Journal of Universal Computer Science, vol. 18, no. 12 (2012), 1643-1661
submitted: 15/9/11, accepted: 15/2/12, appeared: 28/6/12 © J.UCS

A Novel Identity-based Network Architecture for Next Generation Internet

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

Antonio F. Gomez-Skarmeta
(Department of Communication and Information Engineering
University of Murcia, 30100, Murcia, Spain
skarmeta@um.es)

Abstract: In this paper we show a network architecture for Next Generation Internet (NGI) that prevents operation traceability and protects the privacy of communication parties while raising their identity to be a central element of the network. As a side effect, our architecture inherently supports authentication and mobility of the entities involved in the communication. Moreover, it is designed to be agnostic to any underlying network infrastructure and can be used to enhance them with reduced penalty, which makes it a perfect component to take its features to existing networks without defining a brand new transport layer. We also show the successful verification of the protocol security and demonstrate its feasibility and scalability showing its behavior when instantiated on top of two different architectures.
Key Words: Next Generation Internet, Identity, Privacy, Overlay Networks
Category: C.2.1, C.2.2, C.2.6, K.6.5

1 Introduction

The current Internet model was designed as a common infrastructure to let distant networks to be interconnected without monopolizing the underlying communications infrastructure while delivering improved reliability and fault tolerance. This model culminated with the global operation we have today, where communication parties are identified by means of their location dependant IP addresses. Today, this scenario still represents the communication model instantiated in the Internet.

Over time, the communication model described above has exposed many problems later adopted as challenges for the Next Generation Internet (NGI) as shown by [Jain, 2006] from the general perspective and later by [Li, 2011] from the routing perspective. 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. The original Internet design lacks these capabilities. However, there are many patch-on solutions that try to overcome those lacks for the current Internet model (see Section 2) but they do not meet these requirements satisfactorily and as a single/integrated solution.

Without the decoupling of location and identification, also called separation of identifier and locator (loc/id separation), the Internet model uses IP addresses in the whole protocol stack. Therefore, an IP address is used in the network layer protocols as device locator, the host attachment point to the network, to forward packets toward their destination. The same IP address is also used in the transport and upper layer protocols to identify the device in the network. This overloaded/dual role of IP addresses as identifier and locator makes difficult to design efficient solutions for mobility, multihoming, renumbering, and security because such solutions require the provision to dynamically change device locators at the network layer without changing identifiers at the transport and upper layers. In general, this overloading has limited the flexibility of the current Internet architecture.

The scalable mobility support challenge, which has some similarities with the decoupling of location and identification, requires to include in the Internet of the future the capability of a host, network, organization, etc. to change its topological connectivity with respect to the remainder of the Internet without loosing their existing sessions. The existing mechanisms, which are included as patches inside the current Internet model, are based on the renumbering of the mobile host/device and sometimes using a tunnel to keep using old addresses. The approaches using this mobility model restrict the dynamism because the existing network and transport sessions may be broken due to excessive handover time. Also, it introduces new security and privacy problems because the elements of a foreign network will be involved in the operation of a device. With privacy problem we mean the impossibility for an entity to decide what information reveal and to whom. This includes both operations, identity information, and content.

Finally, the original Internet design did not include security mechanisms. They have been introduced over time at different layers but, however, the basic security and privacy requirements at network layer are not met because it is not mandatory and a network entity can not control its security when communicating with other network entity which do not forms part of the same security domain. Therefore, the integrated security and privacy protection is an important aspect for the Internet of the future and thus has become part of the NGI requirements as a key challenge. Moreover, both mobility support and decoupling location from identifiers also present security requirements in the management of the dynamic associations between locations and identifiers as well as in the security of the handover process.

In this paper we present a novel architecture (and protocol) that goes beyond current proposals to overcome the problems commented above by using identities, as defined by ITU-T X.1250 [ITU-T, 2009], to identify communication participants (network entities), by using dynamic identifiers, as also defined by ITU-T X.1250, not coupled but dynamically associated with network addresses to identify network sessions between entities and ensuring the scalable mobility and the prevention of operation traceability, and by using a globally trusted infrastructure to negotiate communication sessions, including their security and privacy aspects. The architecture is built on top of an overlay network that permits entities to reach each other without needing IP addresses, just using the session identifiers.

While the identity of each entity is kept fixed and protected, the identifiers used in communication sessions are disclosed and may dynamically change when needed by a handover event or requested by the entity. Thus, our architecture adds valuable capabilities to the network, such as user/peer identification, identity management, validation of identifiers (entity authentication), and encryption. We approached this issue in previous work [Gomez-Skarmeta et al., 2010, Martinez-Julia et al., 2011] that shows an identity based architecture on top of a distributed overlay network taken from a Distributed Hash Table (DHT), a common technology behind P2P networks. Furthermore, this work gets the concepts raised by the Secure Widespread Identities for Federated Telecommunications (SWIFT) project [López et al., 2009], which defines a framework for identity management and privacy protection for users of multiple identity providers, and leads them to the network.

