# A Lightweight and Identity-based Network Architecture for the Internet of Things

Pedro Martinez-Julia
*Department of Communication*
*and Information Engineering*
*University of Murcia*
*30100, Murcia, Spain*
*Email: pedromj@um.es*

Antonio F. Skarmeta
*Department of Communication*
*and Information Engineering*
*University of Murcia*
*30100, Murcia, Spain*
*Email: skarmeta@um.es*

***Abstract*—The rising of the Internet of Thing has imposed new challenges for the Future Internet to cover the specific requirements in terms of simplicity, low-power, and addressing flexibility. In this paper we present an architecture that allows network elements (*things*) to reach each other in terms of their identities instead of addresses or flat labels. Thus, the network entities are addressed in a *natural* way from the knowledge the entities have about each other. This way we achieve a flexible, scalable, and sustainable naming for the entities. Finally, the *Identity-to-Identity* communication (session) is instantiated in top of a lightweight identifier-based network, so the sessions are the junction between the underlying transport network and the identity-based network architecture.**

***Keywords*-Identity; Network; Architecture; Internet of Things**

## I. Introduction

The current Internet model was designed to let distant networks to be interconnected without monopolizing the underlying communications infrastructure while delivering improved reliability and fault tolerance. Today, it has become the common infrastructure used in our day-to-day communications, the place where we live our digital lives.

Over time, this model has exposed many problems, as described in [1], which have been adopted as challenges for the Future Internet (FI). The most important and widely discussed challenges are the decoupling of location (IP addresses) and identification (identifiers, identities), the scalable mobility support, and, of course, the integrated security.

Moreover, the enormous growth of the Internet, magnified by the rising of the Internet of Things [2], urges to abandon the IP addresses as the mechanism to identify the communication participants (endpoints). This reinforces the necessity of the separation of identification and location (id/loc split).

As discussed in Section II, the current proposals for id/loc split use plain identifiers to identify communication endpoints and locators (addresses) to deliver the information. Thus, they effectively decouple the identification from the location but they do not cover the specific requirements imposed by the Internet of Things (IoT) and thus reflect the wide object space of the future. For instance, IoT requires that the communications are lightweight because of the limited resources of the endpoint network devices (things).

To efficiently address this problem, and paying special attention in the flexible and reliable identification of communication participants, in this paper we discuss an architectural design that goes beyond id/loc separation to allow network entities of the FI to reach each other by means of their identity instead of network addresses or plain identifiers. As introduced in [3], [4] and particularized in [5], the architecture brings a single, evolutionary, and integrated approach. It is based on the definition of an *identity plane* built on top of an overlay network to be instantiated on top of different underlying networks while providing mobility support, maintaining the security of communications, and ensuring the privacy of network entities.

The remainder of this paper is organized as follows. In Section II we introduce the related work and discuss why it does not cover the aforementioned requirements. In Section III we describe the architecture design. Then, in Section IV we describe the experimental environment and procedure we used to evaluate the architecture. In Section V we show the obtained results. Finally, in Section VI we conclude the paper and give some points about the next steps of our work.

## II. Related Work

As discussed above, the search towards the Future Internet (FI), together with the rising of the Internet of Things (IoT), has exposed many challenges [1], outstanding the id/loc split as a key for the FI. We can find some approaches that try to meet these challenges and fix the issues they reveal.

On the one hand, the Locator-Identifier Separation Protocol (LISP) [6] achieves the id/loc split with a map-and-encapsulate scheme on top of the IP architecture. On the other hand, the Host Identity Protocol (HIP) [7] approaches the id/loc split by using a public key security infrastructure to disseminate the cryptographic host identifiers to be used by applications instead of IP addresses. Also, the BLIND architecture [8] leverages identity protection to HIP.

