Integrating User Identity Management Systems with the Host Identity Protocol

Marc Barisch
Institute of Communication Networks
and Computer Engineering
Universität Stuttgart
marc.barisch@ikr.uni-stuttgart.de

Alfredo Matos
Institute of Telecommunications
University of Aveiro
alfredo.matos@av.it.pt

***Abstract*—Identity Management (IdM) on the application layer improves the usability and security for end users by offering features like Single Sign-On and attribute provisioning. Unrelated approaches on the network layer introduce identity concepts to solve mobility problems and support multihoming. This paper describes a novel approach to the integration of IdM on the application layer with identity concepts introduced by the Host Identity Protocol (HIP). We propose an integrated architecture combining the advantages of both domains. In this scope, we tackle the mapping between the HIP namespace and user IdM namespace as well as we the management and assignment of user and host identities. The new architecture provides a unified view over user and host identities, enabling the exchange of user and host attributes, while it also provides enhanced security and network features.**

I. Introduction

As the notion of Identity finds its way into more and more areas of information and communication technology, digital Identity and Identity Management (IdM) are becoming key pillars of the future Internet.

Identity concepts at the application layer are linked to new opportunities for users, like video-sharing, social-networking and context-aware services. From a technical perspective, concepts like attribute sharing and Single Sign-On (SSO), which improve the security and the convenience for users, are coupled to identities and IdM. There are several initiatives, like Microsoft CardSpace [?], Liberty Alliance [?] and OpenID [?] that compete for the best IdM solution.

Additionally, identity related concepts are also emerging on the network layer. Proposals that target the identifier locator split problem are one example of introducing identities on the network layer, trying to solve complex problems like mobility and multihoming, with either an implicit [?] or explicit notion [?] of identities. In the remainder of this paper the term *host identity* will be used to reflect these concepts.

Even if the purpose of identities at the application and network layer is different, the general idea of identities and IdM is shared. In both cases, an identity describes an entity represented by a set of attributes within a specific context [?]. Fig. **??** contrasts the two different perspectives. The left side illustrates a user identity on the application layer, which is made up of attributes like name or postal address. Moreover, legal contracts and credentials to be used with Service Providers (SP) or Identity Providers (IdP) are part of an identity. In contrast, host identities shown on the right side of Fig. **??** are focused on the characteristics of hosts and devices. Thus, locators like the IP address play a major role.

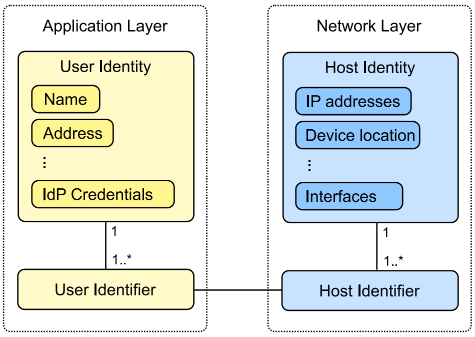

Fig. 1. User and Host Identities

At first glance, user and host identities, each labelled by a identifier, are unrelated. However, we believe that user identities and host identities can not be considered independent of each other. An integrated view on identities across the user and host level is required, due to several reasons, presented below.

First, with the introduction of personal computers and the high distribution of mobile phones, user identities and host identities get more and more coupled. It is not always sensible to differentiate between attributes of the user and attributes of the host, e.g. location.

Second, it is required to consider user and host identities together to evaluate privacy risks. For example, IP addresses which are assigned to hosts can be used to reveal characteristics of the user identity [?].

Third, an integrated view allows to benefit from the advantages, which are provided by both identity concepts. A detailed discussion of the mutual advantages of both identity concepts is provided in Section **??**.

We conclude that host identities are coupled to user identities, which makes an integrated consideration necessary. In this paper we present the integration of user and host identity concepts through an architecture that integrates the Host Identity Protocol (HIP), as a network level protocol capable of delivering mobility and multihoming heavily based on identity concepts, and a SAML-based IdM system [?].

The remainder of this paper is structured as follows. Section ?? gives an overview on user IdM, introduces HIP and discusses the advantages and challenges of such an integration. Next, Section ?? focuses on the integration of two different namespaces. Afterwards Section ?? describes an integrated architecture and exemplifies novel use cases. Related work on the integration of identity concepts is elaborated in Section ?? and put in perspective to our approach in Section ??. Finally, Section ?? concludes this paper.