The remainder of this paper is organized as follows. First, in Section 2 we discuss the previous work that approaches the same problems in one or another way. Then, in Section 3 we describe the architecture and in Section 4 we show its protocol. In Section 5 we discuss how to implement an evaluation proof-of-concept architecture on top different infrastructures and in Section 6 we show the results we obtained from the tests we performed with them. Finally, in Section 7 we discuss our conclusions and next steps of our work.

2 Related Work

As discussed in previous section, the search towards the Next Generation Internet (NGI) has exposed many challenges [Jain, 2006, Li, 2011], outstanding the loc/id separation as a key challenge for the NGI. We can find some approaches that try to meet these challenges and fix the issues they reveal.

From a pure loc/id separation perspective, we can find the Locator-Identifier Separation Protocol (LISP) [Meyer, 2008] that achieves the separation of locator and identifier with a map-and-encapsulate scheme that can be used with the IP architecture. Also, we can find the Host Identity Protocol (HIP) [Moskowitz and Nikander, 2006], which approaches the loc/id separation using a public key security infrastructure to disseminate the cryptographic host identifiers to be used by applications instead of the location-specific IP addresses. Related to HIP we can also find BLIND (A Complete Identity Protection Framework for End-points) [Ylitalo and Nikander, 2006], which leverages identity protection to HIP.

The proposals mentioned above are more or less integrated in the current Internet architecture, but there are other approaches that propose a complete revamp, commonly called *clean-slate* proposals. From this class we can highlight the Enhanced Mobility and multihoming supporting Identifier Locator Split Architecture for naming in the Next Generation Internet (EMILSA) [Pan et al., 2009], which proposes a complete architecture revamp to provide loc/id separation with many advantages over other current and future architectures. Moreover, being halfway between an evolutionary and *clean-slate* approach, we can find the Heterogeneity Inclusion and Mobility Adaptation through Locator ID Separation in New Generation Network (HIMALIS) architecture [Kafle and Inoue, 2010], which approaches loc/id separation like in LISP and HIP but with a totally new approach with less footprint to be used in low power devices such as sensor networks. It is being developed as part of the AKARI[1] architecture design project for New Generation Network [National Institute of Information and Communications Technology, 2010] of the NICT[2] as the reference architecture of Japan for the Internet of the future.

Although these proposals properly address the loc/id separation in one or another way, they do not provide mechanisms to prevent the traceability of the operations performed by the entities, so the privacy of these entities can be violated. Thus, these architectures lack in the integration of security to the network layer. Moreover, they do not separate the actual entities behind the communication from the devices they are using, so an entity is associated-to and identified-by its device. As commented above, these properties are widely accepted as essential challenges to be met in the NGI. Finally, we consider that the identities of the entities involved in communications should be treated in a special manner and the traceability of their operations should be prevented, thus protecting their privacy.

3 Proposed Architecture

As introduced in Section 1, the main contribution of this work is the design of an integrated architecture that fills the gaps found in current Internet model in

[1] AKARI codename comes from the lemma "A small light in the dark pointing to the future" — "AKARI" means "a small light" in Japanese.

[2] National Institute of Information and Communications Technology (NICT) (`http://www.nict.go.jp`).

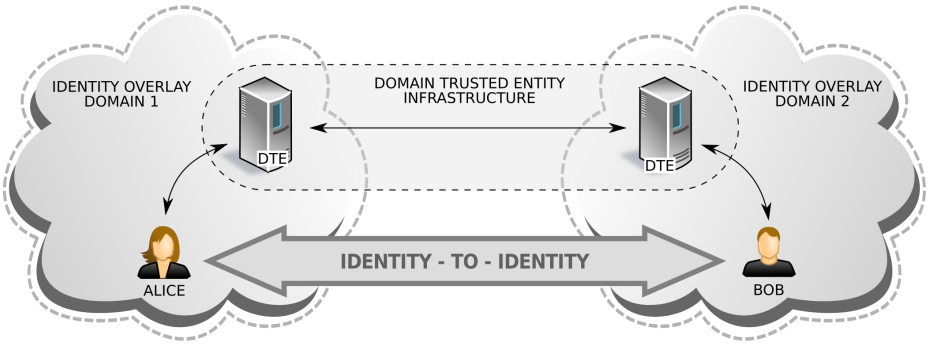

Figure 1: Architecture overview.

terms of loc/id separation, mobility support, and integrated security. These capabilities are delivered in an integrated manner with the concept of secure identity-based end-to-end communications (identity-to-identity) which, by means of the architecture described here, is carried to the network. Thus, apart of loc/id separation, it provides scalable mobility support and integrated security, with special attention to privacy and traceability prevention. In order to achieve this objective we build an identity overlay network whereby entities are addressed by their digital identity, instead of logical address of the device (or host) they use. This overlay network is then divided in many domains of trust which are independent of the actual networks. Each entity is associated with a domain and can have different devices connected to different physical or logical networks at the same time.

To achieve these capabilities, the architecture incorporates many elements and mechanisms. Figure 1 shows an overview of the architecture with its main elements, leaving out the lower layer networking infrastructure used by the devices of the communication parties. The most important elements of the architecture are the entities participating in the communication, which can be people, software (services), hardware (machines), etc. One special element is the Domain Trusted Entity (DTE). It manages the association of entities and identifiers for its domain and permits communication parties to be sure they are *talking* to *who* they want without revealing identity information. This is achieved by just asking the DTE to validate an identifier against a query of identity attributes, which is an abstract representation of an identity and thus a communication endpoint. It can also be used by other elements to obtain certain identity attributes if allowed by policies. The DTEs of different domains are connected forming an infrastructure that supports and protects the identity of the communication parties. Finally, the underlying network infrastructure is used to transmit low-level messages among communication parties.