These proposals are integrated to the current Internet model. Other architectures propose to achieve the id/loc split as a complete architecture revamp, commonly called *clean-slate* approaches. The EMILSA architecture [9] is a

978-0-7695-4684-1/12 $26.00 © 2012 IEEE
DOI 10.1109/IMIS.2012.101

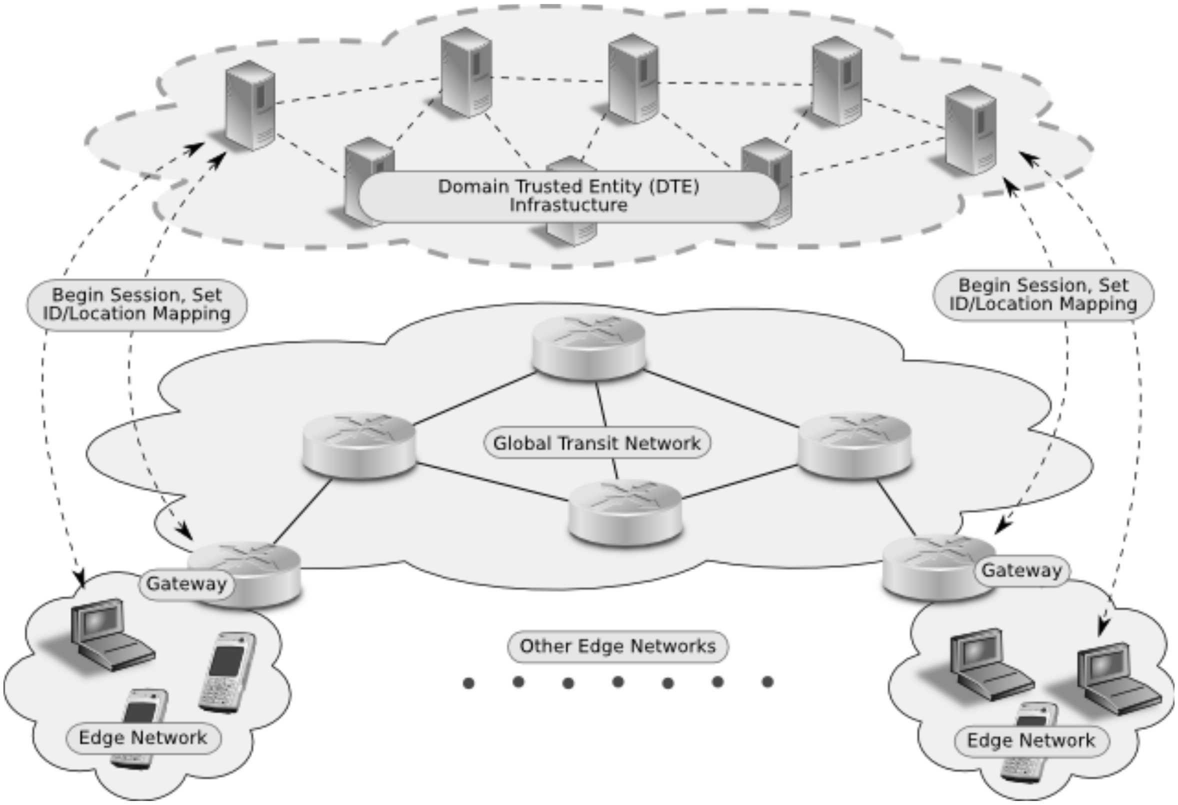

Figure 1. Architecture overview. The nodes from different edge networks communicate through the gateways. The sessions are negotiated through the Domain Trusted Entity (DTE) infrastructure, which updates the id/loc mappings to the necessary gateways.

*clean-slate* proposals to provide id/loc split with URI-like identifiers and other advantages over other proposals.

Moreover, being halfway between an evolutionary and *clean-slate* approach, we can find the HIMALIS architecture [10], which approaches id/loc split like in LISP and HIP but with a totally new approach that provides some advantages over the current Internet model, specifically considering the low-power devices as a target endpoint node.

Although these proposals properly address the id/loc split, they are not as lightweight as necessary to cover the requirements imposed by the IoT. The HIMALIS architecture is near to cover this objective but it lacks to provide a flexible discovery mechanism, so the endpoint network nodes need to deal with many network operations, and this is not desirable in IoT environments. Also, they do not integrate the security to the network layer. These properties, as described above, are well established as challenges for the Future Internet.

## III. Proposed Architecture

To achieve a lightweight and secure architecture with integrated discovery we designed the necessary mechanisms to build an *identity plane* that supports the whole communication together with the typical planes (data, control, and management). It removes as much functionality as possible from the client (endpoint) nodes, so low-power are well supported. Moreover, the clients are reached by their identity, so they communicate directly on the edge networks. To locate them in the global network, the clients are represented by the gateway to which they are connected, which maps the identities to network locators and vice-versa. Finally, when designing the architecture, we considered the infrastructures used in application communications (messaging infrastructures), so we reduced the necessary layers to perform the same operation in our architecture.

