Forwarding/Routing with Dual Names: The NovaGenesis Approach

Antonio M. Alberti[1], Jose R. dos Santos[1],
Everton S. de Morais[1], Rodrigo H. Santos[1], Vaner J. Magalhães[1]

[1]ICT Lab - Instituto Nacional de Telecomunicações (Inatel)
CEP 37540-000 - Santa Rita do Sapuca, Minas Gerais, Brasil.

{alberti,vaner}@inatel.br,{evertonmorais,joserodrigo}@gec.inatel.br

rodrigo_hilario@gea.inatel.br

Abstract. *Name-based forwarding/routing (NBFR) is an emerging approach to provide location-independent delivery in Future Internet (FI). It provides identifier-based communication independently of location. Even tough many FI proposals support NBRF for specific ingredients (e.g. contents or hosts), therefore a generic approach is missing. Also, current proposals, like CCN, XIA and NetInf do not employ self-verifying names (SVNes) alternatively as identifiers and/or locators (making dual use of SVNes, saving storage resources). Another novelty is to employ the same BNFR structure not only horizontally (among nodes), but also inside nodes vertically (among operating systems, processes and its components). This paper provides a proof-of-concept of NovaGenesis* Forwarding/Routing with Dual Names *(FRDN). FRDN enables unlimited name-based data plane with SVNes. Future work includes performance evaluation in a large experimental setup, for instance employing Brazilian FIBRE research network in the context of future Internet interconnection point (FIXP) project.*

1. Introduction

Many Future Internet proposals have explored the so called Identifier/Locator (ID/Loc) splitting approach for supporting location-independent information delivery and node mobility [Ramirez et al. 2014]. ID/Loc splitting means to communicate using ID, independently of locators. Information-Centric Networking (ICN) contends for content ID/Loc splitting, enabling persistent identification of information objects (IOs) independently of the locations where the information is stored. IOs are cached in the network and accessed by their unique IDs. Routing can be performed using content names instead of host names, employing a technique called name based routing (NBR).

Named-Data Network (NDN) is an example of NBR [Jacobson et al. 2009] for content-distribution, in which communication is driven by receivers that forward/route (F/R) interest packets containing the name of the desired IOs. Network caching is applied to store frequent accessed contents. Routers maintain a Pending Interest Table (PIT) for contents that have been requested. When some interest packet founds a proper source, data packets are delivered in the reverse path fulfilling content caches. Another example is the so called Service-Centric Networking (SCN), in which services communicate one another using service names [Nordström et al. 2012], rather than combining host addresses and transport layer port numbers. SCN should support persistent service naming, replication, mobility, and chaining. Many other proposals explore ID/Loc splitting for routing according to domain, host, service and content naming. However, all proposals have a limited number of naming spaces from which the identifiers for communication are defined. NovaGenesis (NG) [Alberti et al. 2017a, Alberti et al. 2017b] is a "clean slate" architecture that has been considering unlimited namespaces for NBR as a key design ingredient since its beginning in 2008. This means that NG NBR is not limited to content, services or host IDs. NG contributes to the state-the-art with a generic approach for NBR, avoiding the use of several isolated solutions — therefore, reducing complexity. In this context, the remainder of this manuscript will first present the fundamentals of NovaGenesis in Section 2. Section 3 will present NovaGenesis current implementation, whereas Section 4 details NG NBR approach. Section 5 provides a proof-of-the-concept of the proposed solution. Finally, Section 6 concludes the paper.

2. NovaGenesis: The Basics

NovaGenesis (NG) is founded in three pillars: naming, entities life-cycling and smart objects. Names are a way of denoting entities or existences. They can or not carry meaning. A Natural Language Name (NLN) is a name that has meaning for people, such as the name "Cat". Names that are meaningless to humans are called Flat Names. Flat names can be generated from passing a NLN through a hash function. This procedure generates a Self-Verifying Name (SVN), since the NLN is shuffled by the hash function and can be verified every time it is required. NovaGenesis (NG) employs a Hash algorithm called MurMurHash 3. Names are related one another via Name Bindings (NBs). A NB is a key/value(s) pair in the format $<$ key, value(s) $>$. As an example, a device called "host1" can have a SVN generated from its processor serial number such as "0xff20122006". In this case, a NB can relate the NLN to this SVN, i.e. $<$ host1, 0xff20122006 $>$. NovaGenesis supports a huge web of name bindings (of any sort), which are stored in distributed hash tables. Information objects (IOs) can be associated to NBs, allowing name-based network caching of contents. For instance, a digital photo called "photo001.jpg" can be stored together with its SVN "oxfe19041078", i.e. $<$ oxfe19041078, photo001.jpg $>$. NBs can also support any sort of address, including MAC, network layer and upper layers. In NovaGenesis, all entities (physical or virtual) take advantage of this distributed Name Resolution and Network Cache Service (NRNCS) to store and exchange their own name graphs. Address resolution among different technologies is provided by the network itself. For instance, it is possible to resolve a content name to a service name, a service name to an operating system (OS), and an OS to a host name.