In this architecture, the communications are established through endpoints that are used in message exchanges and are identified by location independent identifiers. If the underlying network is based on addresses, our architecture requires to allocate many addresses to be associated with the different identifiers which can be dynamically negotiated through the DTE infrastructure. Also, it permits to change any endpoint identifier at any time, so the mobility support is inherent and the privacy can be enhanced with arbitrary identifier renegotiation.

At the identity level, our architecture proposes to manage identities and build identifiers with Extensible Resource Identifiers (XRI) and Extensible Resource Descriptor Sequence (XRDS) [Reed et al., 2008]. XRI is used to build the identifiers and XRDS is used as resolution mechanism to dynamically associate the identifiers to an identity and vice-versa. This way, our architecture has a consistent and coherent identifier scheme that may be coupled with existing identity federation architectures, such as OpenID.

3.1 Identity and Identifier

This architecture emphasizes the differentiation of *identity* and *identifier*. We meet with the ITU-T definition of identity on its X.1250 recommendation as follows: The representation of an entity in the form of one or more information elements which allow the entity(s) to be sufficiently distinguished within context. For identity management purposes, the term identity is understood as contextual identity (subset of attributes), i.e., the variety of attributes is limited by a framework with defined boundary conditions (the context) in which the entity exists and interacts. Thus, each entity is represented by one holistic identity, which comprises all possible information elements characterizing such entity (the attributes). However, this holistic identity is a theoretical issue and eludes any description and practical usage because the number of all possible attributes is indefinite.

On the other hand, also meeting with its ITU-T definition, we consider an identifier to be a piece of fixed-size data that identify something. In a general sense, this architecture uses identifiers to determine the endpoints of communication parties, as well as to obtain information from the identity if permitted. Nevertheless, they are not used to unambiguously associate an identity to an object on time, just in certain moment and communication event.

3.2 Domain Trusted Entity

The aforementioned Domain Trusted Entity (DTE) is a special entity that manages and protects the communication aspects of its domain identities. Moreover, if allowed by the policies, it can reveal some identity attributes to other entities. Thus, the DTE is based or collaborates with other identity management technologies like Security Assertion Markup Language (SAML) and Shibboleth. In this architecture, the DTE stores XRDS documents that belong to its entities and which other entities may request. The XRDS document describes the services offered by an entity and how they can be contacted, their service endpoints. Therefore, the DTE plays the role of the XRI/XRDS resolution infrastructure in the OpenID architecture.

The DTEs are also used to validate that the identifier (or identifiers) used by an entity belongs to such entity. Thus, any communicating party can be sure that it is talking to the entities it wants to talk without knowing any attribute of the actual identities behind them. Again, this functionality is also controlled by policies, so some entities may decide to forbid the validation. Furthermore, when an entity requires anonymity, it may request an *anonymous identity* whose identifiers can be validated and whose attributes can be requested, but the actual information of the real entity is not disclosed in any manner.

Due to the high number of interactions and traffic that is presumably supported by each DTE, it should be constructed in a distributed manner. For instance, it can be constructed using technologies found in DHTs.

As the DTE infrastructure is the central element of our architecture we need to take special care about its security and trust. First, in order to ensure the confidentiality, the communication between entities and DTEs are encrypted with a key negotiated during the authentication, which can be renewed when necessary. This also applies to the communication between DTEs. To ensure a trusted path between DTEs, each DTE has its own public/private key pair that is distributed to the other DTEs when it is instantiated in the network. This may be, a priori, a heavy task but it is necessary to ensure the overall security. A PKI (Public Key Infrastructure) is used to maintain those keys and the global security. Putting all together, the DTE infrastructure offers the secure and trusted channel required by the architecture and protocol to operate correctly.

3.3 Underlying Network

This architecture needs, in one way or another, a special underlying network infrastructure that is capable to deliver messages using identifiers instead of network addresses. When being instantiated on top of an address-based network architecture, it should allow the reservation of many network addresses from the same device. For instance, both IPv4 and IPv6 supports this feature, but current IPv4 based network infrastructures obstruct and/or forbids this operation, so we can only consider IPv6 as a direct underlying network. Thus, the protocol of the architecture, discussed in Section 4, can be instantiated over many different underlying networks and, occasionally, allow the coexistence of multiple underlying architectures.

Many other network architectures are better suited for our architecture than current IP infrastructures. First, we have the overlay network protocols used in many DHT infrastructures, as used in Chord [Stoica et al., 2001] because its simplicity and its performance improvements, such as the Lookup-Parasitic Random Sampling (LPRS) [Zhang et al., 2003]. Then, we have other interesting protocols coming from content-centric or publisher/subscriber network architectures, such as Content-Centric Networking (CCN) [Jacobson et al., 2009] and Publish-Subscribe Internet Routing Paradigm (PSIRP) [Fotiou et al., 2009]. For now, we are using CCN to implement a proof-of-concept of our architecture for evaluating it, which we discuss in Section 5.1.