The identity plane is a mechanism that lets network entities to negotiate the different communication aspects in base of the identity of the communication participants. It concentrates the necessary functionality to allow network entities to reach each other by means of their identities instead of network addresses. Thus, the communication sessions are negotiated through this plane, including the identity search and the security aspects.

To build the identity plane, we need a special entity to offer the mentioned functionality to other networked entities. Thus, as shown by Figure 1, our architecture introduces the Domain Trusted Entity (DTE). It is used to build a global infrastructure (DTEi) that holds and manages the identity information pertaining to the network entities and to offer the negotiation mechanisms commented above. It is used by network elements as well as other entities to unequivocally identify them. Also, when necessary and if permitted by policies, the DTEi may provide some attributes from the identities to the network elements in order to know some aspect of the entities.

Moreover, our architecture extends the network entity definition to include people, software (services), hardware (machines), and many other things (Internet of Things [2]). This way, a single device may represent different entities without needing an additional mechanism to differentiate them.

### A. Identifiers and Identities

The architecture emphasizes the conceptual difference between *identity* and *identifier*. In contrast with other interpretations, we consider that an identity is a collection of attributes that describe an entity, so it is more than just a form of identify it. Thus, as introduced above, an entity may be a person, machine, service, or any other element with an identity in the digital world.

On the other hand, an identifier is a piece of fixed-size data used to identify something [11]. In a general sense, our architecture uses dynamic identifiers to determine the endpoints of the communication participants and their sessions. Nevertheless, in order to guarantee privacy, the identifiers can not be used to link a communication to an identity or its associated entity.

### B. Security and Trust

The security properties of the architecture are twofold. First, the identity plane and consequently the DTEi must be built with high security and trust requirements because they work with delicate information. Second, as one of the key objectives of the architecture is to negotiate the security aspects of the communications, it includes security mechanisms for them.

To keep the confidentiality of the messages exchanged through the DTEi, they are encrypted with a key obtained with the Diffie-Hellman method. Moreover, to ensure that the whole infrastructure is trusted, each DTE provides a digital certificate in the key exchange, which is emitted by a globally trusted authority. Fake DTEs may form part of the DTEi but they can not be involved in *trusted* negotiations.

About the *identity-to-identity* communications security, the mutual authentication of entities is achieved by the nature of communications provided by the architecture. Thus, when required, an entity may be sure that other entity is *who* is claiming to be. It is realized by verifying the session identifiers against the DTEi. To get this, the message integrity must be enforced, so each message incorporates a signature-like field made with a key that is negotiated during the session establishment.

In addition, our architecture proposes and recommends to use an asymmetric encryption mechanism when desired to gain confidentiality. It may be inefficient and processor hungry but with some benefits over weaker encryption mechanisms: 1) Transmitted information will be kept secret for longer; 2) There is no need to negotiate the security terms, with the speed-up it represents; 3) Fits perfectly and performs much better in publisher/subscriber underlying networks. In the future, processor performance improvements may make those methods much less processor hungry in comparison with other tasks. This does not prevent to adapt and use other weaker encryption mechanisms (primarily symmetric) and key exchanges protocols, such as IKEv2 [12].

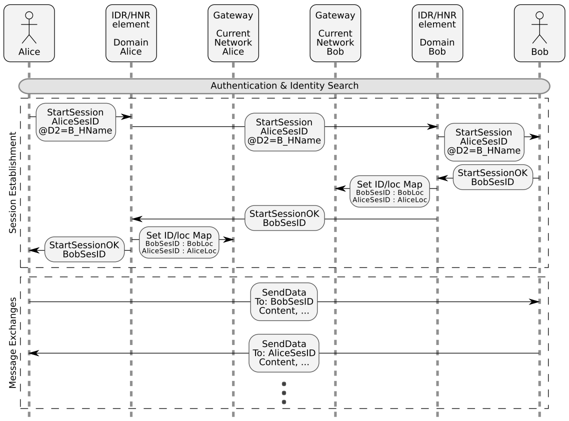
Alice
IDR/HNR element Domain Alice
Gateway Current Network Alice
Gateway Current Network Bob
IDR/HNR element Domain Bob
Bob
Authentication & Identity Search
Session Establishment
StartSession AliceSesID @D2=B_HName
StartSession AliceSesID @D2=B_HName
StartSession AliceSesID @D2=B_HName
StartSessionOK BobSesID
Set ID/loc Map BobSesID : BobLoc AliceSesID : AliceLoc
StartSessionOK BobSesID
Set ID/loc Map BobSesID : BobLoc AliceSesID : AliceLoc
StartSessionOK BobSesID
Message Exchanges
SendData To: BobSesID Content, ...
SendData To: AliceSesID Content, ...