In NG, all entities are seen as services. Even network protocol implementations. Services represent physical devices and ordinary things towards their life-cycles, which include name exposition, discovery, service offering, negotiation, and contracting. A physical devices representative (implemented as a service) is called Smart Object (SO). SOs can represent ordinary things (e.g. a lamppost) or network equipment (e.g. a switch or a router). Quality of service is granted by negotiating contracts among SOs and physical world client services. Network control and operation is performed by SOs that reflect contract requirements in physical devices configurations. The protocol stack is build dynamically by composing Protocols-Implemented-as-a-Service (PIaaS). Novel stacks can be developed and integrated to the existing ones. Evolution is granted by this contract-based service composition, which relays on unlimited namespaces, hierarchical name resolution and network caching (of NBs and IOs) to implement entities life-cycling.

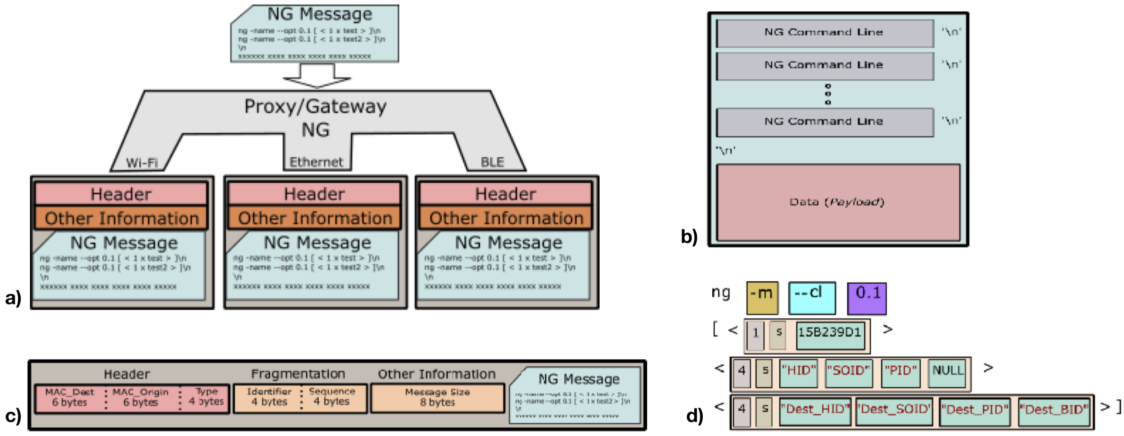

Figure 1. a) NG adaptation to link layer protocols. b) Structure of a NG message. c) NG encapsulated in Ethernet. d) Example of a routing command line.

3. Current Implementation

NovaGenesis messages are composed by Command Lines and payload, if any. They are ASCII coded. A Command Line is a text made in a single line, where each parameter is separated by a space. It can be divided into header and arguments. In the header there is an architecture identifier, the name, the alternatives and the version of a command. Every command begins with the lowercase "ng" characters, which identify the command as NovaGenesis. The next parameter of the header is the name of the command, preceded by a minus sign. The next one is the command alternatives parameter, preceded by two minus signs. And finally, the command version. The second part of the command is made up of arguments. This section is encapsulated by square brackets "[...]". There may be one or more arguments. Each argument is encapsulated by Lower and Greater signals "< >". The first parameter of the argument indicates the amount of elements that it contains. The second parameter indicates the type of elements. And finally, we have the elements themselves. Figure 1a) illustrates how a NG message is encapsulated in link layer protocols, while Figure 1b) depicts the format of a NG message. The current NG implementation is based in four services:

- Publish/Subscribe Service (PSS) - Services can publish or subscribe NBs/contents.
- Generic Indirection Resolution Service (GIRS) - Selects which hash table will store data.
- Hash Table Service (HTS) - Stores NBs and associated contents, if any.
- Proxy/Gateway/Controller Service (PGCS) - Provides (i) message encapsulation over established networking techs, such Ethernet; (ii) a proxy service to represent other NG services or physical resources; (iii) bootstrapping functionalities to initialize a domain. PGCS is able to add header fields, fragmentation data, and NG message size field. Figure 1c) illustrates a message ready to be sent via Wi-Fi.

