IEICE TRANS. INF. & SYST., VOL.E95–D, NO.1 JANUARY 2012

PAPER Special Section on Trust, Security and Privacy in Computing and Communication Systems

Secure and Robust Framework for ID/Locator Mapping System

Pedro MARTINEZ-JULIA[†a)], **Antonio F. GOMEZ-SKARMETA**[†], ***Nonmembers***, **Ved P. KAFLE**[††], ***Member***, ***and*** **Masugi INOUE**[††], ***Senior Member***

SUMMARY The use of IP addresses as host IDs and locators in the present day Internet protocols imposes constraints on designing efficient solutions for mobility, multihoming, renumbering, and security. To eliminate the constraints, different approaches of introducing ID/locator split into future network architectures have been discussed recently. HIMALIS is such an architecture, which uses distinct sets of values for identifiers and locators and allows the network layer to change locators without requiring the upper layers to change identifiers. One of the major challenges of HIMALIS is the design and implementation of a distributed ID-to-locator mapping database system to efficiently store, update and provide the up-to-date mapping data to the network elements. For this purpose, this paper discusses the application of the Domain Trusted Entity (DTE) infrastructure to the HIMALIS architecture. It provides a unified manner to get locators from high level identifiers (names) with enhanced security, privacy, and trust, while maintaining all capabilities and full compatibility with the previous DNR, HNR, and IDR infrastructures found in HIMALIS.

key words: *new generation network, identity, security, privacy*

1. Introduction

The current Internet is based on two kinds of namespaces: domain names and IP addresses. Internet applications resolve the domain name into an IP address during a communication initialization phase via a Domain Name System (DNS) lookup. The IP address is then used by the networking protocols to identify communication sessions and locate the destination host during data communications.

That said, it is widely known that there are some significant problems in the use of IP addresses in the whole host protocol stack. Namely, an IP address is used in the network layer protocols as a locator to forward packets and locate the destination host. The same IP address is also used in the transport and upper layer protocols as the host identifier (IDs). This dual role of IP addresses as host IDs and locators makes difficult to design efficient solutions for mobility, multihoming, renumbering, and security because such solutions require the provision to change locators used at the network layer without changing the host IDs used at the transport, session, and application layers. So, in the search towards the next generation Internet or new generation network, the separation of ID and locator has become one of the most important challenges [1], [2].

Different approaches of introducing ID/locator split into network architectures have been discussed recently [3]–[8]. Among these proposals, only HIMALIS (Heterogeneity Inclusion and Mobility Adaptation through Locator ID Separation) [3] proposes a comprehensive architecture to address the problems of mobitily, multihoming, routing and security as well as supporting heterogeneous protocols in the network layer. The HIMALIS architecture uses distinct sets of values for identifiers and locators and allows the network layer to change locators without requiring the upper layers to change identifiers.

Therefore, the HIMALIS architecture implements ID/locator mapping functions in the end hosts as well as in the edge routers or gateways. A new layer, called identity layer, is introduced between the transport and network layers to execute these functions. The new layer separates the scopes of IDs and locators, i.e. IDs are used in the application and transport layers to identify sockets while locators are used in the network layer to locate destination hosts and forward packets towards them. ID/locator mapping records are required to perform the ID/locator mapping functions. In order to efficiently maintain and retrieve ID/locator mapping records, it introduces three new elements that cooperate with hosts and gateways. The new elements are the Domain Name Registry (DNR), the Host Name Registry (HNR), and the ID Registry (IDR). In summary, the DNR is a name resolution infrastructure used to resolve domain names to the address of the HNR elements. Then, the HNR elements, which in turn are instantiated on edge networks, are responsible of maintaining hostname to ID and locator mapping records. Finally, the IDR is a distributed registry used to maintain the ID/locator mappings and communicate their updates to the routers that connect those edge networks to the global transit network.

In previous work [3], the DNR, HNR and IDR are organized into a hierarchical logical network, which are queried by hosts and gateways in a sequential order. Although the hierarchical structure is good for scalability, it may not optimal for faster updates of records. Therefore, in this paper, we explore an alternative architecture by joining the functionality of the HNR and IDR in one infrastructure and connecting them to form a flat overlay network, such as the network found in Chord [9]. For this purpose, we adapt the

Manuscript received March 16, 2011.
Manuscript revised July 23, 2011.

[†]The authors are with the Department of Communication and Information Engineering, University of Murcia, 30100, Murcia, Spain.
[††]The authors are with National Institute of Information and Communications Technology (NICT), Koganei-shi, 184–8795 Japan.
a) E-mail: pedromj@um.es

DOI: 10.1587/transinf.E95.D.108

Domain Trusted Entity (DTE) infrastructure, an architecture that incorporates a distributed trusted framework to achieve secure identity-based communications [10], [11]. The application of DTE to the joint IDR/HNR infrastructure provides many new features, like enhanced security and robustness, without performance penalty. Moreover, the DNR functionality is unnecessary because the elements of the IDR/HNR can be reached through the overlay network.