II. Integrating user and host identities

The integration of host and user identities provides advantages, while simultaneously posing new challenges. Therefore, we must first understand the key features of user IdM frameworks and of HIP in order to discuss the advantages and challenges of integrating both.

A. *User Identity Management*

The almost infinite number of web services and service providers accounts for new security threats, like identity theft, due to the reuse of usernames and passwords for different accounts. In addition, there is a need to improve the convenience for users by simplified but secure attribute provisioning for providers. These trends fostered the development of standardized and interoperable IdM solutions.

Existing IdM solutions usually differentiate three roles: User, IdP, and SP. The user authenticates against the IdP with which he has a pre-established contract. Based on pre-established trust relationships, the SPs rely on IdP's statements expressing successful user authentication and do not require an explicit authentication. This principle is called Single Sign-On (SSO). Moreover, the IdP can provide additional services like Single Logout or the provisioning of user attributes to the SP.

Several frameworks like Microsoft CardSpace [?], OpenID [?], Liberty Alliance (LA) [?], and Shibboleth [?], are beginning to compose the IdM landscape.

Microsoft CardSpace introduces the notion of managed information cards, comprised of user attributes and meta information about the IdP, to represent the user's identity. Upon successful authentication against the IdP a security token, e.g. a SAML assertion, is created and transferred from the IdP to the SP via the user's terminal. Moreover, it is possible to include user attributes. For the transfer of security tokens, WS-$*$ specifications are used.

In contrast, the specifications from LA and Shibboleth rely not only on the definition of SAML assertions but also employ the corresponding protocols. Beyond the protocols for SSO, the retrieval of user attributes by the SP is specified. Herein, a user is identified by a hierarchical identifier, which points to the responsible IdP. Shibboleth extends the SAML specification towards the requirements of academia, whereas LA is focused on business environments.

Currently, OpenID is gaining momentum through the increasing support by various global players, like Google or Yahoo. OpenID uses URIs as user identifiers that are resolved into the endpoint of an IdP, used for authentication. With OpenID, it is possible for the user to run his own IdP. Recently, OpenID was extended with a protocol [?] to fetch and store attributes about a given user from an IdP.

The mentioned IdM frameworks provide SSO capabilities and allow the basic exchange of user attributes between IdPs and SPs. Moreover, it is common to use hierarchically organized user identifiers (UI), divided into an IdP identifier and a user specific part. The determining factor to select a SAML-based IdM framework for the integrated consideration is the flexibility with respect to user attribute exchange, and the increasing converge towards the re-usage of SAML protocols or derived versions.

B. *Host Identity Protocol*

HIP employs a cryptographic namespace to solve the dual use of IP addresses as topological locators and host identifiers. By introducing an identity concept at the network layer, where every host is represented by an asymmetric key pair, consisting of a public and private key, it turns IP addresses into pure locators.

A public key is used as the HIP Host Identity (HI), while the private key serves as proof of ownership of the public key. To seamlessly integrate HIP with protocols above the network layer, a 128-bit cryptographic hash of the HI, the Host Identity Tag (HIT), was introduced to fit the IPv6 address space. The HIT is a statistically unique flat identifier. When HIP is used, the transport layer binds to HITs and is unaware of the IP addresses used for routing.

The core of the HIP protocol [?] is the four-way base exchange (BE), shown in Fig. ??. The BE provides means for two hosts to prove their HIs and mutually authenticate each other. It also includes a Diffie-Hellman (DH) key exchange to establish secure IPSec security associations.

Both, mobility and multihoming [?], are supported through locator agility. Because the communication is bound to the HI, locators can change over time without disrupting ongoing connections.