Figure 2. Message exchanges to start a session.

### C. Identity-based Communication Sessions

Before end nodes can communicate they need to start a session. To do it, they need to negotiate, through the DTEi, the session identifiers used by each. The DTEi provides strong security to those negotiations. Figure 2 shows the messages exchanged to start a session between two network end nodes (Alice and Bob), which actually represent the devices used by the actual entities. It works as follows:

1) Alice sends to the DTE instance of its domain a message called *StartSession* which contains its automatically generated session identifier and a query to find Bob by means of its identity attributes.
2) The DTE of Alice's domain sends a request to the DTE element of the other domain (Bob's domain, "D2") with the information provided by Alice. The DTE instances are reached each other through the overlay network without needing to resolve any name, just using their domain identifier.
3) Now, the DTE of "D2" checks the corresponding policies set for Bob's identity and searches the entity that responds to it. Then it sends the *StartSession* request to Bob because it responds to such identity.
4) Bob accepts the session, keeps the session identifier specified by Alice, and sends a *StartSessionOK* message with its own session identifier to its corresponding DTE instance.
5) Once the DTE of Bob's domain has the two session identifiers and locators of Alice and Bob, it reports to the current gateway to which Bob is connected

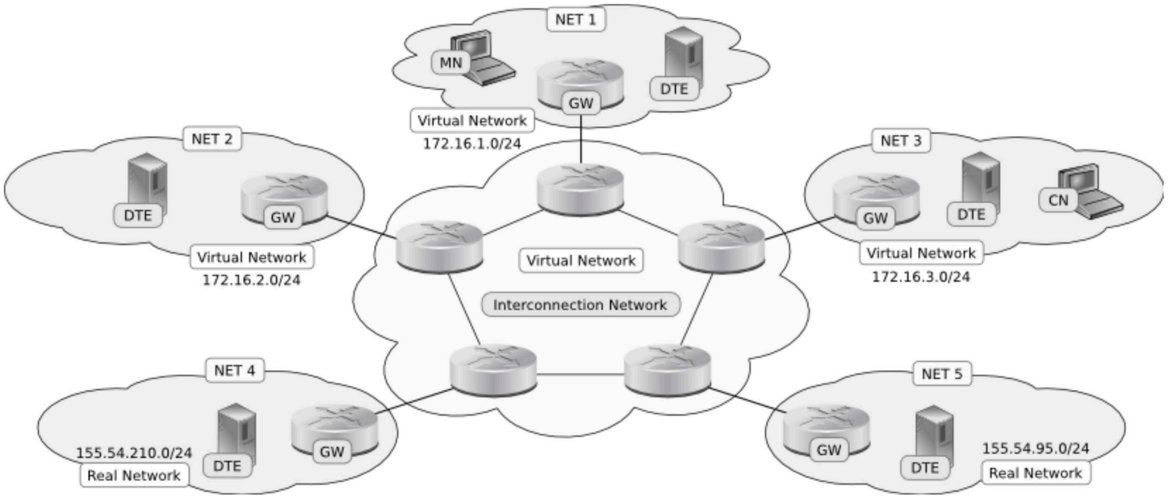

Figure 3. Experimentation environment. The elements of the architecture are labeled GW and DTE, corresponding to the Gateway and Domain Trusted Entity. The MN and CN are the Mobile Node and Correspondent Node respectively. The elements of the real networks are configured to find the virtual networks through their respective GWs.

to set new mappings for the two session identifiers with their corresponding locators (BobSesID, AliceSesID).