Apart from the capabilities commented above, the DTE infrastructure provides additional mechanisms of identity management to the network, like checking policies to control the communications or dynamically change of Host IDs to enhance privacy. Also, it provides to network elements the ability to know that an entity is *who* it is pretending to be and, thus, getting inherent and integrated authentication and, when applying policies, the authorization of network operations. In summary, the DTE infrastructure is able to raise the role of identities in the network, placing them in the middle of future communications.

The rest of this paper is organized as follows. In Sect. 2 we discuss our integration proposal. Then, in Sect. 3 we show the necessary message exchanges to start a session, as well as to support the mobility. In Sect. 4 we discuss the experiments we run to demonstrate the feasibility and performance of the approach and in Sect. 5 discuss the results. Finally, in Sect. 6 we discuss our conclusions and the future work.

2. Integration Proposal

In this section we detail our view of the profile that uses the functionality provided by the joined IDR/HNR infrastructure mentioned above.

First, Fig. 1 shows the architecture of the profile we propose for HIMALIS. On it we depict the general components defined by HIMALIS, changing the IDR by our IDR/HNR infrastructure and, on purpose, not showing the DNR because we assume that IDR/HNR elements can be reached without needing a previous name resolution. Also we depict with arrows the main interactions between the gateways and the IDR/HNR, as well as the end devices and the IDR/HNR infrastructure.

Each element of the IDR/HNR is responsible of a network domain and therefore it is instantiated (usually) in the primary or main network of the same domain. It is also responsible of the identification domain of the entities belonging to its domain. This identification domain may not match with its physical/logical network domain because of mobility. As introduced above, all IDR/HNR elements of each domain are connected to form an overlay network that is independent of the physical/logical topology. Although overlay networks like Chord may seem inefficient, their simplicity have led to the emergence of many improvements, such as LPRS [12], a derivation of Chord that makes it perform very close to its underlaying network. Furthermore, each element of the IDR/HNR infrastructure has certain level of trust with the other elements and has the necessary mechanisms to securely communicate.

Like the HIMALIS architecture, the integrated profile we propose starts with a Host ID that has to be mapped to a locator (address), but now the gateways asks to the joined IDR/HNR infrastructure for those mappings. Also, when necessary, the gateways will update the id/loc mapping information but final devices can also manage that information, because they are the primary authority of their id/loc mapping. Although the mapping is supported by the HostID of the entities taking part of the communication, they are not fixed and can be dynamically generated. In this case, the IDR/HNR infrastructure can be used to resolve the HostID of a communication party from other attributes of its digital identity. To keep security, this process is controlled by policies that determine if the operation can or cannot be done.

Once gateways have received the necessary mappings to get addresses from HostIDs, they keep them on their mapping cache until they expire or new information is pushed by the IDR/HNR. Thus, the IDR/HNR has a close and mutual collaboration with gateways, both in an active manner. The final device can also take part of this collaborative process but it is not strictly necessary, thus devices with reduced

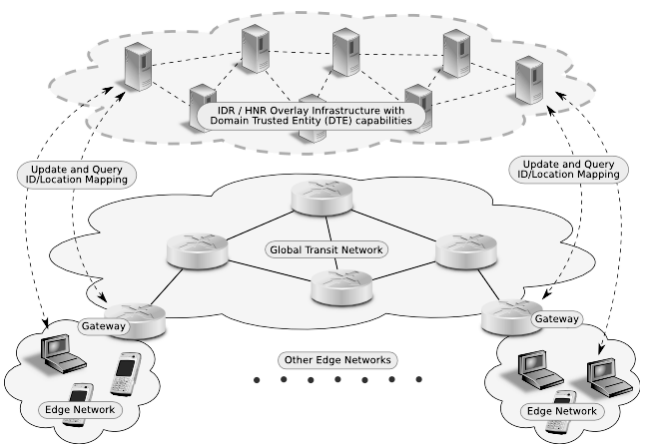


Fig. 1 Integrated architecture.

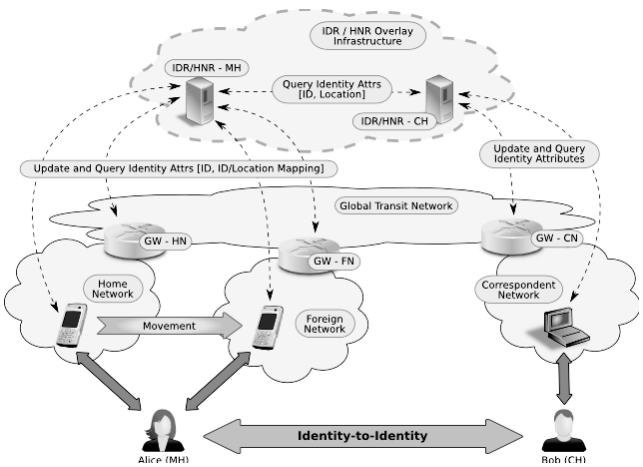


Fig. 2 Integrated architecture showing Mobility and Identities.

capabilities (like those in sensor networks) are totally supported by the architecture.