4. Forwarding/Routing with Dual Names (FRDN)

F/R is based on NRNCS and two other components: (i) the gateway (GW), existing in every service; (ii) the proxy/gateway, implemented exclusively in PGCS. The GW provides internal service and OS limited inter service communication. GW employs an event-driven discipline to process messages according to scheduled times. The PG deals with inter-OS F/R. The first command line carries the information about the sender and receiver of the message. An example is provided in Figure 1d). It is made up of three arguments. The first one informs the SVN of the domain to which the sender device belongs to (15B239D1). The second argument consists of four SVNes for Hardware, OS, Process and Block (a service internal component), respectively. The third and final argument is also composed of four elements with the same meaning, however related to the destination service.

NovaGenesis F/R strategy employs SVNes alternatively as identifier and locators. For example, the identifier of a domain can be the locator of a host inside it. This is why we decided to call this novel approach as Forwarding/Routing with Dual Names (FRDN). The special conditions that validates a name as an identifier and/or locators is explored in [Alberti et al. 2017a]. Unique names in a scope can be employed as identifiers, while names that uniquely point to a position in a space can be used as locators[1]. What makes this FRNB idea viable are NovaGenesis name bindings (NBs). NBs can relate an ID to one or more locators. In this case, a semantic operator "is contained" is associated to the NB, i.e. the entity identified by an ID "is contained" in the position denoted by a certain locator. For instance, the photo "photo001.jpg" can have the hash "0x35ab0229" as its ID, while a service named "App01" can have the hash "0x90280273" as its ID. The NB $<$ 0x35ab0229, 0x90280273 $>$ together with the semantic operator "is contained" can represent the location of "photo001.jpg" at the service "App01". Observe that the ID of the service is employed as a locator for the photo. This explains the dual role of SVNes in NovaGenesis: IDs or locators. Depending on the semantic operator associated to a NB, a name can assume the role of ID or locator. Figure 2 illustrates the algorithm employed at the PGCS PG when F/R a message. The message HID is used to determine if the destination is local or an external node. If it is an external node, the PG block queries its local HT to determine the protocol stack related to this HID. This information is obtained during nodes initialization. Then, PG resolves the HID to the address of the next hop in the path to the destination. Finally, PG determines the socket to call in the OS. The approach proposed in this paper is unique. It resembles CCN [Jacobson et al. 2009], NetInf [Dannewitz et al. 2013] and XIA [Anand et al. 2011] approaches, however the main differences are: (i) it enables name-based routing for any architectural entity; (ii) it employs a dual SVN-based approach, which reuses ID as locators, saving storage resources; (iii) it allows F/R horizontally (among network nodes) and vertically (inside a node, internally to OS).

5. Experimental Proof-of-the-Concept

The experimental scenario is depicted in Figure 3. The aim is to proof NG that messages can be forwarded/routed using SVNes alternately as identifiers and locators. All computers run a PGCS service. NovaGenesis employs a initialization step (called "hello") to exchange Host Identifier (HID), operating system (OSID), process ID (PID), PG block ID (BID), among other information. These IDs are self-verifying names (depicted in yellow). The "hello" is broadcast at Ethernet level. The natural language name of each computer is in blue, while the Ethernet MAC address is in green. Figure 4 depicts a

[1] In this case, they need to provide a notion of distance to other locators.