6) After setting the mapping to the gateway, The DTE of Bob's domain sends a *StartSessionOK* message to the corresponding DTE of Alice's domain.
7) As did the DTE of Bob's domain, the DTE of Alice's domain sends a message to set the id/loc mappings to the current gateway to which Alice is connected.
8) Finally, the DTE of Alice's domain sends the *StartSessionOK* to Alice and the session is considered started, so Alice begins to sends messages to Bob. These messages are intercepted by the gateway that use the mapping to know the location of Bob, that is where it delivers the messages. The same happens when Bob sends messages to Alice.

### D. Gateways

The gateways, as introduced in the beginning of this section, are the network elements that permit the end nodes to reach each other in the global network. Thus, the gateways need to know the specific locator of each node in order to transmit the messages/packets through the global transit network (Internet). As shown in Figure 2, the DTEi will report the specific locators of the communication participants to the corresponding gateways.

## IV. Experimentation

Once we have described the architecture we prepared an experimental implementation to demonstrate its behavior. In the following subsections we describe the implementation, the testbed, and the experiment performed with them.

### A. Proof-of-concept Implementation

The implementation of the architecture consists of many base components and a few final applications to perform the experiments described below. First, we built the DTEi and the clients. Also, we implemented a simple gateway instance with id/loc mapping support to receive ID-based messages and transmit them to other gateway or to a client, depending on the location of the client. All components are implemented in Python but the critical modules are built to binary using Cython.

The DTEi is built through an overlay network that uses the Chord [13] routing algorithm with the only optimization of the finger table that shorts the process of finding far nodes in the overlay. Therefore, although some performance improvements are certainly possible to be applied to the communication of the DTEi, as we show in [4], in this paper we focus on the secure identity-to-identity communication and the mobility support of our architecture applied as an id/loc mapping system.

The client application representing the Mobile Node (MN) asks for a number of messages, specifying the rate at which it wants them, and the Correspondent Node (CN) sends the requested messages.

### B. Experimentation Environment

To perform the experiment we built a testbed consisting of 6 networks: a virtualized interconnection network which acts as the global transit network defined in the architecture, and 5 edge networks, two of which are real networks and the remaining three are virtual networks. All nodes are Virtual Machines (VMs) running in top of the Xen virtualization technology (paravirtualization), a widespread solution in high performance virtualization environments, like in many Cloud Computing infrastructures. All the equipment, virtual and physical, runs the Linux operating system. The virtualized networks are built using Linux kernel bridges in the host machines and using VTun [14] to build ethernet bridges through TCP/IP connections between separated host machines.

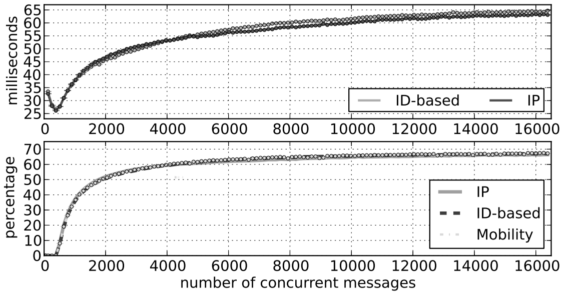

Figure 4. Average time evolution in milliseconds (top) and message loss percentage (bottom). The former compares the behavior of our architecture and IP and the latter compares the message loss percentage of IP, our architecture, and our architecture while a node is moving. As the plots are overlapped, Figure 5 and Figure 6 show the differences of our architecture with the base case (IP).

As shown in Figure 3, the topology of the experimentation environment has all the elements defined in our architecture (DTEs and GWs) together with the client nodes (MN and CN). The edge networks correspond to different network and identity domains. Thus, the MN and CN respectively belong to Domain 1 and Domain 3. As expected, regardless of whether they are connected to the real or virtual networks, the client nodes are configured to route messages through their corresponding GW which is configured to route them through the interconnection network.

### C. Experiments

In order to see the scalability and overall performance of the solution we first run an experiment to get the average time spent in transmit and receive a single message but at different message rates. We set the MN and CN to exchange messages at rates from 1 msg/s to 16384 msgs/s. It is run 30 times for each rate to get the average and the whole solution is run 10 times to get the standard deviation and standard error of the measurements. We also measure the loss rate for each message rate. Finally, we run a IP-based client to get comparable results.

After that, we run the experiment again but set the MN to change from its home network to other network after some messages to obtain the loss rate when the node is moving.

Finally, we select a specific message rate and make the MN to sequentially move to all the domains, each 5 seconds in counterclockwise. Thus, we can demonstrate the behavior with many handover events.