After introducing the whole architecture, Fig. 2 shows the interactions and elements involved in an expanded scenario in which a device moves from an edge network to another. Here we show how IDR/HNR elements interact to bring id/loc mapping information to the necessary gateways (those involved in the communication). Also, it depicts how the address or location of a device is treated as an attribute of the identity that is behind it, so its access can be controlled by policies, like other identity attributes. The result is that an entity represented by a digital identity can *talk* to other entity represented by another digital identity.

3. Message Exchanges

In this section we discuss the necessary message exchanges to implement the integrated architecture discussed in the previous section. Thus we detail the specific messages needed to start a session between two entities and the specific messages needed to update the id/loc mapppings stored in different network elements when when an entity changes its location (handover).

Before two (or more) entities can start a communication they need to establish a session between (among) them. First of all, the entities (or their devices) must be registered in the IDR/HNR elements of their corresponding domains. Here we call this process "authentication" because, on it, the entities behind the devices confirm their identities and thus the IDR/HNR can authenticate them whenever needed.

After the authentication, the entity (or its device) can communicate with other entities but, before, it must find them. To do it, the requester entity contacts with the IDR/HNR element of its domain and asks it to find the desired destination identity from certain provided information. Here, in the HIMALIS integrated profile, the information may be the HostID but, as commented above, it is not limited to it. Thus, the process can operate with a general query to the attributes of the digital identity, always subject to the policies set by its owner and also taking into account the identity of the requester.

When the search process ends, the requester entity and the IDR/HNR element of its domain receive the necessary information to communicate with the other identity (representing single or multiple entities), such as the *facets* it exposes, which here is like a virtual identity. Although this process is needed to search the descriptors of any identity, the results can be cached to be used in subsequent communications. Finally, the requester entity selects a *facet* of the destination identity and the session can start.

From this point, the HIMALIS integrated profile proposed in this paper differs from the process followed by the current HIMALIS architecture. To start a session, generally the first session, instead of let entities communicate by their own, we propose to place the IDR/HNR infrastructure in the middle to negotiate the session attributes, such as the session identifiers used by each endpoint. As the IDR/HNR has a highly secure and reliable way to make this communication, using it to make the initial negotiation strengthens security. Figure 3 shows the messages exchanged to start a session between Alice and Bob, which actually represent the devices used by the actual entities. It works as follows:

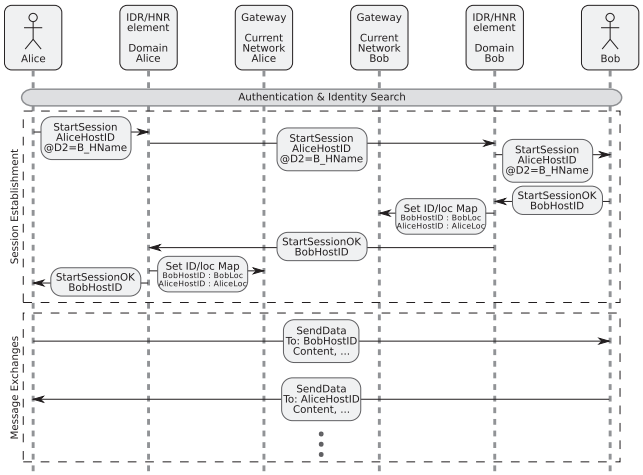


Fig. 3 Messages exchanged to start a session.

1. Alice sends to the IDR/HNR element of its domain a message called *StartSession* which contains its HostID and a query to find Bob by means of its hostname, that is seen as an attribute of Bob's identity.

2. The IDR/HNR element of Alice's domain sends a request to the IDR/HNR element of the other domain (Bob's domain, "D2") with the information provided by Alice. Each IDR/HNR element can reach each other through the overlay network without needing to resolve any name, just using their domain identifier.

3. Now, the IDR/HNR element of "D2" checks the corresponding policies set for Bob's hostname and searches the entity that responds to it. Then it sends the *StartSession* request to Bob because it responds to *Bob*.

4. Bob accepts the session, keeps the HostID of Alice, and sends a *StartSessionOK* message with its HostID to its IDR/HNR element.

5. Once the IDR/HNR element of Bob's domain has the HostIDs and locations of Alice and Bob, it reports to the current gateway to which Bob is connected to set new mappings for the HostIDs with the locations (BobHostID, AliceHostID).

6. After setting the mapping to the gateway, The IDR/HNR element of Bob's domain sends a *StartSessionOK* message to the corresponding IDR/HNR element of Alice's domain.

7. As did the IDR/HNR element of Bob's domain, the IDR/HNR element of Alice's domain sends a message to set the id/loc mappings to the current gateway to which Alice is connected.

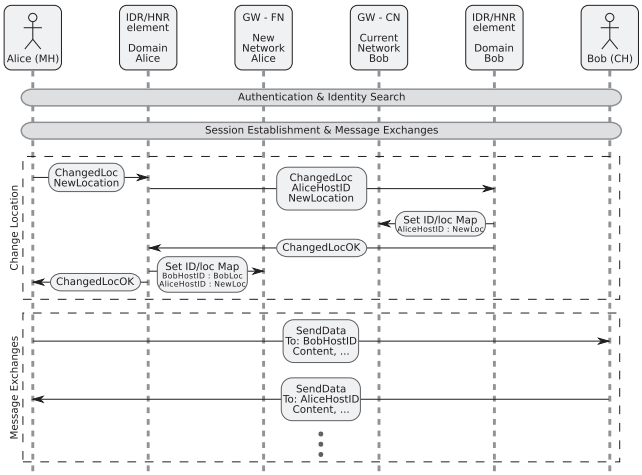


Fig. 4 Messages exchanged after a movement (handover).