HIP provides a unique composition of identity concepts at the network layer, and uses them to seamlessly provide security and mobility, which are key aspects in next generation networks. However, at the same time the identity concepts are under-explored, since as seen above, there is a plethora of mechanisms that can be

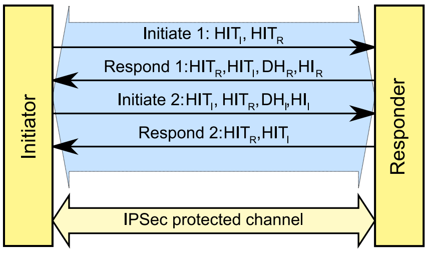
Initiator
Initiate 1: HIT_I, HIT_R
Respond 1:HIT_R,HIT_I,DH_R,HI_R
Initiate 2:HIT_I, HIT_R,DH_I,HI_I
Respond 2:HIT_R,HIT_I
Responder
IPSec protected channel

Fig. 2. HIP Base Exchange

associated with identity, that are currently absent at the network layer.

Consequently, HIP brings for the first time an identity concept to the network layer, where ownership paradigms apply, and information is stored and used around the HI, rather than on specific protocols information. Furthermore, no defined ways of managing Host Identities, which are the conceptual entity behind the HI, or of trustingly verify them, rather than having them signed by a known certificate authority exist. We believe that the distribution of static public key certificates does not fit the dynamics of next generation networks and propose a solution based on the integration with user IdM.

C. Benefits and Challenges

HIP and user IdM are very different when considering the layer on which they operate, the identifiers employed and the problem they solve. This imposes several challenges, but leaves room for new improvements. By providing an integrated solution, we can leverage the best of both worlds, tackling key issues, as described below:

- Security: Most IdM transactions rely on an authenticated and encrypted communication channel, which is in most cases realized by TLS. HIP can provide equivalent security features based on IPSec and should replace TLS in order to avoid duplicate functionality at different layers.
- Trust: Host Identities can be both attributed and verified by the IdM system. This allows to replace global public key infrastructures by reusing existing trust relationships already present in federated IdM systems. This results in cross layer authentication, and systems support.
- Mobility: User IdM systems are focused on the application layer without relationship to network level mechanisms. With the mobility and multihoming support, introduced by HIP, it is possible to make IdM transactions independent of the host location.
- Cross Layer Attribute Exchange: The integration of HIs and user IdM offers new opportunities with respect to cross layer attribute exchange. It is possible to retrieve user attributes based on the HI and host related attributes based on the UI. For example the HI can be used to obtain the current location of the user without using the UI nor relying on the location information contained in the IP address.

However, the integration of HIP and IdM requires to bear several challenges. First, appropriate security mechanisms are required that allow to put trust into presented HI based on an IdM systems. Section **??** proposes a way of solving this. The second challenge is related to different structures of the HIP and IdM namespaces and their identifiers. Section **??** details the different properties of the HIP and IdM namespace and describes resolution possibilities to get from an identifier in the HIP namespace to an identifier in the IdM namespace.

III. Namespaces and Identifiers

Integrating HIP and user IdM requires mapping HI on UI and vice versa, as illustrated in Fig. **??**. However, the namespace in which HIs and HITs are valid is fundamentally different from the UI namespace: A UI is only valid towards an IdP or SP, whereas an HI carries global significance. That means, the HIP namespace is flat and unique in comparison to hierarchical namespaces used in IdM. To allow the coexistence and integration of these two namespaces, we put the IdM system in the center.

To achieve the mapping from UI to HI, the HI can be stored as an attribute at the IdP. Therefore, it is possible to establish a HIP session with a user based on the UI. In addition, when two users are engaged in a service session, it is fairly simple to verify HIP related information as attributes of the user identity.

Traveling the reverse path, from the HI namespace to the user IdM namespace, requires mapping HI to UI. This direction has to work differently, due to the flat namespace that does not allow converting a given HI/HIT to the UI and to the respective IdP. In addition, knowing a HI or HIT should not entitle to resolve the UI out of privacy reasons. It is sufficient to resolve the corresponding IdP, which can be provided in at least two ways.

First, it is possible to use a global distributed hash table (DHT) to resolve the IdP based on the HIT. This is similar to the resolution of the IP address based on the HIT [**?**]. Each IdP registers the HITs under its control in the DHT, and the information requester uses the DHT to resolve the endpoint of the corresponding IdP.

The second, and more obvious solution is to exchange the IdP information during the HIP base exchange that is anyway required. This not only avoids using a global DHT, but also provides advantages concerning additional delay and signaling overhead. Section **??** provides more details on how this piece of information can be incorporated into the base exchange.