3.4 Security

Instead of hiding the identity information of an entity, this architecture offers the possibility to access it in a controlled manner. Thus, the DTE is responsible of managing the identity information, so others may ask it to find out if an entity is exactly "who" it claims to be. Also, we can consider that an entity is *authenticated* just by validating that the identifier (or identifiers) it is using belongs to it and ensuring the integrity of the messages exchanged with it, which is done by a signature (or token) field included in the messages. Therefore, our architecture and protocol provides integrated authentication of all communication ends as well as message integrity.

At the networking layer, the privacy of an entity can be violated by guessing the identity that is behind a device/host just by linking/tracing its network operations. In order to prevent this problem, the architecture described here permits arbitrary changes of session identifiers. This capability does not affect communications because they are bound to *identities* instead of *identifiers* or *addresses*. New session identifiers are negotiated through the DTE infrastructure, which is a totally secure and trusted channel, so attackers can not follow the data flow to guess what identity is behind a concrete identifier. An open issue for general security could be that with this model an attacker can not be identified, but it is not a new issue as soon as any communication must be negotiated before taking place and the DTE infrastructure has (and protects) the binding between identities and identifiers. It is able to log them for subsequent cybercrime investigations, just as the same way it is done today.

Finally, our architecture proposes and recommends to use an asymmetric encryption mechanism to give confidentiality when needed. It may be inefficient and processor hungry but with obvious 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 more feasible. This does not prevent our architecture to adopt symmetric mechanisms and key exchange protocols such as IKEv2 (Internet Key Exchange Version 2) but they are out of the scope of the current work.

3.5 Mobility

The architecture has inherent mobility support because communications are bound to identities instead of identifiers or locators (addresses). This means that applications are totally independent of the network attachment point of the device/host and the mobility effort falls on the network infrastructure.

About actual mobility management, when the architecture is instantiated on top of a locator-independent underlying infrastructure, such as Chord or CCN, the DTE associated to a device just reports to the underlying infrastructure the new attachment point of the device when it moves from one edge network to another. In contrast, when the architecture is instantiated on top of a locator-dependent underlying infrastructure, the end-nodes collaborate with the DTEs to establish a new identifiers and new locator (address) to maintain communications but the applications or existing sessions are still unaffected.

3.6 Application Message Exchanges

Since our architecture provides endpoint semantics and permits services to have their own identifiers, it may be directly used by applications and services to exchange their messages, reducing the final layers used in communication. For instance, in a SOAP (Simple Object Access Protocol) based application, each layer introduces its own headers and message format so it is difficult to take full communication control from the application layer and also makes it difficult to apply traffic engineering because communication semantics are hidden in upper protocol layers. On the contrary, using our protocol the messages are directly delivered through the network, so it is simpler and traffic engineering may easily consider the application level.

3.7 Message Format

Applications in Future Internet may need extensive use of metadata in their communications, so the message format must be extensible to support an arbitrary number of fields but keeping mandatory the minimum necessary fields, such as the source and destination identifiers, and the content.

Our architecture has defined a flexible message format, as commented above. Thus, applications may include specific headers into network messages so identity infrastructure is able to correctly, securely, and efficiently deliver them. Also, other information may be introduced to be used by endpoints, so applications get a fine control over their messages. Finally, actual messages can be instantiated in many low-level message representations that may need specific headers, as name/value/field-separator, JSON (JavaScript Object Notation), XML (Extensible Markup Language), and binary.

4 Protocol

In previous section we defined the integrated architecture we designed to fill the gaps introduced in the beginning of the paper, as well as its components (mainly the DTE infrastructure) and its relevant aspects. In this section we describe the protocol followed by entities to communicate with each other. As discussed in previous sections, the protocol is not bound to any underlying network architecture. By other means, both the architecture and protocol are so generic that they can be instantiated on top of many network architectures.

The protocol is composed of four main operations: 1) The authentication of entities into their corresponding DTEs; 2) The search and discovery of the target/destination entities (communication counterparts) through the DTE infrastructure; 3) The establishment of a communication session between communication participants, also through the DTE infrastructure; and 4) The direct exchange of data messages without using the DTE infrastructure. All messages (entity-to-DTE, and entity-to-entity) are exchanged through the overlay network, so, as discussed above, they are independent of the addressing and delivery mechanisms of the underlying network architectures.

To describe the protocol operation we use a simple scenario in which two entities start a *conversation*. As shown by Figure 2, the scenario is composed of two entities (peers), Alice and Bob, which are from two different domains, domain 1 and 2, with its corresponding DTE instances. Alice belongs to domain 1 while Bob belongs to Domain 2. The four main operations of the protocol are detailed below.

First of all, each entity (peer) authenticates into the DTE of its domain and registers the XRDS document that describe its exposed facets, each one with its own identifier based on XRI. Those facets, also called virtual identities, represent the entities during communication acts to protect their actual identities.