8. Finally, the IDR/HNR 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.

Once the session is started we show what happens when one of the entities move to other network. After this, the gateways involved in the communication (old and new) must be updated with the new mappings. Below we comment how this is achieved by our proposal and Fig. 4 illustrates it. This process operates as follows:

1. After changing its location, Alice sends a *ChangedLoc* message to the IDR/HNR element of its domain indicating its new location.

2. Since Alice's IDR/HNR element knows the opened sessions, it sends a *ChangedLoc* message for each session to their corresponding IDR/HNR element, so here it sends a *ChangedLoc* message to the IDR/HNR element of Bob's domain with *AliceHostID* and Alice's new locator.

3. As the IDR/HNR element of Bob's domain knows that Bob's HostID has not changed, it only sends the new mapping of Alice to the gateway to which Bob is connected using the appropriate message.

4. After knowing that the change has taken place, the IDR/HNR element of Bob's domain confirms it sending a *ChangedLocOK* message to the IDR/HNR element of Alice's domain.

5. Now the IDR/HNR element of Alice's domain sends the id/loc mapping of Alice and Bob to the new gateway to which Alice is connected.

6. Finally, the IDR/HNR of Alice's domain sends the *ChangedLocOK* message to Alice, that now can continue its communication with Bob.

In parallel to this process, after setting the new mappings in the new gateway, the IDR/HNR of Alice's domain reports back the new location of Alice to the old gateway so it can send its missing messages to Alice's new location.

4. Experimentation

Once we have described the architecture design and defined the behavior of the integrated architecture we wanted to get an approach of its performance so we implemented a proof-of-concept solution and used it to perform an experiment to see the behavior of the architecture. In the following subsections we describe the implementation, the testbed, and the experiment performed with them.

4.1 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 a library that implements the protocol used by the Domain Trusted Entities (DTEs) and the clients. We also built a library with an implementation of the Chord [9] overlay routing algorithm to be used by the DTEs to communicate each other. With these libraries, plus a UDP-based low-level transport layer, we built a module for the instantiation of the three DTEs (home, foreign, correspondent) and other module for the two clients (the Mobile Node and the Correspondent Node). We need to note that using UDP implies that, when performing the tests, some (or many) messages are going to be lost for high data-rates. Finally, using also the UDP transport, we have implemented a simple gateway module with id/loc mapping support and which is able 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 with the Python language because of its simplicity and flexibility, but the critical modules, like the Gateway, are then built to binary using Cython.

We decided to implement the overlay network following the Chord approach 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 DTEs, as we show in [11], in this paper we have not focused on this aspect but on the secure identity-to-identity communication and the mobility support of our architecture applied as an id/loc mapping system.

Finally, the client nodes are defined to make a simple communication. The Mobile Node asks for a number of messages, specifying the rate at which she wants them, and the Correspondent Node sends the messages to her. In a mobility scenario it performs the same operations but at certain moment the Mobile Node changes its location (changes its