The previous mechanisms enable both HIP Initiator and Responder to reach an IdP and request information using a HIT, which is now a reference to the user identity. The IdP should allow using the HIT as reference for the retrieval of user identity attributes. Section **??** elaborates the linking of HIs and UIs.

IV. Architecture

The aforementioned integration poses architectural challenges, addressed in this section. We further detail the operations mentioned in the previous section, and how they are accomplished within the proposed architecture.

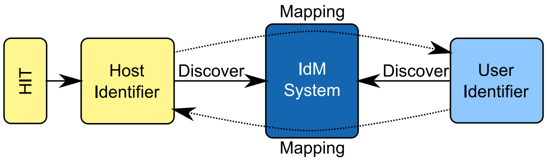

Fig. 3. Integrating identifiers and namespaces.

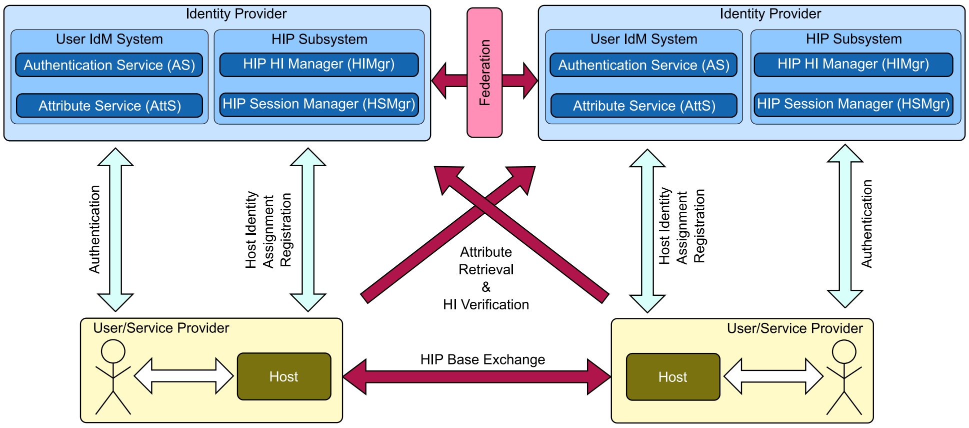

Fig. 4. Architecture and Interaction among Components

Based on the services and functions defined in Section **??**, we describe the key operations of the proposed architecture, which are Host Identity Management (HIM) by the IdP, attribute retrieval using HITs and Host Identity verification through the identity namespace in the subsequent sections.

A. Services and Functions

The IdP is the main entity that allows the interworking of both namespaces. Fig. **??** illustrates the supported IdP services and interactions between users and IdPs.

The user IdM services are geared at an extended SAML architecture, allowing the integration of the host identity namespace. In addition, both IdPs are federated, i.e. a trust relationship exists and attribute exchange between the users of each IdP is possible. The main user IdM services are described below:

- **Authentication Service (AS):** The AS authenticates the user, typically through username and password. Upon successful authentication, it creates a user-specific authentication token (AT) that can be used to consume other services.
- **Attribute Service (AttS):** The AttS manages attributes that belong to a specific user or host. Either the UI or the HI/HIT serves as index for attribute retrieval. An access control function within the AttS restricts access to attributes based on configured policies and information obtained from other services within the architecture.

To support Host Identity integration, we require that the IdP supports HI creation or assignment, along with session management to allow information retrieval about ongoing HIP sessions. These functions, integrated in the HIP subsystem, are described below:

- **HI Manager (HIMgr):** The HIMgr provides the HI assignment and HI registration function. The first one creates and assigns a HI to a host, based on the provided user identifier. In contrast, the HI registration function registers self-assigned host identities and manages this mapping.
- **HIP Session Manager (HSMgr):** An ongoing HIP session can be registered with the HSMgr. This information is valuable for the access control function within the AttS in order to restrict access to some attributes only if a corresponding HIP session exists.

B. Host Identity Management

In the proposed architecture, the IdP is responsible for HIM. HIM implies either the HI generation at the IdP (*Alternative 1*), or user-generated HI that are later registered at the IdP (*Alternative 2*).