After the authentication process, and being registered the services offered by each entity, Alice sends a request to its DTE with a query to get an XRDS document that describes some facets (specified by the query) of Bob, providing the faced Alice wants to use in the associated subsequent sessions. Then, the

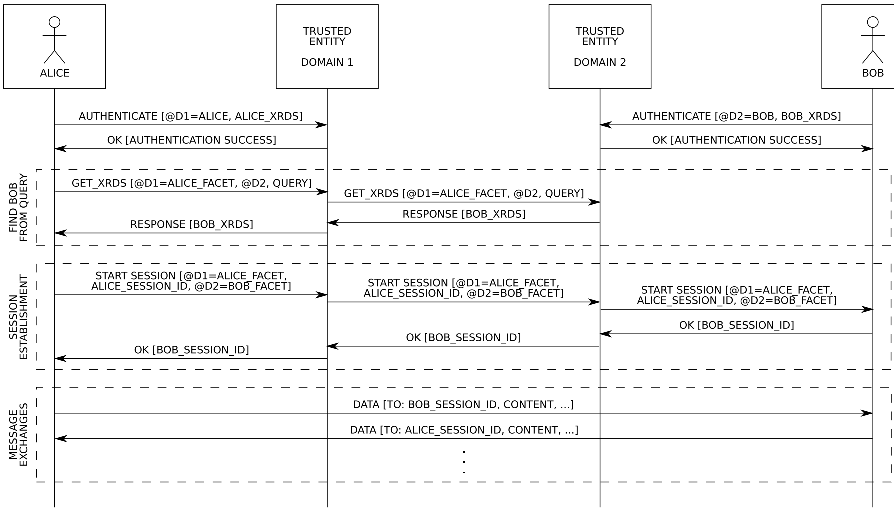

Figure 2: Identity-based Network Protocol.

DTE of domain 1 will contact with the DTE of domain 2 to forward the query and thus get the response with the XRDS document of Bob which is then sent by the DTE of domain 1 to Alice. The response of the DTE of domain 2 is conditioned to the policies set by Bob to regulate the access to its information.

Once Alice gets the XRDS document from Bob, she knows the identifier of the facet (virtual identity) she wants to communicate with ("@D2=BOB_FACET"), which will be usually associated with some service she wants to consume. Now, to start a session, Alice allocates a new session identifier and sends a *start session* request to its DTE, indicating its *source* facet, its session identifier, and the selected facet from Bob. The DTE of domain 1 will forward this request to the DTE of domain 2, which will check the policies to know if the communication is allowed and, if so, will forward the request to Bob. Then, Bob allocates a new session identifier and sends its response to Alice through the DTE infrastructure by sending the response message to its DTE.

Finally, Alice receives the *OK* from its *start session* request and knows the session identifier used by Bob, so she will proceed to send data messages to Bob through its session identifier (BOB_SESSION_IDENTIFIER) and will expect to receive data responses from Bob through its session identifier (ALICE_SESSION_IDENTIFIER). This is done in this way because a message just includes the *destination* identifier, which represents a source and a destination (session direction), so it improves the protocol efficiency.

4.1 Security Analysis

As the main purpose of our architecture is to introduce an identity-based network layer on top of other underlying network to provide enhanced security with integrated privacy and traceability prevention, we want to be sure that our protocol is secure, so here we discuss a security analysis of the protocol presented above. To prevent human mistakes, we perform the security analysis with an automated validation tool. In this case, we opt to use the Automated Validation of Internet Security Protocols and Applications (AVISPA) tool [Viganò, 2006] because its simplicity and strength when analyzing network protocols.

First, we formalize the protocol model shown in Figure 2 and discussed in Section 4 using the protocol security standard notation (Alice-Bob) that can be later used to perform its analysis and validation. We are interested to analyze the portion of the protocol started by Alice to communicate with Bob. The resulting notation is as follows:

```
A -> DTE1 : {{AfID.AsID.BfID}_inv(KA)}_KDTE1
DTE1 -> DTE2 : {{AfID.AsID.BfID}_inv(KDTE1)}_KDTE2
DTE2 -> B : {{AfID.AsID.BfID}_inv(KDTE2)}_KB
B -> DTE2 : {{BsID}_inv(KB)}_KDTE2
DTE2 -> DTE1 : {{BsID}_inv(KDTE2)}_KDTE1
DTE1 -> A : {{BsID}_inv(KDTE1)}_KA
A -> B : BsID
B -> A : AsID
```

In this extract, *A*, *B*, *DTE1*, and *DTE2* are used to represent Alice, Bob, DTE1, and DTE2, which are the entities taking part in the protocol. The remaining tokens are as follows: *AfID* and *BfID* represent facet identifiers of Alice and Bob; *KDTE1*, *KDTE2*, *KA*, and *KB* represent the cryptographic public keys of the entities; *AsID* and *BsID* are the session identifiers of Alice and Bob. Finally, the *inv* function gets the cryptographic private key of a public key.