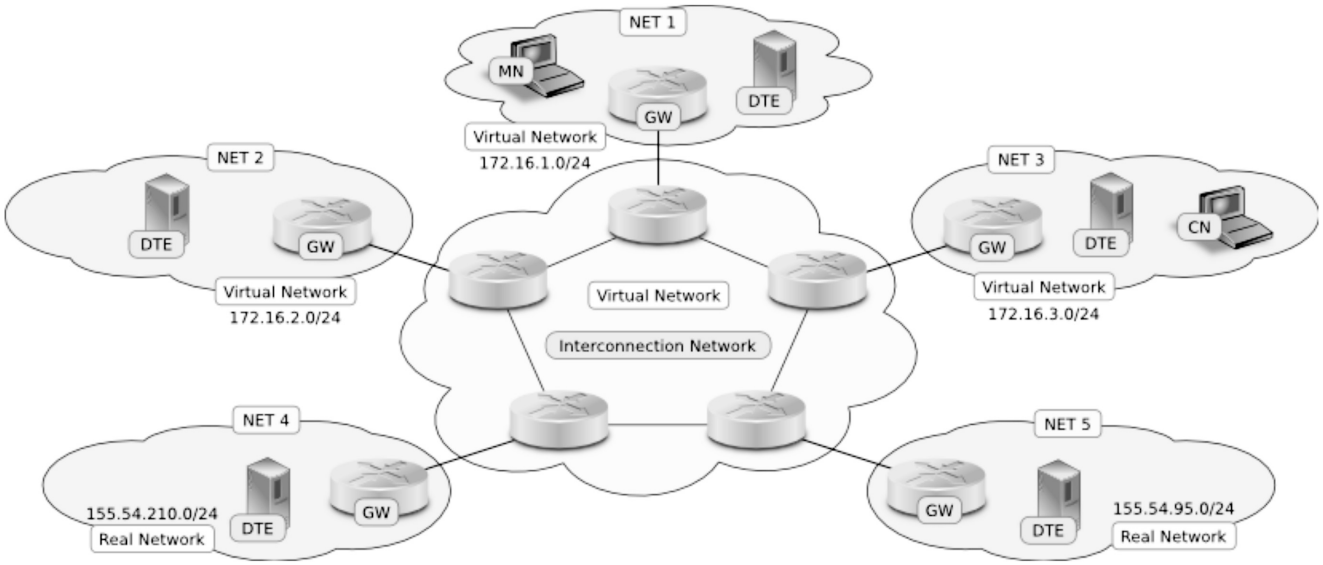


Fig. 5 Experimentation environment. The elements of the architecture are labeled GW and DTE, corresponding to the Gateway and Domain Trusted Entity. The MN and CN are the Mobile Node and Correspondent Node respectively. The elements of the real networks are configured to find the virtual networks through their respective GWs.

current network). Finally, we built other clients using direct UDP exchanges over IP instead of the Gateways to communicate each other.

4.2 Experimentation Environment

To test the solution described above we built an experimentation environment that consists 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 [13] to build ethernet bridges through TCP/IP connections between separated host machines.

Figure 5 shows the topology of the experimentation environment with all the elements defined in our architecture (DTEs and GWs), as well as the client nodes (MN and CN). The edge networks correspond to different network and identity domains. Thus, the Mobile Node (MN) and Correspondent Node (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 Gateway which is configured to route them through the interconnection network.

As defined in our approach, the DTEs build an overlay network. The names used by the DTEs are: “domain1”, “domain2”, “domain3”, “domain4”, and “domain5”. Thus, using the mapping system we implemented in the Chord layer, that uses 16-bit identifiers, they correspond to: “0xEFA2”, “0x804F”, ”0x23CC”, “0xBD92”, and “0x26BB”. These identifiers are disperse enough to make some DTE-to-DTE message exchanges to cross other DTEs. For instance, the DTE of the domain 3 needs to cross the DTE of the domain 2 to reach the DTEs of the domains 1 and 4, but can reach the DTEs of the domains 2 and 5 directly, in just one hop through the overlay network.

4.3 Experiments

With the environment and software components described above we performed different experiments to see the behavior and performance of the proposed architecture. We also compare it with IP and CCN [14], which is a well-known Information-Centric Networking proposal for the Future Internet that we approached in a previous work [10].

In order to see the scalability and overall performance of the solution we first run a test to get the average time spent in transmit and receive a single message but at different message rates. We set the previously described Mobile Node and Correspondent Node to exchange messages at rates from 1 message per second to 16384 messages per second. The test is run 30 times for each rate to get the average, because measuring a single exchange is very imprecise, and the whole solution is run 10 times to get the standard deviation and standard error of the measures. As described above, the infrastructure is built in top of UDP, so we expect to lose messages from a (high) rate onwards. Thus, we also extract the loss percentage for each message rate. To get a valid impression of the results we also run the tests

with the aforementioned IP-based client nodes. Therefore, we can compare the performance of the proposed architecture with the IP approach, as well as with the raw-CCN and the ID-based CCN approaches commented above.

Once we have the results of the average exchange time and loss percentage for several message exchange rates we want to see how behaves our mobility approach so we run again the test described above but now we set the Mobile Node to change from its home network (domain 1) to other network (domain 2) after certain received messages. This experiment will give us a notion of the average message exchange rate and message loss while the node is moving. Since our approach first makes the home (or current) Gateway to route the messages to the new Gateway (as soon as possible) and then contacts with the DTE of the Corresponding Node to update the id/loc mapping of its Gateway, we do not expect to have a big performance penalty or a large increment of the message loss.

Finally, using the results of the experiments commented above we select a specific message rate to run a global experiment that makes the Mobile Node to sequentially change to all the domains. From this experiment we want to extract the actual behavior on mobility events, represented with the message rate during the handover and the whole handover time, as well as an impression of the overall behavior of our approach running in different network domains. To get this we configured the Mobile Node to be virtually connected to all networks but with all network interfaces down, except the interface with its home domain that is always up to reach the node from the outside. Then we added the capability to bring up and down the network interfaces to the software running the Mobile Node and make it to use that capability each 5 seconds to change from one domain to another. Every action is logged together with a timestamp so we can measure the times and message counts and represent the results.

5. Results

In this section we discuss the results obtained from the experiments commented in the previous section. First we show the scalability results and performance comparison between our approach and IP, as well as CCN and ID-based CCN. Then we show the comparison of the loss percentage for IP, our approach, and our approach with the mobility event.