In both cases, the user authenticates first against the AS to obtain an AT for further interaction with the HIP subsystem as illustrated in Fig. **??**. In case of an existing HI, the authentication process can use HIP to increase the registration security on top of a secure channel.

1) Alternative 1: IdP assigned Host Identity: This alternative defines that the user requests a HI from the IdP based on the provisioning of the obtained AT. The HIMgr verifies the AT and creates either a new HI or selects an already existing HI based on the AT. If the HIMgr creates a new HI, it has to generate a public/private key pair. Else the HIMgr retrieves the already existing public/private key pair from an internal storage. Afterwards, it activates the mapping between Host and User Identity by registering the HI and HIT as attributes at the AttS. For HI verification at the host, which is described in more detail in Section **??**, the HIMgr creates a X.509 certificate that the host can present in future base exchanges. Eventually, the host receives the HI together with the X.509 certificate via a secure channel and can put the HI in operation for future transactions.

When the HI is assigned by the IdP, the IdP is in the position to always assign the same HI independent of the actually used device. Thus, a long term relationship is established between the HI and the user identity.

Moreover, the IdP has to store the private key and thus acts as a key escrow for HIP. Even tough this presents a serious security forfeit, it provides the basis for lawful interception through a trusted entity.

2) Alternative 2: Self assigned Host Identity: When key escrowing at the IdP is not desired, the user can generate the HI on its own and register the public key at the IdP, resulting in increased network level privacy. The authenticated user contacts the IdP and provides the HI to the HIMgr. The HIMgr calculates the HIT and registers both at the AttS. Finally, an when the user identity is trusted, it creates a X.509 and acknowledges the HI registration.

With the return of an X.509 certificate, we not only turn the IdM system into an ad-hoc PKI that leverages trust relationships with other IdPs, but also enable attribute retrieval based on HIs.

C. Attribute Retrieval

The second use case for the provided architecture is the attribute retrieval based on HIs. In Section **??** we proposed to augment the BE with additional information to identify the responsible IdP for a given HI. We incorporate this information into the X.509 certificates that are returned by the HIM process.

A host provides the certificate to the correspondent host during the base exchange according to [**?**], which defines a type-length-value field “CERT”. This field is specified to transport X.509 certificates [**?**] and allows HI verification based on a PKI.

We propose to exploit the extension section of X.509 certificates to incorporate endpoint references (EPR) [**?**]. An EPR describes the endpoint of a service and incorporates meta data about the service. We use an EPR to describe the AttS with the IdP, that is used for attribute retrieval.

Based on the EPR and the HI/HIT of the correspondent host, it is possible to use the SAML Assertion Query and Request Protocol [**?**] for attribute retrieval.

Fig. **??** exemplifies attribute retrieval based on HIs by means of a user/SP scenario. First, the user and the service provider perform a HIP BE, and exchange HIs and the corresponding certificates. Herein, the user knows the HI of the SP and can update the access control rights that should be granted to the SP. This could allow setting up dynamic access control policies or informing the HSMgr of a new HIP session, exploited by static access control policies. A static access control policy can describe that all correspondent hosts can access the location of the user, for example.

Once the certificate is exchanged, the service provider knows appropriate endpoints at the IdP and is able to query for the desired attribute based on proper authentication against the IdP and depending on granted access control rights.

D. Host Identity Verification

As already introduced in the previous section, each host obtains at the end of the host identity management process an X.509 certificate that is provided to the correspondent node during the BE. That node can use the presented certificate to verify the HI identity in three different ways.

First, the X.509 certificate is signed by the issuing IdP and provides on its own a measure to verify the validity of HI based on the trust relationship to the IdP.