From the notation described above we create a full description in High-Level Protocol Specification Language (HLPSL) that is used by the AVISPA tool. On it we define a different role for each entity taking part of the communication and fulfill each role with its specific responsibilities defined above in the notation. Then, we indicate that the analyzer tool should check that AsID and BsID can be used to authenticate Alice and Bob respectively, and that it should check the secrecy of AfID and BfID to be sure that there is a secure channel between Alice and Bob through the DTE infrastructure, so the session identifiers can not be publicly associated to their owners. Finally, the HLPSL file is then used as input for the AVISPA tool using the On-the-Fly Model Checker (OFMC) as back-end of the analysis. The output of the AVISPA tool, which indicates that all security tests are correctly passed, confirms and demonstrates the security of the protocol.

5 Evaluation

To demonstrate its feasibility we discuss how to implement the architecture we propose in this paper on top of other network architectures. Below we comment two different implementations, one made with CCN and the other made with the Extensible Messaging and Presence Protocol (XMPP), both working as lower layer network. First we use CCN because it is a novel clean-slate architecture that defines both protocol and routing infrastructure, placing the content in the middle of communications. On the contrary, we also use XMPP to show how we can instantiate our architecture on top of an existing network protocol.

5.1 Instantiation Over CCN

The Content-Centric Networking (CCN) is a network architecture that defines a content-centric protocol, similar to a publisher/subscriber architecture, where identifiers are used to identify the content that is delivered through the network instead of the communication endpoints. Thus, it does not directly provide the possibility to perform end-to-end message exchanges, therefore, we need to build an adaptation layer to allow direct communications among network nodes.

In CCN, a subscriber declares its interest on some content, which is identified by a URI-like identifier, and waits until that content is available. Then, a publisher *updates* the content with that identifier, sending that content to the intermediate elements that deliver it to all interested subscribers. Therefore, when instantiated over CCN, the way an entity of our architecture communicates its identifier to the lower layer network is declaring its interest of some content, the content identified by its own identifier. Our adaptation layer exploits this behavior, so each communication party declares its interest on a content identified by its own endpoint identifier. Then, when other party wants to send a message, it just updates the content identified by the destination identifier. This update makes the message to be received by the destination party.

After building the adaptation layer we define the message format that will be inserted in CCN content elements. As JSON is supported by CCN, we can directly use the message format described in the previous section without any change. Also, because of CCN identifiers are URI-like, we can use directly the XRI identifiers as proposed in our architecture, so it fits perfectly on top of CCN.

Once defined message and content formats we implement the DTE logic responsible of receiving requests and sending back responses for authentication, XRDS, and validation. Then, we instantiate a different DTE for each domain with its own configuration. Finally, we build the clients, Alice and Bob, that send and receive those messages defined in the scenario described above.

5.2 Instantiation Over XMPP

The Extensible Messaging and Presence Protocol (XMPP) is an application layer protocol widely used nowadays by many messaging infrastructures and we think it could be interesting to see how our architecture can be integrated with it. For instance, we can instantiate our architecture on top of XMPP, as we discuss below, but also we can modify XMPP to run on top of our architecture or, finally, get both architectures working side-by-side to achieve the identity-based authentication and privacy protection mechanisms.

Here we describe how to instantiate a solution that implements our architecture over XMPP. The key functionality offered by XMPP to cover our requirements is the dynamic user-name registration and de-registration, which is used by our architecture to reserve the dynamic session identifiers (user names in XMPP) because here XMPP is acting as lower layer network, and bind them to the identities.

After knowing that our architecture can be easily instantiated over XMPP we define the entities that implement the functionality to run the scenario shown in Figure 2. In this case, the role of DTE is played by XMPP server, so we define an XMPP server for each domain. Then, we build XMPP clients to play the role of Alice and Bob. Finally, we noticed that this layer, used to adapt our architecture to XMPP, is very thin because it fits perfectly with our architecture.

6 Results

In this section we show the results of the execution of the tests we performed with each instantiation to exercise the architecture and protocol working over CCN and XMPP. We compared these results with the results obtained from the execution of raw protocols, without our architecture, so we can get a notion of the performance penalties that can be introduced by our architecture. In the tests we measure both the time spent in each message exchange and the total time spent in the whole test. Then, with the former measures we calculate the average time spent for each message exchange and with the latter measures we calculate the time spent by each message in terms of the whole application.

First, Figure 3 and Figure 4 show the results of the tests performed with CCN and XMPP respectively, as well as with our architecture instantiated on top of them. On the plots we can watch the average time spent on each message exchange displayed as “One Way Avg” and “Two Ways Avg”. It also shows the total time taken by the whole execution divided by the number of exchanges, including the extra processing, displayed as “One Way Total” and “Two Ways

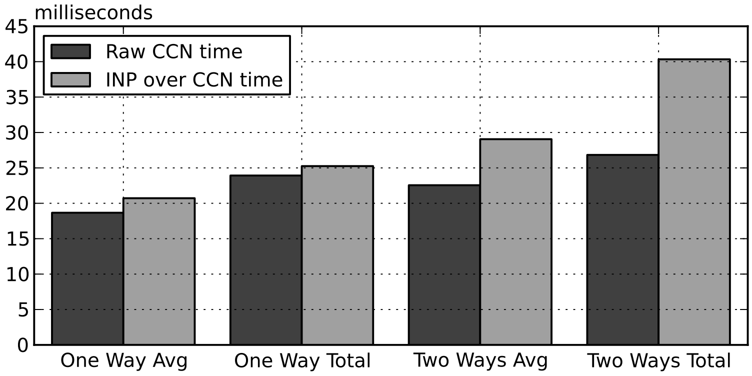

Figure 3: Performance comparison of our protocol over CCN with raw CCN.