Figure 6 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. About the CCN results, we show that our approach adds a few milliseconds to the raw CCN but it is constant

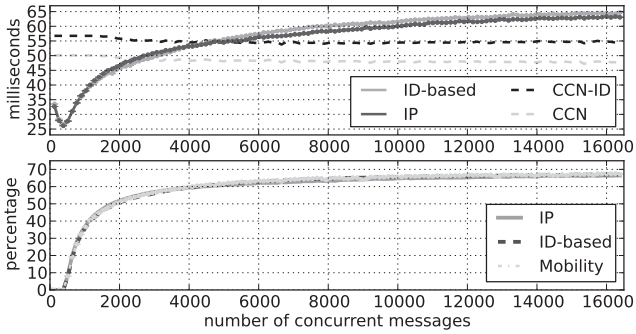


Fig. 6 Average time evolution in milliseconds (top) and message loss percentage (bottom). The former compares the behavior of our architecture over IP in contrast with our architecture over CCN and the latter compares the message loss percentage of IP, our architecture, and our architecture while a node is moving. As the plots are overlapped, Fig. 8 and Fig. 9 show the differences of our architecture with the base case (IP).

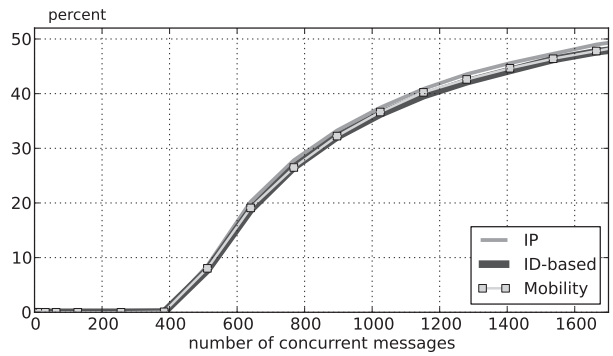


Fig. 7 Excerpt of the evolution of message loss for the different tests that show the turning point where the message loss leaves the 0%, that is around 400 concurrent messages.

for increasing data rates. This also demonstrates that our architecture scales well. The fact that CCN results are under IP and our approach is due to the CCN nature. CCN does not use any routing machinery or gateway, it broadcasts messages to all interested nodes, so it can deliver more messages per second after it has established the path. For low data rates, CCN is worse because it is implemented in Java and the path establishment from the sender and receiver takes more time than the other solutions.

For the loss percentage, in which the three results are together, we see that they do not cross the 70% of message loss even for huge data rates. For rates under 2000 msgs/s, the loss stays under 50% and so on. Figure 7 shows the results for the lower rates amplified to see at which rate the solutions start to loose messages. We can see that no solution looses messages under 400 msgs/s but around 500 msgs/s the loss percentage is still around 10%. We will select this rate to perform our final experiment, as commented in the previous section.

As the results for our approach and IP are very near in the figure commented above and since we want now see the real separation between them, we now show the differences in different figures and comment them.

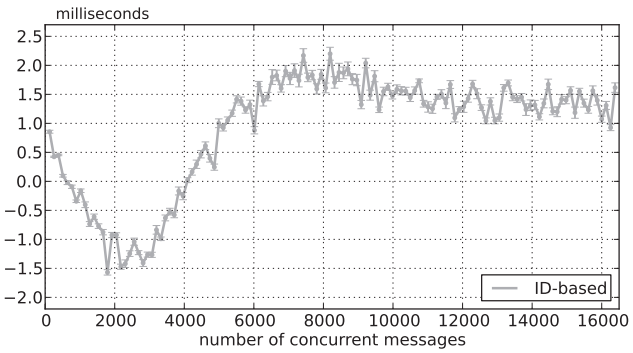


Fig. 8 Evolution of the difference of the average exchange time of our architecture and IP. The error-bars represent the standard error of the original measures of the tests with our architecture.

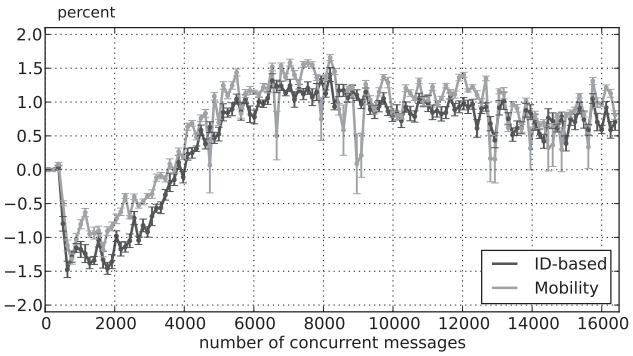


Fig. 9 Evolution of the difference between the message loss of our architecture and IP, and between our architecture during a mobility event and IP. The error-bars represent the standard error of the original measures of the tests with our architecture.

Figure 8 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. We must notice that the plot is saw-shaped because of the enlargement, which is demonstrated because the standard error and hence the standard deviation is very low, what gives us high level of confidence in the results. In the plot we can see that our approach does not differ from IP in more than 2.5 milliseconds and that for some data rates it spends up to 1.5 milliseconds less per each message exchange. In huge message rates, the difference is stabilized between 1 and 2 milliseconds over IP, which is a very good result taking into account that our approach is implemented in top of UDP. The reason that our approach is better than IP for some rates is that for lower rates it is quicker to resolve an id/loc mapping than a routing table entry.