## V. Results

After running the experiment we get the results and illustrate them in some plots. First, Figure 4 shows the aforementioned results. The top subplot shows the average time per message exchange evolution and the bottom subplot shows the evolution of the loss percentage. As we can see, both plots show that our approach is close to the IP approach, either with or without the mobility event. For the average time per message exchange, we can see that it stays under 35 milliseconds (ms) for rates under 1000 messages per second, but do not surpass 65 ms for huge rates. This demonstrates the good scalability of our solution.

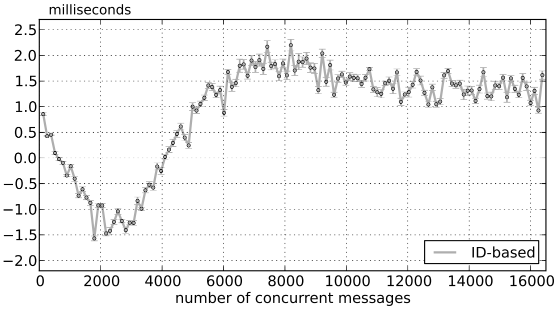

Figure 5. Evolution of the difference of the average exchange time of our architecture and IP. The error-bars represent the standard error of the original measures of the tests with our architecture.

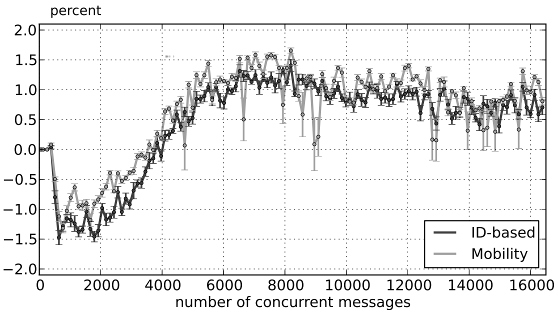

Figure 6. Evolution of the difference between the message loss of our architecture and IP, and between our architecture during a mobility event and IP. The error-bars represent the standard error of the original measures of the tests with our architecture.

Figure 5 shows the separation of the average time per message of our architecture and IP. It also shows the standard error of the measurements, depicted as error-bars wrapping the plot. On it, we can see that our approach does not differ from IP in more than 2.5 ms and that for some data rates it spends up to 1.5 ms less per each message exchange. In huge message rates, the difference is stabilized between 1 and 2 ms over IP, which is a very good result.

Figure 6 shows the difference of the loss percentage between our approach and IP, and between our approach with a mobility event and IP. Both plots also have the standard error of the percentage loss measurements wrapping the plot lines. The two plots are very similar in their global movings, but the mobility plot is more unstable and has higher standard error. In summary, our approach is far from raw IP in 1.5%, with 0% to 1.5% more message loss.

Finally, Figure 7 shows the net message rate between MN and CN for each moment of time, starting from 0 seconds and ending at 28 seconds, while MN is moving to its adjacent domain each 5 seconds. We can see the

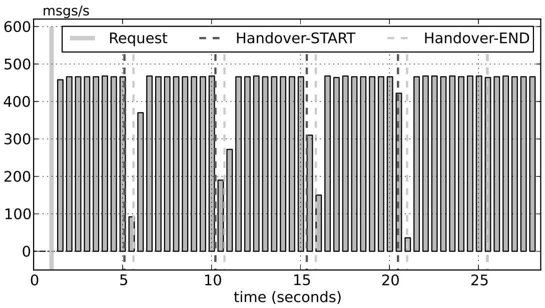

Figure 7. Evolution of the solution behavior as the mobile node moves through all domains while receiving messages at a rate of 500 msgs/s. The vertical solid line marks the time when the request is sent and the vertical dashed lines mark start and end of each handover process, whose respective timespan is 508 ms, 507 ms, 508 ms, 512 ms, and 4 ms.

moment in which MN sends the request and the moments in which it starts and ends the handover between those domains. As expected for the 500 msgs/s rate, the net rate is about 10%, but during the handover processes it looses some messages. These looses are sometimes compensated, as seen in the next bars close to the handover end events. The last handover, moving to the home domain, has almost negligible timespan (4 ms) because the network interface connected to the home domain is already prepared.