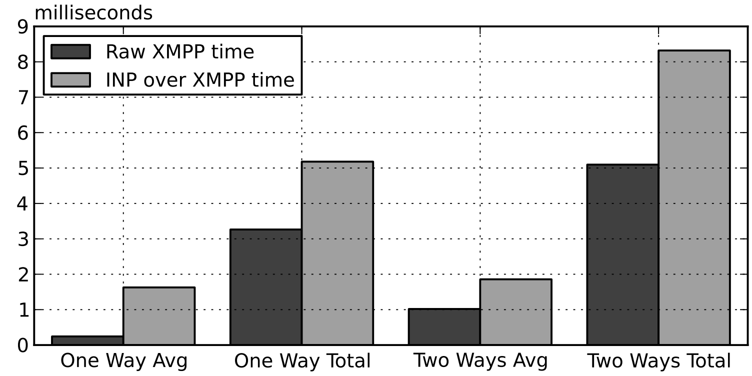

Figure 4: Performance comparison of our protocol over XMPP with raw XMPP.

Total". One-way results are obtained measuring the time spent in sending messages only from an emitter to a receiver, while two-way results are obtained measuring the time spent in sending requests and receiving responses. The two-way test includes the messages exchanged with the other elements of our architecture.

Then, Figure 5 shows the comparison of the overhead of our protocol (aka INP) when instantiated over CCN and XMPP for the two-way tests. The overhead is the principal remark of all tests with respect to our architecture. It lets us see the time increased by using our identity based architecture and protocol on top of the other lower layer network architectures. Furthermore, Figure 6 shows

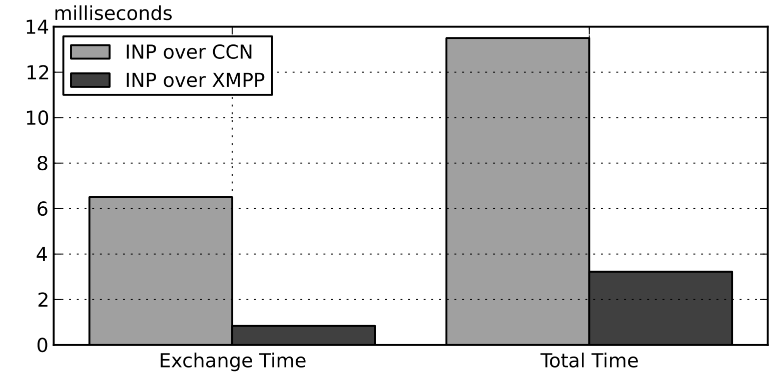

Figure 5: Comparison of the overhead of our protocol on CCN and XMPP.

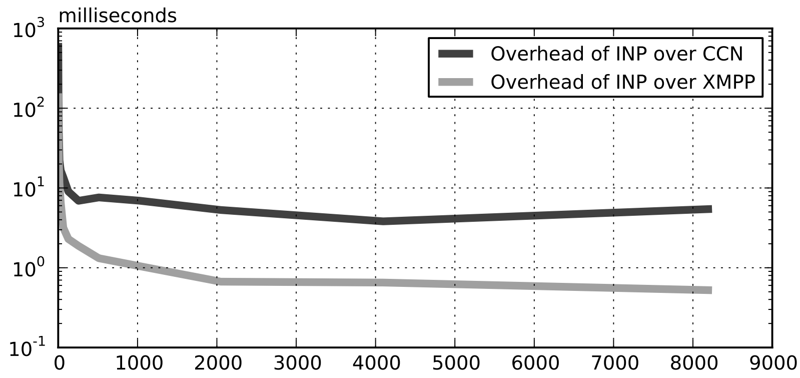

Figure 6: Overhead evolution when increasing the message exchanges.

the evolution of overhead when increasing the number of messages. We can see that the overhead is bigger when there is a small number of exchanges but decreases quickly as the number of exchanges increases, being stabilized below 10 ms in the case of INP over CCN and below 1 ms in the case of INP over XMPP.

Observing the results described above and especially the overhead comparison, we extract that our architecture takes only a few milliseconds (ms) more than raw CCN or XMPP, which is due to the necessary extra time to process JSON formatted messages and certain specific operations of the adaptation layer over each lower layer protocol. The worst case is in the exchange time of the request/response messages because of two extra steps (JSON parsing and encoding) but, as shown in the overhead comparison and reinforced by the overhead evolution, it only takes around 6.5 ms more than raw CCN and less than 1 ms more than raw XMPP. Finally, the extra *total time* observed in the tests is due to the authentication, XRDS exchanges, and identifier validation. Although this extra time is negligible for communications with several exchanges, it must be minimized for those with few exchanges.