Parallel to the above figure we have Fig. 9 that 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. As we can see, the results are with high confidence since the standard error is very low, but the non-mobility test has slightly more confidence. Also, the two plots are very similar in their global movings, but the mobility plot is more unstable and

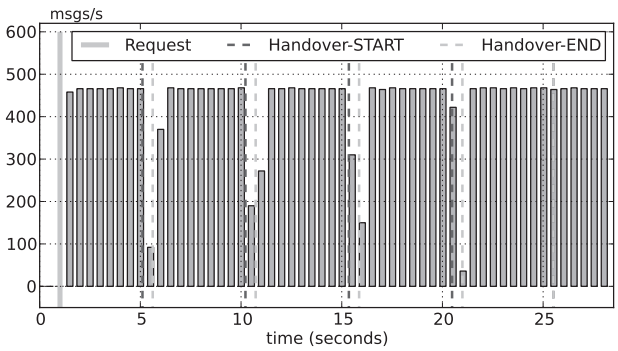


Fig. 10 Evolution of the solution behavior as the mobile node moves through all domains while receiving messages at a rate of 500 msgs/s. The vertical solid line marks the time when the request is sent and the vertical dashed lines mark start and end of each handover process, whose respective timespan is 508 ms, 507 ms, 508 ms, 512 ms, and 4 ms.

has higher standard error. Seeing the results as a whole, our approach is separated around 1.5% of the raw-IP solution, having around 0% and 1.5% more message loss than the raw-IP approach. As the previous results, these results also validates our identity-based approach against the behavior of the raw-IP solution.

Finally, we have run our global experiment with the 500 msgs/s parameter defined from the firstly commented results. Figure 10 shows for each moment of time, starting from 0 seconds and ending at 28 seconds, the net rate of the message exchanges between the Correspondent Node and the Mobile Node. Each 5 seconds the Mobile Node changes its location to the next domain in counterclockwise order. We can see the moment in which the Mobile Node 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% under 500, but during the handover processes it looses more messages. These looses are sometimes compensated, as seen in the next bars close to the handover end events. The handover events are launched each 5 seconds, so they are represented by the vertical dashed lines near the following x labels: 5, 10, 15, 20, and 25. The handover timespan is about 500 ms in all events but the last, demonstrating that the extra time is spent in bringing up the *new* interfaces because, as we commented in the previous section, it is the only configuration difference between this interface and the others, and we purposely left it as is to see what happens. Thus, the last handover has almost negligible timespan (4 ms) because the network interface is already prepared.

6. Conclusions and Future Work

In the present paper we have described how to add the security, privacy, and trust capabilities provided by our previously defined Domain Trusted Entity (DTE) infrastructure to the HIMALIS architecture. In the integrated approach, the DTE infrastructure is placed as the IDR infrastructure of the original HIMALIS and generates a different profile that gets rid of the individual HNR elements and DNR elements

of the original architecture, but offering the same functionality through the new DTE infrastructure that plays the role of the IDR and HNR elements while making the DNR elements totally unnecessary.

As discussed in Sect. 2 and Sect. 3 the overall security in communications is managed by the resulting IDR/HNR infrastructure, which is implemented by our DTE infrastructure. It is achieved by making the *home* DTE of each network domain to be the main authority in the negotiation of network operations involving any of the entities of its domain, such as the resolution from Host ID to locator. Moreover, enhanced privacy is achieved by applying policies defined by the entities to those operations. The DTEs will reveal information pertaining to an entity (identity) only to the allowed entities. Furthermore, the privacy is also enhanced by supporting the dynamic change of host identifiers, which can be used to prevent network operation linkage to concrete entities.

We have demonstrated the feasibility, scalability, and good performance of our proposal through the results of some experiments with a proof-of-concept implementation on a realistic testbed and with many network and administrative domains. We also demonstrated that the proposed mobility approach is totally transparent to the communication parties and almost transparent in terms of performance and message loss. Finally, we discussed a global test that shows the general behavior of the architecture with many handover events through many different network domains. From it we can extract that the main obstacle in the handover is the establishment (set-up) of the new network interface when moving to a new network.

As a future work we plan to investigate the performance differences of the current HIMALIS architecture and the new profile we propose in terms of mapping resolutions and attribute-based session negotiations, and improve if necessary. We also pretend to investigate in a network interface that can be kept up while moving to reduce the necessary handover timespan. Finally, we plan to investigate in possible evolutions of the overlay network algorithm to improve its performance with more knowledge of the underlying network while keeping its flexibility.

Acknowledgments

This work is partially supported by the European Commission's Seventh Framework Programme (FP7/2007-2013) project GN3 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," internet-draft, Internet Research Task Force, 2007.

[2] R. Jain, "Internet 3.0: Ten problems with current internet architecture and solutions for the next generation," Proc. Military Communications Conference, Los Alamitos, CA, USA, pp.1–9, IEEE Computer Society, 2006.

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

[4] V.P. Kafle, H. Otsuki, and M. Inoue, "An id/locator split architecture for future networks," IEEE Commun. Mag., vol.48, no.2, pp.138–144, 2010.