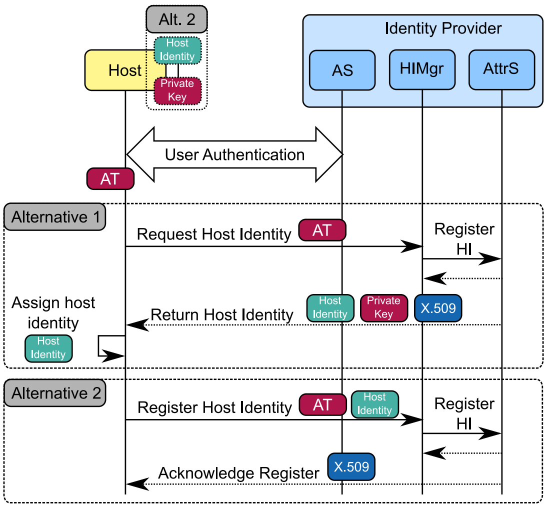

Fig. 5. Host Identity Management Process

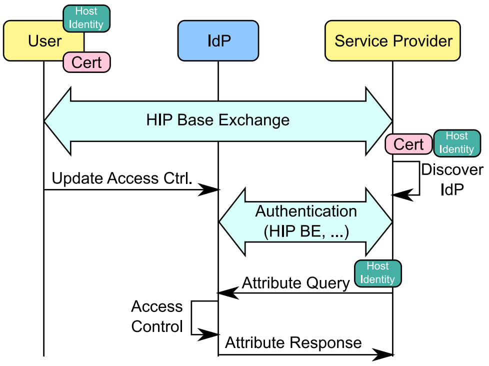

Fig. 6. Extended Base Exchange and attribute retrieval

Second, we can use the AttS to obtain up-to-date information about the used HI and the corresponding certificate, through the attribute retrieval process explained in Section **??**. That means a correspondent node can verify whether the HI has been assigned or registered at that IdP, along with associated trust levels and necessary attributes.

Third, an additional security service is provided by the possibility of verifying the relationship between the HI and the user identity. Given the assumption that a protocol on top of HIP is used, which contains a user identifier, it is possible to use the HIP base exchange as an authentication mechanism by verifying the correlation between HI and UI.

V. Related Work

Coupling distinct IdM application fields is a rising trend, and there have been recent proposals that combine identity management for network and application services.

Sarma et al. [**?**] propose the concept of cross-layer Virtual Identities (VID) to combine network related information, with access control and application layer IdM concepts. This is achieved by using common identifiers across all layers. The VID concept also represents a new paradigm for user privacy protection on application and network layer. Instead of having one identity, the user can have several identities to protect his privacy. Therefore, the use of anonymizing technologies on the network layer to prevent information disclosure is avoided and network identifiers like the IP-address directly reflect the virtual identity. This approach has been realized by the entitled Virtual Network Stacks [?], which are used to endow each VID with a complete communication stack, avoiding correlation. The security impacts of cross-layer identity concepts as well as a methodology to evaluate the privacy impact for the case of Mobile IP are discussed in [?]. Our proposal can be easily combined with the VID concept in order to gain additional privacy.

The concept of mapping identifiers from different layers is not entirely new. A thorough analysis is provided in [?], which is extended in [?] towards an identity driven architecture. It exploits a cross-layer identity approach and provides common addressing functions, based on consistent identifiers to link identities across various layers.

More concrete solutions integrate network authentication functions of identity management with network layer concepts and protocols, which has been neglected so far in our architecture. Lopez et al. [?] propose using network authentication to obtain Single Sign-On (SSO) tokens for the application layer.

The Generic Bootstrapping Architecture (GBA) [?] defines a flexible architecture that allows reusing existing authentication infrastructure within the 3GPP system architecture for additional services. Based on the 3GPP Authentication and Key Agreement mechanism, session keys are generated and distributed between the user and the service. This mechanism forms the foundation for the interaction with Liberty Alliance [?]. The extension allows the usage of GBA for authentication against the Identity Provider and thus the creation of application layer tokens. That means that the secure IdM infrastructure in 3GPP networks can be used to bootstrap more sophisticated applications of IdM with LA.

However, as we distance ourselves from the classic notion of IdM, we begin to encounter other technologies that have identity concepts with different realizations. As already discussed in Section ??, HIP employs a notion of identity based on key pairs. The NodeID [?] concept introduces network layer routing decisions based on HITs, creating the first concepts of identity aware routing protocols. By using DHTs [?], it is possible to distribute both routing and resolution information. Hi3 [?] introduces a routing infrastructure that couples the Internet Indirection Infrastructure [?], with identity information.

By taking the HIP protocol to new grounds based on its identity properties, the identity assignment, also covered in our proposal, gains a new dimension when discussing the network layers. Renewing host identities can be done for privacy reasons, and is discussed by Eggert et al [?].