## VI. Conclusions and Future Work

In this paper we presented a lightweight network architecture that places identities as a central role for the Future Internet, specifically adapted for the workloads found in the Internet of Things. It achieves identity-to-identity networking while enforcing the overall security and privacy of the communication participants, which are identified by identities instead of addresses.

As the necessary mechanism used to achieve this objective, we presented the Domain Trusted Entity Infrastructure (DTEi), which manages the identities of communication participants in a trusted and secure way. Moreover, to validate the architecture we built a proof-of-concept and demonstrated the feasibility of such identity-based architecture in comparison with IP.

For the future work it is necessary to investigate the possibility of using plain user certificates managed by the DTEi to validate the exchanged messages so we can decentralize that task. Moreover, we will further continue with the investigation in the interactions with other overlay network models. Finally, we plan to integrate and validate the architecture against other architectures, as described in [15], in order to provide the identity-to-identity benefits to them.

## Acknowledgments

This work is partially supported by the European Commission's Seventh Framework Programme (FP7/2007-2013) project GN3, by the Ministry of Education of Spain under the FPU program grant AP2010-0576, and by the Program for Research Groups of Excellence of the Séneca Foundation under grant 04552/GERM/06.

## References

[1] T. Li, "Design Goals for Scalable Internet Routing," 2011, http://www.ietf.org/rfc/rfc6227.txt.

[2] A. C. Sarma and J. Girao, "Identities in the future internet of things," *Wireless Personal Communications*, vol. 49, no. 3, pp. 353–363, 2009.

[3] A. F. Gomez-Skarmeta, P. Martinez-Julia, J. Girao, and A. Sarma, "Identity based architecture for secure communication in future internet," in *Proceedings of the 6th ACM Workshop on Digital Identity Management*. New York, NY, USA: ACM, 2010, pp. 45–48.

[4] P. Martinez-Julia, A. F. Gomez-Skarmeta, J. Girao, and A. Sarma, "Protecting digital identities in future networks," in *Proceedings of the Future Network and Mobile Summit 2011*. International Information Management Corporation, 2011, pp. 1–8.

[5] P. Martinez-Julia and A. F. Gomez-Skarmeta, "Using identities to achieve enhanced privacy in future content delivery networks," *Computers and Electrical Engineering*, vol. 38, no. 2, pp. 346–355, 2012.

[6] D. Meyer, "The locator identifier separation protocol (lisp)," *The Internet Protocol Journal*, vol. 11, no. 1, pp. 23–36, 2008.

[7] R. Moskowitz and P. Nikander, "Host Identity Protocol (HIP) Architecture," 2006, http://www.ietf.org/rfc/rfc4423.txt.

[8] J. Ylitalo and P. Nikander, "BLIND: A complete identity protection framework for end-points," *Lecture Notes in Computer Science*, vol. 3957, pp. 163–176, 2006.

[9] J. Pan, R. Jain, S. Paul, M. Bowman, X. Xu, and S. Chen, "Enhanced milsa architecture for naming, addressing, routing and security issues in the next generation internet," in *Proceedings of the International Conference on Communications*. Washington, DC, USA: IEEE, 2009, pp. 14–18.

[10] V. P. Kafle and M. Inoue, "HIMALIS: Heterogeneity inclusion and mobility adaptation through locator id separation in new generation network," *IEICE Transactions on Communications*, vol. E93-B, no. 3, pp. 478–489, 2010.

[11] International Telecommunication Union, Telecommunication Standardization Sector, "Series X: Data Networks, Open system communications and security. Cyberspace security - Identity management. Baseline capabilities for enhancing global identity management and interoperability. Recommendation ITU-T X.1250," 2009.

[12] C. Kaufman *et al.*, "Internet Key Exchange (IKEv2) Protocol," 2005, http://www.ietf.org/rfc/rfc4306.txt.

[13] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan, "Chord: A scalable peer-to-peer lookup service for internet applications," in *Proceedings of the 2001 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*. New York, NY, USA: ACM, 2001, pp. 149–160.

[14] M. Krasnyansky, "Virtual Tunnels over TCP/IP networks (VTun)," 2011, http://vtun.sourceforge.net.

[15] P. Martinez-Julia, A. F. Gomez-Skarmeta, V. P. Kafle, and M. Inoue, "Secure and robust framework for id/locator mapping system," *IEICE Transactions on Information and Systems*, vol. E95-D, no. 1, pp. 108–116, 2012.