[5] E. Nordmark and M. Bugnulo, "Shim6: Level 3 multihoming shim protocol for IPV6," RFC 5533, 2009.

[6] R. Moskowitz and P. Nikander, "Host identity protocol (HIP) architecture," RFC 4423, 2006.

[7] D. Farinacci, V. Fuller, D. Meyer, and D. Lewis, "Locator/id separation protocol (LISP)," internet-draft, 2011.

[8] C. Vogt, "Six/one routers: a scalable and backward compatible solution for provider-independent addressing," Proc. 2003 ACM MobiArch, Seattle, Washington, USA, 2008.

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

[10] P. Martinez-Julia and A.F. Gomez-Skarmeta, "Secure identity-to-identity communications over content-centric networking," Proc. 4th IEEE International Conference on Internet Multimedia Systems Architecture and Applications, pp.1–6, Washington, DC, USA, 2010.

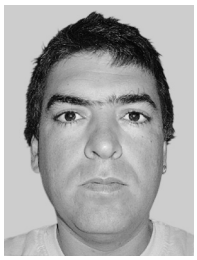
[11] A.F. Gomez-Skarmeta, P. Martinez-Julia, J. Girao, and A. Sarma, "Identity based architecture for secure communication in future internet," Proc. 6th ACM Workshop on Digital Identity Management, pp.45–48, New York, NY, USA 2010.

[12] H. Zhang, A. Goel, and R. Govindan, "Incrementally improving lookup latency in distributed hash table systems," Proc. 2003 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems, pp.114–125, New York, NY, USA, 2003.

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

[14] V. Jacobson, D.K. Smetters, J.D. Thornton, M.F. Plass, N.H. Briggs, and R.L. Braynard, "Networking named content," Proc. 5th International Conference on Emerging Networking Experiments and Technologies (CoNEXT '09), pp.1–12, New York, NY, USA, 2009.

Pedro Martinez-Julia received the B.S. degree in Computer Science from the Open University of Catalonia in 2009 and the M.S. degree in Advanced Information Technology and Telematics from the University of Murcia in 2010. He is currently a Ph.D. candidate at the University of Murcia. Since 2009 he is a research fellow in the Department of Communication and Information Engineering at the University of Murcia, Spain. He has been part of the research staff working in JRA3-T3 of the GN3 project under the European Commission's Seventh Framework Programme (FP7-2007-2013). His main interests are the overlay networks, the security protocols, and the distributed systems and services. He is an associate member of ACM and IEEE.



Antonio F. Gomez-Skarmeta received the M.S. degree in Computer Science from the University of Granada and B.S. (Hons.) and the Ph.D. degrees in Computer Science from the University of Murcia Spain. Since 2009 he is Full Professor at the same department and University. Antonio F. Gómez-Skarmeta has work on different research projects in the national and international area, like Euro6IX, 6Power, Positif, Seinit, Deserec, Enable and Daidalos. His main interested is in the integration of security services at different layers like networking, management and web services. He is associate editor of the IEEE SMC-Part B and reviewer of several international journals. He has published over 90 international papers and being member of several program committees.

Ved P. Kafle received the B.E. degree in electronics and electrical communications from Punjab Engineering College (now PEC University of Technology), Chandigarh, India. He received the M.S. degree in computer science and engineering from Seoul National University, South Korea, and the Ph.D. degree in informatics from the Graduate University for Advanced Studies, Japan. Since 2006, he has been working at the National Institute of Information and Communications Technology (NICT), Tokyo, where he is now a senior researcher and involved in R&D of new generation network architectures and protocols. In particular, his current research interests lie in node identification and location architectures, scalable routing, integration of heterogeneous network layer protocols, and new paradigm of mobility and security in the future Internet. He has been a member of the AKARI Architecture Design Project for the new generation network in Japan. He received the ITU Association of Japan Award in 2009 and the best paper award (second prize) at the ITU-T Kaleidoscope event on Innovations for Digital Inclusion, 2009.

Masugi Inoue received the B.E. degree from Kyoto University in 1992 and the M.E. and D.E. degrees from the University of Tokyo in 1994 and 1997, respectively, all in electrical engineering. He has been engaged in R&D of ultrahigh-speed WLAN systems using 60 GHz bands, wireless and mobile network systems for 4th generation cellular networks, and Future Networks since he joined the Communications Research Laboratory (CRL), which was reorganized as NICT in 2004. He was a visiting researcher at Polytechnic University (now, the Polytechnic Institute of NYU), Brooklyn, NY, in 2000. He is currently a Planning Manager in the Strategic Planning Department of NICT. He is a Vice Chair of the Technical Committee on Mobile Multimedia Communications of IEICE, a member of the Technical Committee on Ubiquitous Sensor Networks of IEICE, and a member of the New Generation Network Promotion Forum. He received the IEICE Young Researcher's Award in 2003, the Best Paper Awards from the Information Processing Society of Japan (IPSJ) in 2006 and 2007, and the Young Scientists' Prize of the Commendation for Science and Technology by the Ministry of Education, Culture, Sports, Science and Technology (MEXT), Japan in 2007.