in proc 5th IEEE int'l workshop on policies for distributed systems and networks (Poilicy'04), 7-9 June 2004

[11] Wack, J., Cutler, K., Pole. J., "Guidelines on Firewalls and Firewall Policy: Recommendations of the National Institute of Standards and Technology". NIST-800-41, 2002.

[12] Yong Lee; H., Won Kim, J.,: "An Approach for Content Sharing among UPnP Devices in Different Home Networks," IEEE Transactions on Consumer Electronics, vol.53, no.4, 2007.

[13] Kang, D., Kang, K., Choi, S., Lee, J.: "UPnP AV architecture multimedia system with a home gateway powered by the OSGi platform", IEEE Transactions on Consumer Electronics, vol. 51, no. 1, 2005

[14] Kim, J., Oh, Y., Lee, H., Paik, E., Park, K.: "Implementation of the DLNA Proxy System for Sharing Home Media Contents", IEEE Transactions on Consumer Electronics, Vol. 53, No. 1, 2007

[15] López de Vergara, J.E., Guerrero, A., Villagrá, V. A., Berrocal, J.: Ontology-Based Network Management: Study Cases and Lessons Learned. J. Network and Systems Management Vo 17, No 3, Sept 2009

[16] Feridun, M., Tanner, A.: "Using Linked Data for Systems Management". In Proc. NOMS 2010.

[17] Nurmela T., Kutvonen. L., "Service level agreement management in federated virtual organizations". In Distributed Applications and Interoperable Systems (DAIS 2007) Paphos, Cyprus, June 2007

[18] Saint-Andre. P., "Extensible messaging and presence protocol (xmpp): Core. IETF", Oct 2004,..http://www.ietf.org/rfc/rfc3920.txt {Online}

[19] Roman, D, Keller, U, Lausen, H, de Bruijn, J, Lara, R, Stollberg, M, Polleres, A, Feier, C, Bussler, C and Fensel, D: "Web Service Modeling Ontology", Applied Ontology, vol 1, no 1, 2005

[20] Bizer, C., Heath, T., Berners-Lee, T., "Linked data - the story so far," International Journal on Semantic Web and Information Systems (IJSWIS), 2009.

[21] Noy, N.F., "Semantic integration: a survey of ontology-based approaches" SIGMOD Rec., Vol. 33, No. 4, Dec 2004

[22] Fitzgerald, W.M., Foley. S.S., "Aligning Semantic Web Applications with Network Access Controls". International Journal on Computer Standards & Interfaces, October 2009.

[23] Fitzgerald, W.M., Foley. S.S., Ó Foghlú. M., "Network Access Control Configuration Management using Semantic Web Techniques." Journal of Research and Practice in Information Technology, Vol 41 no 2, May 2009.

[24] Netfilter. A framework that enables packet filtering, network address translation and packet mangling. http://www.netfilter.org

[25] Venema W., "TCP Wrapper: Network monitoring, access control, and booby traps". 3rd UNIX Security Symposium, Sept 1992.

[26] Foley, S.N., Fitzgerald. W.M.: "An Approach to Security Policy Configuration using Semantic Threat Graphs". IFIP WG 11.3 Working Conference on Data and Applications Security (DBSec), Canada, Jul 2009

[27] Hernan, S., Lambert, S., Ostwald, T., Shostack, A.:. "Uncover Security Design Flaws Using The STRIDE Approach". *http://microsoft.com/*.

[28] Wack, J., Cutler, K., Pole, J.: "Guidelines on Firewalls and Firewall Policy: Recommendations of the National Institute of Standards and Technology". NIST-800-41, 2002

[29] Miller, J., Saint-Andre, P., Hancke, P.: "XEP0220:Server Dialback". *http://xmpp.org*, March 2010.

[30] Saint-Andre, P.:. "XEP-0205: Best Practice to Discourage Denial of Service Attacks". *http://xmpp.org*, January 2009.

[31] Saint-Andre, P., Millard. P.: "XEP0178: Best Practices for Use of SASL EXTERNAL with Certificates". *http://xmpp.org*, Feb 2010.

[32] Fitzgerald. W.M "An Ontology Engineering Approach to Network Access Control Configuration". PhD Dissertation, Department of Computer Science, University College Cork, Ireland, August 2010.

[33] Fitzgerald, W.M., Foley, S.N.:. "Management of Heterogeneous Security Access Control Configuration using an Ontology Engineering Approach". ACM Workshop on Assurable and Usable Security Configuration, Chicago, USA, October 2010

[34] Baader, F., Calvanese, D., McGuinness, D.L., Nardi, D., Patel-Schneider, P.: "The Description Logic Handbook: Theory, Implementation and Applications". Cambridge University Press, 2003.