VI. Discussion

So far, in the presented literature, there have been different protocols and approaches that provide identity at some point in the network stack. They fail by either being to general, or too narrow to the protocol they address.

We provide an approach to integrate identity aware protocols into an IdM control plane, which is suited to handle identity information. By extending the IdM layer with network layer information and identifiers, we open the door to an extremely flexible environment: the IdM layer is no longer limited to service or application specific information, but also includes network information, currently available in scattered points of the network, and addressable through common terms.

By linking layer specific namespaces to broader identity scope, we provide several extensions and improvements to current systems and proposals.

IdM systems are built with security and privacy notions strongly incorporated into their design, along with access control, due to the nature of the handled information. By using such facilities, and leveraging the IdP's trust relationships into a trusted PKI, we are able to couple host and user identity securely and easily. This has a direct impact on the security of the overall system along with the privacy each user obtains. A user is in position of using the same host identity for many transactions regardless of the current device, enabling a new degree of pervasiveness and enriched personalization.

Alternatively, the same device can use several host identities simultaneously, as proposed by [?], and still deliver a trusted environment for both user and service provider, since the dynamic identities can be quickly certified by a well-known and trusted authority, which is the IdP. To the best of our knowledge, no protocol or mechanism exists to assign or register a host identity based on the identity of the user. Existing approaches are based on static assignment or creation of host identities, e.g. based on configuration files. The proposed approach can be easily extended to take additional criteria, like the current device, into account.

It must be also noted that, as mentioned earlier, using IdM assigned private keys, forfeits privacy towards the assigner and should only be used exceptionally. The self-generated identities actually increase the overall privacy of the communication system, as mentioned above. Also, to increase the overall privacy, this approach can be coupled with Virtual Network Stacks for each host identity, decreasing the probability of correlation across different host identities.

Attribute access based on HIs is restricted by access control mechanisms based on policies within the AttS. These policies can be based on ongoing sessions and the HI of the initiator as well as of the responder. Thus, it is possible to have different HIds with different access rights for the correspondent node in a HIP session, opening a granularity that was hard to achieve at lower layers with legacy protocols.

The added value of using HIP, beyond the already existing identity notions and security enhancements, is the fact that both mobility and multihoming are natively supported, which turns our system into a full fledged mobility solution that is in fact integrated with the IdM system.

The proposed approach can be easily integrated into existing IdM systems like LA [?]. The user IdM System in Fig. **??** can be directly mapped to a LA architecture. Only the HIP subsystem has to be built from scratch. Therefore, it provides a simple extension to incorporate host information into IdM systems and allow access to those information without having an explicit service session.

As far as performance is concerned, we consider that the impact is most visible in the pre-session establishment, where the user additionally needs to retrieve and generate host identities, and in the BE. The downsides associated with end-to-end communication are those generated by using HIP in every communication. Even if it is known that the adoption of HIP [?] has an impact on the performance, we do not expect that this is a crucial factor. Most of currently existing IdM systems rely on TLS as the underlying security technology, which has the same performance impact in terms of the number of handshakes to establish a connection as HIP.

Also, due to the added information, it might happen that the packets used for HIP BE exceed the maximum transfer unit (MTU). This might result in packet fragmentation and might cause potential DoS attacks as stated in [?]. Therefore, the consequences of an extended BE have to be further examined.

VII. Conclusion

We have proposed a cross layer approach that harnesses separate identity layers, combining them into an integrated view over user, device and network. This allows us to have a consistent view over the protocol stack, and at the same time providing a consistent approach to multiple identity realizations at different points in the stack. The user identity now contains the host identity as part of the overall information, reachable with several identifiers, which is to the best of our knowledge, not present in literature.

Even though we propose a specific solution for HIP, our solution concept defines a more general process of identity aware network layer protocol integration with user IdM systems. By making lower layer protocols register in the identity cloud, providing addressing structures that identify both endpoints and sessions, we provide a seamless namespace integration that allows architecture design built around attributes.

We are currently working on a proof-of-concept implementation for the proposed approach, in order to properly validate the model and evaluate the key advantages of the combined system.

Acknowledgment

This work was supported in part by the European Union under the FP7 programme (SWIFT project).