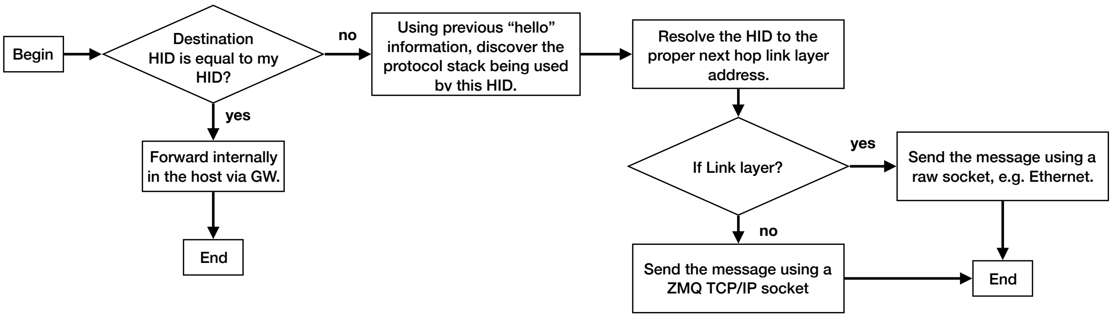

Figure 2. F/R algorithm implemented at the PGCS PG block.

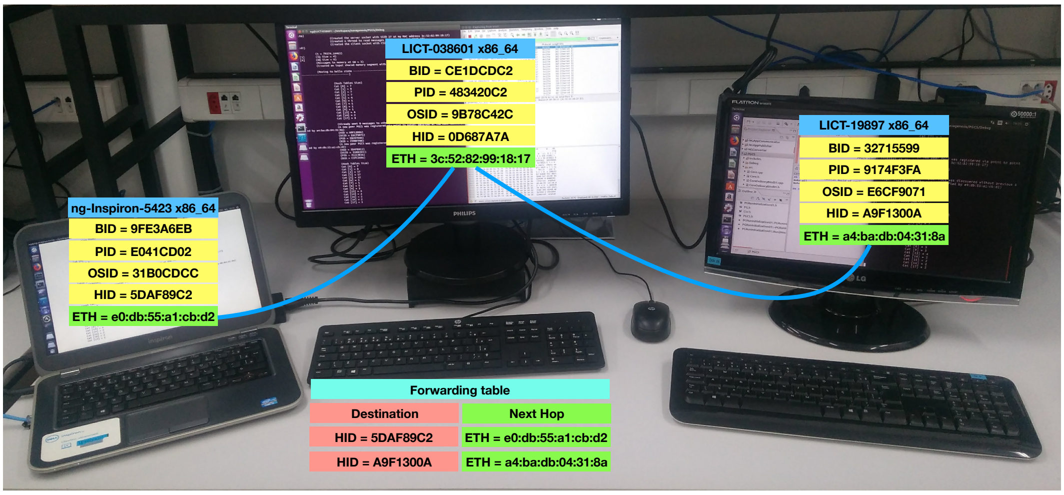

Figure 3. Experimental setup with NovaGenesis over Ethernet.

"hello" message sent by the PGCS in the right to the PGCS in middle. The first command line contains the source tuple of this PGCS, which is running at the "LICT-19897 x86_64" PC. The destination tuple is fulfilled with "empty", which means a broadcast "hello". The HID "A9F1300A" not only identifies the source computer, but also locates the OS named "E6CF9071". In other words, the "HID" defines a space where many OSs can inhabit. Observe also, that the message has a space limiter "15B239D1" which means that its routing is limited to the local domain. Figure 3 also shows a forwarding table build in the HT block of PGCS in the middle. Next hops are achieved by reading an NB with destination HID in the key and next hop in the value, i.e. $<$ HID, next hop MAC $>$. NovaGenesis can store NBs among any technology. Figure 5 shows the same "hello" message captured using Wireshark^{TM}.

```
ng -m --cl 0.1 [ < 1 s 15B239D1 > < 4 s A9F1300A E6CF9071 A585C4C2 BF169815 >
< 4 s empty empty empty empty > ]
ng -hello --ihc 0.1 [ < 9 s A9F1300A E6CF9071 9174F3FA A5B5C4C2 BF169815
54B77C78 Ethernet enp0s25 a4:ba:db:04:31:8a > ]
ng -scn --seq 0.1 [ < 1 s B79F1D60 > ]
```

Figure 4. A "hello" message broadcast from PGCS in the PC in the right.