7 Conclusions and Future Work

In this paper we presented a novel architecture and protocol for the Next Generation Internet that places digital identities in the middle of communications. In this manner, it allows identity-to-identity networking, while keeping the overall security, with special interest to privacy and preventing traceability of the entities taking part of communications. We then verified the security of the protocol using an automated security verification tool. Also, we demonstrated the feasibility of this identity-based network architecture by building a proof-of-concept implementation on top of CCN, a content-based network architecture, and on top of XMPP, a widely used messaging protocol. The results obtained from the test were promising, demonstrating that our architecture scales very well, adding less than 10 ms to each message exchange on top of CCN and less than 1 ms to each message exchange on top of XMPP.

For the future work we plan to investigate the decentralization of identity validation to gain certain level of independence from the DTE. This may accelerate the transactions involving only a few messages. Also, we plan to investigate about a straight adaptation of XRI to XMPP. Furthermore, while this paper shows implementations of our architecture over CCN and XMPP as underlying networks, we plan to study the behaviour of the architecture over other infrastructures, mainly overlay networks, as well as how to interact directly with current IP architectures, trying to provide the features of our architecture to it. Finally, as shown in [Martinez-Julia et al., 2012], we plan to integrate our architecture with other network architectures for the Future Internet.

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

[Fotiou et al., 2009] Fotiou, N., Polyzos, G. and Trossen, D. (2009). Illustrating a Publish-Subscribe Internet Architecture. In 2nd Euro-NF Workshop on Future Internet Architectures and New Trends in Network and Services Architectures.

[Gomez-Skarmeta et al., 2010] Gomez-Skarmeta, A. F., Martinez-Julia, P., Girao, J. and Sarma, A. (2010). Identity Based Architecture for Secure Communication in Future Internet. In Proceedings of the 6th ACM Workshop on Digital Identity Management pp. 45–48, ACM, New York, NY, USA.

[ITU-T, 2009] ITU-T (2009). 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.

[Jacobson et al., 2009] Jacobson, V., Smetters, D. K., Thornton, J. D., Plass, M. F., Briggs, N. H. and Braynard, R. L. (2009). Networking Named Content. In Proceedings of the 5th International Conference on Emerging Networking Experiments and Technologies (CoNEXT '09) pp. 1–12, ACM, New York, NY, USA.

[Jain, 2006] Jain, R. (2006). Internet 3.0: Ten Problems with Current Internet Architecture and Solutions for the Next Generation. In Proceedings of Military Communications Conference pp. 1–9, IEEE Computer Society, Los Alamitos, CA, USA.

[Kafle and Inoue, 2010] Kafle, V. P. and Inoue, M. (2010). HIMALIS: Heterogeneity Inclusion and Mobility Adaptation through Locator ID Separation in New Generation Network. IEICE Transactions on Communications *E93-B*, 478–489.

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

[López et al., 2009] López, G., Cánovas, O., Gómez-Skarmeta, A. F. and Girao, J. (2009). A SWIFT Take on Identity Management. IEEE Computer *42*, 58–65.

[Martinez-Julia et al., 2011] Martinez-Julia, P., Gomez-Skarmeta, A. F., Girao, J. and Sarma, A. (2011). Protecting Digital Identities in Future Networks. In Proceedings of the Future Network and Mobile Summit 2011 pp. 1–8, International Information Management Corporation.

[Martinez-Julia et al., 2012] Martinez-Julia, P., Gomez-Skarmeta, A. F., Kafle, V. P. and Inoue, M. (2012). Secure and Robust Framework for ID/Locator Mapping System. IEICE Transactions on Information and Systems *E95-D*, 108–116.

[Meyer, 2008] Meyer, D. (2008). The Locator Identifier Separation Protocol (LISP). The Internet Protocol Journal *11*, 23–36.

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

[National Institute of Information and Communications Technology, 2010] National Institute of Information and Communications Technology (2010). "AKARI" Architecture Design Project for New Generation Network. `http://akari-project.nict.go.jp`.

[Pan et al., 2009] Pan, J., Jain, R., Paul, S., Bowman, M., Xu, X. and Chen, S. (2009). Enhanced MILSA Architecture for Naming, Addressing, Routing and Security Issues in the Next Generation Internet. In Proceedings of the International Conference on Communications pp. 14–18, IEEE, Washington, DC, USA.

[Reed et al., 2008] Reed, D., Chasen, L. and Tan, W. (2008). OpenID Identity Discovery With XRI and XRDS. In Proceedings of the 7th Symposium on Identity and Trust on the Internet (IDtrust '08) pp. 19–25, ACM, New York, NY, USA.

[Stoica et al., 2001] Stoica, I., Morris, R., Karger, D., Kaashoek, M. F. and Balakrishnan, H. (2001). 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 pp. 149–160, ACM, New York, NY, USA.

[Viganò, 2006] Viganò, L. (2006). Automated Security Protocol Analysis With the AVISPA Tool. Electronic Notes in Theoretical Computer Science *155*, 61–86.

[Ylitalo and Nikander, 2006] Ylitalo, J. and Nikander, P. (2006). BLIND: A Complete Identity Protection Framework for End-Points. Lecture Notes in Computer Science *3957*, 163–176.

[Zhang et al., 2003] Zhang, H., Goel, A. and Govindan, R. (2003). Incrementally Improving Lookup Latency in Distributed Hash Table Systems. In Proceedings of the 2003 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems pp. 114–125, ACM, New York, NY, USA.