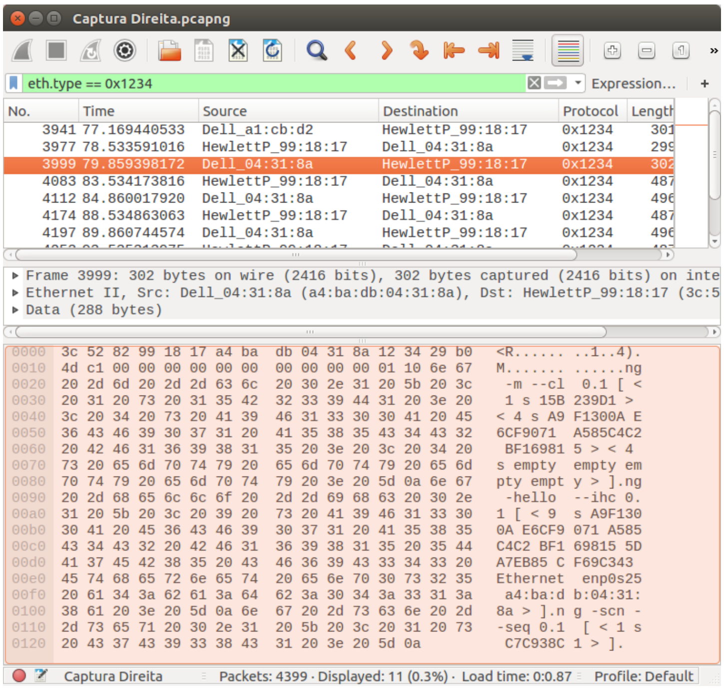

Figure 5. Wireshark™ print of the "hello" message.

6. Final Remarks

This paper presented an experimental proof-of-the-concept of a novel forwarding/routing approach based on name duality. Self-verifying names are employed not only as identifiers, but also as locators. Semantic operators (e.g. "is contained" or "contains") are employed to provide decoerence between ID or locator use case. Name bindings store the forward/routing information: (i) locally in every service (to provide message delivery inside an OS), in the PGCS for (for inter OS message exchanging); and (ii) in the distributed name resolution and network cache service (for domain level address resolution). Name bindings are in the formats: $<$ ID, Loc(s) $>$ or $<$ Loc, ID(s) $>$. Future work includes performance evaluation of the proposed solution in a large testbed and routing over alternative tuples (only HID, HID and OSID, etc.).

Acknowledgements

This work was partially supported by Finep, with resources from Funttel, Grant No. 01.14.0231-.00, under the Radiocommunication Reference Center (Centro de Referência em Radiocomunicações - CRR) project of the National Institute of Telecommunications (Instituto Nacional de Telecomunicações - Inatel), Brazil. The authors also thanks CGI/FAPESP Process 2015/24518-4, CAPES, FAPEMIG, CNPq and FINATEL.

References

Alberti, A. M., Casaroli, M. A. F., Singh, D., and da Rosa Righi, R. (2017a). Naming and name resolution in the future internet: Introducing the novagenesis approach. *Future Generation Computer Systems*, 67:163–179.

Alberti, A. M., Mazzer, D., Bontempo, M. M., de Oliveira, L. H., da Rosa Righi, R., and Sodré Jr., A. C. (2017b). Cognitive radio in the context of internet of things using a novel future internet architecture called novagenesis. *Computers & Electrical Engineering*, 57:147–161.

Anand, A., Dogar, F., Han, D., Li, B., Lim, H., Machado, M., Wu, W., Akella, A., Andersen, D. G., Byers, J. W., Seshan, S., and Steenkiste, P. (2011). Xia: An architecture for an evolvable and trustworthy internet. In *Proceedings of the 10th ACM Workshop on Hot Topics in Networks*, HotNets-X, pages 2:1–2:6, New York, NY, USA. ACM.

Dannewitz, C., Kutscher, D., Ohlman, B., Farrell, S., Ahlgren, B., and Karl, H. (2013). Network of information (netinf) - an information-centric networking architecture. *Comput. Commun.*, 36(7):721–735.

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, pages 1–12, New York, NY, USA. ACM.

Nordström, E., Shue, D., Gopalan, P., Kiefer, R., Arye, M., Ko, S., Rexford, J., and Freedman, M. J. (2012). Serval: An end-host stack for service-centric networking. In Gribble, S. D. and Katabi, D., editors, *NSDI*, pages 85–98. USENIX Association.

Ramirez, W., Masip-Bruin, X., Yannuzzi, M., Serral-Gracia, R., Martinez, A., and Siddiqui, M. (2014). A survey and taxonomy of id/locator split architectures. *Computer Networks*, 60:13 – 33.